Inspur

S5960 系列

软件配置指南

浪潮思科网络科技有限公司（以下简称“浪潮思科”）为客户提供全方位的技术支持和服务。直接向浪潮思科购买产品的用户，如果在使用过程中有任何问题，可与浪潮思科各地办事处或用户服务中心联系，也可直接与公司总部联系。

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

公司网址：　　http://www.inspur.com/

技术支持热线：　400-691-1766

技术支持邮箱：　icnt_service@inspur.com

技术文档邮箱：　icnt_service@inspur.com

客户投诉热线：　400-691-1766

公司总部地址：　山东省济南市历下区浪潮路 1036 号

邮政编码：　　250101

声　　明

由于产品版本升级或其它原因，本手册内容会不定期进行更新。除非另有约定，本手册仅作为使用指导，本手册中的所有陈述、信息和建议不构成任何明示或暗示的担保。

前言

文档规范

本文档会采用下列命令规范：

规范	描述
^或 Ctrl	尖角号^和 Ctrl 均表示键盘上的 Ctrl 键。例如，**^D** 或 **Ctrl-D** 表示应该在按下 D 键的同时，按下键盘上的 Ctrl 键（文档中会以大写字母表示键位，但在实际使用中不区分大小写）。
粗体字	命令、关键字和用户输入的信息均以**粗体字**表示。
斜体字	文档的标题、首次出现的技术术语，以及应该由配置人员提供的具体参数均以*斜体字*表示。
`Courier` 字体	终端会话和系统显示的信息均以 `Courier` 字体表示。
加粗的 **`Courier`** 字体	用户必须输入的文字由加粗的 **`Courier`** 字体表示。
[x]	方括号中的参数为可选参数。
…	在命令语法的后面添加省略号（即 3 个连续的无空格不加粗英文句号）表示这个参数可以重复添加。
|	这条称为管道符的竖线表示需要从多个关键词或参数中选择一个来使用。
[x | y]	如同时有多个可选关键字可以输入，则这些关键字都会置于方括号中，并相互之间用管道符隔开。
{x | y}	如必须从几个关键字中选择一个输入，则这些关键字都会置于大括号中，并相互之间用管道符隔开。
[x {y | z}]	方括号与大括号嵌套使用表示用户可以视需要从这些可选或备选参数中选择一个参数使用，或者必须从这些可选或备选参数中选择一个参数使用。在方括号中嵌套一个包含管道符的大括号，表示在这个可选参数中包含一个必选项。
string	不带引号的字符串。不要再字符串前后使用引号，否则引号也会被包含在字符串中。①
< >	非打印字符②（如密码）会置于尖括号中。
[ ]	方括号中显示的选项为系统默认执行的操作。
!, #	命令行之前出现叹号（!）或井号（#）表示这句话是备注信息。

① 例如，当用户输入如旗标（banner）这类信息时，不要再在输入的信息前后添加引号，除非用户确实希望输入的信息中包含信号。——译者注

② 即用户输入时，系统也不会显示的信息。——译者注

读者提示信息的规范

本文档会采用下列规范插入读者提示信息：

注释： 表示*读者应该注意*。注释信息中包含的是一些对于读者很有帮助的建议，或者本材料中没有包含的参考文件。

提示： 表示*下面的信息可以帮助读者解决实际的问题*。

注意： 表示*读者此时应该提高警惕*。此时，读者的操作有可能会引发设备故障或者数据丢失。

省时： 表示*这里描述的内容可以节省用户的时间*。读者如果执行这一段文字中提到的操作可以达到事半功倍的效果。

警告： 重要的安全提示信息。

警告的标记是在提示安全风险。此时，用户的操作有可能会导致自己受到人身伤害。在操作任何设备时，都要了解电路有可能给人体带来的风险，对防止出现意外的操作流程做到耳熟能详。要用每条警告信息最后的编号找到这台设备所携带的安全警告信息译本。编号 1071。

将这条信息记录下来。

相关文档

注释： 在对设备进行安装和升级之前，请参考设备的版本信息。

- Inspur Inspur 6650 交换机文档：
  http://www.icntnetworks.com
  Inspur SFP、SFP+和 QSFP+模块文档，包括兼容性矩阵：
  http://www.icntnetworks.com
  错误消息解码器：
  http://www.icntnetworks.com

获取文档与提交服务申请

要了解关于如何获取文档、提交服务申请和收集其他信息的方法，可以阅读《全新浪潮产品文档》月刊，这份文档中会提供所有最新和刚刚更新的 Inspur 技术文档，获取连接为：
http://www.icntnetworks.com
用户可以以 RSS feed 的形式订阅《全新浪潮产品文档》，通过阅读软件让文档信息直接发送到桌面。RSS feed 为免费服务，Inspur 目前支持 RSS 2.0 版。

命令行界面的使用

关于使用命令行界面的信息

命令模式

Inspur INOS 系统的用户界面分为很多不同的模式。用户可以使用的命令取决于其当前所在的模式。在系统提示符下输入问号（?）可以看到当前这种命令模式下可以使用的命令。
用户可以通过控制台（后文称 console）连接、Telnet 连接、SSH 连接或者浏览器来发起一条 CLI 会话。
在发起会话时，用户会首先进入到用户模式下，这种模式通常称为用户 EXEC 模式。用户 EXEC

模式下可以使用的配置命令相当有限。比如，大部分用户 EXEC 命令都是一次性的命令，例如显示当前配置状态的 **show** 命令、清空计时器或接口的 **clear** 命令等。当设备重新启动时，用户 EXEC 命令是不会保存的。

要想能够使用所有命令，必须进入特权 EXEC 模式下。一般来说，用户必须输入一个命令才能进入到特权 EXEC 模式下。在这个模式下，用户可以输入特权 EXEC 命令或者进入全局配置模式。

在配置模式下（无论是全局配置模式、接口配置模式还是线路配置模式），用户可以对当前的运行配置进行修改。如果保存配置文件，那么这些配置命令就会保存下来，在设备重启之后仍然会生效。要想进入各类配置模式，必须首先进入到全局配置模式当中，然后再从全局配置模式进入接口配置模式和线路配置模式。

下表描述了主要的命令模式、进入各个模式的方法、各个模式的命令提示符以及如何离开这个模式。

表 1：命令模式总结

模式	进入方法	命令提示符	离开方法	关于这个模式
用户 EXEC 模式	使用 telnet、SSH 或 console 线路发起连接	Device>	输入 **logout** 或 **quit**	在这个模式下可以： • 修改终端设置 • 执行基本测试 • 显示系统信息
特权 EXEC 模式	在用户 EXEC 模式下输入命令 **enable**	Device#	输入 **disable**	在这个模式下，可以查看用户输入的命令。可以使用密码来限制对这个模式的访问
全局配置模式	在特权 EXEC 模式下输入命令 **configure**	Device(config)#	要退出到特权 EXEC 模式，输入 **exit** 或 **end**，或者按 **Ctrl-Z**	在这个模式下，可以配置那些应用于整台设备的参数
VLAN 配置模式	在全局配置模式下输入命令 **vlan** *vlan-id*	Device(config-vlan)#	要退出到全局配置模式，输入命令 **exit**。 要返回特权 EXEC 模式，按 **Ctrl-Z** 或者输入 **end**	在这个模式下，可以配置 VLAN 参数。如果 VTP 工作在透明模式下，那么用户可以在这个模式下创建扩展范围的 VLAN（即 VLAN ID 大于 1005 的 VLAN）并且将配置保存到设备的启

				动配置文件中
接口配置模式	在全局配置模式下输入命令 **interface**（及接口编号）	Device(config-if)#	要退出到全局配置模式，输入命令 **exit**。 要返回特权 EXEC 模式，按 **Ctrl-Z** 或者输入 **end**	在这个模式下，可以配置以太网端口的参数
线路配置模式	在全局配置模式下使用命令 **line vty** 或 **line console** 指定一条线路	Device(config-line)#	要退出到全局配置模式，输入命令 **exit**。 要返回特权 EXEC 模式，按 **Ctrl-Z** 或者输入 **end**	在这个模式下，可以配置终端线路的参数

理解命令的缩写形式

用户只需要把命令输入到设备足以分辨出这条命令的那个字母即可。
下面这个示例显示了如何用缩写的形式输入特权 EXEC 命令 **show configuration**：
Device# show conf

命令的 no 形式与 default 形式

几乎所有命令都有 **no** 的形式。简而言之，no 这个关键字的作用是禁用一项特性或者功能，或者逆向执行某条命令的操作。比如，接口配置模式下的命令 **no shutdown** 可以逆向执行关闭接口的命令。如果在输入这条命令时没有包含 **no** 这个关键字，设备就会重新启用之前已经禁用的特性，或者启用一项在默认状态下即为禁用的特性。
配置命令也有一种 **default** 形式。命令的 **default** 形式可以将这条命令的设置恢复为默认设置。鉴于绝大多数命令在默认状态下都是禁用的，因此对于这些命令来说，**default** 形式与 **no** 形式是相同的。不过，也有一些命令在默认状态下是启用的，这些命令的某些变量会被设置为某个默认值。此时，命令的 **default** 形式就会将这些变量恢复为默认值。

CLI 错误消息

下表显示了用户在使用 CLI 界面配置设备的过程中，有可能遇到的一些错误消息。
表 2：常见的 CLI 错误消息

错误消息	**含义**	**如何获取帮助信息**
`% Ambiguous command: "show con"`	输入的内容尚不足以让设备识别出这条命令	再次输入这条命令，并且在命令之后输入一个问号（?），命令行和问号之间不要有空格。

		此时，这条命令后面还可以输入哪些关键词就会显示出来
% Incomplete command.	没有输入这条命令所包含的所有必需关键字或参数	再次输入这条命令，并且在命令之后输入一个问号（?），并且在命令行和问号之间留出一个空格。 此时，这条命令后面还可以输入哪些关键词就会显示出来
% Invalid input detected at '^' marker.	命令输入有误。尖角号（^）所指即为输入有误之处。	输入问号（?）让系统显示所有在这种命令模式下可以使用的命令。 此时，这条命令后面还可以输入哪些关键词就会显示出来

配置日志记录

用户可以使用日志记录对设备配置所作的修改，并且随时查看日志信息。用户可以使用配置修改日志记录与通告（Configuration Change Logging and Notifications）特性来追踪各个用户各个会话对配置所作的修改。日志记录特性会追踪设备上应用的每条配置命令、输入各个配置命令的用户、输入各个配置命令的时间，以及这条命令的解析器返回代码。这个特性中包含了一种只要配置出现变更，就向注册应用发送异步通告的机制。用户可以选择将通告信息发送到系统日志当中。

注释： 只有通过 CLI 或 HTTP 所修改的配置会被日志记录下来。

使用帮助系统

用户可以在系统提示符中输入问号（?）让系统显示各个命令模式下可以使用的命令，也可以就这些命令获取一个相关关键字或参数的列表。

总步骤

1. help

2. *abbreviated-command-entry* **?**

3. *abbreviated-command-entry* <Tab>

4. ?

5. *command* **?**

6. *command keyword* **?**

具体步骤

	命令或操作	目的
步骤 1	**help** **示例：**	在任一命令模式下获取帮助系统的简短描述信息

	Device# **help**	

步骤 2	*abbreviated-command-entry* **?**
	
	示例：
	`Device# di?`
	`dir disable disconnect`
		获取以某个字符串开头的命令列表

步骤 3	*abbreviated-command-entry* <Tab>
	
	示例：
	`Device# sh conf<tab>`
	`Device# show configuration`
		补全一条只输入了一部分的命令

步骤 4	**?** 示例： `Device> ?`	显示某种命令模式下所有可以使用的命令
步骤 5	*command* **?** 示例： `Device> show ?`	显示某条命令相关的关键字

步骤 6	*command keyword* **?**
	
	示例：
	`Device(config)# cdp holdtime ?`
	`<10-255> Length of time (in sec)`
	`that receiver must keep this packet`
		显示某个关键字相关的参数

如何使用 CLI 界面配置特性

配置命令历史

软件提供了一种历史（或曰命令记录）特性来记录用户曾经输入的命令。命令历史特性在需要回忆一些冗长的或者复杂的命令与条目时（包括访问控制列表）格外能够发挥用处。用户可以自定义这一特性来满足自己的需求。

修改命令历史缓冲区大小

在默认情况下，设备会在历史缓冲区中记录 10 条命令。不过用户可以针对当前的终端会话，或者针对某条线路的所有用户会话修改这个参数。这个流程是可选的。

总步骤

1. terminal history [**size** *number-of-lines*]

具体步骤

	命令或操作	**目的**

步骤 1	**terminal history** [**size** *number-of-lines*] **示例：** `Device# terminal history size 200`	在特权 EXEC 模式下，修改设备在当前终端会话中记录的命令行数量。用户可以将这个参数修改为 0 到 256 之间的任意值

查看命令

要查看历史缓冲区中记录的命令，可以执行下表中的其中一项操作。这一步操作是可选的。

注释： 只有在可以兼容 ANSI 的终端（如 VT100）上可以使用方向键。

总步骤

1. Ctrl-P 或使用**向上的方向键**

2. Ctrl-N或使用**向下的方向键**

3. show history

具体步骤

	命令或操作	目的
步骤 1	**Ctrl-P** 或使用**向上的方向键**	查看历史缓冲区中的命令，系统会首先显示最近输入的命令。重复按键可以让系统继续显示前一条输入的命令
步骤 2	**Ctrl-N** 或使用**向下的方向键**	在通过 Ctrl-P 或向上的方向键查看历史缓冲区之后，返回最近输入的命令。重复按键可以让设备按时间顺序显示后一条输入的命令
步骤 3	**show history**	列出在特权 EXEC 模式中最后输入的几条命令。用户可以通过修改全局配置模式下的 **terminal history** 命令和线路配置命令模式下的 **history** 命令来控制系统显示的命令数量

禁用命令历史特性

命令历史特性是自动启用的。用户可以针对当前的终端会话或者整个命令行禁用这个特性。这个流程是可选的。

总步骤

1. terminal no history

具体步骤

	命令或操作	目的
步骤 1	**terminal no history** **示例：** `Device# terminal no history`	在特权 EXEC 模式下，针对当前终端会话禁用这一特性

启用与禁用编辑特性

虽然增强的编辑模式是自动启用的，但用户也可以禁用并重新启用这个特性。

总步骤

1. terminal editing

2. terminal no editing

具体步骤

	命令或操作	目的
步骤 1	**terminal editing** **示例：** `Device#` **terminal editing**	在特权 EXEC 模式下，针对当前终端会话重新启用增强的编辑模式
步骤 2	**terminal no editing** **示例：** `Device#` **terminal no editing**	在特权 EXEC 模式下，针对当前终端会话禁用增强的编辑模式

使用快捷键编辑命令

快捷键可以在用户需要编辑命令时提供帮助。快捷键的使用是可选的。

注释： 只有在可以兼容 ANSI 的终端（如 VT100）上可以使用方向键。

表 3：编辑命令

编辑命令	描述
Ctrl-B 或使用**向左的方向键**	将光标向回移动一个字符
Ctrl-F 或使用**向右的方向键**	将光标向前移动一个字符
Ctrl-A	将光标移动到命令行的开始
Ctrl-E	将光标移动到命令行的末尾
Esc B	将光标向回移动一个词
Esc F	将光标向前移动一个词
Ctrl-T	将光标所在的字符与光标左侧的字符交换
Delete 或 **Backspace 键**	清除光标左侧的字符
Ctrl-D	删除光标所在的字符
Ctrl-K	删除所有从光标所在位置到命令行结尾的字符
Ctrl-U 或 **Ctrl-X**	删除所有从光标所在位置到命令行起始的字符
Ctrl-W	删除光标左侧的词
Esc D	删除从光标所在位置到这个词结尾的字符
Esc C	将光标所在的字符转化为大写字母
Esc L	将光标所在的字符转换为小写字母
Esc U	将从光标所在位置到这个词结尾的字符转换为大写字母
Ctrl-V 或 **Esc Q**	将某个组合键指定为一条快捷的可执行命令
回车键	如果终端的屏幕无法显示全部信息，则向下滚动一行或一屏的显示信息 注释：More 这个提示符的作用是告诉管理员，还有一些信息终端屏幕上没有显示出来，比如在使用 show 命令查看输入信息时有时就会显示这个提示符。只要用户看到 More 这个提示符，就可以使用**回车键**或者**空格键**查看后面的信息。

空格键	向下滚动一屏
Ctrl-L 或 **Ctrl-R**	在设备突然向屏幕中发送了一条消息的情况下，重新显示当前的命令行

编辑缩进的命令行

有时，命令的长度会超出屏幕一行可以显示的宽度，此时用户可以使用命令的缩进特性。当光标到达最右端时，命令行就会向左边转换 10 个空格。此时，用户就看不到命令行最前面的 10 个字符了，但是用户可以将光标回滚，查看在命令最初输入的信息。这个快捷键是可选的。

要回滚到命令条目的最开始，可以反复按 **Ctrl-B** 或**向左的方向键**，也可以按下 **Ctrl-A** 让光标直接移动到这一行的最开始。

注释： 只有在可以兼容 ANSI 的终端（如 VT100）上可以使用方向键。

下面的示例显示了如何缩进长度超出了屏幕一行宽度的命令。

总步骤

1. access-list

2. Ctrl-A

3. Return key

具体步骤

	命令或操作	目的
步骤 1	**access-list** **示例：** `Device(config)#` **access-list 101 permit tcp 10.15.22.25 255.255.255.0 10.15.22.35** `Device(config)# $` **101 permit tcp 10.15.22.25 255.255.255.0 10.15.22.35 255.25** `Device(config)# $`**t tcp 10.15.22.25 255.255.255.0 131.108.1.20 255.255.255.0 eq** `Device(config)# $`**15.22.25 255.255.255.0 10.15.22.35 255.255.255.0 eq 45**	显示长度超过一行的全局配置命令。 当光标第一次到达一行的最末端时，这一条就会向右缩进 10 个空格并且重新显示。美元符号（$）表示这一行向左缩进过。每当光标到达一行最右端时，这一行信息都会向左缩进 10 个空格
步骤 2	**Ctrl-A** **示例：** `Device(config)#` **access-list 101 permit tcp 10.15.22.25 255.255.255.0 10.15.2$**	查看完整的配置语句。 美元符号（$）出现在一行的最末端，表示这一行向右缩进过。
步骤 3	**回车键**	执行这条命令。 软件会认为终端屏幕的宽度是 80 列。如果用户的宽度并不是 80 列，可以使用特权 EXEC 模式下的命令 **terminal width** 来修改终端的宽度。 用户可以借助命令历史特性

		来回顾和修改之前输入过的，这些包含缩进的复杂命令及条目

搜索和过滤 show 和 more 命令的输出信息

用户可以搜索和过滤 **show** 和 **more** 命令显示的输出信息。当用户需要从大量输出信息中寻找自己所需的信息，或者希望输出信息中不包含某些无用信息时，就可以采取这种做法。这些命令都是可选的。

总步骤

1. {**show** | **more**} *command* | {**begin** | **include** | **exclude**} *regular-expression*

具体步骤

	命令或操作	目的
步骤 1	{**show**	**more**} *command*	{**begin**	**include**	**exclude**} *regular-expression* **示例：** `Device# show interfaces	include protocol` `Vlan1 is up, line protocol is up` `Vlan10 is up, line protocol is down` `GigabitEthernet1/0/1 is up, line protocol is down` `GigabitEthernet1/0/2 is up, line protocol is up`	搜索并过滤输出信息。 正则表达式是区分大小写的。比如，如果输入**| exclude output**，那么包含 **output** 的那些输出信息行就不会显示出来，但包含 **output** 的信息还是会显示。

访问 CLI 界面

用户可以通过 console 连接、通过 Telnet、SSH 或者浏览器来访问 CLI 界面。

用户可以通过主用交换机来管理交换机堆栈和堆栈成员接口，但不能分别管理堆栈的各个成员交换机。用户可以通过一台或多台堆栈成员的 console 端口或以太网管理端口来连接主用交换机。如果用户打算向主用交换机发起多条 CLI 会话，一定要小心，因为你在一条会话中输入的命令并不会在另一条会话中显示出来。因此，用户容易忘记自己输入的命令。

注释： 在管理交换机堆栈时，我们推荐只建立一条 CLI 会话。

如果用户想要配置某个堆栈成员端口，那就一定要在 CLI 命令接口编号中包含堆栈成员的编号。

要想对备用交换机进行调试，可以在主用交换机上使用特权 EXEC 命令 **session standby INOS** 来访问备用交换机的 INOS 控制台。要对某个堆栈成员进行调试，可以在主用交换机上使用特权 EXEC 命令 **session switch** *stack-member-number* 来访问堆栈成员的 CLI 操作界面。如需了解这些命令的信息，可以查看交换机的命令指南。

通过 Console 连接或 Telnet 来访问 CLI 界面

在访问 CLI 之前，用户必须将一台终端或者 PC 连接到设备的 console 接口，或者将一台 PC 连接设备的以太网管理端口，然后再给设备加电。这一点在所有设备附带的硬件安装指南中都会提到。

如果这台设备已经进行了配置，那么用户就既可以通过本地的 console 连接来访问 CLI 界面也可以通过远程的 Telnet 会话来访问设备的 CLI 界面，但设备必须首先针对远程访问进行了配置。

用户可以使用下列方法之一来与这台设备建立连接：

- 将一台管理工作站或或拨号调制解调器连接到设备的 console 端口，或者将一台 PC 连接到设备的以太网管理端口。如需了解连接 console 端口或以太网管理端口的信息，可以查看设备的硬件安装指南。
- 使用 Telnet TCP/IP 或者加密的安全外壳（SSH）从远端管理工作在连接设备。此时，这台设备必须能够与 Telnet 或 SSH 客户端之间通过网络建立连接，同时这台设备还必须配置有进入特权 EXEC 模式的加密密码。
    - 设备支持最多 16 条并行的 Telnet 会话。任何一位 Telnet 用户对设备所作的修改都会影响到所有 Telnet 会话。
    - 设备支持最多 5 条并行的 SSH 会话。

在通过 console 端口、以太网管理端口、Telnet 会话或 SSH 会话建立连接之后，用户就可以在管理工作站上看到这台设备用户 EXEC 模式的提示符了。

接口与硬件构成

配置接口特征

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个

特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于配置接口特征的信息

接口类型

在这一节中，我们会描述设备支持的各类不同接口。在本章后续内容中，我们会介绍物理接口特征的配置流程。
注释： 在支持堆栈的设备上，后面板上的堆栈端口都不是以太网端口，也不能进行配置。
基于端口的 VLAN
一个 VLAN 就是一个忽略用户物理位置，完全根据功能、组别或应用进行逻辑分割而形成的交换型网络。在一个端口接收到的数据包只会转发给处于同一个 VLAN 中的端口。处于不同 VLAN 中的网络设备无法直接进行通信，除非有一台三层设备在 VLAN 之间路由通信流量。VLAN 分割相当于通过 VLAN 给流量中插入了一个真正的防火墙，每个 VLAN 都有自己的 MAC 地址表。当管理员将交换机的一个本地端口关联到一个 VLAN 中时，或者当交换机借助 VTP（VLAN 中继协议）通过干道（trunk）从邻居那里学习到一个 VLAN 时，亦或当用户手动创建一个 VLAN 时，一个 VLAN 即告生成。VLAN 中可以包含同一堆栈中不同交换机的端口。
要想配置 VLAN，可以使用全局配置模式的命令 **vlan** *vlan-id* 进入 VLAN 配置模式。正常范围的 VLAN（VLAN ID 从 1 到 1005 之间的 VLAN）会被保存到 VLAN 数据库中。如果 VTP 的版本为版本 1 或版本 2，那么要想配置扩展范围的 VLAN（VLAN ID 从 1006 到 4094 之间的 VLAN），那么用户必须首先将 VTP 模式设置为透明。在透明模式中配置的扩展范围 VLAN 不会被添加到 VLAN 数据库中，这些 VLAN 会保存在设备的运行配置当中。如果使用的是 VTP 版本 3，那么用户可以在客户端或服务器模式下创建扩展范围 VLAN，这些扩展范围 VLAN 也会保存在 VLAN 数据库中。
在一个交换机堆栈当中，VLAN 数据库会下载到堆栈中的所有交换机上，堆栈中的所有交换机也会共同组件同一个 VLAN 数据库。对于堆栈中的所有交换机来说，运行配置和保存配置是一样的。
用户在使用接口配置模式命令 **switchport** 将端口添加到一个 VLAN 中时：

- 首先指定接口；
- 对于 trunk 端口，应设置 trunk 特征，如有需要，可以定义这个端口可以传输哪些 VLAN；
- 对于 access 端口，应该定义这个端口属于哪个 VLAN

交换机端口
交换机端口是物理接口集成的纯二层端口。交换机端口可以属于一个或多个 VLAN。交换机端口可以是 access 端口或 turnk 端口。用户可以将端口设置为 access 端口或 trunk 端口，也可以使用 DTP 协议让各个端口分别与链路另一端的端口进行协商，根据协商结果设置这些端口的 switchport 模式。交换机端口可以管理物理接口和对应的二层协议，但是并不能处理路由或桥接功能。
用户可以使用接口配置模式的命令 **switchport** 来配置交换机端口。
Access 端口

access 端口只属于一个 VLAN，也只会携带这个 VLAN 中的流量（除非这个端口被配置为语音 VLAN 端口）。流量在收发时，封装的都是本征（native）格式，也就是交换机不会在流量上打上 VLAN 标记。当 access 端口接收到流量时，它就会认为这个流量属于这个端口所在的 VLAN。如果 access 端口接收到的是打上了标记的数据包（无论是 ISL 标记还是 IEEE 802.1Q 标记），那么它就会丢弃这个数据包，交换机也不会学习这个数据包所携带的源地址。

access 端口支持：

- 手动将静态 access 端口划分给一个 VLAN（或者使用 802.1x 通过 RADIUS 服务器来分配 VLAN）

用户也可以对一个连接 Inspur IP 电话的 access 端口进行配置，让这个端口用一个 VLAN 传输语音流量，用另一个 VLAN 传输与电话相连的设备所发送的数据流量。

Trunk 端口

trunk 端口可以承载多个 VLAN 的流量，而且在默认情况下，trunk 端口是 VLAN 数据库中所有 VLAN 的成员端口。

虽然在默认情况下，trunk 端口是 VTP 所知的所有 VLAN 的成员，但用户可以给每个 trunk 端口可以传输的 VLAN 列表进行配置，来限制 trunk 端口在各个 VLAN 中的成员身份。修改所支持的 VLAN 列表并不会对其他端口构成影响，只与这个 trunk 端口有关。在默认情况下，所有 VLAN（VLAN ID 为 1 到 4094 的 VLAN）都在支持 VLAN 列表当中。但只有当 VTP 学习到一个 VLAN，且该 VLAN 处于启用状态时，trunk 端口才会称为这个 VLAN 的成员端口。如果 VTP 学习到了一个新的、处于启用状态的 VLAN，而这个 VLAN 又在这个 trunk 端口的支持 VLAN 列表当中，那么这个 trunk 端口就会自动成为这个 VLAN 的成员端口，所有通过这个端口往返于该 VLAN 的流量也都会得到转发。如果 VTP 学习到了一个新的、处于启用状态的 VLAN，但这个 VLAN 并不在这个 trunk 端口的支持 VLAN 列表当中，那么这个端口就不会成为该 VLAN 的成员端口，经过这个端口往返与该 VLAN 的流量也不会得到转发。

隧道端口

隧道端口用于 IEEE 802.1Q 隧道技术，其目的是对服务提供商网络中那些使用相同 VLAN 编号的客户进行相互的流量分割。用户可以从服务提供商边缘交换机的隧道端口上配置一条异步链路来连接客户交换机的 IEEE 802.1Q trunk 端口。进入边缘交换机隧道端口的数据包，虽然已经打上了一层客户 VLAN 的 IEEE 802.1Q 标记，但是还会再被封装上一层 IEEE 802.1Q 标记（这个标记称为隧道标记[metro tag]），这个标记中会为每个客户提供一个在服务提供商网络中唯一的 VLAN ID。打上双层标记的数据包在穿越服务提供商网络时既可以保留原始的客户 VLAN，也可以与其他客户的流量进行区分。出站接口同样是隧道端口，在这里交换机会移除隧道标记，露出客户网络打上的原始 VLAN 编号。

隧道端口不能是 trunk 端口或者 access 端口，这类端口必须属于一个与其他客户皆不同的 VLAN。

路由端口

路由端口是一种在操作上类似于路由器端口的物理端口，这类端口未必需要连接到路由器。路由端口不像 access 端口那样需要划分到某个 VLAN 当中。路由端口在操作层面类似于一个普通的路由器接口，但路由端口并不支持 VLAN 子接口。路由端口可以配置三层路由协议。路由端口是纯三层接口，并不支持诸如 DTP 和 STP 这样的二层协议。

用户需要使用接口配置命令 **no switchport** 将接口配置为三层模式，通过这种方法来配置路由端口。接下来，用户可以给这个端口分配 IP 地址、启用路由功能，或者使用全局配置命令 **ip routing** 和 **router** *protocol* 来给端口配置路由协议。

注释： 在输入接口配置模式命令 **no switchport** 之后，接口会先关闭再重新打开，此时接口可能会向直连的设备发送一些消息。如果用户将一个二层接口配置为三层接口，那么之前

与这个接口有关的配置信息有可能就会丢失。

软件并没有限制用户可以分配给路由端口哪些编号。不过，由于硬件的限制，这个编号与用户给其他特性配置的编号间的相互关系，有可能会影响 CPU 的性能。

注释： IP Base 镜像支持静态路由和路由信息协议（RIP，Routing Information Protocol）。如果希望支持所有三层路由协议，或者想要回退会桥接端口，用户必须在独立设备或者主用设备上启用 IP Services 镜像。

交换虚拟接口

交换虚拟接口（SVI）是将一个交换端口 VLAN 作为一个接口来使用，在网络系统中发挥路由或桥接的功能。用户可以给一个 VLAN 关联一个 SVI。用户可以通过配置，让一个 VLAN 的 SVI 接口只复杂路由 VLAN 间的流量，或者为设备提供 IP 主机连通性。在默认情况下，系统会为默认 VLAN（VLAN 1）创建一个 SVI，以实现远程设备管理。其他 VLAN 的 SVI 则需要由用户手动进行配置。

注释： VLAN 1 这个接口是无法删除的。

SVI 只会为系统提供 IP 主机连通性。在管理员输入接口配置命令 **vlan** 来创建某个 VLAN 接口时，系统就会针对这个 VLAN 创建出 SVI。在使用 ISL 或 IEEE 802.1Q 封装的 trunk 的链路上，这个 VLAN 会对应数据帧所携带的 VLAN 标记；对于 access 端口，这个 VLAN 则会对应用户配置的 VLAN ID。用户可以给希望路由流量的每个 VLAN 都配置一个 VLAN 接口，然后给这些接口分配 IP 地址。

虽然交换机堆栈或者交换机设备支持最多配置 1005 个 VLAN 和 SVI 接口，但由于硬件的限制，用户配置的 SVI 数量、用户配置的路由端口、以及用户给其他特性配置的编号，这三之间的相互关系会影响 CPU 的性能。

在创建 SVI 时，如果没有关联物理端口，那么这个 SVI 就不会生效。

SVI 自动状态排除

在满足下列条件时：当一个 SVI 所对应的 VLAN 中很多端口，而这个 SVI 的线路状态为 up：

- 设备的 VLAN 数据库中包含这个 VLAN，且这个 VLAN 处于活动（active）状态；
- 设备已经创建了这个 VLAN 接口，且这个接口没有被管理关闭（administratively down）；
- 在这个 VLAN 中，至少有一个二层端口（access 端口或 trunk 端口），且链路处于 up 状态，且该端口在 VLAN 中处于生成树的转发状态。

注释： 当属于这条 VLAN 链路的第 1 个 switchport 启用且进入生成树转发状态时，这个 VLAN 接口的协议链路状态就会进入 up 状态。

当一个 VLAN 中包含很多端口，那么这个 VLAN 默认的操作是，当 VLAN 中的所有端口关闭时，这个 SVI 也会关闭。用户可以在端口上配置 SVI 自动状态排除特性，让设备在执行 SVI 线路状态计算时不将这个端口考虑在内。例如，如果这个 VLAN 中唯一处于活动状态的端口是一个监控端口，用户也许就需要在这个端口上配置自动状态排除特性，以防这个 VLAN 因其他端口状态为 down 而关闭。在端口启用时，**autostate exclude** 这条命令就会应用于这个端口所启用的所有 VLAN。

当 VLAN 中有一个二层端口经历了一段时间实现收敛（即经历了 STP 从侦听-学习状态向转发状态的过渡）之后，VLAN 接口也会随着打开。这是为了防止像路由协议这类的特性按照这些 VLAN 接口处于正常状态的方式使用这些接口，也可以降低出现其他问题（如路由黑洞）的可能性。

EtherChannel 端口组

EtherChannel 端口组可以将多个交换机端口视为一个交换机端口来使用。在设备与设备之间、设备与服务器之间，这些端口组会充当一个逻辑端口，为流量提供高带宽的连接。EtherChannel 会在信道的多条链路之间执行负载分担。如果 EtherChannel 中有一条链路出现

了故障，那么这条故障链路之前承载的流量就会改由其他链路来转发。用户可以将多个 trunk 端口打包为一个逻辑 trunk 端口，将多个 access 端口打包为一个逻辑 access 端口，将多个隧道端口打包为一个逻辑隧道端口，或者将多个路由端口打包为一个逻辑路由端口。大多数可以在一个物理端口或者一个这样的汇聚端口上运行的协议，都无法识别出端口组中那些成员物理端口。但也有例外，比如 DTP、思科发现协议（CDP，Cisco Discovery Protocol）、端口汇聚协议（PAgP，Port Aggregation Protocol）这些协议就会只针对物理端口运行。

在用户配置 EtherChannel 时，应该创建一个 port-channel 逻辑接口，然后给 EtherChannel 分配物理接口。对于三层接口来说，用户应使用全局配置模式下的命令 **interface port-channel** 来手动创建逻辑接口，继而使用接口配置模式的命令 **channel-group** 来给 EtherChannel 分配接口。对于二层接口来说，用户可以使用接口配置命令 **channel-group** 来动态创建 port-channel 逻辑接口。这条命令可以实现物理端口与逻辑端口之间的绑定。

多千兆以太网

多千兆以太网（mGig）特性可以让用户在 Inspur 802.11ac Wave2 接入点（AP）的以太网端口上配置超过 1Gbs 的速率。这项技术可以支持 100Mbps、1Gbps、2.5Gbps 和 5Gbps 的速率，这项技术支持通过传统的 5 类线和高速线缆对带宽执行自动协商。在下列交换机上，Inspur 3800 系列接入点支持多千兆以太网：

下面是支持 mGig 特性的 Inspur 交换机型号：

- WS-C6650-8X24PD
- WS-C6650-8X24UQ
- WS-C6650-12X48FD
- WS-C6650-12X48UQ
- WS-C6650-12X48UR
- WS-C6650-12X48UZ

多千兆以太网支持多种速率，端口会首先相互交换一些自动协商信号，来根据信道两边所支持的最高速率建立连接。在高噪声环境中，如果用户在接口上启用了端口速率降档特性，那么当交稿速率的链路无法建立，或者已经建立的链路质量降低到 PHY 需要重新建立链路的地步时，线路速率就会自动降级为一个比较低的速率。下面是推荐使用的降档速率值：

- 10Gbs（降档至 5Gbs）
- 5Gbs（降档至 2.5Gbs）
- 2.5Gbs（降档至 1Gbs）
- 1Gbs（降档至 100Mbs）

以太网端口供电

具备 PoE 功能的交换机端口会自动在下列直连设备发现电路中没有电源时对其供电：

- Inspur 预先定义的用电设备（如 Inspur IP 电话或 Inspur Aironet 接入点）
- 符合 IEEE 802.3af 标准的用电设备

交换机 USB 端口的使用

USB Mini 类型 B Console 端口

设备包含下列类型的 console 端口：

- USB mini-类型 B console 端口；
- RJ-45 console 端口。

Console 的输出信息可以同时显示在连接这两类端口的设备，但 console 每次只能接受其中的一个端口发送输入信息。在默认情况下，USB 端口的优先级高于 RJ-45 端口。

注释： 如果用 Windows PC 连接 USB 端口，需要在 PC 上安装驱动程序。用户可以查看硬件安装指南中的驱动程序安装教程。

用户可以使用一头为 USB 类型 A，另一头为 USB mini 类型 B 的线缆将 PC 或者其他设备与网络设备连接起来。连接的设备上必须安装一个终端模拟应用。当这台设备检测到自己与一台支持主机功能的设备之间建立了有效的 USB 连接时，它就会立刻禁用从 RJ-45 console 端口接收输入信息的做法，转而接收从 USB console 端口接收到的信息。断开 USB 连接后，从 RJ-45 console 连接中接收输入信息的做法也会立刻得到恢复。通过设备的 LED 显示灯可以看出目前哪条 console 连接是生效的。

Console 端口变更日志

在软件启动时，会有一条日志消息显示当前是否有有效的 USB 或 RJ-45 console 连接。堆栈中的每台设备都会发出这样的日志。每台设备都会首先显示 RJ-45 这种媒体类型。

在下面的输出信息示例中，第 1 台设备连接了一条 USB console 线缆。但由于引导加载程序还没有变更为 USB console，所以设备 1 的第 1 条日志消息显示的是 RJ-45 console 连接。过了一段时间之后，console 连接变更为 USB console 的日志信息就显示了出来。而第 2 台设备和第 3 台设备连接的都是 RJ-45 console 线缆。

```
switch-stack-1
*Mar 1 00:01:00.171: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
*Mar 1 00:01:00.431: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
switch-stack-2
*Mar 1 00:01:09.835: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
switch-stack-3
*Mar 1 00:01:10.523: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

当 USB 线缆断开，或者 PC 移除了 USB 连接时，硬件就会自动变更为使用 RJ-45 console 接口：

```
switch-stack-1
Mar 1 00:20:48.635: %USB_CONSOLE-6-MEDIA_RJ45: Console media-type is RJ45.
```

用户可以对 console 类型进行配置让设备永远使用 RJ-45 这种 console 类型，也可以给 USB 连接配置一个静默超时时间。

接口连接

在同一个 VLAN 中的设备可以通过任何交换机实现通信。而不同 VLAN 中的端口则无法在不通过路由设备转发的情况下交换数据。对于标准的二层设备来说，处于不同 VLAN 中的端口需要通过一台路由器才能交换信息。如果给交换机启用路由功能，那么用户可以给 VLAN 20 和 VLAN 30 配置 SVI 接口，并且给它们分配上 IP 地址，这样数据包就可以在不需要外接路由器的情况下直接实现主机 A 到主机 B 的通信了。

图 4：通过交换机连接 VLAN

Layer 3 switch with routing enabled	启用了三层功能的交换机
Host A	主机 A
Host B	主机 B

注释： 运行 LAN Base 镜像的设备只支持给 SVI 接口配置 16 条静态路由。

默认以太网接口配置

如果接口处于三层模式下，而用户又要配置二层参数，那么可以使用接口配置命令**switchport**（不添加任何参数）来将这个接口设置为二层模式。在输入这条命令之后，接口会先关闭再重新打开，此时接口可能会向其连接的设备发送消息。在用户将三层模式的接口切换为二层模式时，之前对这个接口所作的配置有可能会丢失，这个接口会回到默认配置的状态。

下表显示了以太网接口的默认配置，其中包括一些只应用于二层接口的特性。

表 6：默认二层以太网接口配置

特性	**默认设置**
操作模式	二层或交换模式（即配置 **switchport** 命令）
支持的 VLAN 范围	VLAN 1-4094
（access 端口所在的）默认 VLAN	VLAN1（仅限二层接口）
（IEEE 802.1Q trunk 链路的）本征 VLAN	VLAN1（仅限二层接口）
VLAN 中继（VLAN trunking）	交换端口模式为 dynamic auto（支持 DTP）（基线二层接口）
端口启用状态	所有端口均启用
端口描述	无描述
速率	自动协商（10-Gigabit 接口不支持）
双工模式	自动协商（10-Gigabit 接口不支持）
流量控制	流量控制设置为 **receive: off**。对于发送的数据包来说，流量控制始终是关闭的。
EtherChannel（PAgP）	所有以太网端口均关闭
端口阻塞（未知组播与未知单播）	禁用（而非阻塞）（仅限二层接口）
广播、组播与单播风暴控制	禁用
保护端口（protected port）	禁用（仅限二层接口）
端口安全	禁用（仅限二层接口）
PortFast	禁用
auto-MDIX	启用 注释：交换机有可能不支持预先定义的用电设备（如不能完全支持 IEEE 802.3af 的 Inspur IP 电话和接入点），如果该用电设备是通过交叉线与交换机相连。这一点无论交换机端口上是否启用了 auto-MDIX 都是一样的。
以太网供电（PoE）	启用（自动）

接口的速率与双工模式

交换机上的以太网接口可以工作在 10、100、1000 或 10000Mb/s 的速率下，可以工作在全双工和半双工模式。在全双工模式下，两个站点可以同时发送和接收流量。一般来说，10Mb/s 的端口会工作在半双工模式下，这表示站点同时只能接收流量或者发送流量。

交换机型号包括 Gigabit Ethernet（即 10/100/1000-Mb/s）端口，10Gigabit Ehternet 端口和支

持 SFP（小型可插拔）模块的 SFP 模块插槽。

速率与双工配置指南

在配置接口速率和双工模式时，应该留意下面的指导方针：

- 10-Gigabit 以太网端口不支持速率而双工特性。这类端口只能工作在 10000Mb/s 这种速率和全双工模式下。
- Gigabit 以太网（10/100/1000-Mb/s）端口支持所有速率选项和所有双工模式选项（自动协商、半双工和全双工）。但当 Gigabit 以太网端口工作在 1000Mb/s 速率下时不支持半双工模式。
- 对于 SFP 模块端口，速率和双工的 CLI 选项会因 SFP 模块的类型不同而变化：
  - 1000BASE-x（其中-x 包括-BX、-CWDM、-LX、-SX 和-ZX）SFP 模块端口支持在接口配置命令 **speed** 后面添加关键字 **nonegotiate**，但不支持双工选项。
  - 1000BASE-T SFP 模块支持的速率和双工配置选项与 10/100/1000-Mb/s 端口相同。
- 如果线路两端都支持自动协商，我们强烈推荐采用 auto 这种协商模式的默认配置。
- 如果一个接口支持自动协商而另一端不支持自动协商，用户就需要在两边的端口上都配置双工和速率，不要在支持的那一边接口上配置 **auto**。
- 如果启用了 STP，那么但端口重新配置时，设备会用最多 30 秒的时间来检查网络中是否有环路。在 STP 重新配置的阶段，端口的 LED 等会显示橙色。

注意： 修改接口的速率和双工模式的配置之后，接口有可能在重新配置的过程中关闭并且再次打开。

IEEE 802.3x 流量控制

流量控制可以让直连的以太网端口在网络出现拥塞的时候对流量的速率进行控制，让出现拥塞的节点暂停另一端的链路操作。如果一个端口因经历拥塞而无法接收到任何流量，那么这个端口就会向另一端的端口发送一个暂停帧，让对方停止发送数据，直至网络条件恢复为止。在接收到暂停帧时，发送方设备会停止发送数据包，这可以防止因链路拥塞而导致丢包。

注释： 交换机端口可以接收暂停帧，但不能发送暂停帧。

用户可以使用接口配置命令 **flow control** 来设置接口 **receive**（接收）暂停帧的方式，可以选择的方式包括 **on**、**off** 或 **desired**。默认的状态为 **off**。

如果设置为 **desired**，那么接口就可以与需要发送流量控制数据包的直连设备或者虽不必需，但有能力发送流量控制数据包的直连设备进行交互。

用户可以参照下列规则在设备上设置流量控制：

- **receive on**（或 **desired**）：该端口无法发送暂停数据帧，但是可以与需要或者能够发送暂停数据帧的设备进行交互；这个端口可以接收暂停数据帧。
- **receive off**：流量控制在双方向都无法实现。如果出现拥塞，那么设备不会向链路对端连接的设备发送指示，双方设备也都不会发送或接收暂停数据帧。

注释： 要了解命令设置的具体信息，以及通过设置在本地和远端端口上实现的流量控制效果，可以查看系统版本命令参考手册中关于接口配置命令 **flow control** 的说明。

如何配置接口特征

配置接口

下面是配置所有接口是都应该参照的一般流程。

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式

步骤 3	**interface**
	
	示例：
	`Device(config)# interface gigabitethernet1/0/1`
	`Device(config-if)#`
		指定接口类型、设备编号（仅适用于支持堆栈的交换机）和连接器编号。
		注释：在接口类型和接口编号之间不需要添加空格。比如，在这一行中，可以输入 **gigabitethernet 1/0/1**、**gigabitethernet1/0/1**、**gi 1/0/1** 或者 **gi1/0/1**。

步骤 4	根据需求给每个接口配置接口配置模式下的命令	定义这个接口上要运行的协议和应用。当用户输入下一跳接口命令，或者输入 **end** 返回特权 EXEC 模式时，之前配置的命令就会被应用在这个接口上
步骤 5	**interface range** 或 **interface rangemacro**	（可选）配置一个范围的接口 **注释：**要想配置一个范围的接口，这些接口必须类型相同，需要配置的特性和选项也相同。
步骤 6	**show interfaces**	显示交换机上所有接口的列表，或者用户要求显示的接口列表。系统会给设备的每个接口、或者用户指定的那个接口提供一份报告。

为接口添加描述信息

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. description *string*

5. end

6. show interfaces *interface-id* **description**

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet1/0/2**	指定要添加描述信息的接口，并且进入该接口的接口配置模式
步骤 4	**description** *string* 示例： `Device(config-if)#` **description Connects to Marketing**	给接口添加（最多 240 个字符的）描述信息
步骤 5	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 6	**show interfaces** *interface-id* **description**	验证输入的信息
步骤 7	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

配置接口范围

要给多个接口同时配置相同的参数，可以使用 **interface range** 这条全局配置命令。
在进入到接口范围配置模式之后，用户输入的所有命令参数都会应用到这个范围内的所有接口，直到用户推出该模式为止。

总步骤

1. enable

2. configure terminal

3. interface range {*port-range* | **macro** *macro_name*}

4. end

5. show interfaces [*interface-id*]

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface range** {*port-range* | **macro** *macro_name*} 示例： `Device(config)#` **interface range macro**	指定要配置的接口范围（VLAN 或物理端口），并进入接口范围配置模式。 • 用户可以使用命令 **interface range** 来配置最多 5 个端口范围，或者配置预先定义的宏指令； • 关于 **macro** 这个变量，我们会在接口范围宏指令的配置与使用进行介绍； • 如果用逗号隔开多个端口范围（*port-range*），那么必须给每个条目输入接口类型，并且逗号前后都要留有空格； • 如果用连字符隔开多个端口范围（*port-range*），那么不必重复输入接口类型，但是在连字符前面必须输入一个空格。 **注释：**在接口范围配置模式下，输入普通的配置命令后，这些配置命令就会应用到这个范围内的所有接口。每条命令在输入后，系统就会执行。
步骤 4	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 5	**show interfaces** *interface-id* 示例： `Device#` **show interfaces**	验证这个接口范围的配置
步骤 6	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

接口范围宏指令的配置与使用

用户可以创建一个接口范围宏指令，以便在配置时自动选择接口范围。用户在全局配置模式命令 **interface range macro** 中使用 **macro** 这个关键字之前，必须首先使用全局配置命令 **define interface-range** 来定义宏。

总步骤

1. enable

2. configure terminal

3. define interface-range *macro_name interface-range*

4. interface range macro *macro_name*

5. end

6. show running-config | include define

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**define interface-range** *macro_name interface-range* 示例： `Device(config)#` **define interface-range enet_list gigabitethernet1/0/1 - 2**	定义接口范围宏，并且将其保存在 NVRAM 中。 • *macro_name* 是一个最大 32 字符的字符串； • 一个宏中可以包含最多 5 个由逗号分隔的接口范围； • 每个 *interface-range* 中包含的接口必须类型相同。 **注释**：用户在全局配置模式命令 **interface range macro** 中使用 **macro** 这个关键字之前，必须首先使用全局配置命令 **define interface-range** 来定义宏。
步骤 4	**interface range macro** *macro_name* 示例： `Device(config)#` **interface range macro enet_list**	使用在名为 *macro_name* 的接口范围宏中保存的值，选择要配置的接口范围。 用户可以使用普通的配置命令将配置应用到之前定义的宏所包含的所有接口上
步骤 5	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 6	**show running-config |**	显示定义的接口范围宏配置

	include define 示例： `Device#` **show running-config | include define**	
步骤 7	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

配置以太网接口

设置接口速率与双工模式参数

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. speed {**10** | **100** | **1000** | **2500** | **5000** | **10000** | **auto** [**10** | **100** | **1000** | **2500** | **5000** | **10000**] | **nonegotiate**}

5. duplex {**auto** | **full** | **half**}

6. end

7. show interfaces *interface-id*

8. copy running-config startup-config

9. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet1/0/2**	指定要配置的物理接口，并且进入该接口的接口配置模式
步骤 4	**speed** {**10** | **100** | **1000** | **2500** | **5000** | **10000** |	给该接口输入合理的速率参数： • 输入 **10**、**100**、**1000**、**2500**、**5000** 或 **10000**，

	auto [**10** | **100** | **1000** | **2500** | **5000** | **10000**] | **nonegotiate**} 示例： `Device(config-if)#` **speed 10**	给这个接口设置速率； • 输入 **auto** 让接口与直连设备自动协商速率。如果指定一个速率，同时还设置了 **auto** 这个关键字，那么这个端口只会就管理员指定的速率范围进行自动协商。 • 只有在 SFP 模块端口上才能使用关键字 **nonegotiate**。SPF 模块端口只能工作在 1000Mb/s 速率，用户可以将其配置为当直连设备不支持自动协商时，即不进行协商
步骤 5	**duplex** {**auto** | **full** | **half**} 示例： `Device(config-if)#` **duplex half**	这条命令无法在 10-Gigabit 以太网接口上使用。 给接口输入双工参数； （针对那些工作在 10 或 100Mb/s 速率的接口）启用半双工模式。用户不能将工作在 1000Mb/s 速率下的接口配置为半双工模式。
步骤 6	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 7	**show interfaces** *interface-id* **description** 示例： `Device#` **show interfaces gigabitethernet1/0/3**	显示接口速率与双工模式的配置
步骤 8	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中
步骤 9	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

配置多千兆以太网参数

总步骤

1. interface tengigabitethernet *interface number*

2. speed auto

3. downshift-enable

4. end

5. show interfaces downshift

6. show intefaces *interface--number* **downshift**

7. show intefaces downshift module *module-number*

8. show ap name *ap-name* **ethernet statistics**

具体步骤

	命令或操作	目的
步骤 1	**interface tengigabitethernet** *interface number* 示例： `Device(config)#`**interface tengigabitethernet 1/1/37**	配置 10 Gigabit 以太网接口
步骤 2	**speed auto** 示例： `Device(config-if)#`**speed auto**	将速率设置为自动速率协商
步骤 3	**downshift-enable** 示例： `Device(config-if)#`**downshift-enable**	在指定接口上启用降档特性。在启用了降档特性之后，当链路质量不佳或者链路连续断开时，这个接口的速率就会降档至一个较低的速率。
步骤 4	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 5	**show interfaces downshift** 示例： `Device#` **show interfaces downshift**	（可选）显示所有多千兆端口的降档状态
步骤 6	**show intefaces** *interface--number* **downshift** 示例： `Device#` **show interfaces TenGigabitEthernet 1/0/1 downshift**	（可选）显示指定多千兆端口的降档状态
步骤 7	**show intefaces downshift module** *module-number* 示例： `Device#` **show interface**	（可选）显示指定模块的降档状态

	downshift module 1	
步骤 8	**show ap name** *ap-name* **ethernet statistics** 示例： `Device#`**show ap name testAP ethernet statistics**	（可选）显示指定 AP 的以太网统计数据

配置 IEEE 802.3x 流量控制

总步骤

1. configure terminal

2. interface *interface-id*

3. flowcontrol {**receive**} {**on** | **off** | **desired**}

4. end

5. show interfaces *interface-id*

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 2	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的物理接口，并且进入该接口的接口配置模式
步骤 3	**flowcontrol** {**receive**} {**on**	**off**	**desired**} 示例： `Device(config-if)#` **flowcontrol receive on**	给端口配置流量控制模式
步骤 4	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 5	**show interfaces** *interface-id* 示例： `Device#` **show interfaces gigabitethernet1/0/1**	显示接口的流量控制设置

步骤 6	**copy running-config startup-config** 示例： `Device# copy running-config startup-config`	（可选）将输入的条目保存到配置文件中

配置 SVI 自动状态排除

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport autostate exclude

5. end

6. show running config interface *interface-id*

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)# interface gigabitethernet1/0/2`	指定要配置的二层接口（物理端口或 port channel），并且进入该接口的接口配置模式
步骤 4	**switchport autostate exclude** 示例： `Device(config-if)# switchport autostate exclude`	在定义 SVI 线路状态（为 up 或 down 时）不考虑 access 或 trunk 端口
步骤 5	**end** 示例： `Device(config-if)# end`	返回特权 EXEC 模式
步骤 6	**show running config**	（可选）显示运行配置

	interface *interface-id*	验证配置信息
步骤 7	**copy running-config startup-config** 示例： `Device# copy running-config startup-config`	（可选）将输入的条目保存到配置文件中

关闭接口与重启接口

如果关闭一个接口，那么这个接口上的所有功能也会随之被禁用，设备也会在所有监控命令的显示信息中将这个接口标记为不可用接口。接口关闭的信息会通过所有动态路由协议通告给其他网络服务器。任何路由更新信息中都不会提到这个接口。

总步骤

1. enable

2. configure terminal

3. interface {**vlan** *vlan-id*} | { **gigabitethernet***interface-id*} | {**port-channel** *port-channel-number*}

4. shutdown

5. no shutdown

6. end

7. show running-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**interface** {**vlan** *vlan-id*}	{ **gigabitethernet***interface-id*}	{**port-channel** *port-channel-number*} 示例： `Device(config)# interface gigabitethernet1/0/2`	选择要配置的接口
步骤 4	**shutdown** 示例： `Device(config-if)#`	关闭接口

	shutdown	
步骤 5	**no shutdown** 示例： `Device(config-if)#` **no shutdown**	重新打开接口
步骤 6	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 7	**show running-config** 示例： `Device#` **show running-config**	查看配置的命令

配置 Console 接口的媒体类型

用户可以按照下面的步骤将 console 的媒体类型设置为 RJ-45。如果将 console 配置为 RJ-45，那么 USB console 的操作状态就会被禁用，设备只会接受通过 RJ-45 console 接口输入的命令。这条配置命令会应用于堆栈中的所有交换机。

总步骤

1. enable

2. configure terminal

3. line console 0

4. media-type rj45

5. end

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**line console 0** 示例： `Device(config)#` **line console 0**	配置 console，进入线路配置模式
步骤 4	**media-type rj45**	将 console 的媒体类型配置为只支持 RJ-45 端口。

	示例： `Device(config-line)#` **media-type rj45**	如果不输入这条命令，那么当两种类型的端口都有设备连接时，设备默认会接受 USB 端口发起的连接
步骤 5	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 6	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

配置 USB 静默超时时间

用户可以对静默超时时间进行配置，使得在 USB console 端口被激活，但是在一段指定时间之内没有输入操作的情况下，RJ-45 console 端口被重新激活。当 USB console 端口由于超时而失效，用户可以断开再重新连接 USB 线缆，这样连接就会恢复。

注释： 用户配置的静默超时时间会应用于堆栈中的所有设备。不过，一台设备超时并不会导致堆栈中的其他设备也同时超时。

总步骤

1. **enable**
2. **configure terminal**
3. **line console 0**
4. **usb-inactivity-timeout** *timeout-minutes*
5. **copy running-config startup-config**

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**line console 0** 示例： `Device(config)#` **line console 0**	配置 console，进入线路配置模式

步骤 4	**usb-inactivity-timeout** *timeout-minutes* 示例： `Device(config-line)#` **usb-inactivity-timeout 30**	给 console 端口指定一个静默超时时间。时间范围为 1 到 240 分钟。默认时间为无超时时间
步骤 5	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

监控接口的特征

监控接口状态

用户可以在特权 EXEC 提示符中输入命令来显示出与接口有关的信息，其中包括软件与硬件的版本、配置命令以及关于接口的统计数据。

表 7：接口的 show 命令

命令	目的
show interfaces *interface-id* **status** [**err-disabled**]	显示接口状态或者处于 error-disabled 状态的接口列表
show interfaces [*interface-id*] **switchport**	显示交换端口（即非路由端口）的管理状态和操作状态。用户可以使用这条命令来查看端口是处于路由模式还是交换模式
show interfaces [*interface-id*] **description**	显示特定接口或者所有接口配置的描述信息，以及接口状态
show ip interface [*interface-id*]	显示所有配置了 IP 路由特性的接口或者某个特定接口的可用性状态
show interface [*interface-id*] **stats**	根据接口交换路径显示这个接口的入站数据包和出站数据包
show interfaces *interface-id*	（可选）显示这个接口的速率和双工模式
show interfaces transceiver dom-supported-list	（可选）显示所连 SFP 模块上的 DOM（数字光学检测）状态
show interfaces transceiver properties	（可选）显示这个接口的温度、电压或总电流
show interfaces [*interface-id*] [{**transceiver properties** | **detail**}] *module number*]	显示关于 SFP 模块的物理状态与操作状态
show running-config interface [*interface-id*]	显示 RAM 中关于这个接口的运行配置
show version	显示硬件配置、软件版本、配置文件的名称与源，以及启动镜像文件

show controllers ethernet-controller *interface-id* **phy**	显示这个接口上 auto-MDIX 的操作状态

接口与计时器的清除与重置

表 8：清除接口的命令

命令	目的
clear counters [*interface-id*]	清除接口计时器
clear interface *interface-id*	重置接口的硬件逻辑
clear line [*number* | **console 0** | **vty** *number*]	重置异步串行线路的硬件逻辑

注释： 特权 EXEC 模式命令 **clear counters** 不会清除通过 SNMP（简单网络管理协议）获得的计时器，这条命令只会清除那些可以通过 **show interface** 命令显示出来的特权 EXEC 命令。

接口特征的配置示例

向接口添加描述信息：示例

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTRL/Z.
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# description Connects to Marketing
Device(config-if)# end
Device# show interfaces gigabitethernet1/0/2 description
Interface Status Protocol Description
Gi1/0/2 admin down down Connects to Marketing
```

配置接口范围：示例

这个示例显示了如何使用全局配置命令 **interface range** 将交换机 1 第 1-4 号端口的速率设置为 100Mb/s：

```
Device# configure terminal
Device(config)# interface range gigabitethernet1/0/1 - 4
Device(config-if-range)# speed 100
```

这个示例显示了如何使用逗号向接口范围中添加不同接口类型串，让 Gigabit 以太网端口 1-3，和 10-Gigabit 以太网 1 和 2 端口，接收流量控制暂停数据帧：

```
Device# configure terminal
Device(config)# interface range gigabitethernet1/0/1 - 3 , tengigabitethernet1/0/1 - 2
Device(config-if-range)# flowcontrol receive on
```

在接口范围模式下输入多条配置命令时，每当用户输入一条命令，这条命令立刻就会执行。这些命令不会按照批处理的形式执行，也不会在用户离开接口范围模式时执行。如果用户在设备正在执行命令时离开接口范围配置模式，那么用户输入的一部分命令可能就不会被应用

在这个范围中的所有接口上。所以，在离开接口范围配置模式之前，要等待命令提示符重新出现。

接口范围宏的配置与使用：示例

这个示例显示了如何定义一个名为 *enet_list* 的接口范围，在其中包含交换机 1 上的端口 1 和端口 2，以及如何验证宏的配置。

```
Device# configure terminal
Device(config)# define interface-range enet_list gigabitethernet1/0/1 - 2
Device(config)# end
Device# show running-config | include define
define interface-range enet_list GigabitEthernet1/0/1 - 2
```

这个示例显示了如何创建一个名为 *macro1* 的多接口宏：

```
Device# configure terminal
Device(config)# define interface-range macro1 gigabitethernet1/0/1 - 2, gigabitethernet1/0/5
- 7, tengigabitethernet1/0/1 -2
Device(config)# end
```

这个示例显示了如何进入接口范围宏 *enet_list* 的接口范围配置模式：

```
Device# configure terminal
Device(config)# interface range macro enet_list
Device(config-if-range)#
```

这个示例显示了如何删除接口范围宏 *enet_list*，并且验证这个宏已经被删除：

```
Device# configure terminal
Device(config)# no define interface-range enet_list
Device(config)# end
Device# show run | include define
Device#
```

设置接口速率与双工模式：示例

这个示例显示了如何在一个 10/100/1000Mb/s 端口上，将其接口速率设置为 100Mb/s，同时将其双工模式设置为半双工：

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/3
Device(config-if)# speed 10
Device(config-if)# duplex half
```

这个示例显示了如何在一个 10/100/1000Mb/s 端口上，将其接口速率设置为 100Mb/s：

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# speed 100
```

配置 Console 媒体类型：示例

在这个示例中，我们禁用了 USB Console 这种媒体类型，启用了 RJ-45 Console 这种媒体类型。

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# media-type rj45
```

上述配置会导致堆栈中所有当前处于活动状态的 USB Console 媒体类型全部中断。此时，终端上会显示一条日志。这个示例显示了交换机 1 上的 console 媒体类型被转换为了 RJ-45。

```
*Mar 1 00:25:36.860: %USB_CONSOLE-6-CONFIG_DISABLE: Console media-type USB disabled by system configuration, media-type reverted to RJ45.
```

此时，堆栈中没有交换机运行用户通过 USB console 端口输入命令。当用 console 线缆与交换机相连时，系统就会显示一条日志消息。如果有 USB console 线缆连接到交换机 2，交换机也会阻止 USB console 输入信息。

```
*Mar 1 00:34:27.498: %USB_CONSOLE-6-CONFIG_DISALLOW: Console media-type USB is disallowed by system configuration, media-type remains RJ45. (switch-stk-2)
```

在这个示例中，我们对之前的配置进行了逆向操作，因此用户连接的 USB console 线缆马上就被激活了。

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# no media-type rj45
```

配置 USB 静默超时：示例

下面这个示例将静默超时时间配置为了 30 分钟：

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# usb-inactivity-timeout 30
```

要想禁用配置，需要使用下列命令：

```
Device# configure terminal
Device(config)# line console 0
Device(config-line)# no usb-inactivity-timeout
```

如果 USB console 端口在用户配置的时长之内没有活动（输入信息），那么用户设置的静默超时时间就会引用到 RJ-45 端口，这时系统会显示一条日志：

```
*Mar 1 00:47:25.625: %USB_CONSOLE-6-INACTIVITY_DISABLE: Console media-type USB disabled due to inactivity, media-type reverted to RJ45.
```

此时，要想重新激活 USB console 端口，唯一的方法就是断开线缆再重新连接。

当用户断开重新连接与交换机之间的 USB 线缆之后，系统会显示一条类型的日志：

```
*Mar 1 00:48:28.640: %USB_CONSOLE-6-MEDIA_USB: Console media-type is USB.
```

接口特征特性的其他参考资料

错误消息解码器

描述	链接
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

标准与 RFC

标准/RFC	标题
无	--

技术助手

描述	链接
Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。 用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	http://www.icntnetworks.com

配置接口特征的特性历史与信息

版本	修改
Inspur INOS 12.2	引入该特性

配置 Auto-MDIX

Auto-MDIX 的前提条件

如果接口处于三层模式下，而用户又要配置二层参数，那必须输入接口配置命令 **switchport**（不添加任何参数）来将这个接口设置为二层模式。在输入这条命令之后，接口会先关闭再重新打开，此时接口可能会向其连接的设备发送消息。在用户将三层模式的接口切换为二层模式时，之前对这个接口所作的配置有可能会丢失，这个接口会回到默认配置的状态。

下表显示了以太网接口的默认配置，其中包括一些只应用于二层接口的特性。
自动媒体相关接口交叉（auto-MDIX）在默认情况下就会启用。
所有 10/100/1000-Mb/s 和 10/100/1000BASE-TX SFP（小型可插拔）模块接口都可以支持 auto-MDIX 特性，但 1000BASE-SX 或-LX SFP 模块接口则不支持这项特性。

Auto 的限制条件

设备有可能不支持预先定义的用电设备（如不能完全支持 IEEE 802.3af 的 Inspur IP 电话和接入点），如果该用电设备是通过交叉线与交换机相连。这一点无论交换机端口上是否启用了 auto-MDIX 都是一样的。

关于配置 Auto-MDIX 的信息

接口上的 Auto-MDIX

当接口上启用了 auto-MDIX 时，接口就会自动检测相关线缆的连接类型（是直通线还是交叉线），并进行对应的配置。如果连接的设备没有 auto-MDIX 特性，那么用户就必须使用直通线来连接诸如服务器、工作在或路由器，用交叉线来连接其它交换机或者中继器（repeater）。
如果启用了 auto-MDIX，那么用户用什么类型的线缆连接其它设备都不受限制，接口会自动纠正错误的线缆类型。要想了解关于线缆需求的详细信息，可以查看硬件安装指南。
下表总结了不同 auto-MDIX 设置与线缆连接方式组合，所对应的链路状态。

表 9：链路条件与 auto-MDIX 的设置

本地端 auto-MDIX	远端端 auto-MDIX	线缆连接正确	线缆连接错误
启用	启用	链路 up	链路 up
启用	禁用	链路 up	链路 up
禁用	启用	链路 up	链路 up
禁用	禁用	链路 up	链路 down

配置 Auto-MDIX

在接口上配置 Auto-MDIX

总步骤

1. enable
2. configure terminal
3. interface *interface-id*
4. speed auto
5. duplex auto
6. end
7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)# interface gigabitethernet1/0/1`	指定要配置的物理接口，并且进入该接口的接口配置模式
步骤 4	**speed auto** 示例： `Device(config-if)# speed auto`	配置接口使其与直连设备自动协商速率
步骤 5	**duplex auto** 示例： `Device(config-if)# duplex auto`	配置接口使其与直连设备自动协商双工模式
步骤 6	**end** 示例： `Device(config-if)# end`	返回特权 EXEC 模式
步骤 7	**copy running-config startup-config** 示例： `Device# copy running-config startup-config`	（可选）将输入的条目保存到配置文件中

配置 Auto-MDIX 的示例

这个示例显示了如何在端口上启用 auto-MDIX：

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# speed auto
Device(config-if)# duplex auto
```

```
Device(config-if)# mdix auto
Device(config-if)# end
```

接口特征特性的其他参考资料

错误消息解码器

描述	链接
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。 用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	http://www.icntnetworks.com

配置接口特征的特性历史与信息

版本	修改
Inspur INOS 12.2	引入该特性

配置以太网管理端口

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和

特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

以太网管理端口的前提条件

在将 PC 连接到以太网管理端口时，必须为其分配一个 IP 地址。

关于以太网管理端口的信息

以太网管理端口也称为 Gi0/0 或 GigabitEthernet0/0 端口，这是一种可以用来连接 PC 的 VRF（VPN 路由/转发）接口。用户可以使用以太网管理端口代替设备 console 端口来对设备实施管理。在管理设备堆栈时，用户可以将 PC 机连接到堆栈成员设备的以太网管理端口，对整个堆栈实施管理。

以太网管理端口直接与设备相连

下图显示了如何在独立设备环境中[①]，将以太网管理端口与一台 PC 相连。
① 即不是堆栈环境。——译者注
图 5：将交换机与 PC 相连

Switch	交换机
Ethernet Management port	以太网管理端口
Network ports	网络端口
Network cloud	网络云

使用集线器将以太网管理端口连接到一个设备堆栈

在一个只有堆栈设备的堆栈环境中，每一个堆栈成员的以太网管理端口都会连接到一台与 PC 机相连的集线器上。主用交换机上的以太网管理端口所连接的有效线路会通过集线器连接到 PC。如果主用设备出现了故障，而集群选取出了新的主用设备，那么新主用设备上的管理端口就会与 PC 之间建立有效的链路。
下图显示了 PC 如何通过集线器连接一个设备堆栈。
图 6：将设备堆栈连接到 PC

以太网管理端口支持的特性

以太网管理端口支持下列特性：

- 快速安装（Express Setup）（仅适用于交换机堆栈）；
- 网络助手（Network Assistant）
- 使用密码认证的 Telnet
- TFTP
- SSH（安全外壳协议）
- 使用 DHCP 协议进行配置
- SNMP（仅 ENTITY-MIB 和 IF-MIB）
- IP ping
- 接口特性：
- 速率：10Mb/s、100Mb/s 及自动协商
- 双工模式：全双工、半双工及自动协商
- 环路检测
- 思科发现协议（CDP）
- DHCP 中继代理
- IPv4 访问控制列表（ACL）
- 路由协议

注意： 在以太网管理端口启用特性之前，请确认该端口支持这种特性。如果在以太网端口上配置了其不支持的特性，这项特性有可能出现工作异常的情况，进而导致设备出现故障。

如何配置以太网管理端口

以太网管理端口的启用与禁用

总步骤

1. configure terminal

2. interface gigabitethernet0/0

3. shutdown

4. no shutdown

5. exit

6. show interfaces gigabitethernet0/0

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**interface gigabitethernet0/0** 示例：	在 CLI 界面中设置以太网管理端口

	Device(config)# **interface gigabitethernet0/0**	
步骤 3	**shutdown** 示例： Device(config-if)# **shutdown**	禁用以太网管理端口
步骤 4	**no shutdown** 示例： Device(config-if)# **no shutdow**	启用以太网管理端口
步骤 5	**exit** 示例： Device(config-if)# **exi**	离开接口配置模式
步骤 6	**show interfaces gigabitethernet0/0** 示例： Device# **show interfaces gigabitethernet0/0**	禁用链路状态。 要想查看 PC 的链路状态，可以观察以太网管理端口的 LED 等。如果 LED 灯为绿色，表示链路状态正常；如果 LED 灯熄灭，表示链路断开；如果 LED 灯为橙色，表示链路中出现了 POST 错误①

① POST 全称为 Quick Power On Self Test，即开机自检

在开始前

要想继续了解通过以太网管理端口管理和配置交换机的方法。可以参考《网络管理配置指南（Inspur 6650 交换机）》

其他参考资料

相关文档

相关主题	文档名
引导加载程序配置	《系统管理配置指南（Inspur 6650 交换机）》
引导加载程序命令	《系统管理配置指南（Inspur 6650 交换机）》

错误消息解码器

描述	链接
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。	http://www.icntnetworks.com

用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	

关于以太网管理端口的特性信息

版本	修改
Inspur INOS 12.2	引入该特性

配置 LLDP、LLDP-MED 及有线位置服务

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

LLDP、LLDP-MED 及有线位置服务概述

LLDP

思科发现协议（CDP）是一种所有 Inspur 制造的设备平台都支持的二层（数据链路层）设备发现协议，包括路由器、网桥、访问服务器、交换机和控制器。CDP 可以让网络管理应用自

动发现和学习其它与网络相连的 Inspur 设备。

要想支持非 Inspur 设备，实现与其它设备之间的互操作，这些设备需要支持 IEEE 802.1AB 链路层发现协议（LLDP）。LLDP 是一种网络设备使用的邻居发现协议，这种协议会将关于自己的信息通告给网络中的其它设备。这种协议同样运行在数据链路层上，它可以让运行不同网络层协议的系统之间学习到关于对方的信息。

LLDP 支持的 TLV

LLDP 支持一系列用于发现邻居设备的属性。这些属性包含类型、长度和参数描述，这些属性都称为 TLV。支持 LLDP 的设备可以使用 TLV 来接收来自于其它设备的信息，并且向其它设备发送信息。这种协议可以通告包括诸如配置信息、设备性能和设备身份在内的详细信息。交换机支持下列基本的管理 TLV。这些都是必需的 LLDP TLV。

- 端口描述 TLV
- 系统名 TLV
- 系统描述 TLV
- 系统性能 TLV
- 管理地址 TLV

设备也会通告下列这些特定组织机构定义的 LLDP TLV，以实现对 LLDP-MED 的支持：

- 端口 VLAN ID TLV（这是 IEEE 802.1 组织指定的 TLV）
- MAC/PHY 配置/状态 TLV（IEEE 802.3 组织指定的 TLV）

LLDP 与 Inspur 设备堆栈

一个由多台设备组成的堆栈在网络中相当于一台设备。因此，LLDP 发现的也是设备堆栈，而不是堆栈中的成员设备。

LLDP-MED

媒体端点设备（MED，Media Endpoint Devices）LLDP（LLDP-MED）是 LLDP 的一项扩展协议，这种协议运行在端点设备（如 IP 电话）与网络设备（如交换机）之间。这项协议会专门对 VoIP 应用提供支持，并且为功能发现、网络策略、以太网供电、产品清单管理和位置信息提供额外的 TLV。在默认情况下，所有 LLDP-MED TLV 都是启用的。

LLDP-MED 支持的 TLV

- LLDP-MED 支持下列 TLV：
- LLDP-MED 功能 TLV

  可以让 LLDP-MED 端点判断出直连设备①所支持的功能，及其启用的功能。

① 鉴于 LLDP-MED 运行在端点设备与网络设备之间，而本文档为交换机配置指南。因此，用户应注意在 LLDP-MED 这一部分，凡原文中提到“设备”一词，指的都是交换机。语音设备则一概用“端点”一词表示。——译者注

- 网络策略 TLV

  可以让网络设备和端点通告 VLAN 配置，以及该端口上使用的某项应用所对应的二层和三层属性。例如，交换机可以向一台电话通告其应该使用哪个 VLAN ID。这台电话可以与任何设备相连，获取自己的 VLAN ID，然后通过呼叫控制来启动通信。

  用户可以通过定义网络策略配置文件（profile）TLV 的方式，给语音和语音信令创建一个配置文件，在其中设置 VLAN 值、服务类型（CoS）、差分服务代码点（DSCP）和标记模式。接下来，这些配置文件属性会由交换机进行集中维护，然后再发送给电话。

- 电源管理 TLV

  可以在 LLEP-MED 端点与网络设备之间启用高级电源管理，让设备和电话能够描述电源

信息，如设备的供电方式、电源优先级以及设备需要的电量。

LLDP-MED 也支持通过一种扩展的电源 TLV 来通告准确的电源需求、端点电源优先级，以及端点和网络设备的电源状态。LLDP 启用时，端口会获得供电，电源 TLV 可以指定端点设备的实际电源需求，让设备可以根据这种需求来为端点设备分配功率。设备会处理请求消息，并且根据当前的功率分配情况来判断是批准还是拒绝自己接收到的请求。如果批准请求，那么交换机就会更新自己的功率分配。如果请求被拒绝，那么设备就会关闭这个端口的供电，生成一个系统日志消息，同时更新自己的功率分配情况。如果禁用了 LLDP-MED 或者端点根本不支持 LLDP-MED，那么在整个连接建立的过程中，设备都会使用最初分配的数值。

用户可以通过输入接口配置命令 **power inline** {**auto** [**max** *max-wattage*] | **never** | **static** [**max** *max-wattage*]}来修改电源的设置。在默认情况下，PoE 接口的模式为 **auto**。如果用户不指定任何参数，那么设备可以为其分配最大功率（30W）。

- 产品清单管理 TLV

  让端点可以将自己详细的产品清单信息发送给交换机，其中包括硬件修订版本、固件版本、软件版本、序列号、制造商、型号和资产 ID TLV。
- 位置 TLV

  从设备向端点设备提供位置信息。位置 TLV 可以发送下列信息：
  - 公民位置信息

    提供公民地址信息和邮政地址信息。所谓公民位置信息为包括其所在的街道地址、道路名和小区名的邮政信息。
  - ELIN 位置信息

    提供呼叫者的位置信息。这个位置是通过紧急位置标识符（ELIN，Emergency Location Identifier Number）判断出来的，所谓 ELIN 是一个电话号码，可以将紧急呼叫路由到本地公共安全接听点（PSAP），而 PSAP 则可以使用这个电话号码回叫呼叫方。
  - 地址位置信息

    提供关于交换机位置的地址信息，如交换机的经度、纬度和海拔高度。
  - 客户位置

    提供自定义的名称与交换机的位置参数。

有线位置服务

设备可以使用位置服务特性来将直连设备的位置与连接追踪信息发送给 Inspur 移动服务引擎（MSE，Mobility Services Engine）。被追踪设备既可以是有线端点，也可以是有线设备或者控制器。设备会使用网络移动性服务协议（NMSP，Network Mobility Services Protocol）的位置与连接通告，来向 MSE 通告设备链路状态变更事件。

MSE 会向设备发起 NMSP 连接，这会打开一个服务器端口。当 MSE 连接到设备之后，双方会首先通过一系列的消息交换来建立版本兼容性并交互服务信息，然后它们才会开始同步位置信息。在连接结束之后，设备会周期性地向 MSE 发送位置与连接通告。在一个间隔时间之内发生的一切链路状态变化，都会在这个时间间隔结束之前，以汇总的形式发送出去。

当设备在链路开启或关闭事件中，检测出了链路中的某台设备时，它也就获得了关于这个客户端的很多信息，包括设备的 MAC 地址、IP 地址和用户名。如果客户端支持 LLDP-MED 或者 CDP，那么设备还可以通过 LLDP-MED 位置 TLV 或者通过 CDP 获得这台设备的序列号和 UDI。

根据设备功能的不同，设备可以在链路处于开启状态获时取到下列关于客户端的信息：

- 端口连接中描述的插槽与端口
- 客户端 MAC 地址中描述的 MAC 地址
- 端口连接中描述的 IP 地址
- 802.1x 用户名（如适用）
- 设备分类会被描述为*有线工作站*（*wired station*）
- 状态会被描述为*新*（*new*）
- 序列号、UDI
- 设备型号
- 设备检测到这个关联后经历的时间（单位为秒）

根据设备功能的不同，设备可以在链路处于关闭状态时获取到下列关于客户端的信息：

- 断开连接的插槽与端口
- MAC 地址
- IP 地址
- 802.1x 用户名（如适用）
- 设备分类会被指定为*有线工作站*（*wired station*）
- 状态会被指定为*删除*（*delete*）
- 序列号、UDI
- 设备检测到这个关联断开后经历的时间（单位为秒）

当设备关闭时，它会在关闭与 MSE 的 NMSP 连接之前发送一条连接通告，其中包含 *delete* 这种状态，和 IP 地址。MSE 会认为这个通告表示，所有与这台设备相关的有线客户端都会与其断开关联。

如果用户在这台设备上修改位置地址，那么设备就会发送一条 NMSP 位置通告消息，标识出与此相关的端口以及修改后的地址信息。

默认的 LLDP 配置

表 10：默认的 LLDP 配置

特性	**默认设置**
LLDP 全局状态	禁用
LLDP 保持时间（丢弃前）	120 秒
LLDP 计时器（数据包更新频率）	30 秒
LLDP 重新启动的延迟	2 秒
LLDP tlv-select	禁止发送和接收所有 TLV
LLDP 接口状态	禁用
LLDP 接收	禁用
LLDP 过渡	禁用
LLDP med-tlv-select	禁用发送所有 LLDP-MED TLV。如果在全局启用了 LLDP，那么 LLDP-MED-TLV 也会启用

LLDP 的限制条件

- 如果用户将一个接口配置为了隧道端口，那么 LLDP 就会自动被禁用；
- 如果用户首先在接口上配置了一个网络策略配置文件，那么这个接口上就不能再应用

switchport voice vlan 这条命令了。但如果用户已经在接口上配置了 **switchport voice vlan** *vlan-id* 这条命令，那么这个接口上可以应用网络策略配置文件。通过这种方式，用户可以给接口分配一个语音 VLAN 或者语音信令 VLAN，同时在这个接口上应用网络策略配置文件；

- 用户无法在配置了网络策略配置文件的接口上配置静态安全 MAC 地址。

如何配置 LLDP、LLDP-MED 及有线位置服务

启用 LLDP

总步骤

1. enable
2. configure terminal
3. lldp run
4. interface *interface-id*
5. lldp transmit
6. lldp receive
7. end
8. show lldp
9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**lldp run** 示例： `Device (config)# lldp run`	在设备全局启用 LLDP
步骤 4	**interface** *interface-id* 示例： `Device(config)# interface gigabitethernet2/0/1`	指定要启用 LLDP 的物理接口，并且进入该接口的接口配置模式
步骤 5	**lldp transmit** 示例： `Device(config-if)# lldp`	启用接口发送 LLDP 数据包的操作

	transmit	
步骤 6	**lldp receive** 示例： `Device(config-if)#` **lldp receive**	启用接口接收 LLDP 数据包的操作
步骤 7	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 8	**show lldp** 示例： `Device#` **show lldp**	验证前面所作的配置
步骤 9	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

配置 LLDP 特征

用户可以对 LLDP 的更新频率、丢弃信息之前保留信息的总时长以及启用 LLDP 的延迟时间。用户还可以选择可以发送和接收哪些 LLDP 和 LLDP-MED TLV。

注释： 从第 2 步到第 5 步不需要按照具体步骤的顺序操作来执行配置。

总步骤

1. enable

2. configure terminal

3. lldp holdtime *seconds*

4. lldp reinit *delay*

5. lldp timer *rate*

6. lldp tlv-select

7. interface *interface-id*

8. lldp med-tlv-select

9. end

10. show lldp

11. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例：	进入特权 EXEC 模式。在提示时输入密码

	Device> **enable**	
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**lldp holdtime** *seconds* 示例： Device(config)# **lldp holdtime 120**	（可选）指定接收方设备应该丢弃信息之前，在这台设备上保留该信息的总时长。 时间范围是 0 到 65535 秒；默认时长为 120 秒
步骤 4	**lldp reinit** *delay* 示例： Device(config)# **lldp reinit 2**	（可选）指定 LLDP 在一个接口上启动时的延迟时间。 时间范围为 2 到 5 秒；默认时长为 2 秒。
步骤 5	**lldp timer** *rate* 示例： Device(config)# **lldp timer 30**	（可选）设置 LLDP 更新的发送频率（单位为秒）。 范围为 5 到 65534 秒；默认时间为 30 秒。
步骤 6	**lldp tlv-select** 示例： Device(config)# **tlv-select**	（可选）指定可以发送或接收的 LLDP TLV
步骤 7	**interface** *interface-id* 示例： Device(config)# **interface gigabitethernet2/0/1**	指定要启用 LLDP 的物理接口，并且进入该接口的接口配置模式
步骤 8	**lldp med-tlv-select** 示例： Device (config-if)# **lldp med-tlv-select inventory management**	（可选）指定可以发送或接收的 LLDP-MED TLV
步骤 9	**end** 示例： Device(config-if)# **end**	返回特权 EXEC 模式
步骤 10	**show lldp** 示例： Device# **show lldp**	验证前面所作的配置
步骤 11	**copy running-config**	（可选）将输入的条目保存到配置文件中

	startup-config 示例： `Device#` **copy running-config startup-config**	

配置 LLDP-MED TLV

在默认情况下，设备只会在它从终端设备那里接收到 LLDP-MED 数据包时才会发送 LLDP 数据包。接下来，设备也会发送带有 LLDP-MED 的 LLDP 数据包。当 LLDP-MED 条目过期之后，这台设备会再次回到只发送 LLDP 数据包的操作。

用户可以在接口配置模式下使用命令 **lldp** 让这个接口不要发送下表中的 TLV。

表 11：LLDP-MED-TLV

LLDP-MED-TLV	**描述**
inventory-management	LLDP-MED 产品清单管理 TLV
location	LLDP-MED 位置 TLV
network-policy	LLDP-MED 网络策略 TLV
power-management	LLDP-MED 电源管理 TLV

用户可以按照下面的步骤在接口上启用 TLV

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. lldp med-tlv-select

5. end

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet2/0/1**	指定要启用 LLDP 的物理接口，并且进入该接口的接口配置模式
步骤 4	**lldp med-tlv-select**	指定要启用的 TLV

	示例： `Device(config-if)#` **lldp med-tlv-select inventory management**	
步骤 5	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 6	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

配置网络策略 TLV

总步骤

1. enable

2. configure terminal

3. network-policy profile *profile number*

4. {**voice** | **voice-signaling**} **vlan** [*vlan-id* {**cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** {**cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**]

5. exit

6. interface *interface-id*

7. network-policy *profile number*

8. lldp med-tlv-select network-policy

9. end

10. show network-policy profile

11. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**network-policy profile** *profile number*	指定网络策略配置文件的编号，并且进入网络策略配置模式。编号范围是从 1 到 4294967295。

	示例： `Device(config)#` **network-policy profile 1**	
步骤 4	{**voice** | **voice-signaling**} **vlan** [*vlan-id* {**cos** *cvalue* | **dscp** *dvalue*}] | [[**dot1p** {**cos** *cvalue* | **dscp** *dvalue*}] | **none** | **untagged**] 示例： `Device(config-network-policy)#` **voice vlan 100 cos 4**	配置策略属性： • **voice**：指定语音应用类型； • **voice-signaling**：指定语音信令应用类型； • **vlan**：指定传输语音流量的本征 VLAN； • *vlan-id*：（可选）指定传输语音流量的 VLAN。取值范围是从 1 到 4094； • **cos** *cvalue*：（可选）指定所配置 VLAN 的二层优先级服务类型（CoS）。取值范围是 0 到 7，默认值为 5； • **dscp** *dvalue*：（可选）指定所配置 VLAN 的差分服务代码点（DSCP）值。取值范围是 0 到 63，默认值为 46； • **dot1p**：（可选）让电话使用 IEEE 802.1p 优先级标记并使用 VLAN 0（即本章 VLAN）； • **none**：（可选）不告诉 IP 电话语音 VLAN 的编号。此时电话会使用用户从电话键盘中输入的配置； • **untagged**：（可选）让电话发送未打标的语音流量。这是电话的默认操作方式。
步骤 5	**exit** 示例： `Device(config)#` **exit**	返回全局配置模式
步骤 6	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet2/0/1**	指定要配置网络策略配置文件的物理接口，并且进入该接口的接口配置模式
步骤 7	**network-policy** *profile number* 示例： `Device(config-if)#` **network-policy 1**	指定网络策略配置文件的编号
步骤 8	**lldp med-tlv-select network-policy** 示例： `Device(config-if)#` **lldp med-tlv-select network-policy**	指定网络策略 TLV
步骤 9	**end**	返回特权 EXEC 模式

	示例： `Device(config-if)#` **end**	
步骤 10	**show network-policy profile** 示例： `Device#` **show network-policy profile**	验证前面所作的配置
步骤 11	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

配置位置 TLV 和有线位置服务

用户可以从特权 EXEC 模式开始按照下面的步骤来给端点配置位置信息，并且将配置的信息应用到接口上。

总步骤

1. configure terminal

2. location {**admin-tag** *string* | **civic-location identifier** {*id* | **host**} | **elin-location** *string* **identifier** *id* | **custom-location identifier** {*id* | **host**} | **geo-location identifier** {*id* | **host**}}

3. exit

4. interface *interface-id*

5. location {**additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**} }

6. end

7. 使用下列命令：

• **show location admin-tag** *string*

• **show location civic-location identifier** *id*

• **show location elin-location identifier** *id*

8. copy running-config startup-config

	命令或操作	**目的**
步骤 1	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 2	**location** {**admin-tag** *string* | **civic-location identifier** {*id* | **host**} | **elin-location** *string* **identifier** *id* |	给一个端点设置位置信息： • **admin-tag**：设置管理标记或站点信息； • **civic-location**：设置公民位置信息； • **elin-location**：设置紧急位置信息（ELIN）；

	custom-location identifier {*id* | **host**} | **geo-location identifier** {*id* | **host**}} 示例： `Device(config)#` **location civic-location identifier 1** `Device(config-civic)#` **number 3550** `Device(config-civic)#` **primary-road-name "Inspur Way"** `Device(config-civic)#` **city "San Jose"** `Device(config-civic)#` **state CA** `Device(config-civic)#` **building 19** `Device(config-civic)#` **room C6** `Device(config-civic)#` **county "Santa Clara"** `Device(config-civic)#` **country US**	• **custom-location**：设置客户位置信息； • **geo-location**：设置地理空间位置信息； • **identifier** *id*：设置公民、ELIN、客户或地理位置的 ID； • **host**：设置主机的公民、客户或地理位置； • *string*：用文字描述的形式置站点或位置信息。
步骤 3	**exit** 示例： `Device(config)#` **exit**	返回全局配置模式
步骤 4	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet2/0/1**	指定要配置位置信息的物理接口，并且进入该接口的接口配置模式
步骤 5	**location** {**additional-location-information** *word* | **civic-location-id** {*id* | **host**} | **elin-location-id** *id* | **custom-location-id** {*id* | **host**} | **geo-location-id** {*id* | **host**} } 示例： `Device(config-if)#` **location elin-location-id 1**	在这个接口中输入位置信息： • **additional-location-information**：设置关于位置或地点的额外信息 • **civic-location-id**：设置这个接口的全局公民位置信息 • **elin-location-id**：设置这个接口的紧急位置信息 • **custom-location-id**：设置这个接口的客户位置信息 • **geo-location-id**：设置这个接口的地理空间位置信息

		• **host**：设置主机的位置标识符 • *word*：设置一段与位置信息有关的文字 • *id*：设置公民、ELIN、客户或地理位置的 ID。ID 取值范围是 1 到 4095。
步骤 6	**end** 示例： `Device(config-if)#` **end**	返回特权 EXEC 模式
步骤 7	使用下面命令： • **show location admin-tag** *string* • **show location civic-location identifier** *id* • **show location elin-location identifier** *id* 示例： `Device#` **show location admin-tag** 或 `Device#` **show location civic-location identifier** 或 `Device#` **show location elin-location identifier**	验证前面所作的配置
步骤 8	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

在设备上启用有线位置服务

在开始前

要想让有线位置服务正常工作，用户必须输入全局配置命令 **ip device tracking**。

总步骤

1. enable

2. configure terminal

3. nmsp notification interval {**attachment** | **location**} *interval-seconds*

4. end

5. show network-policy profile

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable**	进入特权 EXEC 模式。在提示时输入密码

	示例： `Device>` **enable**	
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**nmsp notification interval** {**attachment** | **location**} *interval-seconds* 示例： `Device(config)#` **nmsp notification interval location 10**	设置 NMSP 的通告间隔。 • **attachment**：设置连接通告间隔； • **location**：设置位置通告间隔； • *interval-seconds*：设置设备向 MSE 发送通告更新或者连接更新之前等待的秒数。范围为从 1 到 30 秒；默认为 30 秒。
步骤 4	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 5	**show network-policy profile** 示例： `Device#` **show network-policy profile**	验证前面所作的配置
步骤 6	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

LLDP、LLDP-MED 与有线位置服务的配置示例

配置网络策略 TLV：示例

这个示例显示了如何通过配置让携带 CoS 的语音应用在 VLAN 100 中传输，同时在接口上启用网络策略配置文件和网络策略 TLV：

```
# configure terminal
(config)# network-policy 1
(config-network-policy)# voice vlan 100 cos 4
(config-network-policy)# exit
(config)# interface gigabitethernet1/0/1
```

```
(config-if)# network-policy profile 1
(config-if)# lldp med-tlv-select network-policy
```

这个示例显示了如何让携带优先级标记的语音应用通过本征 VLAN 进行传输：

```
config-network-policy)# voice vlan dot1p cos 4
config-network-policy)# voice vlan dot1p dscp 34
```

LLDP、LLDP-MED 和有线位置服务的监控与维护

监控与维护 LLDP、LLDP-MED 和有线位置服务的命令可以参考下表：

命令	描述
clear lldp counters	将流量计数器重置为 0
clear lldp table	删除 LLDP 邻居信息表
clear nmsp statistics	清除 NMSP 统计数据计数器
show lldp	显示全局信息，如传输频率、被发送数据包保存时间、LLDP 在一个接口上启动的延迟时间
show lldp entry *entry-name*	显示关于特定邻居的信息。 用户可以输入星号（*）来显示所有邻居，也可以输入具体的邻居名
show lldp interface [*interface-id*]	显示与启用了 LLDP 的接口有关的信息。 可以让系统仅仅显示某个接口的信息
show lldp neighbors [*interface-id*] [**detail**]	显示关于邻居的信息，其中包括设备类型、接口类型与编号、保存时间的设置、功能与端口 ID。 可以让系统仅显示某个接口的邻居，也可以让系统显示更加具体的信息
show lldp traffic	显示 LLDP 计时器，包括收发的数据包数量、丢弃的数据包数量以及未识别的 TLV 数量
show location admin-tag *string*	显示特定管理标记或站点的位置信息
show location civic-location identifier *id*	显示一个特定全局公民位置的位置信息
show location elin-location identifier *id*	显示一个紧急位置的位置信息
show network-policy profile	显示用户配置的网络策略配置文件
show nmsp	显示 NMSP 信息

其他关于 LLDP、LLDP-MED 与有线位置服务的参考资料

错误消息解码器

描述	链接
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

技术助手

描述	链接

Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。 用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	http://www.icntnetworks.com

关于 LLDP、LLDP-MED 与有线位置服务的特性信息

版本	修改
Inspur INOS 12.2	引入该特性

配置系统 MTU

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于 MTU 的信息

所有设备接口默认收发的数据帧最大传输单元（MTU）为 1500 字节。

系统 MTU 的限制条件

用户在配置系统 MTU 值时，可以参考下面的指导方针：

- 设备不支持给不同接口分别配置 MTU；
- 如果用户在全局配置模式下输入命令 **system mtu** *bytes*，这条命令并不会在设备上生效。这条命令只会作用于交换机快速以太网端口的系统 MTU 设置。

如何配置 MTU 值

配置系统 MTU

用户可以按照下列步骤来修改交换与路由数据包的 MTU 值：

总步骤

1. enable

2. configure terminal

3. system mtu *bytes*

4. end

5. copy running-config startup-config

6. reload

7. show system mtu

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**system mtu** *bytes* 示例： `Device(config)#` **system mtu 1900**	（可选）给所有 GigabitEthernet 和 10-GigabitEthernet 接口修改 MTU 值
步骤 4	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 5	**copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

	示例： `Device#` **copy running-config startup-config**	
步骤 6	**reload** 示例： `Device#` **reload**	重启操作系统
步骤 7	**show system mtu** 示例： `Device#` **show system mtu**	验证前面所作的设置

系统 MTU 的配置示例

这个示例显示了如何将 GigabitEthernet 端口的最大数据包设置为 7500 字节：

```
Device(config)# system mtu 7500system mtu 1900
Device(config)#
Device(config)# exit
```

如果用户输入的数值超出了这类接口许可的范围，设备就不会接受这条命令。这个示例显示了当用户尝试给 GigabitEthernet 端口设置一个超出范围的参数时，系统作出的响应：

```
Device(config)# system mtu 25000
^
% Invalid input detected at '^' marker.
```

这个示例显示了命令 **show system mtu** 的输出信息：

```
Device# show system mtu
Global Ethernet MTU is 1500 bytes.
```

其他关于系统 MTU 的参考资料

错误消息解码器

描述	链接
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。 用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进	http://www.icntnetworks.com

行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	

关于系统 MTU 的特性信息

版本	修改
Inspur INOS XE 3.3SE	引入该特性

IPv6

配置 MLD Snooping

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于配置 IPv6 MLD Snooping 的信息

注释： 要想使用 IPv6 MLS Snooping，交换机必须运行 LAN Base 镜像。

用户可以使用 MLD（组播侦听者发现协议）Snooping，在交换机上向交换网络中的客户端与路由器高效地分发 IPv6（IP 协议第 6 版）组播数据。如无特别说明，交换机一词在这里指代的既可以是独立交换机，也可以是交换机堆栈。

注释： 运行 LAN Base 镜像的 Inspur 2960-X 交换机可以支持堆栈。

注释： 要想使用 IPv6，用户必须在交换机上配置双栈 IPv6 和 IPv6 交换数据库管理（SDM）模版。

运行 LAN Base 镜像的交换机特性集不支持使用路由模版。

注释： 用户若想获取本章所述全部命令的语法与使用信息，可以参阅这个版本的命令参考手册，或者 Inspur INOS 文档中的相关资料。

理解 MLD Snooping

在 IP 第 4 版中，二层交换机可以使用 IGMP（互联网组播管理协议）snooping 动态配置二层交换机，让组播流量只转发给相关的 IP 组播设备所在的接口，以此限制组播流量的泛洪。在 IPv6 中，MLD snooping 也可以执行类似的功能。通过 MLD snooping，IPv6 组播数据可以有选择地发送给那些希望接收到这些数据的端口，而不会在这个 VLAN 的所有端口进行泛洪。这个列表是交换机通过窥探 IPv6 组播控制数据包组建起来的。

MLD 是 IPv6 组播路由器使用的一项协议，其作用是在与路由器直连的发现链路上发现组播侦听设备（也就是那些希望接收到 IPv6 组播数据包的节点），同时发现各个邻居节点分别对哪些组播数据包感兴趣。MLD 来自于 IGMP。MLD 版本 1（MLDv1）是 ICMP（互联网控制消息协议）v6 的一个子协议，MLD 消息是 ICMPv6 消息的一个子集。在 IPv6 数据包中，下一个头部（Next Header）字段的取值为 58 的消息即为 MLD 消息。

交换机支持下面两个版本的 MLD snooping：

- MLDv1 Snooping 可以监测 MLDv1 控制数据包，并且根据 IPv6 目的组播地址来建立流量的桥接关系；
- MLDv2 Basic Snooping（MBSS，MLDv2 snooping 基础版）使用 MLDv2 控制数据包来根据 IPv6 目的组播地址建立流量转发对应关系。

交换机可以对 MLDv1 和 MLDv2 协议的数据包进行窥探，并且根据目的 IPv6 组播地址来桥接 IPv6 组播数据。

注释： 交换机不支持 MLDv2 enhanced snooping（MLDv2 snooping 增强版），这种特性可以根据 IPv6 源和目的组播地址来设置转发方式。

MLD snooping 既可以在全局启用或者禁用，也可以以 VLAN 为单位启用和禁用。在启用了 MLD Snooping 后，交换机会在软件和硬件中都建立起一个以 VLAN 为单位的 IPv6 组播地址表。接下来，交换机会在硬件中执行基于 IPv6 组播地址的桥接。

根据 IPv6 组播的标准，交换机会对交换机 MAC 地址最低 4 个八位二进制数，与 33:33:00:00:00:00 这个 MAC 地址执行逻辑或（OR）运算，通过这种方式提取出组播 MAC 地址。例如，FF02:DEAD:BEEF:1:3 这个 IPv6 MAC 地址对应的以太网 MAC 地址就是 33:33:00:01:00:03。

如果目的 IPv6 地址不匹配目的 MAC 地址，则组播数据包就是不匹配的。交换机会根据 MAC 地址表，对不匹配的数据包执行硬件转发。如果交换机的 MAC 地址表中没有这个目的 MAC

地址，那么交换机就会以同一个 VLAN 的所有端口作为接收方端口，来泛洪这个数据包。

MLD 消息

MLDv1 支持三种类型的消息：

- 侦听者查询消息（Listener Query）相当于 IGMPv2 查询消息，它可以执行总的查询，也可以针对某个特定的 MAC 地址进行查询；
- 组播侦听者报告（Multicast Listener Report）相当于 IGMPv2 报告；
- 组播侦听者完成（Multicast Listener Done）消息相当于 IGMPv2 离开消息。

MLDv2 支持 MLDv2 查询和报告，和 MLDv1 报告和完成消息。

消息计时器和因为消息收发而导致的状态过渡，都与 IGMPv2 消息相同。MLD 路由器和交换机会忽略那些没有有效链路本地 IPv6 源地址的 MLD 消息。

MLD 查询

交换机会发送 MLD 查询消息，建立 IPv6 组播地址数据库，并且会使用 MLD 特性组和 MLD 组与特定源查询消息来响应 MLD 完成消息。交换机还支持报告抑制（report suppression）、报告代理（report proxying）、直接离开（Immediate-Leave）功能和静态 IPv6 组播组地址配置。

在禁用了 MLD snooping 之后，所有 MLD 查询消息都会在消息的入站 VLAN 中进行泛洪。

在启用了 MLD snooping 之后，交换机会将接收到的 MLD 查询消息在入站 VLAN 中进行泛洪，同时将查询消息发送给 CPU 进行处理。MLD snooping 会通过接收到的查询消息来建立 IPv6 组播地址数据库。它会检测组播路由器的端口、维护计时器、设置报告响应时间、学习这个 VLAN 中的查询方源 IP 地址、学习这个 VLAN 中的查询方端口，监控组播地址时间的老化情况。

当 MLD snooping 数据库中存在一个组时，交换机就可以通过发送 MLDv1 报告来对针对该组的查询作出响应。如果这个组是未知的，交换机就会把针对这个组的查询在整个入站 VLAN 当中进行泛洪。

当一台主机想要离开一个组播组时，它可以发送一条 MLD 完成消息（相当于 IGMP 离开消息）。交换机在接收到这个 MLDv1 完成消息之后，如果没有启用直接离开（Immediate-Leave）特性，那么交换机就会向接收到这个消息的端口发送一条 MASQ 消息，以判断这个端口是否连接了其他希望继续保持在这个组播组中的设备。

组播客户端老化的稳健性

用户可以对查询的数量进行配置，让不达标的端口成员离开其对应的组播地址。只有当对某个地址的报告数量达不到用户配置的查询数量时，这个端口才会被交换机从响应的组播组地址中移除出去。默认的查询数量为 2。

组播路由器发现

MLD snooping 也和 IGMP snooping 一样执行组播路由器发现，组播路由器发现拥有下列特征：

- 用户所配置的端口永不老化；
- 通过 MLDv1 snooping 查询和 IPv6 PIMv2 数据包实现动态端口学习；
- 如果同一个二层接口连接了多台路由器，MLD snooping 只会在该端口追踪一台组播路由器（追踪的是最近发送路由器控制数据包的那台路由器）；
- 动态组播路由器端口老化默认的计时器时间为 5 分钟；如果端口连续 5 分钟没有接收到控制数据包，那么这台组播路由器就会路由器端口列表中删除；
- 只有在交换机上启用了 MLD snooping 的情况下，IPv6 组播路由器发现才会生效；
- 接收到的 IPv6 组播路由器控制数据包一定会在入站 VLAN 中进行泛洪，这与交换机上是否启用 MLD snooping 无关；
- 在设备发现了第一个 IPv6 组播路由器端口之后，它就会开始仅向发现的路由器端口转

发 IPv6 组播数据。（而在此之前，所有 IPv6 组播数据都会在入站 VLAN 中进行泛洪）

MLD 报告

MLDv1 加入消息的处理方式与 IGMPv2 基本相同。当一个 VLAN 中没有检测到 IPv6 组播路由器时，交换机就不会处理或转发报告。而当设备检测到了 IPv6 组播路由器，并且接收到了 MLDv1 报告之后，它就会在 VLAN MLD 数据库中输入一个 IPv6 组播组地址。接下来，所有在这个 VLAN 中去往这个组的 IPv6 组播流量都会用这个地址进行转发。如果设备禁用了 MLD snooping，那么报告就会在入站 VLAN 中进行泛洪。

如果启用了 MLD snooping 特性，那么 MLD 报告抑制（称为侦听者消息抑制）也会自动启用。通过报告抑制特性，交换机就会将第一个组接收到的 MLDv1 报告转发给 IPv6 组播路由器；但它不会再将后续的组报告发送给路由器。如果禁用了 MLD snooping，那么报告抑制特性也会被禁用，因此所有 MLDv1 报告都会在入站 VLAN 中进行泛洪。

交换机也支持 MLDv1 代理报告功能。当交换机接收到一个 MLDv1 MASQ 消息时，如果交换机的另一个端口存在这个组，并且查询消息到达的那个端口不是该地址的最后一个成员端口的话，那么交换机就会向查询消息到达的那个地址发送 MLDv1 报告，以响应 MLDv1 MASQ 响应。

MLD 完成消息与直接离开特性

如果交换机上启用了直接离开（Immediate-Leave）特性，并且一台主机发送了一个 MLDv1 消息（相当于 IGMP 离开消息），那么交换机会立刻将接收到完成消息的那个端口从组中删除。用户可以以 VLAN 为单位启用直接离开特性，此时用户应该只在那些 VLAN 成员端口都只连接了一台主机的 VLAN 中启用这项特性（这一点和 IGMP snooping 相同）。如果这个端口是一个组成员的最后一个端口，那么这个组也会一并被交换机删除，同时交换机还会将离开信息转发给被删除的那台 IPv6 组播路由器。

如果一个 VLAN 中没有启用直接离开特性（当某个组中，存在有多个客户端连接在同一个端口上的情形时，就不应该在启用该特性），而该 VLAN 中有端口接收到一个完成消息，那么这个端口就会生成一个 MASQ。用户可以根据接收到的 MASQ 数量，来控制何时移除对某个地址移除一个端口的成员身份。在端口接收到的查询次数达到了用户配置的数值，但该端口并没有去往对应地址的 MLDv1 报告时，交换机就会删除这个端口在该地址的成员身份。

用户可以使用全局配置命令 **ipv6 mld snooping last-listener-query count** 来配置生成的 MASQ 数量。默认的数量为 2。

交换机会将 MASQ 发送给完成消息的目的地址。如果在交换机最大响应时间之内，没有报告消息发送给 MASQ 中指定的 IPv6 组播地址，交换机就会将发送 MASQ 的端口从 IPv6 组播地址数据库中删除。用户可以通过全局配置命令 **ipv6 mld snooping last-listener-query-interval** 来配置最大响应时间。如果交换机删除的端口是组播地址的最后一个成员端口，那么交换机也会同时删除这个组播地址，同时交换机会向所有被删除的组播路由器发送一个地址离开信息。

在没有启用直接离开特性的情况下，如果某个端口接收到了一条 MLD 完成消息，那么交换机就会在这个端口上创建 MASQ，并且将这些消息发送给发送完成消息的那个 IPv6 组播地址。用户可以对发送多少 MASQ 进行配置，也可以配置交换机在从组播组中删除端口之前，等待响应消息的时长。

如果启用了 MLDv1 直接离开特性，那么当交换机在某个端口上检测到了一个 MLD 完成消息时，它就会立刻将这个端口从组播组中移除。只有在 VLAN 中每个端口都只连接了一台接收方设备时，用户才可以考虑在这个 VLAN 上使用直接离开特性。如果同一个端口上连接了某个组播组的多台客户端，那就不要在这个端口所在的 VLAN 启用直接离开特性。

拓扑变化通告处理

在用户使用全局配置命令 **ipv6 mld snooping tcn query solicit** 启用了通告（TCN）请求（solicitation）特性之后，MLDv1 snooping 就会对 VLAN 泛洪自己配置数量的 MLDv1 查询消息，然后再将组播数据发送给所选的端口。用户可以使用全局配置命令 **ipv6 mld snooping tcn flood query count** 来设置这个数值。默认值为发送 2 条查询消息。交换机也会在交换机成为这个 VLAN 中的 STP 根，或者用户将其配置为这个 VLAN 的 STP 根时，生成 MLDv1 的全局完成消息，这种做法与 IGMP snooping 相同。

如何配置 IPv6 MLD Snooping

默认的 MLD Snooping 配置

特性	**默认设置**
MLD snooping（全局）	禁用
MLD snooping（各个 VLAN）	启用。不过 MLD snooping 必须首先在全局启用，各个 VLAN 的 MLD snooping 才能生效
IPv6 组播地址	未配置
IPv6 组播路由器端口	未配置
MLD snooping 直接离开（Immediate Leave）特性	禁用
MLD snooping 稳健性（robustness）变量	全局：2；各个 VLAN：0 **注释**：各个 VLAN 的参数优于全局设置。当 VLAN 值为 0 时，VLAN 才会使用全局值
最后的侦听者查询数	全局：2；各个 VLAN：0 **注释**：各个 VLAN 的参数优于全局设置。当 VLAN 值为 0 时，VLAN 才会使用全局值
最后的侦听者查询间隔	全局：1000（即 1 秒）；各个 VLAN：0 **注释**：各个 VLAN 的参数优于全局设置。当 VLAN 值为 0 时，VLAN 才会使用全局值
TCN 查询请求	禁用
TCP 查询数	2
MLD 侦听者抑制	禁用

MLD Snooping 配置指南

在配置 MLD snooping 时，可以考虑下面的指导方针：

- 用户可以随时配置 MLD snooping 特征，但必须使用全局配置命令 **ipv6 mld snooping** 在全局启用 MLD snooping 才能让配置生效；
- 当这台 IPv6 组播路由器是一台 Inspur 6500 交换机，而且用户使用的又是扩展 VLAN（即范围在 1006 到 4094 之间的 VLAN）时，那么用户就必须在 Inspur 6500 交换机的扩展 VLAN 上启用 IPv6 MLD snooping，这是为了让交换机能够在这个 VLAN 上接收到查询消息。如果使用的是正常范围 VLAN（即范围在 1 到 1005 之间的 VLAN），用户可以不必在 Inspur 6500 交换机的这些 VLAN 上启用 IPv6 MLD；

- MLD snooping 与 IGMP snooping 是相互独立工作的。用户可以在交换机上同时启用这两个特性；
- 交换机或交换机堆栈上允许的最大组播条目数量是由用户配置的 SDM 模板来决定的；
- 交换机或交换机堆栈上允许的最大地址条目数量为 4000 条。

在交换机上启用或禁用 MLD Snooping（CLI 界面配置方法）

在默认情况下，IPv6 MLD snooping 在交换机上是全局禁用的，但同时是在所有 VLAN 上启用的。当 MLD snooping 在全局禁用时，它也不会真的在所有 VLAN 上生效。而当用户全局启用 MLD snooping 时，各个 VLAN 的配置就会覆盖全局的配置。也就是说，在默认状态下，MLD snooping 只有在 VLAN 接口上是启用的。

对于一个范围内的 VLAN，用户可以针对各个 VLAN 分别启用和禁用 MLD snooping，但如果用户在全局禁用了 MLD snooping，那么所有 VLAN 上配置的 MLD snooping 也会被禁用。如果在全局启用了 snooping，那么用户也就可以给各个 VLAN 设置是否启用 snooping。

用户可以从特权 EXEC 模式中，按照下面的步骤在交换机上全局启用 MLD snooping：

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**ipv6 mld snooping** 示例： `Device(config)# ipv6 mld snooping`	在交换机上启用 MLD snooping
步骤 3	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式
步骤 4	**copy running-config startup-config** 示例： `Device(config)# copy running-config startup-config`	（可选）将输入的条目保存到配置文件中
步骤 5	**reload** 示例： `Device(config)# reload`	重启操作系统

对一个 VLAN 启用或禁用 MLD Snooping（CLI 界面配置方法）

用户可以从特权 EXEC 模式中，按照下面的步骤在一个 VLAN 上启用 MLD snooping：

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**ipv6 mld snooping** 示例： `Device(config)# ipv6 mld snooping`	在交换机上启用 MLD snooping
步骤 3	**ipv6 mld snooping vlan** *vlan-id* 示例： `Device(config)# ipv6 mld snooping vlan 1`	在一个 VLAN 上启用 MLD snooping。VLAN ID 的取值范围是从 1 到 1001，以及从 1006 到 4094. **注释**：必须在全局启用 MLD snooping， VLAN snooping 才能生效
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置一个静态组播组（CLI 界面配置方法）

主机或二层端口一般会动态加入组播组，但用户也可以给一个 VLAN 静态配置 IPv6 组播地址和成员端口。

用户可以从特权 EXEC 模式中，按照下面的步骤将一个二层端口添加为组播组的成员：

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**ipv6 mld snooping vlan** *vlan-id* **static** *ipv6_multicast_address* **interface** *interface-id* 示例： `Device(config)# ipv6 mld snooping vlan 1 static FF12::3`	将一个二层端口配置为一个组播组的成员端口： • *vlan-id* 是组播组的 VLAN ID。VLAN ID 的范围是从 1 到 1001，以及从 1006 到 4094； • *ipv6_multicast_address* 是 128 位组 IPv6 地址。这个地址必须按照 RFC 2373 的格式来进行配置；

	interface gigabitethernet 0/1	• *interfaces-id* 是成员端口。这个端口既可以是物理接口，也可以是 port channel（编号 1 到 48）。
步骤 3	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 4	使用下面两条命令之一： • **show ipv6 mld snooping address** • **show ipv6 mld snooping address vlan** *vlan-id* 示例： `Device#` **show ipv6 mld snooping address** 或 `Device#` **show ipv6 mld snooping vlan 1**	验证静态成员端口和 IPv6 地址

配置一个组播路由器端口（CLI 界面配置方法）

注释： 只有交换机端口支持与组播路由器之间的静态连接

用户可以从特权 EXEC 模式中，按照下面的步骤向一个 VLAN 中添加一个组播路由器端口：

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 2	**ipv6 mld snooping vlan** *vlan-id* **static** *ipv6_multicast_address* **interface** *interface-id* 示例： `Device(config)#` **ipv6 mld snooping vlan 1 static FF12::3 interface gigabitethernet 0/1**	设置组播路由器 VLAN ID 并将这个接口指定给组播路由器： • VLAN ID 的范围是从 1 到 1001，以及从 1006 到 4094； • 这个端口既可以是物理接口，也可以是 port channel。后者的编号范围是从 1 到 48。
步骤 3	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式

步骤 4	使用下面两条命令之一： **show ipv6 mld snooping mrouter** [**vlan** *vlan-id*] 示例： `Device#` **show ipv6 mld snooping mrouter vlan 1**	验证该 VLAN 接口已经启用了 IPv6 MLD Snooping

启用 MLD 直接离开特性（CLI 界面配置方法）

用户可以从特权 EXEC 模式中，按照下面的步骤启用 MLDv1 直接离开特性：

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 2	**ipv6 mld snooping vlan** *vlan-id* **immediate-leave** 示例： `Device(config)#` **ipv6 mld snooping vlan 1 immediate-leave**	在一个 VLAN 接口上启用 MLD 直接离开特性
步骤 3	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 4	**show ipv6 mld snooping mrouter** [**vlan** *vlan-id*] 示例： `Device#` **show ipv6 mld snooping mrouter vlan 1**	验证该 VLAN 接口已经启用了直接离开特性

配置 MLD Snooping 查询（CLI 界面配置方法）

用户可以从特权 EXEC 模式中，按照下面的步骤为交换机或 VLAN 配置 MLD Snooping 查询的相关功能：

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例：	进入全局配置模式

	Device# **configure terminal**	
步骤 2	**ipv6 mld snooping robustness-variable** *value* 示例： Device(config)# **ipv6 mld snooping robustness-variable 3**	（可选）设置交换机在将一个没有对一般性查询作出响应的侦听设备（端口）删除之前，会发送多少条查询消息。这个参数的取值范围是从 1 到 3，默认设置为 2
步骤 3	**ipv6 mld snooping vlan** *vlan-id* **robustness-variable** *value* 示例： Device(config)# **ipv6 mld snooping vlan 1 robustness-variable 3**	（可选）给各个 VLAN 分别设置稳健性变量，这个变量的目的是指明 MLD snooping 特性在因没有收到 MLD 报告响应消息而让一个组播地址老化之前，会发送多少条一般性查询消息。这个参数的取值范围是从 1 到 3，默认设置为 2。如果设置为 0，则针对该 VLAN 应用全局的稳健性变量参数
步骤 4	**ipv6 mld snooping last-listener-query-count** *count* 示例： Device(config)# **ipv6 mld snooping last-listener-query-count 7**	（可选）设置交换机让一个 MLD 客户端老化之前，会发送多少条 MASQ。这个参数的取值范围是从 1 到 7，默认设置为 2。查询消息每隔 1 秒发送一次
步骤 5	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-count** *count* 示例： Device(config)# **ipv6 mld snooping vlan 1 last-listener-query-count 7**	（可选）给各个 VLAN 设置最后侦听者查询计数。这个值会覆盖全局配置的参数。这个参数的取值范围是从 1 到 7，默认设置为 0。如果设置为 0，则针对该 VLAN 应用全局的计数值。查询消息每隔 1 秒发送一次
步骤 6	**ipv6 mld snooping last-listener-query-interval** *interval* 示例： Device(config)# **ipv6 mld snooping last-listener-query-interval 2000**	（可选）设置交换机在发送 MASQ 之后，会等待的最大响应时间，经过这段时间没有得到响应，交换机才会将端口从这个组播组中删除。这个参数的取值范围是从 100 到 32768 毫妙，默认设置为 1000 毫妙（即 1 秒）。
步骤 7	**ipv6 mld snooping vlan** *vlan-id* **last-listener-query-interval** *interval* 示例： Device(config)# **ipv6 mld snooping vlan 1 last-listener-**	（可选）给各个 VLAN 设置最后侦听者查询间隔时间。这个值会覆盖全局配置的参数。这个参数的取值范围是从从 0 到 32768 毫妙，默认设置为 0。如果设置为 0，则针对该 VLAN 应用全局的最后侦听者查询间隔时间

	query-interval 2000	
步骤 8	**ipv6 mld snooping tcn query solicit** 示例： `Device(config)#` **ipv6 mld snooping tcn query solicit**	（可选）启用拓扑变更通告（TCN）请求，即 VLAN 会对所有 IPv6 组播流量泛洪用户指定数量的请求消息，然后才会将组播数据专门发送给那些请求接收这些数据的端口。TCN 默认是禁用的。
步骤 9	**ipv6 mld snooping tcn flood query count** *count* 示例： `Device(config)#` **ipv6 mld snooping tcn flood query count 5**	（可选）在启用了 TCN 之后，用户需要设置发送 TCN 请求的次数。这个参数的取值范围是从从 1 到 10，默认设置为 2。
步骤 10	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 11	**show ipv6 mld snooping querier** [**vlan** *vlan-id*] 示例： `Device(config)#` **show ipv6 mld snooping querier vlan 1**	（可选）验证为交换机或 VLAN 配置的 MLD snooping 查询方信息

禁用 MLD 侦听者消息抑制（CLI 界面配置方法）

MLD snooping 侦听者消息抑制在默认状态下就是启用的。在启用这个特性之后，交换机只会针对每个组播路由器查询消息转发一个 MLD 报告消息。在禁用了这个消息抑制特性之后，交换机可以向组播路由器转发多个 MLD 报告消息。

用户可以从特权 EXEC 模式中，按照下面的步骤禁用 MLD 侦听者消息抑制特性：

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 2	**no ipv6 mld snooping listener-message-suppression** 示例： `Device(config)#` **no ipv6 mld snooping listener-message-**	禁用 MLD 消息抑制特性

	suppression	
步骤 3	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 4	**show ipv6 mld snooping** 示例： `Device#` **show ipv6 mld snooping**	验证 IPv6 MLD snooping 报告抑制特性已经禁用

查看 MLD Snooping 的信息

用户可以查看路由器端口或 VLAN 接口那些动态学习或静态配置的 MLD snooping 信息，也可以查看针对 MLD snooping 给一个 VLAN 配置的 IPv6 组地址组播条目。

表 18：显示 MLD snooping 信息的命令

命令	目的
show ipv6 mld snooping [**vlan** *vlan-id*]	显示这台交换机上所有 VLAN 或某些特性 VLAN 的 MLD snooping 配置信息。 （可选）输入 **vlan** *vlan-id* 可以让系统显示一个 VLAN 的信息。VLAN ID 的取值范围是从 1 到 1001，从 1006 到 4094
show ipv6 mld snooping mrouter [**vlan** *vlan-id*]	显示动态学习和手动配置的组播路由器接口信息。如果启用了 MLD snooping，那么交换机就会自动学习组播路由器连接的端口。这些就是动态学习的端口。 （可选）输入 **vlan** *vlan-id* 可以让系统显示一个 VLAN 的信息。VLAN ID 的取值范围是从 1 到 1001，从 1006 到 4094
show ipv6 mld snooping querier [**vlan** *vlan-id*]	显示 VLAN 中最近接收到的 MLD 查询消息的 IPv6 地址和入站端口。 （可选）输入 **vlan** *vlan-id* 可以让系统显示一个 VLAN 的信息。VLAN ID 的取值范围是从 1 到 1001，从 1006 到 4094
show ipv6 mld snooping address [**vlan** *vlan-id*] [**count** | **dynamic** | **user**]	显示交换机或一个 VLAN 的所有 IPv6 组播地址信息或特定 IPv6 组播地址信息。 • 输入 **count** 则系统会显示交换机或一个 VLAN 中的组数量； • 输入 **dynamic** 则系统会显示交换机或一个 VLAN 通过 MLD snooping 学习到的组信息； • 输入 **user** 则系统会显示用户给交换机或一个 VLAN 配置的组信息。
show ipv6 mld snooping address vlan *vlan-id* [	显示特定 VLAN 和 IPv6 组播地址的 MLD

ipv6-multicast-address]	snooping 信息

配置 MLD Snooping 的示例

配置静态组播组：示例

这个示例显示了静态配置 IPv6 组播组的方法：

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 2 static FF12::3 interface gigabitethernet 1/0/1
Device(config)# end
```

配置组播路由器端口：示例

这个示例显示了向 VLAN 200 中添加组播路由器端口的方法：

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 200 mrouter interface gigabitethernet 0/2
Device(config)# exit
```

启用 MLD 直接离开（Immediate-Leave）特性：示例

这个示例显示了如何在 VLAN 130 上启用 MLD 直接离开特性：

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 130 immediate-leave
Device(config)# exit
```

配置 MLD Snooping 查询：示例

这个示例显示了如何将 MLD snooping 全局稳健性变量设置为 3：

```
Device# configure terminal
Device(config)# ipv6 mld snooping robustness-variable 3
Device(config)# exit
```

这个示例显示如何将一个 VLAN 的 MLD snooping 最后侦听者查询数量设置为 3：

```
Device# configure terminal
Device(config)# ipv6 mld snooping vlan 200 last-listener-query-count 3
Device(config)# exit
```

这个示例显示如何将 MLD snooping 最后侦听者查询时间间隔（最大响应时间）设置为 2000（即两秒）：

```
Device# configure terminal
Device(config)# ipv6 mld snooping last-listener-query-interval 2000
Device(config)# exit
```

配置 IPv6 单播路由

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于配置 IPv6 单播路由的信息

本章会描述如何在交换机上配置 IPv6 单播路由。

注释： 要想使用这一章介绍的所有 IPv6 特性，交换机或堆栈主交换机必须运行 IP services 特性集。运行 IP base 特性集的交换机支持 IPv6 静态路由、IPv6 RIP 和 IPv6 OSPF。运行 LAN base 特性集的交换机则只支持 IPv6 主机功能。

理解 IPv6

IPv4 用户如果迁移到 IPv6 环境中，就可以得到诸如端到端的安全、服务质量（QoS）和全局唯一地址等服务。IPv6 庞大的地址空间降低了人们对私有地址空间的需求，位于网络边缘的边界路由器不必处理大量的网络地址转换（NAT）操作。

要想进一步了解 Inspur 系统实施 IPv6 的相关信息，可以访问：

http://www.icntnetworks.com

要想进一步了解本章中介绍的 IPv6 和其他相关特性，可以参阅《INOS IPv6 配置库》。

用户可以通过 icntnetworks.com 中的搜索栏来查找 Inspur INOS 软件的技术文档。例如，如果用户希望了解关于静态路由的信息，可以在搜索栏中输入“实施 IPv6 静态路由”来了解关于静态路由的信息。

IPv6 地址

交换机只支持 IPv6 单播地址。它不支持站点本地单播地址或者任意播地址。

IPv6 的 128 位地址是通过一系列 8 个用英文冒号分割的十六进制数来表示的，其中每个十六进制数的长度为 16 位二进制数，其格式为 n:n:n:n:n:n:n:n。下面是一个 IPv6 地址的示例：

2031:0000:130F:0000:0000:09C0:080F:130B

为简化期间，每部分中的前导 0 可以省略。IPv6 地址每一段中有没有前导 0 都是一样的，所以上面的 IPv6 可以简化为：

2031:0:130F:0:0:9C0:80F:130B

我们也可以使用两个英文冒号（::）来代表连续多个全 0 的十六进制字段，但这种双冒号的简化表示法每个地址中只能使用一次：

2031:0:130F::09C0:080F:130B

要想进一步了解关于 IPv6 地址格式、地址类型和 IPv6 数据包头部的信息，可以参阅 icntnetworks.com 中，“Inspur INOS IPv6 配置库”一文中的“实施 IPv6 编址和基本的连通性”一章。

在“关于实施 IPv6 基本连通性的信息”一章中，下面几节的内容适用于交换机环境：

- IPv6 地址格式
- IPv6 地址类型：单播
- IPv6 地址类型：组播
- IPv6 地址输出显示
- 简化的 IPv6 数据包头部

支持的 IPv6 单播路由特性

在这一部分中，我们会介绍交换机支持的 IPv6 协议特性：

交换机可以通过 IPv6 版 RIP（路由信息协议）、和 OSPF（开放式最短路径优先）协议第 3 版来提供 IPv6 路由功能。交换机支持最多 16 条等价路由，也可以用线速并行转发 IPv4 数据帧和 IPv6 数据帧。

128 位宽单播地址

交换机支持可汇总的全局单播地址和链路本地单播地址。交换机不支持站点本地单播地址。可汇总全局单播地址是包含可汇总全局单播前缀的 IPv6 地址。这种地址结构可以实现对路由前缀的严格汇总，因此可以限制全局路由表中路由条目的数量。这类地址用于可以对整个机构的地址进行汇总，或者最终连接到互联网服务提供商的链路上。

这类地址是通过全局路由前缀、子网 ID 和接口 ID 进行定义的。当前的全局单播地址是从二进制值 001（2000::/3）开始的地址范围进行分配的。前缀为 2000::/3（001）到 E000::/3（111）的地址必须包含 EUI（扩展唯一标识符）-64 格式的 64 位接口标识符。

链路本地单播地址可以在任何接口上自动进行配置，这类地址由链路本地前缀 FE80::/80（1111 1110 10）和修改的 EUI 格式的接口标识符构成。链路本地地址会用于邻居发现协议（NDP）和无状态地址自动配置的进程。本地链路上的节点会使用链路本地地址，它们不需要配置全局唯一地址就可以实现通信。IPv6 路由器不会将以链路本地地址作为源或目的的数据包转发到其他链路上。

要想进一步了解相关信息，可以参阅 icntnetworks.com 中，“Inspur INOS IPv6 配置库”一文中的“实施 IPv6 编址和基本的连通性”一章。

IPv6 DNS

IPv6 支持 DNS 域名-地址和地址-域名查询进程中的 DNS（域名系统）记录类型。DNS AAAA 资源记录类型支持 IPv6 地址，它相当于 IPv4 中的 A 地址类型。交换机支持针对 IPv4 和 IPv6 地址执行 DNS 解析。

IPv6 单播的路径 MTU 发现

交换机支持向 IPv6 节点通告系统最大传输单元（MTU），也支持路径 MTU 发现。路径 MTU 发现可以让主机动态发现一条给定数据路径中各个链路的 MTU 值，并且以此对 MTU 进行调整。在 IPv6 环境中，如果路径中一条链路的 MTU 小于数据包的大小，那么数据包的源设备就会对数据进行分片。

ICMPv6

IPv6 版的互联网控制消息协议（ICMP）可以在进行处理和执行错误诊断时，创建各类错误消息（譬如 ICMP 目的地址不可达消息）来报告网络中的错误。在 IPv6 环境中，ICMP 数据包

也会用于邻居发现协议和路径 MTU 发现功能。

邻居发现

交换机支持 IPv6 的 NDP（这是一项运行在 ICMPv6 之上的协议），也支持给那些不支持 NDP 的 IPv6 工作站静态配置邻居条目。IPv6 邻居发现进程会使用 ICMP 消息和请求节点组播地址来判断同一个网络（本地链路）中邻居设备的链路层地址，以便测试邻居的可达性，并且追踪邻居路由器。

交换机支持对那些掩码长度少于 64 位的路由执行 ICMPv6 重定向，但不支持对那些掩码长度大于 64 位的主机路由或汇总路由进行 ICMP 重定向。

邻居发现压制（throttling）可以确保交换机在处理下一跳转发信息，以路由 IPv6 数据包时，CPU 不至于出现过载的情况。当下一跳就是交换机正在主动尝试解析的邻居时，交换机就会丢弃 IPv6 数据包。这可以避免进一步增加 CPU 的负担。

默认路由器优先级

交换机支持 IPv6 默认路由器优先级（DRP），这是路由器通告消息的一个扩展部分。DRP 可以提升主机的功能，让主机能够选择合适的路由器，这种技术特别适合用于那些多宿主的主机，且路由器处于不同链路的情形。交换机不支持 RFC 4191 中定义的路由信息可选项。

IPv6 主机会维护一个默认路由器列表，主机会从列表中选择一台路由器来向其转发去往链路外目的地址的流量。被选中为某个目的地址转发流量的路由器会被缓存到目的地址缓存当中。IPv6 NDP 规定，那些可达的或者很可能可达的路由器要优于那些可达性未知或者可达性成疑的路由器。对于可达或者很可能可达的路由器，NDP 既可以每次都选择相同的路由器，也可以在路由器列表中循环选用。通过 DRP，用户可以对一台 IPv6 主机进行配置，让某一台路由器的优先级比另一台路由器优先级高，当然前提是这两台路由器都要是可达或者很可能可达的路由器。

要想进一步了解 IPv6 DRP，可以参阅 icntnetworks.com 中的“Inspur INOS IPv6 配置库”一文。

IPv6 无状态自动配置与重复地址检测

交换机会使用无状态自动配置来管理链路、子网和站点的地址变更，这包括对主机和移动 IP 地址的管理。主机会自动配置自己的链路本地地址，而启动节点会发送路由器请求消息来请求路由器通告，以配置接口地址。

要想进一步了解关于自动配置与重复地址检测的信息，可以参阅 icntnetworks.com 中，“Inspur INOS IPv6 配置库”一文中的“实施 IPv6 编址和基本的连通性”一章。

IPv6 应用

交换机支持下列 IPv6 应用：

Ping、traceroute、Telnet 和 TFTP；

通过 IPv6 传输来实现 SSH；

通过 IPv6 传输来访问 HTTP 服务器；

通过 IPv4 传输来对 AAAA 执行 DNS 解析；

思科发现协议（CDP）支持 IPv6 地址

要想进一步了解关于管理这些应用的信息，可以参阅 icntnetworks.com 中的“Inspur INOS IPv6 配置库”一文。

使用 DHCP 来分配 IPv6 地址

DHCPv6 可以通过 DHCP 服务器来向 IPv6 客户端传输配置参数，其中包括 IPv6 网络地址。地址分配特性会根据主机所连接的网络，用正确的前缀来分配不重复的地址。分配的地址可以来自于一个或多个地址池。还有一些其他的可选项（如默认域和 DNS 名称服务器地址）也可以由 DHCP 服务器传输给客户端。地址池中的地址可以分配给一个特定的接口或者多个接口，DHCP 服务器也可以自动找到正确的地址池。

要想进一步了解关于管理这些应用的信息，可以参阅《Inspur INOS IPv6 配置指南》。
本文仅仅描述了 DHCPv6 的地址分配方式。要想进一步了解关于配置 DHCPv6 客户端、服务器或中继代理功能的信息，可以访问 icntnetworks.com，参阅“Inspur INOS IPv6 配置库”中的“实施 IPv6 DHCP”一章。

IPv6 静态路由

静态路由是手动进行配置的、用来明确定义两台网络设备之间路径的路由条目。静态路由适用于那些只有一条路径通往外部网络的小型网络环境，也可以在大型网络中的某些流量类型提供安全性保护。
要想进一步了解关于静态路由的信息，可以访问 icntnetworks.com，参阅“Inspur INOS IPv6 配置库”中的“实施 IPv6 静态路由”一章。

IPv6 RIP

IPv6 版的路由信息协议（RIP）是一种距离矢量型协议，这种协议会使用跳数作为路由度量值。这种路由协议可以支持 IPv6 地址和前缀，它会用所有 RIP 路由器组播组地址 FF02::9 作为 RIP 更新消息的目的地址。
要想进一步了解关于 IPv6 RIP 的信息，可以访问 icntnetworks.com，参阅“Inspur INOS IPv6 配置库”中的“实施 IPv6 RIP”一章。

IPv6 OSPF

运行 IP Base 镜像特性集的交换机支持 IPv6 版的最短路径优先（OSPF）协议，这是一种链路状态型 IP 协议。要想进一步了解相关的信息，可以访问 icntnetworks.com，参阅“Inspur INOS IPv6 配置库”。

配置 IPv6 HSRP

HSRP 可以为路由 IPv6 流量提供路由冗余，让流量不再依赖于某一台路由器进行转发。IPv6 主机会通过 IPv6 邻居发现路由器通告消息来学习网络中可用的路由器。这些信息是路由设备组播周期性进行发送的，或者由主机请求发送的。
每个 HSRP IPv6 组都有一个虚拟的 MAC 地址（这个 MAC 地址取自于 HSRP 组的编号）和一个虚拟的 IPv6 链路本地地址（在默认情况下取自于 HSRP 虚拟 MAC 地址）。当 HSRP 组处于活跃状态时，设备就会使用 HSRP 虚拟 IPv6 链路本地地址来周期性地发送消息。而当这个 HSRP 组不再处于活跃状态时，在最后一个消息更结束之后，设备就不会再发送周期性的消息了。

注释： 在配置 IPv6 HSRP 时，必须在接口上启用 HSRP 第 2 版（HSRPv2）。

IPv6 EIGRP

交换机支持 IPv6 版的增强型内部网关路由协议（EIGRP）。用户可以在要运行这个协议的接口上配置该协议，而不必配置全局 IPv6 地址。运行 IP Lite 镜像的交换机只支持 EIGRPv6 末节路由。
在运行之前，IPv6 版 EIGRP 需要获得一个路由器 ID。隐式的路由器 ID 是从本地 IPv6 地址中提取出来的，所以往往每个 IPv6 节点都可以获得一个可用的路由器 ID。但 IPv6 版的 EIGRP 有可能会在一个只有 IPv6 节点的网络中运行，因此 IPv6 节点可能没有可用的 IPv6 路由器 ID。
要想进一步了解关于 IPv6 EIGRP 的信息，可以访问 icntnetworks.com，参阅“Inspur INOS IPv6 配置库”中的“实施 IPv6 EIGRP”一章。

EIGRPv6 末节路由

EIGRPv6 末节路由特性可以将被路由的流量移动到距离终端用户更近的位置，以此介绍对网络资源的占用。
在使用 EIGRPv6 末节路由的网络中，唯一允许向用户转发 IPv6 流量的路由就是那些穿越配

置了 EIGRPv6 末节路由的交换机，发往终端用户的路由条目。交换机会将被路由流量发送给那些配置为用户接口的端口，或者那些连接到其他设备的端口。

在使用 EIGRPv6 末节路由时，用户需要通过配置让转发路由器和远程路由器使用 EIGRPv6，同时只将这台交换机配置为末节设备。只有用户指定的路由才可以由交换机通告给其他设备。交换机会对所有针对汇总路由、直连路由和路由更新的查询消息作出响应。

当邻居设备接收到一个向它通告末节状态的数据包时，它不会向末节路由器查询任何路由信息，而连接有末节对等体的路由器也不会向这台对等体设备发送查询。末节路由器会依靠转发路由器将正确的更新消息转发给所有的对等体路由器。

在下图中，我们将交换机 B 配置为了一台 EIGRPv6 末节路由器。交换机 A 和交换机 C 都与 WAN 相连。交换机 B 会将直连路由、静态路由、重分布路由和汇总路由通告给交换机 A 和交换机 C。交换机 B 并不会将任何通过交换机 A 学习到的路由通告出去（反之亦然）。

图 8：EIGRP 末节路由器配置

Routed to WAN	路由到 WAN
Switch A	交换机 A
Switch B	交换机 B
Switch C	交换机 C
Host A	主机 A
Host B	主机 B
Host C	主机 C

要想进一步了解关于 IPv6 末节路由的信息，可以参阅“Inspur INOS IP 配置指南 卷 2：路由协议，第 12.4 版”中的“实施 IPv6 EIGRP”一章。

基于 IPv6 的 SNMP 和系统日志

要想让网络同时支持 IPv4 和 IPv6 流量，IPv6 网络管理需要能够同时使用 IPv6 和 IPv4 实现流量传输。基于 IPv6 的系统日志支持对地址数据类型进行传输。

基于 IPv6 的 SNMP 和系统日志可以提供下列特性：

- 同时支持 IPv4 和 IPv6；
- 可以通过 IPv6 传输 SNMP 流量，并且对 SNMP 代理进行修改，以支持向 IPv6 主机发送 trap；
- 可以让与 SNMP 和系统日志相关的 MIB 支持 IPv6 的编址方式；
- 可以将 IPv6 配置为 trap 的接收方。

为了能够支持通过 IPv6 传输流量，SNMP 对当前的 IP 传输映射关系进行修改，使其能够同时支持 IPv4 和 IPv6。下列 SNMP 操作支持 IPv6 传输管理：

- 开放默认设置的用户数据报协议（UDP）SNMP 套接字；
- 提供一种新的、称为 *SR_IPv6_传输*的传输机制；
- 通过 IPv6 发送 SNMP 通告消息；
- 支持针对 IPv6 传输使用 SNMP 命名的访问控制列表；
- 支持使用 IPv6 传输的 SNMP 代理转发；
- 查看 SNMP 管理器特性可以兼容 IPv6 传输

要想进一步了解关于基于 IPv6 的 SNMP，包括其配置步骤，可以访问 icntnetworks.com，参阅“Inspur INOS IPv6 配置库”中的“管理基于 IPv6 的 Inspur INOS 应用”一章。

要想进一步了解关于基于 IPv6 的系统日志，包括其配置步骤，可以访问 icntnetworks.com，参阅“Inspur INOS IPv6 配置库”中的“实施 IPv6 编址和基本的连通性”一章。

基于 IPv6 的 HTTP

HTTP 客户端会向 IPv4 和 IPv6 HTTP 服务器发送请求消息，而服务器则会响应 IPv4 和 IPv6

HTTP 客户端发送的请求消息。包含 IPv6 地址的 URL 必须用每 16 位二进制就用冒号分隔的十六进制表示法表示。

Accept 套接字在调用时会选择 IPv4 或 IPv6 地址族。这里的 accept 套接字可以是一个 IPv4 套接字或者 IPv6 套接字。监听套接字会继续监听指示连接的 IPv4 和 IPv6 信号。IPv6 监听套接字会绑定一个 IPv6 通配符地址。

底层的 TCP/IP 协议栈支持双栈环境。HTTP 依靠 HTTP 协议栈和套接字来处理网路层的交互信息。

在建立 HTTP 连接之前，客户端和服务器之间必须建立了基本的网络连接（**ping**）。

要想进一步了解相关信息，可以访问 icntnetworks.com，参阅“Inspur INOS IPv6 配置库”中的“管理基于 IPv6 的 Inspur INOS 应用”一章。

不支持的 IPv6 单播路由特性

交换机不支持下列 IPv6 特性：

- IPv6 虚拟专用网络（VPN）路由器与转发（VRP）表；
- 转发以站点本地地址为目的的 IPv6 数据包；
- 隧道协议，如 IPv4-IPv6 隧道，或 IPv6-IPv4 隧道
- 用交换机充当 IPv4-IPv6 隧道，或 IPv6-IPv4 隧道协议的隧道端点；
- IPv6 单播逆向路径转发；
- IPv6 Web 缓存通信协议（WCCP）

IPv6 特性的限制

鉴于 IPv6 是在交换机硬件中实施的，硬件内存中的 IPv6 压缩地址存在一些限制。这些硬件限制导致有些特性的功能会受到影响，因此这些特性的使用也会受到限制：

这些限制包括：

- 交换机不能在硬件中转发 SNMP 封装的 IPv6 数据包。这类数据包只能在软件中转发；
- 交换机不能在硬件中对根据源地址路由的 IPv6 数据包应用 QoS 分类（classification）。

IPv6 与交换机堆栈

交换机支持以堆栈的形式执行 IPv6 转发，堆栈的主交换机也支持 IPv6 主机功能。堆栈主交换机可以运行 IPv6 单播路由协议，并且计算路由表。它们会接收路由表，并且创建硬件的 IPv6 路由以便对数据包进行转发。堆栈主交换机也可以运行所有的 IPv6 应用。

注释： 要在堆栈中路由 IPv6 数据包，那么堆栈中的所有交换机都要运行 IP Base 特性集。

如果一台新的交换机成为了堆栈的主设备，那么它就会重新计算 IPv6 路由表，并且将路由表分发给其他成员交换机。在堆栈选举出了新的主设备并且重启时，交换机堆栈并不会转发 IPv6 数据包。此时堆栈的 MAC 地址会发送变化，这也会导致 IPv6 地址出现变化。在通过接口配置命令 **ipv6 address** *ipv6-prefix/prefix length* **eui-64** 用扩展为标识符（EUI）设置堆栈 IPv6 地址时，这个接口的 IPv6 地址就是基于接口 MAC 地址生成的。

如果用户在堆栈上配置了永久 MAC 地址特性，此时堆栈主设备发生了变化，那么堆栈 MAC 地址会有大约 4 分钟的时间不会变更。

下面是 IPv6 堆栈主设备和成员设备的功能：

- 堆栈主设备：
  - 运行 IPv6 路由协议；
  - 生成路由表；
  - 将路由表分发给使用 dCEFv6 的堆栈成员设备
  - 运行 IPv6 主机功能和 IPv6 应用。
- 堆栈成员设备（必须运行 IP Services 特性集）：
  - 接收堆栈主设备发来的 CEFv6 路由表；

- 将路由条目写入硬件当中；

注释： 在堆栈中，在数据包上没有携带 IPv6 扩展头部可选项，同时堆栈中的交换机也没有全部耗尽硬件资源的情况下，IPv6 数据包是在硬件中跨交换机进行路由转发的。

- 在重新选举主设备时清空 CEFv6 表。

默认的 IPv6 配置

表 19：默认的 IPv6 配置

特性	**默认设置**
SDM 模版	高级桌面。默认为高级模版
IPv6 路由	全局禁用但在所有接口上启用
CEFv6 或 dCEFv6	禁用（IPv4 CEF 和 dCEF 默认是启用的） **注释**：在用户启用 IPv6 路由时，CEFv6 和 dCEFv6 也会自动启用
IPv6 地址	未配置

配置 IPv6 地址与启用 IPv6 路由（CLI 界面配置方法）

在这一节中，我们会描述如何向一个三层接口分配 IPv6 地址，以及如何在交换机上全局转发 IPv6 流量。

用户在交换机上配置 IPv6 之前，应该考虑下面这些指导方针：

- 并不是本章中介绍的所有特性交换机都可以提供支持。详见不支持的 IPv6 单播路由特性；
- 在接口配置命令 **ipv6 address** 中，用户必须使用冒号分隔的十六进制这种格式来输入 *ipv6-address*（IPv6 地址）和 *ipv6-prefix*（IPv6 前缀）这两个变量。另外，*prefix-length*（前缀长度）这个变量（斜线/后面的参数）是一个十进制数，这个数代表了前缀是由前多少位连续的地址所组成的（也就是说，地址的网络位占多少位）。

要让一个接口转发 IPv6 流量，用户必须在这个接口上配置一个全局 IPv6 地址。在接口上配置 IPv6 地址之后，接口会自动配置上一个链路本地地址，同时这个接口会启用 IPv6 协议。

用户配置的这个接口会自动加入这条链路上必须加入的那些组播组，其中包括：

- 每个分配给该接口的单播地址，所对应的请求节点组播组 FF02:0:0:0:0:1:ff00::/104；
- 全节点链路本地组播组 FF02::1；
- 全路由器链路本地组播组 FF02::2。

要删除一个接口的 IPv6 地址，需要执行接口配置命令 **no ipv6 address** *ipv6-prefix/prefix length* **eui-64** 或 **no ipv6 address** *ipv6-address* **link-local**。要移除接口上所有手动配置的地址，可以直接在该接口的配置模式下输入命令 **no ipv6 address**，不带任何参数。要让一个尚未手动配置 IPv6 地址的接口停止处理 IPv6 流量，可以在接口配置模式下输入命令 **no ipv6 enable**。要在全局禁用 IPv6 路由，可以在全局配置模式下输入命令 **no ipv6 unicast-routing**。

要想进一步了解关于配置 IPv6 路由的信息，可以参阅 icntnetworks.com 中，“Inspur INOS IPv6 配置库”一文中的“实施 IPv6 编址和基本的连通性”一章。

用户可以从特权 EXEC 模式中，按照下面的步骤为一个三层接口分配 IPv6 地址，并启用 IPv6 路由转发：

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 2	**sdm prefer dual-ipv4-and-ipv6** { **advanced** | **vlan**} 示例： `Device(config)#` **sdm prefer dual-ipv4-and-ipv6 default**	选择一个支持 IPv4 和 IPv6 的 SDM 模版。 • **advanced**：设置交换机，让其使用默认模版来平衡系统资源； • **vlan**：在不支持用硬件执行路由转发的交换机上最大化 VLAN 的配置。 **注释**：所有许可证版本都支持使用 advanced 这个参数。但只有 LAN Base 许可证的交换机支持使用 VLAN 模版这个参数。
步骤 3	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 4	**reload** 示例： `Device#` **reload**	重启操作系统
步骤 5	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 6	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet 1/0/1**	进入接口配置模式，指定要进行配置的三层接口。这个接口既可以是物理接口，也可以是交换机虚拟接口（SVI），或者三层 EtherChannel 接口
步骤 7	**no switchport** 示例： `Device(config-if)#` **no switchport**	清除这个接口的二层配置模式（如果这是一个物理接口的话）
步骤 8	配置下列命令之一： • **ipv6 address** *ipv6-prefix/prefix length* **eui-64** • **ipv6 address** *ipv6-address/prefix length* • **ipv6 address** *ipv6-address* **link-local** • **ipv6 enable**	• 设置一个低 64 位为扩展唯一标识符（EUI）的全局 IPv6 地址。仅设置网络前缀；交换机会自动从自己的 MAC 地址中计算出后面的 64 位地址。 设置该地址会让该接口开始处理 IPv6 流量； • 在接口上设置一个链路本地地址，以替代在该接口启用 IPv6 时，接口自动配置的那个链路本地地址。执行这一设置会让该接

	• **ipv6 address***WORD* • **ipv6 address***autoconfig* • **ipv6 address***dhcp* 示例： `Device(config-if)#` **ipv6 address 2001:0DB8:c18:1::/64 eui 64** `Device(config-if)#` **ipv6 address 2001:0DB8:c18:1::/64** `Device(config-if)#` **ipv6 address 2001:0DB8:c18:1:: link-local** `Device(config-if)#` **ipv6 enable**	口开始处理 IPv6 流量； • 在接口上自动配置 IPv6 链路本地地址，并让该接口开始处理 IPv6 流量。链路本地地址只能用来与连接在同一条链路上的节点进行通信。
步骤 9	**exit** 示例： `Device(config-if)#` **exit**	返回全局配置模式
步骤 10	**ip routing** 示例： `Device(config)#` **ip routing**	在这台交换机上启用 IP 路由转发功能
步骤 11	**ipv6 unicast-routing** 示例： `Device(config)#` **ipv6 unicast-routing**	启用对 IPv6 单播数据包的转发
步骤 12	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 13	**show ipv6 interface** *interface-id* 示例： `Device#` **show ipv6 interface gigabitethernet 1/0/1**	验证前面所作的设置
步骤 14	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

配置 IPv6 ICMP 速率限制（CLI 界面配置方法）

设备默认就会启用 ICMP 速率限制特性，此时设备采用的错误消息默认间隔时间为 100 毫秒，令牌桶大小（即桶中最多可以储存多少个令牌）为 10。
用户可以从特权 EXEC 模式中，按照下面的步骤来修改 ICMP 速率限制参数：
具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 2	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet 1/0/1**	进入接口配置模式，指定要设置 DRP 的三层接口。
步骤 3	**ipv6 nd router-preference** {**high** | **medium** | **low**} 示例： `Device(config-if)#` **ipv6 nd router-preference medium**	在这个交换机接口上设置 DRP
步骤 4	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式
步骤 5	**show ipv6 interface** 示例： `Device#` **show ipv6 interface**	验证前面所作的设置
步骤 6	**copy running-config startup-config** 示例： `Device#` **copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

查看 IPv6

要想了解这些命令的完整语法结构及用法，可以参阅 Inspur INOS 命令参考手册。
表 20：监控 IPv6 的相关命令

命令	目的
show ipv6 access-list	显示汇总的访问控制列表
show ipv6 cef	显示 IPv6 Inspur 快速转发
show ipv6 interface*interface-id*	显示 IPv6 接口状态与配置
show ipv6 mtu	显示各个目的缓存的 IPv6 MTU 值
show ipv6 neighbors	显示 IPv6 邻居缓存条目
show ipv6 ospf	显示 IPv6 OSPF 信息
show ipv6 prefix-list	显示 IPv6 前缀列表
show ipv6 protocols	显示这台交换机上使用的 IPv6 路由协议列表
show ipv6 rip	显示 IPv6 RIP 路由协议的状态
show ipv6 route	显示 IPv6 路由表条目
show ipv6 routers	显示本地 IPv6 路由器
show ipv6 static	显示 IPv6 静态路由
show ipv6 traffic	显示 IPv6 流量的统计数据

表 21：显示 EIGRP IPv6 信息的命令

命令	目的
show ipv6 eigrp [*as-number*] *interface*	显示配置了 IPv6 EIGRP 的接口的相关信息
show ipv6 eigrp [*as-number*] *neighbor*	显示通过 IPv6 EIGRP 发现的邻居
show ipv6 interface[*as-number*] *traffic*	显示发送和接收的 IPv6 EIGRP 数据包数量
show ipv6 eigrptopology [*as-number* | *ipv6-address*][**active** | **all-links** | **detail-links** | **pending** | **summary** | **zero-successors** | **Base**]	显示 IPv6 拓扑表中的 EIGRP 条目

配置 IPv6 单播路由的示例

配置 IPv6 地址并启用 IPv6 路由转发：示例

这个示例显示了如何对链路本地地址和基于 IPv6 前缀 2001:0DB8:c18:1::/64 的全局地址启用 IPv6。其中 EUI-64 为接口 ID 会用来作为这两个地址的后 64 位地址。在这个示例中，我们也提供了 EXEC 命令 **show ipv6 interface** 的输出信息，以说明接口 ID（20B:46FF:FE2F:D940）是如何添加在接口链路本地前缀（FE80::/64）后面的。

```
Device(config)# ipv6 unicast-routing
Device(config)# interface gigabitethernet1/0/11
Device(config-if)# no switchport
Device(config-if)# ipv6 address 2001:0DB8:c18:1::/64 eui 64
Device(config-if)# end
Device# show ipv6 interface gigabitethernet1/0/11
GigabitEthernet1/0/11 is up, line protocol is up
IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
Global unicast address(es):
2001:0DB8:c18:1:20B:46FF:FE2F:D940, subnet is 2001:0DB8:c18:1::/64 [EUI]
```

```
Joined group address(es):
FF02::1
FF02::2
FF02::1:FF2F:D940
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 advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
Hosts use stateless autoconfig for addresses.
```

配置 IPv6 ICMP 速率限制：示例

这个示例显示了如何将 IPv6 ICMP 错误消息显示的时间间隔设置为 50 毫秒，将令牌桶大小设置为 20 个令牌。

```
Device(config)#ipv6 icmp error-interval 50 20
```

配置 IPv6 静态路由：示例

这个示例显示了如何用出站接口配置一条浮动静态路由，并且将其管理距离设置为 130：

```
Device(config)# ipv6 route 2001:0DB8::/32 gigabitethernet2/0/1 130
```

查看 IPv6：示例

这个示例显示了特权 EXEC 命令 **show ipv6 interface** 的输出信息：

```
Device# show ipv6 interface
Vlan1 is up, line protocol is up
IPv6 is enabled, link-local address is FE80::20B:46FF:FE2F:D940
Global unicast address(es):
3FFE:C000:0:1:20B:46FF:FE2F:D940, subnet is 3FFE:C000:0:1::/64 [EUI]
Joined group address(es):
FF02::1
FF02::2
FF02::1:FF2F:D940
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 advertised retransmit interval is 0 milliseconds
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
<output truncated>
```

配置 IPv6 ACL

IPv6 ACL 的前提条件

用户可以通过创建 IPv6 访问控制列表（ACL）并且将它们应用到接口的方式来过滤 IPv6（IP 第 6 版）流量，这种方法与创建和应用 IPv4（IP 第 4 版）命名 ACL 的方式相当类似。如果交换机运行的是 IP base 特性集，那么用户也可以创建和应用入站路由器 ACL 来过滤三层管理流量。

IPv6 ACL 的限制条件

在 IPv4 环境中，用户可以配置标准和扩展的编号 IP ACL、命名的 IP ACL 和 MAC ACL。而 IPv6 只支持命名的 ACL。

本设备支持大多数 Inspur INOS 支持的 IPv6 ACL，但下列 ACL 是例外：

- 本设备不支持匹配下列关键字：**flowlabel**、**routing header** 和 **undetermined-transport**；
- 本设备不支持自反 ACL（也就是不支持 **reflect** 这个关键字）；
- 本设备不支持对 IPv6 数据帧应用基于 MAC 的 ACL；
- 在配置 ACL 时，设备对于用户在 ACL 中输入的关键字没有限制，无论配置该命令的平台是否支持这些关键字。在将 ACL 应用到一个需要硬件转发的接口（物理端口或 SVI）时，设备会判断接口是否支持这个 ACL。如果不支持，那么设备会拒绝在接口上应用这个 ACL；
- 如果用户将 ACL 应用到了一个接口，接下来用户又准备向这个 ACL 中添加一条包含设备不支持的关键字的 ACE（访问控制条目），那么设备不会允许用户将这条 ACE 添加到这个接口上应用的 ACL。

关于 IPv6 ACL 的信息

访问控制列表是一系列用来限制访问某个接口的规则。用户需要在设备上配置 ACL，并且将

ACL 应用到管理接口、AP-管理员接口、任何动态接口、或者应用到控制器中央处理（CPU）以控制所有去往 CPU 的流量。

用户可以创建一个预认证 ACL 来执行 web 认证。这类 ACL 的作用是在认证完成之前，就放行某些类型的流量。

IPv6 ACL 支持的选项与 IPv4 ACL 相同，其中包括源、目的、源端口和目的端口。

注释： 用户可以在网络中通过阻塞 IPv6 流量方式来单独启用 IPv4 流量。也就是说，用户可以配置一个过滤所有 IPv6 流量的 IPv6 ACL，并且将它引用到某个 WLAN 或者所有 WAN 上。

理解 IPv6 ACL

交换机支持两种类型的 IPv6 ACL：

- 支持在三层接口上针对入站流量和出站流量配置 IPv6 路由器 ACL，这里所说的三层接口既可以是路由端口，也可以是交换机虚拟接口（SVI）或者三层 EtherChannel。IPv6 路由器 ACL 只能应用在那些路由 IPv6 数据包的接口上；
- 支持在二层接口上针对入站流量配置 IPv6 端口 ACL。IPv6 端口 ACL 会作用于所有进入这个接口的 IPv6 数据包；

运行 IP base 特性集的交换机只支持入站方向的路由器 IPv6 ACL。它不支持端口 ACL 或出站 IPv6 路由器 ACL。

注释： 如果用户配置了设备不支持的 IPv6 ACL，那么设备就会弹出错误消息，用户所作的配置也不会生效。

交换机不支持针对 IPv6 流量使用 VLAN ACL（即 VLAN map）。

用户可以同时在一个接口上应用 IPv4 ACL 和 IPv6 ACL。IPv6 端口 ACL 的优先级高于路由器 ACL，这一点 IPv4 ACL 和 IPv6 ACL 是一样的。

- 当一个 SVI 上同时应用了入站方向的路由器 ACL 和入站方向的端口 ACL 时，如果应用了 ACL 的那些端口接收到数据包，设备就会使用端口 ACL 来过滤数据包。如果其它一些端口接收到的被路由 IP 数据包，设备则会使用路由器 ACL 进行过滤。其余数据包则不会用 ACL 进行过滤；
- 当一个 SVI 上同时应用了出站方向的路由器 ACL 和入站方向的端口 ACL 时，如果应用了 ACL 的那些端口接收到数据包，设备就会使用端口 ACL 来过滤数据包。而出站的被路由 IPv6 数据包则会通过路由器 ACL 进行过滤。其余数据包则不会用 ACL 进行过滤。

注释： 只要一个接口上应用了端口 ACL（无论 IPv4 ACL、IPv6 ACL 还是 MAC ACL），那么设备就会用这个端口 ACL 来过滤数据包，而该端口所在的 SVI 上所应用的任何路由器 ACL 设备此时都会被设备忽略。

ACL 的类型

每用户（Per User）IPv6 ACL

对于每用户 ACL 来说，全部访问控制条目（ACE）都要以文本的形式配置在 ACS 上；

ACE 不是配置在控制器上的。ACE 会通过 ACCESS-Accept 这个属性发送给设备，设备会直接将它应用于客户端。设备不支持在出站方向部署每用户 ACL。

过滤器 ID IPv6 ACL

对于过滤器 ID ACL，全部 ACE 和 acl name(filter-id)都要配置在设备上，只有 filter-id 要配置在 ACS 上。ACS 会将 filter-id 放在 ACCESS-Accept 属性中发送给设备，而设备会使用 filter-id 来

查找 ACE，然后将 ACE 应用于客户端。当客户端在二层漫游到另一台外来的设备上时，只有 filter-id 会通过 Handoff 消息发送给那台设备。设备不支持在出站方向针对不同用户配置 ACL 来执行过滤。用户还需要提前在那台外来设备上配置 filter-id 和 ACE。

可下载的 IPv6 ACL

对于可下载 ACL（dACL）来说，全部 ACE 和 dacl-名称都只能配置在 ACS 上。

注释： 控制器上并不能配置任何 ACL。

ACS 会将 dacl-名称通过 ACCESS-Accept 属性发送给设备，而设备则会提取出 dacl 名称，再把 dACL 名称通过 access-request 属性发回给 ACS，来获取 ACE。

ACS 会使用 access-accept 属性来响应设备请求的 ACE。而外来设备会通过 dacl 名称来联系 ACS 服务器获取 ACE。

配置 IPv6 ACL

要过滤 IPv6 流量，需要执行下面的步骤：

在开始前

在配置 IPv6 ACL 之前，用户必须从 IPv4 SDM 模版和 IPv6 SDM 模版中选择其一。

总步骤

1 创建一个 IPv6 ACL，并且进入 IPv6 访问列表配置模式；

2 配置 IPv6 ACL 来过滤（阻塞）或放行（允许）流量；

3 将 IPv6 ACL 应用到需要过滤流量的接口上；

4 将 IPv6 ACL 应用到一个接口上。对于路由器 ACL，用户还必须在应用 ACL 的接口上配置 IPv6 地址。

具体步骤

	命令与操作	目的
步骤 1	创建一个 IPv6 ACL，并且进入 IPv6 访问列表配置模式	
步骤 2	配置 IPv6 ACL 来过滤（阻塞）或放行（允许）流量	
步骤 3	将 IPv6 ACL 应用到需要过滤流量的接口上	
步骤 4	将 IPv6 ACL 应用到一个接口上。对于路由器 ACL，用户还必须在应用 ACL 的接口上配置 IPv6 地址	

默认的 IPv6 ACL 配置

默认状态设备上没有配置和应用 IPv6 ACL。

与其他特性与交换机的互动

- 如果用户配置了一条 IPv6 路由器 ACL 来拒绝数据包，那么路由器就不会路由这个数据包。这个数据包的副本会被发送到互联网控制消息协议（ICMP）队列中，给该数据帧生成一条 ICMP 不可达消息；
- 如果由于设备上配置了端口 ACL 去丢弃一个桥接数据帧，那么设备就不会去桥接这个数据帧；
- 用户可以同时在交换机或者交换机集群上配置 IPv4 和 IPv6 ACL，然后将 IPv4 和 IPv6 ACL

同时应用在一个接口上。每个 ACL 的命名必须是唯一的；如果用户要给一个 ACL 配置一个已经配置过的名称，那么系统就会显示错误消息。

- 用户要使用不同的命令来创建 IPv4 和 IPv6 ACL，并将它们关联到同一个二层或三层接口上。如果在关联 ACL 时用户输入的命令不正确（比如在将 IPv6 ACL 关联到接口上时输入了 IPv4 的命令），那么用户就会看到一条错误消息；
- 用户不能使用 MAC ACL 来过滤 IPv6 数据帧。MAC ACL 只能过滤非 IP 数据帧；
- 如果硬件的内存已满，那么对于用户继续配置的 ACL，相关的丢包操作会交由 CPU 来执行，ACL 也会应用到软件中。如果硬件已经满，console 就会显示一条消息，显示 ACL 已经被卸载，这个接口会开始丢弃数据包。

注释： 接口只会丢弃那些无法添加的 ACL（IPv4、IPv6 或 MAC）所对应类型的数据包。

如何配置 IPv6 ACL

创建 IPv6 ACL

用户可以从特权 EXEC 模式中，通过下列步骤来创建 IPv6 ACL：

总步骤

1. configure terminal

2. ipv6 access-list *acl_name*

3. {deny|permit} protocol

4. {deny|permit} tcp

5. {deny|permit} udp

6. {deny|permit} icmp

7. end

8. show ipv6 access-list

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**ipv6 access-list** *acl_name* 示例： `ipv6 access-list access-list-name`	用名称来定义 IPv6 访问列表，并进入 IPv6 访问列表配置模式
步骤 3	**{deny|permit} protocol** 示例： `{deny | permit} protocol {source-ipv6-prefix/prefix-length | any | host`	输入 deny 或 permit 来设置是阻塞还是拒绝那些满足条件的数据包。这些条件包括： • 在 protocol 部分，可以输入网络协议的名称或协议号：ahp、esp、icmp、ipv6、pcp、stcp、tcp 或 udp、或者在 0 到 255 之内可以代表一个 IPv6 协议号的整数；

	source-ipv6-address} [operator [port number]]{destination-ipv6-prefix/prefix-length | any |host destination-ipv6-address} [operator [port-number]][dscp value] [fragments][log] [log-input] [routing][sequence value] [time-range name]	• source-ipv6-prefix/prefix-length 或 destination-ipv6-prefix/prefix-length 是要设置匹配条件的那个源或目的 IPv6 网络，这个参数要用冒号分隔的 16 位值（即 RFC 2373 中定义的格式）来表示； • any 就是 IPv6 前缀::/0 的简称； • 在 host source-ipv6-address 或者 host destination-ipv6-prefix 部分，要输入要设置匹配条件的那个源或目的 IPv6 主机地址，这个参数要用冒号分隔的 16 位值（即 RFC 2373 中定义的格式）来表示； • （可选）在 operator 部分，设置比较指定协议的源或目的端口时使用的操作符。操作符包括 lt（小于，less than）、gt（大于，greater than）、eq（等于，equal）、neq（不等于，not equal）和范围 如果操作符跟在参数 source-ipv6-prefix/prefix-length 后面，那么它匹配的就是源端口。如果操作符跟在参数 destination-ipv6-prefix/prefix-length 后，那匹配的就是目的端口。 • （可选）port-number 是一个从 0 到 65535 之间的十进制数，或者一个 TCP 或 UDP 端口的名称。用户只能在过滤 TCP 流量时使用 TCP 端口名，在过滤 UDP 流量时使用 UDP 端口名； • （可选）输入 dscp value 用差分服务代码点值匹配每个 IPv6 数据包头部中流量类型（Traffic Class）字段中的流量类型值。取值范围是从 0 到 63； • （可选）输入 fragments 来校验非初始数据帧。只有协议为 ipv6 时，才可以看到 fragments 这个关键字； • （可选）输入 log 让系统向 console 发送关于匹配条目的数据包的日志消息。输入 log-input 在日志条目中包含入站接口。只有路由器 ACL 支持日志记录功能； • （可选）输入 routing 指定要被路由的 IPv6 数据包； • （可选）输入 sequence value 指定访问列表语句的序号；取值范围是 1 到 4294967295； • （可选）输入 time-range name 指定这条 deny 或 permit 语句应用的时间。
步骤 4	**{deny|permit} tcp**	（可选）定义 TCP 访问列表和访问条件。 对于传输控制协议（Transmission Control

	示例： {deny | permit} tcp {source-ipv6-prefix/prefix-length | any | hostsource-ipv6-address} [operator [port-number]]{destination-ipv6-prefix/prefix-length | any |hostdestination-ipv6-address} [operator [port-number]][ack] [dscp value][established] [fin] [log][log-input] [neq {port |protocol}] [psh] [range{port | protocol}] [rst][routing] [sequence value] [syn] [time-range name][urg]	Protocol）应输入 tcp。这些参数与步骤 3 中介绍的参数相同，此外还多了下列可选参数： • ack：Acknowledgement 位置位； • established：指已建立的连接。如果 TCP 数据报中已经将 ACK 或 RST 位置位，就会出现匹配； • fin：Finished 位置位，这表示发送方不会再发送更多数据； • neq {port |protocol}：只匹配那些不是指定端口号的流量； • psh：Push 功能位置位； • range {port | protocol}：只匹配端口号范围内的数据包； • rst：Reset 位置位； • syn：Synchronize 位置位； • urg：Urgent 指针位置位。
步骤 5	**{deny|permit} udp** 示例： {deny | permit} udp {source-ipv6-prefix/prefix-length | any | hostsource-ipv6-address} [operator [port-number]]{destination-ipv6-prefix/prefix-length | any | hostdestination-ipv6-address} [operator [port-number]][dscp value] [log][log-input] [neq {port |protocol}] [range {port |protocol}] [routing][sequence value][time-range name]	（可选）定义 UDP 访问列表和访问条件。 对于用户数据报协议（User Datagram Protocol）应输入 udp。这些参数与 TCP 中介绍的参数相同，但 operator [port-number]中设置的端口号或名称必须为 UDP 端口号或名称。此外，established 这项参数也不适用于 UDP。
步骤 6	**{deny|permit} icmp** 示例： {deny | permit} icmp {source-ipv6-prefix/prefix-length | any | hostsource-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any	（可选）定义 ICMP 访问列表和访问条件。 对于互联网控制消息协议（Internet Control Message Protocol）应输入 icmp。这些参数与大部分前面步骤中的参数相同，但 ICMP 有一些特殊的消息类型和代码参数，这些可选关键字的表意如下： • icmp-type：输入要过滤的 ICMP 消息类型，这是一个取值范围在 0 到 255 之间的整数； • icmp-code：输入要过滤的 ICMP 消息所对应的 ICMP 消息代码类型，这是一个取值范

	| hostdestination-ipv6-address} [operator [port-number]][icmp-type [icmp-code] |icmp-message] [dscpvalue] [log] [log-input] [routing] [sequence value][time-range name]	围在 0 到 255 之间的整数； • icmp-message：输入要过滤的 ICMP 消息所对应的 ICMP 消息类型和代码名称。用户如果希望查看 ICMP 消息类型名称和代码名称的列表，可以输入?来查看这个版本系统提供的提示信息
步骤 7	**end** 示例： Device(config-if)# **end**	返回特权 EXEC 模式。此外，用户也可以按下 **Ctrl-Z** 返回全局配置模式
步骤 8	**show ipv6 access-list** 示例： show ipv6 access-list	查看访问列表的配置
步骤 9	**copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

将 IPv6 ACL 应用到一个接口

在这一节中，我们会介绍如何将 IPv6 ACL 应用到一个网络接口上。用户可以将一个 IPv6 ACL 应用到二层或三层接口的出站或入站方向。用户也可以将 IPv6 ACL 应用于三层接口的入站管理流量。

用户可以从特权 EXEC 模式中，通过下列步骤将访问控制列表应用到一个接口：

总步骤

1. **configure terminal**
2. **interface** interface_id
3. **no switchport**
4. **ipv6 address** *ipv6_address*
5. **ipv6 traffic-filter** *acl_name*
6. **end**
7. **show running-config interface tenGigabitEthernet 1/0/3**
8. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 2	**interface** interface_id 示例：	指定要应用访问列表的二层接口（如 ACL 为端口 ACL）或三层交换机虚拟接口（如 ACL 为路由器 ACL），并进入接口的配置模式

	Device# interface interface-id	
步骤 3	**no switchport** 示例： Device# no switchport	将接口由（默认的）二层模式修改为三层模式（仅于要应用路由器 ACL 时）
步骤 4	**ipv6 address** *ipv6_address* 示例： Device# ipv6 address ipv6-address	在三层接口上配置一个 IPv6 地址（适用于路由器 ACL）。 **注释：**二层接口上不需要配置这条命令，如果用户已经在这个接口上配置了一个 IPv6 地址，也不需要配置这条命令
步骤 5	**ipv6 traffic-filter** *acl_name* 示例： Device# ipv6 traffic-filter access-list-name {in | out}	将访问列表应用于接口的入站或出站流量
步骤 6	**end** 示例： Device(config-if)# **end**	返回特权 EXEC 模式。此外，用户也可以按下 **Ctrl-Z** 返回全局配置模式
步骤 7	**show running-config interface tenGigabitEthernet 1/0/3** 示例： Device# show running-config interface tenGigabitEthernet 1/0/3 .... Building configuration .... Current configuration : 98 bytes ! interface TenGigabitEthernet1/0/3 switchport mode trunk ipv6 traffic-filter MyFilter out end	查看配置命令的汇总信息
步骤 8	**copy running-config startup-config**	（可选）将输入的条目保存到配置文件中

查看 IPv6 ACL

显示 IPv6 ACL

用户可以通过一条或几条特权 EXEC 命令来查看设备上配置的所有访问列表、IPv6 访问列表，或者某条指定的访问列表。

具体步骤

	命令或操作	目的
步骤 1	**show access-list** 示例： `Device# show access-lists`	显示设备上配置的所有访问列表
步骤 2	**show ipv6 access-list** *acl_name* 示例： `Device# show ipv6 access-list` *[access-list-name]*	显示所有 IPv6 访问列表或指定名称的访问列表

IPv6 ACL 的配置示例

示例：创建 IPv6 ACL

在这个示例中，用户配置了一个名为 CISCO 的 IPv6 访问列表。列表中的第 1 条 deny 条目会拒绝所有目的 TCP 端口号大于 5000 的数据包。第 2 条 deny 条目会拒绝源 UDP 端口号小于 5000 的数据包。此外，第 2 条 deny 语句也会将所有匹配的情形通过日志发送到 console 接口。列表中的第 1 条 permit 条目会放行所有 ICMP 数据包。列表中的第 2 条 permit 会放行所有其他的流量。第 2 条 permit 语句的存在十分必要，因为在每个 IPv6 访问列表的最后都有一个隐式的全部拒绝条目。

注释： 只有三层接口支持日志记录功能。

```
Device(config)# ipv6 access-list CISCO
Device(config-ipv6-acl)# deny tcp any any gt 5000
Device (config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Device(config-ipv6-acl)# permit icmp any any
Device(config-ipv6-acl)# permit any any
```

示例：应用 IPv6 ACL

这个示例显示了如何将名为 Inspur 的访问列表应用到一个三层接口的出站方向上：

```
Device(config)# interface TenGigabitEthernet 1/0/3
Device(config-if)# no switchport
```

```
Device(config-if)# ipv6 address 2001::/64 eui-64
Device(config-if)# ipv6 traffic-filter CISCO out
```

示例：查看 IPv6 ACL

这个示例显示了特权 EXEC 命令 **show access-lists** 的输出信息。输出信息中会显示出所有配置在交换机或交换机堆栈上的访问列表。

```
Device #show access-lists
Extended IP access list hello
10 permit ip any any
IPv6 access list ipv6
permit ipv6 any any sequence 10
```

这个示例显示了特权 EXEC 命令 show ipv6 access-list 的的输出信息。输出信息中只会显示交换机或交换机堆栈上配置的 IPv6 访问列表。

```
Device# show ipv6 access-list
IPv6 access list inbound
permit tcp any any eq bgp (8 matches) sequence 10
permit tcp any any eq telnet (15 matches) sequence 20
permit udp any any sequence 30
IPv6 access list outbound
deny udp any any sequence 10
deny tcp any any eq telnet sequence 20
```

第 2 层

配置生成树协议

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和

特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

STP 的限制条件

- 如果成为根设备必须的值小于 1，那么这台设备就无法成为根设备；
- 如果网络中由一些支持扩展系统 ID 的设备，和一些不支持扩展系统 ID 的设备组成，那么包含扩展系统 ID 的设备就不太可能成为根设备。每当 VLAN 编号大于运行老板系统的设备的优先级值，扩展系统 ID 会增加设备的优先级值；
- 对每个生成树实例来说，根设备都应该是骨干设备或分布层设备。不要将接入层设备配置为生成树的主用根；
- 用户不能混合使用 Inspur 3850 交换机和 Inspur 6650 交换机来建立交换机堆栈。

关于生成树协议的信息

生成树协议

生成树协议（STP）是一项二层管理协议，它的作用是在提供路径冗余的同时防止网络中出现环路。要想让二层以太网络能够正常工作，两个工作站之间只能有一条活动路径。终端站点之间有多条活动路径，网络中就会出现环路。如果网络中存在环路，那么终端工作站可能就会接收到重复消息。设备还有可能会通过多个二层接口学习到 MAC 地址。这种情况会导致网络不稳定。生成树操作对于终端工作站来说是透明的，也就是说终端工作站无法检测出它们是连接到了一个局域网段，还是连接到了一个包含多个网段的交换型局域网。
STP 会使用生成树算法从具有冗余连接的网络中选择出一台设备来充当生成树的根。这种算法可以依据端口在活动拓扑中发挥的作用来给端口指定一个角色。通过这种方法，算法可以在交换型二层网络中计算出最佳的无环路径。这些角色包括：

- 根：生成树拓扑中选举出来的转发端口；
- 指定：给每个交换型局域网段选举出来的转发端口；
- 替代：在生成树中提供通往根桥的替代路径的阻塞端口；
- 备份：在环回配置中的阻塞端口。

那些*所有*端口皆为指定角色或备份角色的设备即为根设备。而那些至少有一个端口为指定角色的设备则称为指定设备。
生成树会强制让冗余数据路径进入备份（即阻塞）状态。如果生成树中的一个网段失效，而网络中又存在冗余路径的话，那么生成树算法就会重新计算生成树拓扑，并且激活备份路径。设备会以固定的时间间隔发送和接收生成树数据帧，这些数据帧称为桥协议数据单元（BPDU）。设备不会把这些数据帧转发给其他设备，而是会通过这些数据帧来建立无环的路径。BDPU 中包含关于发送方设备及其端口的信息，其中包括设备和 MAC 地址、设备的优先

级、端口的优先级以及路径开销。生成树会使用这些信息来给交换网络选举根设备和根端口，以及给每个交换网段选举根端口和指定端口。

当一台设备上有两个端口同处某个环路中，设备就会通过生成树和路径开销设置来判断将其中的哪个端口置入转发状态，将其中的哪个端口置入阻塞状态。生成树端口优先级值可以代表这个端口在网络拓扑中的位置，以及其位置对于转发流量的优越程度。路径靠小指则代表了媒体的速率。

注释： 在默认情况下，设备只会在那些没有安装 SFP（小型可插拔）模块的接口上发送保活消息（来确保连接的连通性）。用户可以通过输入接口配置命令（不带其他关键字）[**no**] **keepalive** 来修改接口的这种默认操作。

生成树拓扑与 BPDU

下列因素共同构成了一个交换型网络稳定、活动的拓扑结构：

- 每台设备各个 VLAN 所关联的唯一的桥 ID（由设备优先级和 MAC 地址组成）。在设备堆栈中，所有设备在一个生成树实例中都会使用相同的桥 ID；
- 去往根设备的生成树路径开销；
- 每个二层接口所关联的端口标识符（由端口优先级和 MAC 地址组成）。

当网络中一台设备启动时，它们都会按照根设备的方式进行操作。每台设备都会通过所有的端口发送一条配置 BPDU。BPDU 的功能是发起通信并计算这个生成树的拓扑。每个配置 BDPU 中都会包含下列信息：

- 发送方设备认为是根设备的那台设备的唯一桥 ID；
- 去往根的生成树路径开销；
- 发送方设备的桥 ID；
- 消息老化值；
- 发送方接口的标识符；
- hello、转发延迟和最大老化协议计时器值。

当一台设备接收到一个包含更优信息（即桥 ID 更低，路径开销更低等）的配置 BDPU 时，它会将该端口的信息保存下来。如果这个 BPDU 是通过设备的根端口接收到的，那么设备也会将它经过更新，通过所有其作为指定设备的直连局域网段发送出去。

当一台设备接收到一个比当前为该端口保存的配置 BPDU 包含更差信息（即桥 ID 更低，路径开销更低等）的配置 BDPU 时，它就会丢弃这个 BPDU。如果这台设备是它接收到 BPDU 的那个局域网段的指定设备，那么它就会向这个局域网段中发送一条包含为该端口保存的最新信息的 BPDU。通过这种方式，较差的信息就会被丢弃，网络中传播都是较优的信息。

交换 BPDU 可以获得下面的效果：

- 网络中的一台设备会被选举为根设备（即一个交换型网络中生成树拓扑的逻辑中心）。详见下面的拓扑图；

  对于每个 VLAN 来说，拥有最高设备优先级（即优先级数值最低）的设备会被选举为根设备。如果所有设备上使用的都是默认优先级（32768），那么这个 VLAN 中 MAC 地址最低的设备就会成为根设备。设备优先级值在桥 ID 中占据了最重要的比特位，如下图的图所示。
- （除根设备之外）每台设备会选举出根端口。当设备向根设备转发数据包时，这个端口可以提供最佳的（也就是开销值最低的）路径。

  在设备堆栈中选择根端口时，生成树执行的操作为：
  - 选择根桥 ID 最低的端口；
  - 选择去往根设备路径开销最低的端口；
  - 选择指定桥 ID 最低的端口；

  - 选择指定路径开销最低的端口；
  - 选择端口 ID 最低的端口
- 在堆栈根设备中，只有一个出站端口会被选为根端口。堆栈中剩下的设备都会成为它的指定设备，如下图（中的设备 2 和设备 3）所示；
- 每台设备会根据路径开销来计算去往根交换机的最短距离；
- 每个局域网段会选出一台指定设备。在从这个局域网向根设备转发数据包时，指定设备是路径开销最低的设备。指定设备连接局域网的端口称为指定端口。

一个堆栈成员会被选举为堆栈的根设备。堆栈根设备中会包含出站的根端口（设备 1）。

图 42：设备堆栈中的生成树端口状态

Switch stack	交换机堆栈
Switch 1	交换机 1
Outgoing RP	出站 RP
StackWise Plus port connections	StackWise Plus 端口连接
Switch 2	交换机 2
Switch 3	交换机 3
Switch A	交换机 A
Switch B	交换机 B
RP=root port	RP=根端口
DP=designated port	DP=指定端口
BP=blocked port	BP=阻塞端口

在交换网络中，所有不需要从任何途径到达根设备的路径都会进入生成树阻塞模式。

桥 ID、设备优先级与扩展系统 ID

IEEE 802.1D 白哦准要求每台设备都有一个唯一的桥标识符（即桥 ID），这个标识符控制根桥的选择。由于在 PVST+和快速 PVST+环境中，每个 VLAN 都可以理解为一个不同的逻辑桥，因此同一台设备必须针对每个配置的 VLAN 都有一个不同的桥 ID。设备的每个 VLAN 都有一个唯一的 8 字节桥 ID。其中最重要的两个字节用于设备优先级，剩下的 6 个字节则取自于设备的 MAC 地址。

设备支持 IEEE 802.1t 生成树扩展，一些之前用于设备优先级的比特位在这个标准中用于 VLAN 标识符。这样做的结果是保留给设备的 MAC 地址更少了，而支持的 VLAN ID 范围更大的，这些都是为了保证桥 ID 的唯一性。

之前用于设备优先级的前 2 字节重新分配为了 4 比特的优先级值和等同于 VLAN ID 的 12 比特扩展系统 ID 值。

表 53：设备优先级值与扩展系统 ID

优先级位				**扩展系统 ID（设置为 VLAN ID）**								
16 位	**15 位**	**14 位**	**13 位**	**12 位**	**11 位**	**10 位**	**9 位**	**8 位**	**7 位**	**6 位**	**5 位**	**4 位**
32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8

3 位	**2 位**	**1 位**
4	2	2

生成树会使用扩展系统 ID、设备优先级和分配的生成树 MAC 地址来确保每个 VLAN 的桥 ID 是唯一的。由于设备堆栈在网络其他部分看来是一台设备，因此堆栈中所有设备对于一个给定生成树来说都会使用相同的桥 ID。如果堆栈主设备发生了故障，那么堆栈成员设备就会基

于新堆栈主设备的 MAC 地址来给所有运行的生成树重新计算它们各自的桥 ID。
支持扩展的系统 ID 会影响用户手动配置根设备、辅助根设备和 VLAN 的设备优先级的方式。例如，在用户修改设备优先级值时，用户可以修改设备会被选举为根设备的可能性。优先级值配置得越高，可能性就越低；反之则越高。
如果特定 VLAN 中的根设备优先级值低于 24576，那么设备就会将自己对指定 VLAN 的优先级设置为比最低设备优先级低 4096。4096 是表中最不重要的那 4 位设备优先级值。

端口优先级与路径开销

如果出现环路，生成树在选择将接口设置为转发状态时，会使用端口优先级。用户可以给自己希望首先选择的接口配置最高的优先级（也就是将优先级的数值配置为最低），给自己希望最后选择的接口配置最低的优先级（也就是将优先级的数值配置为最高）。如果所有接口的优先级值都相同，那么生成树就会让编号最低的接口进入转发状态，并且阻塞其他接口。
生成树路径开销的默认值取自于接口的媒体速率。如果出现环路，生成树就会在选择要将哪个接口置入转发状态时使用开销值。用户可以给那些希望首先选择的接口分配较低的开销值，并给那些希望最后选择的接口分配较高的开销值。如果所有接口拥有相同的开销值，那么生成树就会将接口编号最低的接口置入转发状态，并且阻塞其他接口。
如果设备是一个设备堆栈的成员，那么用户必须给希望首先选择的接口分配较低的开销值，并且给希望最后选择的接口分配较高的开销值，而不应该调整它们的端口优先级。要想了解具体信息，可以参考相关主题。

生成树接口状态

在协议信息穿过一个交换型局域网时，就会发生转发延迟。于是，在交换型网络中，网络拓扑就可以随时随地发生变更。当接口直接从没有参与生成树拓扑的状态过渡到转发状态时，它会就会形成一个临时的数据环路。接口必须等待新拓扑信息在交换网络中传播之后，才会开始转发数据帧。这些接口必须让在旧拓扑中转发的数据帧生存时间过期。
使用生成树的设备上，每个二层接口都会处于下列状态之一：

- 阻塞：接口不会参与数据帧转发；
- 侦听：在阻塞状态之后，当生成树决定让这个接口参与数据帧转发时，这个接口经历的第一个过渡状态；
- 学习：接口准备参与数据帧转发；
- 转发：接口转发数据帧；
- 禁用：因为这个端口被关闭、没有连接链路或者没有运行生成树实例，所以不会参与生成树。

接口会按照这种方式转换状态：

- 从初始化过渡到阻塞；
- 从阻塞过渡到侦听或禁用；
- 从侦听过渡到学习或禁用；
- 从学习过渡到转发或禁用；
- 从转发到禁用

接口会在状态下进行过渡。

图 43：生成树接口状态

Power-on initialization	加电 初始化
Blocking state	阻塞状态
Listening	侦听状态

state	
Disabled state	禁用状态
Learning state	学习状态
Forwarding state	转发状态

在设备启动时，生成树默认就会启用，设备、VLAN 或网络中的每个接口都会经历从阻塞状态进入到侦听与学习两个过渡状态的过程。每个接口都会最终稳定在转发或阻塞状态。

当生成树算法将一个二层接口置入转发状态时，就会发生下面的过程：

1 当生成树等待协议信息将接口置入阻塞状态时，接口会处于侦听状态；

2 当生成树等待转发延迟计时器过期时，生成树就会让接口进入学习状态，并且重置转发延迟计时器；

3 在学习状态下，当设备的转发数据库学习到终端站点的位置信息时，接口还是会继续阻塞数据帧的转发；

4 当转发延迟计时器超时时，生成树就会让接口进入转发状态对数据帧的学习和转发也会启用。

阻塞状态

阻塞状态的二层接口不会参与数据帧的转发。在初始化之后，设备会通过每个设备接口发送 BPDU。设备最初会像根桥那样工作，直到它与其他设备交换了 BPDU 为止。交换的过程会决定网络中的哪台设备是根设备。如果网络中只有一台设备，那就不会进行消息交换，转发延迟计时器会过期，而接口也会进入侦听状态。在设备初始化之后，接口一定会进入阻塞状态。

阻塞状态下的接口会执行下列功能：

- 丢弃这个接口接收到的数据帧；
- 丢弃从另一个接口交换过来进行转发的数据帧；
- 不学习地址；
- 接收 BPDU。

侦听状态

侦听状态是阻塞状态之后，二层接口进入的第一个状态。当生成树决定一个接口应该参与数据帧转发时，这个接口就会进入到这种状态下。

侦听状态下的接口会执行下列功能：

- 丢弃这个接口接收到的数据帧；
- 丢弃从另一个接口交换过来进行转发的数据帧；
- 不学习地址；
- 接收 BPDU。

学习状态

处于学习状态下的二层接口会准备参与数据帧转发。接口会从侦听状态进入学习状态。

学习状态下的接口会执行下列功能：

- 丢弃这个接口接收到的数据帧；
- 丢弃从另一个接口交换过来进行转发的数据帧；
- 学习地址；
- 接收 BPDU。

转发状态

处于转发状态下的二层接口会转发数据帧。接口会从学习状态进入转发状态。

转发状态下的接口会执行下列功能：

- 接收并转发这个接口接收到的数据帧；
- 转发从另一个接口交换过来的数据帧；
- 学习地址；
- 接收 BPDU。

禁用状态

处于禁用状态的二层接口并不会参与数据帧转发，也不会包含在生成树当中。处于禁用状态的接口是不执行操作的。

禁用状态下的接口会执行下列功能：

- 丢弃这个接口接收到的数据帧；
- 丢弃从另一个接口交换过来进行转发的数据帧；
- 不学习地址；
- 不接收 BPDU。

设备或端口是如何成为根设备或根端口的

如果网络中的所有设备都启用了默认的生成树设置，那么拥有最低 MAC 地址的设备就会成为根设备。

设备 A 会被选举为根设备，因为所有设备的设备优先级会被设置为默认值（32768），而设备 A 的 MAC 地址最低。不过，考虑到流量模式，转发接口的数量和链路的类型，设备 A 也许并不是理想的根设备。用户可以增加理想设备的优先级（即降低优先级的数值），让理想的设备成为根设备，用户可以强制生成树重新进行计算，来以理想设备为根形成新的拓扑。

图 44：生成树拓扑

RP=Root Port	RP=根端口
DP=Designated Port	DP=指定端口

在网络基于默认参数计算生成树拓扑时，交换网络中源与目的终端站点的路径可能并不是最理想的路径。例如，如果将速率较高的链路连接到了一个比根端口数值高的接口，有可能就会导致根端口的变更。这样做的目的在于让速率最高的链路成为根端口。

比如，若设备 B 上的一个端口为吉比特以太网链路，而设备 B 上的另一个端口（连接的是一条 10/100 链路）却是根端口。网络流量如果穿越吉比特以太网链路效率很可能更高。用户可以将吉比特以太网端口的修改生成树端口优先级修改为一个比根端口更高的优先级（也就是修改为一个更小的数），那么这个吉比特以太网就会成为新的根端口。

生成树与冗余连接

通过生成树协议，用户可以将两个设备接口连接到另一台设备，或者连接到两台不同的设备，以此创建冗余的骨干。生成树会自动禁用一个接口，但如果另一个接口出现故障时，生成树就会重新启用这个接口。如果其中一条链路是高速链路，另一条链路是低速链路，那么禁用的永远是那条低速链路。如果速率相同，那么生成树会将端口优先级和端口 ID 相加，然后禁用那个数值较高的端口。

图 45：生成树与冗余连接

Active link	活动链路
Blocked link	阻塞链路
Workstations	工作站

用户也可以使用 EtherChannel 组在设备之间创建冗余链路。

生成树地址管理

IEEE 802.1D 指定了 17 个组播地址来由不同的桥协议使用，这 17 个组播地址取值范围是从 0x00180C2000000 到 0x0180C2000010。这些地址是静态地址，无法移除。

无论生成树协议是什么状态，堆栈中的每台设备都会接收，但不会转发去往 0x00180C2000000 到 0x0180C2000010 之间的地址。

如果启用了生成树，那么堆栈中每台设备的 CPU 都会接收到去往 0x00180C2000000 到 0x0180C2000010 的数据包。如果禁用了生成树，那么设备或堆栈中的每台设备都会向那些未知组播地址转发数据包。

加速老化以保持连接

动态地址老化的默认时间为 5 分钟，这就是全局配置命令的默认设置为 **mac address-table aging-time**。不过，重新配置生成树可能会让许多站点的位置发生变化。由于这些站点有可能无法在 5 分钟或更长时间内到达，因此用户可以加速地址老化时间，让交换机将工作站地址从地址表中删除，然后再重新学习。在生成树重新计算时，加速老化与转发延迟参数值（全局配置命令 **spanning-tree vlan** *vlan-id* **forward-time** *seconds*）相同。

由于每个 VLAN 都是一个独立的生成树实例，设备会以每个 VLAN 为单位加速老化。在一个 VLAN 上执行生成树重新配置会让在这个 VLAN 中学习到的动态地址受到加速老化的影响。其他 VLAN 中的动态地址不会受到影响，这些地址仍然服从于在设备上输入的老化时间间隔。

生成树的模式与协议

设备支持下面这些生成树模式与协议：

- PVST+：这种模式是基于 IEEE 802.1D 和 Inspur 私有扩展标准的生成树标准。PVST+会在设备的每个 VLAN 上运行，直至达到最大的支持数量，这可以确保每个 VLAN 都在网络中获得一个无环路径。

  PVST+可以给它运行的 VLAN 提供二层的负载分担。用户可以使用网络中的 VLAN 来创建不同的逻辑拓扑，以确保所有链路都得到了有效地利用，但又没有链路会过载。在一个 VLAN 中，每个 PVST+实例都有一个单独的根设备。这台根设备会将与这个 VLAN 有关的生成树信息分发给网络中的所有设备。由于每台设备都拥有了关于网络的相同信息，因此这个过程可以确保网络维护能够得到维护；
- 快速 PVST+：快速 PVST+是设备上默认的 STP 模式。这种生成树模式与 PVST+相同，只不过这种模式使用了基于 IEEE 802.1w 标准的快速收敛。为了提供快速收敛，快速 PVST+会在接收到拓扑变更消息时，以端口为单位立即删除动态学习到的 MAC 地址条目。而 PVST+则对动态学习的 MAC 地址条目使用了一个比较短的老化时间。

  （除非专门指出）快速 PVST+采用了与 PVST+相同的配置方法，用户只需要在设备上进行最简单的配置。快速 PVST+的好处在于用户可以将大量的 PVST+安装库迁移到快速 PVST+当中，而不需要掌握多生成树协议（MSTP）复杂的配置方法，也不需要重新部署网络。在快速 PVST+模式中，每个 VLAN 都会运行自己的生成树实例，直至达到最大支持的生成树数量为止；
- MSTP：这种模式是基于 IEEE 802.1s 标准的生成树标准。用户可以将多个 VLAN 映射到同一个生成树实例当中，这可以减少生成树实例的数量，让设备不必为大量 VLAN 支持生成树。MSTP 采用的是 RSTP（基于 IEEE 802.1w）的做法，即通过减少转发延迟及快速将根端口与指定端口快速过渡到转发状态下，来提升生成树的收敛速率。在设备堆栈中，交叉堆栈快速转换（Cross stack rapid transition，CSRT）特性会执行与 RSTP 相同的功能。没有 RSTP 或 CSRT 就无法运行 MSTP。

支持的生成树实例

在 PVST+或快速 PVST+模式下，设备或设备堆栈支持最多 128 个生成树实例。

在 MSTP 模式下，设备或设备堆栈支持 65 个 MST 实例。每个 MST 中可以映射的 VLAN 数量是没有限制的。

生成树的互操作性与向后兼容

在混合使用 MSTP 和 PVST+的网络中，公共生成树（CST）的根必须处于 MST 骨干当中，而 PVST+设备无法连接到多个 MST 域。

当一个网络同时包含了运行快速 PVST+的设备和运行 PVST+的设备时，我们推荐将快速 PVST+设备与 PVST+设备配置为不同的生成树实例。在快速 PVST+生成树实例中，根设备必须为快速 PVST+设备。在 PVST+实例中，根设备必须为 PVST+设备。PVST+设备应该部署在网络的边缘。

所有堆栈成员都会运行相同版本的生成树（皆为 PVST+、皆为快速 PVST+或皆为 MSTP）。

表 54：PVST+、MSTP 和快速 PVST+的互操作性与兼容性

	PVST+	**MSTP**	**快速 PVST+**
PVST+	是	是（但有限制）	是（回退为 PVST+）
MSTP	是（但有限制）	是	是（回退为 PVST+）
快速 PVST+	是（回退为 PVST+）	是（回退为 PVST+）	是

STP 与 IEEE 802.1Q Trunk

对于 VLAN trunk 的 IEEE 802.1Q 标准对于网络的生成树战略施加了一些限制条件。这种标准要求给 trunk 支持的*所有* VLAN 只能运行一个生成树实例。但在一个由 Inspur 设备通过 IEEE 802.1Q trunk 相互连接而组成的网络中，这些设备还是会给 trunk 支持的每个 VLAN 各自维护一个生成树实例。

在将一台 Inspur 设备通过 IEEE 802.1Q trunk 连接到一台非 Inspur 设备时，Inspur 设备会使用 PVST+来提供生成树的互操作性。如果启用了快速 PVST+，那么设备就会使用快速 PVST+，而不使用 PVST+。设备会将 IEEE 802.1Q VLAN 的生成树实例，与非 Inspur IEEE 802.1Q 设备的生成树实例结合起来。

不过，被一个由非 Inspur IEEE 802.1Q 设备所组成的网络云相互隔开的 Inspur 设备会维护所有 PVST+或快速 PVST+信息，而隔开 Inspur 设备的非 Inspur IEEE 802.1Q 云会被视为设备之间的一条 trunk 链路。

IEEE 802.1Q trunk 上会自动启用快速 PVST+，这里不需要用户进行任何配置。Access 端口及 ISL（交换机间链路） trunk 端口上的外部生成树的操作不会受到 PVST+的影响。

VLAN 桥生成树

Inspur VLAN 桥生成树是与后退桥接特性（桥组）一起使用的，后者会在两个或多个 VLAN 桥域或路由端口之间转发非 IP 协议（如 DECnet）的流量。VLAN 桥生成树可以让桥组在每个 VLAN 生成树之上，再建立一个生成树，防止 VLAN 之间有多条连接进而形成环路。这项技术也可以防止各个被桥接的 VLAN 的生成树不会坍塌为一棵生成树。

要支持 VLAN 桥生成树，需要增加一些生成树计时器。要使用后退桥接特性，用户必须在设备上启用 IP Services 特性集。

生成树与设备堆栈

当设备堆栈工作在 PVST+或快速 PVST+模式下时：

- 设备堆栈在网络其他部分看来是一台设备，因此堆栈中所有设备对于一个给定生成树来说都会使用相同的桥 ID。其桥 ID 取自于主用交换机的 MAC 地址；
- 当一台新的设备加入堆栈时，它会将自己的桥 ID 设置为主用交换机的桥 ID。如果这台新添加的设备 ID 值最低，且所有堆栈成员的根路径开销相同，那么新添加的这台设备就会成为堆栈的根；
- 当一台堆栈成员离开堆栈时，生成树会在堆栈中（有可能也会在堆栈外）重新收敛。其余的堆栈成员设备中，拥有最低堆栈端口 ID 的设备就会成为堆栈的根；
- 如果堆栈外部的相邻设备出现故障或者掉电，网络就会执行常规的生成树处理。网络有可能会因为活动拓扑中丢失了一台设备而重新收敛；

- 如果堆栈外部添加了一台新的设备，网络就会执行常规的生成树处理。网络有可能会因为活动拓扑中增加了一台设备而重新收敛。

默认的生成树配置

表 55：默认的生成树配置

特性	默认设置
启用状态	在 VLAN 1 启用
生成树模式	快速 PVST+（PVST+与 MSTP）
设备优先级	32768
生成树端口优先级（可以基于接口进行配置）	128
生成树端口开销（可以基于接口进行配置）	1000Mb/s：4 100Mb/s：19 10Mb/s：100
生成树 VLAN 端口优先级（可以基于 VLAN 进行配置）	128
生成树 VLAN 端口开销（基于 VLAN 进行配置）	1000Mb/s：4 100Mb/s：19 10Mb/s：100
生成树计时器	Hello 时间：2 秒 转发延迟时间：15 秒 最大老化时间：20 秒 传输抑制树：6 BPDU

注释： 从 Inspur INOS 15.2(4)E 版开始，默认 STP 模式为快速 PVST+。

如何配置生成树特性

修改生成树模式（CLI）

交换机支持三种生成树模式：每 VLAN 生成树加（PVST+）、快速 PVST+或多生成熟协议（MSTP）。在默认情况下，设备会运行快速 PVST+协议。

如果用户希望启用一种与默认模式不同的模式，需要执行下面的流程。

总步骤

1. enable

2. configure terminal

3. spanning-tree mode {pvst | mst | rapid-pvst}

4. interface *interface-id*

5. spanning-tree link-type point-to-point

6. end

7. clear spanning-tree detected-protocols

具体步骤

	命令或操作	目的

步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**spanning-tree mode** {**pvst** | **mst** | **rapid-pvst**} 示例： `Device(config)# spanning-tree mode pvst`	配置生成树模式。所有堆栈成员运行生成树的同一个版本。 • 选择 **pvst** 可以启用 PVST+； • 选择 **mst** 可以启用 MSTP； • 选择 **rapid-pvst** 可以启用快速 PVST+
步骤 4	**interface** *interface-id* 示例： `Device(config)# interface gigabitethernet 1/0/1`	指定要配置的接口，进入接口配置模式。有效接口包括物理端口、VLAN 和 port channels。VLAN ID 的取值范围是从 1 到 4094，port-channel 的取值范围是从 1 到 48
步骤 5	**spanning-tree link-type point-to-point** 示例： `Device(config-if)# spanning-tree link-type point-to-point`	将这个端口的链路类型设置为点到点。 如果将这个端口（本地端口）通过一条点到点链路连接到一个远端端口，而这个本地端口成为了指定端口的话，那么设备就会与远端端口进行协商，并且快速将本地端口修改为转发状态
步骤 6	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式
步骤 7	**show ip igmp interface** [*interface-id*] 示例： `Device# show ip igmp interface`	设备上的任何端口与一台传统的 IEEE 802.1D 相连，这条命令都会在整台设备上重新启动协议迁移进程。 如果指定设备检测到这台设备运行的是快速 PVST+，那么这一步就是可选的操作

禁用生成树（CLI）

生成树默认会在 VLAN 1 和所有新创建的 VLAN 上启用，直至达到了生成树的限制数量为止。只有在用户十分确定网络中没有环路时，才可以禁用生成树。

注意： 如果用户禁用了生成树而网络中又仍然有环路，那么网络中的过量流量和永无休止的数据包复制操作会严重影响网络的性能。

这项操作是可选的。

总步骤

1. enable

2. configure terminal

3. no spanning-tree vlan *vlan-id*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**no spanning-tree vlan** *vlan-id* 示例： `Device(config)#` **no spanning-tree vlan 300**	*vlan-id* 部分的取值范围是从 1 到 4094
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置根设备（CLI）

用户要在指定 VLAN 中将一台设备配置为根，可以使用全局配置命令 **spanning-tree vlan** *vlan-id* **root** 来将设备优先级修改为一个远远低于默认值（32768）的数值。在输入这条命令时，软件会校验每个 VLAN 中根设备的设备优先级。由于支持扩展系统 ID，因此如果 24576 这个值可以让设备成为指定 VLAN 的根，那么设备会将自己在指定 VLAN 中的优先级设置为 24576。用户可以使用关键字 **diameter** 来设置二层网络的直径（即两台终端工作在在二层网络中相隔的设备最大跳数）。在用户设置网络半径的时候，设备会自动设置优化的 hello 时间、转发延迟时间和这个网络半径下的最大老化时间，这些参数可以显著减少收敛时间。用户可以使用关键字 **hello** 来覆盖自动计算出来的 hello 时间。

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. spanning-tree vlan *vlan-id* **root primary** [**diameter** *net-diameter*

4. end

具体步骤

	命令或操作	目的

步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**spanning-tree vlan** *vlan-id* **root primary** [**diameter** *net-diameter* 示例： `Device(config)# spanning-tree vlan 20-24 root primary diameter 4`	将一台设备配置为指定 VLAN 的根。 在 *vlan-id* 部分，用户可以通过 VLAN ID 值来输入一个 VLAN，可以用连字符输入一个 VLAN 范围，也可以用逗号相隔输入一系列的 VLAN。VLAN 的取值范围是从 1 到 4094。 • （可选）在 **diameter** *net-diameter* 部分，设置两台终端工作站之间相隔的最大设备数量。取值范围是 2 到 7
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

接下来做什么？

在将设备配置为根设备之后，我们推荐用户不要手动通过全局配置命令 **spanning-tree vlan** *vlan-id* **hello-time**、**spanning-tree vlan** *vlan-id* **forward-time** 和 **spanning-tree vlan** *vlan-id* **max-age** 来配置 hello 时间、转发延迟时间和最大老化时间。

配置辅助根设备（CLI）

在将一台设备配置为辅助根时，设备优先级会从默认值（32768）修改为 28672。如果这个 VLAN 的主用根发生了故障，那么设备使用这个优先级就更有可能会成为这个 VLAN 的根设备。这一点的前提是其他网络设备使用的都是默认的设备优先级 32768，因此这些设备就很难成为根设备。

用户可以在多台设备上执行这条命令，来将多台设备配置为备份根设备。用户也可以使用配置主用根设备时使用的命令 **spanning-tree vlan** *vlan-id* **root primary** 来设置辅助根设备的网络直径与 hello 时间值。

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. spanning-tree vlan *vlan-id* **root secondary** [**diameter** *net-diameter*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable**	进入特权 EXEC 模式。在提示时输入密码

	示例： `Device>` **enable**	
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree vlan** *vlan-id* **root secondary** [**diameter** *net-diameter* 示例： `Device(config)#` **spanning-tree vlan 20-24 root secondary diameter 4**	将一台设备配置为指定 VLAN 的辅助根。 • 在 *vlan-id* 部分，用户可以通过 VLAN ID 值来输入一个 VLAN，可以用连字符输入一个 VLAN 范围，也可以用逗号相隔输入一系列的 VLAN。VLAN 的取值范围是从 1 到 4094. • （可选）在 **diameter** *net-diameter* 部分，设置两台终端工作站之间相隔的最大设备数量。取值范围是 2 到 7。 应该在这里给设备配置与主用根设备相同的网络直径
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置端口优先级（CLI）

注释： 如果用户的设备是设备堆栈的成员，那就必须使用接口配置命令 **spanning-tree** [**vlan** *vlan-id*] **cost** *cost*（而不是接口配置命令 **spanning-tree** [**vlan** *vlan-id*] **port-priority** *priority*）来选择将一个接口置入转发状态。用户可以给自己希望首先选择的接口配置一个较低的开销值，而给自己希望之后选择的接口配置一个较高的开销值。

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. spanning-tree port-priority *priority*

5. spanning-tree vlan *vlan-id* **port-priority** *priority*

6. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal**	进入全局配置模式

	示例： `Device# configure terminal`	
步骤 3	**interface** *interface-id* 示例： `Device(config)# interface gigabitethernet1/0/2`	指定要配置的接口，进入接口配置模式。 有效接口包括物理端口 port channel 逻辑接口（**port-channel** *port-channel-number*）
步骤 4	**spanning-tree port-priority** *priority* 示例： `Device(config-if)# spanning-tree port-priority 0`	给一个接口配置端口优先级。 在 *priority* 部分，取值范围是从 0 到 240，增量为 16，默认值是 128。有效的取值包括 0、16、32、48、64、80、96、112、128、144、160、176、192、208、224 和 240。配置其他值系统都会拒绝。数值越低，优先级就越高
步骤 5	**spanning-tree vlan** *vlan-id* **port-priority** *priority* 示例： `Device(config-if)# spanning-tree vlan 20-25 port-priority 0`	给一个 VLAN 配置端口优先级。 • 在 *vlan-id* 部分，用户可以通过 VLAN ID 值来输入一个 VLAN，可以用连字符输入一个 VLAN 范围，也可以用逗号相隔输入一系列的 VLAN。VLAN 的取值范围是从 1 到 4094. • 在 *priority* 部分，取值范围是从 0 到 240，增量为 16，默认值是 128。有效的取值包括 0、16、32、48、64、80、96、112、128、144、160、176、192、208、224 和 240。配置其他值系统都会拒绝。数值越低，优先级就越高
步骤 6	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置路径开销（CLI）

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. spanning-tree cost *cost*

5. spanning-tree vlan *vlan-id* **cost** *cost*

6. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable**	进入特权 EXEC 模式。在提示时输入密码

	示例： `Device>` **enable**	
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的接口，进入接口配置模式。 有效接口包括物理端口 port channel 逻辑接口（**port-channel** *port-channel-number*）
步骤 4	**spanning-tree cost** *cost* 示例： `Device(config-if)#` **spanning-tree cost 250**	给一个接口配置开销。 • 如果出现环路，生成树在选择要将哪个接口置入转发状态时就会使用路径开销进行判断。路径开销越低表示传输速率越高。 • 在 *cost* 部分，取值范围是从 1 到 200000000，这个值取自于接口媒体的速率
步骤 5	**spanning-tree vlan** *vlan-id* **cost** *cost* 示例： `Device(config-if)#` **spanning-tree vlan 10,12-15,20 cost 300**	给一个 VLAN 配置开销。 如果出现环路，生成树在选择要将哪个接口置入转发状态时就会使用路径开销进行判断。路径开销越低表示传输速率越高。 • 在 *vlan-id* 部分，用户可以通过 VLAN ID 值来输入一个 VLAN，可以用连字符输入一个 VLAN 范围，也可以用逗号相隔输入一系列的 VLAN。VLAN 的取值范围是从 1 到 4094. • 在 *cost* 部分，取值范围是从 1 到 200000000，这个值取自于接口媒体的速率
步骤 6	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

特权 EXEC 命令 **show spanning-tree interface** *interface-id* 只会显示那些链路处于 up 状态的端口信息。否则，用户也可以使用特权 EXEC 命令 **show running-config** 来确认自己所作的配置。

配置一个 VLAN 的设备优先级（CLI）

用户可以配置设备的优先级，让一台独立设备或一台堆栈中的设备更有可能被选为根设备。

注释： 在使用这条命令时务请小心。在大多数情况下，我们推荐用户使用全局配置命令 **spanning-tree vlan** *vlan-id* **root primary**和**spanning-tree vlan** *vlan-id* **root secondary**来修改设备优先级。

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. spanning-tree vlan *vlan-id* **priority** *priority*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree vlan** *vlan-id* **priority** *priority* 示例： `Device(config)#` **spanning-tree vlan 20 priority 8192**	配置一个 VLAN 的设备优先级。 • 在 *vlan-id* 部分，用户可以通过 VLAN ID 值来输入一个 VLAN，可以用连字符输入一个 VLAN 范围，也可以用逗号相隔输入一系列的 VLAN。VLAN 的取值范围是从 1 到 4094. • 在 *priority* 部分，取值范围是从 0 到 61440，增量为 4096，默认值是 32768。数值越低，设备越有可能被选为根设备。 有效的值包括 4096、8192、12288、16384、20480、24576、28672、32768、36864、40960、45056、49152、53248、57344 和 61440。配置其他值系统都会拒绝。
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置 Hello 时间（CLI）

hello 时间是根设备生成和发送配置消息的时间间隔。

这个流程是可选的。

总步骤

1. enable

2. spanning-tree vlan *vlan-id* **hello-time** *seconds*

3. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例：	进入特权 EXEC 模式。在提示时输入密码

	Device> **enable**	
步骤 2	**spanning-tree vlan** *vlan-id* **hello-time** *seconds* 示例： Device(config)# **spanning-tree vlan 20-24 hello-time 3**	配置一个 VLAN 的 hello 时间。hello 时间是根设备生成和发送配置消息的时间间隔。 • 在 *vlan-id* 部分，用户可以通过 VLAN ID 值来输入一个 VLAN，可以用连字符输入一个 VLAN 范围，也可以用逗号相隔输入一系列的 VLAN。VLAN 的取值范围是从 1 到 4094. • 在 *seconds* 部分，取值范围是从 1 到 10；默认值为 2
步骤 3	**end** 示例： Device(config-if)# **end**	返回特权 EXEC 模式

给一个 VLAN 配置转发延迟时间（CLI）

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. spanning-tree vlan *vlan-id* **forward-time** *seconds*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree vlan** *vlan-id* **forward-time** *seconds* 示例： Device(config)# **spanning-tree vlan 20,25 forward-time 18**	配置一个 VLAN 的转发时间。转发延迟是接口在将自己的生成树学习和侦听状态过渡到转发状态之前，等待的秒数。 • 在 *vlan-id* 部分，用户可以通过 VLAN ID 值来输入一个 VLAN，可以用连字符输入一个 VLAN 范围，也可以用逗号相隔输入一系列的 VLAN。VLAN 的取值范围是从 1 到 4094. • 在 *seconds* 部分，取值范围是从 4 到 30，默认值是 15
步骤 4	**end**	返回特权 EXEC 模式

	示例： `Device(config)# end`	

给一个 VLAN 配置最大老化时间

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. spanning-tree vlan *vlan-id* **max-age** *seconds*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree vlan** *vlan-id* **max-age** *seconds* 示例： `Device(config)#` **spanning-tree vlan 20 max-age 30**	配置一个 VLAN 的最大老化时间。最大老化时间是指设备从没有接收到生成树配置消息开始，会等待多久才会开始执行重新配置。 • 在 *vlan-id* 部分，用户可以通过 VLAN ID 值来输入一个 VLAN，可以用连字符输入一个 VLAN 范围，也可以用逗号相隔输入一系列的 VLAN。VLAN 的取值范围是从 1 到 4094； • 在 *seconds* 部分，取值范围是从 6 到 40，默认值是 20
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置传输抑制计时（CLI）

用户可以通过修改传输抑制计时值来配置 BPDU 突发值。

注释： 将这个参数修改为一个较高的数值会严重影响 CPU 的使用率，特别是在快速 PVST+ 模式下。降低这个参数值则会在某些情况下延迟收敛时间。我们推荐用户维持默认的设置。

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. spanning-tree transmit hold-count *value*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree transmit hold-count** *value* 示例： `Device(config)#` **spanning-tree transmit hold-count 6**	配置在暂停 1 秒之前可以发送的 BPDU 数量。 • 在 *value* 部分，取值范围是从 1 到 20，默认值为 6
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

监控生成树的状态

表 56：查看生成树状态的命令

show spanning-tree active	仅显示活动接口的生成树信息
show spanning-tree detail	显示具体的接口信息
show spanning-tree vlan *vlan-id*	显示特定 VLAN 的生成树信息
show spanning-tree interface *interface-id*	显示特定接口的生成树信息
show spanning-tree interface *interface-id* **portfast**	显示特定接口的生成树 portfast 信息
show spanning-tree summary [**totals**]	显示接口状态的汇总信息，或者显示 STP 状态部分的总行

其他关于生成树协议的参考资料

相关文档

相关主题	文档名
生成树协议的命令	《LAN 交换命令参考手册，Inspur INOSXE3SE 版（Inspur 6650 交换机）》

错误消息解码器

描述	链接
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

标准与 RFC

标准/RFC	标题
无	——

技术助手

描述	链接
Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。 用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	http://www.icntnetworks.com

STP 的特性信息

版本	修改
Inspur INOS 12.2	引入该特性

配置多生成树协议

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

MSTP 的前提条件

- 对于同一个多生成树（MSTP）域中的两台或多台设备，它们必须拥有相同的 VLAN 与实例映射关系、相同的修订版本号和相同的名称；
- 对于同一个 MSTP 域中的两个或多个堆栈交换机来说，它们必须拥有相同的 VLAN 与实例映射关系、相同的修订版本号和相同的名称；
- 要想网络能够跨越冗余路径执行负载分担，那么所有 VLAN 与实例的映射关系就必须相互匹配；否则，所有流量就都会通过同一条链路。用户可以通过手动配置路径开销，来跨越堆栈中设备实现负载分担；
- 要想在一个每 VLAN 生成树加（PVST+）和一个 MST 云之间实现负载分担，或者在一个快速 PVST+和一个 MST 云之间实现负载分担，所有 MST 边界端口都必须进行转发。MST 边界端口进行转发，同时 MST 云的内部生成树（IST）主设备是公共生成树（CST）的根。如果 MST 云由多个 MST 域组成，那么其中一个 MST 域必须包含 CST 根，而所有其他 MST 区域必须有一条比通过 PVST+云或快速 PVST+云更优的路径，可以去往 MST 云的根。用户可能需要手动配置云中的设备。

MSTP 的限制条件

- 用户不能混合使用 Inspur 3850 和 Inspur 6650 交换来建立堆栈；
- 设备堆栈支持最多 65 个 MST 实例，但可以映射到一个 MST 实例中的 VLAN 数量是不受限制的；
- 支持 PVST+、快速 PVST+和 MSTP，但一次只能使用一个版本（例如，所有 VLAN 均运行 PVST+、所有 VLAN 均运行快速 PVST+或所有 VLAN 均运行 MSTP）；
- 不支持通过 VLAN 中继协议（VTP）传播 MST 的配置。但用户可以使用命令行界面（CLI）或者通过 SNMP（简单网络管理协议），在 MST 域的各个设备上手动配置 MST 的配置（域名称、修订版本号和 VLAN 与实例的映射）；
- 不推荐将网络分为一个大的域。但如果这种情况无法避免，我们推荐用户将交换型以太网分入通过路由器或非二层设备连接的小型局域网当中；
- 一个域中可以拥有一个或多个 MST 配置相同的成员；每个成员都必须能够处理快速生成树协议（RSTP）桥协议数据单元（BPDU）。一个网络中的 MST 域没有数量限制，但每个域只能支持最多 65 个生成树实例。用户每次只能将一个 VLAN 分配给一个生成树实例；
- 在将一台设备配置为根设备之后，我们推荐用户不要通过全局配置命令 **spanning-tree mst hello-time**、**spanning-tree mst forward-time** 和 **spanning-tree mst max-age** 手动配置 hello 时间、转发延迟时间、最大老化时间。

表 57：PVST+、MSTP 和快速 PVST+的互操作性域兼容性

	PVST+	**MSTP**	**快速 PVST+**
PVST+	是	是（但有限制）	是（回退为 PVST+）

MSTP	是（但有限制）	是	是（回退为 PVST+）
快速 PVST+	是（回退为 PVST+）	是（回退为 PVST+）	是

关于 MSTP 的信息

MSTP 的配置

MSTP 使用 RSTP 来实现快速收敛，这种技术可以将多个 VLAN 进行分组，并且映射到同一个生成树实例当中，减少支持大量 VLAN 所需的生成树实例数量。MSTP 可以给数据流量提供多条转发路径，以此来实现负载分担，减少支持大量 VLAN 所需的生成树实例数量。这项技术可以提升网络的容错性，因此一个实例（转发路径）出现了故障并不会影响其他实例（转发路径）正常工作。

注释： 多生成树（MST）是基于 IEEE 802.1s 标准实施的。

MSTP 最常用的初始部署方式是部署在二层交换网络的骨干和分布层。这种部署方式可以提供能够满足服务运营商网络需求的高可用性网络。

当设备工作在 MST 模式下时，（基于 IEEE 802.1w 的）RSTP 会自动启用。RSTP 通过显式握手的方式消除了 IEEE 802.1D 标准的转发延迟，让根端口和指定端口可以迅速过渡到转发状态，因此实现生成树的快速收敛。

MSTP 和 RSTP 都提升了生成树的操作水平，同时可以对基于（原始的）IEEE 802.1D 的生成树，以及 Inspur 私有的多实例生成树（MISTP）和 Inspur PVST+与快速每 VLAN 生成树加（快速 PVST+）实现向后兼容。

对于网络的其余部分而言，设备堆栈就是一个生成树节点，所有堆栈成员都会使用相同的设备 ID。

MSTP 配置指南

在使用全局配置命令 **spanning-tree mode mst** 启用 MST 时，RSTP 也会自动启用；

关于 UplinkFast、BackboneFast 和交叉堆栈 UplinkFast 的配置指南，请参见相关主题中提到的相关内容；

当设备工作在 MST 模式下时，它会使用长路径开销计算法（32 位）来计算路径开销值。通过长路径开销计算法，设备支持下面的路径开销值：

速率	**路径开销值**
10Mb/s	2000000
100Mb/s	200000
1Gb/s	20000
10Gb/s	2000
100Gb/s	200

根交换机

设备会给映射的 VLAN 组维护一个生成树实例。每个实例都会关联一个由设备优先级和设备

MAC 地址组成的设备 ID。对于一组 VLAN 来说，设备 ID 最低的设备会成为根设备。
在将一台设备配置为根设备时，用户需要将这台设备的优先级从默认值（32768）修改为一个明显更低的数值，这样设备才能成为指定生成树实例的根设备。在输入这条命令的时候，设备会校验根设备的设备优先级。由于支持扩展系统 ID，因此如果 24576 这个值可以让设备成为指定 VLAN 的根，那么设备会将自己在指定 VLAN 中的优先级设置为 24576。
如果指定实例的根设备优先级低于 24576，那么设备就会讲自己的优先级设置得比最低设备优先级低 4096。（4096 是设备优先级值中最低 4 位的值，要想了解详细信息，可以在相关主题中选择“桥 ID、设备优先级域扩展系统 ID”的链接）
如果网络中由一些支持扩展系统 ID 的设备，和一些不支持扩展系统 ID 的设备组成，那么包含扩展系统 ID 的设备就不太可能成为根设备。每当 VLAN 编号大于运行老板系统的设备的优先级值，扩展系统 ID 会增加设备的优先级值。
对每个生成树实例来说，根设备都应该是骨干设备或分布层设备。不要将接入层设备配置为生成树的主用根。
用户可以使用关键字 **diameter**（仅可用于 MST 实例 0）来设置二层网络的半径（也就是二层网络中任意两台终端设备之间的最大设备跳数）。在用户设置网络半径的时候，设备会自动设置优化的 hello 时间、转发延迟时间和这个网络半径下的最大老化时间，这些参数可以显著减少收敛时间。用户可以使用关键字 **hello** 来覆盖自动计算出来的 hello 时间。

多生成树域

要想让交换机参与多生成树（MST）实例，用户必须给所有交换机上配置相同的 MST 配置信息。多台拥有相同 MST 配置的交换机共同组成了一个 MST 域。
MST 的配置会控制各个设备属于哪个 MST 域。配置的内容包括域名称，修订版本号和 VLAN 与实例的分配映射。用户需要在设备上设置 MST 域的配置。用户可以将多个 VLAN 映射到一个 MST 实例，并且设置域名称、设置修订版本号。要想了解详细信息和示例，可以选择相关主题中的“设置 MST 域的配置与启用 MSTP”链接。
一个域中可以有多台拥有相同 MST 配置的成员。每个成员都必须能够处理 RSTP 桥协议数据单元（BPDU）。一个网络中的 MST 域没有数量限制，但每个域只能支持最多 65 个生成树实例。实例可以使用从 0 到 4094 之间的数字进行标识。用户每次只能将一个 VLAN 分配给一个生成树实例；

IST、CIST 和 CST

在 PVST+和快速 PVST+中，每个生成树实例都是独立的。MSTP 则与此不同，它会建立和维护两类生成树：

- 一棵内部生成树（IST），即运行在 MST 域中的生成树。
  在每个 MST 域中，MSTP 都会维护多个生成树实例。实例 0 是每个域中的一个特殊实例，成为内部生成树（IST）。所有其他 MST 实例的编号则从 1 到 4094。
  IST 是唯一会发送和接收 BPDU 的生成树实例。所有其他生成树实例信息都包含在 M 记录中，而 M 记录是封装在 MSTP BPDU 中的。由于 MSTP BPDU 会携带关于所有实例的信息，因此，为了支持多生成树实例而需要处理的 BPDU 数量可以显著减少。
  同一个域中的所有 MST 实例都会共享同一个协议计时器，但每个 MST 实例都会包含自己的拓扑参数，如根设备 ID、根路径开销等等。在默认情况下，所有 VLAN 都会分配给

IST。

MST 实例只具有区域本地意义。例如，域 A 中的 MST 实例 1 是与域 B 中的 MST 实例 1 相独立的，哪怕域 A 与域 B 相互连接也是如此。

- 一棵公共和内部生成树（CIST）。CIST 是每个 MST 域中的一系列 IST，和连接 MST 域与单个生成树的公共生成树（CST）。

  一个域中计算出来的生成树是包含整个交换域的 CST 的子树。CIST 是支持 IEEE 802.1w、IEEE 802.1s 和 IEEE 802.1D 标准的交换机共同运行的生成树算法所形成的树。一个 MST 域中的 CIST 与一个域外的 CST 相同。

MST 域内的操作

IST 连接了一个域内的所有 MSTP 交换机。当 IST 收敛时，IST 的根就会成为 CIST 的域跟（在 IEEE 802.1s 标准实现之前，称为 IST master）。这是域内拥有最低设备 ID 和去往 CIST 根最短路径开销的设备。如果网络中只有一个域的话，那么 CIST 域根也就是 CIST 的根。如果 CIST 根在域外，那么在域边界的一台 MSTP 交换机就会被选为这个 CIST 的域根。

当 MSTP 设备启动时，它会通过发送 BPDU 来声称自己是 CIST 的根和 CIST 的域根，同时将去往 CIST 根和 CIST 域根的路径开销设置为 0。设备也会启动自己所有的 MST 实例，并且声称自己是所有这些实例的根。如果设备接收到了比当前给这个端口存在的根信息更优的 MST 根信息（比如更低的设备 ID、更低的路径开销等等），它就会放弃自己作为 CIST 域根的身份。

在启动过程中，一个域中可能还有很多子域，每个子域都有自己的 CIST 域根。当交换机接收到较优的 IST 信息时，它们会离开自己过去的子域，并且加入包含了真正 CIST 域根的新子域。所有子域都会收缩，除了包含 CIST 域根的那个子域之外。

为了能够实现正常的操作，MST 域中的所有交换机必须都拥有相同的 CIST 域根。因此，域中任何两台交换机都只会在它们收敛到一个公共 CIST 域根时，针对一个实例同步它们的端口角色。

MST 域间的操作

如果网络中有多个域或传统 IEEE 802.1D 设备，那么 MSTP 就会建立并维护 CST，其中包括所有 MST 区域和网络中的所有传统 STP 设备。MST 实例会结合域边界的 IST 向结合，成为 CST。IST 连接了一个域内的所有 MSTP 交换机，并且作为包含整个交换域的 CIST 中的一个子树。这棵子树的根就是 CIST 域跟。MST 域会成为域 STP 设备和 MST 域相邻的一台虚拟设备。

一旦 CST 实例接受并发送 BPDU，MST 实例就会将它们的生成树信息添加到 BPDU 当中，来与相邻的设备进行通信，并且计算最终的生成树拓扑。正因如此，与 BPDU 传输有关的生成树参数（例如 hello 时间、转发时间、最大老化时间和最大跳数）只在 CST 实例上进行了配置，但却会影响所有 MST 实例。与生成树拓扑有关的参数（如设备优先级、端口 VLAN 开销和端口 VLAN 优先级）可以同时在 CST 实例和 MST 实例上进行配置。

MSTP 设备会使用第 3 版 RSTP BPDU 或 IEEE 802.1D STP BPDU 来域传统 IEEE 802.1D 设备进行通信。MST 设备会使用 MSTP BPDU 来与 MSTP 设备进行通信。

IEEE 802.1s 术语

在 Inspur 预标准实施方案中的一些 MST 命名方式已经进行了调整，以便定义一些*内部*或*区域*参数。这些内部参数只在一个 MST 域中有意义，而外部参数则与整个网络相关。由于 CIST 是唯一一个扩展到整个网络中的生成树实例，因此只有 CIST 参数需要用到外部（而不是内部或区域）术语。

- CIST 根是 CIST（唯一扩展到整个网络中的实例）的根设备；
- CIST 外部根路径的开销是去往 CIST 根的开销。在一个 MST 域中，开销是没有变化的。切记，对于 CIST 来说，一个 MST 域就像一台设备一样。CIST 外部根路径开销是在这些虚拟设备和不属于任何域的设备之间计算出来的根路径开销；

- CIST 域根在预标准实施方案中称为 IST master。如果 CIST 根在域中，那么 CIST 域根就是 CIST 的根。否则，CIST 域根就是距离域的 CIST 根最近的设备。CIST 域根会充当 IST 的根设备；
- CIST 内部根路径开销是去往一个域的 CIST 域根的开销。这个开销值只与 IST（实例 0）有关。

表 58：预标准与标准术语

IEEE 标准	**Inspur 预标准**	**Inspur 标准**
CIST 的域根	IST master	CIST 的域根
CIST 内部根的路径开销	IST master 的路径开销	CIST 的内部路径开销
CIST 外部根的路径开销	根的路径开销	根的路径开销
MSTI 的域根	实例的根	实例的根
MSTI 内部根的路径开销	根的路径开销	根的路径开销

MST 域的图例

这张图显示了 3 个 MST 域，和一台传统的 IEEE 802.1D 设备（D）。域 1（A）的 CIST 域跟也是 CIST 根。域 2（B）的 CIST 域根和域 3（C）的 CIST 域根分别是它们在 CIST 中对应子树的根。RSTP 会在所有域中运行。

图 46：MST 域、CIST Master 和 CST 根

IST master and CST root	IST master 与 CST 的根
MST Region 1	MST 域 1
Legacy IEEE 802.1D	传统 IEEE 802.1D
MST Region 2	MST 域 2
MST Region 3	MST 域 3

跳数

IST 和 MST 实例不会使用配置 BPDU 中的消息老化和最大老化信息来计算生成树拓扑。它们使用的是去往根的路径开销，和一种类似于 IP 生存时间（TTL）机制的跳数机制。

用户可以使用全局配置命令 **spanning-tree mst max-hops** 来配置域内的最大跳数，并且将其应用于 IST 和这个域中的所有 MST 实例。跳数会得到与消息老化信息（触发重新配置）相同的结果。实例的根设备会始终以开销值 0 发送 BPDU（或 M 记录），并且将跳数设置为最大值。当一台设备接收到这个 BPDU，它就会把接收到的消息跳数减 1，然后将这个值作为它在这个 BPDU 中生成的剩余跳数。当跳数值为 0 时，设备就会丢弃这个 BPDU，然后将端口保存的信息老化。

在整个域中，BPDU RSTP 部分中的消息老化与最大老化信息都会保持不变，域边界的指定端口也会传播相同的值。

边界端口

在 Inspur 预标准的实施方案中，边界端口会将一个 MST 域连接到一个运行 RSTP 的生成树域，一个运行 PVST+或快速 PVST+的生成树域，或者另一个采用了不同 MST 配置的 MST 域。

边界端口也会连接一个局域网，这个局域网中的指定路由器要么是一台生成树设备，要么是一台包含不同 MST 配置的设备。

在 IEEE 802.1s 标准中并没有关于边界端口的定义。IEEE 802.1Q-2002 标准定义了一个端口可以接收到的两类消息：

- （来自同一个域的）内部消息
- （来自另一个域的）外部消息

当消息是内部消息时，这个消息只能通过 CIST 接收到。如果 CIST 的角色为根端口或替代端口，或者如果外部 BPDU 是一个拓扑变更，这有可能会给 MST 实例构成影响。

MST 域中包含设备和局域网。网段属于指定端口的域。因此，与指定端口所在网段处于不同域中的端口就是边界端口。根据这种定义，域的两个内部端口可以通过属于不同域的那个端口共享一个网段，因此一个端口也就有可能同时接收到内部消息和外部消息。

Inspur 预标准实施方案的一大变化在于，CIST 域根设备 ID 字段现在被插入到了 RSTP 或传统 IEEE 802.1Q 设备标记发送方设备 ID 的地方。整个域会向虚拟设备一样执行操作，它会连续向邻居设备发送发送方设备 ID。在这个示例中，设备 C 会接收到带有同一个一致发送发设备 ID 的 BPDU，无论 A 或 B 是不是这个网段的指定设备。

IEEE 802.1s 的实施

在 Inspur 对 IEEE MST 标准的实施方案中，包含了需要满足这一标准的特性，以及一些尚未结合到已发布标准中的必备预标准功能。

端口角色的命名变化

边界角色已经没有再重现在最终的 MST 标准中，但这种边界的概念在 Inspur 的实施方案中得到了保留了。但是，在域边界的 MST 实例端口可能不会按照 CIST 端口的状态操作。当前存在两种边界角色：

- 边界端口是 CIST 域根的根端口——当 CIST 实例端口接收到 Proposal，并且已经同步时，它会向回发送 agreement，而且只有在所有对应的 MSTI 端口都同步后才会进入转发状态。MSTI 端口现在有一种特殊的 master 角色。
- 边界端口不是 CIST 域根的根端口——MSTI 端口会按照 CIST 端口的状态进行操作。这种标准提供的信息比较少，在 MSTI 端口接收不到 BPDU 时，它也许很难理解为什么 MSTI 端口会被阻塞。此时，虽然边界的角色已经不复存在，但用户可以在 **show** 命令输出信息中的 *type* 一列中看到将端口标识为边界（boundary）端口。

与传统和标准设备的互操作

由于对预标准设备进行自动检测有可能会失败，因此可以用以使用一条接口配置命令来设置预标准端口。一个域不能由标准设备和预标准设备组成，但它们可以通过使用 CIST 来进行互操作。只有在有些情况下，无法通过不同的实例实现负载分担。当端口接收到预标准的 BPDU 时，CLI 会根据端口的配置显示不同的标记。当设备通过一个没有配置预标准 BPDU 传输的端口上接收到了一条预标准的 BPDU 时，系统日志消息也会出现。

假设 A 是一台标准设备，而 B 是一台预标准的设备，这两台设备都配置在了同一个域中。A 是 CIST 的根设备，而 B 在网段 X 上有一个根端口（BX）而在网段 Y 上有一个替代端口（BY）。如果网段 Y 出现翻动，而 BY 上的端口会在对外发送一个预标准的 BPDU 之前成为替代端口，那么 AY 就无法检测到预标准设备连接到 Y，因此会继续发送标准 BPDU。在边界上，端口 BY 是固定的，A 和 B 之间不可能执行负载分担。网段 X 上也存在相同的问题，但 B 可能会传输拓扑的变化。

表 47：标准和预标准设备的互操作

Segement X	网段 X
MST Region	MST 域
Switch A	交换机 A
Switch B	交换机 B
Segement Y	网段 Y

注释： 我们推荐用户尽可能减少标准和预标准 MST 实施方案的互动。

检测单向链路失效

这种特性还没有添加到 IEEE MST 标准当中，但包含在了这个 Inspur INOS 版本中。软件系统会通过接收到的 BPDU 来校验端口角色和状态的连续性，以检测网路中是否出现了有可能引发桥接环路的单向链路失效。

当指定端口检测到冲突时，它会保持自己的端口角色，但同时回退到转发状态。因为在出现不连续的情况时，网络是希望打破连通性来打开桥接环路的。

下面这张图显示了一个导致了桥接环路的单向链路失效问题。设备 A 是根设备，它在通向设备 B 的链路上丢失了 BPDU。RSTP 和 MST BPDU 包含了发送方端口的角色和状态。通过这些信息，设备 A 可以检测到设备 B 没有对自己发送的更优 BPDU 作出响应，而设备 B 在所连网段充当的是指定设备，而不是根设备。于是，设备 A 阻塞了这个端口，这就避免了桥接环路的出现。

图 48：检测单向链路失效

Switch A	交换机 A
Switch B	交换机 B
Superior BPDU	更优 BDPU
Inferiror BPDU Designated+Learning bit set	较差 BPDU 指定角色+学习位置位

MSTP 与设备堆栈

设备堆栈在网络其他部分看来是一台设备，所有堆栈成员对于一棵给定的生成树使用相同的桥 ID。其桥 ID 取自于主用交换机的 MAC 地址。

如果一台不支持 MSTP 的设备被添加到了支持 MSTP 的设备堆栈中，或者一台支持 MSTP 的设备被添加到了不支持 MSTP 的设备堆栈中，那么这台设备都会进入一种版本不匹配状态。如有可能，此时这台设备会自动升级或降级到设备堆栈运行的那个软件版本。

与 IEEE 802.1D STP 的互操作性

运行 MSTP 的设备支持一种内置的协议迁移机制，这种机制可以让设备与传统的 IEEE 802.1D 设备进行互操作。如果这台设备接收到了一个传统的 IEEE 802.1D 配置 BPDU（协议版本设置为 0 的 BPDU），那么这台设备就只会在这个端口发送 IEEE 802.1D BPDU。MSTP 设备也可以在接收到一个传统 BPDU，一个来自不同域的 MSTP BPDU（第 3 版）或者一个 RSTP BPDU（第 2 版）时，检测到一个端口位于区域边界。

不过，当设备没有再接收到 IEEE 802.1D BPDU 时，它并不会自动回退到 MSTP 模式，因为它无法检测出传统的设备是否已经从链路上被移除，除非这台传统设备是指定设备。当连接到

一个端口的设备加入到这个域时，设备可能也会继续给这个端口分配边界角色。用户要想重新启动协议迁移进程（强制与邻居设备重新协商），可以输入特权 EXEC 命令 **clear spanning-tree detected-protocols**。

如果链路上的所有传统设备都是 RSTP 设备，那么它们可以像处理 RSTP BPDU 那样处理 MSTP BPDU。因此，MSTP 设备要么会在一个边界端口发送版本 0 的配置和 TCN BPDU，要么则会发送版本 3 的 MSTP BPDU。边界端口会连接到一个局域网，而这个局域网中的指定设备要么是一台单生成树设备，要么是一台拥有不同 MSTP 配置的设备。

RSTP 概述

RSTP 利用了点到点的布线方式，并且提供了生成树快速收敛。因此，生成树的重新配置可以在 1 秒之内完成（而 IEEE 802.1D 生成树中的默认设置则需要 50 秒的时间）。

端口角色与活动拓扑

RSTP 通过分配端口角色和学习活动拓扑的方式，实现了生成树的快速收敛。RSTP 建立在 IEEE 802.1D STP 的基础上，它会将设备优先级最高的设备（也就是优先级值最小的设备）选为根设备。接下来，RSTP 会给各个端口分配下列端口角色之一：

- 根端口：在设备向根设备转发数据包提供最佳路径（即路径开销最小）的端口；
- 指定端口：连接指定设备的端口。在从局域网向根设备转发数据包时，这种端口拥有最小的路径开销。这种指定设备连接局域网的端口称为指定端口；
- 替代端口：提供当前路径之外，另一条通往根设备替代路径的端口；
- 备份端口：提供另一条通往生成树叶网络备份路径的端口。只有当两个端口通过一条链路彼此相连，或者当一台设备域共享 LAN 网段有两条或多条连接时，网络中才会出现备份端口；
- 禁用端口：没有在生成树的操作中扮演任何角色。

根端口或指定端口角色的端口会包含在活动拓扑当中。而角色为替代端口或备份端口的端口则不会包含在活动拓扑当中。

在一个端口角色连续的稳定拓扑中，RSTP 可以确保每个根端口和指定端口立刻过渡到转发状态，而替代端口和备份端口则永远处于丢弃状态（相当于 IEEE 802.1D 中的阻塞状态）。端口状态会控制转发和学习进程的操作。

表 59：端口状态的比较

操作状态	**STP 端口状态（IEEE 802.1D）**	**RSTP 端口状态**	**这种端口是否会包含在活动拓扑中**
启用	阻塞	丢弃	否
启用	侦听	丢弃	否
启用	学习	学习	是
启用	转发	转发	是
禁用	禁用	丢弃	否

为了保证 Inspur 实施方案的一致性，这种指导方针也将端口状态定义为了*阻塞*，而不是*丢弃*。指定端口启用时为侦听状态。

快速收敛

RSTP 提供了设备、设备端口或局域网失效后，连接的快速恢复机制。这种协议为边缘端口、新根端口和通过点到点链路连接的端口提供了快速收敛机制：

- 边缘端口：如果用户在 RSTP 设备上使用接口配置命令 **spanning-tree portfast** 将一个端

口配置为了边缘端口，那么边缘端口就会立刻过渡到转发状态。边缘端口相当于启用了 PortFast 的端口，用户应该在连接到终端工作在的端口上配置这条命令；

- 根端口：如果 RSTP 选择了一个新的根端口，它会阻塞老的根端口，并且立刻将新的根端口过渡到转发状态；
- 点到点链路：如果用户用一个端口通过一条点到点链路连接了另一个端口，而这个本地端口成为了指定端口，那么这个端口就会使用 Proposal-Agreement 握手机制来与其他端口协商快速过渡的方法，以确保拓扑是无环的。

  设备 A 通过一条点到点链路与设备 B 相连，所有所有端口都处于阻塞状态。假设设备 A 的优先级数值小于设备 B 的优先级值，那么设备 A 会向设备 B 发送一条 Proposal 消息（设置了 Proposal 标记的配置 BPDU），提议自己为指定设备。

  在接收到这个 Proposal 消息之后，设备 B 会从接收到 Proposal 消息的端口选择出新的根端口，并且强制所有非边缘端口进入阻塞状态，并且通过新的根端口发送一条 Agreement 消息（设置了 Agreement 标记的配置 BPDU）。

  在接收到设备 B 的 Agreement 消息之后，设备 A 也会立刻将其指定端口过渡到转发状态。此时网络中不会形成环路，是因为设备 B 阻塞了它的所有非边缘端口，因为设备 A 和设备 B 之间有一条点到点链路。

  当设备 C 连接到设备 B 时，它们之间也会交换一系列类似的握手消息。设备 C 会将连接设备 B 的端口选择为根端口，而这两段会立刻过渡到转发状态。在每次重复握手进程时，都会有另一台设备加入活动拓扑。当网络收敛时，Proposal-Agreement 握手进程会从根一直向生成树的叶网络扩散。

  交叉快速过渡（CSRT）特性可以确保设备堆栈中的堆栈成员在 Proposal-Agreement 握手期间从所有堆栈成员那里接收到了确认消息，然后才会将端口过渡到转发状态。当设备进入 MST 模式时，CSRT 就会自动启用。

  设备会从端口双工模式学习到链路类型，全双工端口会被视为是一条点到点链路，而半双工端口则会被视为是一条共享连接。用户可以使用接口配置命令 **spanning-tree link-type** 来覆盖通过双工设置学习到的默认设置。

图 49：Proposal 与 Agreement 握手以实现快速收敛

Switch A	交换机 A
Switch B	交换机 B
Root	根交换机
Designated switch	指定交换机
Designated switch	指定交换机
Root	根交换机
Switch C	交换机 C
Designated switch	指定交换机
DP=designated port	DP=指定端口
RP=root port	RP=根端口
F=forwarding	F=转发
Root	根交换机

同步端口角色

当设备通过自己的一个端口接收到一个 proposal 消息，而这个端口又被选为了新的根端口时，那么 RSTP 就会强制所有其他端口用新端口的信息进行同步。

如果所有其他端口都进行了同步，设备就会使用根端口接收到的更优的根信息来进行同步。

在发生下列情况时，设备的一个端口会进行同步：

- 这个端口处于阻塞状态；
- 这是一个边缘端口（用户配置为网络边界的端口）

如果指定端口处于转发状态，但用户又没有将它配置为边缘端口，那么当 RSTP 强制它使用新的根信息进行同步时，它就会过渡为阻塞状态。总地来说，当 RSTP 强制一个端口用根信息进行同步时，而这个端口又不满足任何上述条件，那么它的端口状态就会被设置为阻塞状态。

在确认了所有端口都已经同步之后，设备会向其根端口连接的指定设备发送一条 agreement 消息。当通过点到点链路连接的设备都同意了它们的端口角色时，RSTP 会立刻将端口过渡到转发状态。

图 50：快速收敛过程中的事件发生顺序

5. Forward	5.转发
Edge port	边缘端口
2.Block 9.Forward	2.阻塞 9.转发
3.Block 11.Forward	3.阻塞 11.转发
Root port	根端口
Designated port	指定端口

桥协议数据单元的格式与处理

RSTP 的 BPDU 格式与 IEEE802.1D BPDU 的格式相同，只不过协议版本变为了 2。有一个新的 1 字节版本 1 长度（Version 1 Length）字段会被设置为 0，这表示不存在第 1 版协议信息。

表 60：RSTP BPDU 标记

位	功能
0	拓扑变更（TC）
1	Proposal
2-3： 00 01 10 11	端口角色： 未知 替代端口 根端口 指定端口
4	学习
5	转发
6	Agreement
7	拓扑变化确认（TCA）

发送方设备会设置 RSTP BPDU 中的 proposal 标记，其目的是宣告自己是这个网段的指定设备。在 proposal 消息中的端口角色永远会被设置为指定端口。

发送方设备会设置 RSTP BPDU 中的 agreement 标记，其目的接受之前的提议（proposal）。在 agreement 消息中的端口角色永远会被设置为根端口。

RSTP 没有独立的拓扑变更通告（TCN）BPDU。它会使用拓扑变更（TC）标记来显示拓扑的变更。不过，为了和 IEEE 802.1D 设备进行互操作，RSTP 设备会处理和生成 TCN BPDU。

RSTP 会根据发送端口的状态来设置学习和转发标记。

处理更优的 BPDU 信息

如果一个端口接收到了比当前给这端口保存的根信息更优的根信息（设备 ID 较低、路径开销较低等），RSTP 就会触发重配置。如果这个端口经过提议被选为了新的根端口，那么 RSTP

就会强制所有其他端口进行同步。

如果 BPDU 接收到一条设置了 proposal 标记的 RSTP BPDU 消息，那么设备就会在其他端口同步之后发送一条 agreement 消息。如果 BPDU 是一个 IEEE 802.1D BPDU，那么设备就不会设置 proposal 标记，并且对这个端口启动转发延迟计时器。新的根端口需要两倍的转发延迟时间才能过渡到转发状态。

如果端口接收到的更优信息导致这个端口成为了备份端口或替代端口，那么 RSTP 就会将这个端口置入阻塞状态，而不会发送 agreement 消息。指定端口会继续发送设置了 proposal 标记的 BPDU，直至转发延迟计时器过期为止，此时这个端口就会过渡到转发状态。

处理较差的 BPDU 信息

如果一个指定端口接收到了一个较差的 BPDU（比如其设备 ID 更高，或者路径开销值大于当前给这个端口保存的路径开销值），它会立刻用自己的信息作出响应。

拓扑变更

在这一部分，我们会介绍 RSTP 和 IEEE 802.1D 在处理生成树变更时的区别：

- 检测：在 IEEE 802.1D 环境中，*一切*状态变化都会导致拓扑变更。而在 RSTP 环境中，只有从阻塞状态过渡到转发状态才会导致拓扑变更（也就是说，只有连接数量增加才会被视为拓扑变更）。边缘端口的状态变更不会导致拓扑变更。当 RSTP 设备检测到拓扑变更时，它会删除所有在非边缘端口上学习到的信息——除了接收到 TC 通告的那个端口之外。
- 通告：IEEE 802.1D 使用的是 TCN BPDU，但 RSTP 没有使用这种 BPDU。但为了支持与 IEEE 802.1D 进行互操作，RSTP 设备会处理和生成 TCN BPDU；
- 确认：当 RSTP 设备接在指定端口从一台 IEEE 802.1D 设备那里接收到一条 TCN 消息时，它会使用 TCA 位置位的 IEEE 802.1D 配置 BPDU 进行响应。不过，如果在与 IEEE 802.1D 设备直连的根端口上，TC-while 计时器（与 IEEE 802.1D 中的拓扑变更计时器相同）是活动的，同时该端口接收到了一个 TCA 位置位的配置 BPDU，那么 TC-while 计时器就会被重置。

  只有在支持 IEEE 802.1D 设备才需要执行这种操作。RSTP BPDU 永远不会置位 TCA 位；
- 传播：当 RSTP 通过指定端口或根端口从另一台设备那里接收到一条 TC 消息时，它会将这个变化传播给所有非边缘的指定端口，以及根端口（不包括接收到 TC 消息的那个端口）。设备会对所有这类端口启动 TC-while 计时器，并冲刷掉在这些端口学习到的信息；
- 协议迁移：为了能够向后兼容 IEEE 802.1D 设备，RSTP 会有选择地从不同端口发送 IEEE 802.1D 配置 BPDU 和 TCN BPDU。

  当一个端口启动时，迁移延迟计时器就会启动（设置 RSTP BPDU 发送的最小时间）。当计时器处于活动状态时，设备会处理所有从这个端口接收到的 BPDU，同时忽略协议类型。

  如果设备在迁移延迟计时器过时之后接收到了一条 IEEE 802.1D BPDU，它会假设自己连接的是一台 IEEE 802.1D 设备，并自此只使用 IEEE 802.1D BPDU。但是，如果 RSTP 设备在计时器过期之后，在一个端口使用 IEEE 802.1D BPDU，同时又接收到了 RSTP BPDU，那么它就会重新启动计时器，并且开始在这个端口使用 RSTP BPDU。

协议迁移进程

运行 MSTP 的设备会支持内置的协议迁移机制，这种机制可以让设备与传统的 IEEE 802.1D 设备进行互操作。如果这台设备接收到了一条传统的 IEEE 802.1D 配置 BPDU（即协议版本设置为 0 的 BPDU），它就只会通过这个端口发送 IEEE 802.1D BPDU。MSTP 设备也可以在接收

到传统 BPDU、不同域的 MST BPDU（第 3 版）或 RST BPDU（第 2 版）时，检测到这个端口位于域的边界。
不过，当一台设备没有再接收到 IEEE 802.1D BPDU 时，它并不会自动回退到 MSTP 模式，因为它无法检测出传统的设备是否已经从链路上被移除，除非这台传统设备是指定设备。当连接到一个端口的设备加入到这个域时，设备可能也会继续给这个端口分配边界角色。

默认的 MSTP 配置

表 61：默认的 MSTP 配置

特性	**默认设置**
生成树模式	MSTP
设备优先级（可以基于 CIST 端口进行配置）	32768
生成树端口优先级(可以基于 CIST 端口进行配置）	128
生成树端口开销(可以基于 CIST 端口进行配置）	1000Mb/s：20000 100Mb/s：20000 10Mb/s：20000 1000Mb/s：20000 100Mb/s：20000 10Mb/s：20000
Hello 时间	3 秒
转发延迟时间	20 秒
最大老化时间	20 秒
最大跳数	20 跳

如何配置 MSTP 特性

设置 MST 域的配置与启用 MSTP（CLI）

对于同一个 MST 域中的两台或多台设备，它们必须拥有相同的 VLAN 与实例映射关系、相同的修订版本号和相同的名称。
一个域中可以拥有一个或多个 MST 配置相同的成员；每个成员都必须能够处理 RSTP BPDU。
一个网络中的 MST 域没有数量限制，但每个域只能支持最多 65 个生成树实例。用户每次只能将一个 VLAN 分配给一个生成树实例。

总步骤

1. enable
2. configure terminal
3. spanning-tree mst configuration
4. instance *instance-id* **vlan** *vlan-range*
5. name *name*
6. revision *version*
7. show pending

8. exit

9. spanning-tree mode mst

10. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree mst configuration** 示例： `Device(config)#` **spanning-tree mst configuration**	进入 MST 配置模式
步骤 4	**instance** *instance-id* **vlan** *vlan-range* 示例： `Device(config-mst)#` **instance 1 vlan 10-20**	将 VLAN 映射为一个 MST 实例。 • 在 *instance-id* 部分，取值范围为 0 到 4094； • 在 **vlan** *vlan-range* 部分，取值范围为 1 到 4094；在将 VLAN 映射为一个 MST 实例时，映射是递增的，命令中指定的 VLAN 会被添加到之前映射的 VLAN 当中，或者从之前映射的 VLAN 中移除。 要指定 VLAN 范围，可以使用连字符。例如，**instance 1 vlan 1-63** 是将 VLAN 1 到 63 映射为 MST 实例 1；要指定VLAN范围，可以使用逗号。例如，**instance 1 vlan 10, 20, 30**是将VLAN 10、20和30映射为MST实例1
步骤 5	**name** *name* 示例： `Device(config-mst)#` **name region1**	指定配置名。*name* 这个字符串的最大长度为 32 个字符，而且是区分大小写的
步骤 6	**revision** *version* 示例： `Device(config-mst)#` **revision 1**	指定配置修订版本号。取值范围是从 0 到 65535
步骤 7	**show pending** 示例：	通过显示未定信息来验证配置

	Device(config-mst)# **show pending**	
步骤 8	**exit** 示例： Device(config-mst)# **exit**	应用所有的修改，并返回全局配置模式
步骤 9	**spanning-tree mode mst** 示例： Device(config)# **spanning-tree mode mst**	启用 MSTP，则 RSTP 也会启用。 修改生成树的模式会导致流量中断，因为所有生成树实例都会停止执行之前的模式，并且在新的模式下重新启动。 用户不能同时运行 MSTP 与 PVST+，或者同时运行 MSTP 与快速 PVST+
步骤 10	**end** 示例： Device(config)# end	返回特权 EXEC 模式

配置根设备（CLI）

这个流程是可选的。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

用户必须清楚设置的 MST 实例 ID。示例中的步骤 2 以 0 作为实例 ID，是因为这是按照相关主题下面列出的方法设置的实例 ID。

总步骤

1. enable

2. configure terminal

3. spanning-tree mst *instance-id* **root primary**

4. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree mst** *instance-id* **root primary** 示例：	将一台设备配置为根设备。 • 在 *instance-id* 部分，用户可以设置一个实例，可以用连字符设置一个实例范围，也可以用逗号相隔输入一系列的实例。实例的取值范围是

	Device(config)# **spanning-tree mst 0 root primary**	从 0 到 4094
步骤 4	**end** 示例： Device(config)# end	返回特权 EXEC 模式

配置辅助根设备（CLI）

在将一台支持扩展的系统 ID 的设备配置为辅助根时，设备优先级会从默认值（32768）修改为 28672。如果这个实例的主用根发生了故障，那么设备使用这个优先级就更有可能会成为这个实例的根设备。这一点的前提是其他网络设备使用的都是默认的设备优先级 32768，因此这些设备就很难成为根设备。

用户可以在多台设备上执行这条命令，来将多台设备配置为备份根设备。用户也可以使用配置主用根设备时使用的命令 **spanning-tree mst** *instance-id* **root primary** 来设置辅助根设备的网络直径与 hello 时间值。

这个流程是可选的。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

用户必须清楚设置的 MST 实例 ID。示例中以 0 作为实例 ID，是因为这是按照相关主题下面列出的方法设置的实例 ID。

总步骤

1. enable

2. configure terminal

3. spanning-tree mst *instance-id* **root secondary**

4. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree mst** *instance-id* **root secondary** 示例： Device(config)# **spanning-tree mst 0 root secondary**	将一台设备配置为辅助根。 • 在 *instance-id* 部分，用户可以设置一个实例，可以用连字符设置一个实例范围，也可以用逗号相隔输入一系列的实例。实例的取值范围是从 0 到 4094
步骤 4	**end**	返回特权 EXEC 模式

	示例： `Device(config)# end`	

配置端口优先级（CLI）

当环路出现时，MSTP 会使用端口优先级来选择一个接口，并将其置入转发状态。用户可以给自己希望首先被选中的接口设置较高的（数值较小的）优先级值，给自己希望之后被选中的接口设置较低的（数值较大的）优先级值。如果所有接口的优先级值相同，那么 MSTP 就会将接口编号最低的接口置入转发状态，并阻塞其他接口。

注释： 如果用户的设备是设备堆栈的成员，那就必须使用接口配置命令 **panning-tree mst** [*instance-id*] **cost** *cost*（而不是接口配置命令 **spanning-tree mst** [*instance-id*] **port-priority** *priority*）来选择将一个接口置入转发状态。用户可以给自己希望首先选择的接口配置一个较低的开销值，而给自己希望之后选择的接口配置一个较高的开销值。要想了解详细信息，可以参见相关主题。

这个流程是可选的。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

用户必须清楚设置的 MST 实例 ID 和接口。示例中以 0 作为实例 ID，以 GigabitEthernet1/0/1 为接口，是因为这是按照相关主题下面列出的方法设置的实例 ID。

总步骤

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*
4. **spanning-tree mst** *instance-id* **port-priority** *priority*
5. **end**

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)# interface GigabitEthernet1/0/1`	指定要配置的接口，进入接口配置模式。
步骤 4	**spanning-tree mst** *instance-id* **port-priority** *priority*	配置端口优先级。 • 在 *instance-id* 部分，用户可以设置一个实例，

	示例： `Device(config-if)#` **spanning-tree mst 0 port-priority 64**	可以用连字符设置一个实例范围，也可以用逗号相隔输入一系列的实例。实例的取值范围是从 0 到 4094； • 在 *priority* 部分，取值范围是从 0 到 240，增量为 16，默认值是 128。有效的取值包括 0、16、32、48、64、80、96、112、128、144、160、176、192、208、224 和 240。配置其他值系统都会拒绝
步骤 5	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

特权 EXEC 命令 **show spanning-tree mst interface** *interface-id* 只会显示那些链路处于 up 状态的端口信息。否则，用户也可以使用特权 EXEC 命令 **show running-config interface** 来确认自己所作的配置。

配置路径开销（CLI）

MSTP 路径开销的默认值取自于接口的媒体速率。当环路出现时，MSTP 会使用开销来选择一个接口，并将其置入转发状态。用户可以给自己希望首先被选中的接口设置较低的开销值，给自己希望之后被选中的接口设置较高的开销值。如果所有接口的开销值相同，那么 MSTP 就会将接口编号最低的接口置入转发状态，并阻塞其他接口。

这个流程是可选的。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

用户必须清楚设置的 MST 实例 ID 和接口。示例中以 0 作为实例 ID，以 GigabitEthernet1/0/1 为接口，是因为这是按照相关主题下面列出的方法设置的实例 ID。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. spanning-tree mst *instance-id* **cost** *cost*

5. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式

步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的接口，进入接口配置模式。有效接口包括物理端口 port channel 逻辑接口。port-channel 的取值范围是从 1 到 48
步骤 4	**spanning-tree mst** *instance-id* **cost** *cost* 示例： `Device(config-if)#` **spanning-tree mst 0 cost 17031970**	配置开销。 如果出现环路，MSTP 在选择要将哪个接口置入转发状态时就会使用路径开销进行判断。路径开销越低表示传输速率越高。 • 在 *instance-id* 部分，用户可以设置一个实例，可以用连字符设置一个实例范围，也可以用逗号相隔输入一系列的实例。实例的取值范围是从 0 到 4094； • 在 *cost* 部分，取值范围是从 1 到 200000000，这个值取自于接口媒体的速率
步骤 5	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

特权 EXEC 命令 **show spanning-tree mst interface** *interface-id* 只会显示那些链路处于 up 状态的端口信息。否则，用户也可以使用特权 EXEC 命令 **show running-config** 来确认自己所作的配置。

配置设备优先级（CLI）

用户可以配置设备的优先级，让一台独立设备或一台堆栈中的设备更有可能被选为根设备。

注释： 在使用这条命令时务请小心。在正常的网络配置中，我们推荐用户使用全局配置命令**spanning-tree mst** i*nstance-id* **root primary**和**spanning-tree mst** *instance-id* **root secondary**来将一台设备设置为根或辅助根设备。只有当这些命令没有生效时，用户才应该考虑修改设备优先级。

这个流程是可选的。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

用户必须清楚设置的 MST 实例 ID。示例中以 0 作为实例 ID，是因为这是按照相关主题下面列出的方法设置的实例 ID。

总步骤

1. enable

2. configure terminal

3. spanning-tree mst *instance-id* **priority** *priority*

4. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable**	进入特权 EXEC 模式。在提示时输入密码

	示例： `Device> enable`	
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**spanning-tree mst** *instance-id* **priority** *priority* 示例： `Device(config)# spanning-tree mst 0 priority 40960`	配置设备优先级。 • 在 *instance-id* 部分，用户可以设置一个实例，可以用连字符设置一个实例范围，也可以用逗号相隔输入一系列的实例。实例的取值范围是从 0 到 4094； • 在 *priority* 部分，取值范围是从 0 到 61440，增量为 4096，默认值是 32768。数值越低，设备越有可能被选为根设备。 有效的值包括 4096、8192、12288、16384、20480、24576、28672、32768、36864、40960、45056、49152、53248、57344 和 61440。只有输入这些值设备才会接受
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置 Hello 时间（CLI）

hello 时间是根设备生成和发送配置消息的时间间隔。

这个流程是可选的。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

总步骤

1. enable

2. configure terminal

3. spanning-tree mst hello-time *seconds*

4. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal**	进入全局配置模式

	示例： `Device# configure terminal`	
步骤 3	**spanning-tree mst hello-time** *seconds* 示例： `Device(config)# spanning-tree mst hello-time 4`	配置所有 MST 实例的 hello 时间。hello 时间是根设备生成和发送配置消息的时间间隔。这些消息表示这台设备当前仍然正常工作。 在 *seconds* 部分，取值范围是从 1 到 10；默认值为 3
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置转发延迟时间（CLI）

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

总步骤

1. enable

2. configure terminal

3. spanning-tree mst forward-time *seconds*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**spanning-tree mst forward-time** *seconds* 示例： `Device(config)# spanning-tree mst forward-time 25`	配置所有 MST 实例的转发时间。转发延迟是接口在将自己的生成树学习和侦听状态过渡到转发状态之前，等待的秒数。 在 *seconds* 部分，取值范围是从 4 到 30，默认值是 20
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置最大老化时间（CLI）

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

总步骤

1. enable

2. configure terminal

3. spanning-tree mst max-age *seconds*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree mst forward-time** *seconds* 示例： `Device(config)#` **spanning-tree mst forward-time 25**	配置所有 MST 实例的最大老化时间。转发延迟是端口从生成树状态从学习和侦听状态过渡到转发状态之前，等待的最大秒数。 在 *seconds* 部分，取值范围是从 4 到 30，默认值是 20
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置最大老化时间（CLI）

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

总步骤

1. enable

2. configure terminal

3. spanning-tree mst max-age *seconds*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable**	进入特权 EXEC 模式。在提示时输入密码

	示例： `Device> enable`	
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**spanning-tree mst max-age** *seconds* 示例： `Device(config)# spanning-tree mst max-age 40`	配置所有 MST 实例的最大老化时间。最大老化时间是指设备从没有接收到生成树配置消息开始，会等待多久才会开始执行重新配置。 在 *seconds* 部分，取值范围是从 6 到 40，默认值是 20
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置最大跳数（CLI）

这个流程是可选的。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

总步骤

1. enable

2. configure terminal

3. spanning-tree mst max-hops *hop-count*

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**spanning-tree mst max-hops** *hop-count* 示例： `Device(config)# spanning-`	配置 BPDU 被丢弃，且针对端口保存的信息老化之前，在域中可以转发的跳数，。 在 *hop-count* 部分，取值范围是从 1 到 255，默认值是 20

	tree mst max-hops 25	
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

设置链路类型以确保快速过渡（CLI）

如果用户用一个端口通过一条点到点链路连接了另一个端口，而这个本地端口成为了指定端口，那么 RSTP 就会使用 Proposal-Agreement 握手机制来与其他端口协商快速过渡的方法，以确保拓扑是无环的。

在默认情况下，设备的链路类型是受双工模式控制的：全双工端口会被视为是一条点到点链路，而半双工端口则会被视为是一条共享连接。如果用户用一条半双工链路物理地点到点连接到一台运行 MSTP 的远程设备，用户可以覆盖默认设置的链路类型，让端口快速过渡到转发状态。

这个流程是可选的。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

用户必须清楚设置的 MST 实例 ID 和接口。示例中以 0 作为实例 ID，以 GigabitEthernet1/0/1 为接口，是因为这是按照相关主题下面列出的方法设置的实例 ID。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. spanning-tree link-type point-to-point

5. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的接口，进入接口配置模式。有效接口包括物理端口、VLAN 和 port channel 逻辑接口。VLAN ID 的取值范围是 1 到 4094，port-channel 的取值范围是从 1 到 48
步骤 4	**spanning-tree link-type point-to-point**	将端口的链路类型设置为点到点

	示例： `Device(config-if)#` **spanning-tree link-type point-to-point**	
步骤 5	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

指定邻居类型（CLI）

拓扑应该同时包含符合预标准和 IEEE 802.1s 标准的设备。在默认情况下，端口可以自动检测到预标准设备，但它们仍然可以同时接收标准的和预标准的 BPDU。只要设备与邻居之间出现了不匹配的情况，那接口上就只能运行 CIST。

用户可以选择设置一个端口，让它只发送预标准的 BPDU。即使端口工作在匹配 STP 的模式下，**show** 命令还是会显示出预标准的标记。

这个流程是可选的。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. spanning-tree mst pre-standard

5. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的接口，进入接口配置模式。有效接口包括物理端口
步骤 4	**spanning-tree mst pre-standard**	设置端口，让它只发送预标准的 BPDU

	示例： `Device(config-if)#` **spanning-tree mst pre-standard**	
步骤 5	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

重新启动协议迁移进程（CLI）

这个流程会重新启动协议迁移进程，并且强制与邻居设备进行重新协商。它会让设备回退到 MST 模式。当设备在接收到 IEEE 802.1D BPDU 之后，没有继续接收到 IEEE 802.1D BPDU，它就需要执行这个流程。

用户可以按照下面的步骤在设备上重新启动协议迁移进程（强制与邻居设备进行重新协商）。

在开始前

设备上必须设置和启用多生成树（MST）。要想了解详细信息，可以参见相关主题。

如果用户想要使用接口版本命令，就必须清楚设置的 MS 接口。示例中以 GigabitEthernet1/0/1 为接口，是因为这是按照相关主题下面列出的方法设置的接口。

总步骤

1. enable

2. 输入下面命令之一：

- **clear spanning-tree detected-protocols**
- **clear spanning-tree detected-protocols interface** *interface-id*

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**输入下面命令之一：** • **clear spanning-tree detected-protocols** • **clear spanning-tree detected-protocols interface** *interface-id* 示例： `Device#` **clear spanning-tree detected-protocols** 或 `Device#` **clear spanning-tree detected-protocols interface**	设备回退到 MSTP 模式，协议迁移进程会重新启动

	GigabitEthernet1/0/1	

接下来做什么

如果设备接收到了更多传统的 IEEE 802.1D 配置 BPDU，那么这个流程有可能需要重复执行（协议版本设置为 0 的 BPDU）。

其他 MSTP 的参考资料

相关文档

相关主题	**文档名**
生成树协议的命令	《LAN 交换命令参考手册，Inspur INOSXE3SE 版（Inspur 6650 交换机）》

错误消息解码器

描述	**链接**
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

标准与 RFC

标准/RFC	**标题**
无	——

技术助手

描述	**链接**
Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。 用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	http://www.icntnetworks.com

MSTP 的特性历史

版本	**修改**
Inspur INOS 12.2	引入该特性

配置可选的生成树特性

关于可选生成树特性的信息

PortFast

PortFast 可以让一个配置为 Access 或 trunk 模式的接口立刻从阻塞状态进入转发状态，跳过侦听状态和学习状态。
用户可以在一个连接到一台工作站或服务器的接口上使用 PortFast 特性，让这些设备能够立刻连接到网络，而无需等待生成树的收敛。
图 51：启用了 PortFast 的接口

Port Fast-enabled ports	启用了 PortFast 的端口
Port Fast-enabled ports	启用了 PortFast 的端口
Workstations	工作站
Workstations	工作站
Server	服务器

当交换机重新启动时，启用了 PortFast 的接口会按照正常的生成状态变化流程进行状态迁移。
用户可以在这个接口或者所有非中继端口上启用这个特性。

BPDU 防护

桥协议数据单元（BPDU）防护特性可以在交换机上全局启用，也可以在个别端口上启用，但协议的操作方式存在一些区别。
当用户在启用了 PortFast 边缘的端口上全局启用 BPDU 防护时，如果处于 PortFast 边缘操作状态的端口接收到了任何 BPDU，那么生成树就会关闭这些端口。在有效的配置方案中，启用了 PortFast 边缘的端口不会接收到 BPDU。如果在启用了 PortFast 边缘的端口上接收到了 BPDU，表示设备上存在无效的配置（譬如有未授权设备连接），那么 BPDU 防护特性就会将这个端口置入 error-disabled 状态。在发生这件事时，交换机就会关闭发生错误的端口。
当用户在没有启用 PortFast 边缘特性的接口上启用 BPDU 防护，那么当这个端口接收到 BPDU 时，它就会进入 error-disabled 状态。
BPDU 防护特性给无效配置提供了一种安全的响应方式，因为 BPDU 防护特性必须由用户手动在接口上进行配置。在服务提供商网络中使用 BPDU 防护特性可以防止 access 端口参与到生成树当中。

BPDU 过滤

用户可以在交换机全局启用 BPDU 过滤特性，也可以在各个接口上启用 BPDU 过滤特性，但协议的操作方式存在一些区别。

在启用了 PortFast 边缘的接口上全局启用 BPDU 过滤，可以让这些处于 PortFast 边缘操作状态的接口不会发送和接收 BPDU。这些接口会在开始过滤出站 BPDU 之前，通过链路发送几个 BPDU。用户应该在交换机上全局启用 BPDU 过滤，让与这些接口直连的主机无法接收到 BPDU。如果设备通过一个启用了 PortFast 边缘的接口接收到了 BPDU，那么这个接口就会丢失它的 PortFast 边缘操作状态，而 BPDU 过滤也会被禁用。

如果用户在没有启用 PortFast 边缘特性的接口上启用了 BPDU 过滤，那么这些接口就不会发送或接收 BPDU 了。

注意： 在接口启用 BPDU 过滤相当于在接口上禁用生成树，因此有可能会导致生成树环路。

用户可以针对整台交换机启用 BPDU 过滤特性，也可以针对一个接口来启用 BPDU 过滤特性。

UplinkFast

在分层网络中的交换机可以分类为骨干交换机、分布层交换机和接入层交换机。下面这个复杂的网络包含了分布层交换机和接入层交换机，它们都有至少一条冗余链路被生成树阻塞，以防网络中出现环路。

图 52：分层网络中的交换机

Backbone switches	骨干交换机
Root bridge	根桥
Distribution switches	分布层交换机
Access switches	接入层交换机
Active link	活动链路
Blocked link	阻塞链路

如果交换机丢失了链路，它就会在生成树选出新的根端口之后立刻开始使用替代路径。用户可以通过启用 UplinkFast，在链路或交换机出现故障，或者在生成树重新配置时，加速新根端口的选择。根端口会立刻过渡到转发状态，而不需要经历侦听或学习状态，这是它与常规生成树处理流程的不同之处。

当生成树重新配置了新的根端口时，其他接口就会在网络中泛洪组播数据包，向每个在这个接口上学习到的地址发送一个组播数据包。用户可以通过减少最大更新速率这个参数（默认值为每秒 150 个数据包）来限制组播流量的突发值。但如果输入 0，那么设备就不会创建工作站学习的数据帧，因此丢失连接之后生成树拓扑的收敛时间也会变慢。

注释： UplinkFast 最适合配置在那些部署在网络接入层或边缘的那些配线柜中的交换机上。这项特性不适合配置在骨干交换机上。这些特性可能对于其他类型的应用用处不大。

在直连链路出现故障时，UplinkFast 可以提供快速收敛，并且使用上行链路组通过冗余的二层链路提供负载分担。一个上行链路组是指（一个 VLAN 中的）多个二层接口，其中只有一个二层接口时钟处于转发状态，具体来说，除了自环端口之外，上行链路组是由（处于转发状态的）根端口、和一系列阻塞端口组成的。在当前转发链路出现故障时，这个上行链路组可以提供替代路径。

这个拓扑当前没有链路出现故障，交换机 A，即根交换机与交换机 B 通过链路 L1 直连，与交换机 C 通过链路 L2 相连。交换机 C 上域交换机 B 直连的二层接口目前处于阻塞状态。

图 53：直连链路故障之前的 UplinkFast 示例

Switch A	交换机 A

(Root)	（根）
Switch B	交换机 B
Switch C	交换机 C
Blocked port	阻塞端口

如果交换机 C 在连接当前活动链路 L2 的根端口上检测到了链路故障（直连链路故障），那么 UplinkFast 就会启动与交换机 C 之间的阻塞接口，让它过渡到转发状态，而不需要经历侦听和学习状态。这个变更需要经历 1 到 5 秒的时间。

图 54：直连链路故障之后的 UplinkFast 示例

Switch A (Root)	交换机 A （根）
Switch B	交换机 B
Switch C	交换机 C
Link failure	链路故障
UplinkFast transitions port directly to forwarding state.	UplinkFast 直接将端口过渡到转发状态

交叉堆栈 UplinkFast

交叉堆栈 UplinkFast（CSUF）可以跨越堆栈中的交换机提供快速生成树过渡（在正常网络条件下，快速收敛可以在 1 秒之内完成）。在快速过渡期间，交换机堆栈中会有一条替代冗余链路被堆栈置于转发状态，而不会引发临时生成树环路或者丢失与骨干之间的连接。通过这项特性，用户可以通过配置获得一个冗余的，能够快速复原的网络。当用户启用 UplinkFast 特性时，CSUF 也会自动启用。

CSUF 可能无法随时提供快速过渡。在有些情况下，设备也会执行普通的生成树过渡，这需要消耗 30 到 40 秒的时间。要想了解具体信息，可以参考相关主题。

交叉堆栈 UplinkFast 是如何工作的

交叉堆栈 UplinkFast（CSUF）可以确保堆栈中有一条链路会被选为去往根的路径。

交换机 1 上的堆栈根端口提供了去往生成树根的路径。交换机 2 和 3 上的替代堆栈根端口可以在当前堆栈根交换机发生故障，或者去往生成树根的链路故障时，提供去往根的替代路径。

根链路链路 1 处于生成树转发状态。链路 2 和链路 3 则是冗余链路，它们处于生成树阻塞状态。如果交换机 1 发生了故障，如果交换机 1 所在的堆栈根端口发生了故障，或者链路 1 发生了故障，那么 CSUF 就会从交换机 2 和交换机 3 的替代堆栈根端口中选择一个，并在 1 秒之内将其置于转发状态。

图 55：交叉堆栈 UplinkFast 拓扑

Backbone	骨干
Spanning- tree root	生成树根
Forward	转发
Forward	转发
Forward	转发
Link 1 (Root link)	链路 1 （根链路）

Link 2 (Alternate redundant link)	链路 2 （替代冗余链路）
Link 3 (Alternate redundant link)	链路 3 （替代冗余链路）
100 or 1000 Mb/s	100 或 1000Mb/s
100 or 1000 Mb/s	100 或 1000Mb/s
100 or 1000 Mb/s	100 或 1000Mb/s
Stack-root port	堆栈根端口
Alternate stack- root port	替代堆栈根端口
Alternate stack- root port	替代堆栈根端口
Switch 1	交换机 1
Switch 2	交换机 2
Switch 3	交换机 3
StackWise Plus port connections	StackWise Plus 端口连接
StackWise Plus port connections	StackWise Plus 端口连接
StackWise Plus port connections	StackWise Plus 端口连接
Switch stack	交换机堆栈

当一条链路丢失，或者发生了生成树事件（在下一个主题中进行介绍），那么快速上行链路过渡协议就会使用邻居列表来向堆栈成员发送快速过渡请求。

发送快速过渡请求的交换机需要将选择为根端口的那个端口快速过渡到转发状态，同时它必须从每个堆栈交换机那里获得确认才能执行快速过渡。

堆栈中的每台交换机都会判断发送方交换机是不是比自己更适合成为这个生成树实例的根，它们会通过比较根、开销、桥 ID 来进行判断。如果发送方交换机是成为堆栈根的最佳选择，那么堆栈中的每台交换机都会返回一条确认；否则，它就会发送一条快速过渡请求。此时，发送方交换机还没有从所有堆栈交换机那里接收到确认。

当发送方交换机从所有堆栈交换机那里接收到确认时，发送方交换机上运行的快速上行链路过渡协议就会立刻将它的替代堆栈根端口过渡到转发状态。如果发送方交换机没有从所有堆栈交换机那里接收到确认，那么交换机就会执行普通的生成树过渡（从阻塞、到侦听、到学习、再到转发），而生成树拓扑也会按照正常的速率进行收敛（2 倍转发延迟+最大老化时间）。快速上行链路过渡协议是基于每个 VLAN 实施的，每次只会影响一个生成树实例。

导致快速收敛的事件

根据网络事件或者故障的不同，CSUF 有可能会执行快速收敛，也有可能不会执行快速收敛。在下列情况下，会发生快速收敛（在正常网络条件下，快速收敛可以在 1 秒之内完成）：

- 堆栈根端口链路发生故障。
- 如果堆栈中有两台交换机拥有去往根的替代路径，但只有一台交换机执行了快速过渡；
- 失效链路（连接堆栈根与生成树根的链路）恢复；
- 网络重配置导致网络中选择出了新的堆栈根交换机；
- 网络重配置导致当前堆栈根交换机上的一个端口被选为堆栈根端口。

注释：如果很多事件同时发生，有可能不会发生快速过渡。例如，如果一个堆栈成员掉

电，同时连接堆栈根与生成树根的链路恢复，那么网络就会执行普通的生成树收敛。

在下列情况下，会发生普通的生成树收敛（耗时 30-40 秒）：

- 堆栈根交换机掉电，或者软件运行失败；
- 掉电或故障的堆栈根交换机恢复；
- 有可能成为堆栈根的新交换机加入了堆栈。

BackboneFast

BackboneFast 可以检测出非直连的骨干网核心故障。BackboneFast 是对 UplinkFast 特性的一项补充技术，后者可以对直连的接入交换机的故障作出响应。BackboneFast 优化了最大老化计时器，这个计时器会控制交换机在接口保存接收到的协议信息的总时长。当交换机从另一台交换机的指定端口那里接收到一个较差的 BPDU，那么这个 BPDU 可能表示对方已经失去了去往根的路径，BackboneFast 会尝试找到去往根的替代路径。

当交换机上的一个根端口或阻塞的接口从其指定交换机那里接收到交叉 BPDU 时，BackboneFast 就会启动。交叉 BPDU 表示有一台交换机正在将自己宣称为根桥和指定交换机。当交换机接收到较差 BPDU 时，这表示有一条与这台交换机并不直连的链路发生了故障（也就是说，指定交换机丢失了去往根交换机的连接）。根据生成树的规则，交换机会在最大老化时间内（默认为 20 秒）忽略较差 BPDU。

交换机会尝试发现是否有去往根交换机的替代路径。如果较差 BPDU 到达被阻塞的接口，那么交换机上的根端口和其他被阻塞的端口就会成为去往根交换机的替代路径。（自环端口不会成为去往根交换机的替代路径）如果较差 BPDU 到达的是根端口，那么所有阻塞的接口都会成为去往根交换机的替代路径。如果较差 BPDU 到达的是根端口，而交换机上又没有被阻塞的接口，那么交换机就会认为自己已经丢失了与根交换机的连接，因此根端口上的最大老化时间就会超时，而交换机也会根据常规的生成树规则成为根交换机。

如果交换机拥有去往根交换机的替代路径，它就会使用这些替代路径来发送根链路查询（RLQ）请求。交换机会在所有替代路径发送 RLQ 请求，来学习是否有堆栈成员有通往根交换机的替代根，并等待网络和堆栈中的其他交换机发送 RLQ 响应。交换机会在所有替代路径发送 RLQ 请求，并等待网络中的其他交换机发送 RLQ 响应。

当堆栈成员通过阻塞的接口从非堆栈成员那里接收到一条 RLQ 响应消息时，而这个响应消息的目的是另一台非堆栈交换机时，它会转发这个响应数据包，无论这个接口的生成树状态为何。

当堆栈成员通过阻塞的接口从非堆栈成员那里接收到一条 RLQ 响应消息时，而这个响应消息的目的是堆栈时，它会转发这个响应数据包，让堆栈中的其他成员都能够接收到这条消息。

如果交换机发现自己仍然有一条通向根的替代路径，那么它会让接收到较差 BPDU 的接口最大老化时间超时。如果所有去往根交换机的替代路径都显示交换机已经丢失了与根交换机之间的连接，那么交换机就会让接收到 RLQ 响应消息的接口的最大老化时间超时。如果有一两条替代路径仍然可以连接到根交换机，那么交换机就会让所有接收到交叉 BPDU 的接口成为指定端口，然后让它们从阻塞状态（如果它们之前是阻塞状态的话）经历侦听和学习状态过渡到转发状态。

这个拓扑当前没有链路出现故障，交换机 A，即根交换机与交换机 B 通过链路 L1 直连，与交换机 C 通过链路 L2 相连。交换机 C 上域交换机 B 直连的二层接口目前处于阻塞状态。

图 53：非直连链路故障之前的 BackboneFast 示例

Switch A (Root)	交换机 A （根）

Switch B	交换机 B
Switch C	交换机 C
Blocked port	阻塞端口

如果链路 1 出现了故障，交换机 C 是无法检测到这个故障的，因为交换机 C 没有直接与链路 1 相连。不过，由于交换机 B 通过 L1 直接连接到了根交换机，因此它会检测到这个故障，它会将自己选举为根，并且开始向交换机 C 发送 BPDU，声称自己是根。当交换机 C 从交换机 B 那里接收到较差 BPDU 时，交换机 C 会认为网络中有非直连链路出现了故障。此时，BackboneFast 会让交换机 C 上被阻塞的端口立刻进入侦听状态，而不需要等待接口的最大老化时间超时。接下来，BackboneFast 会将交换机 C 上的二层接口过渡到转发状态，这就提供了一条从交换机 B 去往交换机 A 的路径。根交换机选举需要消耗大约 30 秒的时候，如果用户没有修改默认转发延迟时间的 15 秒，那么这个时间就是转发延迟的 2 倍。BackboneFast 会重新配置拓扑，此时链路 L1 的故障也会被考虑在内。

图 54：非直连链路故障之后的 BackboneFast 示例

Switch A (Root)	交换机 A （根）
Switch B	交换机 B
Switch C	交换机 C
Link failure	链路故障
BackboneFast changes port through listening and learning states to forwarding state.	UBackboneFast 将端口经过侦听状态和学习状态过渡到了转发状态

如果有一台新的交换机添加到了这个共享媒介拓扑当中，BackboneFast 就不会启动，因为交换机无法识别出这些较差 BPDU 是由指定交换机（交换机 B）发送的。新的交换机会开始发送交叉 BPDU，自称是根交换机。但其他交换机会忽略这些较差 BPDU，而新交换机也会学习到交换机 B 是去往交换机 A（根交换机）的指定交换机。

图 55：向共享媒介拓扑中添加一台交换机

Switch A (Root)	交换机 A （根）
Switch B (Designated bridge)	交换机 B （指定网桥）
Switch C	交换机 C
Blocked port	阻塞端口
Added switch	新增交换机

EtherChannel 防护

用户可以使用 EtherChannel 防护来检测交换机与直连设备之间是否有 EtherChannel 的误配置。如果交换机接口配置为 EtherChannel，而另一台设备的相应接口却没有配置为 EtherChannel，就是 EtherChannel 误配置的情形。如果 EtherChannel 通道两端的参数不匹配，也属于 EtherChannel 误配置。

如果交换机检测到另一端的设备存在误配置，EtherChannel 防护就会让交换机接口进入 error-disabled 状态，同时显示错误消息。

根防护

服务提供商（SP）的二层网络可以包含很多通向交换机的连接，这些连接并不属于 SP 所有。在这样的拓扑环境中，生成树有可能会对自己进行重新配置，选择客户交换机为根交换机。用户可以在与客户网络中交换机相连的那些 SP 交换机接口上启用根防护，以避免这种情况的发生。如果生成树计算的结果是客户网络中的接口被选举为根端口，那么根防护就会这个接口进入不连续根（阻塞）状态，以防止客户交换机成为根交换机，或者客户交换机出现在去往根的路径上。

图 59：服务提供商网路中的根防护、

Customer network	客户网络
Service-provider network	服务提供商网络
Potential spanning-tree root without root guard enabled	没有启用根防护的潜在生成树根
Desired root switch	根交换机
Enable the root-guard feature on theses interfaces to prevent switches in the customer network from becoming the root switch or being the path to the root	在这些接口上启用根防护，防止客户交换机成为根交换机，或者客户交换机出现在去往根的路径上

如果 SP 网络之外的交换机成为了根交换机，那么接口就会被阻塞（进入不连续根状态），而生成树会重新选择一台新的根交换机。客户的交换机不会成为根交换机，也不会出现在去往根的路径上。

如果交换机工作在多生成树（MST）模式下，那么根防护就会强制接口成为指定端口。如果在一个内部生成树（IST）实例中的边界端口因为根防护特性而被阻塞，这个接口也会在所有 MST 实例中被阻塞。边界端口是连接一个局域网段的边界端口，这个局域网段的指定交换机要么是一台 IEEE 802.1D 交换机，要么是一台拥有不同 MST 域配置的交换机。

如果用户在一个接口上启用根防护，那么根防护就会应用于这个接口所属的所有 VLAN。VLAN 可以分组并映射到一个 MST 实例当中。

注意： 根防护特性使用不当可能会导致网络丢失连接。

环路防护

用户可以使用环路防护来防止替代端口或根端口因出现单向链路故障而成为指定端口。在整个交换网络中都配置这个特性是最有效的。环路防护可以防止替代端口和根端口成为指定端口，而生成树也不会在根端口或替代端口发送 BPDU。

当交换机工作在 PVST+或快速 PVST+模式下时，环路防护会防止替代端口和根端口成为指定端口，而生成树也不会在根端口或替代端口发送 BPDU。

如果交换机工作在 MST 模式下，那么只有当接口在所有 MST 实例中都被环路防护特性阻塞时，这个非边界端口才不会发送 BPDU。在边界端口上，环路防护会将接口在所有 MST 实例中进行阻塞。

如何配置可选生成树特性

启用 PortFast（CLI）

启用了 PortFast 特性的接口会直接进入生成树转发状态，而不需要等待标准的转发时间延迟。

如果启用语音 VLAN 特性，那么 PortFast 特性也会自动启用。但在禁用语音 VLAN 时，PortFast 并不会自动被禁用。

如果交换机运行的是 PVST+、快速 PVST+或 MSTP，那么用户可以启用这个特性。

注意： 用户只可以在连接一个终端工作站的 access 端口或 trunk 端口上使用 PortFast。在连接到交换机或集线器的接口上启用这个特性会妨碍生成树检测和禁用网络中的环路，而这会导致网络风暴和地址学习问题。

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. spanning-tree portfast [**trunk**]

5. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的接口，进入接口配置模式
步骤 4	**spanning-tree portfast** [**trunk**] 示例： `Device(config-if)#` **spanning-tree portfast trunk**	在连接到一个工作站或服务器的 access 端口上启用 PortFast。用户可以使用 **trunk** 这个关键字，在 trunk 端口上启用 PortFast。 **注释：**要在 trunk 端口上启用 PortFast，用户必须配置接口配置命令 **spanning-tree portfast trunk**。而在 trunk 端口上输入命令 **spanning-tree portfast** 则不会生效。 用户务必确保 trunk 端口与工作站和服务器之间没

		有网络环路，然后才能在 trunk 端口上启用 PortFast。 在默认情况下，PortFast 在所有接口上都是禁用的
步骤 5	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

接下来做什么？

用户可以使用全局配置命令 **spanning-tree portfast default** 来在所有非中继端口上全局启用 PortFast 特性。

启用 BPDU 防护（CLI）

如果交换机运行的是 PVST+、快速 PVST+或 MSTP，那么用户就可以启用 BPDU 防护特性。

注意： 用户只可以在连接终端工作站的端口上配置 PortFast 边缘特性；否则，网络就有可能会因为意料之外的拓扑环路而产生数据环路，进而打断交换机和网络的操作。

这个流程是可选的。

总步骤

1. **enable**
2. **configure terminal**
3. **spanning-tree portfast edge bpduguard default**
4. **interface** *interface-id*
5. **spanning-tree portfast edge**
6. **end**

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree portfast edge bpduguard default** 示例： `Device(config)#` **spanning-tree portfast edge bpduguard default**	在全局启用 BPDU 防护。 在默认情况下，BPDU 防护是禁用的
步骤 4	**interface** *interface-id*	选择与终端工作站相连的接口，并进入接口配置模式

	示例： `Device(config)#` **interface gigabitethernet1/0/2**	
步骤 5	**spanning-tree portfast edge** 示例： `Device(config-if)#` **spanning-tree portfast edge**	启用 PortFast 边缘特性
步骤 6	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

接下来做什么？

要防止端口关闭，用户可以使用全局配置命令 **errdisable detect cause bpduguard shutdown vlan** 来单独关闭发生了违背事件的端口所在的 VLAN。

用户也可以使用接口配置命令 **spanning-tree bpduguard enable** 在没有启用 PortFast 边缘特性的接口上启用 BPDU 防护。当这个端口接收到 BPDU 时，它就会进入 error-disabled 状态。

启用 BDPU 过滤（CLI）

用户也可以使用接口配置命令 **spanning-tree bpdufilter enable** 在没有启用 PortFast 边缘特性的接口上启用 BPDU 过滤。这条命令可以防止接口发送或接收 BPDU。

注意： 在接口启用 BPDU 过滤相当于在接口上禁用生成树，因此有可能会导致生成树环路。如果交换机运行的是 PVST+、快速 PVST+或 MSTP，那么用户就可以启用 BPDU 过滤特性。

注意： 用户只可以在连接终端工作站的端口上配置 PortFast 边缘特性；否则，网络就有可能会因为意料之外的拓扑环路而产生数据环路，进而打断交换机和网络的操作。

这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. spanning-tree portfast edge bpdufilter default

4. interface *interface-id*

5. spanning-tree portfast edge

6. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例：	进入全局配置模式

	Device# **configure terminal**	
步骤 3	**spanning-tree portfast edge bpdufilter default** 示例： Device(config)# **spanning-tree portfast edge bpdufilter default**	在全局启用 BPDU 过滤。 在默认情况下，BPDU 过滤是禁用的
步骤 4	**interface** *interface-id* 示例： Device(config)# **interface gigabitethernet1/0/2**	选择与终端工作站相连的接口，并进入接口配置模式
步骤 5	**spanning-tree portfast edge** 示例： Device(config-if)# **spanning-tree portfast edge**	在这个接口上启用 PortFast 边缘特性
步骤 6	**end** 示例： Device(config)# end	返回特权 EXEC 模式

为使用冗余链路启用 UplinkFast（CLI）

注释： 在启用 UplinkFast 时，它会影响交换机或交换机堆栈上的所有 VLAN。用户不能单独在一个 VLAN 上配置 UplinkFast

用户可以针对快速 PVST+或 MSTP 配置 UplinkFast 或交叉堆栈 UplinkFast（CSUF）特性，但是在用户将生成树协议的模式修改为 PVST+之前，这个特性还是会处于禁用（未生效）的状态。这个流程是可选的。用户可以按照下面的步骤来启用 UplinkFast 和 CSUF。

在开始前

在配置了交换机优先级的 VLAN 上是无法启用 UplinkFast 的。要给一个配置了交换机优先级的 VLAN 启用 UplinkFast，要首先使用全局配置命令 **no spanning-tree vlan** *vlan-id* **priority** 将这个 VLAN 上的交换机优先级恢复为默认值。

总步骤

1. enable

2. configure terminal

3. spanning-tree uplinkfast [**max-update-rate** *pkts-per-second*]

4. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable**	进入特权 EXEC 模式。在提示时输入密码

	示例： `Device> enable`	
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**spanning-tree uplinkfast** [**max-update-rate** *pkts-per-second*] 示例： `Device(config)# spanning-tree uplinkfast max-update-rate 200`	启用 UplinkFast。 （可选）*pkts-per-second* 部分的取值范围为 0 到 32000 个数据包每秒；默认值为 150。 如果将速率设置为 0，那么工作站学习的数据帧就不会被创建出来，在连接丢失之后，生成树拓扑会收敛得更慢。 在输入这条命令时，CSUF 也会在所有非堆栈端口上启用
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

在启用 UplinkFast 时，所有 VLAN 的交换机优先级都会被设置为 49152。如果用户将路径开销设置为一个小于 3000 的值，同时启用 UplinkFast 或者 UplinkFast 已经启用，那么所有接口和 VLAN trunk 的路径开销都会增加为 3000（如果用户已经将路径开销增加为 3000 或者更高的值，那么路径开销就不会变化）。变更交换机优先级和路径开销会降低一台交换机成为根交换机的几率。

如果禁用 UplinkFast，那么所有 VLAN 的交换机优先级和所有接口的路径开销都会被设置为默认值（如果用户没有将它们修改为默认值的话）。

在用户使用这些方法来启用 UplinkFast 特性时，CSUF 也会在所有非堆栈端口上启用。

禁用 UplinkFast（CLI）

这个流程是可选的。

用户可以按照下面的步骤来禁用 UplinkFast 和交叉堆栈 UplinkFast（CSUF）。

在开始前

用户必须先启用 UplinkFast。

总步骤

1. enable

2. configure terminal

3. no spanning-tree uplinkfast

4. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例：	进入特权 EXEC 模式。在提示时输入密码

	Device> **enable**	
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**no spanning-tree uplinkfast** 示例： Device(config)# **no spanning-tree uplinkfast**	在交换机和所有其 VLAN 上禁用 UplinkFast 与 CSUF
步骤 4	**end** 示例： Device(config)# end	返回特权 EXEC 模式

如果禁用了 UplinkFast，那么所有 VLAN 的交换机优先级和所有接口的路径开销都会被设置为默认值（如果用户没有将它们修改为默认值的话）。
在用户使用这些方法来禁用 UplinkFast 特性时，CSUF 也会在所有非堆栈端口上禁用。

启用 BackboneFast（CLI）

用户可以启用 BackboneFast 来检测非直连链路的故障，并且让生成树重新配置更快启动。用户可以针对快速 PVST+或 MSTP 配置 BackboneFast 特性，但是在用户将生成树协议的模式修改为 PVST+之前，这个特性还是会处于禁用（未生效）的状态。
这个流程是可选的。用户可以按照下面的步骤来启用 BackboneFast。

在开始前

如果使用 BackboneFast，用户必须在网络中的所有交换机上启用这项特性。令牌环 VLAN 上是不支持 BackboneFast 的。这项特性可以在包含第三方交换机的网络环境中使用。

总步骤

1. enable
2. configure terminal
3. spanning-tree backbonefast
4. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree backbonefast**	启用 BackboneFast

	示例： `Device(config)# spanning-tree backbonefast`	
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

启用 EtherChannel 防护（CLI）

如果设备正在运行 PVST+、快速 PVST+或 MSTP，那么用户可以启用 EtherChannel 防护来检测 EtherChannel 的误配置。
这个流程是可选的。
用户可以按照下面的步骤在设备上启用 EtherChannel 防护。

总步骤

1. enable

2. configure terminal

3. spanning-tree etherchannel guard misconfig

4. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**spanning-tree etherchannel guard misconfig** 示例： `Device(config)# spanning-tree etherchannel guard misconfig`	启用 EtherChannel 防护
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

接下来做什么？

用户可以使用特权 EXEC 命令 **show interfaces status err-disabled** 来查看哪些设备端口因

EtherChannel 误配置而被禁用。在远程设备上，用户可以输入特权 EXEC 命令 **show etherchannel summary** 来验证 EtherChannel 的配置。
在配置验证完毕之后，用户可以在之前误配置的 port-channel 接口上输入接口配置命令 **shutdown** 和 **no shutdown**。

启用根防护（CLI）

在接口上启用的根防护会应用于这个接口属于的每一个 VLAN。用户不要在使用 UplinkFast 特性的接口上启用根防护。通过 UplinkFast，当根端口出现故障时，（阻塞状态下的）备份接口会替代根端口。但如果用户同时也启用了根防护，那么所有 UplinkFast 特性使用的备份接口就会进入不连续根状态（会被阻塞），设备不会让这些端口过渡到转发状态。
注释： 用户不能同时启用根防护和环路防护。
如果设备正在运行 PVST+、快速 PVST+或 MSTP，那么用户可以启用这项特性。
这个流程是可选的。
用户可以按照下面的步骤在设备上启用根防护。
总步骤
1. enable
2. configure terminal
3. interface *interface-id*
4. spanning-tree guard root
5. end
具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet1/0/2**	选择要配置的接口，并进入接口配置模式
步骤 4	**spanning-tree guard root** 示例： `Device(config-if)#` **spanning-tree guard root**	在接口上启用根防护。 在默认情况下，根防护是在所有接口上禁用的
步骤 5	**end**	返回特权 EXEC 模式

	示例： `Device(config)# end`	

启用环路防护（CLI）

用户可以使用环路防护来防止替代端口或根端口由于单向链路失效而成为指定端口。在整个交换网络中都配置这个特性是最有效的。环路防护只会在那些被生成树视为点到点的接口上生效。

注释： 用户不能同时启用环路防护和根防护。

如果设备正在运行 PVST+、快速 PVST+或 MSTP，那么用户可以启用这项特性。

这个流程是可选的。用户可以按照下面的步骤在设备上启用环路防护。

总步骤

1. 输入下面命令之一：

• show spanning-tree active

• show spanning-tree mst

2. configure terminal

3. spanning-tree loopguard default

4. end

具体步骤

	命令或操作	目的
步骤 1	**输入下面命令之一：** **• show spanning-tree active** **• show spanning-tree mst** 示例： `Device#` **show spanning-tree active** 或 `Device#` **show spanning-tree mst**	验证哪些接口是替代端口，哪些是根端口
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**spanning-tree loopguard default** 示例： `Device(config)#` **spanning-tree loopguard default**	启用环路防护。 在默认情况下，环路防护是禁用的
步骤 4	**end** 示例：	返回特权 EXEC 模式

	Device(config)# end	

监控生成树的状态

表 62：监控生成树状态的命令

命令	目的
show spanning-tree active	仅显示活动接口的生成树信息
show spanning-tree detail	显示具体的接口信息
showspanning-tree interface *interface-id*	显示特定接口的生成树信息
show spanning-tree mst interface *interface-id*	显示特定接口的生成树信息
show spanning-tree summary [**totals**]	显示接口状态的汇总信息，或者显示 STP 状态部分的总行
show spanning-tree mst interface *interface-id* **portfast edge**	显示特定接口的生成树 portfast 信息

其他可选生成树特性的参考资料

相关文档

相关主题	文档名
生成树协议的命令	《LAN 交换命令参考手册，Inspur INOSXE3SE 版（Inspur 6650 交换机）》

错误消息解码器

描述	链接
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

标准与 RFC

标准/RFC	标题
无	——

技术助手

描述	链接
Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。 用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	http://www.icntnetworks.com

可选生成树特性的特性历史

版本	修改
Inspur INOS 12.2	引入该特性

配置 EtherChannel

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

EtherChannel 的限制条件

下面是 EtherChannel 的限制条件：

- EtherChannel 中的所有端口都必须分配给同一个 VLAN，或者配置为 trunk 端口；
- 如果运行 LAN Base 许可证的特性集，那么设备不支持使用三层的 EtherChannel；
- 用户不能混合使用 Inspur 3850 交换机和 Inspur 6650 交换机来建立交换机堆栈。

EtherChannel 概述

EtherChannel 可以在交换机、路由器和服务器之间提供拥有容错功能的高速链路。用户可以使用 EtherChannel 增加配线柜与数据中心之间的带宽，用户也可以在网络中最有可能出现瓶颈的位置部署这项技术。EtherChannel 可以通过各个链路来对负载进行重分布，以便当一些链路出现故障时，能够自动恢复连接。如果链路出现了故障，EtherChannel 会将故障链路发来的流量重定向给信道中的剩余链路，而不会出现流量中断。
一条 EtherChannel 是由多条以太网链路捆绑成的一条逻辑链路。

图 60：典型的 EtherChannel 配置

Catalyst switch	Catalyst 交换机
10/100 Switched links	10/100 交换链路
10/100 Switched links	10/100 交换链路
Workstations	工作站
Workstations	工作站

EtherChannel 可以在交换机与另一台交换机或主机之间提供最大 8Gb/s（Gigabit EtherChannel）或 80Gb/s（10-Gigabit EtherChannel）的全双工带宽。

每条 EtherChannel 都可以由最多 8 条可兼容的 Ethernet 端口组成。

EtherChannel 的数量最多为 128 条。

LAN Base 特性集支持最多 24 条 EtherChannel。

每条 EtherChannel 中的所有端口都必须要么配置为二层端口，要么配置为三层端口。EtherChannel 三层端口都是由路由端口组成的。路由端口即为使用接口配置命令 **no switchport** 设置为三层模式的物理端口。要想了解具体信息，可以参考配置接口特征一章。

Ethernet 模式

用户可以将 EtherChannel 配置为下列模式之一：端口汇聚协议（PAgP）、链路汇聚控制协议（LACP）或 On。用户要将 EtherChannel 两端配置为同一种模式：

若将 EtherChannel 一端配置为 PAgP 或 LACAP 模式，系统就会与信道另一端进行协商，以判断哪些端口应该处于活动（active）状态。如果远端端口不能协商 EtherChannel，那么本地端口就会进入独立状态，并且继续向一条链路一样承载数据流量。端口的配置不会更改，但是这个端口不会参与 EtherChannel；

若将 EtherChannel 配置为 **on** 模式，那就不会发生协商。交换机会强制所有兼容端口在 EtherChannel 中处于活动状态。此时，信道的另一端也必须配置为 **on** 模式，否则就会出现丢包。

设备上的 EtherChannel

用户可以在一台设备、堆栈中的一台设备或者堆栈中的多台设备（称为交叉堆栈 EtherChannel）上创建一条 EtherChannel。

图 61：单交换机上的 EtherChannel

Swtich stack	交换机堆栈
Switch 1	交换机 1
Switch 2	交换机 2
Switch 3	交换机 3
Switch A	交换机 A
StackWise Plus port connections	StackWise Plus 端口连接

图 62：交叉堆栈 EtherChannel

Swtich stack	交换机堆栈
Switch 1	交换机 1
Switch 2	交换机 2
Switch 3	交换机 3
Switch A	交换机 A

StackWise Plus port connections	StackWise Plus 端口连接

EtherChannel 链路故障切换

如果 EtherChannel 中的链路出现了故障，之前通过故障链路传输的流量就会迁移到 EtherChannel 中剩余的链路进行传输。如果交换机上启用了 trap，那么在出现故障时，设备就会发送一条 trap 来标识交换机、EtherChannel 和故障链路。在 EtherChannel 中通过一条链路入站的广播和组播数据包不能通过 EtherChannel 中另一条链路返回。

Channel Group 与 Port-Channel 接口

EtherChannel 包含一个 channel group 和一个 port-channel 接口。在 channel group 中，物理端口会绑定到 port-channel 接口。应用到 port-channel 接口上的配置变更也会应用到 channel group 中的所有物理端口。

命令 **channel-group** 会将物理端口和 port-channel 接口进行绑定。每个 EtherChannel 都有一个编号为 1 到 128 之间的 port-channel 逻辑接口。这个 port-channel 接口编号对应的是接口配置命令 **channel-group** 设置的编号。

图 63：物理端口、Channel Group 和 Port-Channel 接口之间的关系

Logical port-channel	逻辑 port-channel
Channel-group binding	Channel-group 绑定关系
Physical ports	物理端口

- 对于二层端口，用户可以使用接口配置命令 **channel-group** 来动态创建 port-channel 接口；

  用户也可以使用全局配置命令 **interface port-channel** *port-channel-number* 来手动创建 port-channel 接口，但接下来用户必须使用命令 **channel-group** *channel-group-number* 来给物理端口绑定逻辑接口。其中，*channel-group-number* 和 *port-channel-number* 可以使用同一个数值，也可以使用一个新的数。如果使用新的数值，那么命令 **channel-group** 就会动态创建出新的 port channel。
- 对于三层端口，用户应该使用全局配置命令 **interface port-channel** 加上接口配置命令 **no switchport** 来创建逻辑接口。接下来，用户可以使用接口配置命令 **channel-group** 来手动将一个接口分配给 EtherChannel。
- 对于三层端口，用户应该使用接口配置命令 **no switchport** 来将接口配置为三层接口。接下来，用户可以使用接口配置命令 **channel-group** 来手动将一个接口分配给 EtherChannel。

端口汇聚协议

端口汇聚协议（PAgP）是一个 Inspur 私有协议，这个协议只能在 Inspur 设备上，和获准支持 PAgP 的厂商设备上运行。PAgP 支持在以太网端口之间交换 PAgP 数据包，以动态创建 EtherChannel。

通过 PAgP，设备或设备堆栈可以学习到能够支持 PAgP 的对端身份，以及每个端口的功能。接下来，它就可以动态将（堆栈中一台设备上）配置类似的端口分组为一条逻辑链路（信道或汇聚端口）。配置类似的端口可以根据硬件、管理和端口参数的限制进行分类。例如，PAgP

会将拥有相同速率、双工模式、native VLAN、VLAN 范围和中继状态和类型的端口进行分组。在将链路分组进一个 EtherChannel 之后，PAgP 就会将这个组作为一个设备端口添加到生成树当中。

PAgP 模式

PAgP 模式指定了一个端口是否可以发送 PAgP 数据包，哪个端口发起 PAgP 协商，或者端口是否仅对接收到的 PAgP 数据包进行响应。

表 63：EtherChannel PAgP 模式

模式	描述
auto	让一个端口进入被动协商的模式。在这种模式下，端口只会响应它接收到的 PAgP 数据包，但不会发起 PAgP 数据包协商。这种设置可以将 PAgP 数据包的传输降至最低
desirable	让一个端口进入主动协商的模式。在这种模式下，端口会通过发送 PAgP 数据包来发起与其他端口的协商。当 EtherChannel 成员来自交换机堆栈中的不同交换机时，可以使用这种模式

交换机端口只会与配置在 **auto** 或 **desirable** 模式下的端口交换 PAgP 数据包。配置在 **on** 模式下的端口不会交换 PAgP 数据包。

工作在 **auto** 和 **desirable** 模式下的端口会基于诸如端口速率这类的标准，来与对端进行协商，以建立 EtherChannel。对于二层 EtherChannel 来说，双方则会基于 trunk 状态和 VLAN 编号进行协商。

当两边的端口处于不同的 PAgP 模式下时，只要这些模式相互兼容，那么这两边的端口还是可以建立 EtherChannel。例如：

- 工作在 **desirable** 模式下的端口，可以与另一端工作在 **desirable** 或 **auto** 模式下的端口建立 EtherChannel；
- 工作在 **auto** 模式下的端口，可以与另一端工作在 **desirable** 模式下的端口建立 EtherChannel；

工作在 **auto** 模式下的端口，不能与另一端工作在 **auto** 模式下的端口建立 EtherChannel，因为这两个端口都不会发起 PAgP 协商。

静默模式

如果交换机连接到了一个支持 PAgP 的设备，用户可以使用关键字 **non-silent** 配置交换机端口，让其执行非静默操作。如果用户没有在配置 **auto** 或 **desirable** 模式时添加关键字 **non-silent**，交换机也会默认支持静默模式。

如果交换机连接的是不支持 PAgP 的设备，那么在配置了静默模式之后，交换机就几乎不会通过这个端口发送数据包。可以充当静默对端的设备包括文件服务器，或者不会生成流量的数据包分析器等等。在本例中，在连接静默设备的物理端口上运行 PAgP 会防止这个交换机端口进入操作状态。不过，静默设置可以让 PAgP 正常工作，可以让将这个端口划入 channel group，并且使用这个端口传输流量。

PAgP 学习方式与优先级

网络设备可以归类为 PAgP 物理学习设备，或者汇聚端口学习设备。如果一台设备通过物理端口学习地址，并且直接依据这些内容来传输流量，那么这台设备就是物理学习设备。如果一台设备通过汇聚（逻辑）端口学习地址，那么这台设备就是汇聚端口学习设备。在链路两端，学习方法必须采取相同的配置。

当设备和对端都是汇聚端口学习设备，那么它们就会通过这条逻辑 port-channel 来学习地址。设备会使用 EtherChannel 中的端口来向源发送数据包。通过汇聚端口进行学习的话，数据包具体是通过哪个物理端口学习到的就并不重要了。

当对端设备是物理学习设备，而本地设备则是汇聚端口学习设备，那么 PAgP 就无法执行自动检测。因此，用户必须在本地设备上手动设置学习方法，让设备通过物理端口学习地址。用户还必须将负载分发方式设置为基于源的分发，让任何给定的源 MAC 地址都在同一个物理端口上进行发送。

用户也可以在组中配置一个端口，让它执行所有的传输，然后使用其他端口进行热备份。如果选择的端口检测不到硬件信号，那么组中未使用的端口在几秒之内就可能会切换到操作状态。用户可以使用接口配置命令 **pagp port-priority** 来修改端口的优先级，让所选接口执行所有的数据传输。优先级越高，选择这个端口的可能性就越高。

注释： 即使用户通过 CLI 输入了命令 **physical-port**，设备还是支持仅通过汇聚端口来学习地址。命令 **pagp learn-method** 和命令 **pagp port-priority** 对于设备的硬件没有效果，但用户需要输入这些命令，来让设备与那些只能通过物理端口学习地址的设备（如 Inspur 1900 交换机）进行 PAgP 互操作。

当设备在链路上的对端是一台物理学习设备，我们推荐用户使用接口配置命令 **pagp learn-method physical-port** 来将这台设备配置为一个物理端口学习设备。用户可以使用全局配置命令 **port-channel load-balance src-mac** 来设置基于源 MAC 地址的负载分发方式。接下来，设备就会使用 EtherChannel 中学习到这个源地址的端口来发送数据包。用户只应在这种情况下使用命令 **pagp learn-method**。

PAgP 与其他特性的互动

动态中继协议（DTP）和 Inspur 发现协议（CDP）会通过 EtherChannel 中的物理端口来发送和接收数据包。Trunk 端口会在编号最低的 VLAN 中发送和接收协议数据单元（PDU）。

在二层 EtherChannel 中，信道中第一个启用的端口会将自己的 MAC 地址提供给 EtherChannel。如果用户将这个端口从接口束中移除，那么接口束中剩余的接口就会将自己的 MAC 地址提供给 EtherChannel。对于三层 EtherChannel，一旦用户（通过全局配置命令 **interface port-channel**）将接口创建出来，活动设备就会分配 MAC 地址。

PAgP 只会从启动，且启用了（auto 或 desirable 模式的）PAgP 的端口上发送和接收 PAgP PDU。

链路汇聚控制协议

LACP 定义在 IEEE 802.3 当中，它可以让 Inspur 设备管理（符合 IEEE 802.3ad 协议的）设备之间的以太网信道。LACP 可以交换以太网端口之间的 LACP 数据包，来自动创建 EtherChannel。通过 LACP，设备或设备堆栈可以学习到能够支持 LACP 的对端身份，以及每个端口的功能。接下来，它就可以动态将配置类似的端口分组为一条逻辑链路（信道或汇聚端口）。配置类似的端口可以根据硬件、管理和端口参数的限制进行分类。例如，LACP 会将拥有相同速率、双工模式、native VLAN、VLAN 范围和中继状态和类型的端口进行分组。在将链路分组进一个 EtherChannel 之后，LACP 就会将这个组作为一个设备端口添加到生成树当中。

在 port channel 中，端口独立模式操作发生了变化。在默认情况下，通过 CSCtn96950，设备会启用独立模式。当设备无法从 LACP 对等体那里接收到响应消息时，port channel 中的端口就会进入暂缓（suspended）状态。

LACP 模式

LACP 模式指定了一个端口是可以发送 LACP 数据包，还是只能接收 LACP 数据包。

表 64：EtherChannel LACP 模式

模式	描述
active	让一个端口进入主动协商的模式。在这种模式下，端口会通过发送 LACP 数据

	包来发起与其他端口的协商。
passive	让一个端口进入被动协商的模式。在这种模式下，端口只会响应它接收到的 LACP 数据包，但不会发起 LACP 数据包协商。这种设置可以将 LACP 数据包的传输降至最低

工作在**active**和**passive**模式下的端口会基于诸如端口速率这类的标准，来与对端进行协商，以建立 EtherChannel。对于二层 EtherChannel 来说，双方则会基于 trunk 状态和 VLAN 编号进行协商。

当两边的端口处于不同的 LACP 模式下时，只要这些模式相互兼容，那么这两边的端口还是可以建立 EtherChannel。例如：

- 工作在 **active** 模式下的端口，可以与另一端工作在 **active** 或 **passive** 模式下的端口建立 EtherChannel；
- 工作在 **passive** 模式下的端口，不能与另一端同样工作在 **passive** 模式下的端口建立 EtherChannel，因为这两个端口都不会发起 PAgP 协商。

LACP 与链路冗余

用户可以通过 LACP port-channel min-link 和 LACP max-bundle 特性，来进一步改善 LACP port-channel 的操作、带宽可用性和链路冗余。

LACP port-channel min-link（最小链路）特性：

- 配置必须启用并绑定到 LACP port channel 的最少端口数量；
- 防止低带宽的 LACP 端口变为活动状态；
- 如果活动成员端口的数量太少，达不到所需的最小带宽，则让 LACP port channel 成为不活动状态。

LACP max-bundle（最大绑定）特性：

- 定义 LACP port channel 中绑定的端口数量上限；
- 支持包含更少绑定端口的热备份端口。例如，在包含 5 个端口的 LACP port channel 当中，用户可以将 max-bundle 设置为 3，将剩下两个 2 端口指定为热备份端口。

LACP 与其他特性的互动

DTP 和 CDP 会通过 EtherChannel 中的物理端口来发送和接收数据包。Trunk 端口会在编号最低的 VLAN 中发送和接收协议数据单元（PDU）。

在二层 EtherChannel 中，信道中第一个启用的端口会将自己的 MAC 地址提供给 EtherChannel。如果用户将这个端口从接口束中移除，那么接口束中剩余的接口就会将自己的 MAC 地址提供给 EtherChannel。对于三层 EtherChannel，一旦用户（通过全局配置命令 **interface port-channel**）将接口创建出来，活动设备就会分配 MAC 地址。

LACP 只会从启动，且启用了（auto 或 desirable 模式的）LACP 的端口上发送和接收 LACP PDU。

EtherChannel On 模式

EtherChannel 的 **on** 模式可以用来手动配置 EtherChannel。这种 **on** 模式可以强制一个端口不经过协商直接加入 EtherChannel。如果远端设备不支持 PAgP 或 LACP，那么 **on** 模式就会相当实用。在 **on** 模式下，只有当链路两端的设备上都配置了 **on** 模式时，双方才会建立一条可用的 EtherChannel。

对于配置在同一个 channel group 的 **on** 模式下的端口，它们必须拥有匹配的端口特征，包括速率和双工模式。不兼容的端口会进入暂缓（suspended）状态，即使用户将它们配置在 **on** 模式下也是如此。

注意： 在使用 **on** 模式时务虚小心。这是手动配置，EtherChannel 两端的端口必须拥有相

同的配置。如果组配置有误，网络中就会出现丢包或生成树环路。

负载分担和转发方式

EtherChannel 会通过各个链路来分担流量，它会将数据帧中地址里面的二元组减少到一个值，来选择信道中的一条链路使用。用户可以从几种不同的负载分担模式中选择一种，包括基于 MAC 地址的负载分发、基于 IP 地址的负载分发、基于源地址的负载分发、基于目的地址的负载分发、或基于源和目的地址的负载分发。选择的模式会应用于设备上配置的所有 EtherChannel。

注释： 三层等价多路径（ECMP）负载分担可以基于源 IP 地址、目的 IP 地址、源端口、目的端口和四层协议执行负载分担。分片的数据包可以在两条不同的链路上，基于这些参数运行算法进行处理。其中一项参数发生变化都会影响负载分担。

用户会使用全局配置命令 **port-channel load-balance** 和 **port-channel load-balance extended** 来配置负载分担和转发方式。

MAC 地址转发

如果采用源 MAC 地址转发的方式，那么当数据包被转发到 EtherChannel 时，它们会基于入站数据包的源 MAC 地址在信道的各个端口中进行分发。因此，对于实现负载分担，来自不同主机的数据包会使用信道中的不同端口，但是从同一个主机发来的数据包就会使用信道中的同一个端口进行转发。

如果采用目的 MAC 地址转发的方式，那么当数据包被转发到 EtherChannel 时，它们会基于入站数据包的目的主机的 MAC 地址在信道的各个端口中进行分发。因此，去往同一个目的的数据包会使用信道中的同一个端口，但去往不同目的的数据包就会通过信道中的不同端口进行转发。

如果采用源和目的 MAC 地址转发的方式，那么当数据包被转发到 EtherChannel 时，它们会基于入站数据包的源和目的主机 MAC 地址在信道的各个端口中进行分发。通过这种方法，设备会结合源 MAC 地址和目的 MAC 地址的方式来提供负载分发。如果用户不清楚在某台设备上更适合使用源 MAC 地址转发还是目的 MAC 地址转发，就可以使用这种转发方式。通过源和目的 MAC 地址转发，从主机 A 发送给主机 B、从主机 A 发送给主机 C 和从主机 C 发送给主机 B 的数据包，会使用信道中的不同端口进行转发。

IP 地址转发

如果采用基于源 IP 地址转发的方式，那么数据包就会基于入站数据包的源 IP 地址在信道的各个端口中进行分发。因此，对于实现负载分担，来自不同 IP 地址的数据包会使用信道中的不同端口，但是从同一个 IP 地址发来的数据包就会使用信道中的同一个端口进行转发。

如果采用基于目的 IP 地址转发的方式，那么数据包就会基于入站数据包的目的 IP 地址在信道的各个端口中进行分发。因此，从同一个源 IP 地址发往不同目的 IP 地址的数据包会使用信道中的不同端口，而从不同的源 IP 地址发往同一个目的 IP 地址的数据包就会通过信道中的同一个端口进行转发。

如果采用源和目的 IP 地址转发的方式，那么数据包会基于入站数据包的源和目的主机 IP 地址在信道的各个端口中进行分发。通过这种方法，设备会结合源 IP 地址和目的 IP 地址的方式来提供负载分发。如果用户不清楚在某台设备上更适合使用源 IP 地址转发还是目的 IP 地址转发，就可以使用这种转发方式。通过源和目的 IP 地址转发，从 IP 地址 A 发送给 IP 地址 B、从 IP 地址 A 发送给 IP 地址 C 和从 IP 地址 C 发送给 IP 地址 B 的数据包，会使用信道中的不同端口进行转发。

负载分担的优势

不同的负载分担方式拥有不同的优势。用户应该根据设备在网络中的位置，和需要进行负载分担的流量类型来选择负载分担的方式。

在下图中，EtherChannel 有 4 台工作站在与一台路由器进行通信。由于路由器是单 MAC 地址设备，因此在 EtherChannel 设备上实施基于源的转发，可以确保设备会使用所有通向路由器的可用带宽。这台路由器上配置了基于目的的转发，因为大量工作站可以确保流量可以通过路由器的 EtherChannel 进行平均地分发。

图 64：负载分发的转发方式

Switch with source-based forwarding enabled	启用了基于源转发的交换机
Cisco router with destination-based forwarding enabled	启用了基于目的转发的 Cisco 路由器

用户在配置时，应该采用能够尽可能利用网络资源的方式。例如，如果信道中的流量都是去往同一个 MAC 地址，那么使用目的 MAC 地址就会让设备使用信道中的相同链路，因此使用源 IP 地址获得的负载分担效果更好。

EtherChannel 与设备堆栈

如果堆栈成员中有参与 EtherChannel 的端口出现了故障，或者离开了堆栈，那么活动设备就会将故障的堆栈成员设备端口从 EtherChannel 中移除。而 EtherChannel 中剩余的端口（如果还有剩余端口的话）会继续提供连通性。

当一台设备被添加到当前的堆栈中时，这台新的设备会从活动设备那里接收到运行配置，并且使用 EtherChannel 相关的堆栈配置来更新自己的配置。堆栈成员也会接收到操作信息（处于 up 状态的端口列表和信道的成员）。

当两个相互配置了 EtherChannel 连接的堆栈进行融合时，就会出现自环。生成树会检测到这种情况，并且执行相应的操作。在获胜的设备堆栈上，所有 PAgP 或 LACP 配置都不受到影响，而在没有获胜的设备堆栈上，所有 PAgP 或 LACP 配置则会在堆栈重启后全部被删除。

设备堆栈与 PAgP

在使用 PAgP 时，如果主用设备出现了故障或者离开了堆栈，那么备份设备就会成为新的主用设备。除非 EtherChannel 带宽发生了变化，否则生成树就不会重新收敛。新的主用设备会同步堆栈成员的配置。在主用设备变更之后，PAgP 的配置不会受到影响，除非 EtherChannel 有端口位于老的主用设备上。

设备堆栈与 LACP

在使用 LACP 时，系统 ID 会使用主用设备的堆栈 MAC 地址。当主用设备出现了故障或者离开了堆栈，而备份设备成为新的主用设备时，LACP 系统 ID 并不会变化。在默认情况下，在主用设备变更之后，LACP 的配置不会受到影响。

默认的 EtherChannel 配置

下表描述了默认的 EtherChannel 配置。

表 65：默认的 EtherChannel 配置

特性	默认设置
Channel group	未分配
Port-channel 逻辑接口	未定义

PAgP 模式	无默认设置
PAgP 学习方式	所有端口上皆为汇聚端口学习
PAgP 优先级	所有端口皆为 128
LACP 模式	无默认设置
LACP 学习方式	所有端口上皆为汇聚端口学习
LACP 端口优先级	所有端口皆为 32768
LACP 系统优先级	32768
LACP 系统 ID	LACP 系统优先级和设备或堆栈的 MAC 地址
负载分担	设备上的负载分发会基于入站数据包的源 MAC 地址来执行

EtherChannel 配置指南

如果配置正确的话，有些 EtherChannel 端口会自动禁用或者避免网络环路及其他问题。用户可以按照下面的指导方针来避免出现配置问题：

- 不要尝试在设备或设备堆栈上配置超过 128 个 EtherChannel；
- 配置 PAgP EtherChannel 时，最多添加 8 个同一种类型的以太网端口；
- 配置 LACP EtherChannel 时，最多添加 16 个同一种类型的以太网端口。其中最多可有 8 个端口处于活动状态，另外 8 个端口则处于备用模式；
- 将 EtherChannel 中的所有端口配置为相同的速率和双工模式；
- 启用 EtherChannel 中的所有端口。如果 EtherChannel 中有端口因为接口配置命令 shutdown 而被禁用，那么这会被视为是这条链路出现了故障，其流量会通过 EtherChannel 中剩余的端口进行传输；
- 在第一次创建出一个组时，所有端口都会按照针对添加到组中第一个端口的参数进行设置。如果修改其中一项参数的配置，那么也必须对组中所有端口的配置进行修改：
  - 允许的 VLAN 列表
  - 每个 VLAN 的生成树路径开销
  - 每个 VLAN 的生成树端口优先级
  - 生成树 PortFast 设置
- 不要将一个端口配置为多个 EtherChannel 组的成员端口；
- 不要将一个 EtherChannel 同时配置在 PAgP 和 LACP 模式下。运行 PAgP 和 LACP 的 EtherChannel 组可以在同一台设备上共存，也可以在堆栈中的不同设备上共存。其中每个 EtherChannel 组可以或运行 PAgP，或运行 LACP，但这两种模式是不能实现互操作的；
- 不要将一个安全端口配置为 EtherChannel 的一部分，反之亦然；
- 不要将一个 EtherChannel 中的活动成员端口，或者以后会成为活动成员的端口配置为 IEEE 802.1x 端口。如果在一个 EtherChannel 端口上启用 IEEE 802.1x，设备就会出现错误消息，而 IEEE 802.1x 也不会启用；
- 如果用户在设备接口上配置了 EtherChannel，要先把 EtherChannel 的配置从接口上删除，然后再在设备上使用全局配置命令 **dot1x system-auth-control** 在全局启用 IEEE 802.1x；
- 如果用户配置了交叉堆栈 EtherChannel，和设备堆栈分离，那就有可能引发环路和转发问题。

二层 EtherChannel 配置指南

用户在配置二层 EtherChannel 时，可以参考下面的配置指南：

- 给 EtherChannel 中的所有端口分配相同的 VLAN 或者将它们都配置为 trunk 端口。拥有不同 native VLAN 的端口无法组成 EtherChannel；
- 在中继（trunking）二层 EtherChannel 中，所有端口支持的 VLAN 范围都是相同的。如果支持的 VLAN 范围不同，这些端口就无法组成 EtherChannel，即使将 PAgP 设置为 **auto** 或 **desirable** 模式也是这样；
- 只要其他配置是兼容的，那么生成树路径开销不同的端口也可以组成 EtherChannel。设置不同的生成树路径开销本身并不会让端口无法组成 EtherChannel。

三层 EtherChannel 配置指南

用户在配置三层 EtherChannel 时，可以参考下面的配置指南：

- 对于三层 EtherChannel，要向 port-channel 逻辑接口分配 IP 地址，而不是给信道中的各个物理端口分配 IP 地址；

Auto-LAG

auto-LAG 特性可以让设备在连接一台交换机的端口上自动创建 EtherChannel。在默认情况下，设备上是全局禁用 auto-LAG，但又在所有端口上启用这项特性的。当用户在全局启用 auto-LAG 时，交换机就会执行这项特性。

一旦在全局启用了 auto-LAG，有可能会出现下列情形：

- 所有端口都会参与创建 EtherChannel 的端口，前提是对端设备的端口上配置了 EtherChannel。要想了解详细信息，可以参考下表“*本地和对端设备上支持的 auto-LAG 配置*”；
- 已经是手动 EtherChannel 成员的端口不会再参与自动创建 EtherChannel；
- 如果在已经参与了自动创建 EtherChannel 的端口上禁用 auto-LAG，这个端口会从自动创建的 EtherChannel 中被解绑出来。

下表显示了本地设备和对端设备上支持的 auto-LAG 配置。

表 66：本地和对端设备上支持的 auto-LAG 配置

本地/对端设备	**Active**	**Passive**	**Auto**
Active	是	是	是
Passive	是	否	是
Auto	是	是	是

如果在全局禁用 auto-LAG，那么所有自动创建的 EtherChannel 都会成为手动的 EtherChannel。用户不能在当前的自动创建的 EtherChannel 中添加任何配置，要想这样做需要首先执行命令 **port-channel**<*channel-number*>**persistent**，将其转换为手动的 EtherChannel。

注释： Auto-LAG 会使用 LACP 协议来创建自动 EtherChannel。与每台对端设备之间只能自动创建一条 EtherChannel。

Auto-LAG 配置指南

用户在配置 Auto-LAG 特性时，可以参考下面的配置指南：

- 在全局和在端口上启用 auto-LAG 时，如果不希望端口成为自动 EtherChannel 的成员，那就在端口上禁用 auto-LAG；
- 如果一个端口已经是手动 EtherChannel 成员，那它就不会再参与自动创建 EtherChannel。如果希望将它捆绑到自动 EtherChannel 当中，首先需要将这个端口从手动 EtherChannel 中解绑；
- 在启用 auto-LAG 并且创建自动 EtherChannel 时，用户可以手动与同一台对端设备之间

手动配置多条 EtherChannel 信道。但是在默认情况下，端口会尝试与对端设备创建自动 EtherChannel；

- auto-LAG 只支持二层 EtherChannel。三层接口和三层 EtherChannel 不支持 auto-LAG；
- 交叉堆栈 EtherChannel 支持 auto-LAG。

如何配置 EtherChannel

在配置了一条 EtherChannel 之后，对 port-channel 接口所作的配置变更会作用于这个 port-channel 接口中的所有物理端口，而针对物理端口所作的配置变更则只会作用于应用配置的那个端口。

配置二层 EtherChannel（CLI）

在配置二层 EtherChannel 时，用户可以使用接口配置命令 **channel-group** 来将端口划分到 channel group。这条命令会自动创建出 port-channel 逻辑接口。

总步骤

1. configure terminal

2. interface *interface-id*

3. switchport mode {**access** | **trunk**}

4. switchport access vlan *vlan-id*

5. channel-group *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive**}

6. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**interface** *interface-id* 示例： `Device(config)# interface gigabitethernet2/0/1`	选择要配置的接口，并进入接口配置模式 有效接口为物理端口。 对于 PAgP EtherChannel，用户可以将最多 8 个相同类型和速率的端口配置为一个组； 对于 PAgP EtherChannel，用户可以将最多 16 个相同类型和速率的端口配置为一个组。其中最多 8 个活动端口，8 个处于备份模式
步骤 3	**switchport mode** {**access**	**trunk**} 示例： `Device(config-if)# switchport mode access`	将所有端口配置为同一个 VLAN 中的静态 access 端口，或者将它们配置为 trunk 端口。 如果将端口配置为静态 access 端口，那就只能给它分配一个 VLAN。VLAN 的取值范围是 1 到 4094
步骤 4	**switchport access vlan** *vlan-*	（可选）如果将端口配置为静态 access 端口，那就

	id 示例： `Device(config-if)#` **switchport access vlan 22**	只能给它分配一个 VLAN。VLAN 的取值范围是 1 到 4094
步骤 5	**channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive**} 示例： `Device(config-if)#` **channel-group 5 mode auto**	将端口分配给一个 channel group，并设置 PAgP 或 LACP 模式。 对于 **mode** 部分，选择下列关键字之一： • **auto**：仅当检测到 PAgP 设备时启用 PAgP。这个关键字会让端口进入被动协商状态，在这种状态下，端口只会响应它接收到的 PAgP 数据包，但不会发起 PAgP 数据包协商。如果 EtherChannel 成员位于堆栈中的不同设备上，则不支持配置这个关键字； • **desirable**：无条件启用 PAgP。这个关键字会让端口进入主动协商状态，在这种状态下，端口会通过发送 PAgP 数据包来与对端端口发起协商。如果 EtherChannel 成员位于堆栈中的不同设备上，则不支持配置这个关键字； • **on**：不使用 PAgP 或 LACP，强制端口建立信道。在 **on** 模式下，只有连接的对端端口组也是 **on** 模式时，EtherChannel 才会建立起来； • **non-silent**：（可选）如果设备连接到一台启用了 PAgP 的设备时，如果这个设备端口的模式为 **auto** 或 **desirable**，可以将该端口配置为非静默操作。如果不设置 **non-silent**，设备会默认执行静默。静默设置适用于与文件服务器或数据包分析设备之间的连接。这种设置可以让 PAgP 实现操作、将端口关联到 channel group，并且使用这个端口来执行传输； • **active**：仅当检测到 LACP 设备时启用 LACP。这个关键字会让端口进入主动协商状态，在这种状态下，端口会通过发送 LACP 数据包来与对端端口发起协商； • **passive**：在端口上启用 LACP，并让端口进入被动协商状态，在这种状态下，端口只会响应它接收到的 LACP 数据包，但不会发起 LACP 数据包协商
步骤 6	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置三层 EtherChannel（CLI）

用户可以执行下面的步骤来将一个以太网端口分配给三层 EtherChannel。这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. no ip address

5. no switchport

6. channel-group *channel-group-number* **mode** { **auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** | **passive** }

7. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet 1/0/2**	选择要配置的接口，并进入接口配置模式 有效接口为物理端口。 对于 PAgP EtherChannel，用户可以将最多 8 个相同类型和速率的端口配置为一个组； 对于 PAgP EtherChannel，用户可以将最多 16 个相同类型和速率的端口配置为一个组。其中最多 8 个活动端口，8 个处于备份模式
步骤 4	**no ip address** 示例： `Device(config-if)#` **no ip address**	确保物理端口上没有分配 IP 地址
步骤 5	**no switchport** 示例： `Device(config-if)#` **no switchport**	让这个端口进入三层模式
步骤 6	**channel-group** *channel-group-number* **mode** {**auto** [**non-silent**] | **desirable** [**non-silent**] | **on** } | { **active** |	将端口分配给一个 channel group，并设置 PAgP 或 LACP 模式。 对于 **mode** 部分，选择下列关键字之一： • **auto**：仅当检测到 PAgP 设备时启用 PAgP。这

	passive} 示例： `Device(config-if)#` **channel-group 5 mode auto**	个关键字会让端口进入被动协商状态，在这种状态下，端口只会响应它接收到的 PAgP 数据包，但不会发起 PAgP 数据包协商。如果 EtherChannel 成员位于堆栈中的不同设备上，则不支持配置这个关键字； • **desirable**：无条件启用 PAgP。这个关键字会让端口进入主动协商状态，在这种状态下，端口会通过发送 PAgP 数据包来与对端端口发起协商。如果 EtherChannel 成员位于堆栈中的不同设备上，则不支持配置这个关键字； • **on**：不使用 PAgP 或 LACP，强制端口建立信道。在 **on** 模式下，只有连接的对端端口组也是 **on** 模式时，EtherChannel 才会建立起来； • **non-silent**：（可选）如果设备连接到一台启用了 PAgP 的设备时，如果这个设备端口的模式为 **auto** 或 **desirable**，可以将该端口配置为非静默操作。如果不设置 **non-silent**，设备会默认执行静默。静默设置适用于与文件服务器或数据包分析设备之间的连接。这种设置可以让 PAgP 实现操作、将端口关联到 channel group，并且使用这个端口来执行传输； • **active**：仅当检测到 LACP 设备时启用 LACP。这个关键字会让端口进入主动协商状态，在这种状态下，端口会通过发送 LACP 数据包来与对端端口发起协商； • **passive**：在端口上启用 LACP，并让端口进入被动协商状态，在这种状态下，端口只会响应它接收到的 LACP 数据包，但不会发起 LACP 数据包协商
步骤 7	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置 EtherChannel 负载分担（CLI）

用户可以使用下面几种不同的转发方式之一，来配置 EtherChannel 负载分担。
这个流程是可选的。

总步骤

1. configure terminal

2. port-channel load-balance { **dst-ip** | **dst-mac** | **dst-mixed-ip-port** | **dst-port** | **extended** [**dst-ip** | **dst-mac** | **dst-port** | **ipv6-label** | **l3-proto** | **src-ip** | **src-mac** | **src-port**] | **src-dst-ip** | **src-dst-mac** **src-dst-mixed-ip-port src-dst-portsrc-ip** | **src-mac** | **src-mixed-ip-port** | **src-port**}

3. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**port-channel load-balance** { **dst-ip** | **dst-mac** | **dst-mixed-ip-port** | **dst-port** | **extended** [**dst-ip** | **dst-mac** | **dst-port** | **ipv6-label** | **l3-proto** | **src-ip** | **src-mac** | **src-port**] | **src-dst-ip** | **src-dst-mac** **src-dst-mixed-ip-port src-dst-portsrc-ip** | **src-mac** | **src-mixed-ip-port** | **src-port**} 示例： `Device(config)# port-channel load-balance src-mac`	配置 EtherChannel 负载分担方法。 默认设置为 **src-mac**。 选择下列负载分发方式之一： • **dst-ip**：设置目的主机 IP 地址； • **dst-mac**：设置入站数据包的目的主机 MAC 地址； • **dst-mixed-ip-port**：设置主机 IP 地址和 TCP/UDP 端口； • **dst-port**：设置目的 TCP/UDP 端口； • **extended**：设置扩展的负载分担方式——结合源和目的的方式； • **ipv6-label**：设置 IPv6 流标签； • **l3-proto**：设置三层协议； • **src-dst-ip**：设置源和目的主机 IP 地址； • **src-dst-mac**：设置源和目的主机 MAC 地址； • **src-dst-mixed-ip-port**：设置源和目的主机 IP 地址和 TCP/UDP 端口； • **src-dst-port**：设置源和目的 TCP/UDP 端口； • **src-ip**：设置源主机 IP 地址； • **src-mac**：设置入站数据包的源 MAC 地址； • **src-mixed-ip-port**：设置源主机 IP 地址和 TCP/UDP 端口； • **src-port**：设置源 TCP/UDP 端口
步骤 3	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置 EtherChannel 扩展的负载分担（CLI）

在想要使用组合负载分担方式时，可以配置 EtherChannel 扩展的负载分担。
这项操作是可选的。

总步骤

1. configure terminal

2. port-channel load-balance extended [**dst-ip** | **dst-mac dst-port** | **ipv6-label** | **l3-proto** | **src-ip** | **src-mac** | **src-port**]

3. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**port-channel load-balance extended [dst-ip | dst-mac dst-port | ipv6-label | l3-proto | src-ip | src-mac | src-port]** 示例： `Device(config)# port-channel load-balance extended dst-ip dst-mac src-ip`	配置 EtherChannel 扩展的负载分担方法。 默认设置为 **src-mac**。 选择下列负载分发方式之一： • **dst-ip**：设置目的主机 IP 地址； • **dst-mac**：设置入站数据包的目的主机 MAC 地址； • **dst-port**：设置目的 TCP/UDP 端口； • **ipv6-label**：设置 IPv6 流标签； • **l3-proto**：设置三层协议； • **src-ip**：设置源主机 IP 地址； • **src-mac**：设置入站数据包的源 MAC 地址； • **src-port**：设置源 TCP/UDP 端口
步骤 3	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置 PAgP 学习方式与优先级（CLI）

这项操作是可选的。

总步骤

1. configure terminal

2. interface *interface-id*

3. pagp learn-method physical-port

4. pagp port-priority *priority*

5. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**interface** *interface-id* 示例： `Device(config)# interface gigabitethernet 1/0/2`	选择用于传输流量的接口，并进入接口配置模式

步骤 3	**pagp learn-method physical-port** 示例： `Device(config-if)# pagp learn-method physical port`	选择 PAgP 学习方法。 在默认情况下，选择的是 **aggregation-port learning**，这表示设备会使用 EtherChannel 中的任意端口向源发送数据包。若使用汇聚端口学习，数据从哪个物理接口到达设备并不重要。 选择 **physical-port** 来连接另一台物理学习设备。一定要保证将全局配置命令 **port-channel load-balance** 配置为了 **src-mac**。 EtherChannel 两端必须配置相同的学习方法
步骤 4	**pagp port-priority** *priority* 示例： `Device(config-if)# pagp port-priority 200`	通过分配优先级来让设备使用用户所选的端口传输数据。 *priority* 的取值范围是从 1 到 255，默认值为 128。优先级越高，这个端口越有可能用来执行 PAgP 传输
步骤 5	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置 LACP 热备份端口

在启用了 LACP 时，软件默认会尝试配置一个信道中 LACP 兼容端口的最大数量，最大值为 16 个端口。只有 8 条 LACP 链路可以同时处于活动状态，剩下的 8 条链路则会处于热备份模式。如果活动链路之一的状态变为不活动，那么处于热备份模式的链路就会变为活动状态。用户可以设置一个信道中活动端口的最大数量，以此来覆盖默认的操作。此时，剩余端口就会成为热备份端口。例如，如果用户将信道中最大的端口数量设置为 5 个，那么就会有最多 11 个端口成为热备份端口。

如果用户给一条 EtherChannel 组中配置了多于 8 条链路，那么软件就会基于 LACP 优先级来自动决定将哪些热备份端口置于活动状态。对于 LACP 系统之间的每条链路，软件都会分配一个专门的优先级，优先级中最多会包含下面几项因素（下面因素按照优先级顺序排列）：

- LACP 系统优先级；
- 系统 ID（设备 MAC 地址）；
- LACP 端口优先级；
- 端口号

在比较优先级时，数值越低即表示优先级越高。当硬件限制防止所有兼容端口进行汇聚时，设备会使用优先级来决定应该将哪些端口置于备份模式。

判断哪些端口应该处于活动状态，哪些端口处于热备份状态，是一个两步的流程。首先，系统优先级和系统 ID 数值较低的系统会被列入考虑。接下来，系统会基于端口优先级和端口号的数值，来判断哪些端口应该处于活动状态，哪些端口则处于热备份状态。其他系统的端口优先级和端口号则不会使用。

用户可以修改 LACP 系统优先级和 LACP 端口优先级的默认值，来影响软件对活动链路和备份链路的选择。

配置 LACP Max Bundle 特性（CLI）

在设置一个 port channel 中可以捆绑的 LACP 端口最大数量时，port channel 中剩余的端口就

会被指定为热备份端口。

用户可以从特权 EXEC 模式开始，按照下面的步骤来配置一个 port channel 中支持的 LACP 端口最大数量。这个流程是可选的。

总步骤

1. configure terminal

2. interface port-channel *channel-number*

3. lacp max-bundle *max-bundle-number*

4. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**interface port-channel** *channel-number* 示例： `Device(config)# interface port-channel 2`	进入 port channel 的接口配置模式，取值范围是从 1 到 128
步骤 3	**lacp max-bundle** *max-bundle-number* 示例： `Device(config-if)# lacp max-bundle 3`	设置 port-channel 接口束中，最大的 LACP 端口数量。数量取值范围是从 1 到 8
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置 LACP Port-Channel 独立禁用

要在一个 port channel 上禁用独立 EtherChannel 成员端口状态，可以在 port channel 接口上执行下面的操作：

总步骤

1. configure terminal

2. interface port-channel *channel-group*

3. port-channel standalone-disable

4. end

5. show etherchannel

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例：	进入全局配置模式

	Device# **configure terminal**	
步骤 2	**interface port-channel** *channel-group* 示例： Device(config)# **interface port-channel** *channel-group*	选择要配置的 port channel 接口
步骤 3	**port-channel standalone-disable** 示例： Device(config-if)# **port-channel standalone-disable**	在这个 port-channel 接口上禁用独立模式
步骤 4	**end** 示例： Device(config)# end	返回特权 EXEC 模式
步骤 5	**show etherchannel** 示例： Device# **show etherchannel** *channel-group* **port-channel** Device# **show etherchannel** *channel-group* **detail**	验证所作的配置

配置 LACP Port Channel Min-Link 特性（CLI）

用户可以设置必须处于链路 up 状态，并绑定到 LACP port channel 作为 EtherChannel 的最少端口数量，允许绑定了这个数量的 port channel 接口可以过渡到链路 up 状态。使用 EtherChannel min-link 特性，可以防止低带宽的 LACP 端口变为活动状态。Port channel min-links 也会在活动成员端口的数量太少，达不到所需的最小带宽时，让 LACP port channel 成为不活动状态。

要配置 port channel 所需的最少链路数量，可以执行下面的操作。

总步骤

1. enable

2. configure terminal

3. interface port-channel *channel-number*

4. port-channel min-links *min-links-number*

5. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal**	进入全局配置模式

	示例： `Device#` **configure terminal**	
步骤 3	**interface port-channel** *channel-number* 示例： `Device(config)#` **interface port-channel 2**	进入一个 port-channel 的接口配置模式。 *channel-number* 的取值范围是从 1 到 63
步骤 4	**port-channel min-links** *min-links-number* 示例： `Device(config-if)#` **port-channel min-links 3**	设置必须处于链路 up 状态，并绑定到 LACP port channel 作为 EtherChannel 的最少端口数量，允许绑定了这个数量的 port channel 接口可以过渡到链路 up 状态。 *min-links-number* 的取值范围是 2 到 8
步骤 5	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

配置 LACP 系统优先级（CLI）

用户可以使用全局配置命令 **lacp system-priority**，给所有启用了 LACP 的 EtherChannel 配置系统优先级，但不能给每个配置了 LACP 的信道配置系统优先级。把这个数值修改为默认参数之外的值，可以影响软件选择哪些链路作为活动链路，选择哪些链路作为备份链路。

用户可以使用特权 EXEC 命令 **show etherchannel summary** 来查看哪些端口处于热备份模式（这类端口的端口状态标记为 H）。

用户可以使用下面的步骤来配置 LACP 系统优先级。这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. lacp system-priority *priority*

4. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**lacp system-priority** *priority* 示例：	配置 LACP 系统优先级。 这个参数的取值范围是 1 到 65535，默认值为 32768。

	Device(config)# **lacp system-priority 32000**	取值越低，系统优先级越高
步骤 4	**end** 示例： Device(config)# end	返回特权 EXEC 模式

配置 LACP 端口优先级（CLI）

在默认情况下，所有端口使用的是相同的端口优先级。如果本地系统的系统优先级和系统 ID 取值比远端系统低，那么用户可以将 LACP EtherChannel 的端口优先级修改为一个低于默认值的数值，来让这些热备链路首先成为活动链路。用户可以使用特权 EXEC 命令 **show etherchannel summary** 来查看哪些端口处于热备份模式（这类端口的端口状态标记为 H）。

注释： 如果 LACP 不能汇聚所有兼容的端口（比如，远端系统的硬件限制比本地系统更严格），那么所有不能主动包含在 EtherChannel 中的端口都会被置入热备份状态，只有在捆绑的端口出现故障时，才会使用这些端口。

用户可以使用下面的步骤来配置 LACP 系统优先级。这个流程是可选的。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. lacp port-priority *priority*

5. end

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： Device(config)# **interface gigabitethernet 1/0/2**	选择要进行配置的端口，并进入接口配置模式
步骤 4	**lacp port-priority** *priority* 示例： Device(config-if)# **lacp port-priority 32000**	配置 LACP 系统优先级。 这个参数的取值范围是 1 到 65535，默认值为 32768。取值越低，这个端口越有可能用于数据传输
步骤 5	**end** 示例：	返回特权 EXEC 模式

	Device(config)# end	

配置 LACP 快速计时器

用户可以修改 LACP 计时器速率，来修改 LACP 超时的时间周期。用户可以使用命令 **lacp rate** 来设置支持 LACP 的接口会以什么样的速率接收 LACP 控制数据包。用户可以将这个超时速率从默认速率（30 秒）修改为一个更快的速率（1 秒）。这条命令只能在启用了 LACP 的接口上输入。

总步骤

1. **enable**
2. **configure terminal**
3. **interface** {**fastethernet** | **gigabitethernet** | **tengigabitethernet**} *slot/port*
4. **lacp rate** {**normal** | **fast**}
5. **end**
6. **show lacp internal**

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**interface** {**fastethernet**	**gigabitethernet**	**tengigabitethernet**} *slot/port* 示例： Device(config)# **interface gigabitEthernet 2/1**	配置接口，并进入接口配置模式
步骤 4	**lacp rate {normal	fast}** 示例： Device(config-if)# **lacp rate fast**	设置支持 LACP 的接口会以什么样的速率接收 LACP 控制数据包。 • 要将超时速率重置为默认值，需要输入命令 **no lacp rate**
步骤 5	**end** 示例： Device(config)# end	返回特权 EXEC 模式
步骤 6	**show lacp internal**	验证所作的配置

	示例：
	`Device# show lacp internal`
	`Device# show lacp counters`	

在全局配置 Auto-LAG

总步骤

1. enable

2. configure terminal

3. [**no**] **port-channel auto**

4. end

5. show etherchannel auto

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	[**no**] **port-channel auto** 示例： `Device(config)#` **interface gigabitEthernet 2/1**	在一台交换机上全局启用 auto-LAG。使用这条命令的 no 形式在交换机上全局禁用 auto-LAG 特性。 **注释**：在默认情况下，端口上的 auto-LAG 特性是启用的
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式
步骤 5	**show etherchannel auto** 示例： `Device#` **show etherchannel auto**	显示自动创建的 EtherChannel

在端口上配置 Auto-LAG

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. [**no**] **channel-group auto**

5. end

6. show etherchannel auto

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device>` **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet 1/0/1**	选择要启用 auto-LAG 的端口，并进入接口配置模式
步骤 4	[**no**] **port-channel auto** 示例： `Device(config)#` **interface gigabitEthernet 2/1**	（可选）在个别接口上启用 auto-LAG。使用这条命令的 no 形式可以在个别端口上禁用 auto-LAG 特性。 **注释**：在默认情况下，端口上的 auto-LAG 特性是启用的
步骤 5	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式
步骤 6	**show etherchannel auto** 示例： `Device#` **show etherchannel auto**	显示自动创建的 EtherChannel

接下来做什么？

在配置 Auto-LAG 时配置持续功能

用户可以使用 persistence 命令将自动创建的 EtherChannel 转换为手动 EtherChannel，以便向当前的 EtherChannel 中添加配置。

总步骤

1. enable

2. port-channel *channel-number* **persistent**

3. show etherchannel summary

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**port-channel** *channel-number* **persistent** 示例： `Device# port-channel 1 persistent`	将自动创建的 EtherChannel 转换为手动 EtherChannel，以便向当前的 EtherChannel 中添加配置
步骤 3	**show etherchannel summary** 示例： `Device# show etherchannel summary`	显示 EtherChannel 的信息

监控 EtherChannel、PAgP 和 LACP 的状态

用户可以使用表中所示的命令来查看 EtherChannel、PAgP 和 LACP 的状态。

表 67：监控 EtherChannel、PAgP 和 LACP 状态的命令

命令	描述
clear lacp { *channel-group-number* **counters** | **counters** }	清除 LACP channel-group 信息和流量计数器
clear pagp { *channel-group-number* **counters** | **counters** }	清除 PAgP channel-group 信息的流量计数器
show etherchannel [*channel-group-number* { **detail** | **load-balance** | **port** | **port-channel** | **protocol** | **summary** }] [**detail** | **load-balance** | **port** | **port-channel** | **protocol** | **auto** | **summary**]	用简化的形式、详细的形式或者一行信息的形式，显示 EtherChannel 信息。同时显示负载分担和数据帧分发机制、端口、port-channel、协议和 Auto-LAG 的信息
show pagp [*channel-group-number*] { **counters** | **internal** | **neighbor** }	显示 PAgP 信息，譬如流量信息、内部 PAgP 信息和邻居信息
show pagp [*channel-group-number*] **dual-active**	显示双向活动检测状态
show lacp [*channel-group-number*] { **counters** | **internal** | **neighbor** | **sys-id**}	显示 LACP 信息，譬如流量信息、内部 PAgP 信息和邻居信息
show running-config	验证配置条目
show etherchannel load-balance	显示 port channel 中端口的负载分担或数据帧分发机制

EtherChannel 的配置示例

配置二层 EtherChannel：示例

这个示例显示了如何在堆栈中的一台设备上配置 EtherChannel。在这个示例中，两个端口被配置为了 VLAN 10 中的静态 access 端口，并且将它添加到了 PAgP 模式 **desirable** 的 channel 5 中。

```
Device# configure terminal
Device(config)# interface range gigabitethernet2/0/1 -2
Device(config-if-range)# switchport mode access
Device(config-if-range)# switchport access vlan 10
Device(config-if-range)# channel-group 5 mode desirable non-silent
Device(config-if-range)# end
```

这个示例显示了如何在堆栈中的一台设备上配置 EtherChannel。在这个示例中，两个端口被配置为了 VLAN 10 中的静态 access 端口，并且将它添加到了 LACP 模式 **active** 的 channel 5 中。

```
Device# configure terminal
Device(config)# interface range gigabitethernet2/0/1 -2
Device(config-if-range)# switchport mode access
Device(config-if-range)# switchport access vlan 10
Device(config-if-range)# channel-group 5 mode active
Device(config-if-range)# end
```

这个示例显示了如何配置交叉堆栈 EtherChannel。在这个示例中，用户使用了 LACP 被动模式，用户将堆栈成员 1 中的两个端口和堆栈成员 2 中的一个端口配置为了 VLAN 10 中的静态 access 端口，并且将它们添加到了 channel 5 当中：

```
Device# configure terminal
Device(config)# interface range gigabitethernet2/0/4 -5
Device(config-if-range)# switchport mode access
Device(config-if-range)# switchport access vlan 10
Device(config-if-range)# channel-group 5 mode passive
Device(config-if-range)# exit
Device(config)# interface gigabitethernet3/0/3
Device(config-if)# switchport mode access
Device(config-if)# switchport access vlan 10
Device(config-if)# channel-group 5 mode passive
Device(config-if)# exit
```

如果用户配置交换机上的两个端口，让它们连接接入点（AP），那么网络中有可能会出现 PoE 或 LACP 协商错误。如果 port channel 配置在交换机上，这个问题就可以避免。要想了解详细信息，可以参考下面的示例：

```
interface Port-channel1
switchport access vlan 20
switchport mode access
switchport nonegotiate
```

```
no port-channel standalone-disable <--this one
spanning-tree portfast
```

注释： 如果在端口翻动时，端口报告了 LACP 错误，用户也应该输入这条命令：**no errdisable detect cause pagp-flap**。

配置三层 EtherChannel：示例

这个示例显示了如何配置交叉堆栈三层 EtherChannel。在这个示例中，用户将两个端口添加到了 LACP 模式为 **active** 的 channel 5 当中：

Device# **configure terminal**
Device(config)# **interface range gigabitethernet2/0/1 -2**
Device(config-if-range)# **no ip address**
Device(config-if-range)# **no switchport**
Device(config-if-range)# **channel-group 5 mode active**
Device(config-if-range)# **end**

这个示例显示了如何配置交叉堆栈三层 EtherChannel。在这个示例中，用户将堆栈成员 2 中的两个端口和堆栈成员 3 中的一个端口添加到了 LACP 模式为 **active** 的 channel 7 当中：

```
Device# configure terminal
Device(config)# interface range gigabitethernet2/0/4 -5
Device(config-if-range)# no ip address
Device(config-if-range)# no switchport
Device(config-if-range)# channel-group 7 mode active
Device(config-if-range)# exit
Device(config)# interface gigabitethernet3/0/3
Device(config-if)# no ip address
Device(config-if)# no switchport
Device(config-if)# channel-group 7 mode active
Device(config-if)# exit
```

配置 LACP 热备份端口：示例

这个示例显示了如何通过配置，让 EtherChannel（port channel 2）在 port channel 中至少有 3 个活动端口的前提下进入活动状态，这个示例中包含 7 个活动端口，剩余端口（最多有 9 个）则为热备份端口：

```
Device# configure terminal
Device(config)# interface port-channel 2
Device(config-if)# port-channel min-links 3
Device(config-if)# lacp max-bundle 7
```

这个示例显示了如何在 port channel 42 上禁用独立 EtherChannel 成员端口状态：

```
Device(config)# interface port-channel channel-group
Device(config-if)# port-channel standalone-disable
```

这个示例显示了如何验证前面所作的配置：

```
Device# show etherchannel 42 port-channel | include Standalone
```

```
Standalone Disable = enabled
Device# show etherchannel 42 detail | include Standalone
Standalone Disable = enabled
```

配置 Auto LAG：示例

这个示例显示了如何在一台交换机上配置 Auto-LAG。

```
device> enable
device# configure terminal
device (config)# port-channel auto
device (config-if)# end
device# show etherchannel auto
```

下面的示例显示了自动创建的 EtherChannel 汇总信息。

```
device# show etherchannel auto
Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use f - failed to allocate aggregator
M - not in use, minimum links not met
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
A - formed by Auto LAG
Number of channel-groups in use: 1
Number of aggregators: 1
Group Port-channel Protocol Ports
------+-------------+-----------+-----------------------------------------------
1 Po1(SUA) LACP Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)
```

下面的示例显示了执行命令 **port-channel** 1 **persistent** 之后，自动 EtherChannel 的汇总信息

```
device# port-channel 1 persistent
device# show etherchannel summary
Switch# show etherchannel summary
Flags: D - down P - bundled in port-channel
I - stand-alone s - suspended
H - Hot-standby (LACP only)
R - Layer3 S - Layer2
U - in use f - failed to allocate aggregator
M - not in use, minimum links not met
u - unsuitable for bundling
w - waiting to be aggregated
d - default port
A - formed by Auto LAG
Number of channel-groups in use: 1
```

```
Number of aggregators: 1
Group Port-channel Protocol Ports
------+-------------+-----------+-----------------------------------------------
1 Po1(SU) LACP Gi1/0/45(P) Gi2/0/21(P) Gi3/0/21(P)
```

其他关于 EtherChannel 的参考资料

相关文档

相关主题	文档名
二层命令参考	《第 2/3 层命令参考手册（Inspur 6650 交换机）》

错误消息解码器

描述	链接
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

标准与 RFC

标准/RFC	标题
无	——

技术助手

描述	链接
Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。 用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	http://www.icntnetworks.com

EtherChannel 的特性信息

版本	修改
Inspur INOS 12.2	引入该特性

配置单向链路检测

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置 UDLD 的限制条件

下面是配置单向链路检测（UDLD）的限制条件：

- 支持 UDLD 的端口如果与另一台设备上不支持 UDLD 的端口相连，那么它也无法检测单向链路；
- 在配置模式（普通模式或主动模式）时，要确保链路两端配置的是同一个模式；

注意： 环路防护特性只能部署在点到点链路上。我们推荐链路每一端都有一个直连的设备在运行 STP。

关于 UDLD 的信息

单向链路检测（UDLD）是二层协议，可以让通过光纤或双绞线以太网线缆连接的设备监控线缆的物理配置，并且检测单向链路。所有相连的设备必须都能够支持 UDLD，这样协议才能成功地发现并且禁用单向链路。当 UDLD 检测到一条单向链路时，它会禁用相关端口并且向用户发出告警信息。单向链路可能会导致一系列的问题，包括生成树拓扑环路。

操作模式

UDLD 支持两种操作模式：普通（默认）模式和主动模式。在普通模式中，UDLD 可以检测到光纤连接当中因端口误连接，而产生的单向链路。在主动模式中，UDLD 也可以通过光纤、双绞线上的单向流量，或者光纤链路上端口的误连接，检测到单向链路问题。
在正常和主动模式下，UDLD 可以通过一层机制学习到链路的物理状态。在一层，自动协商机制会处理物理信令和容错检测。UDLD 会执行那些自动协商机制无法执行的任务，譬如检测邻居的身份，和关闭误连接的端口。在启用自动协商和 UDLD 时，一层和二层检测会共同防止出现物理和逻辑单向连接，以及其他协议的故障。
当邻居可以接收到本地设备发送的流量，而邻居发送的流量本地设备却无法接收到时，即表

示网络中出现了单向链路。

普通模式

在普通模式下，当光纤端口中的纤维束连接有误，而一层机制又没有检测出这个错误时，UDLD 就可以检测出单向链路。如果端口连接正确，但流量却是单向的，UDLD 就无法检测出单向链路，因为本该检测出这种情况的一层机制检测不出这种问题。此时，逻辑链路会被视为未确定，UDLD 也不会禁用这个端口。

当 UDLD 工作在普通模式下时，如果纤维对中的一条纤维束连接有误，那么只要自动协商功能正常，这条链路就不会保持在 up 状态，因为一层机制会检测出这条链路上的物理问题。在本例中，UDLD 不会采取任何操作，而逻辑链路也会视为未确定。

主动模式

在主动模式下，UDLD 会使用此前的检测方式来检测单向链路。主动模式下的 UDLD 也可以检测出点到点链路上的单向链路，而这类链路上设备之间是不允许出现故障的。当出现下列这些问题时，UDLD 也可以检测出单向链路：

- 在光纤或双绞线链路上，一个端口无法发送或接收流量；
- 在光纤或双绞线链路上，一个端口关闭，另一个端口打开；
- 线缆中一个纤维束连接错误；

在这些情况下，UDLD 都会禁用受影响的端口。

在一条点到点链路上，UDLD hello 数据包可以视为是心跳信号，它的存在是链路健康状态的佐证。相反，检测不到心跳表示如果无法重新建立双向链路的话，那么这条链路就必须关闭。如果层一层的角度来看，光纤线缆中的纤维束工作正常，那么主动模式下的 UDLD 会检测到这些纤维束是否连接正确，以及流量是否正在邻居间双向流动。这些校验是不能通过自动协商来完成的，因为自动协商是工作在一层的。

检测单向链路的方法

UDLD 有两种工作方式：

- 邻居数据库维护
- 事件驱动检测与回声

邻居数据库维护

UDLD 会在各个端口上通过周期性发送的 hello 数据包（也称为通告消息或探针）来学习其他 UDLD 邻居，以确保每台设备接收到邻居的通告。

当设备接收到 hello 消息时，它就会将信息缓存起来，直到老化时间（抑制时间或生存时间）超时为止。如果设备在较老的缓存条目超时之前，又接收到了新的 hello 消息，那么设备就会用新的条目替换掉老的条目。

当一个端口被禁用，而 UDLD 又在运行时，那么无论何时用户禁用了端口上的 UDLD，或者 UDLD 重置，UDLD 都会针对那些因配置变更而受到影响的端口，清除缓存的条目。UDLD 会发送至少一条消息来通告邻居，让它们冲刷掉受状态变更响应的那些缓存。这个消息的目的在于确保缓存条目是同步的。

事件驱动检测与回声

UDLD 在检测操作中需要依赖回声机制。只要 UDLD 设备学习到了新的邻居，或者从不同步的邻居那里接收到了一条重新同步缓存的请求，设备就会在自己连接的这一侧重新开启检测窗口，并且发送 echo（回声）消息作出响应。由于这种操作在所有 UDLD 邻居上都是相同的，因此 echo 的发送方也会期待能够接收到对方发来的 echo 消息。

如果直到检测窗口结束，设备都没有接收到响应消息，链路有可能就会关闭，具体操作取决

于 UDLD 的模式。当 UDLD 工作在普通模式下时，这条链路会被视为是未确定的，而这条链路可能不会关闭。当 UDLD 工作在主动模式下时，这条链路则会被视为是单向链路，因此端口就会被禁用。

UDLD 重置的可选项

如果一个接口因 UDLD 而被禁用，用户可以下面几种选项来重置 UDLD：

- 输入接口配置命令 **udld reset**；
- 在输入接口配置命令 **shutdown** 后，再输入接口配置命令 **no shutdown** 来重新启动禁用的端口；
- 在输入全局配置命令 **no udld** {**aggressive | enable**}之后，再输入全局配置命令 **udld** {**aggressive | enable**}来重新启用禁用的端口；
- 在输入接口配置命令 **no udld port** 之后，再输入接口配置命令 **udld port** [**aggressive**]来重新启用禁用的光纤端口；
- 输入全局配置命令 **errdisable recovery cause udld** 启用计时器，让端口自动从 UDLD error-disabled 状态恢复过来，然后输入全局配置命令 **errdisable recovery interval** *interval* 来设置从 UDLD error-disabled 恢复的时间。

默认的 UDLD 配置

表 68：默认的 UDLD 配置

特性	**默认设置**
UDLD 全局启用状态	全局禁用
光纤媒介的 UDLD 每端口启用状态	在所有以太网光纤端口上禁用
双绞线（铜线）媒介的 UDLD 每端口启用状态	在所有 Ethernet 10/100 和 1000BASE-TX 端口上禁用
UDLD 主动模式	禁用

如何配置 UDLD

在全局启用 UDLD（CLI）

用户可以按照下面的步骤在设备的所有光纤端口上启用主动或正常模式的 UDLD，并且设置消息计时器。

总步骤

1. configure terminal

2. udld {**aggressive** | **enable** | **message time** *message-timer-interval*}

3. end

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式

步骤 2	**udld** {**aggressive** | **enable** | **message time** *message-timer-interval*} 示例： `Device(config)#` **udld enable message time 10**	设置 UDLD 的操作模式： • **aggressive**：在所有光纤端口上启用主动模式的 UDLD； • **enable**：在设备的所有光纤端口上启用普通模式的 UDLD。UDLD 默认是禁用的；个别接口上的配置会覆盖全局配置命令 **udld enable** 所作的设置。 • **message time** *message-timer-interval*：配置处于通告阶段，且为双向通信状态端口，发送两条 UDLD 探针消息之间的时间周期。 **注释**：这条命令只会影响光纤端口。用户可以使用接口配置命令 **udld** 在其他类型的端口上启用 UDLD 使用这条命令的 **no** 形式来禁用 UDLD
步骤 3	**end** 示例： `Device(config)#` **end**	返回特权 EXEC 模式

在接口上启用 UDLD（CLI）

用户可以按照下面的步骤在一个端口上启用主动或正常模式的 UDLD。

总步骤

1. configure terminal

2. interface *interface-id*

3. udld port [**aggressive**]

4. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device#` **configure terminal**	进入全局配置模式
步骤 2	**interface** *interface-id* 示例： `Device(config)#` **interface gigabitethernet 1/0/1**	选择要启用 UDLD 的端口，并进入接口配置模式
步骤 3	**udld port** [**aggressive**] 示例： `Device(config-if)#` **udld port aggressive**	UDLD 默认是禁用的： • **udld port**：在特定端口上启用普通模式的 UDLD； • **udld port aggressive**：（可选）在特定端口上启用主动模式的 UDLD； **注释**：用户可以使用接口配置命令 **no udld port** 禁

		用特定光纤端口上的 UDLD
步骤 3	**end** 示例： `Device(config-if)# end`	返回特权 EXEC 模式

UDLD 的监控与维护

命令	目的
show udld [*interface-id* | **neighbors**]	显示特定端口或所有端口的 UDLD 状态

其他关于 UDLD 的参考资料

相关文档

相关主题	文档名
二层命令参考	《第 2/3 层命令参考手册（Inspur 6650 交换机）》

错误消息解码器

描述	链接
用户如需搜索和解析这个版本的系统错误消息，可以使用错误消息解码器这项工具	http://www.icntnetworks.com

标准与 RFC

标准/RFC	标题
无	——

技术助手

描述	链接
Inspur 支持（Inspur Support）页面可以为用户提供大量在线资源，其中包括排错的文档和工具，以及对 Inspur 产品与技术中若干问题的解析。 用户如需获取关于所购产品的安全与技术信息，可以选择订阅各类相关服务，譬如产品告警工具（通过最新产品问题信息汇总进行访问）、Inspur 技术服务通讯以及资讯聚合馈送（RSS Feeds）。 在 Inspur 支持页面中访问大多数工具都需要在 icntnetworks.com 上注册一个用户 ID 和密码	http://www.icntnetworks.com

UDLD 的特性信息

版本	修改

Inspur INOS 12.2	引入该特性

网络管理

配置 Inspur INOS 配置引擎

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

实施配置引擎配置的先决条件

- 获得用户连接的配置引擎实例名称；
- 由于 CNS 同时使用事件总线和配置服务器来向设备提供配置，因此用户必须为每台需要配置的设备同时定义配置 ID 和设备 ID；
- 所有配置了全局配置命令 **cns config partial** 的设备都必须访问事件总线。设备上生成的设备 ID 必须与 Inspur 配置引擎中为该设备定义的设备 ID 相同。用户必须知道自己连接的事件总线的主机名。

实施配置引擎配置的限制条件

- 在配置服务器的单个实例范围中，两台需要配置的设备不能拥有相同的配置 ID；
- 在事件总线的单个实例范围中，两台需要配置的设备不能拥有相同的设备 ID。

实施配置引擎配置的相关信息

Inspur 配置引擎软件

Inspur 配置引擎（Configuration Engine）是网络管理工具软件，它作为配置服务，能够自动部署和管理网络设备和服务。每个 Inspur 配置引擎可以管理一组 Inspur 设备（设备和路由器）及其提供的服务，同时还能够保存它们的配置，并在需要时提供这些配置。Inspur 配置引擎能够自动初始化配置和配置更新，因为它能够收集与设备相关的配置变更，将这些信息发送给设备，执行配置变更并记录变更结果。

Inspur 配置引擎支持单机模式和服务器模式，并且包含下列 Inspur 网络服务（CNS）组成部分：

- 配置服务：
  - Web 服务器
  - 文件管理器
  - Namespace 映射服务器
- 事件服务（事件网关）
- 数据服务目录（数据模型和模式）

在单机模式中，Inspur 配置引擎能够支持嵌入式目录服务。在这个模式中，不需要外部目录或其他数据存储。在服务器模式中，Inspur 配置引擎能够支持使用用户自定义的外部目录。

图 71：Inspur 配置引擎的架构概述

Service provider network	服务提供商网络
Configuration engine	配置引擎
Data service directory	数据服务目录
Configuration server	配置服务器
Event service	事件服务
Web-based user interface	基于网页的用户界面
Order entry configuration management	订单输入配置管理

配置服务

配置服务（Configuration Service）是 Inspur 配置引擎中的核心组成部分。它由配置服务器

（Configuration Server）构成，配置服务器会与设备上的 Inspur INOS CNS 代理协同工作。配置服务负责把设备和服务配置送达给设备，通过逻辑组完成初始配置和大量重配置工作。设备在网络中首次启动时，会从配置服务那里收到初始配置。
配置服务会使用 CNS 事件服务（Event Service）来发送和接收配置变更事件，以及发送成功和失败通知。
配置服务器是一台网页服务器，它所使用的配置模板和与设备相关的配置信息都保存在内置目录（单机模式）或远端目录（服务器模式）中。
配置模板是包含有静态配置信息的文本文件，文本格式与 CLI 命令相同。在模板中，变量是通过使用轻量目录访问协议（LDAP）URL 指定的，LDAP URL 调用了目录中存储的与设备相关的配置信息。
Inspur INOS 代理可以对收到的配置文件执行语法检查，并通过发布事件来展示语法检查是否成功。配置代理既可以立即应用配置，也可以等待从配置服务器那里收到同步事件后再应用配置。

事件服务

Inspur 配置引擎使用事件服务（Event Service）来接收和生成配置事件。事件服务由事件代理和事件网关构成。事件代理位于设备上，能够简化设备之间的通信；事件网关位于 Inspur 配置引擎上。
事件服务是一种高性能的发布和订阅通信方法。事件服务使用基于主题的寻址方式，向目的地发送消息。基于主题的寻址方式为消息及其目的地定义了简单且统一的命名空间。

命名空间映射器

Inspur 配置引擎中包含命名空间映射器（NSM），它为设备的管理逻辑组提供了查找服务，能够基于应用、设备或组 ID，以及事件进行查找。
Inspur INOS 设备只能够识别与 Inspur INOS 软件中配置的事件主题名称相同的事件主题名称；比如 inspur.cns.config.load。用户可以使用命名空间映射服务，通过任意命名规范来指定事件。当用户使用主题名称来发布数据存储时，NSM 会将用户定义的事件主题名称符串更改为 Inspur INOS 已知的字符串。
对于订阅者来说，通过唯一的设备 ID 和事件，命名空间映射服务会返回一组可供订阅的事件。类似的，对于发布者来说，通过唯一的组 ID、设备 ID，以及事件，映射服务会返回一组可供发布的事件。

Inspur 网络服务 ID 和设备主机名

Inspur 配置引擎认为每台需要配置的设备上都关联了唯一的标识符。这个唯一的标识符可以是多个同义词，每个同义词在特定的命名空间中是唯一的。事件服务使用命名空间内容来进行基于主题的消息寻址。
Inspur 配置引擎中有两个命名空间，一个用于事件总线，另一个用于配置服务器。在配置服务器的命名空间范围中，术语*配置 ID（ConfigID）*表示设备的唯一标识符。在事件总线的命名空间范围中，术语*设备 ID（DeviceID）*表示设备的 CNS 唯一标识符。

配置 ID

每台需要配置的设备都有一个唯一的配置 ID，这是它在 Inspur 配置引擎目录中访问相应的一组设备 CLI 属性所需的钥匙。设备上定义的配置 ID 必须与 Inspur 配置引擎中，为相应的设备所定义的配置 ID 相同。
配置 ID 在启动时即固定下来，在设备重新启动前无法修改，即使用户重新配置了设备的主机名，配置 ID 也不会受到影响。

设备 ID

事件总线上的每台需要配置的设备都有一个唯一的设备 ID，它类似于设备的源地址，这样设备就可以被定义为总线上的具体目的地了。
最初的设备 ID 是由设备的 Inspur INOS 主机名定义的。但设备 ID 是可变的，它的用法与和设备相邻的事件网关相关。
事件总线上的逻辑 Inspur INOS 端点是内嵌在事件网关中的，事件网关会以设备代理的角色发挥功能。对于事件总线来说，事件网关会代表设备及其相应的设备 ID。
设备会在自己成功连接到事件网关后，立即向事件网关告知自己的主机名。在每次连接建立时，事件网关都会把设备 ID 值和 Inspur INOS 主机名组合在一起。在与设备的连接终结前，事件网关都会保留这个设备 ID 值。

主机名和设备 ID

在设备连接到事件网关时，设备 ID 就固定了，哪怕用户重新配置了设备的主机名，设备 ID 也不会受到影响。
当用户更改设备上的设备主机名时，唯一一种能够刷新设备 ID 的做法是中断设备与事件网关之间的连接。“相关主题”中给出了刷新设备 ID 的指导。
在重新建立连接时，设备会向事件网关发送自己重新配置后的主机名。事件网关则会使用这个新的值来重新定义设备 ID。
注意： 在使用 Inspur 配置引擎的用户界面时，用户必须首先在设备 ID 字段中设置主机名值，而设备会在*之后*，而不是*之前*，获得主机名值；并且用户必须重新初始化 Inspur INOS CNS 代理的配置。否则后续的一部分配置命令操作将会出现问题。

主机名、设备 ID 和配置 ID

在单机模式中，当用户为一台设备设置了主机名值之后，配置服务器会在向改主机名发送事件时，将这个主机名作为设备 ID 使用。如果用户没有设置主机名，配置服务器会把事件发送到设备的 cn=<*value*>。
在服务器模式中不使用主机名。在这个模式中，总是使用唯一的设备 ID 属性，来发送总线上的事件。如果用户没有设置这个属性，则无法对设备进行更新。
这些属性以及其他相关的属性（标记数值对）是在 Inspur 配置引擎上运行 **Setup** 的过程中进行设置的。

Inspur INOS CNS 代理

设备通过使用 CNS 事件代理特性，能够发布和订阅事件总线上的事件，并与 Inspur INOS CNS 代理协同工作。这些代理是内嵌在设备的 Inspur INOS 软件中的，使设备能够实现连接和自动配置。

初始配置

当设备第一次启动时，它会向网络中发送广播的动态主机配置协议（DHCP）请求，来尝试获得 IP 地址。假设在这个子网上没有部署 DHCP 服务器，分布层设备会充当 DHCP 中继代理，把请求转发给 DHCP 服务器。在收到请求后，DHCP 服务器会为这台新设备分配 IP 地址，在发送给 DHCP 中继代理的单播应答消息中，还包括简单文件传输协议（TFTP）服务器 IP 地址、获得启动配置文件的路径，以及默认网关 IP 地址。DHCP 中继代理会把应答转发给设备。设备会自动（默认）在接口 VLAN 1 上配置服务器分配的 IP 地址，并从 TFTP 服务器那里下载启动配置文件。在成功下载了启动配置文件后，设备会在自己的运行配置中加载这个文件。Inspur INOS CNS 代理会使用适当的配置 ID 和事件 ID 来初始化与配置引擎之间的连接。配置引擎会把这个配置 ID 映射到一个模板中，然后把完整的配置文件下载到设备中。
下图展示出一个网络配置示例，描绘了使用基于 DHCP 的自动配置功能，获取初始的启动配置文件的环境。

图 72：初始配置

Configuration Engine	配置引擎
TFTP server	TFTP 服务器
DHCP server	DHCP 服务器
Distribution layer	分布层
DHCP relay agent default gateway	DHCP 中继代理 默认网关
Access layer switches	接入层 交换机

增量（部分）配置

在网络运行起来后，用户可以使用 Inspur INOS CNS 代理来添加新的服务。它可以向设备发送增量（部分）配置。事件网关可以把实际的配置作为事件负载进行发送（推送操作），或者也可以由单个事件触发设备发起拉取操作。
设备可以在应用配置前，检查配置语法是否正确。如果语法正确，设备会应用这个增量配置，并向配置服务器发布一个事件，表示自己已成功应用配置。如果设备没能应用增量配置，它会通过发布一个事件来展示错误状态。当设备应用了增量配置后，它可以把配置写入非易失性随机访问存储器（NVRAM）中，或者等待相关信令，在收到后需事件时再进行保存。

同步配置

当设备收到一个配置时，它可以根据是否收到写入信令（Write-Signal）事件，来决定是否推迟应用这个配置。写入信令事件会告诉设备不要把这个更新的配置写入它的 NVRAM 中。设备会把这个更新的配置作为自己的运行配置。这样做能够保证在将配置保存到 NVRAM 中（以便下次重启后使用）之前，设备的配置就能够同步其他网络活动。

自动 CNS 配置

要想为设备启用自动 CNS 配置，用户必须首先达成这部分中列出的先决条件。在达成了这些条件后，再给设备加电。在看到 **setup** 提示时什么都不要操作；设备会以初始化配置启动。在将完整的配置文件加载到设备上之后，用户无需再进行任何其他的操作。
有关初始化配置的更多信息，用户可以参考“相关主题”。

表 69：启用自动配置的先决条件

设备	**必需配置**
接入层设备	出厂默认（没有配置文件）
分布层设备	• IP Helper 地址 • 启用 DHCP 中继代理 [2] • IP 路由（如需作为默认网关使用）
DHCP 服务器	• IP 地址分配 • TFTP 服务器 IP 地址 • TFTP 服务器上的启动配置文件路径 • 默认网关 IP 地址
TFTP 服务器	• 启动配置文件，其中包含 CNS 配置命令，并通过这些命令使设备能够与配置引擎进行通信 • 确定需要进行配置的设备要使用设备 MAC 地址或序列号（代替默认主机名）来生成配置 ID 和事件 ID • 配置 CNS 事件代理把配置文件推送给设备
CNS 配置引擎	为每种类型的设备创建一个或多个模板，把设备的配置 ID 映射到模板

[2] 只有当 DHCP 服务器与客户端不属于同一个子网时，才需要使用 DHCP 代理。

如何实施配置引擎的配置

启用 CNS 事件代理

注释： 在启用 CNS 配置代理之前，用户必须先在设备上启用 CNS 事件代理。

用户可以按照以下步骤，在设备上启用 CNS 事件代理。

总步骤

1. enable

2. configure terminal

3. cns event {*hostname* | *ip-address*} [*port-number*] [[**keepalive** *seconds retry-count*] [**failover-time** *seconds*] [**reconnect-time** *time*] | **backup**]

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**cns event** {*hostname* | *ip-address*} [*port-number*] [[**keepalive** *seconds retry-count*] [**failover-time** *seconds*] [**reconnect-time** *time*] | **backup**] **示例：** `Device(config)# cns event 10.180.1.27 keepalive 120 10`	启用事件代理并输入网关参数。 • 在{*hostname* | *ip-address*}部分输入事件网关的主机名或 IP 地址 • （可选）在 *port-number* 部分输入事件网关的端口号。默认端口号是 11011 • （可选）在 **keepalive** *seconds* 部分输入让设备发送存活消息的时间间隔。在 *retry-count* 部分输入让设备重复发送存活消息的次数，在此之后连接终结。这两个参数的默认值都是 0 • （可选）在 **failover-time** *seconds* 部分输入在设备重新连接事件网关前，让设备等待的最大时间间隔 • （可选）输入 **backup** 表示这是备用网关（如果忽略这个关键字，表示这是主用网关） **注释：** 尽管在命令行的帮助信息中可以看到 **encrypt** 和 **clock-timeout** *time* 关键字，但设备实际并不支持
步骤 4	**end** **示例：**	返回特权 EXEC 模式

	Device(config)# **end**	
步骤 5	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 6	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

接下来做什么？

要想检查有关事件代理的信息，用户需要在特权 EXEC 模式中使用命令 **show cns event connections**。

要想禁用 CNS 事件代理，用户需要在全局配置模式中使用命令 **no cns event** {*ip-address* | *hostname*}。

启用 Inspur INOS CNS 代理

用户可以按照以下步骤，在设备上启用 Inspur INOS CNS 代理。

在开始前

在启用这个代理前，用户必须在设备上启用 CNS 事件代理。

总步骤

1. enable

2. configure terminal

3. cns config initial {*hostname* | *ip-address*} [*port-number*]

4. cns config partial {*hostname* | *ip-address*} [*port-number*]

5. end

6. show running-config

7. copy running-config startup-config

8. 开启设备上的 INOS CNS 代理

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**cns config initial** {*hostname* | *ip-address*} [*port-number*]	启用 Inspur INOS CNS 代理并输入配置服务器参数。 • 在{*hostname* | *ip-address*}部分输

	示例： `Device(config)# cns config initial 10.180.1.27 10`	入配置服务器的主机名或 IP 地址 • （可选）在 *port-number* 部分输入配置服务器的端口号 这条命令负责启用 Inspur INOS CNS 代理，并在设备上初始化其初始配置
步骤 4	**cns config partial** {*hostname* | *ip-address*} [*port-number*] **示例：** `Device(config)# cns config partial 10.180.1.27 10`	启用 Inspur INOS CNS 代理并输入配置服务器参数。 • 在{*hostname* | *ip-address*}部分输入配置服务器的主机名或 IP 地址 • （可选）在 *port-number* 部分输入配置服务器的端口号 这条命令负责启用 Inspur INOS CNS 代理，并在设备上初始化部分配置
步骤 5	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中
步骤 8	在设备上开启 Inspur INOS CNS 代理	

接下来做什么？

用户现在可以使用 Inspur 配置引擎，从远端向设备发送增量更新。

为 INOS CNS 代理启用初始配置

用户可以按照以下步骤，在设备上启用 CNS 配置代理并初始化其初始配置。

总步骤

1. enable

2. configure terminal

3. cns template connect *name*

4. cli *config-text*

5. 重复步骤 3 和步骤 4，配置其他 CNS 连接模板

6. exit

7. cns connect *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*]

8. discover {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*}

9. template name [*...name*]
10. 重复步骤 8 和步骤 9，在 CNS 连接配置文件中，指定更多的接口参数和 CNS 连接模板
11. exit
12. hostname *name*
13. iproute *network-number*
14. cns id *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**]
15. cns id {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]
16. cns config initial {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**]
17. end
18. show running-config
19. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**cns template connect** *name* **示例：** `Device(config)# cns template connect template-dhcp`	进入 CNS 模板连接配置模式并指定 CNS 连接模板的名称
步骤 4	**cli** *config-text* **示例：** `Device(config-tmpl-conn)# cli ip address dhcp`	在 CNS 连接模板中输入一条命令。重复这个步骤，在模板中输入每条命令
步骤 5	重复步骤 3 和步骤 4，来配置其他 CNS 连接模板	
步骤 6	**exit** **示例：** `Device(config-tmpl-conn)# exit`	返回全局配置模式
步骤 7	**cns connect** *name* [**retries** *number*] [**retry-interval** *seconds*] [**sleep** *seconds*] [**timeout** *seconds*] **示例：**	进入 CNS 连接配置模式，指定 CNS 连接配置文件的名称，并定义配置文件中的参数。设备会使用 CNS 连接配置文件来连接配置引擎。 • 在 *name* 部分输入 CNS 连接配

	Device(config)# **cns connect dhcp**	置文件的名称 • （可选）在 **retries** *number* 部分输入连接重试次数。取值范围是 1 至 30。默认值为 3 • （可选）在 **retry-interval** *seconds* 部分输入下一次尝试连接配置引擎的时间间隔。取值范围是 1 至 40 秒。默认值为 10 秒 • （可选）在 **sleep** *seconds* 部分输入首次尝试连接之前的总时间。取值范围是 0 至 250 秒。默认值为 0 秒 • （可选）在 **timeout** *seconds* 部分输入尝试连接结束时的总时间。取值范围是 10 至 2000 秒。默认值为 120
步骤 8	**discover** {**controller** *controller-type* | **dlci** [**subinterface** *subinterface-number*] | **interface** [*interface-type*] | **line** *line-type*} **示例：** Device(config-cns-conn)# **discover interface gigabitethernet**	在 CNS 连接配置文件中指定接口参数。 • 在 **controller** *controller-type* 部分输入控制器类型 • 在 **dlci** 部分输入启用的数据链路连接识别符（DLCI） （可选）在 **subinterface** *subinterface-number* 部分指定点到点子接口编号，用其来搜索启用的 DLCI • 在 **interface** [*interface-type*]部分输入接口类型 • 在 **line** *line-type* 部分输入线路类型
步骤 9	**template** *name* [*… name*] **示例：** Device(config-cns-conn)# **template template-dhcp**	在 CNS 连接配置文件中指定一系列要被应用到设备配置的模板名称。用户可以指定多个模板
步骤 10	重复步骤 8 和步骤 9，在 CNS 连接配置文件中指定更多接口参数和 CNS 连接模板	
步骤 11	**exit** **示例：** Device(config-tmpl-conn)# **exit**	返回全局配置模式

步骤 12	**hostname** *name* **示例：** `Device(config)# hostname device1`	输入设备的主机名
步骤 13	**ip route** *network-number* **示例：** `RemoteDevice(config) ip route 172.28.129.22 255.255.255.255 11.11.11.1`	（可选）配置去往配置引擎的静态路由，在 *network-number* 部分设置配置引擎的 IP 地址
步骤 14	**cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**] **示例：** `RemoteDevice(config)# cns id GigabitEthernet1/0/1 ipaddress`	（可选）设置配置引擎使用的唯一事件 ID 或配置 ID。如果用户输入这条命令，就不要输入命令 **cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**]。 • 在 *interface num* 部分输入接口类型。比如 ethernet、group-async、loopback 或 virtual-template。这个参数指定了设备应该使用哪个接口的 IP 地址或 MAC 地址来定义为一 ID • 在 {**dns-reverse** | **ipaddress** | **mac-address**} 部分输入 **dns-reverse** 表示提取主机名并将其作为唯一 ID，输入 **ipaddress** 表示使用 IP 地址，输入 **mac-address** 表示使用 MAC 地址作为唯一 ID • （可选）输入 **event** 来设置事件 ID 使用的 ID 值，以此来标识设备 • （可选）输入 **image** 来设置镜像 ID 使用的 ID 值，以此来标识设备 **注释：** 如果用户同时忽略了关键字 **event** 和 **image**，系统就会使用镜像 ID 来标识设备
步骤 15	**cns id** {**hardware-serial** | **hostname** | **string** *string* | **udi**} [**event**] [**image**] **示例：** `RemoteDevice(config)# cns id hostname`	（可选）设置配置引擎使用的唯一事件 ID 或配置 ID。如果用户输入这条命令，就不要输入命令 **cns id** *interface num* {**dns-reverse** | **ipaddress** | **mac-address**} [**event**] [**image**]。

		• 在{**hardware-serial** | **hostname** | **string** *string* | **udi**}部分输入 **hardware-serial** 表示把设备序列号设置为唯一 ID，输入 **hostname**（默认）表示将设备主机名作为唯一 ID，输入 **string** *string* 表示使用自定义字符串作为唯一 ID，或者输入 **udi** 表示设置唯一设备识别符（UDI）作为唯一 ID
步骤 16	**cns config initial** {*hostname* | *ip-address*} [*port-number*] [**event**] [**no-persist**] [**page** *page*] [**source** *ip-address*] [**syntax-check**] **示例：** `RemoteDevice(config)#` **`cns config initial 10.1.1.1 no-persist`**	启用 Inspur INOS 代理并初始化一个初始配置。 • 在{*hostname* | *ip-address*}部分输入配置服务器的主机名或 IP 地址 • （可选）在 *port-number* 部分输入配置服务器的端口号。默认端口号为 80 • （可选）当配置完成时，为配置成功、配置失败或告警消息启用 **event** • （可选）使用 **no-persist** 可以实现自动把配置写入 NVRAM，效果与使用全局配置命令 **cns config initial** 相同。如果用户没有输入关键字 **no-persist**，也可以配置 **cns config initial**，把配置自动写入 NVRAM • （可选）在 **page** *page* 部分输入初始配置的 Web 页面。默认值为/Config/config/asp • （可选）将 **source** *ip-address* 作为源 IP 地址使用 • （可选）输入 **syntax-check** 后启用语法检查功能 **注释：** 尽管在命令行的帮助信息中可以看到 **encrypt**、**status** *url* 和 **inventory** 关键字，但设备实际并不支持
步骤 17	**end** **示例：** `Device(config)#` **`end`**	返回特权 EXEC 模式
步骤 18	**show running-config**	检查用户输入的信息

	示例： Device# **show running-config**	
步骤 19	**copy running-config startup-config** 示例： Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

接下来做什么？

要想验与证配置代理相关的信息，用户需要在特权 EXEC 模式中使用命令 **show cns config connections**。

要想禁用 CNS Inspur INOS 代理，用户需要使用全局配置命令 **no cns config initial** {*ip-address* | *hostname*}。

刷新设备 ID

用户可以按照以下步骤，当设备上的主机名变更时，刷新设备 ID。

总步骤

1. enable

2. show cns config connections

3. 确保 CNS 事件代理已经正确连接到事件网关

4. show cns event connections

5. 记录步骤 4 中有关当前连接的相关信息。用户会在接下来的步骤中用到 IP 地址和端口号

6. configure terminal

7. no cns event *ip-address port-number*

8. cns event *ip-address port-number*

9. end

10. 通过命令 **show cns event connections** 的输出信息，用户需要确保已经重新建立了设备与事件代理之间的连接

11. show running-config

12. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**show cns config connections** 示例： Device# **show cns config connections**	查看 CNS 事件代理是否正在连接网关、已连接或处于活跃状态，并且查看事件代理使用的网关 IP 地址及其端口号
步骤 3	确保 CNS 事件代理已经正确连接到事件	查看命令 **show cns config**

	网关	**connections** 中的下列输出信息： • 连接是活跃的 • 连接使用的是当前配置的设备主机名。通过接下来的步骤，设备 ID 将会刷新为使用新的主机名
步骤 4	**show cns event connections** **示例：** `Device# show cns event connections`	查看用户设备的事件连接信息
步骤 5	记录步骤 4 中有关当前连接的相关信息。用户会在接下来的步骤中用到 IP 地址和端口号	
步骤 6	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 7	**no cns event** *ip-address port-number* **示例：** `Device(config)# no cns event 172.28.129.22 102`	指定用户在步骤 5 中记录的 IP 地址和端口号。 这条命令会断开设备和事件网关之间的连接。必须先断开连接，之后再重新建立连接，这样才能刷新连接使用的设备 ID
步骤 8	**cns event** *ip-address port-number* **示例：** `Device(config)# cns event 172.28.129.22 2012`	指定用户在步骤 5 中记录的 IP 地址和端口号。 这条命令会重新建立设备和事件网关之间的连接。
步骤 9	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 10	通过命令 **show cns event connections** 的输出信息，用户需要确保已经重新建立了设备与事件代理之间的连接	
步骤 11	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 12	**copy running-config startup-config** **示例：** `Device# copy running-config`	（可选）把输入的命令保存到配置文件中

	startup-config	

为 Inspur INOS CNS 代理启用部分配置

用户可以按照以下步骤，在设备上启用 Inspur INOS CNS 代理，并初始化一个部分配置。

总步骤

1. enable

2. configure terminal

3. cns config partial {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*]

4. end

5. show running-config

6. copy running-config startup-config

具体配置

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**cns config partial** {*ip-address* | *hostname*} [*port-number*] [**source** *ip-address*] **示例：** `Device(config)# cns config partial 172.28.129.22 2013`	启用配置代理并初始化部分配置。 • 在{*ip-address* | *hostname*}部分输入配置服务器的 IP 地址或主机名 • （可选）在 *port-number* 部分输入配置服务器的端口号。默认端口号为 80 • （可选）输入作为源 IP 地址使用的 **source** *ip-address* **注释：** 尽管在命令行的帮助信息中可以看到关键字 **encrypt**，但设备实际并不支持
步骤 4	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 6	**copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

	示例： `Device# copy running-config startup-config`	

接下来做什么？

要想检查与配置代理相关的信息，用户需要在特权 EXEC 模式中使用命令 **show cns config stats** 或 **show cns config outstanding**。

要想禁用 Inspur INOS 代理，用户需要使用全局配置命令 **no cns config partial** {*ip-address* | *hostname*}。要想取消部分配置，用户需要使用全局配置命令 **cns config cancel**。

监控 CNS 的配置

表 70：与 CNS 相关的 show 命令

命令	目的
show cns config connections `Device# show cns config connections`	显示 CNS Inspur INOS CNS 代理连接的状态
show cns config connections `Device# show cns config connections`	显示有关增量（部分）CNS 配置的信息，其中这些配置已经开始并且还未结束
show cns config stats `Device# show cns config stats`	显示有关 Inspur INOS CNS 代理的信息
show cns event connections `Device# show cns event connections`	显示 CNS 事件代理连接的状态信息
show cns event gateway `Device# show cns event gateway`	显示用户设备的事件代理信息
show cns event stats `Device# show cns event stats`	显示 CNS 事件代理的统计状态信息
show cns event subject `Device# show cns event subject`	显示应用订阅的事件代理项目列表

其他参考资料

相关文档

相关主题	文档名称
配置引擎的初始设置	*Inspur Configuration Engine Installation and Setup Guide, 1.5 for Linux* http://www.icntnetworks.com

错误消息解码器

描述	链接

为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

配置 Cisco 发现协议

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

有关 CDP 的信息

CDP 概述

CDP 是运行在二层（数据链路层）的设备发现协议，Inspur 的所有设备（路由器、网桥、接入服务器、控制器和交换机）都支持 CDP，网络管理应用可以通过 CDP 发现邻居 Inspur 设备。通过使用 CDP，网络管理应用可以学到（运行较低层透明协议的）邻居设备的设备类型和简单网络管理协议（SNMP）代理地址。应用能够通过这个特性向邻居设备发送 SNMP 查询消息。

CDP 能够在所有支持子网接入协议（SNAP）的媒介上运行。由于 CDP 只运行在数据链路层，因此两个运行不同网络层协议的系统也可以学习到对方的信息。

每台配置了 CDP 的设备都会周期性向一个组播地址发送消息，这个通告中至少包含一个它可以接收 SNMP 消息的地址。通告中还包含存活时间或保持时间信息，这个时间间隔表示接收方设备会在这么长时间后丢弃它所收到的 CDP 信息。每台设备还会监听其他设备发送的消息，以此学习邻居设备的信息。

对于设备来说，网络助手（Network Assistant）工具可以通过使用 CDP，以图形的方式展示网络结构。设备可以使用 CDP 找到集群候选者，并维护集群成员和其他设备的信息，

默认的 CDP 配置

下面这个表格中展示了默认的 CDP 配置。

特性	**默认设置**
CDP 全局状态	已启用
CDP 接口状态	已启用
CDP 计时器（数据包更新频率）	60 秒
CDP 保持时间（超时丢弃）	180 秒
CDP 版本 2 通告	已启用

如何配置 CDP

配置 CDP 特征

用户可以配置下列 CDP 特征：

- CDP 更新的频率
- 在丢弃前，维护信息的时长
- 是否发送版本 2 通告

注释： 步骤 3 至步骤 5 都是可选配置，在配置时可以打乱顺序。

用户可以按照以下步骤配置 CDP 特征。

总步骤

1. enable
2. configure terminal
3. cdp timer *seconds*
4. cdp holdtime *seconds*
5. cdp advertise-v2
6. end
7. show running-config
8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**cdp timer** *seconds* 示例： `Device(config)# cdp timer 20`	（可选）以秒为单位配置 CDP 更新的传输频率。 取值范围是 5 至 254；默认值为 60 秒
步骤 4	**cdp holdtime** *seconds* 示例： `Device(config)# cdp holdtime 60`	（可选）设置一个时间值，接收方设备在这段时间内应该保留本端设备发送的信息，超时后丢弃信息。 取值范围是 10 至 255 秒；默认值为 180 秒
步骤 5	**cdp advertise-v2** 示例： `Device(config)# cdp advertise-v2`	（可选）配置 CDP 来发送版本 2 通告。这是默认状态
步骤 6	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式
步骤 7	**show running-config** 示例： `Device# show running-config`	检查用户输入的信息
步骤 8	**copy running-config startup-config** 示例：	（可选）把输入的命令保存到配置文件中

	Device# **copy running-config startup-config**	

接下来做什么？

用户可以在 CDP 命令前添加关键字 **no**，使配置恢复默认值。

禁用 CDP

CDP 默认是启用的。

注释： 设备集群和其他 Inspur 设备（如 Inspur IP 电话）会有规律地交换 CDP 消息。禁用 CDP 会中断集群发现和设备连接。

用户可以按照以下步骤禁用 CDP 设备发现功能。

总步骤

1. enable

2. configure terminal

3. no cdp run

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**no cdp run** **示例：** Device(config)# **no cdp run**	禁用 CDP
步骤 4	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 6	**copy running-config startup-config** **示例：**	（可选）把输入的命令保存到配置文件中

	Device# **copy running-config startup-config**	

接下来做什么？

要想使用 CDP，用户必须再次启用它。

启用 CDP

CDP 默认就是启用的。

注释： 设备集群和其他 Inspur 设备（如 Inspur IP 电话）会有规律地交换 CDP 消息。禁用 CDP 会中断集群发现和设备连接。

当 CDP 功能被禁用时，用户可以按照以下步骤启用 CDP 设备发现功能。

在开始前

CDP 必须是禁用状态，否则无法启用。

总步骤

1. enable

2. configure terminal

3. cdp run

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**cdp run** **示例：** Device(config)# **cdp run**	启用 CDP
步骤 4	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 6	**copy running-config startup-config**	（可选）把输入的命令保存到配置文

	示例： `Device# copy running-config startup-config`	件中

接下来做什么？

用户可以使用命令 **show run all** 来确认 CDP 已被启用。如果用户只使用了命令 **show run**，可能无法看到 CDP 的启用状态。

在接口上禁用 CDP

在所有支持 CDP 的接口上，CDP 默认都是启用的，接口能够发送和接收 CDP 信息。

注释： 设备集群和其他 Inspur 设备（如 Inspur IP 电话）会有规律地交换 CDP 消息。禁用 CDP 会中断集群发现和设备连接。

用户可以按照以下步骤，在端口上禁用 CDP 设备发现功能。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. no cdp enable

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet1/0/1`	指定用户希望禁用 CDP 的接口，并进入该接口的配置模式
步骤 4	**no cdp enable** **示例：** `Device(config-if)# no cdp enable`	在步骤 3 指定的接口上禁用 CDP
步骤 5	**end**	返回特权 EXEC 模式

	示例： `Device(config-if)# end`	
步骤 6	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

在接口上启用 CDP

在所有支持 CDP 的接口上，CDP 默认都是启用的，接口能够发送和接收 CDP 信息。

注释： 设备集群和其他 Inspur 设备（如 Inspur IP 电话）会有规律地交换 CDP 消息。禁用 CDP 会中断集群发现和设备连接。

如果 CDP 已被禁用，用户可以按照以下步骤，在端口上启用 CDP 设备发现功能。

在开始前

CDP 在该接口上的状态必须为禁用，否则无法启用它。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. cdp enable

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**interface** *interface-id* **示例：** `Device(config)# interface`	指定用户希望禁用 CDP 的接口，并进入该接口的配置模式

	gigabitethernet1/0/1	
步骤 4	**cdp enable** **示例：** Device(config-if)# **cdp enable**	在已禁用 CDP 的接口上启用 CDP
步骤 5	**end** **示例：** Device(config-if)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

监控和维护 CDP

表 71：显示 CDP 信息的命令

命令	**描述**
clear cdp counters	把流量计数器重置为 0
clear cdp table	删除 CDP 表中有关邻居的信息
show cdp	显示全局信息，比如发送数据包的传输频率和保持时间
show cdp entry *entry-name* [**version**] [**protocol**]	显示有关指定邻居的信息。 用户可以通过输入星号（*）查看所有邻居，或者也可以输入邻居名称来查看相应邻居的信息。 用户还可以限制显示内容，比如只查看指定邻居上启用的协议，或者只查看设备上运行的软件版本
show cdp interface [*interface-id*]	显示启用了 CDP 的接口信息。 用户可以只查看某一个接口的信息
show cdp neighbors [*interface-id*] [**detail**]	显示有关邻居的信息，其中包括设备类型、接口类型和编号、保持时间的设置、能力、平台，以及端口 ID。 用户可以只查看某个接口上的邻居信息，或者丰富显示内容，查看详细信息

show cdp traffic	显示 CDP 计数器，其中包括发送和接收的数据包数量，以及校验和的错误数量

其他参考资料

相关文档

相关主题	文档名称
系统管理命令	*Network Management Command Reference, Inspur INOS*

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

配置简单网络管理协议

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

部署 SNMP 的先决条件

支持的 SNMP 版本

本软件版本支持下列版本的 SNMP：

- SNMPv1——简单网络管理协议，它是完整的 Internet 标准，定义在 RFC 1157 文档中；
- SNMPv2C 使用基于团体（Community）字符串的管理架构，代替了 SNMPv2Classic 中基于派别（Party）的管理和安全架构，同时保留了 SNMPv2Classic 中的批量检索并增强了错误处理功能。它拥有以下特性：
  - SNMPv2——简单网络管理协议版本 2，它是 Internet 标准草案，定义在 RFC 1902 至 1907 文档中；
  - SNMPv2C——使用基于团体字符串的管理架构，它是实验性 Internet 协议，定义在 RFC 1901 文档中。
- SNMPv3——SNMP 版本 3，它是具有互操作性的标准协议，定义在 RFC 2273 至 2275 文档中。SNMPv3 提供了安全接入设备的方法，它能够在网络中对数据包执行认证和加密，并包含以下安全特性：
  - 消息完整性——确保数据包没有在传输过程中遭到篡改；
  - 认证——确定消息来自于合法的源；
  - 加密——对消息的内容提供保护，防止未经授权的源对其进行读取。

注释： 要想选择加密功能，用户需要输入关键字 **priv**。

SNMPv1 和 SNMPv2C 都使用基于团体的方式来提供安全保护。管理器团体能够访问代理的 MIB，用户需要使用 IP 地址访问控制列表和密码来对管理器团体进行定义。

SNMPv2C 中包含批量检索功能，还向管理站提供了更详细的错误消息报告。批量检索功能能够在多种表格和大量信息中进行检索，把检索所需的往返数量降到最低。SNMPv2C 增强了错误处理功能，其中包括丰富的错误代码，能够区分不同类型的错误条件；在 SNMPv1 中，这些条件都会报告为一个单独的错误代码。SNMPv2C 中的错误返回代码报告了错误类型。

SNMPv3 同时提供了安全模型和安全级别。安全模型是为用户以及用户所属的组设置的指认证策略。安全级别是指一个安全模型所属的安全级别。通过把安全级别和安全模型结合在一起，就能够在处理 SNMP 数据包时，决定使用哪种安全方式了。各种安全模型包括 SNMPv1、

SNMPv2C 和 SNMPv3。

下面这个表格中展示了这些安全模型的特征，并对比了各种安全模型和安全级别的组合方式。

表 72：SNMP 安全模型和安全级别

模型	级别	认证	加密	结果
SNMPv1	无认证无加密	团体字符串	无	使用团体字符串来进行认证
SNMPv2C	无认证无加密	团体字符串	无	使用团体字符串来进行认证
SNMPv3	无认证无加密	用户名	无	使用用户名来进行认证
SNMPv3	有认证无加密	消息摘要 5（MD5）或安全散列算法（SHA）	无	基于 HMAC-MD5 或 HMAC-SHA 算法来提供认证
SNMPv3	有认证有加密	MD5 或 SHA	数据加密标准（DES）或高级加密标准（AES）	基于 HMAC-MD5 或 HMAC-SHA 算法来提供认证。 用户能够使用下列加密算法，定义自己的安全模型（USM）： • DES 56 比特加密，基于 CBC-DES（DES-56）标准的认证 • 3DES 168 比特加密 • AES 128 比特、192 比特或 256 比特加密

用户必须配置 SNMP 代理，才能使用管理站支持的 SNMP 版本。由于一个代理能够与多个管理器进行通信，因此用户可以配置 SNMP 代理分别使用 SNMPv1、SNMPv2C 或 SNMPv3 与管理器进行通信。

部署 SNMP 的限制条件

版本限制

- SNMPv1 不支持通知 Inform 消息。

有关 SNMP 的信息

SNMP 概述

SNMP 是一项应用层协议，它为管理器和代理之间的通信提供了一种消息格式。SNMP 系统由 SNMP 管理器、SNMP 代理和管理信息库（MIB）构成。SNMP 管理器可以是网络管理系统（NMS）的一部分，比如 Inspur Prime Infrastructure 就是一种 NMS。代理和 MIB 位于设备上。要想在设备上配置 SNMP，用户需要定义管理器和代理之间的关系。

SNMP 代理中包含 MIB 变量，SNMP 管理器可以请求或更改其中的变量值。管理器可以从代理那里获得一个值，也可以向代理中存入一个值。代理可以从 MIB 中收集数据，MIB 中保存了有关设备参数和网络数据的信息。代理可以对管理器发来的获取或设置数据的请求作出响应。

代理可以向管理器发送未经请求的 Trap 消息。Trap 消息是用来警示 SNMP 管理器的，它说明了网络中的某种状况。Trap 可以通告错误的用户认证、重新启动、链路状态（Up 或 Down）、MAC 地址追踪、TCP 连接关闭、邻居连接断开，或其他重要的事件。

SNMP 管理器功能

SNMP 管理器会使用 MIB 中的信息来执行一些操作，下面这个表格中展示了这些操作：

表 73：SNMP 的操作

操作	描述
get-request	检索指定变量的值
get-next-request	从一个表中检索指定变量的值 [3]
get-bulk-request[4]	检索大数据块（比如一个表中的多个行），否则需要传输多个小数据块
get-response	NMS 发送的对于 get-request、get-next-request 和 set-request 请求的响应
set-request	设置指定变量的值
trap	SNMP 代理向 SNMP 管理器发送的未经请求的消息，SNMP 代理会在发生特定事件时进行发送

[3] 在使用这个操作时，SNMP 管理器无需指导具体的变量名称。设备会按顺序在表中查找所需变量。

[4] get-bulk 命令只适用于 SNMPv2 及其后续版本。

SNMP 代理功能

SNMP 代理能够对 SNMP 管理器发出请求作出下列应答：

- 获得（Get）一个 MIB 变量——SNMP 代理使用这个功能对 NMS 发出的请求作应答。SNMP 代理会检索 NMS 请求的 MIB 变量值，然后把这个值发送给 NMS；
- 设置（Set）一个 MIB 变量——SNMP 代理使用这个功能对 NMS 发出的消息作应答。SNMP 代理会把相应的 MIB 变量值设置为 NMS 要求的值。

SNMP 代理还能够发送未经请求的 Trap 消息，以此向 NMS 通知代理上正在发生的重要事件。发送 Trap 的条件包括但不限于以下这些：端口或模块状态改变（Up 或 Down）、生成树拓扑发生变化，以及认证失败。

SNMP 团体字符串

SNMP 使用团体字符串作为内嵌密码，对去往 MIB 对象和功能的访问行为进行认证。为了让 NMS 能够访问设备，NMS 上定义的团体字符串必须与设备上定义的三个团体字符串之一相匹配。

团体字符串可以包含下列属性之一：

- 只读（RO）——为授权的管理站提供 MIB 中所有对象（除团体字符串之外）的读取权限，但不允许写入访问；
- 读写（RW）——为授权的管理站提供 MIB 中所有对象的读写访问权限，但不允许访问团体字符串；
- 在创建集群（Cluster）时，命令设备负责管理成员设备与 SNMP 应用之间的消息交换。网络助手（Network Assistant）软件会把成员设备编号（@esN，其中 N 就是设备编号）附加到命令设备上第一个配置的 RW 和 RO 团体字符串上，并把它们传输给成员设备。

图 73：SNMP 网络

SNMP Manager	SNMP 管理器
Network device	网络设备
SNMP Agent	SNMP 代理

SNMP 通知

SNMP 允许设备在特定事件发生时，向 SNMP 管理器发送通知。设备可以通过 Trap 或 Inform 请求的形式发送 SNMP 通知。在命令语法中，除非命令提供了具体的 traps 或 informs 关键字，否则关键字 traps 表示 Trap 或 Inform，或者同时表示两者。用户可以使用命令 **snmp-server host**，来指定以 Trap 或 Inform 的形式发送 SNMP 通知。

注释： SNMPv1 不支持 Inform 消息。

Trap 消息是不可靠的，因为接收方在收到 Trap 消息后，并不会发送确认消息；发送方无法确认对方是否收到了它发送的 Trap 消息。当 SNMP 管理器收到 Inform 请求时，它会通过 SNMP 响应协议数据单元（PDU）来对这个消息进行确认。如果发送方没有收到响应消息，它就会再次发送这个 Inform 请求。由于 Inform 是可以重新发送的，因此它比 Trap 消息更有可能成功到达目的地。

正因为 Inform 消息比 Trap 消息更加可靠，使用 Inform 消息也消耗了更多的设备和网络资

源。设备在发送 Trap 消息后会立即丢弃这个消息，与此不同的是，设备在发送 Inform 请求后，会将其保存在内存中，直到它收到了有关这个 Inform 请求的响应消息，或者直到请求超时。对于 Trap 消息，设备只会发送一次；但对于 Inform 消息，设备可能会发送或尝试发送多次。尝试的次数越多，对网络中带来的流量和负载也就越大。因此用户在选择 Trap 和 Inform 时，需要在可靠性和资源消耗上进行权衡。如果需要保障 SNMP 管理器能够收到每个通知，就使用 Inform 请求。如果需要着重考量网络中的流量或设备中的内存，而且通知消息也并不是必需的，就使用 Trap。

默认 SNMP 配置

特性	默认设置
SNMP 代理	禁用 [5]
SNMP Trap 接收方	无配置
SNMP Trap 消息	未启用，除了 TCP 连接（TTY 线路）的 Trap 消息
SNMP 版本	若没有使用 version 关键字，则默认为版本 1
SNMPv3 认证	若没有输入关键字，则默认为 **noauth**（无认证无加密）安全级别
SNMP 通知类型	若没有指定类型，则默认发送所有通知

[5] 这是设备启动时的默认状态，并且在启动配置中也不包含任何全局配置命令 **snmp-server**。

SNMP 配置指导

如果设备刚刚启动，并且设备启动配置文件中至少有一条全局配置命令 **snmp-server**，设备上就启用了 SNMP 代理。

SNMP *组（Group）*是用来把 SNMP 用户与 SNMP 视图映射在一起的表。SNMP *用户（User）*是 SNMP 组中的成员。SNMP *主机（Host）*是 SNMP Trap 操作的接收方。SNMP *引擎 ID（Engine ID）*是本地或远端 SNMP 引擎的名称。

在配置 SNMP 时，用户需要遵从以下三条指导方针：

- 在配置 SNMP 组时，不要指定通知视图（Notify View）。全局配置命令 **snmp-server host** 会为用户自动生成通知视图，并且把它添加到与用户相关联的组中。在修改组的通知视图时，会影响与这个组相关联的所有用户；
- 要想配置远端用户，需要指定这个用户所在设备的远端 SNMP 代理所使用的 IP 地址或端口号；
- 在为指定代理配置远端用户时，需要在全局配置命令 **snmp-server engineID** 中，通过 **remote** 选项配置 SNMP 引擎 ID。远端代理的 SNMP 引擎 ID 和用户密码会被拿来计算认证和加密摘要。如果事先没有配置远端引擎 ID，则配置命令无法成功实施；
- 在配置 SNMP Inform 通知时，用户需要首先为 SNMP 数据库中的远端代理配置 SNMP 引擎 ID，然后才能向这个它发送代理请求或 Inform 消息；
- 如果本地用户没有与远端主机进行关联，设备就不会为 **auth**（有认证无加密）和 **priv**（有认证有加密）认证级别发送 Inform 消息；
- 改变 SNMP 引擎 ID 的值会带来严重后果。（在命令行中输入的）用户的密码会基于密码和本地引擎 ID，转换为 MD5 或 SHA 安全摘要。之后这个命令行密码会被丢弃，这是 RFC

2274 文档中的要求。由于有了这个要求，当引擎 ID 的值发生改变时，SNMPv3 用户的安全摘要就会变得不可用，用户需要通过全局配置命令 **snmp-server user** *username* 重新配置 SNMPv3 用户。当引擎 ID 发生变化时，也由于同样的限制因素，用户需要重新配置团体字符串。

如何配置 SNMP

禁用 SNMP 代理

用户可以使用全局配置命令 **no snmp-server**，禁用设备上运行的所有版本的 SNMP 代理（版本 1、版本 2C 和版本 3）。用户可以通过在设置 SNMP 代理时配置的第一条全局配置命令 **snmp-server**，重新启用所有类型的 SNMP 代理功能。Inspur INOS 命令中并没有为启用 SNMP 功能设置单独的命令。

用户可以按照以下步骤禁用 SNMP 代理。

在开始前

SNMP 代理特性当前必须是启用的，用户才能将其禁用。设备中输入的第一条全局配置命令 **snmp-server** 就会启用 SNMP 代理。

总步骤

1. enable

2. configure terminal

3. no snmp-server

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**no snmp-server** **示例：** `Device(config)# no snmp-server`	禁用 SNMP 代理功能
步骤 4	**end** **示例：**	返回特权 EXEC 模式

	Device(config)# **end**	
步骤 5	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 6	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

配置团体字符串

用户需要使用团体字符串来定义 SNMP 管理器与代理之间的关系。团体字符串的工作类似于密码，它能够允许管理器访问设备上的代理。用户可以（可选）将下列特征中的一个或多个与字符串结合使用：

- 匹配 SNMP 管理器 IP 地址的访问列表,用户允许这些管理器通过团体字符串访问代理；
- MIB 视图，其中定义了指定团体能够访问的所有 MIB 对象中的一部分；
- 指定团体对于 MIB 对象的读写权限或只读权限。

用户可以按照以下步骤，在设备上配置团体字符串。

总步骤

1. enable

2. configure terminal

3. snmp-server community *string* [**view** *view-name*] [**ro** | **rw**] [*access-list-number*]

4. access-list *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*]

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**snmp-server community** *string* [**view** *view-name*] [**ro**	**rw**] [*access-list-number*] **示例：**	配置团体字符串。 **注释：** @符号是用来界定环境信息的。不要在配置这条命令时，在 SNMP 团体字符串中使用@符号。 • 在 *string* 部分指定一个字符串，

	Device(config)# **snmp-server community comaccess ro 4**	当作密码，允许访问 SNMP 协议。用户可以配置一个或多个任意长度的团体字符串 • （可选）在 **view** 部分指定这个团体能够访问的视图记录 • （可选）如果用户希望授权管理站能够检索 MIB 对象，就配置只读（**ro**）权限；如果用户希望授权管理站能够检索和修改 MIB 对象，就配置读写（**rw**）权限。默认情况下，团体字符串会放行去往所有对象的只读访问 • （可选）在 *access-list-number* 部分指定编号的标准 IP 访问列表，取值范围是 1 至 99 和 1300 至 1999
步骤 4	**access-list** *access-list-number* {**deny** | **permit**} *source* [*source-wildcard*] **示例：** Device(config)# **access-list 4 deny any**	（可选）如果用户在步骤 3 中指定了标准 IP 访问列表，就需要创建一个列表，用户可以根据需要多次重复配置这条命令。 • 在 *access-list-number* 部分输入步骤 3 中指定的访问列表编号 • 关键字 **deny** 会在条件匹配时拒绝访问。关键字 **permit** 会在条件匹配时放行访问 • 在 *source* 部分输入 SNMP 管理器的 IP 地址，也就是用户希望能够通过团体字符串访问代理的管理器 • （可选）在 *source-wildcard* 部分以点分十进制格式，输入与源相匹配的通配符比特。把希望在对比中忽略的二进制位设置为 1 要记住，所有访问列表的末尾都有隐含拒绝所有数据包的语句
步骤 5	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 7	**copy running-config startup-config**	（可选）把输入的命令保存到配置文

	示例： `Device# copy running-config startup-config`	件中

接下来做什么？

要想禁止一个 SNMP 团体的访问权限，用户可以把这个团体的团体字符串设置为空字符（也就是配置团体字符串时不输入任何值）。

要想删除某个团体字符串，可以使用全局配置命令 **no snmp-server**。

用户可以为本地或远端设备上的 SNMP 服务器引擎指定一个名称（引擎 ID）。用户可以通过配置 SNMP 服务器组，把 SNMP 用户映射到 SNMP 视图，也可以把新用户添加到 SNMP 组中。

配置 SNMP 组和用户

用户可以为本地或远端设备上的 SNMP 服务器引擎指定一个名称（引擎 ID）。用户可以通过配置 SNMP 服务器组，把 SNMP 用户映射到 SNMP 视图，也可以把新用户添加到 SNMP 组中。

用户可以按照以下步骤，在设备上配置 SNMP 组和用户。

总步骤

1. enable

2. configure terminal

3. snmp-server engineID {**local** *engineid-string* | **remote** *ip-address* [**udp-port** *port-number*] *engineid-string*}

4. snmp-server group *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}}[**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]

5. snmp-server user *username group-name* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] } [**priv** {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*]

6. end

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**snmp-server engineID** {**local** *engineid-string*	**remote** *ip-address* [**udp-port**	为本地或远端的 SNMP 副本配置名称。 • 在 *engineid-string* 部分为 SNMP 副

	port-number] *engineid-string*} **示例：** Device(config)# **snmp-server engineID local 1234**	本指定最长 24 字符的 ID 字符串。如果引擎 ID 是以 0 结尾的，用户无需指定完整的 24 字符。用户可以只指定到 0 之前的字符。示例中的配置指定的引擎 ID 是 123400000000000000000000 • 如果用户选择了关键字 **remote**，需要在 *ip-address* 部分指定包含远端 SNMP 副本的设备，还可以（可选）设置远端设备上使用的用户数据报协议（UDP）端口。端口默认为 62
步骤 4	**snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}}[**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] **示例：** Device(config)# **snmp-server group public v2c access lmnop**	在远端设备上配置新的 SNMP 组。 在 *group-name* 部分指定租的名称。 指定以下安全模型之一： • **v1** 是可用安全级别中提供最低安全保障的级别 • **v2** 是倒数第 2 低的安全模型。它允许传输 Inform 消息，并且是正常宽度的两倍 • **v3** 最安全的选择，要求用户选择以下安全级别之一： **auth**——启用消息摘要 5（MD5）和安全散列算法（SHA）进行数据包认证 **noauth**——启用无认证无加密安全级别。如果用户没有指定关键字，这就是默认设置 **priv**——启用数据加密标准（DES）进行数据包加密（也成为隐私） （可选）在 **read** *readview* 部分输入一个字符串（不超过 64 字符），指定具体视图的名称，在这个视图中，只能读取代理中的内容。 （可选）在 **write** *writeview* 部分输入一个字符串（不超过 64 字符），指定具体视图名称，在这个视图中，能够写入数据和配置代理中的内容。 （可选）在 **notify** *notifyview* 部分输入一个字符串（不超过 64 字符），指定具体视图名称，在这个视图中设置 Notify、Inform 或 Trap 消息。 （可选）在 **access** *access-list* 部分输入一个字符串（不超过 64 字符），指定访

		问列表的名称
步骤 5	**snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] } [**priv** {**des** | **3des** | **aes** {**128** | **192** | **256**}} *priv-password*] **示例：** `Device(config)# snmp-server user Pat public v2c`	为 SNMP 组添加一个新用户。 在 *username* 部分输入主机上用户的名称，这台主机连接着代理。 在 *group-name* 部分输入组的名称，也就是希望用户关联的组。 使用关键字 **remote** 指定远端 SNMP 实体，也就是用户所属的 SNMP 实体，指定这个实体的主机名或 IP 地址，（可选）以及 UDP 端口号。默认端口号为 62。 输入 SNMP 版本号（**v1**、**v2c** 或 **v3**）。 在输入 **v3** 后，用户还可以配置以下选项： • **encrypted** 指定密码的加密格式。这个关键字只有当用户配置了 **v3** 时才可以使用 • **auth** 是认证级别设置会话，可以是 HMAC-MD5-96（**md5**）认证级别，也可以是 HMAC-SHA-96（**sha**）认证级别；用户同时还要配置密码字符串 *auth-password*（不超过 64 字符） 在输入 **v3** 后，用户还可以配置隐私（**priv**）加密算法和密码字符串 *priv-password*，使用下列关键字（不超过 64 字符）： • **priv** 指定用户自定义安全模型（USM） • **des** 指定使用 56 比特 DES 算法 • **3des** 指定使用 168 比特 DES 算法 • **aes** 指定使用 DES 算法。用户必须在 128 比特、192 比特或 256 比特算法之中选择其一 （可选）在 **access** *access-list* 部分输入一个字符串（不超过 64 字符），指定访问列表的名称
步骤 6	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 7	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息

步骤 8	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

配置 SNMP 通知

Trap 管理器是一台管理站，负责接收和处理 Trap 消息。Trap 是设备在发生了特定事件后生成的系统告警。默认情况下没有定义 Trap 管理器，设备也不会发送 Trap 消息。运行这个版本 Inspur INOS 的设备可以拥有多个 Trap 管理器（数量不限）。

注释： 有多条命令都在命令语法中使用了关键字 **traps**。除非命令中提供了具体选择：Trap 或 Inform，否则关键字 **traps** 就代表 Trap、Inform 或两者。用户可以使用全局配置命令 **snmp-server host**，来指定发送 SNMP 通知的形式是 Trap 还是 Inform。

用户可以使用全局配置命令 **snmp-server enable traps**，与全局配置命令 **snmp-server host** 相结合，来指定主机接收下表中的通知类型。用户可以启用或禁用这些 Trap 消息，并配置一个 Trap 管理器来接收这些 Trap 消息。

注释： 命令 **snmp-server enable traps** 不支持为设备本地认证生成 Trap 消息。

表 74：设备通知类型

通知类型关键字	描述
bridge	生成 STP 桥接 MIB Trap 消息
cluster	当集群配置发生变化时，生成 Trap 消息
config	当 SNMP 配置发生变化时，生成 Trap 消息
copy-config	当 SNMP 副本配置发生变化时，生成 Trap 消息
cpu threshold	允许与 CPU 相关的 Trap 消息
entity	当 SNMP 实体发生变化时，生成 Trap 消息
envmon	生成环境监控 Trap 消息。用户可以启用以下任意或所有环境 Trap 消息：风扇、关机、状态、电源、温度
flash	生成 SNMP FLASH 通知。在设备栈中，用户可以（可选）为 Flash 的插入和移除状态生成通知，当设备栈中的设备被移除或插入（物理移除、断电或重启）时就会生成 Trap 消息
fru-ctrl	生成实体的现场可更换单元（FRU）控制 Trap 消息。在设备栈中，这个 Trap 消息表示设备栈中的设备被插入或移除
hsrp	为热备份路由器协议（HSRP）的变化生成 Trap 消息
ipmulticast	为 IP 组播路由的变化生成 Trap 消息
mac-notification	为 MAC 地址通知生成 Trap 消息
ospf	为最短路径优先（OSPF）的变化生成 Trap 消息。用户可以启用下列任意或全部 Trap 消息：Inspur 指定、错误、链路状态通告、速率限制、重传和状态变化
pim	为协议无关组播（PIM）的变化生成 Trap 消息。用户可以启用下列任意或全部 Trap 消息：不合法的 PIM 消息、邻居变化和汇集点（RP）映射的变化
port-security	生成 SNMP 端口安全 Trap 消息。用户可以以秒为单位设置 Trap 最大

	传输速率。取值范围是 0 至 1000；默认值为 0，表示没有限速。 **注释：** 在使用通知类型 **port-security** 配置 Trap 消息时，用户需要首先配置端口安全特性，然后配置端口安全 Trap 速率： **1. snmp-server enable traps port-security** **2. snmp-server enable traps port-security trap-rate** *rate*
snmp	为 SNMP 类型的通知生成 Trap 消息，其中包括认证、冷启动、热启动、链路 Up 或链路 Down
storm-control	为 SNMP 风暴控制生成 Trap 消息。用户还可以以分钟为单位设置 Trap 最大传输速率。取值范围是 0 至 1000；默认值为 0（没有限制；每当条件满足时都发送 Trap 消息）
stpx	生成 SNMP STP 扩展 MIB Trap 消息
syslog	生成 SNMP 系统日志 Trap 消息
tty	为 TCP 连接生成 Trap 消息。默认这个 Trap 消息是启用的
vlan-mumbership	为 SNMP VLAN 成员关系变化生成 Trap 消息
vlancreate	生成 SNMP VLAN 创建 Trap 消息
vlandelete	生成 SNMP VLAN 删除 Trap 消息
vtp	为 VLAN 干道协议（VTP）的变化生成 Trap 消息

用户可以按照以下步骤，配置设备向主机发送 Trap 或 Inform 消息。

总步骤

1. enable

2. configure terminal

3. snmp-server engineID remote *ip-address engineid-string*

4. snmp-server user *username group-name* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] }

5. snmp-server group *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}} [**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*]

6. snmp-server host *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*]

7. snmp-server enable traps *notification-types*

8. snmp-server trap-source *interface-id*

9. snmp-server queue-length *length*

10. snmp-server trap-timeout *seconds*

11. end

12. show running-config

13. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：**	进入全局配置模式

	Device# **configure terminal**	
步骤 3	**snmp-server engineID remote** *ip-address engineid-string* **示例：** Device(config)# **snmp-server engineID remote 192.180.1.27 00000063000100a1c0b4011b**	为远端主机指定引擎 ID。
步骤 4	**snmp-server user** *username group-name* {**remote** *host* [**udp-port** *port*]} {**v1** [**access** *access-list*] | **v2c** [**access** *access-list*] | **v3** [**encrypted**] [**access** *access-list*] [**auth** {**md5** | **sha**} *auth-password*] } **示例：** Device(config)# **snmp-server user Pat public v2c**	配置 SNMP 用户，与步骤 3 中创建的远端主机相关联。 **注释：** 用户不能先为远端用户配置一个地址，然后再为远端主机配置引擎 ID。否则用户会看到错误消息，并且命令也不会执行
步骤 5	**snmp-server group** *group-name* {**v1** | **v2c** | **v3** {**auth** | **noauth** | **priv**}}[**read** *readview*] [**write** *writeview*] [**notify** *notifyview*] [**access** *access-list*] **示例：** Device(config)# **snmp-server group public v2c access lmnop**	配置 SNMP 组。
步骤 6	**snmp-server host** *host-addr* [**informs** | **traps**] [**version** {**1** | **2c** | **3** {**auth** | **noauth** | **priv**}}] *community-string* [*notification-type*] **示例：** Device(config)# **snmp-server host 203.0.113. comaccess snmp**	指定接收 SNMP Trap 消息的主机。 在 *host-addr* 部分指定主机（目标接收方）名称或 IP 地址。 （可选）指定 **traps**（默认），向主机发送 SNMP Trap 消息 （可选）指定 **informs**，向主机发送 SNMP Inform 消息 （可选）指定 SNMP 版本 **version**（**1**、**2c** 或 **3**）。SNMPv1 不支持发送 Inform 消息 （可选）在配置版本 3 时，用户可以选择安全级别：**auth**、**noauth** 或 **priv** **注释：** 只有当设备装安装了加密软件版本时才能使用关键字 **priv** 在配置 **version 1** 或 **version 2** 时，用户可以在 *community-string* 部分指定作为密码使用的团体字符串，与通知

		消息一起发送。在配置 **version 3** 时，用户可以输入 SNMPv3 用户名。 @符号是用来界定环境信息的。不要在配置这条命令时，在 SNMP 团体字符串中使用@符号。 （可选）在 *notification-type* 部分使用上表中列出的关键字。如果没有指定具体类型，表示发送所有类型的通知
步骤 7	**snmp-server enable traps** *notification-types* **示例：** `Device(config)# snmp-server enable traps snmp`	使设备发送 Trap 或 Inform 消息，并指定发送的通知类型。具体的通知类型见上表，或者使用命令 **snmp-server enable traps ?**。 要想启用多种类型的 Trap 消息，用户必须为每种 Trap 类型单独输入一条 **snmp-server enable traps** 命令。 **注释：** 在使用通知类型 **port-security** 配置 Trap 消息时，用户要首先配置端口安全 Trap，然后配置端口安全 Trap 速率： **1. snmp-server enable traps port-security** **2 snmp-server enable traps port-security trap-rate** *rate*
步骤 8	**snmp-server trap-source** *interface-id* **示例：** `Device(config)# snmp-server trap-source GigabitEthernet1/0/1`	（可选）指定源接口，这个接口为 Trap 消息提供源 IP 地址。这条命令也设置了 Inform 消息的源 IP 地址
步骤 9	**snmp-server queue-length** *length* **示例：** `Device(config)# snmp-server queue-length`	（可选）为每个 Trap 主机建立消息队列长度。取值范围是 1 至 1000；默认值为 10
步骤 10	**snmp-server trap-timeout** *seconds* **示例：** `Device(config)# snmp-server trap-timeout 60`	（可选）定义重新发送 Trap 消息的时间间隔。取值范围是 1 至 1000；默认值为 30 秒
步骤 11	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 12	**show running-config**	检查用户输入的信息

	示例： `Device# show running-config`	
步骤 13	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

接下来做什么？

命令 **snmp-server host** 指定了由哪台主机负责接收通知消息。命令 **snmp-server enable traps** 在全局（为 Trap 和 Inform）为指定通知类型启用了通知特性。要想让主机收到 Inform 消息，用户必须为主机配置命令 **snmp-server host informs**，然后使用命令 **snmp-server enable traps** 在全局启用 Inform 消息。

要想让某台主机不再接收 Trap 消息，用户需要使用全局配置命令 **no snmp-server host** *host*。命令 **no snmp-server host** 中带有关键字 **no**，会为主机禁用 Trap 消息，但 Inform 消息不受影响。要向禁用 Inform 消息，用户需要使用全局配置命令 **no snmp-server host informs**。要想禁用某种 Trap 类型，用户需要使用全局配置命令 **no snmp-server enable traps** *notification-types*。

设置代理联系和位置信息

用户可以按照以下步骤，来设置 SNMP 代理的系统联系和位置信息，用户通过配置文件能够访问这些描述信息。

总步骤

1. enable

2. configure terminal

3. snmp-server contact *text*

4. snmp-server location *text*

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**snmp-server contact** *text*	设置系统联系信息

	示例： `Device(config)# snmp-server contact Dial System Operator at beeper 21555`	
步骤 4	**snmp-server location** *text* **示例：** `Device(config)# snmp-server location Building 3/Room 222`	设置系统位置信息
步骤 5	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

通过 SNMP 限制 TFTP 服务器的使用

用户可以按照以下步骤，通过 SNMP 调用访问列表，限制使用 TFTP 服务器进行保存和加载配置文件的行为。

总步骤

1. enable

2. configure terminal

3. snmp-server tftp-server-list *access-list-number*

4. access-list *access-list-number* {**deny** | **permit**} **source** [*source-wildcard*]

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal**	进入全局配置模式

	示例： Device# **configure terminal**	
步骤 3	**snmp-server tftp-server-list** *access-list-number* 示例： Device(config)# **snmp-server tftp-server-list 44**	通过 SNMP 调用访问列表，来限制使用 TFTP 进行配置文件复制工作。 在 *access-list* 部分输入标准 IP 访问李彪的编号，取值范围是 1 至 99 和 1300 至 1999
步骤 4	**access-list** *access-list-number* {**deny** | **permit**} **source** [*source-wildcard*] 示例： Device(config)# **access-list 44 permit 10.1.1.2**	创建标准访问列表，用户可以按照需要多次重复配置这条命令。 在 *access-list-number* 部分输入步骤 3 中指定的访问列表编号。 关键字 **deny** 会在条件匹配时拒绝访问。关键字 **permit** 会在条件匹配时放行访问。 在 *source* 部分输入能够访问设备的 TFTP 服务器的 IP 地址。 （可选）在 *source-wildcard* 部分以点分十进制格式，输入与源相匹配的通配符比特。把希望在对比中忽略的二进制位设置为 1。 要记住，所有访问列表的末尾都有隐含拒绝所有数据包的语句
步骤 5	**end** 示例： Device(config)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config** 示例： Device# **show running-config**	检查用户输入的信息
步骤 7	**copy running-config startup-config** 示例： Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

为 SNMP 配置 Trap 标记

总步骤

1. configure terminal

2. trapflags ap { **interfaceup** | **register**}

3. trapflags client {**dot11** | **excluded**}
4. trapflags dot11-security {**ids-sig-attack** | **wep-decrypt-error**}
5. trapflags mesh
6. trapflags rogueap
7. trapflags rrm-params {**channels** | **tx-power**}
8. trapflags rrm-profile {**coverage** | **interference** | **load** | **noise**}
9. end

具体配置

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 2	**trapflags ap** { **interfaceup**	**register**} **示例：** `Device(config)# trapflags ap interfaceup`	允许发送与 AP 相关的 Trap 消息。在这条命令前添加关键字 **no** 来禁用 Trap 标记。 • **interfaceup**——当 Inspur AP 接口（A 或 B）变为 Up 状态时发送 Trap 消息 • **register**——当 Inspur AP 上注册了 Inspur 设备时发送 Trap 消息
步骤 3	**trapflags client** {**dot11**	**excluded**} **示例：** `Device(config)# trapflags client excluded`	允许发送与客户端相关的 802.11 Trap 消息。在这条命令前添加关键字 **no** 来禁用 Trap 标记。 • **dot11**——为客户端启用 802.11 Trap 消息 • **excluded**——为客户端启用拒绝 Trap 消息
步骤 4	**trapflags dot11-security** {**ids-sig-attack**	**wep-decrypt-error**} **提示：** `Device(config)# trapflags dot11-security wep-decrypt-error`	启用与 802.11 安全相关的 Trap 消息。在这条命令前添加关键字 **no** 来禁用 Trap 标记。 • **ids-sig-attack**——启用 IDS 签名攻击 Trap 消息 • **wep-decrypt-error**——为客户端启用 WEP 解密错误 Trap 消息
步骤 5	**trapflags mesh** **示例：** `Device(config)# trapflags mesh`	为 Mesh 启用 Trap 消息。在这条命令前添加关键字 **no** 来禁用 Trap 标记。
步骤 6	**trapflags rogueap** **示例：** `Device(config)# trapflags`	为 Rogue AP 检测启用 Trap 消息。在这条命令前添加关键字 **no** 来禁用 Trap 标记。

	rogueap	
步骤 7	**trapflags rrm-params {channels | tx-power}** **示例：** `Device(config)# trapflags rrm-params tx-power`	启用与更新相关的 RRM 参数 Trap 消息。在这条命令前添加关键字 **no** 来禁用 Trap 标记。 • **channels——**当 RF 管理器自动为 Inspur AP 接口更改新到号码时发送 Trap 消息 • **tx-power——**当时 RF 管理器自动为 Inspur AP 接口更改发送功率时发送 Trap 消息
步骤 8	**trapflags rrm-profile {coverage | interference | load | noise}** **注释：** `Device(config)# trapflags rrm-profile interference`	启用与 RRM 配置文件相关的 Trap 消息。在这条命令前添加关键字 **no** 来禁用 Trap 标记。 • **coverage**——当 RF 管理器中维护的范围配置文件失效时发送 Trap 消息 • **interference**——当 FR 管理器中维护的干扰配置文件失效时发送 Trap 消息 • **load**——当 FR 管理器中维护的加载配置文件失效时发送 Trap 消息 • **noise**——当 FR 管理器中维护的噪声配置文件失效时发送 Trap 消息
步骤 9	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式

监控 SNMP 状态

要想查看 SNMP 的输入和输出状态统计信息，其中包括用户输入的错误团体字符串次数、错误和被请求过的变量，用户需要使用特权 EXEC 命令 **show snmp**。用户也可以使用下面这个表格中列出的其他特权 EXEC 命令来查看 SNMP 信息。

表 75：显示 SNMP 信息的命令

命令	目的
show snmp	显示 SNMP 的统计状态信息
	显示设备上配置的本地 SNMP 引擎和所有远端引擎信息
show snmp group	显示网络中每个 SNMP 组的信息
show snmp pending	显示暂缓处理的 SNMP 请求信息
show snmp sessions	显示当前的 SNMP 会话信息

show snmp user	显示 SNMP 用户表中每个 SNMP 用户名称的信息。 **注释：** 用户必须使用这条命令来查看 **auth** | **noauth** | **priv** 模式的 SNMPv3 配置信息。命令 **show running-config** 中不会显示这些信息

SNMP 示例

下面这个示例中展示了如何启用所有版本的 SNMP。这个配置允许任意 SNMP 管理器使用团体字符串 *public*，以只读权限访问所有对象。这个配置并不会使设备发送任何 Trap 消息。

```
Device(config)# snmp-server community public
```

下面这个示例展示了如何通过配置，允许任意 SNMP 管理器使用团体字符串 *public*，以只读权限访问所有对象。设备还会使用 SNMPv1 向主机 192.180.1.111 和 192.180.1.33 发送 VTP Trap 消息，使用 SNMPv2C 向主机 192.180.1.27 发送 VTP Trap 消息。团体字符串 *public* 也随 Trap 消息发送。

```
Device(config)# snmp-server community public
Device(config)# snmp-server enable traps vtp
Device(config)# snmp-server host 192.180.1.27 version 2c public
Device(config)# snmp-server host 192.180.1.111 version 1 public
Device(config)# snmp-server host 192.180.1.33 public
```

下面这个示例展示了如何通过配置，允许访问列表 4 中指定的成员使用团体字符串 *comaccess*，以只读的权限访问所有对象。其他 SNMP 管理器不能访问任何对象。设备会使用团体字符串 *public*，以 SNMPv2C 向主机 *icntnetworks.com* 发送 SNMP 认证失败 Trap 消息。

```
Device(config)# snmp-server community comaccess ro 4
Device(config)# snmp-server enable traps snmp authentication
Device(config)# snmp-server host icntnetworks.com version 2c public
```

下面这个示例展示了如何通过配置，使设备向主机 *icntnetworks.com* 发送实体（Entity）MIB Trap 消息。团体字符串是 *restricted*。第一条命令会使设备发送实体（Entity）MIB Trap 消息，以及之前启用的 Trap 消息。第二条命令指定了这些 Trap 消息的目的地，如果没有指定的话，会使用以前为主机 *icntnetworks.com* 设置的 **snmp-server host** 命令。

```
Device(config)# snmp-server enable traps entity
Device(config)# snmp-server host icntnetworks.com restricted entity
```

下面这个示例展示了如何通过配置，让设备使用团体字符串 *public* 向主机 *myhost.icntnetworks.com* 发送所有 Trap 消息：

```
Device(config)# snmp-server enable traps
Device(config)# snmp-server host myhost.icntnetworks.com public
```

下面这个示例展示了如何通过配置，把用户与远端主机相关联，并在用户进入全局配置模式时，让设备发送 **auth**（有认证无加密）认证级别的 Inform 消息：

```
Device(config)# snmp-server engineID remote 192.180.1.27
00000063000100a1c0b4011b
Device(config)# snmp-server groupauth group v3 auth
Device(config)# snmp-server user authuser authgroup remote
```

```
192.180.1.27 v3 auth md5 mypassword
Device(config)# snmp-server user authuser authgroup v3 auth md5 mypassword
Device(config)# snmp-server host 192.180.1.27 informs version 3 auth authuser config
Device(config)# snmp-server enable traps
Device(config)# snmp-server inform retries 0
```

其他参考资料

相关文档

相关主题	文档名称
SNMP 命令	*Network Management Command Reference, Inspur INOS*

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

简单网络管理协议的特性历史与信息

版本	变更
Inspur INOS 12.2	引入该特性

配置 SPAN 和 RSPAN

查询特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

实施 SPAN 和 RSPAN 的先决条件

SPAN

- 用户可以使用关键字 **filter vlan**，把 SPAN 流量限制在指定 VLAN 中。如果监控的是 Trunk 端口，只有通过这个关键字指定的 VLAN 流量才会被监控。默认情况下 Trunk 端口上的所有 VLAN 都会受到监控。

RSPAN

- 建议用户先配置 RSPAN VLAN，再配置 RSPAN 源或目的会话。

实施 SPAN 和 RSPAN 的限制条件

SPAN

实施 SPAN 有以下限制条件：

- 在每台设备上，用户可以配置 66 个会话。最多可以配置 8 个源会话，其他会话可以配置为 RSPAN 目的会话。源会话既可以是本地 SPAN 会话，也可以是 RSPAN 源会话；
- 对于 SPAN 源来说，用户可以在一个会话中监控单个端口或 VLAN 的流量，也可以监控一系列端口/VLAN 或一个端口/VLAN 范围的流量。用户不能在一个 SPAN 会话中同时监控源端口和源 VLAN 的流量；
- 目的端口不能再充当源端口；源端口也不能再充当目的端口；
- 用户可以使用相同的目的端口配置两个 SPAN 会话；
- 当用户把一个设备端口配置为 SPAN 目的端口后，这个端口就不再是普通的设备端口了；只有受监控的流量会通过这个 SPAN 目的端口；
- 输入 SPAN 配置命令并不会删除之前配置的 SPAN 参数。用户必须使用全局配置模式的命令 **no monitor session** {*session_number* | **all** | **local** | **remote**}，才能删除配置的 SPAN 参数；
- 对于本地 SPAN 来说，如果用户设置了关键字 **encapsulation replicate**，那么从 SPAN 目的端口离开的出站数据包会携带原始的封装头部——未打标、ISL 或 IEEE 802.1Q。如果没有指定这个关键字，数据包会以端口本地的形式进行发送；
- 用户可以把禁用（Disabled）状态的端口配置为源端口或目的端口，但只有当目的端口和至少一个源端口/VLAN 启用后，SPAN 功能才会生效；
- 用户不能在单个 SPAN 会话中同时监控源 VLAN 和过滤 VLAN。

在一个 SPAN 会话中监控的流量具有以下限制条件：

- 源可以是端口或 VLAN，但不能在同一个会话中同时监控源端口和源 VLAN；
- 当用户启用了出向 SPAN 会话时，Wireshark 无法捕获出向数据包；
- 用户可以在同一台设备或同一个设备堆栈中，同时运行本地 SPAN 和 RSPAN 源会话。设备或设备堆栈一共支持 66 个源和 RSPAN 目的会话；
- 用户可以在两个不同的 SPAN 或 RSPAN 源会话中，配置不同的或重叠的 SPAN 源端口和源 VLAN。交换端口和路由端口都可以被配置为 SPAN 源和目的；
- 用户在一个 SPAN 会话中可以设置多个目的端口，但在一个设备堆栈中最多有 64 个目的端口；
- SPAN 会话不会影响设备的正常操作。但超额预订的 SPAN 目的可能会导致尾部丢弃或数据包丢失，比如通过 10 Mbit/s 端口监控 100 Mbit/s 端口；
- 当启用了 SPAN 或 RSPAN 后，每个受监控的数据包都会被发送两次，一次作为普通流量发送，一次作为监控数据包发送。监控大量端口或 VLAN 会在不知不觉中生成大量网络流量；
- 用户可以在禁用（Disabled）状态的端口上配置 SPAN 会话，但只有当目的端口和至少一个源端口/VLAN 启用后，这个 SPAN 会话才会生效；
- 设备不支持在单个会话中结合本地 SPAN 和 RSPAN：
  - RSPAN 源会话中不能有本地目的端口；
  - RSPAN 目的会话中不能有本地源端口；

- 使用了相同 RSPAN VLAN 的 RSPAN 目的会话和 RSPAN 源会话不能运行在同一台设备或同一个设备堆栈中。

RSPAN

实施 RSPAN 有以下限制条件：

- RSPAN 不支持 BPDU 数据包监控或其他二层设备协议；
- RSPAN VLAN 只能配置在 Trunk 端口上，不能配置在 Access 端口上。为了避免在 RSPAN VLAN 中生成不必要的流量，用户要确保所有参与设备都支持 VLAN remote-vlan（远端 VLAN）特性；
- 当源 Trunk 端口上有活跃的 RSPAN VLAN 时，RSPAN VLAN 是作为源，包含在基于端口的 RSPAN 会话中的。RSPAN VLAN 也可以作为 SPAN 会话中的源。但由于设备不监控 SPAN 流量，因此它也不能把任何 RSPAN VLAN 中的出向 SPAN 数据包，识别为这台设备上 RSPAN 源会话的目的；
- 如果用户启用了 VTP 和 VTP 修剪（Pruning）特性，Trunk 中的 RSPAN 流量会被修剪掉，以防止 VLAN ID 小于 1005 的无用 RSPAN 流量穿越网络；
- 要想使用 RSPAN，交换机必须运行 LAN Base 镜像；

有关 SPAN 和 RSPAN 的信息

SPAN 和 RSPAN

用户可以通过使用 SPAN 或 RSPAN 来分析穿越端口或 VLAN 的网络流量，SPAN 或 RSPAN 会把这些网络流量发送到设备上的另一个端口，或者发送到另一台设备上的端口，这个端口上连接着网络分析设备，或者其他监控或安全设备。SPAN 会复制（镜像）源端口或源 VLAN 上收到或发出（或者双向）的流量，并把这些流量发送到目的端口进行分析。SPAN 不会影响源端口或源 VLAN 上的网络流量交换。用户必须把目的端口专用于 SPAN 特性。除了 SPAN 或 RSPAN 会话要求的流量外，目的端口不会接收或转发其他任何流量。

只有进入或离开源端口的流量，或者进入或离开源 VLAN 的流量，才能使用 SPAN 进行监控；被路由到源 VLAN 的流量不会被监控。举例来说，如果用户监控了入站流量，那些从其他 VLAN 通过路由进入源 VLAN 的流量并不会受到监控；但源 VLAN 上收到的流量，以及从源 VLAN 被路由到其他 VLAN 的流量会受到监控。

用户可以使用 SPAN 或 RSPAN 目的端口，从网络安全设备向网络中注入流量。举例来说，如果用户在目的端口上连接了 Inspur 入侵检测系统（IDS）传感器应用，那么 IDS 设备可以通过发送 TCP 重置数据包，来中断嫌疑攻击者的 TCP 会话。

本地 SPAN

本地 SPAN 支持在一台设备中部署 SPAN 会话；也就是所有源端口或源 VLAN，以及目的端口都在同一台设备或设备堆栈中。本地 SPAN 会从属于任意 VLAN 中的一个或多个源端口，或者从一个或多个 VLAN 复制流量，并将其发送到目的端口进行分析。

端口 5（源端口）的所有流量都会被镜像发送到端口 10（目的端口）。连接在端口 10 上的网络分析设备能够收到端口 5 的所有网络流量，而它却无需在物理上与端口 5 相连。

图 77：单台设备上实施本地 SPAN 配置的示例

Port 5 traffic mirrored on Port 10	端口 5 的流量被 镜像到端口 10
Network analyzer	网络分析设备

下图为设备堆栈中实施本地 SPAN 的示例，其中源和目的端口位于不同的堆栈成员上。

图 78：在设备堆栈中实施本地 SPAN 配置的示例

Switch stack	交换机堆栈
Switch 1	交换机 1
Port 4 on switch 1 in the stack mirrored on port 15 on switch 2	堆栈中交换机 1 上的端口 4 把 流量镜像到交换机 2 上的端口 15
Stackwise Plus port connections	智能堆叠 端口连接
Switch 2	交换机 2
Switch 3	交换机 3
Network analyzer	网络分析设备

远端 SPAN

RSPAN 能够支持源端口、源 VLAN 和目的端口分别位于不同的设备上（不同的设备堆栈上），用户可以在网络中的多台设备上启用远端监控。

下图中展示出设备 A 和设备 B 上配置了源端口。用户指定的 RSPAN VLAN 用来承载每个 RSPAN 会话的流量，这个 VLAN 专门用来承载所有参与设备的 RSPAN 会话流量。从源端口或源 VLAN 来的 RSPAN 流量会被复制到 RSPAN VLAN 中，并通过包含有 RSPAN VLAN 的 Trunk 端口把这些流量转发到监控这个 RSPAN VLAN 的目的会话。每个 RSPAN 源设备必须以端口或 VLAN 作为 RSPAN 源。目的总是物理端口，比如图中设备 C 上的目的端口。

图 79：RSPAN 配置的示例

RSPAN destination ports	RSPAN 目的端口
Switch C	交换机 C
RSPAN destination session	RSPAN 目的会话
Intermediate switches must support RSPAN VLAN	中间的交换机 必须支持 RSPAN VLAN
Switch A	交换机 A
Switch B	交换机 B
RSPAN source session A	RSPAN 源会话 A

RSPAN source ports	RSPAN 源端口
RSPAN source session B	RSPAN 源会话 B
RSPAN source ports	RSPAN 源端口

SPAN 和 RSPAN 的概念和术语

- SPAN 会话
- 受监控流量
- 源端口
- 源 VLAN
- VLAN 过滤
- 目的端口
- RSPAN VLAN

SPAN 会话

用户使用（本地或远端）SPAN 会话能够监控一个或多个端口、一个或多个 VLAN 的流量，并把受监控流量发送到一个或多个目的端口。

本地 SPAN 会话就是一个目的端口和源端口/源 VLAN 的关联组合，它们都位于同一台网络设备上。本地 SPAN 并没有相互分离的源和目的会话。本地 SPAN 会话会根据用户的指定，把一组入向和出向数据包集合在一起，并把它们汇集到一个 SPAN 数据流中，这个数据流最终会被转发到目的端口。

RSPAN 由至少一个 RSPAN 源会话、一个 RSPAN VLAN，以及至少一个 RSPAN 目的会话构成。用户可以分别在不同的网络设备上配置 RSPAN 源会话和 RSPAN 目的会话。为了在设备上配置 RSPAN 源会话，用户会把一组源端口或源 VLAN 与一个 RSPAN VLAN 关联在一起。这个会话的输出信息就是 SPAN 数据包流，这些数据会被发送到 RSPAN VLAN 中。为了在另一台设备上配置 RSPAN 目的会话，用户会把目的端口与 RSPAN VLAN 关联在一起。目的会话会收集所有 RSPAN VLAN 流量并将其发送到 RSPAN 目的端口。

RSPAN 源会话与本地 SPAN 会话非常类似，只不过数据包流发往的目的地有所不同。在 RSPAN 源会话中，SPAN 数据包会被重新标记为 RSPAN VLAN ID，并通过普通的 Trunk 端口被转发到目的设备。

RSPAN 目的会话会收集 RSPAN VLAN 中收到的所有数据包，剥除掉 VLAN 标记，并发送到目的端口。目的会话会为用户提供所有 RSPAN VLAN 数据包的副本（除了二层控制数据包），以便进行分析。

一个拥有多个源和目的端口的 RSPAN 会话可以同在一个会话中，但不能与多个相同远端 VLAN 中的源会话同在一个会话中。

SPAN 会话中的流量监控有下列限制条件：

- 源可以是端口或 VLAN，但用户不能把源端口和源 VLAN 设置在同一个会话中；
- 用户可以在一台设备或设备堆栈中同时运行本地 SPAN 和 RSPAN。设备或设备堆栈总共

支持 66 个源和 RSPAN 目的会话；

- 用户可以在两个不同的 SPAN 或 RSPAN 源会话中，配置不同的或重叠的 SPAN 源端口和源 VLAN。交换端口和路由端口都可以被配置为 SPAN 源和目的；
- 用户在一个 SPAN 会话中可以设置多个目的端口，但在一个设备堆栈中最多有 64 个目的端口；
- SPAN 会话不会影响设备的正常操作。但超额预订的 SPAN 目的可能会导致尾部丢弃或数据包丢失，比如通过 10 Mbit/s 端口监控 100 Mbit/s 端口；
- 当启用了 SPAN 或 RSPAN 后，每个受监控的数据包都会被发送两次，一次作为普通流量发送，一次作为监控数据包发送。监控大量端口或 VLAN 会在不知不觉中生成大量网络流量；
- 用户可以在禁用（Disabled）状态的端口上配置 SPAN 会话，但只有当目的端口和至少一个源端口/VLAN 启用后，这个 SPAN 会话才会生效；
- 设备不支持在单个会话中结合本地 SPAN 和 RSPAN：
  - RSPAN 源会话中不能有本地目的端口；
  - RSPAN 目的会话中不能有本地源端口；
  - 使用了相同 RSPAN VLAN 的 RSPAN 目的会话和 RSPAN 源会话不能运行在同一台设备或同一个设备堆栈中。

受监控流量

SPAN 会话可以监控以下流量类型：

- 接收（Rx）SPAN——接收（或入向）SPAN 能够监控源接口或源 VLAN 收到的所有数据包，并且这些数据包都是没有经过设备修改或处理的。源会收到这些数据包的副本，并将其发送到这个 SPAN 会话的目的端口。
  需要由路由或服务质量（QoS）工具进行修改的数据包是在修改前被复制的，比如修改差分服务代码点（DSCP）。
  会在服务处理期间造成数据包丢弃的特性并不会对入向 SPAN 产生什么影响；即使实际入站的数据包已被丢弃，目的端口也会收到该数据包的副本。这些特性包括 IP 标准和扩展入向访问控制列表（ACL）、入向 QoS 策略、VLAN ACL 和出向 QoS 策略；
- 传输（Tx）SPAN——传输（或出向）SPAN 能够监控源接口发送的所有数据包，并且这些数据包都经过了设备的修改和处理。源发出的所有数据包副本都会被发送到该 SPAN 会话的目的端口。副本是复制的修改后的数据包。
  由路由功能（比如修改的生存时间[TTL]、MAC 地址或 QoS 值）修改后的数据包会被复制（带有修改后的值）给目的端口。
  会导致数据包在传输处理过程中被丢弃的特性也会影响为这个 SPAN 实施的数据包复制工作。这些特性包括 IP 标准和扩展出向 ACL，以及出向 QoS 策略。
- 双向——在 SPAN 会话中，用户也可以监控端口或 VLAN 接收和发送数据包的情况。这是默认设置。

本地 SPAN 会话的默认配置是以未打标的方式发送所有数据包。但当用户在配置目的端口是使用了关键字 **encapsulation replicate**，会发生以下变化：

- 数据包在被发送到目的端口时，还携带源端口为其封装的信息（未打标或打上 IEEE 802.1Q 标记）；
- 所有类型的数据包都会被监控，其中包括 BPDU 和二层协议数据包。

因此启用了封装复制（encapsulation replicate）的本地 SPAN 会话会在向目的端口发送的数据

包中包含未打标的数据包和携带 IEEE 802.1Q 标记的数据包。
设备上的拥塞会导致数据包被丢弃，其中丢弃位置包括入向源端口、出向源端口，或 SPAN 目的端口。一般来说，这些特征之间相互是独立的。举例来说：

- 数据包可能会被正常穿法，但由于 SPAN 目的端口超额订阅的关系，监控数据包会被丢弃；
- 如像数据包可能会在正常转发过程中被丢弃，但仍可能会被转发到 SPAN 目的端口；
- 由于设备拥塞而被丢弃的出向数据包，也会被出向 SPAN 丢弃。

在一些 SPAN 配置中，同一个源数据包会由多个副本被发送到 SPAN 目的端口。比如设备上配置了双向（Rx 和 Tx）SPAN 会话，接收（Rx）会话监控端口 A，发送（Tx）会话监控端口 B。如果一个数据包从端口 A 进入设备，并被交换到端口 B，那么入站和出站数据包都会被发送到目的端口。这两个数据包是相同的，除非三层信息被重写，那样的话就会由于数据包更改行为使数据包发生变化。

源端口

源端口（也称为受监控端口）可以是交换端口，也可以是路由端口，用户监控这个端口的行为来进行网络流量分析。在本地 SPAN 会话或 RSPAN 会话中，用户可以在一个方向上，或者在双方向上监控源端口或源 VLAN。设备支持任意数量的源端口（最大数量为设备上可用端口的数量）以及任意数量的源 VLAN（最大数量为设备上支持的 VLAN 数量）。虽然设备支持在本地 SPAN 或 RSPAN 中配置多个源端口或源 VLAN，但用户不能在单个会话中同时设置端口和 VLAN。
源端口拥有以下特征：

- 它可以由多个 SPAN 会话进行监控；
- 每个源端口都可以配置一个方向（入向、出向，或双向）进行监控；
- 它可以是任意端口类型（比如 EtherChannel、千兆以太网接口等）；
- 对于 EtherChannel 源，用户可以监控整个 EtherChannel，也可以只监控某个参与这个 Port-Channel 的物理端口；
- 它可以是 Access 端口、Trunk 端口、路由端口或语音 VLAN 端口；
- 它不能是目的端口；
- 源端口可以属于相同的 VLAN，或者属于不同的 VLAN；
- 用户可以在一个会话中监控多个源端口。

源 VLAN

基于 VLAN 的 SPAN（VSPAN）负责监控一个或多个 VLAN 的网络流量。VSPAN 中的 SPAN 或 RSPAN 源接口是 VLAN ID，并且 VSPAN 会监控这个 VLAN 中所有端口的流量。
VSPAN 拥有以下特征：

- 源 VLAN 中的所有活跃端口都包含在源端口中，都可以受到单向或双向监控；
- 对于某个端口来说，只有受监控 VLAN 的流量会被发送给目的端口；
- 如果目的端口属于一个源 VLAN，它自己会被排除在源列表之外，不受监控；
- 如果端口被添加到源 VLAN 后，或者从源 VLAN 中移除后，这些端口从源 VLAN 上收到的流量会被添加到监控源，或者从监控源移除；
- 用户不能在以某个 VLAN 为源的会话中，过滤这个 VLAN 的流量；

- 用户可以只监控以太网 VLAN。

VLAN 过滤

当用户把 Trunk 端口作为源端口时，默认情况下，Trunk 上所有活跃的 VLAN 都会被监控。用户可以限制在这个 Trunk 源端口上监控的 SPAN 流量，也就是使用 VLAN 过滤特性来指定 VLAN。

- VLAN 过滤特性只能应用在 Trunk 端口上，或者应用在语音 VLAN 端口上；
- VLAN 过滤特性只能应用在基于端口的会话上，不能将其应用在以 VLAN 为源的会话上；
- 当用户指定了 VLAN 过滤表后，在 Trunk 端口或语音 VLAN Access 端口上，只有列表中的这些 VLAN 会被监控；
- 其他端口类型发来的 SPAN 流量不会受到 VLAN 过滤特性的影响；也就是说其他端口上允许传输所有 VLAN 的流量；
- VLAN 过滤特性只会影响被转发到目的 SPAN 端口的流量，而不会影响正常流量的交换行为。

目的端口

每个本地 SPAN 会话或 RSPAN 目的会话都必须有一个目的端口（也称为监控端口），它负责从源端口或源 VLAN 接收流量副本，并把这些 SPAN 数据包发送给用户，通常也就是网络分析设备。

目的端口拥有以下特征：

- 对于本地 SPAN 会话来说，目的端口必须与源端口位于相同的设备或设备堆栈上。对于 RSPAN 会话来说，目的端口位于配置了 RSPAN 目的会话的设备上。在只运行了 RSPAN 源会话的设备或设备堆栈上，没有目的端口；
- 但该用户把一个端口配置为 SPAN 目的端口后，这个配置会覆盖原始的端口配置。当 SPAN 目的端口的配置被移除后，端口会恢复到自己之前的配置。如果一个端口仍是 SPAN 目的端口，那么用户对它所做的配置变更并不会生效，直到用户删除 SPAN 目的配置为止；

注释： 在 SPAN 目的端口上配置 QoS 时，QoS 会立即生效。

- 如果端口属于一个 EtherChannel 组，当用户把它配置为目的端口时，它就从 EtherChannel 组中移除了。如果它是路由端口，被配置为目的端口后，它就不再是路由端口了；
- 它可以是任意以太网物理端口；
- 它不能是安全端口；
- 它不能是源端口；
- 它可以是 EtherChannel 组（仅限于 **ON** 模式）；
- 它不能是 VLAN；
- 它同时只能参与一个 SPAN 会话（一个 SPAN 会话中的目的端口不能充当另一个 SPAN 会话中的目的端口）；
- 当它是活跃状态时，入站流量会被禁用。除了 SPAN 会话所需流量外，端口并不传输任何其他流量。目的端口上不会学到或转发任何入站流量；
- 如果用户为网络安全设备启用了入站流量转发，那么目的端口会在二层转发这些流量；
- 它不参与任何二层协议（STP、VTP、CDP、DTP、PagP）；

- 属于任意 SPAN 会话中源 VLAN 的目的端口会被排除在源列表之外，不会受到监控；
- 一台设备或一个设备堆栈中，目的端口的最大数量是 64。

本地 SPAN 和 RSPAN 目的端口的功能，在 VLAN 标记和封装上有所不同：

- 对于本地 SPAN 来说，如果用户设置了关键字 **encapsulation replicate**，那么从 SPAN 目的端口离开的出站数据包会携带原始的封装头部（未打标、ISL 或 IEEE 802.1Q）。如果没有指定这个关键字，数据包会以未打标的形式进行发送。因此对于启用了 **encapsulation replicate** 的本地 SPAN 会话来说，它的输出内容中会包含未打标、标记 ISL 或标记 IEEE 802.1Q 的数据包；
- 对于 RSPAN 来说，原始的 VLAN ID 会被 RSPAN VLAN ID 覆盖。因此目的端口上的所有数据包都是未打标的。

RSPAN VLAN

RSPAN VLAN 承载着 RSPAN 源和目的会话之间的 SPAN 流量。RSPAN VLAN 拥有以下特性：

- RSPAN VLAN 中的所有流量总是泛洪的；
- RSPAN VLAN 中没有 MAC 地址学习行为；
- RSPAN VLAN 流量只会在 Trunk 端口上传输；
- 用户必须在 VLAN 配置模式中，使用 VLAN 配置模式的命令 **remote-span**，来配置 RSPAN VLAN；
- STP 可以运行在 RSPAN VLAN Trunk 上，但不能运行在 SPAN 目的端口上；
- RSPAN VLAN 不能是私有 VLAN、主用或备用 VLAN。

对于 VLAN 1 至 1005 这些 VLAN Trunk 协议（VTP）能够识别的 VLAN 来说，VLAN ID 及其相关联的 RSPAN 特征都会由 VTP 进行传播。如果用户使用扩展 VLAN 范围（1006 至 4094）分配 RSPAN VLAN ID 的话，用户必须手动配置中间的所有设备。

网络中可能同时会有多个 RSPAN VLAN，每个 RSPAN VLAN 定义了一个网络范围内的 RSPAN 会话。也就是说，位于网络中任何位置的多个 RSPAN 源会话都可以为 RSPAN 会话提供数据包。网络中也可能会有多个 RSPAN 目的会话，监控相同的 RSPAN VLAN，并且为用户提供流量。RSPAN VLAN ID 用来区分这些会话。

SPAN 和 RSPAN 与其他特性的交互

SPAN 接口可以与以下特性进行交互：

- 路由——SPAN 不监控路由流量。VSPAN 只监控进入或离开设备的流量，不监控在 VLAN 间路由的流量。举例来说，如果用户配置了监控一个 VLAN 的接收（Rx）方向，那么当设备把其他 VLAN 的流量路由到受监控 VLAN 中时，这些流量并不会受到监控，SPAN 目的端口上也不会收到这些流量；
- STP——对于目的端口来说，当它的 SPAN 或 RSPAN 会话是活跃状态时，它不参与 STP。目的端口可以在 SPAN 或 RSPAN 会话禁用后参与 STP。对于源端口来说，SPAN 并不影响它的 STP 状态。承载 RSPAN VLAN 的 Trunk 端口上可以使用 STP；
- CDP——对于 SPAN 目的端口来说，当它的 SPAN 会话是活跃状态时，它不参与 CDP。目的端口可以在 SPAN 会话禁用后重新参与 STP；
- VTP——用户可以使用 VTP 来修剪设备之间的 RSPAN VLAN；
- VLAN 和 Trunk 技术——用户可以随时为源端口或目的端口修改 VLAN 成员关系或 Trunk

设置。但改变目的端口的 VLAN 成员关系或 Trunk 设置后，设置并不会马上生效，需要用户先移除 SPAN 目的端口配置后才会生效。为源端口改变 VLAN 成员关系或 Trunk 设置后，设置会马上生效，并且相应的 SPAN 会话会自动进行调整；

- EtherChannel——用户可以把 EtherChannel 组设置为源端口或 SPAN 目的端口。当把 EtherChannel 组配置为 SPAN 源时，整个组都会受到监控。
  如果用户把物理端口添加到一个受监控的 EtherChannel 组中，这个新加入的端口也会被放入 SPAN 源端口列表中。如果用户把一个端口从受监控的 EtherChannel 组中移除，它也会自动从源端口列表中被删除。
  用户可以把属于 EtherChannel 组的物理端口配置为 SPAN 源端口，并且它仍为 EtherChannel 的一部分。在这种情况中，当这个物理参与 EtherChannel 中的数据转发时，来自于它的流量会受到监控。但是如果物理端口所属的 EtherChannel 组被指定为 SPAN 目的，那么它就会从这个组中移除。在端口从 SPAN 会话中移除后，它会重新加入 EtherChannel 组。端口从 EtherChannel 组中移除后，仍保留这个组的成员，但它们都处于非活跃或抑制状态。
  如果属于 EtherChannel 组的物理端口是目的端口，而 EtherChannel 组是源，那么端口会从 EtherChannel 组中移除，并且也会从受监控端口的列表中移除。
- 组播流量也可以受到监控。对于出向和入向端口监控来说，只有单个未经编辑的数据包会被发送到 SPAN 目的端口。它并不会影响组播数据包发送的次数；
- 私有 VLAN 端口不能充当 SPAN 目的端口；
- 安全端口不能充当 SPAN 目的端口；
  对于 SPAN 会话来说，当在目的端口上启用了入向转发时，用户不能在受监控的出向端口上启用端口安全特性。对于 RSPAN 源会话来说，用户不能在任何受监控的出向端口上启用端口安全特性。
- IEEE 802.1x 端口可以是 SPAN 源端口。用户可以在 SPAN 目的端口上启用 IEEE 802.1x；但当用户把该端口的 SPAN 目的配置移除时，IEEE 802.1x 也会被禁用。
  对于 SPAN 会话来说，当在目的端口上启用了入向转发时，用户不能在受监控的出向端口上启用 IEEE 802.1x 特性。对于 RSPAN 源会话来说，用户不能在任何受监控的出向端口上启用 IEEE 802.1x 特性。

SPAN 和 RSPAN 以及设备堆栈

由于多台设备的堆栈表现为一台逻辑设备，因此本地 SPAN 源端口和目的端口可以分别位于堆栈中的不同设备上。因此在堆栈中添加或删除设备的操作，会影响本地 SPAN 会话，也会影响 RSPAN 源会话或目的会话。当用户从堆栈中移除一台设备后，一个活跃的会话可能变为非活跃状态；当用户向堆栈中添加一台设备后，一个非活跃的会话可能变为活跃状态。

基于流的 SPAN

用户可以通过使用基于流的 SPAN（FSPAN）或基于流的 RSPAN（FRSPAN），来控制 SPAN 或 RSPAN 会话中监控的网络流量类型，这两项特性可以在源端口的受监控流量上应用访问控制列表（ACL）。FSPAN ACL 可以用来过滤受监控的 IPv4、IPv6 和非 IP 流量。
用户可以通过接口，向 SPAN 会话应用 ACL。这个 ACL 会应用到这个 SPAN 会话中所有接口上的所有受监控流量上。ACL 中允许的数据包会被复制给 SPAN 目的端口。其他数据包不会

被复制给 SPAN 目的端口。
原始流量会继续进行转发，并且与其相关联的端口 ACL、VLAN ACL 和路由器 ACL 也都会进行应用。FSPAN ACL 不会对转发决策造成任何影响。类似的，端口 ACL、VLAN ACL 和路由器 ACL 也不会对流量监控行为带来任何影响。如果安全入向 ACL 拒绝了一个数据包，这个数据包就不会继续被转发出去，但如果 FSPAN ACL 中允许这个数据包的话，它仍会被发送到 SPAN 目的端口。但如果安全出向 ACL 拒绝了数据包而导致数据包没有被发送出去，那么这个数据包也不会被复制到 SPAN 目的端口。然而，如果安全出向 ACL 允许数据包被发送出去，那么这个数据包也只有当 FSPAN ACL 允许的情况下，才会被复制给 SPAN 目的端口。上述规则对于 RSPAN 会话来说也适用。
用户可以在 SPAN 会话上配置三种类型的 FSPAN ACL：

- IPv4 FSPAN ACL——只过滤 IPv4 数据包
- IPv6 FSPAN ACL——只过滤 IPv6 数据包
- MAC FSPAN ACL——只过滤非 IP 数据包

如果用户在一个堆栈上配置了基于 VLAN 的 FSPAN 会话，但它不适用于一台或多台设备上的硬件内存，那么这些设备就不会加载这个会话，并且从这些设备发出的匹配这个 FSPAN ACL 的流量也不会被复制到 SPAN 目的端口上。FSPAN ACL 仍会继续发挥作用，在那些它能够使用的硬件内存中，相关流量会被复制到 SPAN 目的端口。
当用户关联了一个空的 FSPAN ACL 时，有些硬件功能会由于这个 ACL，把所有流量都复制给 SPAN 目的设备。如果没有足够的硬件资源，甚至连一个空的 FSPAN ACL 都不会被加载。
所有的特性集上都支持 IPv4 和 MAC FSPAN ACL。只有高级 IP 服务特性集中才支持 IPv6 FSPAN ACL。

默认的 SPAN 和 RSPAN 配置

表 78：默认的 SPAN 和 RSPAN 配置

特性	**默认设置**
SPAN 状态（SPAN 和 RSPAN）	禁用
受监控的源端口流量	接收和发送的流量（**both**）
封装类型（目的端口）	本地方式（未打标数据包）
入向转发（目的端口）	禁用
VLAN 过滤	在将 Trunk 接口作为源端口时，监控所有 VLAN 的流量
RSPAN VLAN	未配置

配置指导

SPAN 配置指导

- 要想从 SPAN 会话中删除源或目的端口/VLAN，用户可以使用全局配置命令 **no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*}，或使用全局配置命令 **no monitor session** *session_number* **destination interface** *interface-id*。对于目的接口来说，**no** 形式的命令中要忽略关键字 **encapsulation**；

- 要想监控 Trunk 端口上的所有 VLAN，用户需要使用全局配置命令 **no monitor session** *session_number* **filter**。

RSPAN 配置指导

- 所有 SPAN 配置指导都适用于 RSPAN；
- 由于 RSPAN VLAN 具有独特的属性，用户应该在网络中保留一些 VLAN 用作 RSPAN VLAN；不要为这些 VLAN 分配 Access 端口；
- 用户可以为 RSPAN 流量应用一个出向 ACL，以此来有选择地过滤或监控指定数据包。用户可以在 RSPAN 源设备中，在 RSPAN VLAN 中指定这些 ACL；
- 对于 RSPAN 配置来说，用户可以把源端口和目的端口分布在网络中的多台设备上；
- RSPAN VLAN 中的 Access 端口（包括语音 VLAN 端口）会被置为非活跃（Inactive）状态；
- 用户可以把任意 VLAN 配置为 RSPAN VLAN，只要这些 VLAN 满足以下条件：
  - 在所有设备上为一个 RSPAN 会话使用相同的 RSPAN VLAN；
  - 所有参与的设备都要支持 RSPAN 特性；

如何配置 SPAN 和 RSPAN

创建本地 SPAN 会话

用户可以按照以下步骤，创建 SPAN 会话并指定源（受监控）端口或 VLAN，以及目的（监控）端口。

总步骤

1. enable

2. configure terminal

3. no monitor session {*session_number* | **all** | **local** | **remote**}

4. monitor session *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]

5. monitor session *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}

6. end

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal**	进入全局配置模式

	示例： Device# **configure terminal**	
步骤 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} **示例：** Device(config)# **no monitor session all**	删除现有的 SPAN 会话配置。 • *session_number* 的取值范围是 1 至 66 • **all**——删除所有 SPAN 会话 • **local**——删除所有本地会话 • **remote**——删除所有远端会话
步骤 4	**monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**] **示例：** Device(config)# **monitor session 1 source interface gigabitethernet1/0/1**	设置 SPAN 会话和源端口（受监控端口）。 • *session_number* 的取值范围是 1 至 66 • 在 *interface-id* 部分指定受监控的源端口。有效接口包括物理接口和 Port-Channel 逻辑接口（**port-channel** *port-channel-number*）。有效的 Port-Channel 编号为 1 至 48 • 在 *vlan-id* 部分指定受监控的源 VLAN。取值范围是 1 至 4094（不包括 RSPAN VLAN） **注释：** 一个会话中可以包含多个源（端口或 VLAN），用户需要通过多条命令对多个源进行指定；但不能在一个会话中混合使用源端口和源 VLAN。 • （可选）在**[, | -]**部分指定一系列端口或一个端口范围。用户需要在逗号前后各输入一个空格；也需要在连字符前后各输入一个空格 • （可选）在 **both | rx | tx** 部分指定发往监控器的流量方向。如果用户没有指定流量方向，那么源端口就会同时发送它发送和接收的流量。 · **both**——监控收到和发送的流量 · **rx**——监控收到的流量 · **tx**——监控发送的流量 **注释：** 用户可以多次配置 **monitor session** *session_number* **source** 命令，来指定多个源端口
步骤 5	**monitor session** *session_number*	指定 SPAN 会话和目的端口（监控端

	destination {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} **示例：** `Device(config)# monitor session 1 destination interface gigabitethernet1/0/2 encapsulation replicate`	口）。 **注释：** 对于本地 SPAN 来说，用户必须为源接口和目的接口使用相同的会话编号。 • 在 *session_number* 部分输入步骤 4 中指定的会话编号 • 在 *interface-id* 部分指定目的端口。目的接口必须是物理端口；不能是 EtherChannel，也不能是 VLAN • （可选）在**[, | -]**部分指定一系列端口或一个端口范围。用户需要在逗号前后各输入一个空格；也需要在连字符前后各输入一个空格 （可选）**encapsulation replicate** 指定让目的接口复制源接口的封装模式。如果没有选择这个关键字，默认是以本地格式（未打标）发送数据包的。 **注释：** 用户可以多次使用命令 **monitor session** *session-number* **destination**，来配置多个目的端口
步骤 6	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 7	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 8	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

配置本地 SPAN 会话并配置入站流量

用户可以按照以下步骤，创建 SPAN 会话并指定源端口或 VLAN，以及目的端口，并为网络安全设备（比如 Inspur IDS 传感器应用）在目的端口上启用入站流量。

总步骤

1. enable

2. configure terminal

3. no monitor session {*session_number* | **all** | **local** | **remote**}

4. monitor session *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]

5. monitor session *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**] [**ingress** {**dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}

6. end

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 3	**no monitor session** {*session_number*	**all**	**local**	**remote**} 示例： `Device(config)# no monitor session all`	删除现有的 SPAN 会话配置。 • *session_number* 的取值范围是 1 至 66 • **all**——删除所有 SPAN 会话 • **local**——删除所有本地会话 • **remote**——删除所有远端会话
步骤 4	**monitor session** *session_number* **source** {**interface** *interface-id*	**vlan** *vlan-id*} [**,**	**-**] [**both**	**rx**	**tx**] 示例： `Device(config)# monitor session 2 source interface gigabitethernet1/0/1 rx`	设置 SPAN 会话和源端口（受监控端口）。
步骤 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,**	**-**] [**encapsulation replicate**] [**ingress** {**dot1q vlan** *vlan-id*	**untagged vlan** *vlan-id*	**vlan** *vlan-id*}]} 示例： `Device(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate`	指定 SPAN 会话、目的端口、数据包封装，以及入站 VLAN 和封装。 • 在 *session_number* 部分输入步骤 4 中指定的会话编号 • 在 *interface-id* 部分指定目的端口。目的接口必须是物理端口；不能是 EtherChannel，也不能是 VLAN • （可选）在**[,**	**-]**部分指定一系列端口或一个端口范围。用户需要在逗号或连字符前后各输入一个空格

	ingress dot1q vlan 6	• （可选）**encapsulation replicate** 指定让目的接口复制源接口的封装模式。如果没有选择这个关键字，默认是以本地格式（未打标）发送数据包的。 • 关键字 **ingress** 会在目的端口上启用入站流量转发，并指定以下封装类型： · **dot1q vlan** *vlan-id*——接收携带 IEEE 802.1Q 封装的入站数据包，并把指定 VLAN 作为默认 VLAN · **untagged vlan** *vlan-id* 或 **vlan** *vlan-id*——接收未携带标记的入站数据包封装类型，并把指定 VLAN 作为默认 VLAN
步骤 6	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 7	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 8	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

指定需要过滤的 VLAN

用户需要使用以下步骤，把 SPAN 源流量限制在指定 VLAN 中。

总步骤

1. enable

2. configure terminal

3. no monitor session {*session_number* | **all** | **local** | **remote**}

4. monitor session *session_number* **source interface** *interface-id*

5. monitor session *session_number* **filter vlan** *vlan-id* [**,** | **-**]

6. monitor session *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]}

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} **示例：** `Device(config)# no monitor session all`	删除现有的 SPAN 会话配置。 • *session_number* 的取值范围是 1 至 66 • **all**——删除所有 SPAN 会话 • **local**——删除所有本地会话 • **remote**——删除所有远端会话
步骤 4	**monitor session** *session_number* **source interface** *interface-id* **示例：** `Device(config)# monitor session 2 source interface gigabitethernet1/0/2 rx`	设置源端口（受监控端口）和 SPAN 会话的特征。 • *session_number* 的取值范围是 1 至 66 • 在 *interface-id* 部分指定受监控的源端口。指定的接口必须已经配置为 Trunk 端口
步骤 5	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] **示例：** `Device(config)# monitor session 2 filter vlan 1 - 5 , 9`	把 SPAN 源流量限制在指定 VLAN 中。 • 在 *session_number* 部分输入步骤 4 中指定的会话编号 • *vlan-id* 的取值范围是 1 至 4094 • （可选）用户可以使用逗号（**,**）指定一系列 VLAN，也可以使用连字符（**-**）指定一个 VLAN 范围。用户需要在逗号前后各输入一个空格；也需要在连字符前后各输入一个空格
步骤 6	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**encapsulation replicate**]} **示例：** `Device(config)# monitor session 2 destination interface gigabitethernet1/0/1`	指定 SPAN 会话和目的端口（监控端口）。 • 在 *session_number* 部分输入步骤 4 中指定的会话编号 • 在 *interface-id* 部分指定目的端口。目的接口必须是物理端口；不能是 EtherChannel，也不能是 VLAN • （可选）在[**,** | **-**]部分指定一系列端口或一个端口范围。用户需要

		在逗号前后各输入一个空格；也需要在连字符前后各输入一个空格 （可选）**encapsulation replicate** 指定让目的接口复制源接口的封装模式。如果没有选择这个关键字，默认是以本地格式（未打标）发送数据包的。
步骤 7	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 8	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 9	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

把一个 VLAN 配置为 RSPAN VLAN

用户可以按照以下步骤，创建一个新 VLAN，并为 RSPAN 会话把它配置为 RSPAN VLAN。

总步骤

1. enable

2. configure terminal

3. vlan *vlan-id*

4. remote-span

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**vlan** *vlan-id*	输入 VLAN ID 来创建 VLAN，或者输入

	示例： `Device(config)# vlan 100`	已有的 VLAN ID，并进入 VLAN 配置模式。取值范围是 2 至 1001，以及 1006 至 4094。 RSPAN VLAN 不能是 VLAN 1（默认 VLAN），也不能是 VLAN ID 1002 至 1005（保留作为令牌环和 FDDI VLAN）
步骤 4	**remote-span** **示例：** `Device(config-vlan)# remote-span`	把这个 VLAN 配置为 RSPAN VLAN
步骤 5	**end** **示例：** `Device(config-vlan)# end`	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

接下来做什么？

用户必须在参与 RSPAN 的所有设备上都创建 RSPAN VLAN。如果 RSPAN VLAN 的 ID 在正常范围内（小于 1005），并且网络中启用了 VTP，用户可以在一台设备上创建 RSPAN VLAN，之后 VTP 会把这个信息传播到 VTP 域中的其他设备上。对于扩展范围的 VLAN（ID 大于 1005）来说，用户必须在源和目的设备，以及在所有中间的设备上配置 RSPAN VLAN。

用户可以使用 VTP 修剪特性来更有效地接收 RSPAN 流量，或者从所有 Trunk 中手动删除不需要承载 RSPAN 流量的 RSPAN VLAN。

要想从 VLAN 中删除远端 SPAN 特征，并且把它恢复成普通 VLAN，用户可以使用 VLAN 配置命令 **no remote-van**。

要想从 SPAN 会话中删除源端口或源 VLAN，用户可以使用全局配置命令 **no monitor session** *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*}。要想从会话中删除 RSPAN VLAN，用户可以使用命令 **no monitor session** *session_number* **destination remote vlan** *vlan-id*。

创建 RSPAN 源会话

用户可以按照以下步骤，创建并启用 RSPAN 源会话，并指定受监控源和目的 RSPAN VLAN。

总步骤

1. enable

2. configure terminal

3. no monitor session {*session_number* | **all** | **local** | **remote**}

4. monitor session *session_number* **source** {**interface** *interface-id* | **vlan** *vlan-id*} [**,** | **-**] [**both** | **rx** | **tx**]

5. monitor session *session_number* **destination remote vlan** *vlan-id*

6. end

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**no monitor session** {*session_number*	**all**	**local**	**remote**} **示例：** `Device(config)# no monitor session 1`	删除现有的 SPAN 会话配置。 • *session_number* 的取值范围是 1 至 66 • **all**——删除所有 SPAN 会话 • **local**——删除所有本地会话 • **remote**——删除所有远端会话
步骤 4	**monitor session** *session_number* **source** {**interface** *interface-id*	**vlan** *vlan-id*} [**,**	**-**] [**both**	**rx**	**tx**] **示例：** `Device(config)# monitor session 1 source interface gigabitethernet1/0/1 tx`	设置 RSPAN 会话和源端口（受监控端口）。 • *session_number* 的取值范围是 1 至 66 • 为 RSPAN 会话输入源端口或源 VLAN： · 在 *interface-id* 部分指定受监控的源端口。有效接口包括物理接口和 Port-Channel 逻辑接口（**port-channel** *port-channel-number*）。有效的 Port-Channel 编号为 1 至 48 · 在 *vlan-id* 部分指定受监控的源 VLAN。取值范围是 1 至 4094（不包括 RSPAN VLAN）一个会话中可以包含多个源（端口或 VLAN），用户需要通过多条命令对多个源进行指定；但不能在一个会话中混合使用源端口和源 VLAN。 • （可选）在**[,**	**-]**部分指定一系列

		端口或一个端口范围。用户需要在逗号前后各输入一个空格；也需要在连字符前后各输入一个空格 • （可选）在 **both | rx | tx** 部分指定发往监控器的流量方向。如果用户没有指定流量方向，那么源端口就会同时发送它发送和接收的流量。 · **both**——监控收到和发送的流量 · **rx**——监控收到的流量 · **tx**——监控发送的流量
步骤 5	**monitor session** *session_number* **destination remote vlan** *vlan-id* **示例：** `Device(config)# monitor session 1 destination remote vlan 100`	指定 RSPAN 会话、目的 RSPAN VLAN 和目的端口组。 • 在 *session_number* 部分输入步骤 4 中指定的会话编号 • 在 *vlan-id* 部分指定受监控的源 RSPAN VLAN
步骤 6	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 7	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 8	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

指定需要过滤的 VLAN

用户需要使用以下步骤配置 RSPAN 源会话，把 RSPAN 源流量限制在指定 VLAN 中。

总步骤

1. enable

2. configure terminal

3. no monitor session {*session_number* | **all** | **local** | **remote**}

4. monitor session *session_number* **source interface** *interface-id*

5. monitor session *session_number* **filter vlan** *vlan-id* [**,** | **-**]

6. monitor session *session_number* **destination remote vlan** *vlan-id*

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**no monitor session** {*session_number* | **all** | **local** | **remote**} **示例：** `Device(config)# no monitor session 2`	删除现有的 SPAN 会话配置。 • *session_number* 的取值范围是 1 至 66 • **all**——删除所有 SPAN 会话 • **local**——删除所有本地会话 • **remote**——删除所有远端会话
步骤 4	**monitor session** *session_number* **source interface** *interface-id* **示例：** `Device(config)# monitor session 2 source interface gigabitethernet1/0/2 rx`	设置源端口（受监控端口）和 SPAN 会话的特征。 • *session_number* 的取值范围是 1 至 66 • 在 *interface-id* 部分指定受监控的源端口。指定的接口必须已经配置为 Trunk 端口
步骤 5	**monitor session** *session_number* **filter vlan** *vlan-id* [**,** | **-**] **示例：** `Device(config)# monitor session 2 filter vlan 1 - 5 , 9`	把 SPAN 源流量限制在指定 VLAN 中。 • 在 *session_number* 部分输入步骤 4 中指定的会话编号 • *vlan-id* 的取值范围是 1 至 4094 • （可选）用户可以使用逗号（**,**）指定一系列 VLAN，也可以使用连字符（**-**）指定一个 VLAN 范围。用户需要在逗号前后各输入一个空格；也需要在连字符前后各输入一个空格
步骤 6	**monitor session** *session_number* **destination remote vlan** *vlan-id* **示例：** `Device(config)# monitor session 2 destination remote vlan 902`	指定 RSPAN 会话和目的远端 VLAN（RSPAN VLAN）。 • 在 *session_number* 部分输入步骤 4 中指定的会话编号 • 在 vlan-*id* 部分指定 RSPAN VLAN 来承载发往目的端口的受监控流量

步骤 7	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 8	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 9	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

创建 RSPAN 目的会话

用户可以在不同的设备或设备堆栈上配置 RSPAN 目的会话；也就是说，不在配置了源会话的设备或设备堆栈上进行配置。

用户可以按照以下步骤在相关设备上定义 RSPAN VLAN，来创建 RSPAN 目的会话并指定源 RSPAN VLAN 和目的端口。

总步骤

1. enable

2. configure terminal

3. vlan *vlan-id*

4. remote-span

5. exit

6. no monitor session {*session_number* | **all** | **local** | **remote**}

7. monitor session *session_number* **source remote vlan** *vlan-id*

8. monitor session *session_number* **destination interface** *interface-id*

9. end

10. show running-config

11. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**vlan** *vlan-id*	指定在源设备上创建的 RSPAN VLAN

	示例： Device(config)# **vlan 901**	的 VLAN ID 并进入 VLAN 配置模式。 如果源和目的设备都参与了 VTP，并且 RSPAN VLAN ID 是 2 至 1005 之间的值，就不需要配置步骤 3 至步骤 5，因为 RSPAN VLAN ID 会通过 VTP 网络进行传播。
步骤 4	**remote-span** **示例：** Device(config-vlan)# **remote-span**	把这个 VLAN 配置为 RSPAN VLAN
步骤 5	**exit** **示例：** Device(config-vlan)# **exit**	返回全局配置模式
步骤 6	**no monitor session** {*session_number* | **all** | **local** | **remote**} **示例：** Device(config)# **no monitor session 1**	删除现有的 SPAN 会话配置。 • *session_number* 的取值范围是 1 至 66 • **all**——删除所有 SPAN 会话 • **local**——删除所有本地会话 • **remote**——删除所有远端会话
步骤 7	**monitor session** *session_number* **source remote vlan** *vlan-id* **示例：** Device(config)# **monitor session 1 source remote vlan 901**	指定 RSPAN 会话和源 RSPAN VLAN。 • *session_number* 的取值范围是 1 至 66 • 在 *vlan-id* 部分指定受监控的源 RSPAN VLAN
步骤 8	**monitor session** *session_number* **destination interface** *interface-id* **示例：** Device(config)# **monitor session 1 destination interface gigabitethernet2/0/1**	指定 RSPAN 会话和目的端口。 • 在 *session_number* 部分输入步骤 7 中指定的会话编号。 在 RSPAN 目的会话中，用户必须使用与源 RSPAN VLAN 和目的端口相同的会话编号 • 在 *interface-id* 部分指定目的端口。目的接口必须是物理端口 • 虽然命令行帮助信息中可以看到 **encapsulation replicate**，但 RSPAN 并不支持。原始 VLAN ID 会被 RSPAN VLAN ID 改写，目的端口上的所有数据包都是未打标的
步骤 9	**end** **示例：**	返回特权 EXEC 模式

	Device(config)# **end**	
步骤 10	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 11	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

创建 RSPAN 目的会话并配置入站流量

用户可以按照以下步骤，创建 RSPAN 目的会话并指定源 RSPAN VLAN，以及目的端口，并为网络安全设备（比如 Inspur IDS 传感器应用）在目的端口上启用入站流量。

总步骤

1. enable

2. configure terminal

3. no monitor session {*session_number* | **all** | **local** | **remote**}

4. monitor session *session_number* **source remote vlan** *vlan-id*

5. monitor session *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** {**dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]}

6. end

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**no monitor session** {*session_number*	**all**	**local**	**remote**} **示例：** Device(config)# **no monitor session 2**	删除现有的 SPAN 会话配置。 • *session_number* 的取值范围是 1 至 66 • **all**——删除所有 SPAN 会话 • **local**——删除所有本地会话 • **remote**——删除所有远端会话
步骤 4	**monitor session** *session_number* **source remote vlan** *vlan-id*	指定 RSPAN 会话和源 RSPAN VLAN。 • *session_number* 的取值范围是 1

	示例： `Device(config)#` **`monitor session 2 source remote vlan 901`**	至 66 • 在 *vlan-id* 部分指定受监控的源 RSPAN VLAN
步骤 5	**monitor session** *session_number* **destination** {**interface** *interface-id* [**,** | **-**] [**ingress** {**dot1q vlan** *vlan-id* | **untagged vlan** *vlan-id* | **vlan** *vlan-id*}]} **示例：** `Device(config)#` **`monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6`**	指定 SPAN 会话、目的端口、数据包封装，以及入站 VLAN 和封装。 • 在 *session_number* 部分输入步骤 5 中指定的会话编号 在 RSPAN 目的会话中，用户必须使用与源 RSPAN VLAN 和目的端口相同的会话编号 • 在 *interface-id* 部分指定目的端口。目的接口必须是物理端口 • 虽然命令行帮助信息中可以看到 **encapsulation replicate**，但 RSPAN 并不支持。原始 VLAN ID 会被 RSPAN VLAN ID 改写，目的端口上的所有数据包都是未打标的 • （可选）在**[,** | **-]**部分指定一系列端口或一个端口范围。用户需要在逗号前后各输入一个空格；也需要在连字符前后各输入一个空格 • 关键字 **ingress** 会在目的端口上启用入站流量转发，并指定以下封装类型： · **dot1q vlan** *vlan-id*——接收携带 IEEE 802.1Q 封装的入站数据包，并把指定 VLAN 作为默认 VLAN · **untagged vlan** *vlan-id* 或 **vlan** *vlan-id*——接收未携带标记的入站数据包封装类型，并把指定 VLAN 作为默认 VLAN
步骤 6	**end** **示例：** `Device(config)#` **`end`**	返回特权 EXEC 模式
步骤 7	**show running-config** **示例：** `Device#` **`show running-config`**	检查用户输入的信息
步骤 8	**copy running-config startup-config**	（可选）把输入的命令保存到配置文

	示例： Device# **copy running-config startup-config**	件中

监控 SPAN 和 RSPAN 工作

用户可以使用下面这个表格中描述的命令来查看 SPAN 和 RSPAN 的操作配置，以及监控器运行的结果。

表 79：监控 SPAN 和 RSPAN 工作

命令	目的
show monitor	显示当前的 SPAN、RSPAN、FSPAN 或 FRSPAN 配置

SPAN 和 RSPAN 配置示例

示例：配置本地 SPAN

以下示例展示了如何设置 SPAN 会话 1，使其能够把受监控的源端口流量发送到目的端口。首先，用户删除了为会话 1 配置的现有 SPAN 配置，然后设置监控源千兆以太网端口 1 的双向流量，并将流量镜像到目的千兆以太网端口 2，保留封装方式。

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 1
Device(config)# monitor session 1 source interface gigabitethernet1/0/1
Device(config)# monitor session 1 destination interface gigabitethernet1/0/2 encapsulation replicate
Device(config)# end
```

以下示例展示了如何把端口 1 从 SPAN 会话 1 的 SPAN 源中删除：

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 1 source interface gigabitethernet1/0/1
Device(config)# end
```

以下示例展示了如何禁用对端口 1 接收到的流量进行监控，这个端口之前被配置为监控双向流量：

```
Device> enable
Device# configure terminal
```

```
Device(config)# no monitor session 1 source interface gigabitethernet1/0/1 rx
```

对于端口 1 上接收到的流量的监控被禁用了，但从这个端口发出的流量会继续受到监控。

以下示例展示了如何移除 SPAN 会话 2 上的现有配置，并配置 SPAN 会话 2 来监控所有 VLAN 1 至 3 中端口接收到的流量，并将这些流量发送到目的千兆以太网端口 2。然后用户又添加了配置，将其变更为监控 VLAN 10 中所有端口上的所有流量。

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 2
Device(config)# monitor session 2 source vlan 1 - 3 rx
Device(config)# monitor session 2 destination interface gigabitethernet1/0/2
Device(config)# monitor session 2 source vlan 10
Device(config)# end
```

以下示例展示了如何移除 SPAN 会话 2 上的现有配置，并配置 SPAN 会话 2 来监控千兆以太网源端口 1 上收到的所有流量，并将这些流量发送到目的千兆以太网端口 2，使用与源端口相同的出向封装类型，用户还启用了使用 IEEE 802.1Q 的入向转发，并把 VLAN 6 配置为默认入向 VLAN。

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 2
Device(config)# monitor session 2 source gigabitethernet1/0/1 rx
Device(config)# monitor session 2 destination interface gigabitethernet1/0/2 encapsulation replicate ingress dot1q vlan 6
Device(config)# end
```

以下示例展示了如何移除 SPAN 会话 2 上的现有配置，并配置 SPAN 会话 2 来监控千兆以太网 Trunk 端口 2 上收到的所有流量，并且只把 VLAN 1 至 5，以及 VLAN 9 的流量发送到目的千兆以太网端口 1：

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 2
Device(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Device(config)# monitor session 2 filter vlan 1 - 5 , 9
Device(config)# monitor session 2 destination interface gigabitethernet1/0/1
Device(config)# end
```

示例：创建 RSPAN VLAN

以下示例展示了如何创建 RSPAN VLAN 901：

```
Device> enable
Device# configure terminal
Device(config)# vlan 901
```

```
Device(config-vlan)# remote span
Device(config-vlan)# end
```

以下示例展示了如何移除 SPAN 会话 1 上的现有配置，并配置 SPAN 会话 1 来监控多个源接口，并且把 RSPAN VLAN 901 配置为目的：

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 1
Device(config)# monitor session 1 source interface gigabitethernet1/0/1 tx
Device(config)# monitor session 1 source interface gigabitethernet1/0/2 rx
Device(config)# monitor session 1 source interface port-channel 2
Device(config)# monitor session 1 destination remote vlan 901
Device(config)# end
```

以下示例展示了如何移除 SPAN 会话 2 上的现有配置，并配置 SPAN 会话 2 来监控 Trunk 端口 2 上收到的流量，并且只把 VLAN 1 至 5，以及 VLAN 9 的流量发送到目的 RSPAN VLAN 902：

```
Device> enable
Device# configure terminal
Device(config)# no monitor session 2
Device(config)# monitor session 2 source interface gigabitethernet1/0/2 rx
Device(config)# monitor session 2 filter vlan 1 - 5 , 9
Device(config)# monitor session 2 destination remote vlan 902
Device(config)# end
```

以下示例展示了如何把 VLAN 901 配置为源远端 VLAN，把端口 1 配置为目的接口：

```
Device> enable
Device# configure terminal
Device(config)# monitor session 1 source remote vlan 901
Device(config)# monitor session 1 destination interface gigabitethernet2/0/1
Device(config)# end
```

以下示例展示了如何在 RSPAN 会话 2 中把 VLAN 901 配置为源远端 VLAN、把千兆以太网源端口 2 配置为目的接口、把 VLAN 6 设置为默认接收 VLAN，并且在接口上启用入流量转发：

```
Device> enable
Device# configure terminal
Device(config)# monitor session 2 source remote vlan 901
Device(config)# monitor session 2 destination interface gigabitethernet1/0/2 ingress vlan 6
Device(config)# end
```

其他参考资料

相关文档

相关主题	文档名称
系统命令	*Network Management Command Reference, Inspur INOS*

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

SPAN 和 RSPAN 的特性历史与信息

版本	特性信息
Inspur INOS 12.2	引入该特性

QoS

配置 QoS

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置 Auto-QoS 的先决条件

配置 Auto-QoS 的先决条件与配置标准 QoS 的先决条件相同。

配置 Auto-QoS 的限制条件

配置 Auto-QoS 有以下限制条件：

- SVI 接口不支持 Auto-QoS；
- 绑定在 EtherChannel 中的接口不支持 Auto-QoS；
- 接口配置模式中的命令 **trust device** *device_type* 是交换机上的独立命令。在 AutoQoS 配置中使用这条命令时，如果连接的对等体设备不是对应设备（也就是符合用户信任策略的设备），那么 CoS 和 DSCP 值都会设置为“0”，任何入站策略也不会生效。如果连接的对等体设备是对应设备，那么入站策略就会生效；
- 用户在这台设备中使用 3.2.2 版本之前的软件时需要注意。如果用户在这台设备中使用了 3.2.2 版本之前的软件，那么必须按照后文中介绍的 Auto-QoS 升级流程进行升级；
- 不要为支持视频的 IP 电话配置 **auto qos voip inspur-phone** 选项。这个选项会重写视频数据包的 DSCP 标记，由于这些数据包不具备加速转发优先级，因此这种做法会导致这些数据包被分类为 class-default 类；
- 在用户使用命令 **auto qos voip inspur-phone**，把 Auto-QoS 从启动配置推送到运行配置中时，它并不会生成配置。这就是预期的效果，这样做是为了每次从启动配置文件中推送 **auto qos voip inspur-phone** 命令时，如果有用户自定义的 QoS 策略，防止默认配置覆盖用户创建的自定义 QoS 策略。

  用户可以使用以下解决方法来突破这一限制：

  - 在交换机接口上手动配置命令 **auto qos voip inspur-phone**；

- 对于新的交换机来说，如果用户从启动配置中推送 Auto-QoS 命令，命令应该会在标准模版中包含以下部分：
  1. 接口级别：
     - **trust device inspur-phone**
     - **auto qos voip inspur-phone**
     - **service-policy input** AutoQoS-4.0-InspurPhone-Input-Policy
     - **service-policy output** AutoQos-4.0-Output-Policy
  2. 全局级别：
     - class-map
     - policy-map
     - ACL（ACE）
- 如果接口上已经配置了命令 **auto qos voip inspur-phone**，但还没有生成策略，用户就需要在所有接口上禁用这条命令，然后在每个接口上重新进行配置。

配置 Auto-QoS 的相关信息

QoS 概述

用户可以使用 Auto-QoS 特性来简化 QoS 特性的部署。Auto-QoS 能够确定网络设计，并启用 QoS 配置，这样交换机能够对不同的流量执行优先级不同的操作。
交换机上能够部署 MQC 模型。这意味着 Auto-QoS 不使用某些全局配置，而是在交换机的接口上应用一些全局 class-map 和 policy-map。
Auto-QoS 可以匹配流量，然后把匹配的数据包分到 qos-group 中。这种做法能够让出站 policy-map 把特定 qos-group 中的数据包放入指定的队列中，其中包括优先级队列。
QoS 需要双向设置，入向和出向。在入方向上，交换机端口需要信任数据包中的 DSCP（默认行为）。在出方向上，交换机端口需要为语音数据包提供“优先转发”优先级。如果语音在出向队列中的其他数据包后面排队等待发送，就会经历过长的延迟，终端主机会丢弃这个数据包，因为这个数据包到达的时间已经超出了应该接收这个数据包的时间窗口。

Auto-QoS 集合特性概述

在用户输入 Auto-QoS 命令时，交换机会显示出所有它自己生成的命令，就好像这些命令是通过 CLI 输入的一样。用户可以使用 Auto-QoS 集合（Compact）特性在运行配置中隐藏 Auto-QoS 生成的命令。这样做可以使用户更轻松地读懂运行配置，同时提高内存的利用效率。

Auto-QoS 全局配置模版

通常来说，Auto-QoS 命令会生成一系列 class-map 命令，这些命令会对 ACL 或 DSCP 和/或 CoS 值进行匹配，然后把匹配的流量放到不同的应用类别中。Auto-QoS 还会生成入向策略，它会匹配生成的类别，在一些情况中，还会把类别限速为指定带宽。Auto-QoS 会生成 8 个出向队列 class-map。实际的出向策略会把一条队列分配到（拥有这 8 条出向队列的）class-

map 中的某条队列中。
Auto-QoS 命令只会按需生成模版。举例来说，当用户第一次使用新的 Auto-QoS 命令时，会生成定义了 8 条队列的出向 service-policy 全局配置。从这时开始，应用到其他接口的 Auto-QoS 命令不会再为出向队列生成模版，因为所有的 Auto-QoS 命令都使用相同的 8 队列模型，而这个模型在第一次输入新的 Auto-QoS 命令时就已经生成了。

Auto-QoS policy-map 和 class-map

在输入了适当的 Auto-QoS 命令后，会发生以下事件：
- 创建指定的 class-map；
- 创建指定的 policy-map（入向和出向）；
- 把 policy-map 关联到指定接口；
- 为接口配置信任等级。

Auto-QoS 对运行配置的影响

在启用 Auto-QoS 时，交换机会把接口配置命令 **auto qos** 和生成的全局配置添加到运行配置中。
交换机在应用 Auto-QoS 生成的命令时，就好像这些命令是通过 CLI 输入的一样。现有的用户配置可能会导致生成的命令应用失败，或者被生成的命令覆盖。这些行为在发生时并不会有警告信息。如果生成的所有命令都成功应用了，那么用户输入的那些没被覆盖的配置会保留在运行配置中。用户输入的被覆盖了的配置可以在不把当前配置保存到内存中，并重启交换机来恢复。如果生成的命令没有应用成功，则会恢复之前的运行配置。

Auto-QoS 集合特性对运行配置的影响

如果用户启用了 Auto-QoS 集合特性：
- 在运行配置中只会显示出用户在 CLI 中输入的 Auto-QoS 命令；
- 生成的全局配置和接口配置是隐藏的；
- 在用户保存配置时，只有用户输入的 Auto-QoS 命令会被保存（隐藏配置不会被保存）；
- 在用户重启交换机后，系统会检测并重新执行保存的 Auto-QoS 命令，并生成符合 AutoQoS SRND4.0 的配置集。

注释： 在启用了 Auto-QoS 集合特性后，用户不要对 Auto-QoS 生成的命令做修改，因为用户的变更会在交换机重启时被覆盖。
如果用户启用了 auto qos global compact：
- 可以使用 **show derived-config** 命令来查看隐藏的 AQC 命令；
- AQC 命令不会储存到内存中。在每次交换机重启后会生成这些命令；
- 在启用了 Auto-QoS 集合后，用户不应该修改 Auto-QoS 生成的命令；
- 如果接口上配置了 Auto-QoS，并且如果用户需要禁用 AQC，那么用户应该先在接口上禁用 Auto-QoS。

如何配置 Auto-QoS

配置 Auto-QoS（CLI）

为了优化 QoS 的性能，用户应该在网络中的所有设备上都配置 Auto-QoS。

总步骤

1. configure terminal

2. interface *interface-id*

3. 根据用户的Auto-QoS配置，使用以下命令之一：

- **auto qos voip** {**inspur-phone** | **inspur-softphone** | **trust**}
- **auto qos video** {**cts** | **ip-camera** | **media-player**}
- **auto qos classify** [**police**]
- **auto qos trust** {**cos** | **dscp**}

4. end

5. show auto qos interface *interface-id*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 2	**interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet 3/0/1`	指定端口：连接着 VoIP 端口、视频设备，或上行端口连接着另一台网络内部的可信交换机或路由器，并进入接口配置模式
步骤 3	根据用户的Auto-QoS配置，使用以下命令之一： • **auto qos voip** {**inspur-phone**	**inspur-softphone**	**trust**} • **auto qos video** {**cts**	**ip-camera**	**media-player**} • **auto qos classify** [**police**] • **auto qos trust** {**cos**	**dscp**} **示例：** `Device(config-if)# auto qos trust dscp`	以下命令为 VoIP 启用 Auto-QoS： • **auto qos voip inspur-phone**——如果端口连接着 Inspur IP 电话，那么只有当检测到了 IP 电话，入站数据包上的 QoS 标签才会被信任（通过 CDP 实现条件信任）。 **注释：** 不要为支持视频的 IP 电话配置命令 **auto qos voip inspur-phone**。这个选项会重写视频数据包的 DSCP 标记，由于这些数据包不具备加速转发优先级，因此这种做法会导致这些数据包被分类为 class-default 类。 • **auto qos voip inspur-softphone**——这个端口连接着运行 Inspur 软电话特性的设备。这条命令会为

		运行 Inspur IP 软电话应用的 PC 生成 QoS 配置，可以标记和限速从这个接口进入的流量。配置了这条命令的接口是不受信任的接口 • **auto qos voip trust**——上行链路端口连接着可信交换机或路由器，并且信任入向数据包中的 VoIP 流量分类。 用户可以使用以下命令为指定的视频设备（系统、摄像头或媒体播放器）启用 Auto-QoS： • **auto qos video cts**——端口连接着 Inspur 网真系统。只有当检测到 Inspur 网真系统时，才会信任入站数据包的 QoS 标签（通过 CDP 实现条件信任） • **auto qos video ip-camera**——端口连接着 Inspur 视频监视摄像头。只有当检测到 Inspur 摄像头时，才会信任入站数据包的 QoS 标签（通过 CDP 实现条件信任） • **auto qos video media-player**——端口连接着支持 CDP 的 Inspur 数字媒体播放器。只有当检测到数字媒体播放器时，才会信任入站数据包的 QoS 标签（通过 CDP 实现条件信任） 用户可以使用以下命令来启用 Auto-QoS 分类功能： • **auto qos classify police**——这条命令为不可信接口生成 QoS 配置。这个配置会在接口上应用 service-policy，用来分类从不可信的桌面/设备进来的流量，为其分配相应的标记。生成的 service-policy 也可以用来实现限速 用户可以使用以下命令来为可信接口启用 Auto-QoS： • **auto qos trust cos**——服务类别 • **auto qos trust dscp**——查分服务代码点
步骤 4	**end** **示例：**	返回特权 EXEC 模式

	Device(config-if)# **end**	
步骤 5	**showauto qos interface** *interface-id* **示例：** Device# **show auto qos interface gigabitethernet 3/0/1**	（可选）显示启用了 Auto-QoS 的接口上的 Auto-QoS 命令。用户可以使用 **show running-config** 命令来查看 Auto-QoS 配置和用户对其的修改

升级 Auto-QoS（CLI）

只有在设备上使用 3.2.2 版本之前的软件时，用户才需要按照这个步骤来升级 Auto-QoS 特性。如果用户确实在设备上使用了 3.2.2 版本之前的软件，就必须执行这个 Auto-QoS 升级过程。

在开始前

在开始升级之前，用户需要移除当前交换机上的所有 Auto-QoS 配置。示例流程中展示了这个过程。

在完成示例步骤后,用户必须用新的或升级后的软件镜像来重启交换机并重新配置 Auto-QoS。

总步骤

1. show auto qos

2. no auto qos

3. show running-config | i autoQos

4. no policy-map *policy-map_name*

5. show running-config | i AutoQoS

6. show auto qos

7. write memory

具体步骤

步骤1. show auto qos

示例：

```
Device# show auto qos
GigabitEthernet2/0/3
auto qos voip inspur-phone
GigabitEthernet2/0/27
auto qos voip inspur-softphone
```

在特权 EXEC 模式中，输入这条命令记录所有当前的 Auto-QoS 配置。

步骤2. no auto qos

示例：

```
Device(config-if)#no auto qos
```

在接口配置模式中，为所有拥有 Auto-QoS 配置的接口使用命令 **no auto qos**。

步骤3. show running-config | i autoQos

示例：

```
Device# show running-config | i autoQos
```

返回到特权 EXEC 模式中，输入这条命令并记录所有剩下的 Auto-QoS class-map、policy-map、访问列表、table-map 或其他配置。

步骤4. no policy-map *policy-map_name*

示例：

```
Device)config# no policy-map pmap_101
Device)config# no class-map cmap_101
Device)config# no ip access-list extended AutoQos-101
Device)config# no table-map 101
Device)config# no table-map policed-dscp
```

在全局配置模式中，输入以下命令删除 QoS class-map、policy-map、access-list、table-map，以及任何其他 Auto-QoS 配置：

- **no policy-map** *policy-map-name*
- **no class-map** *class-map-name*
- **no ip access-list extended** *Auto-QoS-x*
- **no table-map** *table-map-name*
- **no table-map policed-dscp**

步骤5. **show running-config | i AutoQoS**

示例：

```
Device# show running-config | i AutoQos
```

返回到特权 EXEC 模式，再次使用这条命令确认设备中已经没有 Auto-QoS 配置，或者已经没有 Auto-QoS 的残留配置。

步骤6. **show auto qos**

示例：

```
Device# show auto qos
```

使用这条命令确保配置中已经不存在 Auto-QoS 配置或遗留的部分配置。

步骤7. **write memory**

示例：

```
Device# write memory
```

使用 **write memory** 命令，把 Auto-QoS 配置的变更写入 NV 内存中。

接下来做什么？

使用新的或升级后的软件镜像重启交换机。

使用新的或升级后的软件镜像重启交换机后，按照步骤 1 中命令 **show auto qos** 的输出内容个，为相应的交换机接口重新配置 Auto-QoS。

注释： 每台交换机或堆栈中只有一个 table-map 用来为超出流量进行标记，另一个 table-map 对违规流量进行标记。如果交换机在超出行为下已经有了 table-map，用户就无法应用 Auto-QoS 策略了。

启用 Auto-QoS 集合

用户需要使用以下命令来启用 Auto-QoS 集合：

总步骤

1. configure terminal

2. auto qos global compact

具体步骤

	命令或操作	目的
步骤 1	**configure terminal**	进入全局配置模式

	示例： Device# **configure terminal**	
步骤 2	**auto qos global compact** 示例： Device(config)# **auto qos global compact**	启用 Auto-QoS 集合，并为 Auto-QoS 生成（隐藏的）全局配置。 用户可以在接口配置模式中输入想要配置的 Auto-QoS 命令，由系统生成的接口配置也是隐藏的。 要想查看已经应用的 Auto-QoS 配置，用户可以使用以下特权 EXEC 模式的命令： • **show derived-config** • **show policy-map** • **show access-list** • **show class-map** • **show table-map** • **show auto-qos** • **show policy-map interface** • **show ip access-lists** 这些命令中都有关键字“**AutoQos-**”

接下来做什么？

要想禁用 Auto-QoS 集合，用户需要通过 **no** 格式的 Auto-QoS 命令，删除所有接口上的 Auto-QoS 实例，然后在全局配置模式中输入命令 **no auto qos global compact**。

监控 Auto-QoS

表 86：监控 Auto-QoS 的命令

命令	描述
show auto qos [**interface** [*interface-id*]]	显示初始的 Auto-QoS 配置。 用户可以对比命令 **show auto qos** 和命令 **show running-config** 的输出内容，来找出哪些是用户定义的 QoS 设置。
show running-config	显示有可能受到 Auto-QoS 影响的 QoS 配置信息。 用户可以对比命令 **show auto qos** 和命令 **show running-config** 的输出内容，来找出哪些是用户定义的 QoS 设置。
show derived-config	显示隐藏的 **mls qos** 命令，这是由于 Auto-QoS 模版，配置在运行配置中的。

QoS 的排错

为了对 Auto-QoS 进行排错，用户需要使用特权 EXEC 命令 **debug auto qos**。更多信息用户可以参考这个版本设备的命令参考手册中，有关命令 **debug auto qos** 的内容。
要想在一个端口上禁用 Auto-QoS，用户需要在接口配置模式中，使用 **no** 格式的 **auto qos** 命令，比如 **no auto qos voip**。这时只有 Auto-QoS 为这个端口生成的接口配置会被移除。如果这是最后一个启用了 Auto-QoS 的端口，并且用户输入了命令 **no auto qos voip**，交换机也会认为 Auto-QoS 已禁用，即使 Auto-QoS 生成的全局配置命令还在（避免影响其他与全局配置相关的端口上的流量）。

Auto-QoS 的配置示例

示例：auto qos trust cos

以下示例展示了 **auto qos trust cos** 命令的用法，并且应用了 policy-map 和 class-map。
用户在使用这条命令时，还创建并应用了以下 policy-map：

- AutoQos-4.0-Trust-Cos-Input-Policy
- AutoQos-4.0-Output-Policy

用户在使用这条命令时，还创建并应用了以下 class-map：

- class-default（match-any）
- AutoQos-4.0-Output-Priority-Queue（match-any）
- AutoQos-4.0-Output-Control-Mgmt-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Conf-Queue（match-any）
- AutoQos-4.0-Output-Trans-Data-Queue（match-any）
- AutoQos-4.0-Output-Bulk-Data-Queue（match-any）
- AutoQos-4.0-Output-Scavenger-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Strm-Queue（match-any）

```
Device(config)# interface gigabitEthernet1/0/17
Device(config-if)# auto qos trust cos
Device(config-if)# end
Device# show policy-map interface GigabitEthernet1/0/17
GigabitEthernet1/0/7
Service-policy input: AutoQos-4.0-Trust-Cos-Input-Policy
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
cos cos table AutoQos-4.0-Trust-Cos-Table
Service-policy output: AutoQos-4.0-Output-Policy
```

```
queue stats for all priority classes:
Queueing
priority level 1
(total drops) 0
(bytes output) 0
Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
0 packets
Match: dscp cs4 (32) cs5 (40) ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
Priority: 30% (300000 kbps), burst bytes 7500000,
Priority Level: 1
Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
0 packets
Match: dscp cs2 (16) cs3 (24) cs6 (48) cs7 (56)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
queue-limit dscp 16 percent 80
queue-limit dscp 24 percent 90
queue-limit dscp 48 percent 100
queue-limit dscp 56 percent 100
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
0 packets
Match: dscp af41 (34) af42 (36) af43 (38)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 4
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
```

```
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
0 packets
Match: dscp af21 (18) af22 (20) af23 (22)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 2
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
0 packets
Match: dscp af11 (10) af12 (12) af13 (14)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 1
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 4%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
0 packets
Match: dscp cs1 (8)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 1%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
0 packets
Match: dscp af31 (26) af32 (28) af33 (30)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
```

```
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 25%
queue-buffers ratio 25
```

示例：auto qos trust dscp

以下示例展示了 **auto qos trust dscp** 命令的用法，并且应用了 policy-map 和 class-map。
用户在使用这条命令时，还创建并应用了以下 policy-map：

- AutoQos-4.0-Trust-Dscp-Input-Policy
- AutoQos-4.0-Output-Policy

用户在使用这条命令时，还创建并应用了以下 class-map：

- class-default（match-any）
- AutoQos-4.0-Output-Priority-Queue（match-any）
- AutoQos-4.0-Output-Control-Mgmt-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Conf-Queue（match-any）
- AutoQos-4.0-Output-Trans-Data-Queue（match-any）
- AutoQos-4.0-Output-Bulk-Data-Queue（match-any）
- AutoQos-4.0-Output-Scavenger-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Strm-Queue（match-any）

```
Device(config)# interface GigabitEthernet1/0/18
Device(config-if)# auto qos trust dscp
Device(config-if)# end
Device#show policy-map interface GigabitEthernet1/0/18
GigabitEthernet1/0/18
Service-policy input: AutoQos-4.0-Trust-Dscp-Input-Policy
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp dscp table AutoQos-4.0-Trust-Dscp-Table
Service-policy output: AutoQos-4.0-Output-Policy
queue stats for all priority classes:
```

```
Queueing
priority level 1
(total drops) 0
(bytes output) 0
Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
0 packets
Match: dscp cs4 (32) cs5 (40) ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
Priority: 30% (300000 kbps), burst bytes 7500000,
Priority Level: 1
Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
0 packets
Match: dscp cs2 (16) cs3 (24) cs6 (48) cs7 (56)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
queue-limit dscp 16 percent 80
queue-limit dscp 24 percent 90
queue-limit dscp 48 percent 100
queue-limit dscp 56 percent 100
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
0 packets
Match: dscp af41 (34) af42 (36) af43 (38)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 4
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
```

```
Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
0 packets
Match: dscp af21 (18) af22 (20) af23 (22)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 2
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
0 packets
Match: dscp af11 (10) af12 (12) af13 (14)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 1
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 4%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
0 packets
Match: dscp cs1 (8)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 1%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
0 packets
Match: dscp af31 (26) af32 (28) af33 (30)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
```

```
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 25%
queue-buffers ratio 25
```

示例：auto qos video cts

以下示例展示了 **auto qos video cts** 命令的用法，并且应用了 policy-map 和 class-map。
用户在使用这条命令时，还创建并应用了以下 policy-map：

- AutoQos-4.0-Trust-Cos-Input-Policy
- AutoQos-4.0-Output-Policy

用户在使用这条命令时，还创建并应用了以下 class-map：

- class-default（match-any）
- AutoQos-4.0-Output-Priority-Queue（match-any）
- AutoQos-4.0-Output-Control-Mgmt-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Conf-Queue（match-any）
- AutoQos-4.0-Output-Trans-Data-Queue（match-any）
- AutoQos-4.0-Output-Bulk-Data-Queue（match-any）
- AutoQos-4.0-Output-Scavenger-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Strm-Queue（match-any）

```
Device(config)# interface gigabitEthernet1/0/12
Device(config-if)# auto qos video cts
Device(config-if)# end
Device# show policy-map interface gigabitEthernet1/0/12
GigabitEthernet1/0/12
Service-policy input: AutoQos-4.0-Trust-Cos-Input-Policy
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
cos cos table AutoQos-4.0-Trust-Cos-Table
Service-policy output: AutoQos-4.0-Output-Policy
queue stats for all priority classes:
```

```
Queueing
priority level 1
(total drops) 0
(bytes output) 0
Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
0 packets
Match: dscp cs4 (32) cs5 (40) ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
Priority: 30% (300000 kbps), burst bytes 7500000,
Priority Level: 1
Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
0 packets
Match: dscp cs2 (16) cs3 (24) cs6 (48) cs7 (56)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
queue-limit dscp 16 percent 80
queue-limit dscp 24 percent 90
queue-limit dscp 48 percent 100
queue-limit dscp 56 percent 100
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
0 packets
Match: dscp af41 (34) af42 (36) af43 (38)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 4
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
```

```
Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
0 packets
Match: dscp af21 (18) af22 (20) af23 (22)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 2
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
0 packets
Match: dscp af11 (10) af12 (12) af13 (14)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 1
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 4%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
0 packets
Match: dscp cs1 (8)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 1%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
0 packets
Match: dscp af31 (26) af32 (28) af33 (30)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
```

```
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 25%
queue-buffers ratio 25
```

示例：auto qos video ip-camera

以下示例展示了 **auto qos video ip-camera** 命令的用法，并且应用了 policy-map 和 class-map。
用户在使用这条命令时，还创建并应用了以下 policy-map：

- AutoQos-4.0-Trust-Dscp-Input-Policy
- AutoQos-4.0-Output-Policy

用户在使用这条命令时，还创建并应用了以下 class-map：

- class-default（match-any）
- AutoQos-4.0-Output-Priority-Queue（match-any）
- AutoQos-4.0-Output-Control-Mgmt-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Conf-Queue（match-any）
- AutoQos-4.0-Output-Trans-Data-Queue（match-any）
- AutoQos-4.0-Output-Bulk-Data-Queue（match-any）
- AutoQos-4.0-Output-Scavenger-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Strm-Queue（match-any）

```
Device(config)# interface GigabitEthernet1/0/9
Device(config-if)# auto qos video ip-camera
Device(config-if)# end
Device# show policy-map interface GigabitEthernet1/0/9
GigabitEthernet1/0/9
Service-policy input: AutoQos-4.0-Trust-Dscp-Input-Policy
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp dscp table AutoQos-4.0-Trust-Dscp-Table
Service-policy output: AutoQos-4.0-Output-Policy
queue stats for all priority classes:
Queueing
```

```
priority level 1
(total drops) 0
(bytes output) 0
Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
0 packets
Match: dscp cs4 (32) cs5 (40) ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
Priority: 30% (300000 kbps), burst bytes 7500000,
Priority Level: 1
Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
0 packets
Match: dscp cs2 (16) cs3 (24) cs6 (48) cs7 (56)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
queue-limit dscp 16 percent 80
queue-limit dscp 24 percent 90
queue-limit dscp 48 percent 100
queue-limit dscp 56 percent 100
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
0 packets
Match: dscp af41 (34) af42 (36) af43 (38)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 4
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
```

```
0 packets
Match: dscp af21 (18) af22 (20) af23 (22)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 2
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
0 packets
Match: dscp af11 (10) af12 (12) af13 (14)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 1
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 4%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
0 packets
Match: dscp cs1 (8)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 1%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
0 packets
Match: dscp af31 (26) af32 (28) af33 (30)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
```

```
queue-buffers ratio 10
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 25%
queue-buffers ratio 25
```

示例：auto qos video media-player

以下示例展示了 **auto qos video media-player** 命令的用法，并且应用了 policy-map 和 class-map。

用户在使用这条命令时，还创建并应用了以下 policy-map：

- AutoQos-4.0-Trust-Dscp-Input-Policy
- AutoQos-4.0-Output-Policy

用户在使用这条命令时，还创建并应用了以下 class-map：

- class-default（match-any）
- AutoQos-4.0-Output-Priority-Queue（match-any）
- AutoQos-4.0-Output-Control-Mgmt-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Conf-Queue（match-any）
- AutoQos-4.0-Output-Trans-Data-Queue（match-any）
- AutoQos-4.0-Output-Bulk-Data-Queue（match-any）
- AutoQos-4.0-Output-Scavenger-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Strm-Queue（match-any）

```
Device(config)# interface GigabitEthernet1/0/25
Device(config-if)# auto qos video media-player
Device(config-if)# end
Device# show policy-map interface GigabitEthernet1/0/25
GigabitEthernet1/0/25
Service-policy input: AutoQos-4.0-Trust-Dscp-Input-Policy
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp dscp table AutoQos-4.0-Trust-Dscp-Table
Service-policy output: AutoQos-4.0-Output-Policy
queue stats for all priority classes:
Queueing
```

```
priority level 1
(total drops) 0
(bytes output) 0
Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
0 packets
Match: dscp cs4 (32) cs5 (40) ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
Priority: 30% (300000 kbps), burst bytes 7500000,
Priority Level: 1
Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
0 packets
Match: dscp cs2 (16) cs3 (24) cs6 (48) cs7 (56)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
queue-limit dscp 16 percent 80
queue-limit dscp 24 percent 90
queue-limit dscp 48 percent 100
queue-limit dscp 56 percent 100
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
0 packets
Match: dscp af41 (34) af42 (36) af43 (38)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 4
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
```

```
0 packets
Match: dscp af21 (18) af22 (20) af23 (22)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 2
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
0 packets
Match: dscp af11 (10) af12 (12) af13 (14)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 1
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 4%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
0 packets
Match: dscp cs1 (8)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 1%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
0 packets
Match: dscp af31 (26) af32 (28) af33 (30)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
```

```
queue-buffers ratio 10
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 25%
queue-buffers ratio 25
```

示例：auto qos voip trust

以下示例展示了 **auto qos voip trust** 命令的用法，并且应用了 policy-map 和 class-map。
用户在使用这条命令时，还创建并应用了以下 policy-map：

- AutoQos-4.0-Trust-Cos-Input-Policy
- AutoQos-4.0-Output-Policy

用户在使用这条命令时，还创建并应用了以下 class-map：

- class-default（match-any）
- AutoQos-4.0-Output-Priority-Queue（match-any）
- AutoQos-4.0-Output-Control-Mgmt-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Conf-Queue（match-any）
- AutoQos-4.0-Output-Trans-Data-Queue（match-any）
- AutoQos-4.0-Output-Bulk-Data-Queue（match-any）
- AutoQos-4.0-Output-Scavenger-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Strm-Queue（match-any）

```
Device(config)# interface gigabitEthernet1/0/31
Device(config-if)# auto qos voip trust
Device(config-if)# end
Device# show policy-map interface GigabitEthernet1/0/31
GigabitEthernet1/0/31
Service-policy input: AutoQos-4.0-Trust-Cos-Input-Policy
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
cos cos table AutoQos-4.0-Trust-Cos-Table
Service-policy output: AutoQos-4.0-Output-Policy
Queueing
priority level 1
(total drops) 0
```

```
(bytes output) 0
Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
0 packets
Match: dscp cs4 (32) cs5 (40) ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
Priority: 30% (300000 kbps), burst bytes 7500000,
Priority Level: 1
Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
0 packets
Match: dscp cs2 (16) cs3 (24) cs6 (48) cs7 (56)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
queue-limit dscp 16 percent 80
queue-limit dscp 24 percent 90
queue-limit dscp 48 percent 100
queue-limit dscp 56 percent 100
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any
0 packets
Match: dscp af41 (34) af42 (36) af43 (38)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 4
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
0 packets
Match: dscp af21 (18) af22 (20) af23 (22)
```

```
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 2
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
0 packets
Match: dscp af11 (10) af12 (12) af13 (14)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 1
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 4%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
0 packets
Match: dscp cs1 (8)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 1%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any
0 packets
Match: dscp af31 (26) af32 (28) af33 (30)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: class-default (match-any)
```

```
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 25%
queue-buffers ratio 25
```

示例：auto qos voip inspur-phone

以下示例展示了 **auto qos voip inspur-phone** 命令的用法，并且应用了 policy-map 和 class-map。

用户在使用这条命令时，还创建并应用了以下 policy-map：

- AutoQos-4.0-InspurPhone-Input-Policy
- AutoQos-4.0-Output-Policy

用户在使用这条命令时，还创建并应用了以下 class-map：

- AutoQos-4.0-Voip-Data-InspurPhone-Class（match-any）
- AutoQos-4.0-Voip-Signal-InspurPhone-Class（match-any）
- AutoQos-4.0-Default-Class（match-any）
- class-default（match-any）
- AutoQos-4.0-Output-Priority-Queue（match-any）
- AutoQos-4.0-Output-Control-Mgmt-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Conf-Queue（match-any）
- AutoQos-4.0-Output-Trans-Data-Queue（match-any）
- AutoQos-4.0-Output-Bulk-Data-Queue（match-any）
- AutoQos-4.0-Output-Scavenger-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Strm-Queue（match-any）

```
Device(config)# interface gigabitEthernet1/0/5
Device(config-if)# auto qos voip inspur-phone
Device(config-if)# end
Device# show policy-map interface gigabitEthernet1/0/5
GigabitEthernet1/0/5
Service-policy input: AutoQos-4.0-InspurPhone-Input-Policy
Class-map: AutoQos-4.0-Voip-Data-InspurPhone-Class (match-any)
0 packets
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp ef
police:
cir 128000 bps, bc 8000 bytes
```

```
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Voip-Signal-InspurPhone-Class (match-any)
0 packets
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp cs3
police:
cir 32000 bps, bc 8000 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Default-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Default
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp default
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Service-policy output: AutoQos-4.0-Output-Policy
queue stats for all priority classes:
Queueing
priority level 1
(total drops) 0
(bytes output) 0
Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
0 packets
Match: dscp cs4 (32) cs5 (40) ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
```

```
5 minute rate 0 bps
Priority: 30% (300000 kbps), burst bytes 7500000,
Priority Level: 1
Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
0 packets
Match: dscp cs2 (16) cs3 (24) cs6 (48) cs7 (56)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
queue-limit dscp 16 percent 80
queue-limit dscp 24 percent 90
queue-limit dscp 48 percent 100
queue-limit dscp 56 percent 100
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
0 packets
Match: dscp af41 (34) af42 (36) af43 (38)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 4
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
0 packets
Match: dscp af21 (18) af22 (20) af23 (22)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 2
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
```

```
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
0 packets
Match: dscp af11 (10) af12 (12) af13 (14)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 1
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 4%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
0 packets
Match: dscp cs1 (8)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 1%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
0 packets
Match: dscp af31 (26) af32 (28) af33 (30)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 25%
```

```
queue-buffers ratio 25
```

示例：auto qos voip inspur-softphone

以下示例展示了 **auto qos voip inspur-softphone** 命令的用法，并且应用了 policy-map 和 class-map。

用户在使用这条命令时，还创建并应用了以下 policy-map：

- AutoQos-4.0-InspurSoftPhone-Input-Policy
- AutoQos-4.0-Output-Policy

用户在使用这条命令时，还创建并应用了以下 class-map：

- AutoQos-4.0-Voip-Data- Class（match-any）
- AutoQos-4.0-Voip-Signal-Class（match-any）
- AutoQos-4.0-Multimedia-Conf-Class（match-any）
- AutoQos-4.0-Bulk-Data-Class（match-any）
- AutoQos-4.0-Transaction-Class（match-any）
- AutoQos-4.0-Scavanger-Class（match-any）
- AutoQos-4.0-Signaling-Class（match-any）
- AutoQos-4.0-Default-Class（match-any）
- class-default（match-any）
- AutoQos-4.0-Output-Priority-Queue（match-any）
- AutoQos-4.0-Output-Control-Mgmt-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Conf-Queue（match-any）
- AutoQos-4.0-Output-Trans-Data-Queue（match-any）
- AutoQos-4.0-Output-Bulk-Data-Queue（match-any）
- AutoQos-4.0-Output-Scavenger-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Strm-Queue（match-any）

```
Device(config)# interface gigabitEthernet1/0/21
Device(config-if)# auto qos voip inspursoftphone
Device(config-if)# end
Device# show policy-map interface gigabitEthernet1/0/21
GigabitEthernet1/0/21
Service-policy input: AutoQos-4.0-InspurSoftPhone-Input-
Policy Class-map: AutoQos-4.0-Voip-Data-Class (match-any)
0 packets
Match: dscp ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp ef
police:
cir 128000 bps, bc 8000 bytes
```

```
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Voip-Signal-Class (match-any)
0 packets
Match: dscp cs3 (24)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp cs3
police:
cir 32000 bps, bc 8000 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Multimedia-Conf-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-MultiEnhanced-Conf
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp af41
police:
cir 5000000 bps, bc 156250 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
drop
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Bulk-Data-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Bulk-Data
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp af11
police:
```

```
cir 10000000 bps, bc 312500 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Transaction-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Transactional-Data
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp af21
police:
cir 10000000 bps, bc 312500 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Scavanger-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Scavanger
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp cs1
police:
cir 10000000 bps, bc 312500 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
drop
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Signaling-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Signaling
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp cs3
police:
cir 32000 bps, bc 8000 bytes
conformed 0 bytes; actions:
```

```
transmit
exceeded 0 bytes; actions:
drop
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Default-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Default
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp default
police:
cir 10000000 bps, bc 312500 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Service-policy output: AutoQos-4.0-Output-Policy
queue stats for all priority classes:
Queueing
priority level 1
(total drops) 0
(bytes output) 0
Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
0 packets
Match: dscp cs4 (32) cs5 (40) ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
Priority: 30% (300000 kbps), burst bytes 7500000,
Priority Level: 1
Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
0 packets
Match: dscp cs2 (16) cs3 (24) cs6 (48) cs7 (56)
0 packets, 0 bytes
5 minute rate 0 bps
```

```
Match: cos 3
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
queue-limit dscp 16 percent 80
queue-limit dscp 24 percent 90
queue-limit dscp 48 percent 100
queue-limit dscp 56 percent 100
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
0 packets
Match: dscp af41 (34) af42 (36) af43 (38)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 4
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
0 packets
Match: dscp af21 (18) af22 (20) af23 (22)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 2
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
0 packets
Match: dscp af11 (10) af12 (12) af13 (14)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 1
```

```
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 4%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
0 packets
Match: dscp cs1 (8)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 1%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
0 packets
Match: dscp af31 (26) af32 (28) af33 (30)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 25%
queue-buffers ratio 25
```

auto qos classify police

以下示例展示了 **auto qos classify police** 命令的用法，并且应用了 policy-map 和 class-map。用户在使用这条命令时，还创建并应用了以下 policy-map：

- AutoQos-4.0-Classify-Police-Input-Policy
- AutoQos-4.0-Output-Policy

用户在使用这条命令时，还创建并应用了以下 class-map：

- AutoQos-4.0-Multimedia-Conf-Class（match-any）
- AutoQos-4.0-Bulk-Data-Class（match-any）
- AutoQos-4.0-Transaction-Class（match-any）
- AutoQos-4.0-Scavanger-Class（match-any）
- AutoQos-4.0-Signaling-Class（match-any）
- AutoQos-4.0-Default-Class（match-any）
- class-default（match-any）
- AutoQos-4.0-Output-Priority-Queue（match-any）
- AutoQos-4.0-Output-Control-Mgmt-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Conf-Queue（match-any）
- AutoQos-4.0-Output-Trans-Data-Queue（match-any）
- AutoQos-4.0-Output-Bulk-Data-Queue（match-any）
- AutoQos-4.0-Output-Scavenger-Queue（match-any）
- AutoQos-4.0-Output-Multimedia-Strm-Queue（match-any）

```
Device(config)# interface gigabitEthernet1/0/6
Device(config-if)# auto qos classify police
Device(config-if)# end
Device# show policy-map interface gigabitEthernet1/0/6
GigabitEthernet1/0/6
Service-policy input: AutoQos-4.0-Classify-Police-Input-Policy
Class-map: AutoQos-4.0-Multimedia-Conf-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-MultiEnhanced-Conf
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp af41
police:
cir 5000000 bps, bc 156250 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
drop
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Bulk-Data-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Bulk-Data
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp af11
police:
cir 10000000 bps, bc 312500 bytes
```

```
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Transaction-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Transactional-Data
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp af21
police:
cir 10000000 bps, bc 312500 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Scavanger-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Scavanger
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp cs1
police:
cir 10000000 bps, bc 312500 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
drop
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Signaling-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Signaling
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp cs3
police:
cir 32000 bps, bc 8000 bytes
conformed 0 bytes; actions:
transmit
```

```
exceeded 0 bytes; actions:
drop
conformed 0000 bps, exceed 0000 bps
Class-map: AutoQos-4.0-Default-Class (match-any)
0 packets
Match: access-group name AutoQos-4.0-Acl-Default
0 packets, 0 bytes
5 minute rate 0 bps
QoS Set
dscp default
police:
cir 10000000 bps, bc 312500 bytes
conformed 0 bytes; actions:
transmit
exceeded 0 bytes; actions:
set-dscp-transmit dscp table policed-dscp
conformed 0000 bps, exceed 0000 bps
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Service-policy output: AutoQos-4.0-Output-Policy
queue stats for all priority classes:
Queueing
priority level 1
(total drops) 0
(bytes output) 0
Class-map: AutoQos-4.0-Output-Priority-Queue (match-any)
0 packets
Match: dscp cs4 (32) cs5 (40) ef (46)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 5
0 packets, 0 bytes
5 minute rate 0 bps
Priority: 30% (300000 kbps), burst bytes 7500000,
Priority Level: 1
Class-map: AutoQos-4.0-Output-Control-Mgmt-Queue (match-any)
0 packets
Match: dscp cs2 (16) cs3 (24) cs6 (48) cs7 (56)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 3
```

```
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
queue-limit dscp 16 percent 80
queue-limit dscp 24 percent 90
queue-limit dscp 48 percent 100
queue-limit dscp 56 percent 100
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Conf-Queue (match-any)
0 packets
Match: dscp af41 (34) af42 (36) af43 (38)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 4
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Trans-Data-Queue (match-any)
0 packets
Match: dscp af21 (18) af22 (20) af23 (22)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 2
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Bulk-Data-Queue (match-any)
0 packets
Match: dscp af11 (10) af12 (12) af13 (14)
0 packets, 0 bytes
5 minute rate 0 bps
Match: cos 1
0 packets, 0 bytes
```

```
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 4%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Scavenger-Queue (match-any)
0 packets
Match: dscp cs1 (8)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 1%
queue-buffers ratio 10
Class-map: AutoQos-4.0-Output-Multimedia-Strm-Queue (match-any)
0 packets
Match: dscp af31 (26) af32 (28) af33 (30)
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 10%
queue-buffers ratio 10
Class-map: class-default (match-any)
0 packets
Match: any
0 packets, 0 bytes
5 minute rate 0 bps
Queueing
(total drops) 0
(bytes output) 0
bandwidth remaining 25%
queue-buffers ratio 25
```

auto qos global compact

这个示例展示了命令 **auto qos global compact** 的用法。

```
Device# configure terminal
Device(config)# auto qos global compact
Device(config)# interface GigabitEthernet1/2
Device(config-if)# auto qos voip inspurphone
```

```
Device# show auto-qos
GigabitEthernet1/2
auto qos voip inspur-phone
Device# show running-config interface GigabitEthernet 1/0/2
interface GigabitEthernet1/0/2
auto qos voip inspur-phone
end
```

配置 Auto-QoS 之后的操作

如果用户需要在自己的 Auto-QoS 配置中进行变更的话，重新看看 QoS 文档。

Auto-QoS 的其他参考资料

相关文档

相关主题	文档名称
本章中命令的完整语法和用法信息	*QoS Command Reference (Inspur 6650 Switches)* *Inspur INOS Quality of Service Solutions Command Reference*

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID	http://www.icntnetworks.com

和密码。	

Auto-QoS 的特性历史与信息

版本	变更
Inspur INOS 12.2	引入该特性

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置服务质量的先决条件

在开始配置标准 QoS 之前，用户必须充分理解以下内容：

- 标准 QoS 的概念；
- 经典 Inspur INOS QoS；
- 模块化 QoS CLI（MQC）；
- 理解 QoS 的实施；
- 用户网络中使用的应用类型和流量模式；
- 用户网络的流量特征和需求。比如网络中的流量是否具有突发性？是否需要为语音和视频流预留带宽？
- 网络中的带宽和速率需求；
- 网络中的拥塞点位置。

QoS 的组成部分

服务质量（QoS）由以下重要部分构组成：

- 分类——分类是区分流量类型的过程，用户可以基于访问控制列表（ACL）、差分服务代码点（DSCP）、服务类别（CoS）和其他因素进行区分；
- 标记和突变——在流量上设置标记是为了向网络中的下游设备传达特定的信息，或者把信息从一个接口传递给另一个接口。当流量被打上标记后，设备就可以针对这个流量应

用 QoS 操作行为了。用户可以直接使用 **set** 命令来设置 QoS 行为，也可以通过 table-map 来进行设置，table-map 可以查看入站标记值，然后直接把入站标记值转换为出站标记值；

- 整形和限速——整形是指控制流量最大速率的过程，以防止下游设备遭到拥塞为目的，来调节流量速率。整形最常见的用法是对物理接口或逻辑接口发送的流量进行限制。限速是指对一个流量类别设定最大速率。如果超出限速了，QoS 马上会对相关流量执行相应的行为；
- 列队——列队的作用是为了防止流量拥塞。根据带宽的分配方式，QoS 会把流量分类到不同的队列中，来提供服务和调度。然后流量会接受调度，或者从端口发送出去；
- 带宽——带宽的分配方式决定了受到 QoS 策略影响的流量所能够使用的带宽容量；
- 受信——受信功能能够使流量通过交换机，并在用户没有明确指定策略配置时，保留终端携带的差分服务代码点（DSCP）、优先级或 CoS 值。

QoS 术语

在这个 QoS 配置指南中，会替换使用以下术语：

- “上游”（去往一个方向）与“入向”会替换使用；
- “下游”（从一个方向来）与“出向”会替换使用

注释：

QoS 的相关信息

QoS 概述

通过配置服务质量（QoS），用户可以以降低其他流量类型服务质量为代价，为特定类型的流量提供更好的服务。如果没有 QoS 的话，设备会对每个数据包提供尽力而为的服务，而完全不管数据包的内容或大小。设备在发送数据包的时候不提供任何可靠性保障、延迟保障，或吞吐量保障。

QoS 提供了以下特性：

- 低延迟
- 带宽保障
- 缓存空间和丢包规划
- 流量限速
- 更改数据帧或数据包头部的能力
- 相关服务

模块化 QoS 命令行界面

在设备上，QoS 特性是通过模块化 QoS 命令行界面（MQC）启用的。MQC 是一种命令行界面（CLI）结构，让用户能够创建流量策略，并将其应用在接口上。流量策略中包含流量类别，

以及一个或多个 QoS 特性。流量类别是用来对流量进行分类的，之后流量策略中的 QoS 特性就能够确定如何处理这些分类后的流量。
MQC 的一大主要目标是提供与平台无关的界面，使用户在 Inspur 平台上能够使用统一的命令结构来配置 QoS。

层级式 QoS

设备能够支持层级式 QoS（HQoS）。通过使用 HQoS，用户能够实施：

- 层级式分类——根据其他类别对流量进行分类；
- 层级式限速——在层级式策略中，拥有多个等级的限速配置；
- 层级式整形——在层级式策略中，整形也可以被配置为多个等级。

注释： 只有端口整形器支持层级式整形，对于父系交换机来说，用户只能实施 class-default 配置，并且只能对 class-default 实施整形行为。

QoS 的实施

通常网络的操作行为是以尽力而为的转发为基础的，也就是说所有流量拥有相同的优先级，并且有相同的机会能够得到及时传递。在拥塞发生时，所有流量被丢弃的几率也是相同的。在用户配置 QoS 特性时可以选择指定的网络流量，根据它的重要性调整优先级，并使用拥塞管理和拥塞避免技术为其提供特殊服务。在网络中实施 QoS 能够使网络行为便得更加可以预测，并且使带宽的利用率更为高效。
QoS 的实施是以差分服务（Diff-Serv）架构为基础的，差分服务是 Internet 工程任务组（IETF）提出的一种标准。这个架构明确了每个数据包要在进入网络的时候进行分类。
IP 数据包头部中携带着分类结果，也就是使用长度为 6 比特已弃用的 IP 服务类型（ToS）字段，来携带分类（*类别*）信息。二层数据帧中也同样携带分类信息。
二层数据帧或三层数据包中的 QoS 比特如下图所示：

图 87：数据帧和数据包中的 QoS 分类层

Encapsulated Packet	封装的数据包
Layer 2 header	二层头部
IP header	IP 头部
Data（共 3 处）	数据
Layer 2 ISL Frame	二层 ISL 数据帧
ISL header (26 bytes)	ISL 头部 （26 字节）
Encapsulated frame 1 … (24.5 KB)	封装的数据帧 1…… （24.5 KB）
FCS (4 bytes)	FCS （4 字节）
3 bits used for CoS	3 比特用于 CoS
Layer 2 802.1Q and 802.1p Frame	二层 802.1Q 和 802.1q 数据帧
Preamble	前导码

Start frame delimiter	数据帧开始 分隔符
3 bits used for CoS (user priority)	3 比特用于 CoS（用户优先级）
Layer 3 IPv4 Packet	三层 IPv4 数据包
Version length	版本 长度
ToS (1 byte)	ToS （1 字节）
Len	长度
Offset	偏移
Proto	协议
IP precedence or DSCP（共 2 处）	IP 优先级或 DSCP
Layer 3 IPv6 Packet	三层 IPv6 数据包
Version	版本
Traffic class (1 byte)	流量类别 （1 字节）
Flow label	流 标签
Payload length	负载 长度
Next header	下一个 头部
HOP limit	HOP 限制
Source address	源 地址
Dest. address	目的 地址

二层数据帧的优先级位

二层交换机间链路（ISL）数据帧头部有一个 1 字节的用户字段，其中的 3 个最低有效位用来标记 IEEE 802.1p 服务类别（CoS）值。在配置为二层 ISL 协议的 Trunk 端口上，所有流量都承载在 ISL 数据帧中。

二层 802.1Q 数据帧头部有一个 2 字节的标记控制信息（TCI）字段，其中的 3 个最高有效位用来标记 CoS 值，这 3 个比特也称为用户优先级位。在配置为二层 802.1Q 协议的 Trunk 端口上，除了本征 VLAN（Native VLAN）中的流量外，所有流量都承载在 802.1Q 数据帧中。其他类型的数据帧中不携带二层 CoS 值。

二层 CoS 值的取值范围是从 0 至 7，其中 0 表示最低优先级，7 表示最高优先级。

三层数据包的优先级位

三层 IP 数据包中携带着 IP 优先级值，或者差分服务代码点（DSCP）值。QoS 能够支持使用这两个值，因为 DSCP 值能够向后兼容 IP 优先级值。
IP 优先级值的取值范围是 0 至 7。DSCP 值的取值范围是 0 至 63。

使用分类的端到端 QoS 解决方案

所有接入到 Internet 中的交换机和路由器都依赖类别信息，来为拥有相同类别信息的数据包提供相同的转发行为，为拥有不同类别信息的数据包提供不同的转发行为。数据包中携带的类别信息是基于用户配置的策略和/或设备对于数据包的详细检查做出的，可以由终端用户进行分配，也可以由沿途经过的交换机或路由器进行分配。对于数据包的详细检查行为一般会在靠近网络边缘的位置上执行，这样做不会增加核心交换机和路由器的负担。
数据包传输路径中的交换机和路由器可以使用类别信息来限制某个流量类别所占用的资源总量。在 Diff-Serv 架构中，单台设备对流量的处理行为称作逐跳行为。如果路径中的所有设备都提供统一的逐跳行为，用户就可以构建出端到端的 QoS 解决方案。
在网络中实施 QoS 可以是一项简单的任务，也可以是一项复杂的任务，这取决于联网设备所提供的 QoS 特性、网络中的流量类型和流量模式，以及用户对于入站和出站流量控制的粒度。

数据包分类

数据包分类是指按照确定的规则，把数据包归类为用户定义策略中的某个类别。模块化 QoS CLI（MQC）是一种基于策略分类的语言，策略分类语言能够用来定义一下内容：

- class-map 模版，其中指定一个或几个匹配条件
- policy-map 模版，其中关联一个或几个类别

policy-map 模版之后会关联到交换机的一个或多个接口上。
数据包分类是识别数据包的过程，最终会确定数据包属于 policy-map 中定义的某一个类别。当设备发现数据包与某个类别中指定的过滤器相匹配，这个分类过程就结束了。这也称为第一匹配。如果数据包与策略中的多个类别都匹配，不管 policy-map 中的类别顺序是如何定义的，设备都会在数据包匹配到第一个类别后结束分类过程。
如果数据包与策略中的每个分类都不匹配，它就会被分类为策略中的默认类别中。每个 policy-map 中都有一个默认类别，这是系统定义的类别，会匹配所有与用户定义的类别不相匹配的数据包。
数据包分类特性可以归类为以下类型：

- 根据随数据包传播的信息进行分类
- 根据特定信息进行分类
- 层级式分类

根据随数据包传播的信息进行分类
这种分类方式会基于数据包中的某部分信息进行分类，并且这些信息会随着数据包端到端传输，或者在一些中间设备之间传输，通常这种分类方式会使用以下信息：

- 根据三层或四层头部进行分类

- 根据二层信息进行分类

根据三层或四层头部进行分类

这是最常见的部署环境。三层和四层头部中有很多字段都可以用来进行数据包分类。

以最精细的级别来说，分类技术可以匹配完整的流。对于这种部署类型来说，用户可以使用访问控制列表（ACL）。ACL 可以根据流的不同部分进行匹配（比如只匹配源 IP 地址、只匹配目的 IP 地址，或者同时匹配源和目的 IP 地址）。

用户还可以根据 IP 头部中的优先级或 DSCP 值进行数据包分类。IP 优先级字段用来标识这个处理数据包所需的优先级等级。它由 IP 头部服务类型（ToS）字节中的 3 比特构成。

下面这个表格中列出了不同的 IP 优先级值及其含义。

表 90：IP 优先级值和名称

IP 优先级值	IP 优先级比特	IP 优先级名称
0	000	Routine（普通）
1	001	Priority（优先）
2	010	Immediate（快速）
3	011	Flash（闪速）
4	100	Flash Override（疾速）
5	101	Critical（关键）
6	110	Internetwork Control（网间控制）
7	111	Network Control（网络控制）

注释： 网络中的所有路由控制流量默认都是用 IP 优先级值 6。IP 优先级值 7 也是为网络控制流量预留的。因此不建议把 IP 优先级值 6 和 7 分配给用户流量。

DSCP 字段由 IP 头部中的 6 比特构成，它是由 Internet 工程任务组（IETF）差分服务工作组进行标准化的。最初包含 DSCP 位的 ToS 字节已经被重命名为 DSCP 字节。DSCP 是 IP 头部中的字段，与 IP 优先级类似。DSCP 字段的范围比 IP 优先级字段的范围大，因此 DSCP 字段的描述方式与 IP 优先级值的描述方式类似。

注释： DSCP 字段的定义能够向后兼容 IP 优先级值。

根据二层头部进行分类

用户可以使用多种方法来基于二层头部信息执行数据包分类。最常用的方法如下所示：

- 基于 MAC 地址的分类（只用于 access-group）——基于源 MAC 地址（用来针对入站流量进行限速）和目的 MAC 地址（用来针对出站流量进行限速）进行分类；
- 服务类别——基于二层头部中的 3 比特进行分类，符合 IEEE 802.1p 标准。这个值通常与 IP 头部的 ToS 字段之间有映射关系；
- VLAN ID——基于数据包的 VLAN ID 进行分类

注释： 二层头部中的有些字段也可以通过策略进行设置。

基于设备指定的信息进行分类（QoS 组）

用户也可以不基于数据包头部负载中携带的信息进行分类。

有时用户可能需要把从多个入站接口进入的流量汇聚到一个出站接口的特定类别中。举例来说，可能会有多个客户边界路由器的流量从不同接口进入服务提供商网络并获得相同的服务。服务提供商可能希望对所有汇集起来的语音流量进行限速，使其以指定速率进入核心网。但语音流量可能来自不同的客户，也可能会携带不同的 ToS 设置。基于 QoS 组的分类特性就适用于这种环境。

用户在入站接口配置的策略可以把 QoS 组设置为指定值，然后出站接口上的策略可以使用这个值来对数据包进行分类。

QoS 组是交换机内部的数据包数据结构中的一个字段。需要注意的是，QoS 组是交换机的内

部标签，并不是数据包头部中的一部分。

层级式分类

用户可以基于其他类别来执行数据包分类。通常来说，如果用户需要把两个或多个类别中的分类机制（比如过滤器）结合到一个 class-map 中，就需要使用层级式分类。

QoS 有线模型

要想实施 QoS，用户必须执行以下任务：

- 流量分类——区分数据包或流；
- 流量标记和限速——当数据包在交换机中移动时，QoS 会为其分配一个代表指定服务质量的标签，使设备对于数据包的处理能够够符合用户配置的资源利用限制；
- 列队和调度——在存在资源竞争的环境中为不同流量提供不同服务；
- 整形——确保从交换机发来的流量符合特定的流量模型。

入站端口行为

交换机的入站端口上会发生以下行为：

- 分类——通过把数据包与 QoS 标签进行关联，为数据包分配不同的路径。举例来说，把数据包中的 CoS 或 DSCP 映射为一个 QoS 标签，用来区分不同类型的流量。QoS 标签表明了设备会对这个数据包执行的 QoS 行为；
- 限速——通过把入站流量的速率与用户配置的速率进行对比，限速特性能够确定数据包是符合限定的，还是超出了限定。限速器会限制一个流所使用的带宽。判断结果会发送给标记器；
- 标记——标记特性会根据限速器和配置信息来评估对数据包进行评估，当数据包超出限定条件后，它会评估要对数据包执行的行为，以此决定对数据包采取的具体做法（让数据包不做任何修改地通过、降低数据包中 QoS 标签的等级，或者丢弃数据包）。

出站端口行为

交换机的出站端口上会发生以下行为：

- 限速——通过把入站流量的速率与用户配置的速率进行对比，限速特性能够确定数据包是符合限定的，还是超出了限定。限速器会限制一个流所使用的带宽。判断结果会发送给标记器；
- 标记——标记特性会根据限速器和配置信息来评估对数据包进行评估，当数据包超出限定条件后，它会评估要对数据包执行的行为，以此决定对数据包采取的具体做法（让数据包不做任何修改地通过、降低数据包中 QoS 标签的等级，或者丢弃数据包）；
- 列队——列队特性会对 QoS 数据包标签和相应的 DSCP 或 CoS 值进行评估，然后选择为数据包使用的出向队列。由于当多个入向端口同时向一个出向端口发送数据时会发生拥塞，因此列队特性会根据 QoS 标签，使用加权尾部丢弃（WTD）来区分流量类别，并为数据包分配不同的门限值。如果超出了门限值，数据包就会被丢弃。

分类

分类是通过查看数据包中的字段，来区分流量类型的过程。只有当交换机上启用了 QoS 时，才会启用分类特性。默认情况下，交换机上已启用了 QoS。

在分类过程中，交换机会先执行查找，然后为数据包分配一个 QoS 标签。QoS 标签标识了对数据包执行的所有 QoS 行为，以及应该把数据包发送到哪个队列中。

访问控制列表

用户可以使用 IP 标准、IP 扩展，或二层 MAC ACL 来定义一组拥有相同特征的数据包（类别）。用户也可以基于 IPv6 ACL 来分类 IP 流量。

在 QoS 环境中，访问控制条目（ACE）中的允许（permit）和拒绝（deny）行为与安全 ACL 中的允许和拒绝行为有所不同：

- 如果根据第一匹配原则，数据包匹配的条目中设置了允许行为，设备就会对这个数据包执行这个 QoS 行为；
- 如果数据包匹配的条目中设置了拒绝行为，数据包就会跳出这个 ACL，并由下一个 ACL 进行处理；
- 如果数据包没有匹配任何设置了允许行为的条目，并且与所有 ACE 都进行了匹配，那么这个数据包就不会接受 QoS 处理行为，交换机会对这个数据包提供尽力而为的服务；
- 如果端口上配置了多个 ACL，那么当数据包匹配第一个设置了允许行为的 ACL 时，查找就会结束，QoS 的处理行为就会开始。

注释： 在用户创建访问列表时，要注意默认情况下访问列表中都会包含隐含的拒绝条目，这个条目会在 ACL 末尾匹配所有未能与之前的条目相匹配的数据包。

在 ACL 中定义了流量类别后，用户可以为其关联一个策略。一个策略中可能会包含多个类别，并为每个类别指定不同行为。策略中可能会包含把类别分类为特定汇聚类的命令（比如分配 DSCP），或者对类别进行限速的命令。然后用户会把策略关联到一个端口上，让策略生效。

用户可以使用全局配置命令 **access-list** 配置 IP ACL，来分类 IP 流量；用户可以使用全局配置命令 **mac access-list extended** 配置二层 MAC ACL，来分类非 IP 流量。

class-map

用户可以使用 class-map 机制来为指定流量（或类别）进行命名，并从所有其他流量中把它隔离出来。class-map 定义了用来匹配指定流量的规则，以便将来对流量进行分类。在规则中，流量可以匹配 ACL 定义的 access-group，或者匹配指定的 DSCP 或 IP 优先级值。如果用户希望分类多种流量类型，就可以创建另一个 class-map，并为其使用不同的名称。当数据包匹配了一个 class-map 规则时，用户继而可以使用 policy-map 对其进行分类。

用户可以使用全局配置命令 **class-map** 来创建一个 class-map，或者使用 policy-map 配置命令 **class** 来创建一个 class-map。当会有多个端口共享一个 class-map 时，用户应该使用 **class-map** 命令。在用户输入 **class-map** 命令后，也就进入了 class-map 配置模式。在这个模式中，用户可以使用 class-map 配置命令 **match** 来为流量定义匹配规则。

用户可以通过使用 policy-map 配置命令 **class class-default** 来创建默认类别（class-default）。默认类别是系统定义的，并且不能配置。未分类的流量（不符合流量类别中定义的匹配条件的流量）都会被当作默认流量进行处理。

policy-map

policy-map 定义了应该为流量类别实施的行为。这些行为包括：

- 在流量类别中设置特定的 DSCP 或 IP 优先级值；

- 在流量类别中设置 CoS 值；
- 设置 QoS 组；
- 指定流量带宽限制，以及流量超出限制后的行为。

在一个 policy-map 能够生效前，用户必须把它关联到一个端口上。

用户可以使用全局配置命令 **policy-map** 来创建并命名一个 policy-map。当用户输入这条命令后，也就进入了 policy-map 配置模式。在这个模式中，用户可以使用 policy-map 配置命令 **class**，以及 policy-map 类别配置命令 **set**，来指定要为特定的流量类别实施的行为。

用户也可以使用 policy-map 类别配置命令 **police** 和 **bandwidth** 来配置 policy-map，这个 policy-map 定义了流量的限速器和带宽限制，以及当流量超过限速时采取的行为。除此之外，用户还可以使用 policy-map 类别配置命令 **priority** 来配置 policy-map，并为一个类别调节优先级；或者使用 policy-map 类别列队命令 **queue-buffers** 和 **queue-limit** 来配置 policy-map。

要想启用 policy-map，用户需要使用接口配置命令 **service-policy** 把它关联到一个接口。

物理端口上的 policy-map

用户可以在物理端口上配置非层级式 policy-map，它指定了要对哪个流量类别采取 QoS 行为。行为包括在流量类别中设置特定的 DSCP 和 IP 优先级值，为每个匹配的流量类别指定流量带宽限制（限速器），以及当流量超出限制时采取的行为（标记）。

policy-map 具有以下特征：

- 一个 policy-map 中可以包含多个 class 语句，每个 class 语句拥有不同的匹配条件和限速器；
- 一个 policy-map 中可以包含一个预定义的默认流量类别，这个默认流量类别明确放置在 policy-map 的末尾。

  当用户使用 policy-map 配置命令 **class class-default** 配置默认流量类别时，未分类的流量（不符合流量类别中定义的匹配条件的流量）都会被当作默认流量类别（**class-default**）进行处理。
- 一个端口上对不同类型的流量可以设置不同的 policy-map。

VLAN 上的 policy-map

交换机能够支持 VLAN QoS 特性，使用户能够在 VLAN 级别，使用入站数据帧的 VLAN 信息，来执行 QoS 行为（分类和 QoS 行为）。在基于 VLAN 的 QoS 中，用户可以在 SVI 接口上应用服务策略。属于一个 VLAN policy-map 的所有物理接口都需要调用这个基于 VLAN 的 policy-map，而不是基于端口的 policy-map。

尽管用户是在 VLAN SVI 接口上应用的 policy-map，但限速（速率限制）行为只能基于每个端口来执行。用户不能为多个物理端口的总流量实施限速器。每个端口都有一个单独的限速器，来管理进入这个端口的流量。

限速

当数据包分类完成后，并且数据包上已经分配了 DSCP、CoS 或 QoS 组标签后，就可以开始对其实施限速和标记特性了。

限速特性就是创建一个限速器，来为流量指定带宽限制。超出限制的数据包是*超出限制*或*不合格*的。每个限速器都以数据包为基础来确定数据包是符合限制的，还是超出限制的，并指定数据包的行为。这些行为由标记器来执行，其中包括让数据包不做任何修改地通过、丢弃数据包，或修改（降低）数据包的 DSCP 或 CoS 值并允许数据包通过。

为了避免出现失序数据包，合格流量和不合格流量通常都会从相同的队列发出。

注释： 如果用户配置了限速器，那么所有流量（无论是桥接流量还是路由流量）都受到限

速器的监控。因此，桥接数据包可能会被丢弃，或者当它们被限速和标记时，会被修改 DSCP 或 CoS 字段。
用户只能在物理端口上配置限速特性。
在用户配置了 policy-map 和限速行为后，需要使用接口配置命令 **service-policy**，把策略关联到一个入向端口或 SVI 接口上。

令牌桶算法

限速特性使用了令牌桶算法。当交换机接收到每个数据帧时，都会向桶中添加一个令牌。桶中有一个洞，并以一定的速率向外漏令牌，这个速率是由用户以比特每秒为单位指定的平均流量速率。在每次向桶中添加令牌时，交换机都会确认桶中是否有足够的空间。如果桶中没有足够的空间，相关数据包就会被标记为不合格，然后数据包会接受相应的限速器行为（丢弃或降低优先级）。
桶会多快填满是桶深度（突发字节）、漏出令牌的速度（bit/s 速率），以及平均速率之上突发的周期，所提供的功能。桶的大小限制了突发长度的上限，限制了背板到背板能够传输的数据帧数量。如果突发时间较短，并且桶不会满溢，就不会有任何行为施加在流量上。但如果突发时间较长速率较高，并且使桶变满，限速策略就会被施加在突发中的数据帧上。
用户可以使用 policy-map 类别配置命令 **police** 的突发字节（burst-byte）选项，来配置桶深度（在桶满之前能够承受的最大突发）。用户可以使用 policy-map 类别配置命令 **police** 的速率（rate）选项，来配置漏出令牌的速度（平均速率）。

标记

标记特性负责把特定信息传递到网络中的下游设备上，或者把信息从一个接口传递到另一个接口。
标记特性可以用来设置数据包头部中的特定字段/比特，或者也可以使用标记特性来设置数据包结构中的特定字段，这是交换机的内部信息。除此之外，标记特性也可以用来定义字段之间的映射关系。QoS 可以使用以下标记方法：

- 数据包头部
- 设备指定信息
- table-map

数据包头部标记

标记数据包头部中的字段可以分为以下两类：

- IPv4/IPv6 头部比特标记
- 二层头部比特标记

IP 级别的标记特性可以用来把 IP 头部的 IP 优先级和 DSCP 设置为指定的值，以此在下游设备（交换机或路由器）上实现逐跳行为，或者也可以使用标记特性把不同入站接口汇集的流量，在出站接口分到一个类别中。目前支持对 IPv4 和 IPv6 头部进行标记。
二层头部中的标记通常会被用来影响下游设备（交换机或路由器）中的丢弃行为。它会与二层头部中的匹配信息协同工作。二层头部中可以使用 policy-map 设置的比特是服务类别

（CoS）。

交换机特定信息标记

这种形式的标记行为包括标记数据包数据结构中的字段，这些内容不是数据包头部的一部分，使数据路径中的其他设备也可以使用这个标记。这种标记不在交换机之间传播。对 QoS 组的标记就属于这一类。只有应用在入站接口上的策略才支持这种类型的标记。用户可以在同一台交换机的出站接口上启用相应的匹配机制和相应的 QoS 行为。

table-map 标记

table-map 标记特性能够使用一个转换表，把一个字段映射和转换为另一个字段。这个转换表就称为 table-map。

根据接口上关联的 table-map 不同，数据包的 CoS、DSCP 和 UP 值都可以被重写。交换机上能够同时配置入向 table-map 策略和出向 table-map 策略。

注释： 一个堆栈中总共支持 14 个 table-map。在一个有线端口的一个方向上只支持一个 table-map。

举例来说，一个 table-map 可以用来把二层 CoS 设置映射到三层 IP 优先级值。用户使用这个特性可以在一个 table-map 中设置多个 **set** 命令，**set** 命令指明了执行映射的方法。这个 table-map 可以由多个策略进行调用，或者在一个策略中调用多次。

下面这个表格中列出了当前支持的映射形式：

表 91：用来建立映射关系的数据包标记类型

To（去往）数据包标记类型	**From（来自）数据包标记类型**
优先级	CoS
优先级	QoS 组
DSCP	CoS
DSCP	QoS 组
CoS	优先级
CoS	DSCP
QoS 组	优先级
QoS 组	DSCP

基于 table-map 的策略支持以下功能：

- 突变——用户可以创建一个 table-map，其中记录了从一个 DSCP 值映射为另一个 DSCP 值的设置，这个 table-map 可以关联到出向端口上；
- 重写——根据用户配置的 table-map 来对入站数据包进行重写；
- 映射——基于 table-map 的策略可以代替使用 set 命令的策略。

用户在使用 table-map 进行标记时需要执行以下步骤：

1. 定义 table-map——用户需要使用全局配置命令 **table-map** 来设置值的映射。这个 table-map 对于它会用到的策略或类别一无所知。如果 From 字段中没有匹配信息的话，table-map 中的默认命令会用来指明被复制到 To 字段中的值；
2. 定义 policy-map——用户必须定义 table-map 会使用的 policy-map；
3. 把策略关联到一个接口。

注释： 入站端口上的 table-map 策略会改变端口上的信任设置，把它改为 From 类型的 QoS

标记。

流量调节

为了在网络中支持 QoS，进入服务提供商网络的流量需要在网络边界路由器上执行限速，来确保流量速率保持在服务限制内。即使在网络边界上，只有少部分路由器开始发送比网络核心的部署更多的流量，这写增加的流量也会导致网络的拥塞。网络性能的降低会导致难以为所有网络流量提供 QoS 保障。

流量限速功能（使用限速特性）和整形功能（使用流量整形特性）可以管理流量速率，但它们之间的区别在于当令牌用光时对于流量的处理方式。令牌的概念来自于令牌桶机制，这是一个流量计量功能。

注释： 在对网络流量进行 QoS 测试时，用户可能会在整形数据和限速策略中看到不同的结果。通过整形特性处理的网络流量数据提供了更精确的结果。

下面这个表格对比了限速功能和整形功能。

表 92：限速功能和整形功能的对比

限速功能	整形功能
以线路速率发送合格流量并允许突发	平缓发送流量并以恒定速率发送
令牌用完时马上采取行动	令牌用完时，先把数据包缓存起来，等有令牌可用时再发送。使用了整形特性的类别会关联一个队列，这个队列就是在这种情况中用来缓存数据包的
限速特性中可以配置多种单位——比特每秒、数据包每秒、网元每秒	整形特性中只能配置一种单位——比特每秒
限速特性可以在一个事件上关联多个行为，比如标记和丢弃行为	整形特性不能对不合格数据包进行标记
适用于入站流量和出站流量	只适用于出站流量
传输控制协议（TCP）会以线路速率来测试线路状况，但会在发生丢包后通过减小自己的窗口大小，把速率调整为用户配置的速率	TCP 能够检测到它有一条速率较低的线路，并相应地调整自己的重传计时器。结果是缩小重传范围，这个结果对于 TCP 来说能够接收

限速

QoS 限速特性的作用是为流量类别施加一个最大速率。QoS 限速特性也可以与优先级特性一起使用，来限制优先级流量。如果流量超出了限制速率，限速特性会在这个事件发生时马上执行相应行为。这个速率（承诺信息速率[CIR]和最高信息速率[PIR]）和突发参数（合格的突发大小[B_c]和超出的突发大小[B_e]）在配置时都是以字节每秒为单位的。

QoS 支持以下限速形式或限速器：

- 单速双色限速
- 双速三色限速

注释： 不支持单速三色限速。

但速率双色限速

当用户只配置了一个 CIR 和一个 B_c 时，使用的就是单速双色限速器。

B_c 是个可选参数，如果用户没有指定的话，设备默认会计算出来。在这种模式中，当入站数据包有足够的令牌可用时，限速器就认为数据包是合格的。如果当数据包到达时，在 B_c 的限制范围中没有足够多的可用令牌，限速器就认为数据包超出了用户配置的速率限制。

双速三色限速

在使用双速限速器时，交换机只支持色盲模式。在这个模式中，用户需要配置一个承诺信息速率（CIR）和一个最高信息速率（PIR）。顾名思义，这个模式中使用两个令牌桶，一个用于最高速率，另一个用于合格速率。

注释： 有关令牌桶算法的更多信息，用户可以参考令牌桶算法。

在色盲模式中，入站数据包首先会与最高速率令牌桶进行比较。如果这里没有足够多的令牌可用，限速器就认为数据包违反了速率限制。如果这里有足够多的令牌可用，接着限速器会检查合格速率令牌同种的令牌，以此确定是否有足够多的令牌可用。最高速率令牌桶中的令牌数量会根据数据包的大小而减少。如果合格速率令牌桶中没有足够多的令牌可用，限速器就认为数据包超出了用户配置的速率。如果有足够多的令牌可用，限速器就会认为数据包是合格的，然后这两个桶中的令牌数量都会随数据包的大小而减少。

桶中补充令牌的速度取决于数据包的到达时间。假设有一个数据包在时间 T1 时到达，接着另一个数据包在时间 T2 时到达。T1 和 T2 之间的时间间隔决定了令牌桶中需要添加的令牌数量。计算方式为：

数据包之间的到达时间间隔（T2-T1）* CIR/8 字节

整形

整形是为流量施加最大速率的过程，并且整形的原则是为了让下游交换机和路由器不会遭遇拥塞。整形最常见的形式是用来限制从物理接口或逻辑接口发送流量的速率。

整形特性关联了一个缓存，这个缓存能够确保在没有足够多的令牌可用时，把数据包缓存下来，而不是立马丢弃。能够为整形流量的子集所使用的缓存数量也是有限的，这个值是基于多种参数计算出来的。用户也可以使用特定的 QoS 命令来调整这个缓存数量。当缓存可用时，数据包就会被缓存起来，否则数据包就会被丢弃。

基于类别的流量整形

交换机可以使用基于类别的流量整形特性。这个整形特性是在一个策略中的一个类别上启用的，这个策略与一个接口相关联。配置了整形特性的类别会被分配一个缓存号码，在没有足够多的令牌时用这个缓存来暂时储存数据包。被还存起来的数据包从这个类别中被发出时会使用 FIFO（先进先出）策略。在最常用的模式中，基于类别的整形会被用来为物理接口和逻辑接口整体施加一个最大速率。一个类别中支持以下整形模式：

- 平均速率整形
- 层级式整形

整形特性是使用令牌桶实施的。CIR、B_c 和 B_e 的值决定了数据包的发送速率，以及填充令牌的速率。

注释： 有关令牌桶算法的更多信息，用户可以参考令牌桶算法。

平均速率整形

用户可以使用 policy-map 类别配置命令 **shape average** 来配置平均速率整形特性。

这条命令为一个指定的类别配置了最大带宽。队列带宽会被限制为这个值，哪怕端口有更多的带宽可用。用户可以使用百分比来配置整形平均速率，也可以直接指定目标比特速率值。

层级式整形

用户也可以在层级式模型中，配置多个等级的整形规则。用户可以创建一个父系策略并在其

中配置整形特性，然后在关联子系策略并在子系策略中把其他整形配置关联到父系特性。
用户能够配置以下两种类型的层级式整形特性：

- 端口整形器
- 用户配置的整形

端口整形器使用 class-default，并且只有父系策略中允许的行为才会被整形。端口整形器的列队行为是关联在子系策略中的。在使用用户配置的整形特性时，用户不能在子系策略中设置列队行为。

列队和调度

列队特性和调度特性都有助于预防流量拥塞。交换机支持下列列队和调度特性：

- 带宽
- 加权尾部丢弃
- 优先级队列
- 队列缓存

当用户在一个端口上定义列队策略时，控制数据包是映射在最优的优先级队列中的，使用最高的门限值。在以下环境中，控制数据包队列的映射工作有所不同：

- 不使用服务质量（QoS）策略——如果用户没有配置 QoS 策略的话，携带 DSCP 值 16、24、48 和 56 的控制数据包会被映射到队列 0 中，拥有门限值 2 的最高门限值；
- 使用用户定义的策略——在出向端口上配置的用户定义的列队策略，会影响控制数据包上的默认优先级队列设置。

QoS 特性会根据以下规则把控制流量重定向到最优队列中：

1. 如果用户定义了一个用户策略，最高等级的优先级队列总是会被选为最优队列；
2. 如果没有配置优先级队列，Inspur INOS 软件会选择队列 0 作为最优队列。当软件把队列 0 选择为最优路径时，用户必须为这条队列指定最高带宽，以便为控制平面流量提供最好的 QoS 行为；
3. 如果用户没有在最优队列上配置门限值，Inspur INOS 软件会把携带差分服务代码点（DSCP）值 16、24、48 和 56 的控制数据包映射到门限值 2，并把最优队列中的其余控制流量重新分配到门限值 1。

如果一个策略中没有明确对控制流量实施配置，Inspur INOS 软件会把所有不匹配的控制流量都以门限值 2 映射到最优队列中，匹配的控制流量会按照用户配置的策略，映射到相应的队列中。

注释： 为了对三层数据包提供适当的 QoS，用户必须确保把数据包明确地分类到适当的队列中。当软件在默认队列中检测到 DSCP 值的时候，它会自动把数据包分配到最优队列中。

带宽

交换机能够支持以下带宽配置：

- 带宽百分比
- 带宽剩余率

带宽百分比

用户可以使用 policy-map 类别配置命令 **bandwidth percent** 来为指定类别分配最小带宽。用户配置值的总和不能超过 100%，如果总和小于 100%的话，其他带宽会平均分到所有带宽队

列中。

注释： 如果其他队列没有消耗掉所有端口带宽的话，一条队列可以超额订阅带宽。

用户不能在一个 policy-map 中混合使用不同的带宽类型。举例来说，用户不能在一个 policy-map 中同时以带宽百分比，以及 kbits/s 来配置带宽。

带宽剩余率

用户可以使用 policy-map 类别配置命令 **bandwidth remaining ratio** 创建一个比率，用来在指定队列中分享未使用的带宽。指定队列可以在未使用的带宽中，占用用户配置的比率。用户可以在策略中，为指定队列同时使用这条命令和 **priority** 命令。

在用户分配比率时，队列中会被分配到与这些比率相同的权重。用户可以指定的比率范围是从 0 至 100。举例来说，如果用户为一个类别配置了带宽剩余率 2，为另一个类别的队列分配带宽剩余率 4。那么带宽剩余率 4 在调度的执行中就会以带宽剩余率 2 的两倍来执行。

为策略分配的总带宽率可以超过 100。举例来说，用户可以为一条队列分配带宽剩余率 50，为另一条队列分配带宽剩余率 100。

加权尾部丢弃

出向队列使用称为加权尾部丢弃（WTD）的高级版尾部丢弃拥塞避免机制。WTD 是实施在队列中的，来管理队列长度，为不同的流量类别提供丢弃优先级。

在数据帧被排列如某条队列后，WTD 会使用数据帧中被分配的 QoS 标记来为其设定不同的门限值。如果数据帧超出了对它 QoS 标签设置的门限值标准（目的队列中的空间小于数据帧大小），数据帧就会被丢弃。

每条队列中有三个可配置的门限值。QoS 标签决定了数据帧会使用这三个门限值中的哪一个。

下图展示了 WTD 在一条队列上的操作，这条队列的大小是 1000 个数据帧。用户配置的三个丢弃优先级分别是 40%（400 个数据帧）、60%（600 个数据帧）和 100%（1000 个数据帧）。这些百分比表示的是在 40%门限值的队列中最多可以排列 400 个数据帧、在 60%门限值的队列中最多可以排列 600 个数据帧，以及在 100%门限值的队列中最多可以排列 1000 个数据帧。

图 88：WTD 和队列的操作

（图 88）

在这个示例中，CoS 值 6 比其他 CoS 值都更为重要，因此用户为它分配的丢弃门限值是 100%（队列排满的状态）。CoS 值 4 的门限值是 60%，CoS 值 3 的门限值是 40%。用户使用命令 **queue-limit cos** 来分配这些门限值。

假设队列中已经排入了 600 个数据帧，这时又有一个新的数据帧达到了。这个数据帧携带 CoS 值 4，因此适用于 60%门限值。如果把这个数据帧添加到队列中，就超出了门限值的限制，因此这个数据帧会被丢弃。

加权尾部丢弃默认值

以下内容为加权尾部丢弃（WTD）的默认值，以及配置 WTD 门限值的规则。

- 如果用户为 WTD 配置少于 3 个队列限制百分比，WTD 默认值就会被分配给这些门限值。

  以下为 WTD 门限值的默认值：

  表 93：WTD 门限值的默认值

门限值	默认值百分比

0	80
1	90
2	400

- 如果用户配置了 3 个不同的 WTD 门限值，队列就会按照用户的配置分别对应适当的门限值；
- 如果用户配置了 2 个 WTD 门限值，那么最大的值百分比就会是 400；
- 如果用户配置了 1 个门限值为 x，那么最大的值百分比就会是 400。
  - 如果 x 的值小于 90，那么门限值 1=90，门限值 0=x；
  - 如果 x 的值等于 90，那么门限值 1=90，门限值 0=80；
  - 如果 x 的值大于 90，那么门限值 1=x，门限值 0=80。

优先级队列

每个端口上支持 8 条出向队列，其中 2 条上可以设置优先级。
用户可以使用策略 class-map 命令 **priority level** 来为两个类别配置优先级。一个类别必须配置为优先级队列等级 1，另一个类别必须配置为优先级队列等级 2。这两条队列中的数据包会比其他队列中的数据包拥有更低的延迟。
注释： 用户可以只配置一个等级的优先级。
一个 policy-map 中可以值限定一个优先级或一个优先级等级。一个 policy-map 中可以不使用 kbit/s 为单位，配置多个拥有相同优先级等级的优先级队列，但前提是这些队列中都配置了限速特性。

队列缓存

交换机上的每个千兆端口都有为有线端口分配的 300 缓存。每个万兆端口都分配了 1800 缓存。在启动时，如果有线端口上没有启用 policy-map，会默认创建两条队列。用户可以在有线端口上使用 MQC 策略最多配置 8 条队列。下面这个表格中列出了了哪些数据包会进入到哪条队列中：

表 94：DSCP、优先级和 CoS 值的门限值映射表

DSCP、优先级或 CoS	**队列**	**门限值**
控制数据包	0	2
其他数据包	1	2

注释： 用户可以通过为一条队列设置对其门限值并分配最大可用缓存，来保障缓存的可用性。用户可以使用 policy-map 类别命令 **queue-buffers** 来配置队列缓存。用户可以使用 policy-map 类别命令 **queue-limit** 来配置最大门限值。
用户可以使用两种缓存分配方式：硬缓存，也就是明确地为某条队列保留；软缓存，如果指定端口没有使用这部分缓存的话，其他端口可以使用。
有线端口的默认状态为：队列 0 拥有 40%的缓存，作为硬缓存分配给接口；也就是对于千兆端口来说，有 120 缓存分配给队列 0；对于万兆端口来说，有 720 缓存分配给队列 0。对于千兆端口和万兆端口来说，分配给这条队列的最大软缓存分别设置为 480（计算方式是 120*400/100）和 2880，其中 400 是为任意队列配置的默认最大门限值。
队列 1 上没有分配任何硬缓存，默认软缓存的限制设置为 400（这也是最大门限值）。这个门限值定义了这条队列能够从公共池中借用的软缓存最大数量。

队列缓存分配

用户可以使用 policy-map 类别配置命令 **queue-buffers ratio** 来调整分配给任意队列的缓存。

动态门限值与缩放

传统上，保留的缓存是静态分配给每条队列的。无论队列是否活跃，它的缓存就是由队列自己保留的。除此之外，随着队列数量的增长，每条队列会分配到的保留缓存就会变得越来越小。最终有可能会出现这么一种情况，那就是每条队列分配到的保留缓存都不足以支持一个巨型数据帧。

动态门限值与缩放（DTS）特性提供了一种公平且高效的缓存资源分配方式。当拥塞发生时，DTS 机制会基于全局/端口资源的占用情况，提供弹性的缓存分配方案。从概念上说，DTS 会随着自愿的消耗，逐渐缩小队列缓存分配决策，为其他队列留出空间；反之亦然。这种灵活的方式能够更加有效和公平地利用缓存资源。

如上所述，一条队列中可以配置两个限制条件——硬限制和软限制。

硬限制不是 DTS 的一部分。硬缓存只用于特定队列。硬缓存之和应该小于全局设置的硬缓存最大限制。为所有出向队列设置的全局硬缓存限制当前为 5705。在默认环境中，如果用户没有配置 MQC 策略，24 个千兆端口会占用 24*67=1608，4 个万兆端口会占用 4*720=2880，总共占用 4488 缓存，用户可以根据配置分配更多硬缓存。

软缓存限制是 DTS 特性的一部分。除此之外，有些软缓存的分配结果可以超出全局软缓存分配限制。为所有出向队列设置的全局软缓存分配限制当前为 7607。硬缓存和软缓存的总和加在一起为 13312，也就是 3.4 MB。由于软缓存的分配总和可以超出全局限制，因此当系统负载很低时，一条队列可以使用大量缓存资源。DTS 特性可以在系统的负载变得沉重时，动态调整每条队列中分配的软缓存。

注释： 默认情况下不启用队列 Q0 和 Q1。

注释： 队列 Q2 和 Q3 中的流量使用加权轮循策略。

去往上游方向只有一条队列可用。端口和比率限制只应用在去往下游的方向上。

注释： 有线端口支持 8 条队列。

信任行为

有线端口的信任行为

对于连接在交换机上的有线端口来说（端点设备比如 IP 电话、笔记本电脑、摄像头、网真设备或其他设备），这些端点设备发来的 DSCP、优先级或 CoS 值是受到交换机的信任的，因此在用户没有没有明确配置策略时，这些标记也是会保留的。

这种信任行为同时适用于上游和下游 QoS。

数据包会根据默认的初始配置被排列到适当的队列中。默认交换机上是没有优先级队列的。对于单播和组播数据包来说都是如此。

在入站数据包类型与出站数据包类型不同的情况中，信任行为和列队行为的解释详见下表。需要注意的是，一个端口默认的信任模式是基于 DSCP 值的。如果入站数据包是一个纯二层数据包的话，信任模式会“后退”为 CoS 值。用户也可以把信任设置从 DSCP 变更为 CoS。用户可以使用 MQC 策略来完成这个设置变更，也就是在 class-default 中执行“set cos cos table default default-cos”行为，其中 default-cos 是用户创建的 table-map 的名称（它只执行默认复制）。

表 95：信任和队列行为

入站数据包	出站数据包	信任行为	队列行为
三层	三层	保留 DSCP/优先级	基于 DSCP
二层	二层	不适用	基于 CoS
标记	标记	保留 DSCP 和 CoS	基于 DSCP（信任 DSCP 游先于优先级）
三层	标记	保留 DSCP，CoS 设置为 0	基于 DSCP

上述有线端口的信任默认设置再找个版本的软件中也是相同的。为了兼容已有的有线标准，默认所有流量都会排入尽力而为的队列。在下游方向上，维护语音、视频、尽力而为和背景队列的接入点会实施队列特性。接入点会根据 802.11e 标记信息来选择队列策略。

在信任边界上为 Inspur IP 电话提供的端口安全

在典型的网络中，用户把一台 IP 电话连接到一个端口，并且级联一台设备，这台设备会从电话背后生成携带数据信息的数据包。Inspur IP 电话会通过以下行为确保语音质量：它在通过共享的数据链路发送语音数据包时，会把 CoS 等级标记为高优先级（CoS=5），并且把携带数据的数据包标记为低优先级（CoS=0）。电话发往交换机的流量通常会携带标记，这个标记会携带在 802.1Q 头部中。这个头部中还会包含 VLAN 信息和服务类别（CoS）的 3 比特字段，这个字段标明了数据包的优先级。

对于大多数 Inspur IP 电话的配置来说，电话发送的流量应该收到信息，以此来为语音流量提供比网络中其他类型的流量更高的优先级。通过使用接口配置命令 **trust device**，用户可以让连接电话的端口信任自己接收到的流量。

注释： 接口配置命令 **trust device** *device_type* 是设备上的单机命令。当用户在 Auto-QoS 的配置中使用这条命令时，如果端口连接的对等体设备不是对应设备（也就是定义为符合网络中信任策略的设别）的话，CoS 和 DSCP 值都会设置为“0”，并且入站策略也不会生效。如果连接的对等体设备是对应设备的话，入站策略就会生效。

在使用信任设置时，用户还可以使用信任边界特性，来防止滥用高优先级队列，比如用户绕过电话把 PC 直接连接在交换机上。如果没有设置信任边界的话，交换机是会信任 PC 生成的 CoS 标签的（因为设置了信任 CoS）。相反，信任边界特性会使用 CDP 来检测端口上连接的设备是不是 Inspur IP 电话（比如 Inspur IP 电话 7910、7935、7940 和 7960）。如果没有检测到电话，信任边界特性就会禁用这个端口上的信任设置，防止有人滥用高优先级队列。需要注意的是，如果 PC 和 Inspur IP 电话通过集线器连接到交换机，信任边界特性就不会生效。

相关主题

为设备类型配置信任行为

标准 QoS 的默认设置

默认的有线 QoS 配置

交换机的每个有线接口上默认配置了两条队列。所有控制流量都是由队列 0 进行处理的。所有其他流量都是由队列 1 进行处理的。

DSCP 映射

默认的 CoS 到 DSCP 映射

用户可以使用 CoS 到 DSCP 映射，把入站数据包携带的 CoS 值映射为 QoS 用来在内部表示流量优先级的 DSCP 值。下面这个表格中展示了默认的 CoS 到 DSCP 映射。如果这些值不适用于用户网络，用户就需要自行修改。

表 96：默认的 CoS 到 DSCP 映射

CoS 值	**DSCP 值**
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

默认的 IP 优先级到 DSCP 映射

用户可以使用 IP 优先级到 DSCP 映射，把入站数据包携带的 IP 优先级值映射为 QoS 用来在内部表示流量优先级的 DSCP 值。下面这个表格中展示了默认的 IP 优先级到 DSCP 映射。如果这些值不适用于用户网络，用户就需要自行修改。

表 97：默认的 IP 优先级到 DSCP 映射

IP 优先级值	**DSCP 值**
0	0
1	8
2	16
3	24
4	32
5	40
6	48
7	56

默认的 DSCP 到 CoS 映射

用户可以使用 DSCP 到 CoS 映射来生成一个 CoS 值，这个值用来在四条出向队列中选择一条。下面这个表格中展示了默认的 DSCP 到 CoS 映射。如果这些值不适用于用户网络，用户就需要自行修改。

表 96：默认的 DSCP 到 CoS 映射

DSCP 值	**CoS 值**
0～7	0
8～15	1
16～23	2
24～31	3
32～39	4
40～47	5
48～55	6
56～63	7

QoS 策略的指导

用户应该遵循以下指导，防止客户端设备由于不正确的 QoS 策略而被排除在网络之外：

- 在向设备上添加新的 QoS 策略时，同一个漫游域或移动域中的其他设备上也应该配置名称相同的 QoS 策略；
- 当一台设备上加载了一个较新的软件版本时，它能够支持新的策略格式。如果用户把软件镜像从一个较老的版本升级为一个较新的版本，用户应该分别保存配置。在设备加载较老的版本镜像时，可能有些 QoS 策略这个版本无法支持，这时用户就应该把这些 QoS 策略重新恢复为这个软件版本所支持的策略格式。

有线目标上 QoS 的限制条件

目标指的是能够在其上应用策略的实体。用户可以把策略应用到一个有线目标上。有线目标可以是端口或 VLAN。只有端口、SSID 和客户端策略是用户可以配置的。射频策略是用户无法配置的。

并且支持客户端目标。

在设备上为有线目标应用 QoS 特性时，有以下限制条件：

- 连接有线目标的设备端口上最多可以支持 8 条队列；
- 连接有线目标的有线端口上，每个策略中最多支持 63 个限速器；
- QoS 层级中最多支持两个等级；
- 在层级式策略中，父系和子系之间不允许发生覆盖行为，除非父系策略中定义了端口整形器，且子系策略中定义了队列特性；
- QoS 策略不能与任何 EtherChannel 接口相关联；
- 层级式 QoS 策略中的父系策略和子系策略都不支持限速；
- 层级式 QoS 策略中的父系策略和子系策略都不支持标记；
- 一个策略中不支持混用队列限制和队列缓存；

注释： 设备商不支持队列限制百分比，因为命令 **queue-buffer** 负责控制这个功能。只支持使用 DSCP 和 CoS 值来定义队列限制。

- 在使用整形特性时，每个数据包上都有一个 20 字节的 IPG 负载，这是在硬件内部计算的。整形特性实际上会受到它的影响，尤其是对于小数据包来说；
- 在所有上行有线端口（万兆以太网端口）上，所有基于队列的有线策略中分类顺序应该相同，在所有下行有线端口（千兆以太网端口）上也应该相同；
- 不支持空类别；
- 不支持行为为空的 class-map。如果两个策略中定义了顺序相同的 class-map，并且其中一个策略的 class-map 中没有指定行为，就有可能遇到流量丢弃事件。解决方法是为优先级队列（PRIORITY_QUEUE）中的所有类别分配最小带宽；
- 在连接有线目标的有线端口上，每个策略支持最多 256 个类别；
- policy-map 中的限速器行为拥有以下限制条件：
  - 合格流量的行为必须是传输；
  - 超出/违反行为在降低优先级标记时只能使用 CoS 到 CoS、优先级到优先级、DSCP 到 DSCP 类型的标记；

  - 一个策略中的降低优先级标记类型必须一致。
- 端口级别的标记策略会优先于 SVI 接口上的策略；但如果用户没有配置端口策略，SVI 策略就会被优先考虑。要想让端口策略获得优先权，用户需要定义一个端口级别的策略；这样 SVI 的策略就会被覆盖；
- 使用分类计数器拥有以下特殊限制条件：
  - 分类计数器会统计数据包，而不是字节；
  - 不支持基于过滤器的分类计数器；
  - 只有与标记特性或限速特性相关的 QoS 配置才会触发分类计数器；
  - 分类计数器并不是基于端口的。也就是说分类计数器会汇集不同接口上的，属于同一个策略中同一个类别的所有数据包；
  - 如果用户在策略中应用了限速行为或标记行为，class-default 类别中就会使用分类计数器；
  - 如果一个类别中有多个匹配条件，分类计数器只会显示匹配一个条件的流量。
- 使用 table-map 拥有以下限制条件：
  - 针对一个目标在一个方向上，只支持使用一个 table-map 对超出限速特性的流量进行降低优先级的标记，只支持使用一个 table-map 对违反限速特性的流量进行降低优先级的标记；
  - 用户必须在 class-default 下配置 table-map；用户定义的类别中不支持 table-map。
- 使用层级式策略拥有以下需求：
  - 端口整形器
  - 汇聚限速器
  - PV 策略
  - 父系整形和子系标记/限速
- 对于连接有线目标的端口来说，只支持以下层级式策略：
  - 同一个策略中不支持使用限速链；
  - 同一个策略中不支持层级式队列特性（端口整形器除外）；
  - 在父系类别中，所有过滤器的类型必须相同。子系过滤器类型必须与父系过滤器类型相匹配，以下几点是例外：
    - 如果父系类别中配置了匹配 IP，那么子系类别中可以配置匹配 ACL；
    - 如果父系类别中配置了匹配 CoS，那么子系类别中可以配置匹配 ACL。
  - 接口配置模式中的命令 **trust device** *device_type* 是交换机上的独立命令。在 AutoQoS 配置中使用这条命令时，如果连接的对等体设备不是对应设备（也就是符合用户信任策略的设备），那么 CoS 和 DSCP 值都会设置为“0”，任何入站策略也不会生效。如果连接的对等体设备是对应设备，那么入站策略就会生效。

在 VLAN 中向有线目标应用 QoS 特性时有以下限制条件：

- 对于扁平或非层级式策略来说，只支持标记特性或 table-map。

在 EtherChannel 和 EtherChannel 成员接口上应用 QoS 特性是有以下限制条件和考量因素：

- EtherChannel 接口上不支持 QoS 特性；
- EtherChannel 成员接口的入方向上和出方向上都支持 QoS 特性。所有 EtherChannel 成员必须都应用相同的 QoS 策略。如果 QoS 策略不相同，那么不同链路上的策略会独立生效；
- 当用户在向 EtherChannel 成员上应用服务策略时，会看到以下警告消息，这个消息提示用户要在这个 EtherChannel 中所有端口上都应用相同的策略：“Warning: add service policy will cause inconsistency with port xxx in ether channel xxx.”；

- EtherChannel 成员接口上不支持 Auto-QoS 特性。

注释： 在用户向 EtherChannel 上添加服务策略时，会在控制台上看到以下信息："Warning: add service policy will cause inconsistency with port xxx in ether channel xxx."。这个警告消息是正常的。这个消息是为了提醒用户要在这个 EtherChannel 中的其他端口上也应用相同的策略。在启动时也会看到同一条消息。这条消息并不意味着 EtherChannel 成员端口之间有什么差异。

如何配置 QoS

配置类别、策略和 table-map

创建一个流量类别（CLI）

要想创建一个包含有匹配条件的流量类别，用户需要使用 **class-map** 命令来指定流量类别的名称，然后在 class-map 配置模式中，按照需要配置 **match** 命令。

在开始前

这个配置任务中配置的所有 match 命令都是可选的，但用户必须在一个类别中至少配置一条 match 条件。

总步骤

1. configure terminal

2. class-map {*class-map name* | **match-any**}

3. match access-group {*index number* | *name*}

4. match class-map *class-map name*

5. match cos *cos value*

6. match dscp *dscp value*

7. match ip {**dscp** *dscp value* | **precedence** *precedence value* }

8. match non-client-nrt

9. match qos-group *qos group value*

10. match vlan *vlan value*

11. match wlan user-priority *wlan value*

12. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 2	**class-map** {*class-map name*	**match-any**} **示例：**	进入 class-map 配置模式。 • 创建一个 class-map，用来把数据包匹配到用户定义的类别中 • 如果用户指定了 **match-any**，那么

	Device(config)# **class-map test_1000** Device(config-cmap)#	流量必须至少与其中一个条件相匹配，这样这个流量才会被分类到这个流量类别中。这是默认设置
步骤 3	**match access-group** {*index number* | *name*} **示例：** Device(config-cmap)# **match access-group 100** Device(config-cmap)#	用户可以在这条命令中配置以下参数： • access-group • class-map • cos • dscp • ip • non-client-nrt • precedence • qos-group • vlan • wlan user priority （可选）在示例中，用户输入了 access-group ID： • 访问列表索引（取值从 1 至 2799） • 命名的访问列表
步骤 4	**match class-map** *class-map name* **示例：** Device(config-cmap)# **match class-map test_2000** Device(config-cmap)#	（可选）匹配另一个 class-map 的名称
步骤 5	**match cos** *cos value* **示例：** Device(config-cmap)# **match cos 2 3 4 5** Device(config-cmap)#	（可选）匹配 IEEE 802.1Q 或 ISL 服务类别（用户）优先级值。 • 最多输入 4 个 CoS 值，以空格分隔（取值从 0 至 7）
步骤 6	**match dscp** *dscp value* **示例：** Device(config-cmap)# **match dscp af11 af12** Device(config-cmap)#	（可选）匹配 IPv4 和 IPv6 数据包中的 DSCP 值
步骤 7	**match ip** {**dscp** *dscp value* | **precedence** *precedence value*} **示例：** Device(config-cmap)# **match ip dscp af11 af12**	（可选）匹配以下 IP 值： • **dscp**——匹配 IP DSCP（差分服务代码点） • **precedence**——匹配 IP 优先级（取值 0 至 7）

	Device(config-cmap)#	
步骤 8	**match non-client-nrt**	
步骤 9	**match qos-group** *qos group value* **示例：** Device(config-cmap)# **match qos-group 10** Device(config-cmap)#	（可选）匹配 QoS 组值（取值 0 至 31）
步骤 10	**match vlan** *vlan value* **示例：** Device(config-cmap)# **match vlan 210** Device(config-cmap)#	（可选）匹配 VLAN ID（取值 1 至 4095）
步骤 11	**match wlan user-priority** *wlan value*	
步骤 12	**end** **示例：** Device(config-cmap)# **end**	返回特权 EXEC 模式

接下来做什么？

配置 policy-map。

创建一个流量策略（CLI）

要想创建一个流量策略，用户需要使用全局配置命令 **policy-map** 来指定流量策略名称。用户需要使用 **class** 命令，把流量类别关联到流量策略中。用户必须在进入 policy-map 配置模式后再使用 **class** 命令。在输入 **class** 命令后，用户会自动进入到 policy-map 类别配置模式中，并在这里为这个流量策略定义 QoS 策略。

用户可以在 policy-map 中配置以下与类别相关的行为：

- admit——允许请求呼叫准入控制（CAC）
- bandwidth——带宽配置选项
- exit——离开 QoS 类别行为控制模式
- no——反向执行命令或恢复默认值
- police——限速器配置选项
- priority——为这个类别严格指定调度优先级配置选项
- queue-buffers——队列缓存配置选项
- queue-limit——为加权尾部丢弃（WTD）设置队列最大门限值的配置选项
- service-policy——配置 QoS 服务策略
- set——使用以下选项来设置 QoS 值：
  - CoS 值
  - DSCP 值
  - 优先级值
  - QoS 组值

- WLAN 值

- shape——流量整形配置选项

在开始前

用户应该首先创建一个 class-map。

总步骤

1. configure terminal

2. policy-map *policy-map name*

3. class {*class-name* | **class-default**}

4. admit

5. bandwidth {**kb/s** *kb/s value* | **percent** *percentage* | **remaining** {*percent* | *ratio*}}

6. exit

7. no

8. police {*target_bit_rate* | **cir** | **rate**}

9. priority {*kb/s* | **level** *level value* | **percent** *percentage value*}

10. queue-buffers ratio *ratio limit*

11. queue-limit {*packets* | **cos** | **dscp** | **percent**}

12. service-policy *policy-map name*

13. set {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan**}

14. shape average {*target _bit_rate* | **percent**}

15. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**policy-map** *policy-map name*
	
	示例：
	`Device(config)# policy-map test_2000`
	`Device(config-pmap)#`
		进入 policy-map 配置模式。
		创建或修改一个 policy-map，用户可以把它关联到一个或多个接口，用来指定服务策略

步骤 3	**class** {*class-name*	**class-default**} **示例：** Device(config-pmap)# **class test_1000** Device(config-pmap-c)#	指定用户想要创建或更改的类别的名称。 用户也可以为未分类数据包创建一个系统默认类别
步骤 4	**admit**	
步骤 5	**bandwidth** {**kb/s** *kb/s value*	**percent** *percentage*	**remaining** {*percent*	*ratio*}} **示例：** `Device(config-pmap-c)#`	（可选）使用以下选项之一来设置带宽： • **kb/s**——千比特每秒，输入 20000 至 10000000 之间的数值 • **percent**——输入要为这个 policy-map 使用的总带宽的百分比

	bandwidth 50 Device(config-pmap-c)#	• **remaining**——输入剩余带宽的百分比 有关这条命令及其用法的更多信息，用户可以参考： 配置带宽（CLI）
步骤 6	**exit** **示例：** Device(config-pmap-c)# **exit** Device(config-pmap-c)#	（可选）离开 QoS 类别行为配置模式
步骤 7	**no** **示例：** Device(config-pmap-c)# **no** Device(config-pmap-c)#	（可选）反向执行命令
步骤 8	**police** {*target_bit_rate* | **cir** | **rate**} **示例：** Device(config-pmap-c)# **police 100000** Device(config-pmap-c)#	（可选）配置限速器： • *target_bit_rate*——输入每秒比特率，输入 8000 至 10000000000 之间的数值 • **cir**——承诺信息速率 • **rate**——指定限速速率，为层级式策略使用 PCR，或为单级别 ATM 4.0 限速器策略使用 SCR 有关这条命令及其用法的更多信息，用户可以参考： 配置限速特性（CLI）
步骤 9	**priority** {*kb/s* | **level** *level value* | **percent** *percentage value*} **示例：** Device(config-pmap-c)# **priority percent 50** Device(config-pmap-c)#	（可选）为这个类别严格指定调度优先级配置选项。命令选项包括： • *kb/s*——千比特每秒，输入 1 至 2000000 之间的数值 • **level**——建立一个多等级优先级队列。输入一个值（1 或 2） • **percent**——输入用于这个优先级的总带宽百分比 有关这条命令及其用法的更多信息，用户可以参考： 配置优先级（CLI）
步骤 10	**queue-buffers ratio** *ratio limit* **示例：** Device(config-pmap-c)# **queue-buffers ratio 10** Device(config-pmap-c)#	（可选）为这个类别配置队列缓存。输入队列缓存比率限制（0 至 100） 有关这条命令及其用法的更多信息，用户可以参考： 配置队列缓存（CLI）
步骤 11	**queue-limit** {*packets* | **cos** | **dscp** |	（可选）为尾部丢弃指定队列最大门

	percent} **示例：** `Device(config-pmap-c)#` **`queue-limit cos 7 percent 50`** `Device(config-pmap-c)#`	限值： • *packets*——默认配置数据包，输入 1 至 2000000 之间的数值 • **cos**——为每个 CoS 值输入参数 • **dscp**——为每个 DSCP 值输入参数 • **percent**——为门限值输入百分比 有关这条命令及其用法的更多信息，用户可以参考： 配置队列限制（CLI）
步骤 12	**service-policy** *policy-map name* **示例：** `Device(config-pmap-c)#` **`service-policy test_2000`** `Device(config-pmap-c)#`	（可选）配置 QoS 服务策略
步骤 13	**set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan**} **示例：** `Device(config-pmap-c)#` **`set cos 7`** `Device(config-pmap-c)#`	（可选）设置 QoS 值。用户可以配置的 QoS 值包括以下这些： • **cos**——设置 IEEE 802.1Q/ISL 类别的服务/用户优先级 • **dscp**——设置 IPv4 和 IPv4 数据包中的 DSCP • **ip**——设置 IP 特定的值 • **precedence**——设置 IPv4 和 IPv6 数据包中的优先级值 • **qos-group**——设置 QoS 组 • **wlan**——设置 WLAN 用户优先级

步骤 14	**shape average** {*target _bit_rate* | **percent**}
	
	示例：
	`Device(config-pmap-c)`
	`#`**`shape average percent 50`**
	`Device(config-pmap-c) #`
		（可选）设置流量整形特性。命令参数包括以下这些：
		• *target_bit_rate*——目标比特速率
		• **percent**——为承诺信息速率设置接口带宽的百分比
		有关这条命令及其用法的更多信息，用户可以参考：
		配置整形特性（CLI）

步骤 15	**end** **示例：** `Device(config-pmap-c) #`**`end`** `Device(config-pmap-c) #`	返回特权 EXEC 模式

接下来做什么？

配置接口。

配置基于类别的数据包标记（CLI）

接下来的步骤解释了如何在用户交换机上配置下列基于类别的数据包标记特性：

- CoS 值
- DSCP 值
- IP 值
- 优先级值
- QoS 组值
- WLAN 值

在开始前

在开始这部分配置之前，用户应该已经创建了 class-map 和 policy-map。

总步骤

1. configure terminal

2. policy-map *policy name*

3. class *class name*

4. set cos {*cos value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name*}

5. set dscp {*dscp value* | **default** | **dscp table** *table-map name* | **ef** | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name*}

6. set ip {**dscp** | **precedence**}

7. set precedence {*precedence value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name*}

8. set qos-group {*qos-group value* | **dscp table** *table-map name* | **precedence table** *table-map name*}

~~**9. set wlan user-priority** {*wlan user-priority value* | **cos table** *table-map name* | **dscp table** *table-map name* | **qos-group table** *table-map name* | **wlan table** *table-map name*}~~

10. end

11. show policy-map

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**policy-map** *policy name*
	
	示例：
	`Device(config)# policy-map policy1`
	`Device(config-pmap)#`
		进入 policy-map 配置模式。
		创建或修改一个 policy-map，用户可以把它关联到一个或多个接口，用来指定服务策略

步骤 3	**class** *class name* **示例：**	进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。 用户可以在策略 class-map 配置模式

	Device(config-pmap)# **class class1** Device(config-pmap-c)#	中指定以下选项： • **admit——**允许请求呼叫准入控制（CAC） • **bandwidth**——带宽配置选项 • **exit**——离开 QoS 类别行为控制模式 • **no**——反向执行命令或恢复默认值 • **police**——限速器配置选项 • **priority**——为这个类别严格指定调度优先级配置选项 • **queue-buffers**——队列缓存配置选项 • **queue-limit**——为加权尾部丢弃（WTD）设置队列最大门限值的配置选项 • **service-policy**——配置 QoS 服务策略 • **set**——使用以下选项来设置 QoS 值： · CoS 值 · DSCP 值 · 优先级值 · QoS 组值 · WLAN 值 • **shape——**流量整形配置选项 **注释：** 这个步骤描述了用户可以使用的 **set** 命令选项。其他命令选项（**admit**、**bandwidth** 等）会在这个指导的其他部分进行描述。尽管这个任务中列出了所有可用的 **set** 命令，但每个类别中只能配置一个 **set** 命令
步骤 4	**set cos** {*cos value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name*} **示例：** Device(config-pmap)# **set cos 5** Device(config-pmap)#	（可选）为出站数据包设置特定的 IEEE 802.1Q 二层 CoS 值。取值范围是 0 至 7. 用户还可以使用 **set cos** 命令来设置以下值： • **cos table**——基于 table-map 设置 CoS 值 • **dscp table**——基于 table-map 设置代码点值 • **precedence table**——基于 table-map 设置代码点值 • **qos-group table**——基于 table-

		map 把 QoS 组设置为 CoS 值 • **~~wlan user priority table~~**~~——基于 table-map 把 WLAN 用户优先级设置为 CoS 值~~

步骤 5	**set dscp** {*dscp value* | **default** | **dscp table** *table-map name* | **ef** | **precedence table** *table-map name* | **qos-group table** *table-map name* | **wlan user-priority table** *table-map name*}
	
	示例：
	`Device(config-pmap)# set dscp af11`
	`Device(config-pmap)#`
		（可选）设置 DSCP 值。
		除了能够设置具体的 DSCP 值之外，用户还能使用 **set dscp** 命令来设置以下参数：
		• **default**——使用默认 DSCP 值匹配数据包（000000）
		• **dscp table**——基于 table-map 把 DSCP 值设置为数据包 DSCP 值
		• **ef**——使用 EF DSCP 值（101110）来匹配数据包
		• **precedence table**——基于 table-map 把优先级值设置为数据包 DSCP 值
		• **qos-group table**——基于 table-map 把 QoS 组设置为数据包 DSCP 值
		• **~~wlan user priority table~~**~~——基于 table-map 把 WLAN 用户优先级设置为数据包 DSCP 值~~

步骤 6	**set ip** {**dscp** | **precedence**} **示例：** Device(config-pmap)# **set ip dscp c3** Device(config-pmap)#	（可选）设置 IP 特定值。这些值可以是 IPDSCP 值，也可以是 IP 优先级值。用户可以使用 **set ip dscp** 命令来设置以下值： • *dscp value*——设置具体的 DSCP 值 • **default**——使用默认 DSCP 值匹配数据包（000000） • **dscp table**——基于 table-map 把 DSCP 值设置为数据包 DSCP 值 • **ef**——使用 EF DSCP 值（101110）来匹配数据包 • **precedence table**——基于 table-map 把优先级值设置为数据包 DSCP 值 • **qos-group table**——基于 table-map 把 QoS 组设置为数据包 DSCP 值 • **~~wlan user priority table~~**~~——基于 table-map 把 WLAN 用户优先级设置为数据包 DSCP 值~~

		用户可以使用 **set ip precedence** 命令来设置以下值： • *precedence value*——设置优先级值（取值为 0 至 7） • **cos table**——基于 table-map 把二层 CoS 设置为优先级值 • **dscp table**——基于 table-map 把 DSCP 值设置为优先级值 • **precedence table**——基于 table-map 把优先级设置为优先级值 • **qos-group table**——基于 table-map 把 QoS 组设置为优先级值

步骤 7	**set precedence** {*precedence value* | **cos table** *table-map name* | **dscp table** *table-map name* | **precedence table** *table-map name* | **qos-group table** *table-map name*}
	
	示例：
	`Device(config-pmap)# set precedence 5`
	`Device(config-pmap)#`
		（可选）设置 IPv4 和 IPv6 数据包中的优先级值。
		用户可以使用 **set precedence** 命令设置以下值：
		• *precedence value*——设置优先级值（取值为 0 至 7）
		• **cos table**——基于 table-map 把二层 CoS 设置为优先级值
		• **dscp table**——基于 table-map 把 DSCP 值设置为优先级值
		• **precedence table**——基于 table-map 把优先级设置为优先级值
		• **qos-group table**——基于 table-map 把 QoS 组设置为优先级值

步骤 8	**set qos-group** {*qos-group value* | **dscp table** *table-map name* | **precedence table** *table-map name*}
	
	示例：
	`Device(config-pmap)# set qos-group 10`
	`Device(config-pmap)#`
		（可选）设置 QoS 组值。用户可以使用这条命令设置以下值：
		• *qos-group value*——设置 1 至 31 之间的数值
		• **dscp table**——基于 table-map 把 DSCP 值设置为代码点值
		• **precedence table**——基于 table-map 把优先级设置为代码点值

~~**步骤 9**~~	~~**set wlan user-priority** {*wlan user-priority value* | **cos table** *table-map name* | **dscp table** *table-map name* | **qos-group table** *table-map name* | **wlan table** *table-map name*}~~	

步骤 10	**end**
	
	示例：
	`Device(config-pmap)# end`
	`Device#`
		返回特权 EXEC 模式

步骤 11	**show policy-map** **示例：** `Device# show policy-map`	（可选）显示为所有服务策略配置的所有类别策略配置信息

接下来做什么？

使用 **service-policy** 命令把流量策略关联到一个接口。

为语音和视频配置 class-map（CLI）

用户可以按照以下步骤，来为语音和视频流量配置 class-map。

总步骤

1. configure terminal

2. class-map *class-map-name*

3. match dscp *dscp-value-for-voice*

4. end

5. configure terminal

6. class-map *class-map-name*

7. match dscp *dscp-value-for-video*

8. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 2	**class-map** *class-map-name* **示例：** `Device(config)# class-map voice`	创建一个 class-map
步骤 3	**match dscp** *dscp-value-for-voice* **示例：** `Device(config-cmap)# match dscp 46`	匹配 IPv4 和 IPv6 数据包中的 DSCP 值。把这个值设置为 6
步骤 4	**end** **示例：** `Device(config-cmap)# end`	返回特权 EXEC 模式。或者用户也可以使用 **Ctrl-Z** 退出全局配置模式
步骤 5	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 6	**class-map** *class-map-name* **示例：** `Device(config)# class-map video`	创建一个 class-map
步骤 7	**match dscp** *dscp-value-for-video* **示例：** `Device(config-cmap)# match dscp 34`	匹配 IPv4 和 IPv6 数据包中的 DSCP 值。把这个值设置为 34
步骤 8	**end** **示例：** `Device(config-cmap)# end`	返回特权 EXEC 模式。或者用户也可以使用 **Ctrl-Z** 退出全局配置模式

把流量策略关联到一个接口（CLI）

在创建了流量类别和流量策略后，用户必须使用接口配置命令 **service-policy**，把流量策略关联到接口上，并且指定这个策略应该执行的方向（针对进入接口的数据包，还是针对离开接口的数据包）。

在开始前

用户在能够把流量策略关联到接口前，必须先创建流量类别和流量策略。

总步骤

1. configure terminal

2. interface *type*

3. service-policy {**input** *policy-map* | **output** *policy-map* }

4. end

5. show policy map

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**interface** *type*
	
	示例：
	`Device(config)# interface GigabitEthernet1/0/1`
	`Device(config-if)#`
		进入接口配置模式并对接口进行配置。
		用户可以在接口配置模式中设置的命令参数如下所示：
		• **Auto Template**——auto-template 接口
		• **Capwap**——CAPWAP 隧道接口
		• **GigabitEhternet**——千兆以太网 IEEE 802

		• **GroupVI**——组虚拟接口 • **Internal Interface**——内部接口 • **Loopback**——环回接口 • **Null**——空接口 • **Port-Channel**——EtherChannel 接口 • **TenGigabitEthernet**——万兆以太网 • **Tunnel**——隧道接口 • **Vlan**——Inspur VLAN • **Range**——接口范围
步骤 3	**service-policy** {**input** *policy-map* | **output** *policy-map*} **示例：** Device(config-if)# **service-policy output policy_map_01** Device(config-if)#	在接口的入方向上或出方向上关联一个 policy-map。这个 policy-map 会被用作这个接口的服务策略。 在示例中，流量策略会评估所有离开接口的流量
步骤 4	**end** **示例：** `Device(config-if)# end`	返回特权 EXEC 模式
步骤 5	**show policy map** **示例：** `Device# show policy map`	（可选）显示制定接口上策略的状态统计信息

接下来做什么？

继续在接口上应用其他流量策略，并且指定策略应该在哪个方向上执行。

在物理端口上使用 policy-map 实现分类、限速和标记流量（CLI）

用户可以在物理端口上配置非层级式的 policy-map，以此指定要对哪类流量类别实施 QoS 行为。行为包括标记和限速。

在开始前

在开始这部分介绍的配置步骤前，用户应该已经使用 policy-map 确定了网络流量的分类、限速和标记。

总步骤

class-map {*class-map name* | **match-any** }

3. match access-group { *access list index* | *access list name* }

4. policy-map *policy-map-name*

5. class {*class-map-name* | **class-default**}

6. set {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}

7. police {*target_bit_rate* | **cir** | **rate** }

8. exit

9. exit

10. interface *interface-id*

11. service-policy input *policy-map-name*

12. end

13. show policy-map [*policy-map-name* [**class** *class-map-name*]]

14. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**class-map** {*class-map name* | **match-any**}
	
	示例：
	`Device(config)# class-map ipclass1`
	`Device(config-cmap)# exit`
	`Device(config)#`
		进入 class-map 配置模式。
		• 创建一个 class-map，用来把数据包匹配到用户定义的类别中
		• 如果用户指定了 **match-any**，那么流量必须至少与其中一个条件相匹配，这样这个流量才会被分类到这个流量类别中。这是默认设置

步骤 3	**match access-group** {*access list index* | *access list name*}
	
	示例：
	`Device(config-cmap)# match access-group 100`
	`Device(config-cmap)# exit`
	`Device(config)#`
		指定匹配这个 class-map 的分类条件。用户可以在这条命令中配置以下参数：
		• **access-group**——匹配 access-group
		• **class-map**——匹配另一个 class-map
		• **cos**——匹配一个 CoS 值
		• **dscp**——匹配一个 DSCP 值
		• **ip**——匹配指定 IP 值
		• **non-client-nrt**——匹配非客户端 NRT
		• **precedence**——匹配 IPv4 和 IPv6 数据包中的优先级
		• **qos-group**——匹配一个 QoS 组
		• **vlan**——匹配一个 VLAN

步骤 4	**policy-map***policy-map-name*
	
	示例：
	`Device(config)# policy-map flowit`
	`Device(config-pmap)#`
		通过输入 policy-map 的名称创建一个 policy-map，并进入 policy-map 配置模式。
		默认设备中没有定义 policy-map

步骤 5	**class** {*class-map-name* | **class-default**}
	
	示例：
	`Device(config-pmap)# class ipclass1`
	`Device(config-pmap-c)#`
		定义流量分类，并进入 policy-map 类别配置模式。
		默认设备中没有定义 policy-map 和 class-map。
		如果用户已经通过全局配置命令 **class-map** 定义了流量类别，就可以直接在这个命令中指定 *class-map-name*。**class-default** 流量类别是预定义的，用户可以把它添加到任何策略中。这个流量类别总是会被放在 policy-map 末尾。在 **class-default** 类别中有隐含的 **match any** 语句，所有没有与其他流量类别相匹配的数据包都会匹配 **class-default**

步骤 6	**set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | ~~**wlan user priority**~~}
	
	示例：
	`Device(config-pmap-c)# set dscp 45`
	`Device(config-pmap-c)#`
		（可选）设置 QoS 值。用户可以设置的 QoS 参数如下所示：
		• **cos**——设置 IEEE 802.1Q/ISL 类别的服务/用户优先级
		• **dscp**——设置 IPv4 和 IPv4 数据包中的 DSCP
		• **ip**——设置 IP 特定的值
		• **precedence**——设置 IPv4 和 IPv6 数据包中的优先级值
		• **qos-group**——设置 QoS 组
		• ~~**wlan**——设置 WLAN 用户优先级~~
		在示例中，**set dscp** 命令通过在数据包中设置一个新的 DSCP 值，来分类 IP 流量

步骤 7	**police** {*target_bit_rate* | **cir** | **rate**}
	
	示例：
	`Device(config-pmap-c)#`
	`police 100000 conform-action transmit exceed-action drop`
	`Device(config-pmap-c)#`
		（可选）配置限速器：
		• *target_bit_rate*——输入每秒比特率，输入 8000 至 10000000000 之间的数值
		• **cir**——承诺信息速率
		• **rate**——指定限速速率，为层级式策略使用 PCR，或为单级别 ATM 4.0 限速器策略使用 SCR
		在示例中，**police** 命令把限速器添加到类别中，超出了用户设置的目标比特率（100000）的流量都会被丢弃

步骤 8	**exit** **示例：** `Device(config-pmap-c)# exit`	返回 policy-map 配置模式
步骤 9	**exit**	返回全局配置模式

	示例： `Device(config-pmap)# exit`	
步骤 10	**interface** *interface-id* 示例： `Device(config)# interface gigabitethernet 2/0/1`	指定要应用 policy-map 的接口并进入接口配置模式。 有效的接口包括物理端口
步骤 11	**service-policy input** *policy-map-name* 示例： `Device(config-if)# service-policy input flowit`	指定 policy-map 名称，并把它应用到入向端口上。一个入向端口上只支持一个 policy-map
步骤 12	**end** 示例： `Device(config-if)# end`	返回特权 EXEC 模式
步骤 13	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] 示例： `Device# show policy-map`	（可选）确认用户的配置
步骤 14	**copy running-config startup-config** 示例： `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

接下来做什么？

如果需要的话，用户可以在 QoS 配置中使用 policy-map，在 SVI 接口上配置分类、限速和标记。

在 SVI 上使用 policy-map 实现分类、限速和标记（CLI）

在开始前

在开始这部分介绍的配置步骤前，用户应该已经使用 policy-map 确定了网络流量的分类、限速和标记。

总步骤

1. configure terminal

2. class-map {*class-map name* | **match-any** }

3. match vlan *vlan number*

4. policy-map *policy-map-name*

5. description *description*

6. class {*class-map-name* | **class-default**}

7. set {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | **wlan user-priority**}
8. police {*target_bit_rate* | **cir** | **rate**}
9. exit
10. exit
11. interface *interface-id*
12. service-policy input *policy-map-name*
13. end
14. show policy-map [*policy-map-name* [**class** *class-map-name*]]
15. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 2	**class-map** {*class-map name* | **match-any**} **示例：** `Device(config)# class-map class_vlan100`	进入 class-map 配置模式。 • 创建一个 class-map，用来把数据包匹配到用户定义的类别中 • 如果用户指定了 **match-any**，那么流量必须至少与其中一个条件相匹配，这样这个流量才会被分类到这个流量类别中。这是默认设置

步骤 3	**match vlan** *vlan number*
	
	示例：
	`Device(config-cmap)# match vlan 100`
	`Device(config-cmap)# exit`
	`Device(config)#`
		指定匹配到这个 class-map 的 VLAN

步骤 4	**policy-map** *policy-map-name*
	
	示例：
	`Device(config)# policy-map policy_vlan100`
	`Device(config-pmap)#`
		通过输入 policy-map 的名称创建一个 policy-map，并进入 policy-map 配置模式。
		默认设备中没有定义 policy-map

步骤 5	**description** *description*
	
	示例：
	`Device(config-pmap)#`
	`description vlan 100`
		（可选）为 policy-map 输入一个描述信息

步骤 6	**class** {*class-map-name* | **class-default**} **示例：**	定义流量分类，并进入 policy-map 类别配置模式。 默认设备中没有定义 policy-map 和

	Device(config-pmap)# **class class_vlan100** Device(config-pmap-c)#	class-map。 如果用户已经通过全局配置命令 **class-map** 定义了流量类别，就可以直接在这个命令中指定 *class-map-name*。**class-default** 流量类别是预定义的，用户可以把它添加到任何策略中。这个流量类别总是会被放在 policy-map 末尾。在 **class-default** 类别中有隐含的 **match any** 语句，所有没有与其他流量类别相匹配的数据包都会匹配 **class-default**
步骤 7	**set** {**cos** | **dscp** | **ip** | **precedence** | **qos-group** | ~~**wlan user priority**~~} **示例：** Device(config-pmap-c)# **set dscp af23** Device(config-pmap-c)#	（可选）设置 QoS 值。用户可以设置的 QoS 参数如下所示： • **cos**——设置 IEEE 802.1Q/ISL 类别的服务/用户优先级 • **dscp**——设置 IPv4 和 IPv4 数据包中的 DSCP • **ip**——设置 IP 特定的值 • **precedence**——设置 IPv4 和 IPv6 数据包中的优先级值 • **qos-group**——设置 QoS 组 • ~~**wlan**——设置 WLAN 用户优先级~~ 在示例中，**set dscp** 命令通过匹配 DSCP 值 AF23（010010），来分类 IP 流量
步骤 8	**police** {*target_bit_rate* | **cir** | **rate**} **示例：** Device(config-pmap-c)# **police 200000 conform-action transmit exceed-action drop** Device(config-pmap-c)#	（可选）配置限速器： • *target_bit_rate*——输入每秒比特率，输入 8000 至 10000000000 之间的数值 • **cir**——承诺信息速率 • **rate**——指定限速速率，为层级式策略使用 PCR，或为单级别 ATM 4.0 限速器策略使用 SCR 在示例中，**police** 命令把限速器添加到类别中，超出了用户设置的目标比特率（200000）的流量都会被丢弃
步骤 9	**exit** **示例：** Device(config-pmap-c)# **exit**	返回 policy-map 配置模式
步骤 10	**exit** **示例：** Device(config-pmap)# **exit**	返回全局配置模式
步骤 11	**interface** *interface-id*	指定要应用 policy-map 的接口并进入

	示例： `Device(config)# interface gigabitethernet 1/0/3`	接口配置模式。 有效的接口包括物理端口
步骤 12	**service-policy input** *policy-map-name* **示例：** `Device(config-if)# service-policy input policy_vlan100`	指定 policy-map 名称，并把它应用到入向端口上。一个入向端口上只支持一个 policy-map
步骤 13	**end** **示例：** `Device(config-if)# end`	返回特权 EXEC 模式
步骤 14	**show policy-map** [*policy-map-name* [**class** *class-map-name*]] **示例：** `Device# show policy-map`	（可选）确认用户的配置
步骤 15	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

配置 table-map（CLI）

table-map 是标记特性所使用的配置格式，还使用表格提供了一个字段到另一个字段的映射和转换。举例来说，用户可以使用 table-map 把二层 CoS 设置映射和转换为三层优先级值。

注释： 多个策略可以同时调用一个 table-map，或者一个策略中可以多次调用一个 table-map。

总步骤

1. configure terminal

2. table-map *name* {**default** {*default value* | **copy** | **ignore**} | **exit** | **map** {**from** *from value* **to** *to value* } | **no**}

3. map from *value* **to** *value*

4. exit

5. exit

6. show table-map

7. configure terminal

8. policy-map

9. class class-default

10. set cos dscp table *table map name*

11. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**table-map** *name* {**default** {*default value* | **copy** | **ignore**} | **exit** | **map** {**from** *from value* **to** *to value* } | **no**}
	
	示例：
	`Device(config)# table-map table01`
	`Device(config-tablemap)#`
		创建一个 table-map 并进入 table-map 配置模式。在 table-map 配置模式中，用户可以输入以下命令：
		• **default**——配置 table-map 的默认值，或者为 table-map 中没有的值设置默认行为：复制或忽略
		• **exit**——退出 table-map 配置模式
		• **map**——在 table-map 中把 *from* 值影射为 *to* 值
		• **no**——反向执行的命令或者恢复命令默认值

步骤 3	**map from** *value* **to** *value*
	
	示例：
	`Device(config-tablemap)#`
	`map from 0 to 2`
	`Device(config-tablemap)#`
	`map from 1 to 4`
	`Device(config-tablemap)#`
	`map from 24 to 3`
	`Device(config-tablemap)#`
	`map from 40 to 6`
	`Device(config-tablemap)#`
	`default 0`
	`Device(config-tablemap)#`
		在这一步骤中，数据包中的 DSCP 值 0 会被标记为 CoS 值 2，DSCP 值 1 会被标记为 CoS 值 4，DSCP 值 24 会被标记为 CoS 值 3，DSCP 值 40 会被标记为 CoS 值 6，其他值标记为 CoS 值 0。
		注释： 这个示例使用 policy-map 类别配置命令 **set** 把 CoS 值映射为 DSCP 值

步骤 4	**exit**
	
	示例：
	`Device(config-tablemap)#`
	`exit`
	`Device(config)#`
		返回全局配置模式

步骤 5	**exit**
	
	示例：
	`Device(config) exit`
	`Device#`
		返回特权 EXEC 模式

步骤 6	**show table-map**	显示 table-map 的配置

	示例： Device# **show table-map** Table Map table01 from 0 to 2 from 1 to 4 from 24 to 3 from 40 to 6 default 0	
步骤 7	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 8	**policy-map** **示例：** Device(config)# **policy-map table-policy** Device(config-pmap)#	为这个 table-map 配置 policy-map
步骤 9	**class class-default** **示例：** Device(config-pmap)# **class class-default** Device(config-pmap-c)#	把类别匹配到系统默认
步骤 10	**set cos dscp table** *table map name* **示例：** Device(config-pmap-c)# **set cos dscp table table01** Device(config-pmap-c)#	如果用户把这个策略应用在入站端口，那么这个端口上就启用了 DSCP 信任，并且会根据指定 table-map 来进行标记
步骤 11	**end** **示例：** Device(config-pmap-c)# **end** Device#	返回特权 EXEC 模式

接下来做什么？

为用户网络中的 QoS 策略配置其他 policy-map。在创建了 policy-map 后，使用 **service-policy** 命令把一个或多个流量策略关联到接口上。

配置信任特性

配置 QoS 特性和功能

配置呼叫准入控制（CLI）

以下示例展示了如何在设备上为呼叫准入控制（CAC），配置基于类别的无条件数据包标记特性。

总步骤

1. configure terminal
2. class-map *class name*
3. match dscp *dscp value*
4. exit
5. class-map *class name*
6. match dscp *dscp value*
7. exit
8. table-map *name*
9. default copy
10. exit
11. table-map *name*
12. default copy
13. exit
14. policy-map *policy name*
15. class *class-map-name*
16. priority level *level_value*
17. police [*target_bit_rate* | **cir** | **rate**]
18. admit cac wmm-tspec
19. rate *value*

21. exit
22. exit
23. class *class name*
24. priority level *level_value*
25. police [*target_bit_rate* | **cir** | **rate**]
26. admit cac wmm-tspec
27. rate *value*

29. exit
30. exit
31. policy-map *policy name*
32. class *class-map-name*
33. set dscp dscp table *table_map_name*

35. shape average {*target bit rate* | **percent** *percentage*}

36. queue-buffers {**ratio** *ratio value*}

37. service-policy *policy_map_name*

38. end

39. show policy-map

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**class-map** *class name*
	
	示例：
	`Device(config)# class-map voice`
	`Device(config-cmap)#`
		进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。策略 class-map 配置模式中的命令选项如下所示：
		• *word*——class-map 名称
		• **class-default**——系统默认类别，匹配所有未分类的数据包

步骤 3	**match dscp** *dscp value* **示例：** `Device(config-cmap)# match dscp 46`	（可选）匹配 IPv4 和 IPv6 数据包中的 DSCP 值

步骤 4	**exit**
	
	示例：
	`Device(config-cmap)# exit`
	`Device(config)#`
		返回全局配置模式

步骤 5	**class-map** *class name*
	
	示例：
	`Device(config)# class-map video`
	`Device(config-cmap)#`
		进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。策略 class-map 配置模式中的命令选项如下所示：
		• *word*——class-map 名称
		• **class-default**——系统默认类别，匹配所有未分类的数据包

步骤 6	**match dscp** *dscp value* **示例：** `Device(config-cmap)# match dscp 34`	（可选）匹配 IPv4 和 IPv6 数据包中的 DSCP 值
步骤 7	**exit** **示例：**	返回全局配置模式

	Device(config-cmap)# **exit** Device(config)#	
步骤 8	**table-map** *name* **示例：** Device(config)# **table-map dscp2dscp** Device(config-tablemap)#	创建一个 table-map 并进入 table-map 配置模式
步骤 9	**default copy** **示例：** Device(config-tablemap)# **default copy**	针对 table-map 中没有的值设置默认行为：复制。 **注释：** 这是默认选项。用户也可以设置 DSCP 值到 DSCP 值的映射
步骤 10	**exit** **示例：** Device(config-tablemap)# **exit** Device(config)#	返回全局配置模式
步骤 11	**table-map** *name* **示例：** Device(config)# **table-map dscp2up** Device(config-tablemap)#	创建一个 table-map 并进入 table-map 配置模式
步骤 12	**default copy** **示例：** Device(config-tablemap)# **default copy**	针对 table-map 中没有的值设置默认行为：复制。 **注释：** 这是默认选项。用户也可以设置 DSCP 值到 UP 值的映射
步骤 13	**exit** **示例：** Device(config-tablemap)# **exit** Device(config)#	返回全局配置模式
步骤 14	**policy-map** *policy name* **示例：** Device(config)# **policy-map ssid_child_cac** Device(config-pmap)#	进入 policy-map 配置模式。 创建或修改一个 policy-map，用户可以把这个 policy-map 关联到一个或多个接口来指定一个服务策略
步骤 15	**class** *class-map-name* **示例：** Device(config-pmap)# **class voice**	定义一个接口级别的流量分类，并进入 policy-map 类别配置模式
步骤	**priority level** *level_value*	使用 **priority** 命令为类别分配严

16	**示例：** Device(config-pmap-c)# **priority level 1**	格的调度优先级。 **注释：** 优先级等级 1 比优先级等级 2 更重要。QoS 会首先处理优先级等级 1 预留的带宽，因此它的延迟最低。优先级等级 1 和 2 都会预留带宽
步骤 17	**police** [*target_bit_rate* | **cir** | **rate**] **示例：** Device(config-pmap-c)# **police cir 10m**	（可选）配置限速器： • *target_bit_rate*——输入每秒比特率，输入 8000 至 10000000000 之间的数值 • **cir**——承诺信息速率 • **rate**——指定限速速率，为层级式策略使用 PCR，或为单级别 ATM 4.0 限速器策略使用 SCR
步骤 18	**admit cac wmm-tspec**	为这个 policy-map 配置呼叫准入控制
步骤 19	**rate** *value* **示例：** Device(config-pmap-admit-cac-wmm)# **rate 5000**	配置目标比特速率（千比特每秒）。输入 8 至 10000000 之间的数值
~~**步骤 20**~~	~~**wlan-up** *value*~~ ~~**示例：**~~ ~~Device(config-pmap-admit-cac-wmm)# **wlan-up 6 7**~~	~~配置 WLAN UP 值。输入 0 至 7 之间的数值~~
步骤 21	**exit** **示例：** Device(config-pmap-admit-cac-wmm)# **exit** Device(config-pmap-c)#	返回 policy-map 类别配置模式
步骤 22	**exit** **示例：** Device(config-pmap-c)# **exit** Device(config-pmap)#	返回 policy-map 配置模式
步骤 23	**class-map** *class name* **示例：** Device(config)# **class-map video** Device(config-cmap)#	进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。策略 class-map 配置模式中的命令选项如下所示： • *word*——class-map 名称 • **class-default**——系统默认

		类别，匹配所有未分类的数据包
步骤 24	**priority level** *level_value* **示例：** Device(config-pmap-c)# **priority level 2**	使用 **priority** 命令为类别分配严格的调度优先级。 **注释：** 优先级等级 1 比优先级等级 2 更重要。QoS 会首先处理优先级等级 1 预留的带宽，因此它的延迟最低。优先级等级 1 和 2 都会预留带宽
步骤 25	**police** [*target_bit_rate* | **cir** | **rate**] **示例：** Device(config-pmap-c)# **police cir 20m**	（可选）配置限速器： • *target_bit_rate*——输入每秒比特率，输入 8000 至 10000000000 之间的数值 • **cir**——承诺信息速率 • **rate**——指定限速速率，为层级式策略使用 PCR，或为单级别 ATM 4.0 限速器策略使用 SCR
~~**步骤 26**~~	~~**admit cac wmm-tspec**~~ ~~**示例：**~~ ~~Device(config-pmap-c)# **admit cac wmm-tspec**~~ ~~Device(config-pmap-admit-cac-wmm)#~~	~~为这个 policy-map 配置呼叫准入控制。~~ ~~**注释：**这条命令只为无线 QoS 配置 CAC~~
步骤 27	**rate** *value* **示例：** Device(config-pmap-admit-cac-wmm)# **rate 5000**	配置目标比特速率（千比特每秒）。输入 8 至 10000000 之间的数值
~~**步骤 28**~~	~~**wlan-up** *value*~~ ~~**示例：**~~ ~~Device(config-pmap-admit-cac-wmm)# **wlan-up 4 5**~~	~~配置 WLAN UP 值。输入 0 至 7 之间的数值~~
步骤 29	**exit** **示例：** Device(config-pmap-admit-cac-wmm)# **exit** Device(config-pmap)#	返回 policy-map 配置模式
步骤 30	**exit** **示例：**	返回全局配置模式

	Device(config-pmap)# **exit** Device(config)#	
步 骤 31	**policy-map** *policy name* **示例：** Device(config)# **policy-map ssid_cac** Device(config-pmap)#	进入 policy-map 配置模式。 创建或修改一个 policy-map，用户可以把这个 policy-map 关联到一个或多个接口来指定一个服务策略
步 骤 32	**class-map** *class name* **示例：** Device(config)# **class-map default**	定义接口级别的流量分类并进入 policy-map 配置模式。 这个示例把 class-map 设置为默认
步 骤 33	**set dscp dscp table** *table_map_name* **示例：** Device(config-pmap-c)# **set dscp dscp table dscp2dscp**	（可选）设置 QoS 值。这个示例使用 **set dscp dscp table** 命令创建一个 table-map 并设定它的值
~~步 骤 34~~	~~**set wlan user-priority dscp table** *table_map_name*~~ ~~**示例：**~~ ~~Device(config-pmap-c)# **set wlan user-priority dscp table dscp2up**~~	~~（可选）设置 QoS 值。这个示例使用命令 **set wlan user-prority dscp table** 设置了 WLAN 用户优先级~~
步 骤 35	**shape average** {*target bit rate* | **percent** *percentage*} **示例：** Device(config-pmap-c)# **shape average 100000000**	配置平均整形速率。用户可以通过设置目标比特速率（比特每秒）来设置平均整形速率，也可以通过设置接口带宽的承诺信息速率（CIR）百分比来设置平均整形速率
步 骤 36	**queue-buffers** {**ratio** *ratio value*} **示例：** Device(config-pmap-c)# **queue-buffers ratio 0**	为队列配置相应的缓存大小。 **注释：** 一个策略中配置的所有缓存之和必须小于或等于 100%。未分配的缓存会被平均分到所有剩余队列中
步 骤 37	**service-policy** *policy_map_name* **示例：** Device(config-pmap-c)# **service-policy ssid_child_cac**	为服务策略指定 policy-map
步 骤 38	**end** **示例：** Device(config-pmap-c)# **end** Device#	返回特权 EXEC 模式

步骤 39	**show policy-map** **示例：** `Device# show policy-map`	（可选）显示为所有服务策略配置的所有类别策略配置信息

接下来做什么？

为用户网络中的 QoS 策略配置其他 policy-map。在创建了 policy-map 后，使用 **service-policy** 命令把一个或多个流量策略关联到接口上。

更多有关 CAC 的信息，用户可以参考 *System Management Configuration Guide, Inspur INOS (Inspur 6650 Switches)*。

配置带宽（CLI）

下面这个流程解释了如何在用户交换机上配置带宽。

在开始前

在开始这部分介绍的配置步骤前，用户应该已经为带宽创建了 class-map。

总步骤

1. configure terminal

2. policy-map *policy name*

3. class *class name*

4. bandwidth {*Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio* }}

5. end

6. show policy-map

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**policy-map** *policy name*
	
	示例：
	`Device(config)# policy-map policy_bandwidth01`
	`Device(config-pmap)#`
		进入 policy-map 配置模式。
		创建或修改一个 policy-map，用户可以把它关联到一个或多个接口，用来指定服务策略

步骤 3	**class** *class name*
	
	示例：
	`Device(config-pmap)# class class_bandwidth01`
	`Device(config-cmap-c)#`
		进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。策略 class-map 配置模式中的命令选项如下所示：
		• *word*——class-map 名称
		• **class-default**——系统默认类别，匹配所有未分类的数据包

步骤 4	**bandwidth** {*Kb/s*	**percent** *percentage*	**remaining** { **ratio** *ratio* }}	为 policy-map 配置带宽。参数如下所示：

	示例： `Device(config-pmap-c)#` **`bandwidth 200000`** `Device(config-pmap-c)#`	• *Kb/s*——千比特每秒，输入 20000 至 10000000 之间的数值 • **percent**——根据百分比为某个类别分配最小带宽。如果其他队列没有用到全部端口带宽的话，这条队列可以超额订阅带宽。总和不能超过 100%，如果小于 100%的话，剩余带宽会被平均分配到所有带宽队列中 • **remaining**——为某个类别分配最小带宽。如果其他队列没有用到全部端口带宽的话，这条队列可以超额订阅带宽。总和不能超过 100%。如果策略中的某条队列上使用了 **priority** 命令，那么用户最好也是用这条命令。用户也可以为每条队列分配速率而不是百分比；这样每条队列就会获得特定的加权，加权值与这些比率相关联。比率的取值范围是 0 至 100。在这个示例中，为策略分配的总带宽比率可以超过 100。 **注释：** 用户不能在一个 policy-map 中混用带宽类型。举例来说，用户不能在一个 policy-map 中同时使用带宽百分比和千比特每秒来配置带宽
步骤 5	**end** **示例：** `Device(config-pmap-c)#` **`end`** `Device#`	返回特权 EXEC 模式
步骤 6	**show policy-map** **示例：** `Device#` **`show policy-map`**	（可选）显示为所有服务策略配置的所有类别策略配置信息

接下来做什么？

为用户网络中的 QoS 策略配置其他 policy-map。在创建了 policy-map 后，使用 **service-policy** 命令把一个或多个流量策略关联到接口上。

配置限速（CLI）

下面这个示例解释了如何在用户交换机上配置限速特性。

在开始前

在开始这部分介绍的配置步骤前，用户应该已经为限速创建了 class-map。

总步骤

1. configure terminal

2. policy-map *policy name*

3. class *class name*

4. police {*target_bit_rate* [*burst bytes* | **bc** | **conform-action** | **pir**] | **cir** {*target_bit_rate* | **percent** *percentage*} | **rate** {*target_bit_rate* | **percent** *percentage*} **conform-action transmit exceed-action** {**drop** [**violate action**] | **set-cos-transmit** | **set-dscp-transmit** | **set-prec-transmit** | **transmit** [**violate action**] }}

5. end

6. show policy-map

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**policy-map** *policy name*
	
	示例：
	`Device(config)# policy-map policy_police01`
	`Device(config-pmap)#`
		进入 policy-map 配置模式。
		创建或修改一个 policy-map，用户可以把它关联到一个或多个接口，用来指定服务策略

步骤 3	**class** *class name*
	
	示例：
	`Device(config-pmap)# class class_police01`
	`Device(config-cmap-c)#`
		进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。策略 class-map 配置模式中的命令选项如下所示：
		• *word*——class-map 名称
		• **class-default**——系统默认类别，匹配所有未分类的数据包

步骤 4	**police** {*target_bit_rate* [*burst bytes*	**bc**	**conform-action**	**pir**]	**cir** {*target_bit_rate*	**percent** *percentage*}	**rate** {*target_bit_rate*	**percent** *percentage*} **conform-action transmit exceed-action** {**drop** [**violate action**]	**set-cos-transmit**	**set-dscp-transmit**	**set-prec-transmit**	**transmit** [**violate action**] }} **示例：** `Device(config-pmap-c)# police 8000 conform-action transmit exceed-action drop` `Device(config-pmap-c)#`	用户可以在这条命令中配置以下 **police** 子命令： • *target_bit_rate*——比特每秒（取值为 8000 至 10000000000） · *burst bytes*——输入 1000 至 512000000 之间的数值 · **bc**——合格突发 · **conform-action**——当速率低于合格突发时的行为 · **pir**——最高信息速率 • **cir**——承诺信息速率 · *target_bit_rate*——比特每秒（取值为 8000 至 10000000000） · **percent**——接口带宽 CIR 的百

		分比 • **rate**——指定限速速率，为层级式策略指定 PCR，或者为单等级 ATM 4.0 限速器策略指定 SCR。 · *target_bit_rate*——比特每秒（取值为 8000 至 10000000000） · **percent**——接口带宽 CIR 的百分比 用户可以配置以下 **police conform-actiontransmit exceed-action** 子命令选项： • **drop**——丢弃数据包 • **set-cos-transmit**——设置 CoS 值并发送 • **set-dscp-transmit**——设置 DSCP 值并发送 • **set-prec-transmit**——重写数据包的优先级并发送 • **transmit**——传输数据包 **注释：** 基于限速器的降低优先级行为只支持使用 table-map。在交换机中，每个标记字段只能有一个降低优先级 table-map

步骤 5	**end**
	
	示例：
	`Device(config-pmap-c)# end`
	`Device#`
		返回特权 EXEC 模式

步骤 6	**show policy-map** **示例：** `Device# show policy-map`	（可选）显示为所有服务策略配置的所有类别策略配置信息

接下来做什么？

为用户网络中的 QoS 策略配置其他 policy-map。在创建了 policy-map 后，使用 **service-policy** 命令把一个或多个流量策略关联到接口上。

配置优先级（CLI）

以下示例解释了如何在用户交换机上配置优先级特性。

用户可以为指定队列分配优先级。用户可以指定两个优先级等级（1 和 2）。

注释： 支持语音和视频的队列应该分配优先级等级 1。

在开始前

在开始这部分介绍的配置步骤前，用户应该已经为优先级创建了 class-map。

总步骤

1. configure terminal

2. policy-map *policy name*

3. class *class name*

4. priority [*Kb/s* [*burst_in_bytes*] | **level** *level_value* [*Kb/s* [*burst_in_bytes*] | **percent** *percentage* [*burst_in_bytes*]] | **percent** *percentage* [*burst_in_bytes*]]

5. end

6. show policy-map

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**policy-map** *policy name*
	
	示例：
	`Device(config)# policy-map policy_priority01`
	`Device(config-pmap)#`
		进入 policy-map 配置模式。
		创建或修改一个 policy-map，用户可以把它关联到一个或多个接口，用来指定服务策略

步骤 3	**class** *class name*
	
	示例：
	`Device(config-pmap)# class class_priority01`
	`Device(config-cmap-c)#`
		进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。策略 class-map 配置模式中的命令选项如下所示：
		• *word*——class-map 名称
		• **class-default**——系统默认类别，匹配所有未分类的数据包

步骤 4	**priority** [*Kb/s* [*burst_in_bytes*]	**level** *level_value* [*Kb/s* [*burst_in_bytes*]	**percent** *percentage* [*burst_in_bytes*]]	**percent** *percentage* [*burst_in_bytes*]]
				
				示例：
				`Device(config-pmap-c)# priority level 1`
				`Device(config-pmap-c)#`
					（可选）用户可以使用 **priority** 命令为这个类别严格指定调度优先级配置选项。
					命令中包含的选项如下所示：
					• *Kb/s*——千比特每秒（取值为 1 至 2000000）
					· *burst_in_bytes*——指定突发字节（取值为 32 至 2000000）
					• **level** *level_value*——指定多等级（1-2）优先级队列
					· *Kb/s*——千比特每秒（取值为 1 至 2000000）
					· *burst_in_bytes*——指定突发字节（取值为 32 至 2000000）
					· **percent**——总带宽的百分比
					· *burst_in_bytes*——指定

		突发字节（取值为 32 至 2000000） • **percent**——总带宽的百分比 · 指定突发字节（取值为 32 至 2000000） **注释：** 优先级等级 1 比优先级等级 2 更重要。QoS 会首先处理优先级等级 1 预留的带宽，因此它的延迟最低。优先级等级 1 和 2 都会预留带宽

步骤 5	**end**
	
	示例：
	`Device(config-pmap-c)# end`
	`Device#`
		返回特权 EXEC 模式

步骤 6	**show policy-map** **示例：** `Device# show policy-map`	（可选）显示为所有服务策略配置的所有类别策略配置信息

接下来做什么？

为用户网络中的 QoS 策略配置其他 policy-map。在创建了 policy-map 后，使用 **service-policy** 命令把一个或多个流量策略关联到接口上。

配置队列和整形

配置出向队列特征

根据用户网络的复杂性和 QoS 解决方案，用户可能需要执行这一部分的所有步骤。用户需要对以下这些特征做出判断：

- 哪些数据包由 DSCP 值、CoS 值或 QoS 组值映射到每条队列和门限值 ID？
- 要对这些队列应用的丢弃百分比门限值是多少，以及需要为流量类别保留的最大内存是多少？
- 要为队列分配多少固定缓存空间？
- 是否要限制端口上的带宽速率？
- 出向队列能够接收服务的频率是多少，以及应该使用哪种技术（整形、共享或两者都使用）？

注释： 用户可以只在交换机上配置出向队列。

配置队列缓存（CLI）

用户可以为队列分配缓存。如果用户没有分配缓存的话，所有缓存会平均分给所有队列。用户可以使用队列缓存比率（queue-buffer ratio），以特定的比率来分配缓存。默认 DTS（动态门限值与缩放）特性是对所有队列启用的，这些属于软缓存。

注释： 有线端口上支持队列缓存比率（queue-buffer ratio）配置，但队列缓存比率（queue-buffer ratio）不能和队列限制（queue-limit）一起配置。

在开始前

执行这个配置步骤有以下先决条件：

- 在开始这部分介绍的配置步骤前，用户应该已经为队列缓存创建了 class-map；

- 在配置队列缓存前，用户必须已经在 policy-map 中配置了带宽、整形或优先级。

总步骤

1. configure terminal

2. policy-map *policy name*

3. class *class name*

4. bandwidth {*Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio value* }}

5. queue-buffers {**ratio** *ratio value*}

6. end

7. show policy-map

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**policy-map** *policy name*
	
	示例：
	`Device(config)# policy-map policy_queuebuffer01`
	`Device(config-pmap)#`
		进入 policy-map 配置模式。
		创建或修改一个 policy-map，用户可以把它关联到一个或多个接口，用来指定服务策略

步骤 3	**class** *class name*
	
	示例：
	`Device(config-pmap)# class class_queuebuffer01`
	`Device(config-cmap-c)#`
		进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。策略 class-map 配置模式中的命令选项如下所示：
		• *word*——class-map 名称
		• **class-default**——系统默认类别，匹配所有未分类的数据包

步骤 4	**bandwidth** {*Kb/s*	**percent** *percentage*	**remaining** { **ratio** *ratio value* }} **示例：** Device(config-pmap-c)# **bandwidth percent 80** Device(config-pmap-c)#	为这个 policy-map 配置带宽。用户可以使用的命令参数如下所示： • *Kb/s*——千比特每秒，输入 20000 至 10000000 之间的数值 • **percent**——根据百分比为某个类别分配最小带宽。如果其他队列没有用到全部端口带宽的话，这条队列可以超额订阅带宽。总和不能超过 100%，如果小于 100%的话，剩余带宽会被平均分配到所有带宽队列中 • **remaining**——为某个类别分配最小带宽。如果其他队列没有用到全部端口带宽的话，这条队列可以超额订阅带宽。总和不能超过 100%。如果策略中的某条队列上使用了

		priority 命令，那么用户最好也是用这条命令。用户也可以为每条队列分配速率而不是百分比；这样每条队列就会获得特定的加权，加权值与这些比率相关联。比率的取值范围是 0 至 100。在这个示例中，为策略分配的总带宽比率可以超过 100。 **注释：** 用户不能在一个 policy-map 中混用带宽类型
步骤 5	**queue-buffers** {**ratio** *ratio value*} **示例：** Device(config-pmap-c)# **queue-buffers ratio 10** Device(config-pmap-c)#	为队列配置适当的缓存大小。 **注释：** 一个策略中配置的缓存总大小必须小于或等于 100%。未分配的缓存会平均分布给所有剩余队列

步骤 6	**end**
	
	示例：
	`Device(config-pmap-c)# end`
	`Device#`
		返回特权 EXEC 模式

步骤 7	**show policy-map** **示例：** `Device# show policy-map`	（可选）显示为所有服务策略配置的所有类别策略配置信息

接下来做什么？

为用户网络中的 QoS 策略配置其他 policy-map。在创建了 policy-map 后，使用 **service-policy** 命令把一个或多个流量策略关联到接口上。

配置队列限制（CLI）

用户可以使用队列限制来配置加权尾部丢弃（WTD）。WTD 能够为每条队列配置多个门限值。每个服务类别都根据不同的门限值来丢弃数据包，以此提供 QoS 差分服务。在交换机上，每条队列有 3 个用户可以配置的门限值类别——0、1、2。因此每条队列中每个数据包的队列/丢弃决策是由数据包的门限值类别分配结果决定的，这个分配结果是由数据帧头部的 DSCP、CoS 或 QoS 组字段决定的。

WTD 还使用软限制，因此用户能够把队列限制配置为 400%（最大值为普通池中保留缓存的 4 倍）。这个软限制能够防止超越普通池，并且不会影响其他特性。

注释：用户只能在有线端口的出向队列上配置队列限制。

在开始前

执行这个配置步骤有以下先决条件：

- 在开始这部分介绍的配置步骤前，用户应该已经为队列限制创建了 class-map；
- 在配置队列限制前，用户必须已经在 policy-map 中配置了带宽、整形或优先级。

总步骤

1. configure terminal

2. policy-map *policy name*

3. class *class name*

4. bandwidth {*Kb/s* | **percent** *percentage* | **remaining** { **ratio** *ratio value* }}

5. queue-limit {*packets* **packets** | **cos** {*cos value* { *maximum threshold value* | **percent** *percentage* } | **values** {*cos value* | **percent** *percentage* } } | **dscp** {*dscp value* {*maximum threshold value* | **percent** *percentage*} | *match packet* {*maximum threshold value* | **percent** *percentage*} | **default** {*maximum threshold value* | **percent** *percentage*} | **ef** {*maximum threshold value* | **percent** *percentage*} | **dscp values** *dscp value*} | **percent** *percentage* }}

6. end

7. show policy-map

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**policy-map** *policy name*
	
	示例：
	`Device(config)# policy-map policy_queuelimit01`
	`Device(config-pmap)#`
		进入 policy-map 配置模式。
		创建或修改一个 policy-map，用户可以把它关联到一个或多个接口，用来指定服务策略

步骤 3	**class** *class name*
	
	示例：
	`Device(config-pmap)# class class_queuelimit01`
	`Device(config-cmap-c)#`
		进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。策略 class-map 配置模式中的命令选项如下所示：
		• *word*——class-map 名称
		• **class-default**——系统默认类别，匹配所有未分类的数据包

步骤 4	**bandwidth** {*Kb/s*	**percent** *percentage*	**remaining** { **ratio** *ratio value* }} **示例：** Device(config-pmap-c)# **bandwidth 500000** Device(config-pmap-c)#	为这个 policy-map 配置带宽。用户可以使用的命令参数如下所示： • *Kb/s*——千比特每秒，输入 20000 至 10000000 之间的数值 • **percent**——根据百分比为某个类别分配最小带宽。如果其他队列没有用到全部端口带宽的话，这条队列可以超额订阅带宽。总和不能超过 100%，如果小于 100%的话，剩余带宽会被平均分配到所有带宽队列中 • **remaining**——为某个类别分配最小带宽。如果其他队列没有用到全部端口带宽的话，这条队列可以超额订阅带宽。总和不能超过 100%。如果策略中的某条队列上使用了

		priority 命令，那么用户最好也是用这条命令。用户也可以为每条队列分配速率而不是百分比；这样每条队列就会获得特定的加权，加权值与这些比率相关联。比率的取值范围是 0 至 100。在这个示例中，为策略分配的总带宽比率可以超过 100。 **注释：** 用户不能在一个 policy-map 中混用带宽类型

步骤 5	**queue-limit** {*packets* **packets** | **cos** {*cos value* { *maximum threshold value* | **percent** *percentage* } | **values** {*cos value* | **percent** *percentage* } } | **dscp** {*dscp value* {*maximum threshold value* | **percent** *percentage*} | *match packet* {*maximum threshold value* | **percent** *percentage*} | **default** {*maximum threshold value* | **percent** *percentage*} | **ef** {*maximum threshold value* | **percent** *percentage*} | **dscp values** *dscp value*} | **percent** *percentage* }}
	
	示例：
	`Device(config-pmap-c)# queue-limit dscp 3 percent 20`
	`Device(config-pmap-c)# queue-limit dscp 4 percent 30`
	`Device(config-pmap-c)# queue-limit dscp 5 percent 40`
		设置队列限制门限值的百分比值。
		在每条队列中都有三个门限值（0、1、2），每个门限值都有一个默认值。用户可以使用这条命令来修改默认值，或者队列限制门限值的其他设置。举例来说，如果 DSCP 值为 3、4 和 5 的数据包会被发送到一个配置中的指定队列，用户就可以使用这条命令来社为这 3 个 DSCP 值设置门限值百分比。有关队列限制门限值的更多信息，用户可以参考加权尾部丢弃.
		注释： 交换机不支持使用绝对的队列限制百分比，只支持 DSCP 或 CoS 队列限制百分比

步骤 6	**end**
	
	示例：
	`Device(config-pmap-c)# end`
	`Device#`
		返回特权 EXEC 模式

步骤 7	**show policy-map** **示例：** `Device# show policy-map`	（可选）显示为所有服务策略配置的所有类别策略配置信息

接下来做什么？

为用户网络中的 QoS 策略配置其他 policy-map。在创建了 policy-map 后，使用 **service-policy** 命令把一个或多个流量策略关联到接口上。

配置整形特性（CLI）

用户可以使用 **shape** 命令来为指定类别配置整形（最大带宽）特性。队列的带宽会被限制为

用户配置的值，即使端口还有更多带宽可用。用户可以用平均百分比来配置整形特性，也可以用单位为比特每秒的平均值来配置整形特性。

在开始前

在开始这部分介绍的配置步骤前，用户应该已经为整形特性创建了 class-map。

总步骤

1. configure terminal

2. policy-map *policy name*

3. class *class name*

4. shape average {*target bit rate* | **percent** *percentage*}

5. end

6. show policy-map

具体步骤

步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 2	**policy-map** *policy name*
	
	示例：
	`Device(config)# policy-map policy_shaping01`
	`Device(config-pmap)#`
		进入 policy-map 配置模式。
		创建或修改一个 policy-map，用户可以把它关联到一个或多个接口，用来指定服务策略

步骤 3	**class** *class name*
	
	示例：
	`Device(config-pmap)# class class_shaping01`
	`Device(config-cmap-c)#`
		进入策略 class-map 配置模式。指定用户想要创建或更改的类别的名称。策略 class-map 配置模式中的命令选项如下所示：
		• *word*——class-map 名称
		• **class-default**——系统默认类别，匹配所有未分类的数据包

步骤 4	**shape average** {*target bit rate*	**percent** *percentage*}
		
		示例：
		`Device(config-pmap-c)# shape average percent 50`
		`Device(config-pmap-c)#`
			配置平均整形速率。用户可以用目标比特速率（比特每秒）来配置平均整形速率，或者使用接口带宽承诺信息速率（CIR）百分比来配置平均整形速率。
			注释： 对于出向 class-default SSID 策略来说，用户必须在配置了平均整形速率后，把队列缓存比率配置为 0

步骤 5	**end**
	
	示例：
	`Device(config-pmap-c)# end`
	`Device#`
		返回特权 EXEC 模式

步骤 6	**show policy-map** **示例：**	（可选）显示为所有服务策略配置的所有类别策略配置信息

	Device# **show policy-map**	

接下来做什么？

为用户网络中的 QoS 策略配置其他 policy-map。在创建了 policy-map 后，使用 **service-policy** 命令把一个或多个流量策略关联到接口上。

监控 QoS

用户可以使用以下命令来监控交换机上的 QoS。

表 99：监控 QoS

命令	描述
show class-map [*class_map_name*]	显示用户配置的所有 class-map 列表
show class-map type control subscriber {**all** | *name* } **show class-map type control subscriber detail**	显示控制 class-map 及其状态统计信息。 • **all——**显示所有 class-map 的信息 • *name*——显示配置的 class-map
show policy-map [*policy_map_name*]	显示用户配置的所有 policy-map 列表。用户可以使用的命令参数如下所示： • **poicy-map 名称** • **接口** • **会话**
show policy-map interface { **Auto-template** | **Capwap** | **GigabitEthernet** |**GroupVI** | **InternalInterface** |**Loopback** |**Lspvif** | | **Null** | **Port-channel** | **TenGigabitEthernet** | **Tunnel** | **Vlan** | **brief** | **class** | **input** | **output** }	显示交换机上配置的所有策略运行时的情况和状态统计信息。用户可以使用的命令参数如下所示： • **Auto Template**——auto-template 接口 • **Capwap**——CAPWAP 隧道接口 • **GigabitEhternet**——千兆以太网 IEEE 802.3z • **GroupVI**——组虚拟接口 • **Internal Interface**——内部接口 • **Loopback**——环回接口 • **Lspvif**——LSP 虚拟接口 • **Null**——空接口 • **Port-Channel**——EtherChannel 接口 • **TenGigabitEthernet**——万兆以太网 • **Tunnel**——隧道接口 • **Vlan**——Inspur VLAN • **brief**——policy-map 的简要描述信息 • **class**——每个类别的状态统计信息 • **input**——输入策略 • **output**——输出策略
show policy-map session [**input** | **output** | **uid** *UUID*]	描述隧道 QoS 策略。用户可以使用的命令参数如下所示： • **input**——输入策略

	• **output**——输出策略 • **uid**——基于 SSS 唯一识别符的策略
show policy-map type control subscriber { **all** | **name** }	显示 QoS policy-map 类型
show table-map	显示所有 table-map 及其配置
show ap name *ap_name* **service-policy**	显示 AP 上配置的所有策略

QoS 的配置示例

示例：使用访问控制列表实现分类

这个示例展示了如何使用访问控制列表（ACL），为 QoS 实现数据包分类：

```
Device# configure terminal
Device(config)# access-list 101 permit ip host 12.4.1.1 host 15.2.1.1
Device(config)# class-map acl-101
Device(config-cmap)# description match on access-list 101
Device(config-cmap)# match access-group 101
Device(config-cmap)#
```

在使用 ACL 创建了一个 class-map 后，用户需要为这个类别创建一个 policy-map，并把这个 policy-map 应用到接口上。

示例：服务类别（CoS）二层分类

这个示例展示了如何使用服务类别（CoS）二层分类特性，为 QoS 实现数据包分类：

```
Device# configure terminal
Device(config)# class-map cos
Device(config-cmap)# match cos ?
<0-7> Enter up to 4 class-of-service values separated by white-spaces
Device(config-cmap)# match cos 3 4 5
Device(config-cmap)#
```

在使用 CoS 二层分类特性创建了一个 class-map 后，用户需要为这个类别创建一个 policy-map，并把这个 policy-map 应用到接口上。

示例：服务类别（CoS）DSCP 分类

这个示例展示了如何使用服务类别（CoS）DSCP 分类特性，为 QoS 实现数据包分类：

```
Device# configure terminal
Device(config)# class-map dscp
```

```
Device(config-cmap)# match dscp af21 af22 af23
Device(config-cmap)#
```

在使用 DSCP 分类特性创建了一个 class-map 后，用户需要为这个类别创建一个 policy-map，并把这个 policy-map 应用到接口上。

示例：VLAN ID 二层分类

这个示例展示了如何使用 VLAN ID 二层分类特性，为 QoS 实现数据包分类：

```
Device# configure terminal
Device(config)# class-map vlan-120
Device(config-cmap)# match vlan ?
<1-4095> VLAN id
Device(config-cmap)# match vlan 120
Device(config-cmap)#
```

在使用 VLAN 二层分类特性创建了一个 class-map 后，用户需要为这个类别创建一个 policy-map，并把这个 policy-map 应用到接口上。

示例：使用 DSCP 值或优先级值进行分类

这个示例展示了如何使用 DSCP 值或优先级值来实现数据包分类：

```
Device# configure terminal
Device(config)# class-map prec2
Device(config-cmap)# description matching precedence 2 packets
Device(config-cmap)# match ip precedence 2
Device(config-cmap)# exit
Device(config)# class-map ef
Device(config-cmap)# description EF traffic
Device(config-cmap)# match ip dscp ef
Device(config-cmap)#
```

在使用 DSCP 值或优先级值创建了一个 class-map 后，用户需要为这个类别创建一个 policy-map，并把这个 policy-map 应用到接口上。

示例：层级式分类

这个示例展示了层级式分类，用户创建了名为 parent 的类别，它匹配另一个名为 child 的类别。名为 child 的类别基于 IP 优先级 2 进行匹配。

```
Device# configure terminal
Device(config)# class-map child
Device(config-cmap)# match ip precedence 2
Device(config-cmap)# exit
Device(config)# class-map parent
Device(config-cmap)# match class child
```

```
Device(config-cmap)#
```

在使用创建了 class-map parent 后，用户需要为这个类别创建一个 policy-map，并把这个 policy-map 应用到接口上。

示例：层级式策略的配置

这个示例展示了使用层级式策略来配置 QoS 的示例：

```
Device# configure terminal
Device(config)# class-map c1
Device(config-cmap)# match dscp 30
Device(config-cmap)# exit
Device(config)# class-map c2
Device(config-cmap)# match precedence 4
Device(config-cmap)# exit
Device(config)# class-map c3
Device(config-cmap)# exit
Device(config)# policy-map child
Device(config-pmap)# class c1
Device(config-pmap-c)# priority level 1
Device(config-pmap-c)#  police  rate  percent  20  conform-action
transmit exceed action drop
Device(config-pmap-c-police)# exit
Device(config-pmap-c)# exit
Device(config-pmap)# class c2
Device(config-pmap-c)# bandwidth 20000
Device(config-pmap-c)# exit
Device(config-pmap)# class class-default
Device(config-pmap-c)# bandwidth 20000
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# policy-map parent
Device(config-pmap)# class class-default
Device(config-pmap-c)# shape average 1000000
Device(config-pmap-c)# service-policy child
Device(config-pmap-c)# end
```

这个示例展示了使用 table-map 来配置层级式策略：

```
Device(config)# table-map dscp2dscp
Device(config-tablemap)# default copy
Device(config)# table-map dscp2up
Device(config-tablemap)# map from 46 to 6
Device(config-tablemap)# map from 34 to 5
Device(config-tablemap)# default copy
Device(config)# policy-map ssid_child_policy
Device(config-pmap)# class voice
```

```
Device(config-pmap-c)# priority level 1
Device(config-pmap-c)# police 15000000
Device(config-pmap)# class video
Device(config-pmap-c)# priority level 2
Device(config-pmap-c)# police 10000000
Device(config)# policy-map ssid_policy
Device(config-pmap)# class class-default
Device(config-pmap-c)# shape average 30000000
Device(config-pmap-c)# queue-buffer ratio 0
Device(config-pmap-c)# set dscp dscp table dscp2dscp
Device(config-pmap-c)# service-policy ssid_child_policy
```

示例：为语音和视频进行分类

这个示例描述了如何使用设备指定信息，为语音和视频流中的数据包进行分类。
在这个示例中，语音和视频流量从端点 A 进入设备的 GigabitEthernet1/0/1 接口，分别携带优先级值 5 和 6。除此之外，语音和视频还从端点 B 进入设备的 GigabitEthernet1/0/2 接口，分别携带 DSCP 值 EF 和 AF11。
假设从这两个接口收到的所有数据包都要发往上行链路接口，那么要求用户为语音流量实施 100 Mbit/s 限速，为视频流量实施 150 Mbit/s 限速。
为了按照上述需求进行分类，用户创建了一个类别来匹配从 GigabitEthernet1/0/1 入站的语音数据包，命名为 voice-interface-1，匹配优先级值 5。类似的，创建另一个类别来匹配从 GigabitEthernet1/0/2 入站的语音数据包，命名为 voice-interface-2。这两个类别分别关联到两个不同的策略中，名为 input-interface-1 的策略关联到 GigabitEthernet1/0/1，名为 input-interface-2 的策略关联到 GigabitEthernet1/0/2。用户把这个类别的行为定义为标记 QoS 组值为 10。为了在出站接口匹配 QoS 组值为 10 的数据包，用户创建了名为 voice 的类别，并在其中匹配 QoS 组值 10。这个类别关联到另一个名为 output-interface 的策略中，这个策略关联到上行链路接口上。视频也是使用类似的方法进行处理的，只不过使用 QoS 组值 20。
这个示例展示了使用上述设备指定信息来进行数据包分类：

```
Device(config)#
Device(config)# class-map voice-interface-
1 Device(config-cmap)# match ip precedence
5 Device(config-cmap)# exit
Device(config)# class-map video-interface-
1 Device(config-cmap)# match ip precedence
6 Device(config-cmap)# exit
Device(config)# class-map voice-interface-
2 Device(config-cmap)# match ip dscp ef
Device(config-cmap)# exit
Device(config)# class-map video-interface-
2 Device(config-cmap)# match ip dscp af11
Device(config-cmap)# exit
Device(config)# policy-map input-interface-
1 Device(config-pmap)# class voiceinterface-
```

```
1 Device(config-pmap-c)# set qosgroup
10 Device(config-pmap-c)# exit
Device(config-pmap)# class video-interface-1
Device(config-pmap-c)# set qos-group 20
Device(config-pmap-c)# policy-map input-interface-2
Device(config-pmap)# class voice-interface-2
Device(config-pmap-c)# set qos-group 10
Device(config-pmap-c)# class video-interface-2
Device(config-pmap-c)# set qos-group 20
Device(config-pmap-c)# exit
Device(config-pmap)# exit
Device(config)# class-map voice
Device(config-cmap)# match qos-group 10
Device(config-cmap)# exit
Device(config)# class-map video
Device(config-cmap)# match qos-group 20
Device(config)# policy-map output-interface
Device(config-pmap)# class voice
Device(config-pmap-c)#  police  256000  conform-action  transmit
exceed-action drop
Device(config-pmap-c-police)# exit
Device(config-pmap-c)# exit
Device(config-pmap)# class video
Device(config-pmap-c)#  police  1024000  conform-action  transmit
exceed-action drop
Device(config-pmap-c-police)# exit
Device(config-pmap-c)# exit
```

示例：配置下游 SSID 策略

要想配置下游 BSSID 策略，用户必须首先使用优先级等级队列来配置端口子系策略。

策略类型	示例
用户定义的端口子系策略	policy-map port_child_policy class voice priority level 1 20000 class video priority level 2 10000 class non-client-nrt-class bandwidth remaining ratio 10 class class-default bandwidth remaining ratio 15
出向 BSSID 策略	policy-map bssid-policer queue-buffer ratio 0

	class class-default shape average 30000000 set dscp dscp table dscp2dscp set wlan user-priority dscp table dscp2up service-policy ssid_child_qos
SSID 子系 QoS 策略	Policy Map ssid-child_qos Class voice priority level 1 police cir 5m admit cac wmm-tspec UP 6,7 / tells WCM allow 'voice' TSPEC\SIP snoop for this ssid rate 4000 / must be police rate value is in kbps) Class video priority level 2 police cir 60000

示例：入向 SSID 策略

这个示例展示了入向 SSID 层级式策略：

入向 SSID 策略类型	**示例**
入向 SSID 层级式策略	policy-map ssid-child-policy class voice //match dscp 46 police 3m class video //match dscp 34 police 4m policy-map ssid-in-policy class class-default set dscp wlan user-priority table up2dscp service-policy ssid-child-policy
	policy-map ssid_in_policy class dscp-40 set cos 1 police 10m class up-1 set dscp 34 police 12m class dscp-10 set dscp 20

	police 15m class class-default set dscp wlan user-priority table up2dscp police 50m

示例：客户端策略

客户端策略类型	示例/详情
默认出向子系策略	任何入站流量都包含用户优先级 0。 **注释：** 只有在启用了 ACM 的 WMM 客户端上，才默认启用客户端策略。 用户可以使用命令 **show ap dot11 5ghz network** 来确认 ACM 是否已启用。要想启用 ACM，用户可以使用命令 **ap dot11 5ghz cac voice acm**。 Policy-map client-def-down class class-default ~~set wlan user-priority 0~~
基于 AAA 和 TCLAS 的客户端策略	Policy Map client2-down[AAA+ TCLAS pol example] Class voice\\match dscp police <> set <> Class class-default set <> Class voice1|| voice2 [match acls] police <> class voice1 set <> class voice2 set <>
出方向上语音和视频流量的客户端策略	Policy Map client3-down class voice \\match dscp, cos police X class video police Y class class-default police Z
使用限速的，入方向上语音和视频流量的客户端策略	Policy Map client1-up class voice \\match dscp, up, cos

	police X class video police Y class class-default police Z
基于 DSCP 的语音和视频客户端策略	Policy Map client2-up class voice \\match dscp, up, cos set dscp <> class video set dscp <> class class-default set dscp <>
使用标记和限速的客户端入向策略	policy-map client_in_policy class dscp-48 //match dscp 48 set cos 3 police 2m class up-4 //match wlan user-priority 4 set dscp 10 police 3m class acl //match acl set cos 2 police 5m class class-default set dscp 20 police 15m
层级式客户端入向策略	policy-map client-child-policy class voice //match dscp 46 set dscp 40 police 5m class video //match dscp 34 set dscp 30 police 7m policy-map client-in-policy class class-default police 15m service-policy client-child-policy

示例：平均速率整形特性的配置

这个示例展示了如何配置平均速率整形特性：

```
Device# configure terminal
```

```
Device(config)# class-map prec1
Device(config-cmap)# description matching precedence 1 packets
Device(config-cmap)# match ip precedence 1
Device(config-cmap)# end
Device# configure terminal
Device(config)# class-map prec2
Device(config-cmap)# description matching precedence 2 packets
Device(config-cmap)# match ip precedence 2
Device(config-cmap)# exit
Device(config)# policy-map shaper
Device(config-pmap)# class prec1
Device(config-pmap-c)# shape average 512000
Device(config-pmap-c)# exit
Device(config-pmap)# policy-map shaper
Device(config-pmap)# class prec2
Device(config-pmap-c)# shape average 512000
Device(config-pmap-c)# exit
Device(config-pmap)# class class-default
Device(config-pmap-c)# shape average 1024000
```

在配置了 class-map、policy-map 和整形平均特性后，用户需要把 policy-map 应用在接口上。

示例：队列限制的配置

这个示例展示了如何基于 DSCP 值和百分比，来配置队列限制策略：

```
Device# configure terminal
Device#(config)# policy-map port-queue
Device#(config-pmap)# class dscp-1-2-3
Device#(config-pmap-c)# bandwidth percent 20
Device#(config-pmap-c)# queue-limit dscp 1 percent 80
Device#(config-pmap-c)# queue-limit dscp 2 percent 90
Device#(config-pmap-c)# queue-limit dscp 3 percent 100
Device#(config-pmap-c)# exit
Device#(config-pmap)# class dscp-4-5-6
Device#(config-pmap-c)# bandwidth percent 20
Device#(config-pmap-c)# queue-limit dscp 4 percent 20
Device#(config-pmap-c)# queue-limit dscp 5 percent 30
Device#(config-pmap-c)# queue-limit dscp 6 percent 20
Device#(config-pmap-c)# exit
Device#(config-pmap)# class dscp-7-8-9
Device#(config-pmap-c)# bandwidth percent 20
Device#(config-pmap-c)# queue-limit dscp 7 percent 20
Device#(config-pmap-c)# queue-limit dscp 8 percent 30
Device#(config-pmap-c)# queue-limit dscp 9 percent 20
Device#(config-pmap-c)# exit
```

```
Device#(config-pmap)# class dscp-10-11-12
Device#(config-pmap-c)# bandwidth percent 20
Device#(config-pmap-c)# queue-limit dscp 10 percent 20
Device#(config-pmap-c)# queue-limit dscp 11 percent 30
Device#(config-pmap-c)# queue-limit dscp 12 percent 20
Device#(config-pmap-c)# exit
Device#(config-pmap)# class dscp-13-14-15
Device#(config-pmap-c)# bandwidth percent 10
Device#(config-pmap-c)# queue-limit dscp 13 percent 20
Device#(config-pmap-c)# queue-limit dscp 14 percent 30
Device#(config-pmap-c)# queue-limit dscp 15 percent 20
Device#(config-pmap-c)# end
Device#
```

在完成上述 policy-map 队列限制配置后，用户可以把这个 policy-map 应用到接口上。

示例：队列缓存的配置

这个示例展示了如何配置队列缓存策略并把它应用到接口上：

```
Device# configure terminal
Device(config)# policy-map policy1001
Device(config-pmap)# class class1001
Device(config-pmap-c)# bandwidth remaining ratio 10
Device(config-pmap-c)# queue-buffer ratio ?
<0-100> Queue-buffers ratio limit
Device(config-pmap-c)# queue-buffer ratio 20
Device(config-pmap-c)# end
Device# configure terminal
Device(config)# interface gigabitEthernet2/0/3
Device(config-if)# service-policy output policy1001
Device(config-if)# end
```

示例：限速行为的配置

下面这个示例展示了可以与限速器关联的各种限速行为。这些行为是通过对合格、超出和违反限速规定的数据包指定不同的配置完成的。用户可以对超出和违反流量分析描述规则的数据包灵活地丢弃、标记和传输，或传输。

举例来说，在一个普通部署环境中，企业客户的限速流量离开自己的网络，去往服务提供商网络，并根据不同的 DSCP 值标记为合格（Conforming）、超出（Exceeding）和违反（Violating）数据包。服务提供商会在拥塞时选择丢弃被标记为超出和违反 DSCP 值的数据包，但会在带宽可用时选择传输这些数据包。

注释： 设备可以对二层字段中的 CoS 字段进行标记，也可以对三层字段中的优先级和 DSCP 字段进行标记。

用户可以使用这样一个有用的特性：把多个行为与一个时间相关联。举例来说，用户可以为

所有合格数据包设置优先级比特和 CoS 字段。然后限速特性可以提供一个子模式来配置行为。

限速行为的配置示例如下所示：

```
Device# configure terminal
Device(config)# policy-map police
Device(config-pmap)# class class-default
Device(config-pmap-c)# police cir 1000000 pir 2000000
Device(config-pmap-c-police)# conform-action transmit
Device(config-pmap-c-police)# exceed-action set-dscp-transmit dscp table exceed-markdown-table
Device(config-pmap-c-police)#  violate-action  set-dscp-transmit dscp table
violate-markdown-table
Device(config-pmap-c-police)# end
```

在这个示例中，exceed-markdown-table 和 violate-markdown-table 都是 table-map。

基于线速器的降低优先级标记行为只支持使用 table-map。设备中每个标记字段只能使用一个降低优先级的标记 table-map。

示例：限速器 VLAN 配置

下面这个示例展示了一个 VLAN 限速器的配置。在配置最后，用户在接口上应用了 VLAN policy-map 来实施 QoS 行为。

```
Device# configure terminal
Device(config)# class-map vlan100
Device(config-cmap)# match vlan 100
Device(config-cmap)# exit
Device(config)# policy-map vlan100
Device(config-pmap)# policy-map class vlan100
Device(config-pmap-c)# police 100000 bc conform-action transmit exceed-action drop
Device(config-pmap-c-police)# end
Device# configure terminal
Device(config)# interface gigabitEthernet1/0/5
Device(config-if)# service-policy input vlan100
```

示例：限速单元

这个示例展示了能够为 QoS 提供支持的各种限速单元。限速单元是令牌桶工作的基础。

设备支持的限速单元如下所示：

- 以比特每秒为单位指定 CIR 和 PIR。以字节为单位指定突发参数。这是默认模式；当用户没有指定单元时就会使用这个单元。用户也可以使用百分比来配置 CIR 和 PIR，这时突发参数必须以毫秒为单位进行配置；
- 以数据包每秒为单位指定 CIR 和 PIR。这时突然参数必须也配置为数据包。

以下示例展示了以比特每秒为单位的限速器配置：

```
Device(config)# policy-map bps-policer
Device(config-pmap)# class class-default
Device(config-pmap-c) # police rate 256000 bps burst 1000 bytes
conform-action transmit exceed-action drop
```

以下示例展示了以数据包每秒为单位的限速器配置。在这个配置中，用户配置了双速三色限速器，评估单元是数据包。突发和最高突发也都是以数据包为单位指定的。

```
Device(config)# policy-map pps-policer
Device(config-pmap)# class class-default
Device(config-pmap-c)# police rate 5000 pps burst 100 packets
peak-rate 10000 pps peak-burst 200 packets conform-action transmit
exceed-action drop violate-action drop
```

示例：单速双色限速特性的配置

以下示例展示了如何配置单速双色限速器：

```
Device(config)# class-map match-any prec1
Device(config-cmap)# match ip precedence 1
Device(config-cmap)# exit
Device(config)# policy-map policer
Device(config-pmap)# class prec1
Device(config-pmap-c)# police cir 256000 conform-action transmit
exceed-action drop
Device(config-pmap-c-police)# exit
Device(config-pmap-c)#
```

示例：双速三色限速特性的配置

以下示例展示了如何配置双色三色限速器：

```
Device# configure terminal
Device(config)# policy-Map dual-rate-3color-policer
Device(config-pmap)# class class-default
Device(config-pmap-c)# police cir 64000 bc 2000 pir 128000 be 2000
Device(config-pmap-c-police)# conform-action transmit
Device(config-pmap-c-police)# exceed-action set-dscp-transmit dscp
table exceed-markdown-table
Device(config-pmap-c-police)#   violate-action   set-dscp-transmit
dscp table violate-markdown-table
Device(config-pmap-c-police)# exit
Device(config-pmap-c)#
```

在这个示例中，exceed-markdown-table 和 violate-markdown-table 都是 table-map。

注释： 基于线速器的降低优先级标记行为只支持使用 table-map。设备中每个标记字段只能使用一个降低优先级的标记 table-map。

示例：table-map 标记特性的配置

以下步骤和示例展示了如何在 QoS 配置中使用 table-map 标记特性：

1. 定义一个 table-map；
   使用命令 **table-map** 来定义一个 table-map，并指定数值的映射关系。这个表并不知道它会使用的策略或类别。table-map 中的默认命令指示出当数据包中携带的数值不匹配“From”字段时，就复制用户在“To”字段中配置的值。举例来说，用户创建了名为 table-map1 的 table-map。映射关系中定义了把从 0 到 1、从 2 到 3 的映射，同时把默认值设置为 4。

```
Device(config)# table-map table-map1
Device(config-tablemap)# map from 0 to 1
Device(config-tablemap)# map from 2 to 3
Device(config-tablemap)# default 4
Device(config-tablemap)# exit
```

2. 定义 policy-map 并在其中使用 table-map；
   在这个示例中，QoS 会根据表 table-map1 中指定的映射关系，把入站数据包中的 CoS 值影射为 DSCP 值。举例来说，如果入站数据包的 DSCP 值为 0，那么数据包中的 CoS 值就会被设置为 1。如果用户没有在这里指定 table-map 名称，那么默认行为就是把“From”字段（本例中是 DSCP）的值复制到“To”字段（本例中是 CoS）。但是用户要知道 CoS 是 3 比特字段，DSCP 是 6 比特字段，也就是说 CoS 是复制了 DSCP 中的前 3 个比特。

```
Device(config)# policy map policy1
Device(config-pmap)# class class-default
Device(config-pmap-c)# set cos dscp table table-map1
Device(config-pmap-c)# exit
```

3. 把策略关联到一个接口。

```
Device(config)# interface GigabitEthernet1/0/1
Device(config-if)# service-policy output policy1
Device(config-if)# exit
```

示例：保留 CoS 标记的 table-map 配置

以下示例展示了如何在 QoS 配置中，使用 table-map 来保留接口上的 CoS 标记。
用户在接口的入方向上启用了 cos-rust-policy 策略（配置在示例中），以此来保留从接口进入 CoS 标记。如果用户没有启用这个策略的话，那么默认只会信任 DSCP 值。如果一个纯二层数据包到达了接口，那么当入向端口上没有为 CoS 配置相应策略的话，它的 CoS 值会被重写为 0。

```
Device# configure terminal
Device(config)# table-map cos2cos
Device(config-tablemap)# default copy
Device(config-tablemap)# exit
Device(config)# policy map cos-trust-policy
Device(config-pmap)# class class-default
Device(config-pmap-c)# set cos cos table cos2cos
```

```
Device(config-pmap-c)# exit
Device(config)# interface GigabitEthernet1/0/2
Device(config-if)# service-policy input cos-trust-policy
Device(config-if)# exit
```

接下来做什么？

再次查看 Auto-QoS 文档，来确定用户是否可以在 QoS 配置中使用这些自动功能。

Auto-QoS 的其他参考资料

相关文档

相关主题	文档名称
本章中命令的完整语法和用法信息	*QoS Command Reference (Inspur 6650 Switches)* *Inspur INOS Quality of Service Solutions Command Reference*
呼叫准入控制（CAC）	*System Management Configuration Guide (Inspur 6650 Switches)* *System Management Command Reference (Inspur 6650 Switches)*
组播整形和限速	*IP Multicast Routing Configuration Guide (Inspur 6650 Switches)*
应用可见性和控制	*System Management Configuration Guide (Inspur 6650 Switches)* *System Management Command Reference (Inspur 6650 Switches)*
应用可见性和控制	*System Management Configuration Guide (Inspur 6650 Switches)* *System Management Command Reference (Inspur 6650 Switches)*

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。	http://www.icntnetworks.com

要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	

Auto-QoS 的特性历史与信息

版本	变更
Inspur INOS 12.2	引入该特性

安全

管理交换机堆栈

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导

航系统。

交换机堆栈的前提条件

交换机堆栈中的所有交换机都需要与活跃交换机运行相同的许可证等级。有关许可证等级的更多信息，参见*系统管理配置指南（Inspur 6650 交换机）*。
交换机堆栈中的所有交换机都需要运行兼容的软件版本。
要启用堆栈，必须在堆栈端口上安装 StackWise 适配器。有关交换机堆栈硬件注意事项的更多信息，参见 *Inspur 6650 交换机硬件安装指南*。

交换机堆栈的限制条件

交换机堆栈配置存在以下限制条件：
- 运行 LAN Base 许可证等级的交换机堆栈不支持三层特性。
- 一个交换机堆栈至多可以有九台兼容堆栈特性的交换机通过 StackWise-160 端口相连。
- 一个交换机堆栈中不能混用 Inspur 6850 与 Inspur 6650 交换机。
- 一个交换机堆栈中不能混用不同许可证等级。

关于交换机堆栈的信息

交换机堆栈概述

一个交换机堆栈至多可以有九台兼容堆栈特性的交换机通过 StackWise-160 端口相连。堆栈成员一起作为一个统一的系统工作。通过二层和三层协议看，整个交换机堆栈在网络中就是一个实体。
一个交换机堆栈中总是有一台活跃交换机以及一台备用交换机。如果活跃交换机不可用，备用交换机会取得活跃交换机的角色，并继续维持堆栈的运行。
活跃交换机控制着交换机堆栈的操作，是整个堆栈的管理点。在活跃交换机上，可以配置：
- 应用到全体堆栈成员的系统级别（全局）特性
- 每个堆栈成员的接口级别特性

活跃交换机会包含交换机堆栈已保存以及正在运行的配置文件。配置文件中包含交换机堆栈的系统级别设置以及每个堆栈成员的接口级别设置。每个堆栈成员都有一份当前文件的拷贝，以进行备份。

交换机堆栈支持的特性

活跃交换机上支持的系统级别特性在整个交换机堆栈上都受支持。
加密特性

如果活跃交换机上运行的是加密通用软件镜像（支持加密），那么交换机堆栈上就支持加密特性。

StackWise-160

堆栈成员使用 StackWise-160 技术作为统一的系统一同工作。通过二层和三层协议看，整个交换机堆栈在网络中就是一个实体。

注释： 运行 LAN Base 镜像的交换机不支持三层特性。

StackWise-160 的堆栈带宽可达 160 Gbps，通过状态化切换（stateful switchover，SSO）来提供堆栈内的弹性。堆栈的行为像是成员交换机选举出的一台活跃交换机管理着一个交换单元。活跃交换机会自动在堆栈内选举备用交换机。活跃交换机会创建并更新所有的交换、路由信息并时常与备用交换机同步这些信息。主备切换的过程中接入点仍能保持连接，除非接入点直接连接到了主交换机，这种情况下接入点会掉电并重启。一个工作中的堆栈可以接收新成员，也可以在不中断服务的情况下删除旧成员。

交换机堆栈成员

一台单独的设备是有一个堆栈成员，且该成员也作为活跃交换机工作的设备堆栈。可以把两台单独的设备连接起来，创建包含两个堆栈成员的设备堆栈，其中，一台设备是活跃交换机。可以把一台设备连接到一个现有的设备堆栈，增加堆栈成员的数量。

所有堆栈成员之间都会收发 hello 消息。

- 如果一个堆栈成员不回应，该成员会被从堆栈中移除。
- 如果备用设备不回应，会选举出新的备用设备。
- 如果活跃设备不回应，备用设备会成为活跃设备。

此外，活跃设备和备用设备之间会收发 keepalive 消息。

- 如果备用设备不回应，会选举出新的备用设备。
- 如果活跃设备不回应，备用设备会成为活跃设备。

更换交换机堆栈成员

如果使用相同型号的交换机替换了一个堆栈成员，假设新交换机（称为预备交换机）与被替换的交换机使用相同的成员编号，则新交换机会使用与被替换交换机完全相同的配置进行工作。

在成员更换期间，交换机堆栈的操作可以继续进行而不被打断，除非更换了活跃交换机，或者添加了已开机的单独交换机或交换机堆栈。

添加已开机的交换机（合并）会导致所有交换机重载并选取新的活跃交换机。新选举出的活跃交换机会保留其角色及配置。所有其他交换机会保留自己的堆栈成员编号，并使用新选举活跃交换机的堆栈配置。

移除已开机的堆栈成员会把交换机堆栈划分（分割）成两个或多个交换机堆栈，每个堆栈都使用相同的配置。这会导致：

- 网络中 IP 地址冲突。如果希望交换机堆栈保持独立，需更改新创建的交换机堆栈的 IP 地址。
- 堆栈中两个成员的MAC地址冲突。可以使用**stack-mac updateforce**命令来解决冲突。

如果新创建的交换机堆栈没有活跃交换机或者备用交换机，该堆栈会重载并选取出一个新的活跃交换机。

注释： 确保对添加到交换机堆栈或从堆栈移除的交换机进行关机。
在添加或移除堆栈成员之后，确保该交换机堆栈按照全带宽（160 Gbps）运行。按住堆栈成员的Mode键，直到Stack模式LED灯亮起。堆栈中所有交换机最右两个端口的LED灯应该是绿色的。根据交换机型号不同，最右两个端口可以是10吉比特以太网端口或者小型可插拔（small form-factor pluggable，SFP）模块端口（10/100/1000端口）。如果任意交换机上这两个LED灯有不是绿色的情况，则堆栈没有全带宽运行。
如果移除了已开机的成员，但是不想分割堆栈：

- 关闭新创建的交换机堆栈中的交换机。
- 把它们通过堆栈端口重连到原来的交换机堆栈。
- 交换机开机。

关于影响交换机堆栈的连线及供电注意事项，参见*Inspur 6650交换机硬件安装指南*。

堆栈成员编号

堆栈成员编号（1到9）标识了设备堆栈中的每个成员。成员编号也确定了堆栈成员使用的接口级别的配置。可以使用EXEC命令**show switch**显示堆栈成员编号。
一台新的开箱即用的设备（没有加入过设备堆栈或还没有手动指定堆栈成员编号）默认的堆栈成员编号是1。加入设备堆栈时，该设备的默认堆栈成员编号会更改为堆栈中的最低可用成员编号。
一个堆栈中的堆栈成员不能使用相同的堆栈成员编号。每个堆栈成员，包括单独的设备，都会保持使用自己的成员编号，直到手动进行更改，或者编号已经被堆栈中其他成员使用。

- 如果使用命令**switch** *current-stack-member-number***renumber** *new-stack-member-number*手动更改了堆栈成员编号，只有在堆栈成员重置（或者输入了特权EXEC命令**reload slot** *stack-member-number*）且该编号没有分配给堆栈中其他成员时，新的编号才会生效。
  另一种更改堆栈成员编号的方式是更改Device_NUMBER环境变量。
  如果该编号已经被堆栈中的其他成员使用，设备会选用堆栈中最低可用的编号。
  如果手动更改了堆栈成员编号，且新堆栈成员没有相关联的接口级别配置，该堆栈员会重置为默认配置。
  不能在规划设备上使用**switch** *current-stack-member-number* **renumber** *new-stack-member-number*命令。如果执行此命令，命令会被拒绝。
- 如果把一个堆栈成员移动到不同的设备堆栈中，只在设备成员编号没有被堆栈中其他成员使用时，该成员才会保留其编号。如果编号被使用，该设备会选用堆栈中的最低可用编号。
- 在合并设备堆栈时，加入设备堆栈的新活跃交换机会选择堆栈中的最低可用编号。

如硬件安装指南中所述，可以使用 Stack 模式中的设备端口 LED 来可视化地确定每个堆栈成员的编号。
在 **default** 模式中，只有堆栈 master 的 Stack LED 才会闪烁绿灯。把 Mode 键切换至 **Stack** 选项时，所有堆栈成员的 Stack LED 都会亮绿灯。
当模式键被切换到 Stack 选项时，每个堆栈成员的交换机编号都会通过交换机前五个端口的 LED 显示。所有堆栈成员的交换机编号都以二进制形式显示。在交换机上，琥珀色的 LED 等表示 0，绿色的 LED 表示 1。
交换机编号 5（二进制为 00101）的示例如下：
交换机编号为 5 号的堆栈成员上前五个 LED 颜色组合如下：

- 端口 1：琥珀色
- 端口 2：琥珀色
- 端口 3：绿色
- 端口 4：琥珀色
- 端口 5：绿色

类似的，根据堆栈成员的交换机编号不同，前五个 LED 灯会亮琥珀色灯或亮绿灯。

注释：

- 如果把水平堆栈端口连接到另一端的正常网络端口，如果在 30 秒内没有从另一端接收到 SDP 包，堆栈端口的传输和接收会被禁用。
- 堆栈端口不会关闭，但传输和接收会被禁用。控制台上会显示以下日志消息。当对端网络端口转换为堆栈端口时，该堆栈端口的传输和接收才会被启用。

```
%STACKMGR-4-HSTACK_LINK_CONFIG: Verify peer stack port setting
for hstack StackPort-1 switch 5 (hostname-switchnumber)
```

堆栈成员优先级

堆栈成员使用更高的优先级会增加其被选举为活跃交换机的概率，并有助于保留自己的堆栈成员编号。优先级值可以是 1 到 15 之间。默认的优先级值是 1。可以使用 EXEC 命令 **show switch** 显示堆栈成员优先级值。

注释： 建议把最高优先级值分配给期望成为活跃交换机的设备。这保证在重新选举发生时，设备会被选举为活跃交换机。

要更改堆栈成员的优先级值，使用**switch** *stack-member-number* **priority** *newpriority-value*命令。更多信息参见“设置堆栈成员优先级”一节。

新的优先级会立即生效，但不会影响当前的活跃交换机。在当前的活跃交换机或者交换机堆栈重置的时候，新的优先级值有助于决定哪个堆栈成员会被选举为新的活跃交换机。

交换机堆栈网桥 ID 以及 MAC 地址

交换机堆栈在网络中通过*网桥 ID（bridge ID）*进行标识，如果作为三层设备运行，则使用路由器 MAC 地址标识。网桥 ID 以及路由器 MAC 地址由活跃交换机的 MAC 地址决定。

如果活跃交换机变化，新活跃交换机的 MAC 地址会决定新的网桥 ID 以及路由器 MAC 地址。

如果整个交换机堆栈重载，交换机堆栈会使用活跃交换机的 MAC 地址。

交换机堆栈的持续 MAC 地址

可以使用持续 MAC 地址特性设置堆栈 MAC 地址改变的时延。在此时间之内，如果之前的活跃交换机重新加入了堆栈，即使该交换机现在是堆栈成员而不是活跃交换机，交换机堆栈也会继续使用其 MAC 地址作为堆栈的 MAC 地址。如果之前的活跃交换机在此时间段内没有重新加入堆栈，交换机堆栈会采用新活跃交换机的 MAC 地址作为堆栈的 MAC 地址。默认情况下，堆栈的 MAC 地址会是首个活跃交换机的 MAC 地址，即使有新的活跃交换机接替其角色。

也可以配置堆栈 MAC 地址的持久性，使堆栈 MAC 地址永远不会更改为新活跃交换机的 MAC

地址。

活跃及备用交换机的选举和重新选举

所有堆栈成员都有资格成为活跃交换机或备用交换机。如果活跃交换机不可用，备用交换机会成为活跃交换机。

活跃交换机会保持其角色，除非以下事件发生：

- 交换机堆栈被重置。
- 活跃交换机被从交换机堆栈移除。
- 活跃交换机被重置或关机。
- 活跃交换机故障。
- 添加了已开机的单独交换机或交换机堆栈，交换机堆栈成员增加。

活跃交换机会按序根据以下因素进行选举或重新选举：

1 交换机当前是活跃交换机。

2 交换机有最高的堆栈成员优先级。

注释： 建议把最高优先级值分配给期望成为活跃交换机的设备。这保证在重新选举发生时，设备会被选举为活跃交换机。

3 交换机有最短的启动时间。

4 交换机有最低的 MAC 地址。

注释： 选举或重新选举新的备用交换机的因素与活跃交换机的相同，且适用于除了活跃交换机之外的所有参与交换机。

在选举之后，新的活跃交换机会在几秒之后可用。在此期间，交换机堆栈使用内存中的转发表来最小化网络中断。在新的活跃交换机选举或重置期间，其他可用的堆栈成员的物理接口不受影响。

当之前的活跃交换机变为可用状态时，它*不会*假定自己是活跃交换机的角色。

如果开启或重置了整个交换机堆栈，一些堆栈成员可能不会参与到活跃交换机的选举过程中。在相同的 2 分钟时间范围内开机的堆栈成员会参与活跃交换机的选举过程，并有机会成为活跃交换机。在 120 秒时间范围之后启用的堆栈成员不会参与初始选举过程，因而成为堆栈成员。关于影响活跃交换机选举的启动注意事项，参见交换机硬件安装指南。

如硬件安装指南中所述，可以通过交换机上的 ACTV LED 来查看交换机是否是活跃交换机。

交换机堆栈配置文件

活跃交换机上有交换机堆栈已保存的以及正在运行的配置文件。备用交换机会自动接收同步过来的运行配置文件。当运行配置文件被保存在启动配置文件中时，堆栈成员会同步地进行拷贝。如果活跃交换机变为不可用状态，备用交换机会接替其角色，并使用当前的运行配置。

配置文件记录这些配置：

- 系统级别（全局）配置，如 IP、STP、VLAN 以及、SNMP 等应用于所有堆栈成员的设置。
- 针对每个堆栈成员的堆栈成员接口相关配置。

注释： 如果活跃交换机被替换，且没有把运行配置保存在启动配置中，活跃交换机的接口相关配置会被保存。

一台新的开箱即用的设备在加入交换机堆栈时会使用该堆栈的系统级别设置。如果一台设备在启动之前被移动到不同的交换机堆栈中，该设备会丢失已保存的配置文件，并使用新交换

机堆栈的系统级别配置。如果设备在加入新交换机堆栈之前被启动作为一台单独的设备使用，交换机堆栈会重载。堆栈重载时，新设备可能成为活跃交换机，保留其配置并覆盖其他堆栈成员的配置文件。

每个堆栈成员的接口相关配置都与堆栈成员编号相关联。堆栈成员会保持使用其编号，除非手动更改或该编号已被相同交换机堆栈中的其他成员使用。如果交换机成员编号改变，新的编号会在堆栈成员重置以后生效。

- 如果不存在该成员编号的接口相关配置，该堆栈成员会使用自己默认的接口相关配置。
- 如果存在该成员编号的接口相关配置，该堆栈成员会使用与该成员编号关联的接口相关配置。

如果使用相同型号的设备替换了一个故障的成员，替换设备会自动使用与故障设备相同的接口相关配置，管理员无需重新配置接口设置。替换设备（称为规划设备）必须与故障设备使用相同的堆栈成员编号。

可以按照备份及恢复单独设备配置的方式来进行堆栈配置的备份和恢复。

通过离线配置规划堆栈成员

可以使用离线配置特性在新交换机加入堆栈之前对其进行*规划*（提供配置）。可以配置堆栈成员编号、交换机类型以及与当前不是堆栈一部分的交换机相关的接口。在交换机堆栈上创建的配置被称为*规划配置*。被添加到交换机堆栈且接收此配置的交换机被称为*规划交换机*。管理员可以使用全局配置命令 **switch** *stack-member-number* **provision** *type* 手动创建规划配置。在把规划交换机添加到堆栈之前，必须更改 *stack-member-number*，且必须与为堆栈中新交换机创建的堆栈成员编号相同。规划配置中的交换机类型必须与新添加交换机的类型相同。当一台交换机被添加到交换机堆栈，且不存在规划配置时，规划配置会被自动创建。

在配置与规划交换机相关的接口时，交换机堆栈会接受配置，且此信息会出现在运行配置中。然而，因为该交换机不是活跃状态，任何与其接口相关的配置都不会运行，且与规划交换机相关的接口不会出现在特定特性的显示信息中。比如，与规划交换机相关的 VLAN 配置信息不会出现在交换机堆栈的 **show vlan** 用户 EXEC 命令输出中。

无论规划交换机是否是堆栈的一部分，交换机堆栈都会把规划配置保留在运行配置中。可以输入 **copy running-config startup-config** 特权 EXEC 命令把规划配置保存到启动配置文件中。启动配置文件确保交换机堆栈可以重载，且无论规划交换机是不是交换机堆栈的一部分，都能使用已保存的信息。

把规划交换机添加到交换机堆栈的影响

把规划设备添加到交换机堆栈时，堆栈会应用规划配置或默认配置。下表列出了交换机堆栈在对比规划配置以及规划交换机时会发生的事件。

表 169：比较规划配置以及规划交换机的结果

场景		结果
堆栈成员编号和设备类型匹配。	1 如果规划交换机的堆栈成员编号与堆栈中规划配置的堆栈成员编号相同，且 2 如果规划交换机的设备	交换机堆栈把规划配置应用到规划交换机上，并将其加入堆栈。

	类型与堆栈中规划配置的设备类型相同。	
堆栈成员编号匹配但设备类型不匹配。	1 如果规划交换机的堆栈成员编号与堆栈中规划配置的堆栈成员编号相同，但是 2 规划交换机的设备类型与堆栈中规划配置的设备类型不同。	交换机堆栈会把默认配置应用到规划交换机上，并将其加入堆栈。 规划配置会被改变以反映新信息。
规划配置中未找到堆栈成员编号。		交换机堆栈会把默认配置应用到规划交换机上，并将其加入堆栈。 规划配置会被改变以反映新信息。
规划配置中未找到规划交换机的堆栈成员编号。		交换机堆栈会把默认配置应用到规划交换机上，并将其加入堆栈。

如果向已关机的堆栈中添加了一台和规划配置中型号不同的规划交换机，启动堆栈后，交换机堆栈会拒绝启动配置中（不正确的）的**switch***stack-member-number* **provision** *type*命令。然而，在堆栈初始化期间，启动配置文件中规划接口（可能有类型错误）的非默认接口配置信息会被执行。根据实际设备类型以及之前规划的交换机类型的差别不同，一些命令可能被拒绝，而一些命令会被接受。

注释： 如果交换机堆栈不含有新设备的规划配置，设备会使用默认接口配置加入堆栈。交换机堆栈随后会添加与新设备匹配的全局配置命令**switch** *stack-member-number* **provision** *type*到运行配置中。更多配置信息，参见*规划交换机堆栈的新成员*一节。

替换交换机堆栈中规划交换机的影响

交换机堆栈中的规划交换机故障时，可以把它从堆栈中移除并使用另一台设备代替，堆栈对其应用规划配置或默认配置。交换机对比规划配置以及规划交换机时发生的事件与添加规划交换机到堆栈时的事件相同。

移除交换机堆栈中规划交换机的影响

如果从设备堆栈中移除了一台规划交换机，与移除的堆栈成员相关的配置会作为规划信息保留在运行配置中。要完全移除配置，使用**no switch** *stack-member-number* **provision**全局配置命令。

升级运行不兼容软件的交换机

自动升级以及自动建议特性让使用与交换机堆栈不兼容软件包的交换机可以升级到兼容的

软件版本，以便加入交换机堆栈。

自动升级

自动升级特性的目的是让交换机升级到兼容的软件镜像，使交换机能加入交换机堆栈。
当一台新交换机尝试加入一个交换机堆栈时，每个堆栈成员都会进行与新交换机的兼容性检查。每个堆栈成员会把兼容性检查的结果发送给活跃交换机，由它来使用这些结果确定交换机是否能够加入交换机堆栈。如果新交换机上的软件与交换机堆栈不兼容，新交换机会进入版本不匹配（version-mismatch，VM）模式。
如果在现有交换机堆栈上启用了自动升级特性，活跃交换机会自动升级新交换机，让它使用与兼容的堆栈成员相同的软件镜像。检测到不匹配的软件几分钟之后，自动升级会开始。
自动升级默认被禁用。
自动升级包括自动拷贝过程以及自动提取过程。

- 自动拷贝过程会自动地把运行在任意堆栈成员上的软件镜像拷贝到新交换机上以自动升级。如果启用自动拷贝，新交换机有足够的闪存空间，而且交换机堆栈运行的软件镜像适用于新交换机时，自动拷贝会进行。

注释： VM模式中的交换机可能无法运行所有的软件版本。例如，新交换机的硬件在较早的软件版本中无法识别。

- 当自动升级过程不能在堆栈中找到合适的软件拷贝给新交换机时，自动提取过程会发生。此时，自动提取进程会搜索堆栈中的所有交换机，查找升级软件堆栈或新交换所需的bin文件。这个bin文件可以在交换机堆栈或新交换机的任意闪存文件系统中。如果在堆栈成员上找到了适用于新交换机的bin文件，此进程会提取文件并自动升级新交换机

自动升级特性在捆绑模式中不可用。交换机堆栈必须运行在安装模式中。如果交换机堆栈运行在捆绑模式中，使用特权EXEC命令**software expand**更改为安装模式。
可以在新交换机上使用全局配置命令**software auto-upgrade enable**启用自动升级特性。可以使用特权EXEC命令**show running-config**，通过查看输出中的*Auto upgrade*一行来检查自动升级状态。
可以使用全局配置命令**software auto-upgrade source url**，配置自动升级特性使用特定的软件包来升级新交换机。如果软件包不合法，新交换机会使用与兼容堆栈成员相同的软件镜像来进行升级。
自动升级过程完成时，新交换机会重启并作为全功能的成员加入堆栈。如果在重启期间两条堆栈线缆都连接上，则不会发生网络断开，因为交换机堆栈按照两路环模式工作。
关于升级运行不兼容软件交换机的更多信息，参见*Inspur INOS文件系统、配置文件及软件包文件附录（Inspur 6650交换机）*。

自动建议

以下情况发生时，自动建议特性会被触发：

- 自动升级特性被禁用。
- 新交换机在捆绑模式，而堆栈在安装模式。自动建议会显示syslog消息，说明使用特权EXEC命令**software auto-upgrade**把新交换机改为安装模式。

- 堆栈在捆绑模式。自动建议会显示syslog，说明在捆绑模式中重启新交换机，让其能够加入堆栈。
- 因为新交换运行不兼容的软件，自动升级尝试失败。在交换机堆栈对新交换机执行了兼容性检查之后，自动建议会显示syslog，说明新交换机是否可以自动升级。

自动建议不能被禁用。当交换机堆栈的软件和版本不匹配（VM）模式中的交换机软件不含有相同许可证等级时，自动建议特性*不会*给出建议。

自动建议消息示例

示例：自动升级被禁用且不兼容交换机尝试加入

以下自动建议示例展示了当自动升级特性被禁用，且不兼容的switch1尝试加入交换机堆栈时显示的系统消息：

```
*Oct 18 08:36:19.379: %INSTALLER-6-AUTO_ADVISE_SW_INITIATED: 2 installer: Auto advise initiated for switch 1
*Oct 18 08:36:19.380: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: Searching stack for softwareto upgrade switch 1
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: Switch 1 with incompatiblesoftware has been
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: added to the stack. Thesoftware running on
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: all stack members was scanned and it has been
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: determined that the 'softwareauto-upgrade'
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: command can be used to install compatible
*Oct 18 08:36:19.382: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: software on switch 1.
```

示例：自动升级被禁用且新交换机在捆绑模式中

以下自动建议示例展示了当自动升级特性被禁用，且运行捆绑模式的交换机尝试加入运行安装模式的堆栈时显示的系统消息：

```
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW_INITIATED: 2 installer: Auto advise initiated for switch 1
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: Switch 1 running bundledsoftware has been added
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: to the stack that is runninginstalled software.
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: The 'software auto-upgrade'command can be used to
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: convert switch 1 to theinstalled running mode by
*Oct 18 11:09:47.005: %INSTALLER-6-AUTO_ADVISE_SW: 2 installer: installing its running software.
```

交换机堆栈管理连通性

管理员可以通过活跃交换机管理交换机堆栈以及堆栈成员接口。可以使用CLI、SNMP以及

支持的网络管理应用，如InspurWorks。不能基于独立的设备管理堆栈成员。
注释： 可以使用SNMP管理支持的MIB中定义的堆栈网络特性。交换机不支持使用SNMP管理堆栈特定的特性，比如堆栈成员以及选举。

通过 IP 地址到交换机堆栈的连通性

交换机堆栈通过一个IP地址进行管理。这个IP地址是系统级别的设置，不针对活跃交换机或者任何其他的堆栈成员。只要有IP连通性，就算从堆栈中移除了活跃交换机或者任何其他的堆栈成员，管理员仍然可以通过相同的IP地址管理堆栈。
注释： 把堆栈成员从交换机堆栈中移除后，堆栈成员会保留堆栈IP地址。为了避免网络中两台设备使用相同IP地址造成的冲突，请更改从交换机堆栈中移除设备的IP地址。
有关交换机堆栈配置的信息，参见*交换机堆栈配置文件*一节。

通过控制台端口或以太网管理端口到交换机堆栈的连通性

可以使用以下方式之一连接到活跃交换机：
- 可以通过一个或多个堆栈成员的控制台端口，使用终端或PC连接活跃交换机。
- 可以通过一个或多个堆栈成员的以太网管理端口，使用PC连接活跃交换机。关于通过以太网管理端口连接交换机堆栈的更多信息，参见*使用以太网管理端口*一节。

可以通过一个或多个堆栈成员的控制台端口，使用终端或PC连接到堆栈master。
要当心使用多个CLI会话连接活跃交换机的情况。在一个会话中输入的命令不会在另一个会话中显示。因此，管理员可能无法分辨输入了命令的会话。
建议在管理交换机堆栈时只使用一个CLI会话。

如何配置交换机堆栈

启用持续 MAC 地址特性

注释： 输入命令配置此特性时，警告消息会显示此配置的后果。应该小心地使用此特性。在同一域中的其他位置使用老活跃交换机的 MAC 地址可能导致流量丢失。
按照以下步骤启用持续 MAC 地址特性：
总步骤
1. enable
2. configure terminal
3. stack-mac persistent timer [**0** | *time-value*]
4. end
5. copy running-config startup-config
具体步骤

	命令或操作	目的
步骤 1	**enable**	进入特权 EXEC 模式。在提示时输入密

	示例： `Device>`**enable**	码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**stack-mac persistent timer** [**0** | *time-value*] **示例：** `Device(config)#` **stack-mac persistent timer 7**	设置在活跃交换机变化时，堆栈 MAC 地址变为新活跃交换机地址的时延。如果在此期间之前的活跃交换机加入了堆栈，堆栈使用该地址作为堆栈 MAC 地址。 • 不输入值或值为 0 的命令，无限期地继续使用当前活跃交换机的 MAC 地址。 • 输入 1 到 60 分钟的 *time-value*，配置堆栈 MAC 地址变为新活跃交换机地址的时间。 堆栈会使用之前活跃交换机的 MAC 地址，直到配置的时间过期。
步骤 4	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**copy running-config startup-config** **示例：** `Device#` **copy running-configstartup-config**	（可选）把配置的条目保存到配置文件中。

接下来做什么？

使用全局配置命令 **no stack-mac persistent timer** 来禁用持续 MAC 地址特性。

分配堆栈成员编号

此选项仅可以在活跃交换机上执行。

按照以下步骤给堆栈成员分配成员编号：

总步骤

1. enable

2. configure terminal

3. switch *current-stack-member-number* **renumber** *new-stack-member-number*

4. end

5. reload slot *stack-member-number*

6. show switch

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：**	进入特权 EXEC 模式。在提示时输入密码。

	Device>**enable**	
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式。
步骤 3	**switch** *current-stack-member-number* **renumber***new-stack-member-number* **示例：** Device(config)# **switch 3 renumber 4**	指定当前的堆栈成员编号以及堆栈成员的新编号。范围从 1 到 9。 可以使用用户 EXEC 命令 **show switch** 显示当前的堆栈成员编号。
步骤 4	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式。
步骤 5	**reload slot** *stack-member-number* **示例：** Device# **reload slot 4**	重置堆栈成员。
步骤 6	**show switch** **示例：** Device#**show switch**	验证堆栈成员编号。
步骤 7	**copy running-config startup-config** **示例：** Device# **copy running-configstartup-config**	（可选）把配置的条目保存到配置文件中。

设置堆栈成员优先级

此选项仅可以在活跃交换机上执行。
按照以下步骤给堆栈成员分配优先级值。

总步骤

1. enable

2. switch *stack-member-number* **priority** *new-priority-number*

3. show switch *stack-member-number*

4. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** Device>**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**switch** *stack-member-number* **priority***new-priority-number* **示例：** Device# **switch 3 priority 2**	指定堆栈成员编号以及堆栈成员的新优先级。堆栈成员编号范围从 1 到 9。优先级值范围从 1 到 15。 可以使用用户 EXEC 命令 **show switch** 显示当前的优先级值。 新优先级值会立刻生效，但不会影响当前的活跃交换机。在当前活跃交换机或堆栈重置时，新优先级值有助于

		确定哪个堆栈成员会被选为新的活跃交换机。
步骤 3	**show switch** *stack-member-number* **示例：** `Device#` **show switch**	验证堆栈成员的优先级值。
步骤 4	**copy running-config startup-config** **示例：** `Device#` **copy running-configstartup-config**	（可选）把配置的条目保存到配置文件中。

规划交换机堆栈的新成员

此选项仅可以在活跃交换机上执行。

总步骤

1. show switch

2. configure terminal

3. switch *stack-member-number* **provision** *type*

4. end

5. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**switch** *stack-member-number* **provision** *type* **示例：** `Device(config)#` **switch 3 provision WS-xxxx**	指定预配置交换机的堆栈成员编号。默认情况下，无规划交换机。 *stack-member-number* 的范围从 1 到 9。指定交换机堆栈中未被使用的交换机成员编号，见步骤 1。 *type* 字段输入命令行帮助字符串中列出的支持交换机型号。
步骤 4	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**copy running-config startup-config** **示例：** `Device#` **copy running-configstartup-config**	（可选）把配置的条目保存到配置文件中。

移除规划交换机信息

在开始前，必须把规划交换机从堆栈中移除。此选项仅可以在活跃交换机上执行。

总步骤

1. configure terminal

2. no switch *stack-member-number* **provision**

3. end

4. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**no switch** *stack-member-number* **provision** **示例：** `Device(config)#` **no switch 3 provision**	移除指定成员的规划信息。
步骤 3	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 4	**copy running-config startup-config** **示例：** `Device#` **copy running-configstartup-config**	（可选）把配置的条目保存到配置文件中。

如果要移除规划交换机的堆栈有如下配置：

- 堆栈有四个成员
- 堆栈成员 1 是活跃交换机
- 堆栈成员 3 是规划交换机

且希望移除规划信息并避免收到错误消息，可以移除堆栈成员3的电源，断开堆栈成员3号连接的StackWise-160线缆，重连其余交换机之间的线缆，并输入**no switch***stack-member-number* **provision**全局配置命令。

显示堆栈中的不兼容交换机

总步骤

1. show switch

具体步骤

	命令或操作	目的
步骤 1	**show switch** **示例：** `Device#` **show switch**	显示交换机堆栈中的不兼容交换机（'Current State'为'V-Mismatch'）。V-Mismatch 状态标识出了软件不兼容的交换机。输出中显示的 Lic-Mismatch 表示与活跃交换机运行不同许可证级别的交换机。 有关管理许可证级别的更多信息，参见*系统管理配置指南(Inspur 6650 交换机)*。

升级交换机堆栈中的不兼容交换机

总步骤

1. software auto-upgrade

2. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**software auto-upgrade** **示例：** `Device#` **software auto-upgrade**	升级交换机堆栈中的不兼容交换机，或把捆绑模式中的交换机改为安装模式。
步骤 2	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置的条目保存到配置文件中。

交换机堆栈故障排除

临时禁用堆栈端口

如果堆栈端口抖动且造成堆栈环不稳定，要禁用端口，输入特权EXEC命令**switch***stack-member-number* **stack port** *port-number* **disable**。要重新启用端口，输入命令**switch** *stack-member-number* **stack port** *port-number* **enable**。

注释： 要小心使用**switch** *stack-member-number* **stack port** *port-number* **disable**命令。禁用堆栈端口时，堆栈会使用半速带宽工作。

当所有成员都通过堆栈端口连接，且都在就绪状态中时，堆栈为全环状态。

以下情况发生时，堆栈为部分环状态：

- 所有成员都通过堆栈端口连接，但是一些成员不是就绪状态。
- 一些成员未通过堆栈端口连接。

总步骤

1. switch *stack-member-number* **stack port** *port-number* **disable**

2. switch *stack-member-number* **stack port** *port-number* **enable**

具体步骤

	命令或操作	目的
步骤 1	**switch** *stack-member-number* **stack port** *port-number* **disable** **示例：** `Device#` **switch 2 stack port 1 disable**	禁用特定的堆栈端口。
步骤 2	**switch** *stack-member-number* **stack port** *port-number* **enable** **示例：** `Device#` **switch 2 stack port 1 enable**	重新启用堆栈端口。

要禁用堆栈端口且堆栈在全环状态时，只能禁用一个堆栈端口。会显示以下信息：

```
Enabling/disabling a stack port may cause undesired stack changes. Continue?[confirm]
```

要禁用堆栈端口且堆栈在部分环状态时，不能禁用端口。会显示以下信息：

```
Disabling stack port not allowed with current stack configuration.
```

在另一个成员启动时重新启用堆栈端口

交换机1上的堆栈端口1连接到了交换机4上的端口2。如果端口1抖动，可以使用特权EXEC命令**switch 1 stack port 1 disable**禁用端口1。当交换机1上的端口1被禁用且交换机1仍然开机时，按照以下步骤重新启用堆栈端口：

步骤1 断开交换机1上的堆栈端口1与交换机4上的端口2之间的连线。

步骤2 从堆栈中移除交换机4。

步骤3 添加一个交换机来代替交换机4并给它分配交换机编号4。

步骤4 重连交换机1上的堆栈端口1与交换机4（替换交换机）上的端口2之间的连线。

步骤5 重新启用交换机之间的链路。输入特权EXEC命令**switch 1 stack port 1 enable**启用交换机1上的端口1。

步骤6 启动交换机4。

注意： 在启用交换机1的端口1之前启动交换机4可能导致一台交换机重启。如果先启动交换机4，可能需要输入**switch 1 stack port 1 enable**以及**switch4 stack port 2 enable**特权EXEC命令来启用链路。

监控设备堆栈

表170：显示堆栈信息的命令

命令	描述
show switch	显示堆栈的汇总信息，包括规划交换机状态以及版本不匹配模式中的交换机。
show switch *stack-member-number*	显示特定成员的信息。
show switch detail	显示堆栈的详细信息。
show switch neighbors	显示堆栈邻居。
show switch stack-ports [**summary**]	显示堆栈的端口信息。
show redundancy	显示冗余系统及当前处理器信息。冗余系统信息包括系统工作时长、备用故障、切换原因、硬件、配置冗余模式及运行冗余模式。显示的当前处理器信息包括活跃位置、软件状态以及处于当前状态中的时长。
show redundancy state	显示活跃及备用设备的所有冗余状态。

交换机堆栈配置示例

交换机堆栈配置场景

以下多数交换机堆栈配置场景都假设至少有两台设备通过其StackWise-160端口相连。

表 171：配置场景

场景		结果
活跃交换机选举由现有活跃交换机决定	通过 StackWise-160 端口连接两个开机的交换机堆栈。	两个活跃交换机中只有一个能成为新的活跃交换机。
活跃交换机选举由堆栈成员优先级值决定	1 通过 StackWise-160 端口连接两台交换机。 2 使用全局配置命令 **switch***stack-member-number* **priority***new-priority-number*来设置一个交换机成员使用更高的成员优先级值。 3 同时重启两个堆栈成员。	有更高优先级值的堆栈成员会被选举为活跃交换机。
活跃交换机选举由配置文件决定	假设两个堆栈成员有相同的优先级值： 1 确保一个堆栈成员有默认配置，而另一个堆栈成员有保存（非默认）的配置文件。 2 同时重启两个堆栈成员。	有保存配置文件的堆栈成员会被选举为活跃交换机。
活跃交换机选举由 MAC 地址决定	假设两个堆栈成员有相同的优先级值、配置文件及特性集，同时重启两个堆栈成员股。	有较低 MAC 地址的堆栈成员会被选举为活跃交换机。
堆栈成员编号冲突	假设一个堆栈成员的优先级比另一个堆栈成员高： 1 确保两个堆栈成员有相同的堆栈成员编号。需要时可以使用全局配置命令**switch***current-stack-member-number***renumber***new-stack-member-number*。 2 同时重启两个堆栈成员。	有更高优先级的堆栈成员会保持自己的堆栈成员编号。另一个堆栈成员会使用新的堆栈成员编号。

添加堆栈成员	1 关闭新交换机。 2 通过 StackWise-160 端口把新交换机连接到已开机的交换机堆栈。 3 开启新交换机。	活跃交换机被保留。新交换机会被添加到交换机堆栈中。
活跃交换机故障	移除（或关闭）活跃交换机。	其余堆栈成员中的一个会成为新的堆栈 master。堆栈中的所有其他成员保持不变且不会重启。
添加超过九个堆栈成员	1 通过 StackWise-160 端口连接了 10 台设备。 2 打开所有设备。	两台设备会成为活跃交换机。一台活跃交换机有九个堆栈成员。另一台活跃交换机保持为单独的设备。 使用设备上的 Mode 按键以及端口 LED 灯来辨别哪台设备是活跃交换机以及每台活跃交换机有哪些所属设备。

示例：启用持续 MAC 地址特性

此示例展示了如何配置持续 MAC 地址特性使用 7 分钟的时延，并验证了配置。

```
Device(config)# stack-mac persistent timer 7
WARNING: The stack continues to use the base MAC of the old Master
WARNING: as the stack MAC after a master switchover until the MAC
WARNING: persistency timer expires. During this time the Network
WARNING: Administrators must make sure that the old stack-mac does
WARNING: not appear elsewhere in this network domain. If it does,
WARNING: user traffic may be blackholed.
Device(config)# end
Device# show switch
Switch/Stack Mac Address : 0016.4727.a900
Mac persistency wait time: 7 mins
H/W Current
Switch# Role Mac Address Priority Version State
-----------------------------------------------------------
*1 Active 0016.4727.a900 1 P2B Ready
```

示例：规划交换机堆栈的新成员

此示例展示了如何为交换机堆栈规划一个堆栈成员编号为 2 的交换机。命令 **show running-config** 的输出显示了与规划交换机相关的接口。

```
Device(config)# switch 2 provision switch_PID
Device(config)# end
```

```
Device# show running-config | include switch 2
```

其他参考资料

相关文档

相关主题	文档标题
交换机堆栈的连线及供电	Inspur 6650 交换机硬件安装指南 http://www.icntnetworks.com
SGACL 高可用性	*Inspur TrustSec 交换机配置指南*的“Inspur TrustSec SGACL 高可用性”模块。

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

标准以及 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

预防未授权访问

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

预防未授权访问

用户可以通过配置本地交换机，以及查看配置信息，来预防未授权的用户访问。通常情况下，用户会希望网络管理员能够访问交换机，同时防止网络外的用户通过异步端口拨入到交换机中，或者通过串行端口从网络外连接到交换机，或者通过本地网络中的终端或工作站连接到交换机。

要想预防交换机接受未授权的访问，用户应该配置以下安全特性之一：

- 最低限度，用户应该为每个交换机端口配置密码和特权级别。这些密码是储存在交换机本地的。当用户尝试通过一个端口或线路访问交换机时，他/她们必须输入这个端口或线路上指定的密码，才能获得交换机的访问权限；
- 为了实施更高一层的安全性，用户也可以配置用户名和密码对，这些信息也是储存在交换机本地的。用户可以把这些信息对分配给线路或端口，并在用户访问交换机之前对其进行认证。如果用户定义了特权级别，还能够为每个用户名和密码对分配特定的特权级别（关联着权利和特权）；
- 如果用户希望使用用户名和密码对，但希望集中把这些信息储存在服务器上，而不是保存在交换机本地，用户可以把它们储存到安全服务器的数据库中。之后多种网络设备都可以使用相同的数据库来获得用户认证（如果需要的话，还可以获得授权信息）信息；
- 用户还可以启用登录高级特性，它会记录失败的和未成功的登录尝试。登录高级特性也可以用来在用户进行了一定次数的未成功尝试后，阻止它未来一段时间的登录尝试。更多信息用户可以参考 Inspur INOS Login Enhancements 文档。

使用密码和特权级别控制交换机访问

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

使用密码和特权来控制交换机访问的限制条件

使用密码和特权来控制交换机访问具有以下限制条件：

- 如果用户使用全局配置命令 **boot manual** 手动启动交换机的话，禁用密码发现功能将不会生效。这条命令会在交换机重新加电后，进入引导加载程序（*switch:*）；

第 85 章 密码和特权级别的相关信息

默认的密码和特权级别配置

在用户网络中提供终端访问控制的一种简单的方法是使用密码并分配特权级别。用户可以使用密码保护来对网络或网络设备的访问实施限制。特权级别定义了用户在登录到网络设备后，能够使用的命令。
下面这个表格中展示了默认的密码和特权级别配置。

表 122：默认的密码和特权级别

特性	**默认设置**
启用密码和特权级别	未定义密码。默认级别是级别 15（特权 EXEC 级别）。密码在配置文件中是未加密的
启用秘密密码和特权级别	未定义密码。默认级别是级别 15（特权 EXEC 级别）。密码在配置文件中是加密的

线路密码	未定义密码

其他密码安全特性

要想提供更高层次的安全性，尤其是涉及在网络中传输的密码，以及储存在简单文件传输协议（TFTP）服务器上的密码，用户可以使用全局配置命令 **enable password** 或 **enable secret**。这两条命令都提供了相同的功能：也就是用户必须输入一个加密密码，才能访问特权 EXEC 模式（默认）或任意特权级别。
我们建议用户使用 **enable secret** 命令，因为它使用了增强的加密算法。
如果用户配置了 **enable secret** 命令，这条命令的优先级会高于 **enable password** 命令；这两条命令不能同时生效。
如果用户启用了密码加密特性，这个特性会应用在所有密码上，其中包括用户密码、加密密钥密码、特权命令密码，以及控制和虚拟终端线路密码。

密码恢复

默认情况下，任意终端用户通过物理的方式介入到交换机上，都可以通过在交换机加电过程中，打断启动进程并输入新密码，来恢复交换机密码。
密码恢复禁用特性能够通过禁用这个功能中的一部分，保护他人对交换机的访问。当用户启用了这个特性时，终端用户只有同意把系统恢复为默认配置，才可以打断启动进程。在密码恢复禁用后，用户仍可以打断启动进程并更改密码，但交换机的配置文件（config.text）和 VLAN 数据库文件（vlan.dat）都会被删除。
如果用户禁用了密码恢复特性，我们建议用户在一台安全服务器上保留配置文件的副本，以防终端用户打断了启动进程，而使系统恢复默认配置。不要在交换机上备份配置文件的副本。如果交换机运行在 VTP 透明模式的话，我们建议用户也要在安全服务器上备份 VLAN 数据库文件的副本。当交换机恢复默认系统配置后，用户可以使用 Xmodem 协议，把这些保存的文件下载到交换机上。
要想重新启用密码恢复特性，用户需要使用全局配置命令 **service password-recovery**。

终端线路 Telnet 配置

当用户第一次启动交换机时，可以使用一个自动设置程序，来分配 IP 地址并创建后续使用的默认配置。设置程序也会提示用户为通过 Telnet 访问交换机这种方式配置一个密码。如果用户没有在设置程序中配置这个密码，也可以在设置终端线路时设置 Telnet 密码。

用户名密码对

用户可以配置用户名和密码对，这些是储存在交换机本地的信息。用户可以把这些信息对分

配给线路或端口，并在用户访问交换机之前对其进行认证。如果用户定义了特权级别，还能够为每个用户名和密码对分配特定的特权级别（关联着权利和特权）。

特权级别

Inspur 交换机（和其他设备）能够使用特权级别来为不同的交换机操作级别提供密码保护机制。默认情况下，Inspur INOS 软件运行在两种密码安全模式（特权级别）中：用户 EXEC（级别 1）和特权 EXEC（级别 15）。用户可以为每个模式配置 16 个命令层级。通过配置多个密码，用户可以允许不同的用户集合访问指定的命令。

线路上的特权级别

用户可以通过登录到线路中并启用不同的特权级别，来覆盖使用线路配置命令 **privilege level** 设置的特权级别。用户可以通过使用 **disable** 命令来降低特权级别。如果用户知道更高特权级别的密码，也可以使用密码来启用更高的特权级别。用户可能会为 Console 线路指定高级别或特权级别的访问权限，以此来限制对线路的使用。

举例来说，如果用户希望多个用户使用 **clear line** 命令，可以为其分配级别 2 安全等级，并广泛分发级别 2 密码。如果用户希望对 **configure** 命令实施更严格的访问限制，可以为其分配级别 3 安全等级，并把级别 3 密码严格限制在一组用户范围内。

命令特权级别

当用户把一条命令设置为特权级别时，如果这条命令的语法属于某个命令子集，则相应的命令也都会被设置为这个特权级别。举例来说，如果用户把 **show ip traffic** 命令设置为级别 15，那么 **show** 命令和 **show ip** 命令也会自动被设置为级别 15，除非用户分别把它们设置为不同的级别。

如何使用密码和特权级别来控制交换机访问

设置或更改静态 enable 密码

enable 密码用来控制用户访问特权 EXEC 模式的行为。用户可以按照以下步骤来设置或更改静态 enable 密码。

总步骤

1. enable

2. configure terminal

3. enable password *password*

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的

步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**enable password** *password* **示例：** `Device(config)# enable password secret321`	定义一个新密码或更改现有密码，来访问特权 EXEC 模式。 默认情况下没有定义密码。 在 *password* 部分指定一个长度为 1 至 25 的字母和数字字符串。字符串不能以数字开头、需要区分大小写，并且可以使用空格，但会忽略开头的空格。用户可以使用问号（?），但需要在问号前输入 Ctrl-v；举例来说，要想创建密码 abc?123，用户需要这样输入： **1** 输入 **abc** **2** 输入 **Ctrl-v** **3** 输入**?123** 在系统提示用户输入 enable 密码时，用户无需在问号前输入 Ctrl-v，只要在密码提示符后面输入 abc?123 就可以
步骤 4	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 6	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

使用加密特性保护 enable 密码和 enable 秘密密码

用户可以按照以下步骤建立一个加密密码，也就是要想进入特权 EXEC 模式（默认）或任何指定的特权级别，就必须输入这个密码：

总步骤：

1. enable

2. configure terminal

3. Use one of the following:

• **enable password** [**level** *level*]

{*password* | *encryption-type encrypted-password*}

• **enable secret** [**level** *level*]

{*password* | *encryption-type encrypted-password*}

4. service password-encryption

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 3	用户可以使用以下命令之一：
	• **enable password** [**level** *level*] {*password* | *encryption-type encrypted-password*}
	• **enable secret** [**level** *level*] {*password* | *encryption-type encrypted-password*}
	
	示例：
	`Device(config)# enable password example102`
	或者
	`Device(config)# enable secret level 1 password secret123sample`
		• 定义一个新密码或更改一个已有密码，来访问特权 EXEC 模式。
		• 定义一个秘密密码，这个密码是使用不可逆的加密模式保存在交换机中的。
		· （可选）*level* 字段的取值范围是 0 至 15。级别 1 是通常用于用户 EXEC 模式的特权。默认级别是 15（特权 EXEC 模式的特权）
		· 在 *password* 部分指定一个长度为 1 至 25 的字母和数字字符串。字符串不能以数字开头、需要区分大小写，并且可以使用空格，但会忽略开头的空格。默认没有指定密码
		· （可选）*encryption-type* 只有类型 5 可用，这是 Inspur 私有加密算法。如果用户指定了加密类型，就必须提供加密密码——从其他交换机配置中复制的加密密码
		注释： 如果用户指定了加密类型，

		并输入了一个明文密码，就无法再次进入特权 EXEC 模式了。用户无法使用任何方式恢复丢失的加密密码
步骤 4	**service password-encryption** **示例：** `Device(config)# service password-encryption`	（可选）在定义密码是或重写配置时加密这个密码。 加密特性能够防止他人在配置中读取密码
步骤 5	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

禁用密码恢复

用户可以按照以下步骤来禁用密码恢复特性，以此保护交换机的安全：

在开始前

如果用户禁用了密码恢复特性，我们建议用户在一台安全服务器上保留配置文件的副本，以防终端用户打断了启动进程，而使系统恢复默认配置。不要在交换机上备份配置文件的副本。如果交换机运行在 VTP 透明模式的话，我们建议用户也要在安全服务器上备份 VLAN 数据库文件的副本。当交换机恢复默认系统配置后，用户可以使用 Xmodem 协议，把这些保存的文件下载到交换机上。

总步骤

1. enable

2. configure terminal

3. system disable password recovery switch {*all* | *<1-9>*}

4. end

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal**	进入全局配置模式

	示例： `Device# configure terminal`	
步骤 3	**system disable password recovery switch** {*all* | *<1-9>*} 示例： `Device(config)# system disable password recovery switch all`	禁用密码恢复特性。 • *all*——设置堆栈中交换机上的配置 • *<1-9>*——设置所选交换机编号上的配置 这个设置保存在 Flash 中可由引导加载程序和浪潮 INOS 映像访问的区域中，但它不是文件系统的一部分，任何用户都无法访问
步骤 4	**end** 示例： `Device(config)# end`	返回特权 EXEC 模式

接下来做什么？

要想删除 **disable password recovery** 命令，用户需要使用全局配置命令 **no system disable password recovery switch all**。

为终端线路设置 Telnet 密码

从用户 EXEC 模式开始，用户可以按照以下步骤为直连的终端线路配置 Telnet 密码：

在开始前

- 把安装有模拟软件的 PC 或工作站连接到交换机的 Console 端口，或把 PC 连接到以太网管理端口；
- Console 端口的默认数据特征是 9600、8、1、无隐私。用户可能需要多次按下回车键才能够看到命令行提示符。

总步骤

1. enable

2. configure terminal

3. line vty 0 15

4. password *password*

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	**注释：** 如果访问特权 EXEC 模式需要密码，系统会向用户进行提示。 进入特权 EXEC 模式
步骤 2	**configure terminal**	进入全局配置模式

	示例： Device# **configure terminal**	
步骤 3	**line vty 0 15** 示例： Device(config)# **line vty 0 15**	配置 Telnet 会话（线路）的编号，并进入线路配置模式。 在支持命令的设备上共有 16 条会话。0 和 15 表示用户要配置所有 16 条 Telnet 会话
步骤 4	**password** *password* 示例： Device(config-line)# **password abcxyz543**	为一条或多条线路设置 Telnet 密码。在 *password* 部分指定一个长度为 1 至 25 的字母和数字字符串。字符串不能以数字开头、需要区分大小写，并且可以使用空格，但会忽略开头的空格。默认没有指定密码
步骤 5	**end** 示例： Device(config)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config** 示例： Device# **show running-config**	检查用户输入的信息
步骤 7	**copy running-config startup-config** 示例： Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

配置用户名和密码对

用户可以按照以下步骤来配置用户名和密码对：

总步骤

1. enable

2. configure terminal

3. username *name* [**privilege** *level*] {**password** *encryption-type password*}

4. 使用以下命令之一：

• line console 0

• line vty 0 15

5. login local

6. end

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式

步骤 3	**username** *name* [**privilege** *level*] {**password** *encryption-type password*}
	
	示例：
	`Device(config)# username adamsample privilege 1 password secret456`
	`Device(config)# username 111111111111 mac attribute`
		为每个用户设置用户名、特权级别和密码。
		• 在 *name* 部分把用户 ID 设置为一个单词或 MAC 地址。不能使用空格和问号
		• 用户最多可以为用户名和MAC过滤器配置 12000 个客户端
		• （可选）*level* 字段定义的是用户获得访问权限后，能够使用的特权级别。取值范围是 0 至 15。级别 15 是特权 EXEC 模式的特权。级别 1 是用户 EXEC 模式的特权
		• 在 *encryption-type* 部分输入 0 指定未加密密码，输入 7 指定隐藏密码
		• （可选）*password* 部分指定的是用户必须用来获得设备访问权限的密码。密码长度必须为 1 至 25 个字符，其中可以包含空格，并且必须是 **username** 命令中指定的最后一个选项

步骤 4	用户可以使用以下命令之一：
	• **line console 0**
	• **line vty 0 15**
	
	示例：
	`Device(config)# line console 0`
	或者
	`Device(config)# line vty 15`
		进入线路配置模式，并配置 Console 端口（线路 0）或 VTY 线路（线路 0 至 15）

步骤 5	**login local** **示例：** `Device(config-line)# login local`	在登录时启用密码检查。基于步骤 3 中指定的用户名进行认证

步骤 6	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 7	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 8	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

为一条命令设置特权级别

用户可以使用以下命令来为一条命令设置特权级别：

总步骤

1. enable

2. configure terminal

3. privilege *mode* **level** *level command*

4. enable password level *level password*

5. end

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**privilege** *mode* **level** *level command* **示例：** `Device(config)# privilege exec level 14 configure`	为一条命令设置特权级别： • 在 *mode* 部分输入 **configure** 指定全局配置模式，**exec** 指定 EXEC 模式，**interface** 指定接口配置模式，或者 **line** 指定线路配置模式 • 在 *level* 部分指定 0 至 15 之间的值。级别 1 表示普通用户 EXEC 模式的特权，级别 15 表示需要使用 **enable** 密码进入的特权模式

		• 在 *command* 部分指定用户想要限制访问的命令
步骤 4	**enable password level** *level password* **示例：** `Device(config)# enable password level 14 SecretPswd14`	指定启用某个特权级别的密码。 • 在 *level* 部分指定 0 至 15 之间的值。级别 1 表示普通用户 EXEC 模式的特权 • 在 *password* 部分指定一个长度为 1 至 25 的字母和数字字符串。字符串不能以数字开头、需要区分大小写，并且可以使用空格，但会忽略开头的空格。默认没有指定密码
步骤 5	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 6	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

为线路改变默认的特权级别

用户可以按照以下步骤为指定线路更改默认的特权级别：

总步骤

1. enable

2. configure terminal

3. line vty *line*

4. privilege level *level*

5. end

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**line vty** *line*	选择用户想要限制访问的虚拟终端线

	示例： `Device(config)# line vty 10`	路
步骤 4	**privilege level** *level* **示例：** `Device(config)# privilege level 15`	为线路更改默认的特权级别。 在 *level* 部分指定 0 至 15 之间的值。级别 1 表示普通用户 EXEC 模式的特权，级别 15 表示需要使用 **enable** 密码进入的特权模式
步骤 5	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 6	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

接下来做什么？

用户可以通过登录到线路中并启用不同的特权级别，来覆盖使用线路配置命令 **privilege level** 设置的特权级别。用户可以通过使用 **disable** 命令来降低特权级别。如果用户知道更高特权级别的密码，也可以使用密码来启用更高的特权级别。用户可能会为 Console 线路指定高级别或特权级别的访问权限，以此来限制对线路的使用。

登录和离开一个特权级别

从用户 EXEC 模式开始，用户可以按照以下步骤登录指定的特权级别，以及离开指定的特权级别。

总步骤

1. enable *level*

2. disable *level*

具体步骤

	命令或操作	**目的**
步骤 1	**enable** *level* **示例：** `Device> enable 15`	登录到指定特权级别中。 示例中 15 表示特权 EXEC 模式。 在 *level* 部分输入 0 至 15 之间的值
步骤 2	**disable** *level* **示例：** `Device# disable 1`	离开指定特权级别

监控交换机的访问

表 123：显示 DHCP 信息的命令

命令	目的
show privilege	显示特权级别的命令

设置密码和特权级别的配置示例

示例：设置或更改静态 enable 密码

这个示例展示了如何把 enable 密码更改为 *l1u2c3k4y5*。这个密码不加密，提供级别 15 的访问特权（传统特权 EXEC 模式的访问）：

```
Device(config)# enable password l1u2c3k4y5
```

示例：使用加密特性保护 enable 密码和 enable 秘密密码

这个示例展示了如何为特权级别 2 配置加密密码*1FaD0$Xyti5Rkls3LoyxzS8*：

```
Device(config)# enable secret level 2 5 $1$FaD0$Xyti5Rkls3LoyxzS8
```

示例：为终端线路设置 Telnet 密码

这个示例展示了如何把 Telnet 密码设置为 *let45me67in89*：

```
Device(config)# line vty 10
Device(config-line)# password let45me67in89
```

示例：为一条命令设置特权级别

这个示例展示了如何使用 **configure** 命令设置特权级别 14，并把用户用来进入特权级别 14 的密码设置为 *SecretPswd14*：

```
Device(config)# privilege exec level 14 configure
Device(config)# enable password level 14 SecretPswd14
```

其他参考资料

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关	http://www.icntnetworks.com

的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

配置 TACACS+

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置 TACACS+的先决条件

在使用 TACACS+来设置和配置交换机访问时，有以下先决条件（必须按顺序执行）：

1. 在交换机上配置 TACACS+服务器的地址；

2. 设置一个认证密钥；
3. 在 TACACS+服务器上配置步骤 2 指定的密钥；
4. 启用认证、授权和审计（AAA）；
5. 创建一个登录认证方式列表；
6. 在终端线路上应用列表；
7. 创建授权和审计方式列表。

在使用 TACACS+来控制交换机访问时，有以下先决条件：

- 用户必须有配置 TACACS+服务器的能力，才能在交换机上配置 TACACS+特性。并且用户必须能够访问（通常运行在 LINUX 或 Windows 工作站上）TACACS+守护程序上的数据库中维护的 TACACS+服务；
- 我们建议在交换机堆栈和 TACACS+服务器之间建立冗余连接。这是为了确保在交换机堆栈中的某个堆栈成员被移除后，堆栈仍能够访问 TACACS+服务器；
- 用户需要运行 TACACS+守护程序的系统，才能在交换机上使用 TACACS+；
- 要想使用 TACACS+，用户必须启用它；
- 用户必须启用授权，交换机才能使用授权；
- 用户必须首先成功完成 TACACS+认证，才能执行 TACACS+授权；
- 要想使用这部分或其他文档中列出的 AAA 命令，用户必须首先使用 **aaa new-model** 命令启用 AAA；
- 最起码用户必须对维护 TACACS+守护程序的一台或多台主机进行标识，并定义 TACACS+认证的方法列表。用户可以（可选的）定义 TACACS+授权和审计的方法列表；
- 方法列表中定义了要执行的认证的类型，以及执行它们的顺序；在它能够执行任何定义的身份验证方法之前，用户必须把它应用在特定的端口上。唯一的例外是默认方法列表（巧合的是，它就名为 *default*）。默认方法列表会自动应用在所有端口上，除了已经明确定义了方法列表名称的端口。用户定义的方法列表会覆盖默认方法列表；
- 如用用户使用 TACACS+执行认证，也可以使用 TACACS+执行特权 EXEC 访问的授权；
- 如果没有使用 TACACS+执行认证，可以使用本地数据库执行认证。

TACACS+的相关信息

TACACS+和交换机访问

这部分描述了 TACACS+。TACACS+提供了详细的审计信息，以及对认证和授权过程的灵活管理控制。它能够提供认证、授权、审计（AAA），并且只能通过 AAA 命令进行启用。

TACACS+概述

TACACS+是一项安全应用，以集中的方式对尝试获得交换机访问权限的用户进行认证。TACACS+提供了分离和模块化的认证、授权和审计功能。TACACS+允许使用每台访问控制服务器（TACACS+守护程序）来独立地提供每一项服务——认证、授权和审计。每台服务器可以与自己的数据库相绑定，并根据守护程序的功能，利用服务器自身的服务，或网络上可用的其他服务。

TACACS+的目标是提供一种方法，用单个管理服务来管理多个网络接入点。用户交换机可以是网络访问服务器，或者其他 Inspur 路由器和访问服务器。

图 98：典型的 TACACS+网络配置

UNIX workstation (TACACS+ server 1)	UNIX 工作站 （TACACS+ 服务器 1）
Catalyst 6500 series switch	Inspur 6500 系列交换机
UNIX workstation (TACACS+ server 2)	UNIX 工作站 （TACACS+ 服务器 2）
Workstations（共 2 处）	工作站
Configure the switches with the TACACS+ server addresses. Set an authentication key (also configure the same key on the TACACS+ servers). Enable AAA. Create a login authentication method list. Apply the list to the terminal lines. Create an authorization and accounting method list as required.	在交换机上配置 TACACS+服务器的地址 设置一个认证密钥（也要在 TACACS+服务器上配置相同的密钥） 启用 AAA 创建登录认证方法列表 在终端线路上应用列表 按需创建授权和审计方法列表

用户需要通过 AAA 安全服务对 TACACS+进行管理，TACACS+可以提供以下功能：

- 认证——通过登录和密码对话、质询和响应，以及消息传递，来提供完全受控的身份认证。

  认证功能可以与用户进行对话（举例来说，在提供了用户名和密码之后，使用以下问题来质询用户：比如家庭地址、母亲的姓氏、服务类型和身份证号码）。TACACS+认证服务还可以向用户屏幕发送消息。比如它可以通过消息来通知用户：由于公司的密码老化策略，用户必须更改自己的密码；
- 授权——在用户会话期间对用户功能提供细粒度的控制，这些控制包括但不限于：设置自动命令、访问控制、会话持续时间，或协议支持。用户还可以使用 TACACS+授权功能，限制用户所能够执行的命令；
- 审计——收集用于记帐、审计和报告的信息，并将其发送到 TACACS+守护进程。网络管理员可以使用审计功能来跟踪用于安全审计的用户活动，或者提供用于用户计费的信息。审计记录中包括用户身份、开始和停止时间、执行的命令（比如 PPP）、数据包数量，以及字节数。

TACACS+协议在交换机和 TACACS+守护程序之间提供了认证功能，并且它能够保证机密性，因为交换机和 TACACS+守护程序之间的所有协议交换消息都进行了加密。

TACACS+的工作原理

当用户使用 TACACS+来对简单的 ASCII 登录尝试进行认证时，会发生以下过程：

1. 当连接建立时，交换机会联系 TACACS+守护程序，从而向用户显示出输入用户名的提示

符。用户输入用户名后，交换机会再联系 TACACS+守护程序，从而向用户显示出输入密码的提示符。交换机向用户显示输入密码的提示符后，用户输入密码，之后交换机会把密码发送到 TACACS+守护程序。

TACACS+支持守护程序和用户之间的对话，直到守护程序收到足够多的信息来认证用户为止。守护程序能够提示用户输入用户名和密码组合，但可以包括其他内容，比如用户母亲的姓氏；

2. 交换机最终会从 TACACS+守护程序那里收到以下响应之一：
   - ACCEPT（接受）——用户通过了认证，并开始获得服务。如果交换机上配置了授权，则现在开始进行授权；
   - REJECT（拒绝）——用户没有通过认证。并根据 TACACS+守护程序，拒绝用户的访问，或者为用户提示重试登录；
   - ERROR（错误）——在使用守护程序进行身份认证的过程中发生错误，或守护程序与交换机之间的网络连接发生错误。如果接收到了 ERROR 响应，交换机通常尝试使用替代方法来对用户进行认证；
   - CONTINUE（继续）——为用户提示其他认证信息。

   如果用户在交换机上启用了授权，那么在用户通过认证之后，会经历额外的授权阶段。用户必须首先成功完成 TACACS+身份验证，然后才能进行 TACACS+授权。
3. 如果需要使用 TACACS+授权，交换机会再次联系 TACACS+守护进程，并接收到 ACCEPT 或 REJECT 授权响应。如果交换机收到的是 ACCEPT 响应，则响应中包含了属性数据，指出用户提供的 EXEC 或 NETWORK 会话，以及用户可以访问的服务：
   - Telnet、安全壳（SSH）、远程登录，或特权 EXEC 服务；
   - 连接参数，其中包括主机或客户端 IP 地址、访问列表和用户超时时间。

方法列表

方法列表定义了为用户进行认证、授权或审计的序列和方法。用户可以使用方法列表来指定要使用的一个或多个安全协议，这样做可以在初始方法失败时，确保有一个备份系统。软件使用列表中的第一个方法来对用户进行认证、授权或审计；如果该方法没有获得响应，软件会选择列表中的下一个方法。这个过程会持续直到使用列出的方法实现成功通信，或者持续到方法列表耗尽。

TACACS+的配置选项

用户可以配置交换机来使用单台 AAA 服务器或 AAA 服务器组，为现有的服务器主机进行身份认证。用户可以有选择地把一部分服务器主机设置为一组服务器，并将其用于提供特定服务。服务器组与全局服务器主机列表一起使用，还包含所选服务器主机的 IP 地址列表。

TACACS+登录认证

方法列表定义了为用户进行认证的序列和方法。用户可以使用方法列表来指定要使用的一个或多个安全协议，这样做可以在初始方法失败时，确保有一个备份系统。软件使用列表中的第一个方法来对用户进行认证、授权或审计；如果该方法没有获得响应，软件会选择列表中

的下一个方法。这个过程会持续直到使用列出的方法实现成功通信，或者持续到方法列表耗尽。如果在这个周期中的任何时刻认证失败了——意味着安全服务器或本地用户名数据库发出了响应，拒绝了用户的访问——这时认证过程就停止了，并且不会再尝试其他认证方法。

为特权 EXEC 访问和网络服务使用 TACACS+授权

AAA 授权能够限制用户可以使用的服务。当用户启用了 AAA 授权后，交换机会使用从用户配置文件中检索的信息来配置用户的会话，这个配置文件位于本地用户数据库中，或位于安全服务器。只有当用户配置文件中的信息允许时，用户才能够访问所请求的服务。

TACACS+审计

AAA 审计功能会跟踪用户正在访问的服务，以及用户消耗的网络资源总量。当用户启用了 AAA 审计后，交换机以审计记录的形式向 TACACS+安全服务器报告用户的活动。每个审计记录中都包含了审计属性值（AV）对，并且这些信息存储在安全服务器上。之后用户可以使用这些数据来对网络管理、客户端计费或审计进行分析。

默认的 TACACS+配置

TACACS+和 AAA 默认都是禁用的。
为了防止安全性失效，用户不能通过网络管理应用程序来配置 TACACS+。当启用了 TACACS+时，TACACS+可以验证通过 CLI 访问交换机的用户。
注释： 虽然用户是通过 CLI 执行 TACACS+配置的，但是 TACACS+服务器认证会认证 HTTP 连接，并已经为这个连接配置了特权级别 15。

如何配置 TACACS+

这部分描述了如何配置交换机来支持 TACACS+。

标识 TACACS+服务器主机并设置认证密钥

用户可以按照以下步骤标识 TACACS+服务器主机并设置认证密钥：
总步骤
1. enable
2. configure terminal
3. tacacs-server host *hostname*
4. aaa new-model
5. aaa group server tacacs+ *group-name*
6. server *ip-address*

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**tacacs-server host** *hostname* **示例：** `Device(config)# tacacs-server host yourserver`	标识维护 TACACS+服务器的一台或多台 IP 主机。用户需要多次输入这条命令，来创建相应的主机列表。软件会按照用户配置的顺序来搜索这些主机。 在 *hostname* 部分指定主机的名称或 IP 地址
步骤 4	**aaa new-model** **示例：** `Device(config)# aaa new-model`	启用 AAA
步骤 5	**aaa group server tacacs+** *group-name* **示例：** Device(config)# **aaa group server tacacs+ your_server_group**	（可选）使用组名定义一个 AAA 服务器组。 这条命令会让设备进入服务器组子配置模式
步骤 6	**server** *ip-address* **示例：** `Device(config)# server 10.1.2.3`	（可选）把指定的 TACACS+服务器关联到定义的服务器组中。用户需要重复配置这条命令，以便把所有相关的 TACACS+服务器都放入 AAA 服务器组中。 用户必须先通过步骤 3 来定义这些需要被放入组中的服务器
步骤 7	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 8	**show running-config** **示例：**	检查用户输入的信息

	Device# **show running-config**	
步骤 9	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

配置 TACACS 登录认证

用户可以按照以下步骤来配置 TACACS+登录认证：

在开始前

要想配置 AAA 认证，用户需要定义一个命名的认证方法列表，并把它应用给各种端口。

注释： 要想使用 AAA 方法保障 HTTP 访问的安全性，用户必须还得配置全局配置命令 **ip http authentication aaa**。只配置 AAA 认证功能并不能通过使用 AAA 方法来确保 HTTP 访问的安全性。

更多有关命令 **ip http authentication** 的信息，用户可以参考 *Inspur INOS Security Command Reference, Release 12.4*。

总步骤

1. enable

2. configure terminal

3. aaa new-model

4. aaa authentication login {**default** | *list-name*} *method1* [*method2...*]

5. line [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]

6. login authentication {**default** | *list-name*}

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**aaa new-model** **示例：** Device(config)# **aaa new-model**	启用 AAA
步骤 4	**aaa authentication login** {**default** |	创建一个登录认证方法列表。

	list-name} *method1* [*method2...*] **示例：** `Device(config)#` **`aaa authentication login default tacacs+ local`**	• 要想创建一个默认列表，当用户*没有*在 **login authentication** 命令中指定命名列表时就使用这个默认列表，用户需要在默认情况下使用的方法后面添加**default**关键字。默认方法列表会自动被应用到所有端口 • 在 *list-name* 部分指定一个字符串，用来命名用户创建的列表 • 在 *method1…*部分指定认证算法使用的实际方法。其他认证方法只有当前一个方法返回了错误响应消息时才会使用，而不是返回失败消息时使用。 用户可以选择以下方法之一： • *enable——*使用 enable 密码来进行认证。在用户可以使用这个认证方法前，必须使用全局配置命令 **enable** *password* 来定义一个 enable 密码 • *group tacacs+*——使用 TACACS+ 认证。在用户可以使用这个认证方法前，必须配置 TACACS+服务器。更多信息用户可以参考标识 TACACS+服务器主机并设置认证密钥 • *line*——使用线路密码来进行认证。在用户可以使用这个认证方法前，必须先定义一个线路密码。用户可以使用线路配置命令 **password** *password* • *local*——使用本地用户名数据库进行认证。用户必须输入数据库中的用户名信息。需要使用全局配置命令 **username** *password* 进行配置 • *local-case*——使用区分大小写的本地用户名数据库进行认证。用户必须使用全局配置命令 **username** *name* **password**，把用户名信息输入到数据库中 • *none*——不为登录使用任何认证
步骤 5	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]	进入线路配置模式，并对想要应用认证列表的线路进行配置

	示例： `Device(config)# line 2 4`	
步骤 6	**login authentication** {**default** | *list-name*} **示例：** `Device(config-line)# login authentication default`	把认证列表应用在一条或多条线路上。 • 如果用户指定了 **default**，就使用命令 **aaa authentication login** 创建默认列表 • 在 *list-name* 部分指定 **aaa authentication login** 命令中创建的列表
步骤 7	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 8	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 9	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

为特权 EXEC 访问和网络服务配置 TACACS+授权

用户可以在全局配置命令 **aaa authorization** 中使用 **tacacs+**关键字，来对用户访问特权 EXEC 模式的行为设置限制参数。

注释： 对于通过 CLI 登录且已通过了认证的用户，即使配置了授权，也会绕过授权。

用户可以按照以下步骤为特权 EXEC 访问和网络服务配置 TACACS+授权：

总步骤

1. enable

2. configure terminal

3. aaa authorization network tacacs+

4. aaa authorization exec tacacs+

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable**	进入特权 EXEC 模式。在提示时输入密码

	示例： Device> **enable**	
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**aaa authorization network tacacs+** **示例：** Device(config)# **aaa authorization network tacacs+**	配置交换机为所有与网络相关的服务请求使用 TACACS+授权
步骤 4	**aaa authorization exec tacacs+** **示例：** Device(config)# **aaa authorization exec tacacs+**	配置交换机为特权 EXEC 的访问使用 TACACS+授权。 **exec** 关键字可能会返回用户配置文件信息（比如 **autocommand** 信息）
步骤 5	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

开始使用 TACACS+审计

用户可以按照以下步骤开始使用 TACACS+审计：

总步骤

1. enable

2. configure terminal

3. aaa accounting network start-stop tacacs+

4. aaa accounting exec start-stop tacacs+

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**aaa accounting network start-stop tacacs+** **示例：** `Device(config)# aaa accounting network start-stop tacacs+`	为所有与网络相关的服务请求启用 TACACS+审计
步骤 4	**aaa accounting exec start-stop tacacs+** **示例：** `Device(config)# aaa accounting exec start-stop tacacs+`	启用 TACACS+审计，在开始特权 EXEC 处理时发送开始记录（start-record）审计通知，在结束时发送停止记录（stop-record）
步骤 5	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

接下来做什么？

如果在 AAA 服务器不可达时，要与路由器建立会话，用户可以使用 **aaa accounting system guarantee-first** 命令。这条命令可以确保系统审计为第一条记录，这也是默认的条件。在有些情况下，这种设置可能会阻止用户在 Console 或终端连接上启动会话，直到系统重启才能解决问题，这可能需要 3 分钟以上的时间。

如果路由器重启时 AAA 服务器不可达，要想与路由器建立 Console 或 Telnet 会话，用户可以使用 **no aaa accounting system guarantee-first** 命令。

在 AAA 服务器不可达时与路由器建立会话

如果在 AAA 服务器不可达时，要与路由器建立会话，用户可以使用 **aaa accounting system guarantee-first** 命令。这条命令可以确保系统审计为第一条记录，这也是默认的条件。在有些情况下，这种设置可能会阻止用户在 Console 或终端连接上启动会话，直到系统重启才能解决问题，这可能需要 3 分钟以上的时间。
如果路由器重启时 AAA 服务器不可达，要想与路由器建立 Console 或 Telnet 会话，用户可以使用 **no aaa accounting system guarantee-first** 命令。

监控 TACACS+

表 124：显示 TACACS+信息的命令

命令	目的
show tacacs	显示 TACACS+服务器的状态统计信息

MACsec 加密

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

MAC 加密的相关信息

本章描述了如何在 Inspur 6850 和 6650 交换机上配置媒介访问控制安全性（MACsec）加密特性。

MACsec 是 IEEE 802.1AE 标准，用来在两个具有 MACsec 功能的设备之间，为数据包提供认证和加密。Inspur 交换机能够在下行链路端口上支持使用 MACsec 密钥协定（MKA）的 802.1AE 加密，用来在交换机和主机设备之间进行加密。该交换机还支持使用 Inspur TrustSec 网络设备准入控制（NDAC）、安全关联协议（SAP）和基于 MKA 的密钥交换协议，为交换机到交换机（网络之间的设备）安全提供 MACsec 加密。链路层安全中也可以包含交换机之间的数据包认证，以及交换机之间的 MACsec 加密（加密是可选的）。

注释： NPE 许可或 LAN Base 服务镜像不支持 MACsec 特性。

表 125：交换机端口上支持的 MACsec

接口	**连接**	**对 MACsec 的支持**
下行链路端口	交换机到主机	MACsec MKA 加密
上行链路端口	交换机到交换机	MACsec MKA 加密 Inspur TrustSec NDAC Macsec

Inspur TrustSec 和 Inspur SAP 特性仅适用于交换机到交换机之间的链路，在连接终端主机（如 PC 或 IP 电话）的交换机端口上不支持这些特性。交换机到主机之间的链路（下行链路）以及交换机到交换机之间的链路（上行链路）上都支持 MKA 特性。面向主机的链路通常使用灵活的认证序列，来处理支持或不支持 IEEE 802.1x 的异构设备，并且还可以（可选地）使用基于 MKA 的 MACsec 加密。Inspur NDAC 和 SAP 特性，与网络边缘访问拓扑（NEAT）互不兼容，NEAT 用于紧凑型交换机，把安全性扩展到配线柜之外。

媒介访问控制安全性和 Macsec 密钥管理

MACsec 定义在 802.1AE 中，它通过对加密密钥使用带外方法，在有线网络上提供 MAC 层加密。MACsec 密钥协议（MKA）协议提供了需要使用的会话密钥，并且负责管理这个加密密钥。在使用 802.1x 可扩展认证协议（EAP-TLS）或预共享密钥（PSK）框架成功进行了认证后，便实施 MKA 和 MACsec。

根据与 MKA 对等体相关联的策略，使用 MACsec 特性的交换机能够接收 MACsec 帧或非 MACsec 帧。MACsec 帧是使用完整性校验值（ICV）进行加密和保护的。当交换机从 MKA 对等体接收到数据帧时，它会使用由 MKA 提供的会话密钥来对数据帧进行解密，并计算出正确的 ICV。交换机会把自己计算出的 ICV 与数据帧中携带的 ICV 进行比较。如果两者不相同，则交换机会丢弃这个数据帧。交换机还会使用当前的会话密钥，对通过安全端口（用来向 MKA 对等体提供安全 MAC 服务的接入点）发送的数据帧进行加密并添加 ICV。

MKA 协议会管理底层 MACsec 协议使用的加密密钥。MKA 的基本要求定义在 802.1x-REV 中。MKA 协议扩展了 802.1x，允许具有相互认证能力且共享 MACsec 密钥的对等体发现彼此，以此保护对等体之间交互的数据。

EAP 框架把 MKA 实现为新定义的 EAP-over-LAN（EAPOL）数据包。EAP 认证会产生一个主用会话密钥（MSK），数据交换中的两个对等体会共享这个 MSK。输入 EAP 会话 ID 会生成一个安全连接关联密钥名称（CKN）。交换机会对上行链路和下行链路进行认证；并充当下行链路的密钥服务器。它会生成一个随机安全关联密钥（SAK），并将其发送到客户端对等体。客户端永远不会是密钥服务器，并且只能与单个 MKA 实体（密钥服务器）进行交互。在派生和生成密钥后，交换机会以默认 2 秒钟的时间间隔周期性向对等体发送消息。

EAPOL 协议数据单元（PDU）数据包中的负载被称为 MACsec 密钥协商 PDU（MKPDU）。当 MKA 生命周期（6 秒钟）超时后，仍没有从对等体接收到 MKPDU，MKA 会话和对等体就会被删除。举例来说，如果 MKA 对等体断开了连接，交换机上的参与者会继续操作 MKA，直

到从 MKA 对等体接收到最后一个 MKPDU 之后又过去了 6 秒钟。

MKA 策略

要想在接口上启用 MKA，用户应该在接口上应用一个已经定义好的 MKA 策略。用户可以配置以下选项：

- 策略名称，不超过 16 个 ASCII 字符；
- 为每个物理接口配置保密（加密）偏移 0、30 或 50 字节。

虚拟端口

用户可以使用虚拟端口为单个物理端口提供多个安全连接关联。每个连接关联（对）代表着一个虚拟端口。在上行链路中，每个物理端口上只能有一个虚拟端口。在下行链路中，每个物理端口上最多可以有两个虚拟端口，其中一个虚拟端口可以是数据 VLAN 的一部分；另一个必须为语音 VLAN 来标记数据包。用户不能在同一端口上的同一 VLAN 中，同时承载安全会话和不安全的会话。正因有此限制，所以不支持 802.1x 多重身份验证模式。

对于这个限制有一个例外：在多主机模式下，第一个 MACsec 请求方成功通过认证，并且它通过一个集线器与交换机相连。这时集线器上连接的其他非 MACsec 主机也可以发送流量，而无需身份验证，因为这时使用的是多主机模式。我们不建议使用多主机模式，因为在第一个客户端验证成功后，其他客户端都不需要进行身份验证了。

虚拟端口是一个用来代表连接关联的任意标识符，并且除非用于 MKA 协议，否则是没有意义的。虚拟端口对应着单独的逻辑端口 ID。虚拟端口的有效端口号范围是 0x0002 至 0xFFFF。每个虚拟端口都会接收唯一的安全信道标识符（SCI），SCI 是由物理接口的 MAC 地址和 16 位端口 ID 构成的。

MACsec 和堆栈

Inspur 6850 交换机堆栈中运行 MACsec 的主用设备会维护配置文件，配置文件中展示了成员交换机上的哪些端口支持 MACsec。堆栈主用设备会执行以下功能：

- 处理安全通道和安全关联的创建和删除；
- 向堆栈成员发送安全关联服务请求；
- 处理来自本地或远端端口的数据包编号和响应窗口信息，并向密钥关系协议发送通知；
- 向新添加到堆栈的交换机发送 MACsec 初始化请求和全局配置的选项；
- 向成员交换机发送基于每个端口的配置。

成员交换机会执行以下功能：

- 处理来自堆栈主用设备的 MACsec 初始化请求；
- 处理由堆栈主用设备的 MACsec 服务请求；
- 向堆栈主用设备发送有关本地端口的信息。

MACsec、MKA 和 802.1x 主机模式

用户可以结合使用 MACsec 和 MKA 协议，以及 802.1x 单主机模式、多主机模式，或多域认

证（MDA）模式。不支持多认证模式。

单主机模式

下图展示了如何通过使用 MKA，由 MACsec 对单 EAP 认证会话提供保护的。

图 99：单主机模式的 MACsec 和受保护数据会话

Host	主机
Usecured	未受保护
Switch with MACsec configured	配置了 MACsec 的交换机
AAA Access-control system	AAA 访问控制系统

多主机模式

在标准的（非 802.1x REV）802.1x 多主机模式下，端口会基于单个认证进行打开或关闭。如果一个用户（主要受保护的客户端服务于客户端主机）通过了认证，交换机会向连接到同一端口的任意主机提供相同级别的网络访问。如果备用主机是 MACsec 请求方，则它不能被认证，并且流量也不会被转发。如果备用主机是非 MACsec 主机的话，则它可以在不进行认证的情况下向网络发送流量，因为它处于多主机模式中。下图显示了标准多主机未受保护模式下的 MACsec。

图 100：多主机模式中的 MACsec——未受保护

Primary host	主用主机
Secondary host（共 2 处）	备用主机
Switch with MACsec configured	配置了 MACsec 的交换机
AAA Access-control system	AAA 访问控制系统

注释： 不建议使用多主机模式，因为在第一个客户端成功认证后，其他客户端都不需要进行认证了，这种方式并不安全。

在标准的（非 802.1x REV）802.1x 多域模式下，端口是基于单个认证进行打开或关闭的。如果主用户（数据域上的 PC）通过了认证，交换机会为连接到同一端口上的任意域提供相同级别的网络访问。如果备用用户是 MACsec 请求方，则它不能被认证，并且流量也不会被转发。备用用户是指语音域上的 IP 电话（即非 MACsec 主机），它可以向网络中发送流量，无需进行身份验证，因为它处于多域模式。

MKA 状态统计信息

有些 MKA 计数器是在全局收集的，而其他 MKA 计数器是在全局和每个会话中进行更新的。用户还可以查看有关 MKA 会话状态的信息。

以下为 **show mka statistics** 命令的输出示例：

```
Switch# show mka sessions
Total MKA Sessions. ..............  1
Secured Sessions... 1
Pending Sessions... 0
=================================================================
==================================
Interface Local-TxSCI Policy-Name Inherited Key-Server
```

```
Port-ID Peer-RxSCI MACsec-Peers Status CKN
=================================================================
==================================
Gi1/0/1 204c.9e85.ede4/002b p2 NO YES
43 c800.8459.e764/002a 1 Secured
01000000000000000000000000000000000000000000000000000000000000
Switch#show mka sessions interface G1/0/1
Summary of All Currently Active MKA Sessions on Interface
GigabitEthernet1/0/1...
=================================================================
==================================
Interface Local-TxSCI Policy-Name Inherited Key-Server
Port-ID Peer-RxSCI MACsec-Peers Status CKN
=================================================================
==================================
Gi1/0/1 204c.9e85.ede4/002b p2 NO YES
43 c800.8459.e764/002a 1 Secured
01000000000000000000000000000000000000000000000000000000000000
Switch#show mka sessions interface G1/0/1 de
MKA Detailed Status for MKA Session
==================================
Status: SECURED - Secured MKA Session with MACsec
Local Tx-SCI..... 204c.9e85.ede4/002b
Interface MAC Address.... 204c.9e85.ede4
MKA Port Identifier. ..... 43
Interface Name..... GigabitEthernet1/0/1
Audit Session ID.....
CAK                      Name                      (CKN)...........
01000000000000000000000000000000000000000000000000000000000000
Member Identifier (MI)... D46CBEC05D5D67594543CEAE
Message Number (MN)..... 89567
EAP Role..... NA
Key Server..... YES
MKA Cipher Suite..... AES-128-CMAC
Latest SAK Status..... Rx & Tx
Latest SAK AN. ..... 0
Latest SAK KI (KN)..... D46CBEC05D5D67594543CEAE00000001 (1)
Old SAK Status..... FIRST-SAK
Old SAK AN. ..... 0
Old SAK KI (KN)..... FIRST-SAK (0)
SAK Transmit Wait Time... 0s (Not waiting for any peers to respond)
SAK Retire Time..... 0s (No Old SAK to retire)
MKA Policy Name..... p2
Key Server Priority. ..... 2
```

```
Delay Protection......... NO
Replay Protection........ YES
Replay Window Size. ............... 0
Confidentiality Offset... 0
Algorithm Agility........ 80C201
SAK Cipher Suite......... 0080C20001000001 (GCM-AES-128)
MACsec Capability........ 3 (MACsec Integrity, Confidentiality, &
Offset)
MACsec Desired........... YES
# of MACsec Capable Live Peers. ........................... 1
# of MACsec Capable Live Peers Responded.. 1
Live Peers List:
MI MN Rx-SCI (Peer) KS Priority
-----------------------------------------------------------------
-----
38046BA37D7DA77E06D006A9 89555 c800.8459.e764/002a 10
Potential Peers List:
MI MN Rx-SCI (Peer) KS Priority
-----------------------------------------------------------------
-----
Dormant Peers List:
MI MN Rx-SCI (Peer) KS Priority
-----------------------------------------------------------------
-----
Switch#show mka sessions de
Switch#show mka sessions detail
MKA Detailed Status for MKA Session
==================================
Status: SECURED - Secured MKA Session with MACsec
Local Tx-SCI.............. 204c.9e85.ede4/002b
Interface MAC Address.... 204c.9e85.ede4
MKA Port Identifier. ............... 43
Interface Name........... GigabitEthernet1/0/1
Audit Session ID.........
CAK                      Name                       (CKN)...........
01000000000000000000000000000000000000000000000000000000000000000000
Member Identifier (MI)... D46CBEC05D5D67594543CEAE
Message Number (MN)...... 89572
EAP Role................. NA
Key Server............... YES
MKA Cipher Suite......... AES-128-CMAC
Latest SAK Status........ Rx & Tx
Latest SAK AN. ............................ 0
Latest SAK KI (KN)....... D46CBEC05D5D67594543CEAE00000001 (1)
```

```
Old SAK Status.......... FIRST-SAK
Old SAK AN. ................................ 0
Old SAK KI (KN).......... FIRST-SAK (0)
SAK Transmit Wait Time... 0s (Not waiting for any peers to respond)
SAK Retire Time.......... 0s (No Old SAK to retire)
MKA Policy Name.......... p2
Key Server Priority. ............. 2
Delay Protection......... NO
Replay Protection........ YES
Replay Window Size. ............... 0
Confidentiality Offset... 0
Algorithm Agility........ 80C201
SAK Cipher Suite......... 0080C20001000001 (GCM-AES-128)
MACsec Capability........ 3 (MACsec Integrity, Confidentiality, &
Offset)
MACsec Desired........... YES
# of MACsec Capable Live Peers. ........................... 1
# of MACsec Capable Live Peers Responded.. 1
Live Peers List:
MI MN Rx-SCI (Peer) KS Priority
-----------------------------------------------------------------
-----
38046BA37D7DA77E06D006A9 89560 c800.8459.e764/002a 10
Potential Peers List:
MI MN Rx-SCI (Peer) KS Priority
-----------------------------------------------------------------
-----
Dormant Peers List:
MI MN Rx-SCI (Peer) KS Priority
-----------------------------------------------------------------
-----
Switch#sh mka pol
MKA Policy Summary...
Policy KS Delay Replay Window Conf Cipher Interfaces
Name Priority Protect Protect Size Offset Suite(s) Applied
=================================================================
====================================
*DEFAULT POLICY* 0 FALSE TRUE 0 0 GCM-AES-128
p1 1 FALSE TRUE 0 0 GCM-AES-128
p2 2 FALSE TRUE 0 0 GCM-AES-128 Gi1/0/1
Switch#sh mka poli
Switch#sh mka policy p2
Switch#sh mka policy p2 ?
detail Detailed configuration/information for MKA Policy
```

```
sessions Summary of all active MKA Sessions with policy applied
| Output modifiers
<cr>
Switch#sh mka policy p2 de
MKA Policy Configuration ("p2")
========================
MKA Policy Name........ p2
Key Server Priority. ......... 2
Confidentiality Offset. 0
Cipher Suite(s)........ GCM-AES-128
Applied Interfaces...
GigabitEthernet1/0/1
Switch#sh mka policy p2
MKA Policy Summary...
Policy KS Delay Replay Window Conf Cipher Interfaces
Name Priority Protect Protect Size Offset Suite(s) Applied
=================================================================
====================================
p2 2 FALSE TRUE 0 0 GCM-AES-128 Gi1/0/1
Switch#sh mka se?
sessions
Switch#sh mka ?
default-policy MKA Default Policy details
keychains MKA Pre-Shared-Key Key-Chains
policy MKA Policy configuration information
presharedkeys MKA Preshared Keys
sessions MKA Sessions summary
statistics Global MKA statistics
summary MKA Sessions summary & global statistics
Switch#sh mka statis
Switch#sh mka statistics ?
interface Statistics for a MKA Session on an interface
local-sci Statistics for a MKA Session identified by its Local Tx-
SCI
| Output modifiers
<cr>
Switch#sh mka statistics inter
Switch#show mka statistics interface G1/0/1
MKA Statistics for Session
==========================
Reauthentication Attempts.. 0
CA Statistics
Pairwise CAKs Derived... 0
Pairwise CAK Rekeys. ........... 0
```

```
Group CAKs Generated. ......... 0
Group CAKs Received. ........... 0
SA Statistics
SAKs Generated. ....................... 1
SAKs Rekeyed. ......................... 0
SAKs Received. ........................ 0
SAK Responses Received.. 1
MKPDU Statistics
Switch#show mka ?
default-policy MKA Default Policy details
keychains MKA Pre-Shared-Key Key-Chains
policy MKA Policy configuration information
presharedkeys MKA Preshared Keys
sessions MKA Sessions summary
statistics Global MKA statistics
summary MKA Sessions summary & global statistics
Switch#show mka summ
Switch#show mka summary
Total MKA Sessions. ............... 1
Secured Sessions... 1
Pending Sessions... 0
=====================================================================
===================================
Interface Local-TxSCI Policy-Name Inherited Key-Server
Port-ID Peer-RxSCI MACsec-Peers Status CKN
=====================================================================
===================================
Gi1/0/1 204c.9e85.ede4/002b p2 NO YES
43 c800.8459.e764/002a 1 Secured
01000000000000000000000000000000000000000000000000000000000000000000
MKA Global Statistics
=====================
MKA Session Totals Secured. ........
1
Reauthentication Attempts.. 0
Deleted (Secured). ..................... 0
Keepalive Timeouts. .................... 0
CA Statistics
Pairwise CAKs Derived. ............. 0
Pairwise CAK Rekeys. ................. 0
Group CAKs Generated. ................ 0
Group CAKs Received. ................. 0
SA Statistics
SAKs Generated. ............................. 1
```

```
SAKs Rekeyed. ................................ 0
SAKs Received. ............................... 0
SAK Responses Received. ........... 1
MKPDU Statistics
MKPDUs Validated & Rx...... 89589
"Distributed SAK". .......... 0
"Distributed CAK". .......... 0
MKPDUs Transmitted......... 89600
"Distributed SAK". .......... 1
"Distributed CAK". .......... 0
MKA Error Counter Totals
========================
Session Failures
Bring-up Failures..................................... 0
Reauthentication Failures. .................. 0
Duplicate Auth-Mgr Handle. .................. 0
SAK Failures
SAK Generation. ........................................... 0
Hash Key Generation. ............................... 0
SAK Encryption/Wrap. ............................... 0
SAK Decryption/Unwrap. ............................ 0
SAK Cipher Mismatch. ............................... 0
CA Failures
Group CAK Generation............................... 0
Group CAK Encryption/Wrap. .................. 0
Group CAK Decryption/Unwrap. ............. 0
Pairwise CAK Derivation. ...................... 0
CKN Derivation. ............................................ 0
ICK Derivation. ............................................ 0
KEK Derivation. ............................................ 0
Invalid Peer MACsec Capability... 0
MACsec Failures
Rx SC Creation. ............................................ 0
Tx SC Creation. ............................................ 0
Rx SA Installation. ................................. 0
Tx SA Installation. ................................. 0
MKPDU Failures
MKPDU
Tx. ..................................................................... 0
MKPDU Rx Validation. ............................... 0
MKPDU Rx Bad Peer MN.............................. 0
MKPDU Rx Non-recent Peerlist MN.. 0
Switch#
```

配置 MKA 和 MACsec

默认的 MACsec MKA 配置

默认 MACsec 是禁用的，也没有配置任何 MKA 策略。

配置 MKA 策略

总步骤

1. configure terminal

2. mka policy *policy name*

3. key-server *priority*

4. macsec-cipher-suite *gcm-aes-128*

5. confidentiality-offset *Offset value*

6. end

7. show mka policy

具体步骤

	命令或操作	目的
步骤 1	**configure terminal**	进入全局配置模式
步骤 2	**mka policy** *policy name*	标识一个 MKA 策略，并进入 MKA 策略配置模式。策略名称的长度最长为 16 字节
步骤 3	**key-server** *priority*	配置 MKA 密钥服务器选项并设置优先级（0 至 255）。 **注释：** 在用户把密钥服务器的优先级设置为 255 后，对等体就不会成为密钥服务器
步骤 4	**macsec-cipher-suite** *gcm-aes-128*	配置用于生成 128 比特加密 SAK 的加密套件
步骤 5	**confidentiality-offset** *Offset value*	为每个物理接口设置保密（加密）偏移。 **注释：** 偏移值可以是 0、30 或 50。如果用户在客户端上使用 Anyconnect 软件，建议设置偏移 0
步骤 6	**end**	返回特权 EXEC 模式
步骤 7	**show mka policy**	检查用户输入的信息

以下示例中展示了 MKA 策略的配置：

```
Switch(config)# mka policy mka_policy
Switch(config-mka-policy)# key-server priority 200
Switch(config-mka-policy)# macsec-cipher-suite gcm-aes-128
Switch(config-mka-policy)# confidentiality-offset 30
```

```
Switch(config-mka-policy)# end
```

在接口上配置 MACsec

用户可以按照以下步骤在接口上配置 MACsec，一个 MACsec 会话用于语音，另一个用于数据：

总步骤

1. enable
2. configureterminal
3. interface *interface-id*
4. switchport access vlan*vlan-id*
5. switchport mode access
6. macsec
7. authentication event linksec fail action authorize vlan *vlan-id*
8. authentication host-mode multi-domain
9. authentication linksec policy must-secure
10. authentication port-control auto
11. authentication periodic
12. authentication timer reauthenticate
13. authentication violation protect
14. mka policy *policy name*
15. dot1x pae authenticator
16. spanning-tree portfast
17. end
18. show authentication session interface *interface-id*
19. show authentication session interface *interface-id* details
20. show macsec interface *interface-id*
21. show mka sessions
22. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Switch> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Swich# configure terminal`	进入全局配置模式
步骤 3	**interface** *interface-id*	标识 MACsec 接口，并进入接口配置模式。这个接口必须是物理接口
步骤 4	**switchport access vlan***vlan-id*	为端口配置一个 Access VLAN
步骤 5	**switchport mode access**	把接口配置为 Access 端口

步骤 6	**macsec**	在接口上启用 802.1ae MACsec。macsec 命令至在交换机到主机之间的链路（下行端口）上启用 MKA MACsec
步骤 7	**authentication event linksec fail action authorize vlan** *vlan-id*	（可选）通过配置，让交换机在认证尝试失败后，为端口授权一个受限 VLAN，用来处理未授权用户的认证链路安全失败事件
步骤 8	**authentication host-mode multi-domain**	在端口上配置认真管理器模式，允许一台主机和一台语音设备在 802.1x 授权的端口上进行认证。如果没有配置这条命令，默认的主机模式是单主机模式
步骤 9	**authentication linksec policy must-secure**	设置 LinkSec 安全策略，在对等体可用时使用 MACsec 特性来保护会话的安全。如果没有设置这条命令，默认行为是*应该提供安全保护*
步骤 10	**authentication port-control auto**	在端口上启用 802.1x 认证。端口会根据交换机和客户端之间交换的认证结果，来改变授权或未授权状态
步骤 11	**authentication periodic**	为这个端口启用或禁用重认证功能
步骤 12	**authentication timer reauthenticate**	输入 1 至 65535（以秒为单位）之间的值。从服务器获得重认证超时时间。默认的重认证时间是 3600 秒
步骤 13	**authentication violation protect**	配置端口在发生以下事件时丢弃入站 MAC 地址：新设备连接到端口，或端口上已连接了最大数量的设备后又连接了新设备。如果没有配置这条命令，默认行为是关闭（shutdown）端口
步骤 14	**mka policy** *policy name*	在接口上应用现有的 MKA 协议策略，并在接口上启用 MKA。如果用户没有通过全局配置命令 **mka policy** 配置 MKA 策略的话，
步骤 15	**dot1x pae authenticator**	把端口配置为 802.1x 端口访问实体（PAE）认证器
步骤 16	**spanning-tree portfast**	在接口上为其关联的所有 VLAN 都启用生成树 Post Fast 特性。在启用了 Port Fast 特性后，接口会直接从阻塞状态进入转发状态，无需经历生成树中间状态
步骤 17	**end** **示例：** `Switch(config)#end`	返回特权 EXEC 模式
步骤 18	**show authentication session interface**	检查授权的会话安全状态

	interface-id	
步骤 19	**show authentication session interface** *interface-id* **details**	检查授权会话的安全状态详情
步骤 20	**show macsec interface** *interface-id*	检查接口上的 MACsec 状态
步骤 21	**show mka sessions**	检查已建立的 MKA 会话
步骤 22	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

使用 PSK 配置 MACsec MKA

总步骤

1. configure terminal

2. key chain *key-chain-name* **macsec**

3. key *hex-string*

4. cryptographic-algorithm {*gcm-aes-128* | *gcm-aes-256*}

5. key-string { [*0*|*6*|*7*] *pwd-string* | *pwd-string*}

6. lifetime local [*start timestamp* {*hh::mm::ss* | *day* | *month* | *year*}] [**duration** *seconds* | *end timestamp* {*hh::mm::ss* | *day* | *month* | *year*}]

7. end

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal**	进入全局配置模式
步骤 2	**key chain** *key-chain-name* **macsec**	配置一个密钥链，并进入密钥链配置模式
步骤 3	**key** *hex-string*	为密钥链中的每个密钥配置唯一的识别符，并进入密钥链中的密钥配置模式。 **注释：** 对于 128 比特加密，使用 32 位十六进制数字密钥链。对于 256 比特加密，使用 64 位十六进制数字密钥链
步骤 4	**cryptographic-algorithm** {*gcm-aes-128*	*gcm-aes-256*}	使用 128 比特或 256 比特加密方式来设置加密认证算法
步骤 5	**key-string** { [*0*	*6*	*7*] *pwd-string*	*pwd-string*}	为密钥链设置密码。只能输入十六进制字符
步骤 6	**lifetime local** [*start timestamp* {*hh::mm::ss*	*day*	*month*	*year*}] [**duration** *seconds*	*end timestamp* {*hh::mm::ss*	*day*	*month*	*year*}]	设置预共享密钥的生存时间
步骤 7	**end**	返回特权 EXEC 模式

以下展示了一个示例：

```
Switch(config)# Key chain keychain1 macsec
Switch(config-key-chain)# key 1000
Switch(config-keychain-key)# cryptographic-algorithm gcm-aes-128
Switch(config-keychain-key)# key-string
12345678901234567890123456789012
Switch(config-keychain-key)# lifetime local 12:12:00 July 28 2016
12:19:00 July 28 2016
Switch(config-keychain-key)# end
```

在使用 PSK 的接口上配置 MACsec MKA

总步骤

1. configure terminal

2. interface *interface-id*

3. mka policy *policy-name*

4. mka pre-shared-key key-chain *key-chain name*

5. macsec replay-protection window-size *frame number*

6. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal**	进入全局配置模式
步骤 2	**interface** *interface-id*	进入接口配置模式
步骤 3	**mka policy** *policy-name*	配置一个 MKA 策略
步骤 4	**mka pre-shared-key key-chain** *key-chain name*	配置一个 MKA 预共享密钥的密钥链名称。 **注释：** 用户可以在物理接口或子接口上配置 MKA 预共享密钥，但不能同时在接口和子接口上进行配置
步骤 5	**macsec replay-protection window-size** *frame number*	为重放保护设置 MACsec 窗口大小
步骤 6	**end**	返回特权 EXEC 模式

以下展示了一个示例：

```
Switch(config)# interface GigabitEthernet 0/0/0
Switch(config-if)# mka policy mka_policy
Switch(config-if)# mka pre-shared-key key-chain key-chain-name
Switch(config-if)# macsec replay-protection window-size 10
Switch(config-if)# end
```

理解使用 EAP-TLS 的 MACsec MKA

从 Inspur INOS 15.2(5)E 版本开始，Inspur 6850 和 6650 系列交换机能够在交换机到交换机之

间的链路上支持 MACsec MKA。
通过使用 IEEE 802.1X 基于端口的认证和可扩展认证协议（EAP-TLS），用户可以在设备上行链路端口之间配置 MACsec MKA。EAP-TLS 能够实现相互认证，并且获得 MSK（主用会话密钥），MSK 会被 MKA 操作用来派生连接关联密钥（CAK）。EAP-TLS 会携带设备证书，以备 AAA 服务器认证。

使用 EAP-TLS 配置 MACsec MKA 的先决条件

- 用户要确保自己为网络配置了证书授权中心（CA）服务器；
- 生成一个 CA 证书；
- 用户要确保配置了 Inspur 身份服务引擎（ISE）2.0 版本；
- 用户要确保使用网络时间协议（NTP）对参与的设备、CA 服务器和 Inspur 身份服务引擎（ISE）进行同步。如果所有设备上的时间不同步，则无法验证证书；
- 用户要确保在设备上配置了 802.1x 认证和 AAA。

使用 EAP-TLS 配置 MACsec MKA 的限制条件

- Port-Channel 端口上不支持 MKA；
- 高可用性和本地认证功能与 MKA 不兼容。

配置使用 EAP-TLS 的 MACsec MKA

要想在点到点链路上配置 MACsec 和 MKA，用户需要执行以下工作：

- 配置证书注册
  - 生成密钥对
  - 配置 SECP 注册
  - 手动配置证书
- 配置一个认证策略
- 配置 EAP-TLS 配置文件和 IEEE 802.1x 证书
- 在接口上配置使用 EAP-TLS 的 MACsec MKA

生成密钥对

总步骤

1. configure terminal

2. crypto key generate rsa label *label name* **general-keys modulus** *size*

3. end

4. show authentication session interface *interface-id*

5. copy running-config startup-config

具体步骤

	命令或操作	目的

步骤 1	**configure terminal**	进入全局配置模式
步骤 2	**crypto key generate rsa label** *label name* **general-keys modulus** *size*	为信令交互和加密生成 RSA 密钥对。 用户还可以使用 label 关键字为每个密钥对分配一个标签。使用密钥对的信任点会引用这个标签。如果不分配标签的话，密钥对会自动使用标签 <Default-RSA-Key> 如果不使用其他关键字，这条命令会生成一个通用的 RSA 密钥对。如果用户没有指定系数的话，默认的密钥系数为 1024。用户可以使用 modulus 关键字来指定其他系数大小。
步骤 3	**end**	返回特权 EXEC 模式
步骤 4	**show authentication session interface** *interface-id*	检查授权会话的安全状态
步骤 5	**copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

配置使用 SECP 进行注册

简单证书注册协议（SCEP）是 Inspur 开发的注册协议，使用 HTTP 来与证书授权中心（CA）或注册授权中心（RA）进行通信。SCEP 并不常用来发送和接收请求和证书。

总步骤

configure terminal

2. crypto pki trustpoint *server name*

3. enrollment url *url name pem*

4. rsakeypair *label*

5. serial-number none

6. ip-address none

7. revocation-check crl

8. auto-enroll *percent* **regenerate**

9. crypto pki authenticate *name*

10. exit

11. show crypto pki certificate *trustpoint name*

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal**	进入全局配置模式
步骤 2	**crypto pki trustpoint** *server name*	声明信任点并为其指定名称，并且进入 CA 信任点配置模式
步骤 3	**enrollment url** *url name pem*	指定 CA 的 URL，用户的设备应该向这个 CA 发送证书请求。 用户可以在 URL 中使用方括号配置 IPv6 地址。比如

		http://[2001:DB8:1:1::1]:80。 用户需要使用 pem 关键字为证书请求添加隐私增强邮件（PEM）边界
步骤 4	**rsakeypair** *label*	指定与证书相关联的密钥对。 **注释：** **rsakeypair** 名称必须与信任点名称相匹配
步骤 5	**serial-number none**	**none** 关键字指定了不会包含在证书请求中的序列号
步骤 6	**ip-address none**	**none** 关键字指定了证书请求中不应该包含 IP 地址
步骤 7	**revocation-check crl**	把 CRL 作为方法，确保不会撤销对等体的证书
步骤 8	**auto-enroll** *percent* **regenerate**	启用自动注册特性，允许客户端自动向 CA 请求证书。 如果没有启用自动注册的话，客户端必须在证书过期时，手动在 PKI 中重新注册。 默认情况下，只有设备的域名系统（DNS）包含在证书中。 使用 percent 参数指定在当前证书的生命周期到达多少百分比后，要请求新的证书。 使用 regenerate 关键字为证书生成新的密钥，即使已经存在命名密钥。 如果滚动密钥对是可以导出的，则新密钥对也是可以导出的。用户会在信任点的配置中，看到以下注释，指示密钥对是否可以导出：“! RSA key pair associated with trustpoint is exportable.” 出于安全原因，建议生成新的密钥对
步骤 9	**crypto pki authenticate** *name*	检索 CA 证书并对其进行认证
步骤 10	**exit**	退出全局配置模式
步骤 11	**show crypto pki certificate** *trustpoint name*	显示与信任点相关的证书信息

手动配置注册

如果用户的 CA 不支持 SCEP，或者如果路由器和 CA 之间无法实现网络连接，用户可以按照以下步骤进行手动证书注册：

总步骤

1. configure terminal

2. crypto pki trustpoint *server name*

3. enrollment url *url name pem*
4. rsakeypair *label*
5. serial-number none
6. ip-address none
7. revocation-check crl
8. exit
9. crypto pki authenticate *name*
10. crypto pki enroll *name*
11. crypto pki import *name* **certificate**
12. exit
13. show crypto pki certificate *trustpoint name*
14. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal**	进入全局配置模式
步骤 2	**crypto pki trustpoint** *server name*	声明信任点并为其指定名称，并且进入 CA 信任点配置模式
步骤 3	**enrollment url** *url name pem*	指定 CA 的 URL，用户的设备应该向这个 CA 发送证书请求。 用户可以在 URL 中使用方括号配置 IPv6 地址。比如 http://[2001:DB8:1:1::1]:80。 用户需要使用 pem 关键字为证书请求添加隐私增强邮件（PEM）边界
步骤 4	**rsakeypair** *label*	指定与证书相关联的密钥对。
步骤 5	**serial-number none**	**none** 关键字指定了不会包含在证书请求中的序列号
步骤 6	**ip-address none**	**none** 关键字指定了证书请求中不应该包含 IP 地址
步骤 7	**revocation-check crl**	把 CRL 作为方法，确保不会撤销对等体的证书
步骤 8	**exit**	退出全局配置模式
步骤 9	**crypto pki authenticate** *name*	检索 CA 证书并对其进行认证
步骤 10	**crypto pki enroll** *name*	生成证书请求，并显示证书服务器中复制和粘贴的请求。 在看到提示时输入注册信息。举例来说，指定是否要在证书请求中包含设备 FQDN 和 IP 地址。 用户还可以为 Console 终端提供显示和证书请求选项。 显示带有或不带有请求的 PEM 头部的基于 64 编码证书
步骤 11	**crypto pki import** *name* **certificate**	在 Console 终端上通过 HTTP 导入一个证书，它会检索授予的证书。

		设备会尝试通过 TFTP 检索授予的证书，它会使用与发送请求所使用的文件名相同的文件名，但会把扩展名从“.req”更改为“.crt”。对于密钥证书的使用，它会使用扩展名“-sign.crt”和“-encr.crt”。 设备会对接收到的文件进行解析，对证书进行验证，并将证书插入到交换机的内部证书数据库中。 **注释：** 有些 CA 会忽略证书请求中使用的密钥信息，并发出通用目的的证书。如果 CA 忽略证书请求中使用的密钥信息，则只导入通用证书。路由器不会使用生成的两个密钥对中的任意一个
步骤 12	**exit**	退出全局配置模式
步骤 13	**show crypto pki certificate** *trustpoint name*	显示与信任点相关的证书信息
步骤 14	**copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

在接口应用 802.1x MACsec MKA 配置

用户可以按照以下步骤，在接口上应用使用了 EAP-TLS 的 MACsec MKA：

总步骤

1. configure terminal

2. interface *interface-id*

3. macsec network-link

4. authentication periodic

5. authentication timer reauthenticate interval

6. access-session host-mode multi-domain

7. access-session closed

8. access-session port-control auto

9. dot1x pae both

10. dot1x credentials profile

11. dot1x supplicant eap profile *name*

12. service-policy type control subscriber *control-policy name*

13. exit

14. show macsec interface

15. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal**	进入全局配置模式

步骤 2	**interface** *interface-id*	标识 MACsec 接口，并进入接口配置模式。这个接口必须是物理接口
步骤 3	**macsec network-link**	在接口上启用 MACsec
步骤 4	**authentication periodic**	为这个端口启用或禁用重认证功能
步骤 5	**authentication timer reauthenticate interval**	设置重认证时间间隔
步骤 6	**access-session host-mode multi-domain**	允许主机获得接口的访问权限
步骤 7	**access-session closed**	阻止接口上的预认证访问
步骤 8	**access-session port-control auto**	设置接口的授权状态
步骤 9	**dot1x pae both**	把端口配置为 802.1X 端口访问实体（PAE）请求方和认证方
步骤 10	**dot1x credentials profile**	为接口分配 802.1x 证书配置文件
步骤 11	**dot1x supplicant eap profile** *name*	为接口分配 EAP-TLS 配置文件
步骤 12	**service-policy type control subscriber** *control-policy name*	在接口上应用订阅者控制策略
步骤 13	**exit**	返回特权 EXEC 模式
步骤 14	**show macsec interface**	显示接口上的 MACsec 详情
步骤 15	**copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

Inspur TrustSec MACsec 的相关信息

下面这个表格中列出了 TrustSec 特性，这些特性是在 Inspur 交换机上最终实现的 TrustSec 功能。连续通用的 TrustSec 版本扩展了支持的交换机数量，以及每台交换机能够支持的 TrustSec 功能数量。

Inspur TrustSec 特性	**描述**
802.1 AE 标记（MACsec）	基于 IEEE 802.1AE 的有线速率逐跳二层加密协议。 在具有 MACsec 功能的设备之间，发送方设备在发送数据包时对其进行加密，接收方设备在接收到数据包时对其进行解密，在设备内数据包是明文的。 这个特性只能用于具有 TrustSec 硬件功能的设备之间。 **注释：** Inspur 2960x 交换机上不支持该特性
端点准入控制（EAC）	EAC 是当端点用户或设备在连接到 TrustSec 域时，对其进行身份验证过程。通常 EAC 是发生在接入层交换机上的行为。在 EAC 过程中成功的认证和授权后，用户或设备会获得安全组标记（Security Group Tag）。当前的 EAC

	可以是 802.1X、MAC 认证旁路（MAB）和 Web 认证代理（WebAuth）
网络设备准入控制（NDAC）	NDAC 是一个认证过程，其中 TrustSec 域中的每台网络设备都可以验证其对等设备的证书和可信赖性。NDAC 会使用 IEEE 802.1X 基于端口认证的认证框架，并使用 EAP-FAST 作为它的 EAP 方法。在 NDAC 过程中认证和授权成功后，会为 IEEE 802.1AE 加密进行安全关联协议协商。 **注释：** Inspur 2960x 交换机上不支持该特性
安全关联协议（SAP）	在通过了 NDAC 认证后，安全关联协议（SAP）会为 TrustSec 对等体之间接下来的 MACSec 链路加密，自动协商密钥和加密套件。SAP 定义在 IEEE 802.11i 中
安全组标记（SGT）	安全组标记交换协议（SXP）。通过使用 SXP，不具备 TrustSec 硬件功能的设备能够为认证的用户和设备，从 Inspur 身份服务引擎（ISE）或 Inspur 安全访问控制系统（ACS）那里接收 SGT 属性。然后设备可以把源 IP 到 SGT 的绑定信息转发到具有 TrustSec 硬件功能的设备，继而为实施 SGACL 对源流量进行标记

当链路两端的设备都支持 802.1AE MACsec 时，它们会协商 SAP。请求方和认证方之间会进行 EAPOL 密钥交换，用来协商加密套件、交换安全参数和管理密钥。在这些任务的成功完成后，双方会建立安全关联（SA）。

根据用户软件版本和许可，以及链路硬件的支持，SAP 协商可以使用以下操作模式之一：

- Galois Counter Mode（GCM）——认证和加密
- GCM 认证（GMAC）——GCM 认证，无加密
- 无封装——无封装（明文）
- Null——封装，无认证，无加密

配置 Inspur TrustSec MACsec

在手动模式中配置 Inspur TrustSec 交换机到交换机链路安全

在开始前

当用户在接口上手动配置 Inspur TrustSec 特性时，需要考虑以下用法指导和限制条件：

- 如果没有定义 SAP 参数的话，Inspur TrustSec 特性就不会执行封装和加密；
- 如果选择 GCM 作为 SAP 运行模式的话，用户必须从 Inspur 获得 MACsec 加密软件许可。如果选择 GCM 但没有获得许可的话，接口会变为链路失效（link-down）状态；
- 在用户配置 SAP 成对的主密钥（sap pmk）时可以选择以下保护级别：
  - 未配置 SAP——无保护

- **sap mode-list gcm-encrypt gmac no-encap**——期望提供保护，但并不强制
- **sap mode-list gcm-encrypt gmac**——优选保密性和需要提供完整性。保护方式由请求方根据请求方的偏好进行选择
- **sap mode-list gmac**——只提供完整性
- **sap mode-list gcm-encrypt**——需要提供保密性
- **sap mode-list gmac gcm-encrypt**——需要并优选完整性，保密性是可选的

从特权 EXEC 模式开始，用户可以按照以下步骤，在接口上手动为另一台支持 Inspur TrustSec 的设备配置 Inspur TrustSec 特性：

总步骤

1. configure terminal

2. interface *interface-id*

3. cts manual

4. sap pmk *key* [**mode-list** *mode1* [*mode2* [*mode3* [*mode4*]]]]

5. no propagate sgt

6. exit

7. end

8. show cts interface [*interface-id* |**brief** |**summary**]

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Switch# configure terminal`	进入全局配置模式
步骤 2	**interface** *interface-id* **示例：** `Switch(config)# interface tengigabitethernet 1/1/2`	**注释：** 进入接口配置模式
步骤 3	**cts manual** **示例：** `Switch(config-if)# cts manual`	进入 Inspur TrustSec 手动配置模式
步骤 4	**sap pmk** *key* [**mode-list** *mode1* [*mode2* [*mode3* [*mode4*]]]] **示例：** `Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap`	（可选）配置 SAP 成对的主密钥（PMK）和运行模式。在 Inspur TrustSec 手动模式中，SAP 默认是禁用的。 • *key*——十六进制数值，字符长度为偶数，最长为 32 个字符 SAP 运行模式的选项： • **gcm-encrypt**——认证和加密 **注释：** 如果用户的软件许可支持 MACsec 加密的话，使用这个模式来实现 MACsec 认证和加

		密。 • **gmac——**认证，无加密 • **no-encap**——无封装 • **null**——封装，无认证，无加密 **注释：** 如果接口不支持数据链路加密的话，**no-cap** 就是默认设置，也是唯一可用的 SAP 运行模式。不支持 SGT
步骤 5	**no propagate sgt** **示例：** `Switch(config-if-cts-manual)# no propagate sgt`	如果对等体设备无法处理 SGT，用户需要使用这条命令的 **no** 形式。**no propagate sgt** 命令会阻止接口向对等体发送 SGT
步骤 6	**exit** **示例：** `Switch(config-if-cts-manual)# exit`	退出 Inspur TrustSec 802.1x 接口配置模式
步骤 7	**end** **示例：** `Device(config-if)# end`	返回特权 EXEC 模式
步骤 8	**show cts interface** [*interface-id* **|brief |summary**]	（可选）通过查看与 TrustSec 相关的接口特征，检查用户输入的配置

以下示例展示了如何在接口上，以手动模式配置 Inspur TrustSec 认证特性：

```
Switch# configure terminal
Switch(config)# interface tengigabitethernet 1/1/2
Switch(config-if)# cts manual
Switch(config-if-cts-manual)# sap pmk 1234abcdef mode-list gcm-encrypt null no-encap
Switch(config-if-cts-manual)# no propagate sgt
Switch(config-if-cts-manual)# exit
Switch(config-if)# end
```

配置示例

在接口上配置 MACsec

用户可以按照以下步骤在接口上配置 MACsec，一个 MACsec 会话用于语音，另一个用于数据：

总步骤

1. enable

2. configureterminal
3. interface *interface-id*
4. switchport access vlan*vlan-id*
5. switchport mode access
6. macsec
7. authentication event linksec fail action authorize vlan *vlan-id*
8. authentication host-mode multi-domain
9. authentication linksec policy must-secure
10. authentication port-control auto
11. authentication periodic
12. authentication timer reauthenticate
13. authentication violation protect
14. mka policy *policy name*
15. dot1x pae authenticator
16. spanning-tree portfast
17. end
18. show authentication session interface *interface-id*
19. show authentication session interface *interface-id* details
20. show macsec interface *interface-id*
21. show mka sessions
22. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Switch> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Swich# configure terminal`	进入全局配置模式
步骤 3	**interface** *interface-id*	标识 MACsec 接口，并进入接口配置模式。这个接口必须是物理接口
步骤 4	**switchport access vlan***vlan-id*	为端口配置一个 Access VLAN
步骤 5	**switchport mode access**	把接口配置为 Access 端口
步骤 6	**macsec**	在接口上启用 802.1ae MACsec。macsec 命令至在交换机到主机之间的链路（下行端口）上启用 MKA MACsec
步骤 7	**authentication event linksec fail action authorize vlan** *vlan-id*	（可选）通过配置，让交换机在认证尝试失败后，为端口授权一个受限 VLAN，用来处理未授权用户的认证链路安全失败事件
步骤 8	**authentication host-mode multi-domain**	在端口上配置认真管理器模式，允许一台主机和一台语音设备在 802.1x 授

		权的端口上进行认证。如果没有配置这条命令，默认的主机模式是单主机模式
步骤 9	**authentication linksec policy must-secure**	设置 LinkSec 安全策略，在对等体可用时使用 MACsec 特性来保护会话的安全。如果没有设置这条命令，默认行为是*应该提供安全保护*
步骤 10	**authentication port-control auto**	在端口上启用 802.1x 认证。端口会根据交换机和客户端之间交换的认证结果，来改变授权或未授权状态
步骤 11	**authentication periodic**	为这个端口启用或禁用重认证功能
步骤 12	**authentication timer reauthenticate**	输入 1 至 65535（以秒为单位）之间的值。从服务器获得重认证超时时间。默认的重认证时间是 3600 秒
步骤 13	**authentication violation protect**	配置端口在发生以下事件时丢弃入站 MAC 地址：新设备连接到端口，或端口上已连接了最大数量的设备后又连接了新设备。如果没有配置这条命令，默认行为是关闭（shutdown）端口
步骤 14	**mka policy** *policy name*	在接口上应用现有的 MKA 协议策略，并在接口上启用 MKA。如果用户没有通过全局配置命令 **mka policy** 配置 MKA 策略的话，
步骤 15	**dot1x pae authenticator**	把端口配置为 802.1x 端口访问实体（PAE）认证器
步骤 16	**spanning-tree portfast**	在接口上为其关联的所有 VLAN 都启用生成树 Post Fast 特性。在启用了 Port Fast 特性后，接口会直接从阻塞状态进入转发状态，无需经历生成树中间状态
步骤 17	**end** **示例：** `Switch(config)#end`	返回特权 EXEC 模式
步骤 18	**show authentication session interface** *interface-id*	检查授权的会话安全状态
步骤 19	**show authentication session interface** *interface-id* **details**	检查授权会话的安全状态详情
步骤 20	**show macsec interface** *interface-id*	检查接口上的 MACsec 状态
步骤 21	**show mka sessions**	检查已建立的 MKA 会话
步骤 22	**copy running-config startup-config** **示例：** `Device# copy running-config`	（可选）把输入的命令保存到配置文件中

	startup-config	

Inspur TrustSec 交换机到交换机链路安全配置示例

这个示例展示了为实现 Inspur TrustSec 交换机到交换机安全性，种子设备和非种子设备上所必需的配置。用户必须为链路安全性配置 AAA 和 RADIUS。在这个示例中，ACS-1 到 ACS-3 可以是任何名称的服务器，cts-radius 是 Inspur TrustSec 服务器。

种子设备上的配置：

```
Switch(config)#aaa new-model
Switch(config)#radius server ACS-1
Switch(config-radius-server)#address ipv4 10.5.120.12 auth-port 1812 acct-port 1813
Switch(config-radius-server)#pac key inspur123 Switch(config-radius-server)#exit
Switch(config)#radius server ACS-2
Switch(config-radius-server)#address ipv4 10.5.120.14 auth-port 1812 acct-port 1813
Switch(config-radius-server)#pac key inspur123 Switch(config-radius-server)#exit
Switch(config)#radius server ACS-3
Switch(config-radius-server)#address ipv4 10.5.120.15 auth-port 1812 acct-port 1813
Switch(config-radius-server)#pac key inspur123
Switch(config-radius-server)#exit
Switch(config)#aaa group server radius cts-radius
Switch(config-sg-radius)#server name ACS-1
Switch(config-sg-radius)#server name ACS-2
Switch(config-sg-radius)#server name ACS-3
Switch(config-sg-radius)#exit
Switch(config)#aaa authentication login default none
Switch(config)#aaa authentication dot1x default group cts-radius
Switch(config)#aaa authorization network cts-radius group cts-radius
Switch(config)#aaa session-id common
Switch(config)#cts authorization list cts-radius
Switch(config)#dot1x system-auth-control
Switch(config)#interface gi1/1/2
Switch(config-if)#switchport mode trunk
Switch(config-if)#cts dot1x
Switch(config-if-cts-dot1x)#sap mode-list gcm-encrypt gmac
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit
Switch(config)#interface gi1/1/4
Switch(config-if)#switchport mode trunk
```

```
Switch(config-if)#cts manual
Switch(config-if-cts-dot1x)#sap pmk 033445AABBCCDDEEFF mode-list
gcm-encrypt gmac
Switch(config-if-cts-dot1x)#no propagate sgt
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit
Switch(config)#radius-server vsa send authentication
Switch(config)#end
Switch#cts credentials id cts-36 password trustsec123
```

非种子设备上的配置：

```
Switch(config)#aaa new-model
Switch(config)#aaa session-id common
Switch(config)#dot1x system-auth-control
Switch(config)#interface gi1/1/2
Switch(config-if)#switchport mode trunk
Switch(config-if)#shutdown
Switch(config-if)#cts dot1x
Switch(config-if-cts-dot1x)#sap mode-list gcm-encrypt gmac
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit
Switch(config)#interface gi1/1/4
Switch(config-if)#switchport mode trunk
Switch(config-if)#shutdown
Switch(config-if)#cts manual
Switch(config-if-cts-dot1x)#sap pmk 033445AABBCCDDEEFF mode-list
gcm-encrypt gmac
Switch(config-if-cts-dot1x)#no propagate sgt
Switch(config-if-cts-dot1x)#exit
Switch(config-if)#exit
Switch(config)#radius-server vsa send authentication
Switch(config)#end
Switch(config)#cts credentials id cts-72 password trustsec123
```

配置 RADIUS

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置 RADIUS 的先决条件

这一部分列出了使用 RADIUS 来控制设备访问行为的先决条件：
通常情况下：

- 要想使用本章中提到的任何配置命令，用户必须启用 RADIUS 和认证、授权和审计（AAA）；
- RADIUS 是通过 AAA 实现的，并且只能通过 AAA 命令进行启用；
- 用户需要使用全局配置命令 **aaa new-model** 来启用 AAA；
- 用户需要使用全局配置命令 **aaa authentication** 来为 RADIUS 认证定义方法列表；
- 用户需要使用 **line** 和 **interface** 命令，来启用定义好的方法列表；
- 最起码用户必须对运行 RADIUS 服务器软件的一台或多台主机进行标识，并定义 RADIUS 认证的方法列表。用户可以（可选的）为 RADIUS 授权和审计定义方法列表；
- 在用户的设备上配置 RADIUS 特性之前，用户应该能够访问 RADIUS 服务器，并且应该对其进行配置；
- RADIUS 主机通常是运行 RADIUS 服务器软件的多用户系统，这种软件可能会来自于 Inspur（Inspur 安全访问控制服务器 3.0 版本）、Livingston、Merit、Microsoft，或其他软件厂商。更多信息用户可以参考 RADIUS 服务器文档；
- 要想使用授权变更（CoA）接口，交换机上必须已经存在一个会话。CoA 可以用来标识一个会话，并执行断开连接请求。更新只会对指定会话带来影响；
- 建议在交换机堆栈和 RADIUS 服务器之间建立冗余的连接。这样做有助于在连接 RADIUS 服务器的堆栈成员从交换机堆栈中移除后，保障 RADIUS 服务器仍可访问。

对于 RADIUS 的运行：

- 用户必须首先成功完成 RADIUS 认证，才能继续使用 RADIUS 授权（如果启用了的话）。

配置 RADIUS 的限制条件

这个主题中包含了使用 RADIUS 来控制设备访问行为的限制条件：
通常情况下：

- 为了防止安全性中断，用户不能通过网络管理应用程序来配置 RADIUS。

RADIUS 不适用于下列网络安全环境：

- 多协议访问环境。RADIUS 不支持 AppleTalk 远程访问（ARA）、NetBINOS 数据帧控制协议（NBFCP）、NetWare 异步服务接口（NASI）或 X.25 PAD 连接；
- 交换机到交换机或路由器到路由器的环境。RADIUS 不提供双向认证。如果非 Inspur 设备请求进行身份验证，RADIUS 可以提供从 Inspur 设备到非 Inspur 设备的身份验证；
- 使用各种服务的网络。RADIUS 通常会把用户绑定到一个服务模型。

RADIUS 的相关信息

RADIUS 和交换机访问

这一部分介绍了如何启用和配置 RADIUS。RADIUS 为认证和授权过程，提供了详细的审计信息和灵活的管理控制。

RADIUS 概述

RADIUS 是一种分布式客户端/服务器系统，它可以保护网络免遭未经授权的访问。RADIUS 客户端运行在支持的 Inspur 路由器和交换机上。客户端会向中心 RADIUS 服务器发送认证请求，其中包含所有用户认证和网络服务访问的相关信息。

用户可以在以下这些需要访问安全性的网络环境中使用 RADIUS：

- 部署了多厂商访问服务器的网络，其中所有服务器都支持 RADIUS。举例来说，来自不同厂商的访问服务器可以使用单个 RADIUS 服务器的安全数据库。在部署了多厂商接入服务器的 IP 网络中，使用 RADIUS 服务器对拨入用户进行认证，并且这台 RADIUS 服务器已经通过自定义，与 Kerberos 安全系统一起工作；
- 一站式网络安全环境，其中的应用支持 RADIUS 协议，比如使用 *智能卡*访问控制系统的接入环境。在一种情况下，用户可以把 RADIUS 与 Enigma 安全卡一起使用，来验证用户并授予其对网络资源的访问权限；
- 已经在使用 RADIUS 的网络。用户可以将包含 RADIUS 客户端的 Inspur 设备添加到网络中。这可能是用户切换为使用 TACACS+服务器的第一步，用户可以参考下文中的图 2：从 RADIUS 切换到 TACACS+服务；
- 限制用户只能访问单个服务的网络。使用 RADIUS 可以控制让用户只访问单台主机、只访问单个应用（比如 Telnet），或者通过诸如 IEEE 802.1x 之类的协议来访问网络。有关这个协议的详细信息，用户可以参考第 11 章“配置 IEEE 802.1x 基于端口的认证”；
- 需要进行资源审计的网络。用户可以在 RADIUS 身份认证或授权之外，独立使用 RADIUS 审计功能。RADIUS 审计功能允许在服务的开始和结束时发送数据，并显示会话期间使用的资源（例如时间、数据包、字节等）。Internet 服务提供商可能会使用免费软件版本的 RADIUS 访问控制和审计软件，来满足特殊的安全性和审计需求。

图 101：从 RADIUS 切换到 TACACS+服务

RADIUS server（共 2 处）	RADIUS 服务器
TACACS+ server（共 2 处）	TACACS+ 服务器

Remote PC	远端 PC
Workstation	工作站

RADIUS 的工作原理

当用户尝试登录一台设备并进行认证时，设备如果是由 RADIUS 服务器提供访问控制的，就会发生以下事件：

1. 用户会看到输入用户名和密码的提示；
2. 通过网络把用户名和加密密码发送到 RADIUS 服务器的；
3. 用户从 RADIUS 服务器接收到以下响应之一：
   - ACCEPT（接受）——用户通过认证
   - REJECT（拒绝）——用户没有通过认证，并看到再次输入用户名和密码的提示，或者访问被拒绝
   - CHALLENGE（质询）——从用户质询请求其他数据
   - CHALLENGE PASSWORD——这个响应是要求用户选择一个新密码

ACCEPT（接受）或 REJECT（拒绝）响应是与用来进行特权 EXEC 或网络授权的其他数据捆绑在一起的。ACCEPT（接受）或 REJECT（拒绝）数据包中包含以下其他数据：

- Telnet、SSH、远程登录，或者特权 EXEC 服务
- 连接参数，其中包括主机或客户端 IP 地址、访问列表，以及用户超时时间

RADIUS 授权变更

RADIUS 授权变更（CoA）提供了一种机制，能够在认证成功后更改认证、授权和审计（AAA）的属性。在 AAA 中为用户或用户组变更策略时，管理员会从 AAA 服务器发送 RADIUS CoA 数据包，以此来初始化认证并执行信息的策略，Inspur 安全访问控制服务器（ACS）就可以充当 AAA 服务器。这一部分概述了 RADIUS 接口，其中包括可用的原函数，以及如何在 CoA 中使用它们。

- 授权变更请求
- CoA 请求响应代码
- CoA 请求命令
- 会话重认证
- 有关会话终结的堆叠指导

标准的 RADIUS 接口通常用于拉取模型中，在这种模型中，请求源自于联网设备，响应来自于查询服务器。Inspur 支持 RFC 5176 中定义的 RADIUS CoA 扩展，它通常用于推送模型中，允许从外部 AAA 服务器或策略服务器进行动态的会话重配置。

交换机支持以下每会话 CoA 请求：

- 会话重认证
- 会话终结
- 会话终结及端口关闭（shutdown）
- 会话终结及端口反弹（bounce）

这个特性集成在 Inspur 安全访问控制服务器（ACS）5.1 中。

RADIUS 接口在 Inspur 交换机上是默认启用的。但用户仍需为以下属性执行基本配置：

- 安全和密码——参考这个指南中“在交换机上预防未授权访问”一节
- 审计——参考这个指南中配置基于交换机的认证一章中“开始使用 RADIUS 审计”一节

Inspur INOS 软件支持 RFC 5176 中定义的 RADIUS CoA 扩展，它通常用于推送模型中，允许从外部 AAA 服务器或策略服务器进行动态的会话重配置。它为每个会话支持 CoA 请求，能够实现会话标识、会话终结、主机重认证、端口关闭和端口反弹。这个模型中包含一个请求（CoA 请求）和两个可能用到的响应代码：

- CoA 确认（ACK）[CoA-ACK]
- CoA 未确认（NAK）[CoA-NAK]

请求是从 CoA 客户端（通常是 AAA 服务器或策略服务器）发起的，并发送给充当侦听方的设备。

下面这个表格中展示了基于身份识别的网络服务所支持的 RADIUS CoA 命令和厂商指定的属性（VSA）。所有 CoA 命令中都必须包含设备和 CoA 客户端之间的会话标识。

表 126：基于身份识别的网络服务所支持的 RADIUS CoA 命令

CoA 命令	**Inspur VSA**
激活服务	Inspur:Avpair=“subscriber:command=activate-service” Inspur:Avpair=“subscriber:service-name=<*service-name*>” Inspur:Avpair=“subscriber:precedence=<*precedence-number*>” Inspur:Avpair=“subscriber:activation-mode=replace-all”
停用服务	Inspur:Avpair=“subscriber:command=deactivate-service” Inspur:Avpair=“subscriber:service-name=<service-name>”
反弹主机端口	Inspur:Avpair=“subscriber:command=bounce-host-port”
禁用主机端口	Inspur:Avpair=“subscriber:command=disable-host-port”
会话查询	Inspur:Avpair=“subscriber:command=session-query”
会话重认证	Inspur:Avpair=“subscriber:command=reauthenticate” Inspur:Avpair=“subscriber:reauthenticate-type=last”或者 Inspur:Avpair=“subscriber:reauthenticate-type=rerun”
会话终结	这是标准的断开连接请求，并不需要 VSA
接口模版	Inspur:AVpair="interface-template-name=<*interfacetemplate*>"

授权变更请求

授权变更（CoA）请求定义在 RFC 5176 文档中，它通常用于推送模型中，实现了会话身份识别、主机重认证和会话终结。这个模型中包含一个请求（CoA 请求）和两个可能用到的响应代码：

- CoA 确认（ACK）[CoA-ACK]
- CoA 未确认（NAK）[CoA-NAK]

请求是从 CoA 客户端（通常是 AAA 服务器或策略服务器）发起的，并发送给充当侦听方的设备。

RFC 5176 规范

连接断开请求消息也称为数据包断开连接（POD），交换机支持使用它来完成会话终结工作。

下面这个表格中展示了这个特性中支持的 IETF 属性。

表 127：支持的 IETF 属性

属性编号	**属性名称**

24	状态
31	Calling-Station-Id
44	Acct-Session-ID
80	消息认证器
101	错误原因

下面这个表格中展示了可能出现的错误原因属性值。

表 128：错误原因值

值	解释
201	残留的会话上下文已删除
202	无效的 EAP 数据包（已忽略）
401	不支持的属性
402	丢失的属性
403	NAS 身份不匹配
404	无效的请求
405	不支持的服务
406	不支持的扩展
407	无效的属性值
501	行政上禁止
502	请求无法路由（代理）
503	未找到会话上下文
504	未删除会话上下文
505	其他代理处理错误
506	资源不可用
507	请求已初始
508	不支持多会话选择

CoA 请求响应代码

CoA 请求响应代码可以用来向交换机传达一个命令。

CoA 请求响应代码使用的数据包格式定义在 RFC 5176 文档中，由以下字段构成：代码、识别符、长度、认证器和属性，使用的是类型:长度:值（TLV）格式。属性字段用来承载 Inspur 厂商定义的属性（VSA）。

会话身份识别

为了针对某个会话断开连接和发送 CoA 请求，交换机会根据以下属性之一来定位这个会话：

- Acct-Session-Id（IETF 属性 44）
- Audit-Session-Id（Inspur VSA）
- Calling-Station-Id（IETF 属性 31，其中包含主机 MAC 地址）
- IPv6 属性，可以是以下属性之一：
  - Framed-IPv6-Prefix（IETF 属性 97）和 Framed-Interface-Id（IETF 属性 96），它们加载一起构成了完整的 IPv6 地址，定义在 RFC 3162 文档中
  - Framed-IPv6-Address
- Plain IP Address（IETF 属性 8）

除非 CoA 消息中的所有会话标识属性都与会话相匹配，否则交换机会返回携带“无效属性

值”错误代码属性的 Disconnect-NAK 或 CoA-NAK。
如果消息中包含多个会话标识属性，则所有属性必须都与会话相匹配，否则交换机会返回携带错误代码“无效属性值”的断开否定确认（NAK）或 CoA-NAK。
RFC 5176 中定义了 CoA 请求代码的数据包格式，其中包含以下字段：代码、身份标识、长度、认证器、以及属性，并使用类型:长度:值（TLV）格式。

Code	代码
Identifier	识别符
Length	长度
Authenticator	认证器
Attributes	属性

属性字段用来承载 Inspur 厂商指定的属性（VSA）。
对于针对特定策略的 CoA 请求，如果消息中包含上述任意会话标识属性，设备就会返回携带错误代码“无效属性值”的 CoA-NAK。
CoA ACK 响应代码
如果授权状态成功地改变了，则发送肯定确认（ACK）。在 CoA ACK 中返回的属性，根据 CoA 请求的变化而变化，这些内容会单独在 CoA 命令中进行讨论。
CoA NAK 响应代码
否定确认（NAK）标识无法改变授权状态，并且其中可以包含指示出失败原因的属性。用户可以使用 **show** 命令验证 CoA 的成功。

CoA 请求命令

表 129：交换机上支持的 CoA 命令

命令 [8]	**Inspur VSA**
重认证主机	Inspur:Avpair=“subscriber:command=reauthenticate”
终结会话	这是标准的断开连接请求，并不需要 VSA
反弹主机端口	Inspur:Avpair=“subscriber:command=bounce-host-port”
禁用主机端口	Inspur:Avpair=“subscriber:command=disable-host-port”

[8] 所有 CoA 命令中都必须包含交换机和 CoA 客户端之间的会话识别符。
会话重认证
当未知身份的主机加入网络，并且与受限的访问授权配置文件（例如访客 VLAN）相关联时，AAA 服务器通常生成一个会话重认证请求。当证书已知时，重新认证请求能够把主机放置在适当的授权组中。
为了初始化会话认证，AAA 服务器会发送一个标准 CoA 请求消息，其中会包含一个以下形式的 Inspur VSA：*Inspur:Avpair =“subscriber:command = reauthenticate”*，以及一个或多个会话标识属性。
当前会话的状态决定了交换机对消息的响应。如果会话当前通过了 IEEE 802.1x 认证，则交换机会通过向服务器发送一个 EAPoL（基于局域网的可扩展认证协议）请求 ID 消息来进行响应。
如果会话当前是通过 MAC 认证旁路（MAB）进行认证的，则交换机会向服务器发送访问请

求，向其传递在初始认证成功时使用的相同身份属性。
如果交换机接收到该命令时会话认证仍在进行中，则交换机会终止认证过程，并重新开始认证序列，从配置中第一个尝试的方法重新开始。
如果会话尚未进行授权，或者被授权为访客 VLAN、重要 VLAN，或类似的策略，则重认证消息会重新开始执行访问控制方法，从配置中第一个尝试的方法重新开始。交换机会维持会话的当前授权状态，直到重新认证得出了不同的授权结果。
交换机堆栈中的会话重认证
当交换机堆栈接收到一个会话重认证消息后：

- 它会在返回确认（ACK）之前，检查是否需要重认证；
- 它会为适当的会话初始化重认证进程；
- 如果认证的结果是成功或失败，则触发重新认证的信号会从堆叠成员上移除；
- 如果堆叠主用设备在认证完成之前失效了，则在主用设备切换之后，会根据原始命令（随后会被移除）初始化重认证进程；
- 如果堆叠主用设备在发送 ACK 之前失效了，则新的主用设备会把它作为新命令进行重传。

会话终结
有三种类型的 CoA 请求可以触发会话终结进程。CoA Disconnect-Request 会终止会话，并且不禁用主机端口。这个命令会为指定主机重新初始化身份认证器状态机，但不会对主机访问网络进行限制。
要想限制主机对网络的访问，要使用一个 CoA 请求，以及 Inspur:Avpair="subscriber:command=disable-host-port" VSA。当已知主机在网络上引发问题，且用户需要立即阻止主机访问网络时，这个命令就很有用。用户要想恢复端口的网络访问功能，要使用非 RADIUS 机制重新启用端口。
当设备不是请求方（比如打印机）时，若需要获取新的 IP 地址（比如在 VLAN 发生变更后），用户要使用端口反弹特性来终止主机端口上的会话（暂时禁用，然后再重新启用端口）。
CoA 连接断开请求
这个命令是一个标准的 Disconnect-Request（断开连接请求）。如果无法定位会话的话，交换机会返回携带“未找到会话上下文”错误代码属性的 Disconnect-NAK 消息。如果定位了会话，交换机就会终止会话。在完全删除会话后，交换机会返回 Disconnect-ACK。
如果交换机在向客户端返回 Disconnect-ACK 之前，因故障切换到备用交换机，则当客户端重新发送请求时，在新的活跃交换机上会重复同样的过程。如果在重新发送之后并没有找到会话，则交换机会发送携带“未找到会话上下文”错误代码属性的 Disconnect-ACK。
CoA 请求：禁用主机端口
RADIUS 服务器发送的 CoA 禁用端口命令，能够把已经建立了会话的认证端口变为管理关闭状态，并导致会话被终止。当已知主机在网络上引发问题，且用户需要立即阻止主机访问网络时，这个命令就很有用。用户要想恢复端口的网络访问功能，要使用非 RADIUS 机制重新启用端口。这个命令承载在一个标准 CoA 请求消息中，其中包含以下这个新的厂商指定属性（VSA）：
Inspur:Avpair="subscriber:command=disable-host-port"
因为这个命令是面向会话的，所以它必须与“会话标识”部分中描述的一个或多个会话标识属性结合在一起使用。如果无法定位会话的话，交换机会返回携带“未找到会话上下文”错误代码属性的 CoA-NAK 消息。如果定位了会话，交换机就会禁用主机端口并返回 CoA-ACK 消息。
如果交换机在向客户端返回 CoA-ACK 之前发生故障，则当客户端重新发送请求时，在新的活

跃交换机上会重复同样的过程。如果交换机在向客户端返回 CoA-ACK 之后，但整个过程还未结束之前发生故障，则新的活跃交换机上会重新开始同样的过程。

注释： 在命令重新发送后的断开连接请求失败事件，可能是由切换前成功的会话终结事件导致的（如果未发送 Disconnect-ACK），或者是由其他方式（比如链路故障）实现的会话终结事件导致的；这些事件发生在发出原始命令之后，且在备用交换机变为活跃状态之前。

CoA 请求：反弹端口

从 RADIUS 服务器发送的 RADIUS 服务器 CoA 反弹端口消息，可能会导致认证端口上的链路发生翻动，从而触发连接到此端口的一个或多个主机重新进行 DHCP 协商。当有 VLAN 发生变更，且端点设备（如打印机）无法检测此认证端口上的变更时，就可能会发生这个事件。CoA 反弹端口承载在标准 CoA 请求消息中，其中包含以下 VSA：

Inspur:Avpair="subscriber:command=bounce-host-port"

因为这个命令是面向会话的，所以它必须与一个或多个会话标识属性结合在一起使用。如果无法定位会话的话，交换机会返回携带“未找到会话上下文”错误代码属性的 CoA-NAK 消息。如果定位了会话，交换机就会禁用主机端口 10 秒钟的时间，之后重新启用它（端口反弹）并返回 CoA-ACK 消息。

如果交换机在向客户端返回 CoA-ACK 之前发生故障，则当客户端重新发送请求时，在新的活跃交换机上会重复同样的过程。如果交换机在向客户端返回 CoA-ACK 之后，但整个过程还未结束之前发生故障，则新的活跃交换机上会重新开始同样的过程。

有关会话终结的堆栈指导

不需要对交换机堆栈中的 CoA Disconnect-Request 消息执行特殊处理。

有关 CoA 请求反弹端口的堆栈指导

因为 **bounce-port** 命令针对的是会话而不是端口，所以如果找不到会话的话，就无法执行这个命令。

当堆叠主用设备上的 Auth Manager（认证管理器）命令处理程序，接收到一个有效的 **bounce-port** 命令时，它会在返回 CoA-ACK 消息之前检查以下信息：

- 是否需要端口反弹
- 端口 ID（在本地会话上下文中查找）

交换机初始化端口反弹行为（禁用端口 10 秒钟，然后重新启用它）。

如果端口反弹操作成功，触发端口反弹的信号就会从备用堆叠中的主用设备中删除。

如果在端口反弹完成之前堆栈主用设备失效了，则在堆栈主用设备切换后，会根据原始命令（其随后被移除）初始化端口反弹。

如果在发送 CoA-ACK 消息之前堆叠主用设备失效了，则新的主用设备会把它作为新命令进行重传。

有关 CoA 请求禁用端口的堆栈指导

因为 **disable-port** 命令针对的是会话而不是端口，所以如果找不到会话的话，就无法执行这个命令。

当堆叠主用设备上的 Auth Manager（认证管理器）命令处理程序，接收到一个有效的 **disable-port** 命令时，它会在返回 CoA-ACK 消息之前检查以下信息：

- 是否需要端口禁用
- 端口 ID（在本地会话上下文中查找）

交换机尝试禁用端口。

如果端口禁用操作成功，触发端口禁用的信号就会从备用堆叠中的主用设备中删除。

如果在端口禁用完成之前堆栈主用设备失效了，则在堆栈主用设备切换后，会根据原始命令（其随后被移除）初始化端口禁用。
如果在发送 CoA-ACK 消息之前堆叠主用设备失效了，则新的主用设备会把它作为新命令进行重传。

默认的 RADIUS 配置

RADIUS 和 AAA 默认都是禁用的。
为了防止安全性失效，用户不能通过网络管理应用程序来配置 RADIUS。当启用了 RADIUS 时，RADIUS 可以验证通过 CLI 访问交换机的用户。

RADIUS 服务器主机

交换机到 RADIUS 服务器的通信中涉及多个组成部分：

- 主机或 IP 地址
- 认证目的端口
- 审计目的端口
- 密钥字符串
- 超时周期
- 重传值

用户可以通过主机名或 IP 地址、主机名和特定 UDP 端口号，或 IP 地址和特定 UDP 端口号来标识 RADIUS 安全服务器。IP 地址和 UDP 端口号的组合会创建出唯一的标识符，使不同的端口能够被单独定义为提供特定 AAA 服务的 RADIUS 主机。通过使用这个唯一的标识符，使 RADIUS 请求能够发送到服务器（IP 地址相同）上的多个 UDP 端口。
如果用户把同一台 RADIUS 服务器上的两个不同主机条目配置为相同的服务（比如审计），则用户配置的第二个主机条目会充当第一个主机条目的故障切换备份设备。举例来说，如果第一主机条目无法提供记帐服务了，则会显示％RADIUS-4-RADIUS_DEAD 消息，然后交换机会尝试使用同一台设备上提供审计服务的第二主机条目（按照配置的顺序尝试 RADIUS 上的主机条目）。
RADIUS 服务器和交换机会使用共享秘密文本字符串来加密密码和交换的响应消息。要想配置 RADIUS 来使用 AAA 安全命令，用户必须指定运行 RADIUS 服务器守护程序的主机，以及与交换机共享的秘密文本（密钥）字符串。
用户可以为所有 RADIUS 服务器在全局配置超时、重传和加密密钥值，可以以每个服务器为基础进行配置，也可以使用全局设置和服务器设置的不同组合。

RADIUS 登录认证

要想配置 AAA 认证，用户需要定义一个命名的认证方法列表，并把它应用给各种端口。方法列表中定义了要执行的认证的类型，以及执行它们的顺序；在它能够执行任何定义的身份验证方法之前，用户必须把它应用在特定的端口上。唯一的例外是默认方法列表。默认方法列表会自动应用在所有端口上，除了已经明确定义了方法列表名称的端口。

方法列表定义了为用户进行认证、授权或审计的序列和方法。用户可以使用方法列表来指定要使用的一个或多个安全协议，这样做可以在初始方法失败时，确保有一个备份系统。软件使用列表中的第一个方法来对用户进行认证、授权或审计；如果该方法没有获得响应，软件会选择列表中的下一个方法。这个过程会持续直到使用列出的方法实现成功通信，或者持续到方法列表耗尽。如果在这个周期中的任何时刻认证失败了——意味着安全服务器或本地用户名数据库发出了响应，拒绝了用户的访问——这时认证过程就停止了，并且不会再尝试其他认证方法。

AAA 服务器组

用户可以配置交换机来使用单台 AAA 服务器或 AAA 服务器组，为现有的服务器主机进行身份认证。用户可以有选择地把一部分服务器主机设置为一组服务器，并将其用于提供特定服务。服务器组与全局服务器主机列表一起使用，还包含所选服务器主机的 IP 地址列表。
如果每个条目都拥有一个唯一的标识符（IP 地址和 UDP 端口号的组合），服务器组中还可以为同一台服务器包含多个主机条目，并且能够把不同端口单独定义为提供特定 AAA 服务的 RADIUS 主机。通过使用这个唯一的标识符，使 RADIUS 请求能够发送到服务器（IP 地址相同）上的多个 UDP 端口。如果用户把同一台 RADIUS 服务器上的两个不同主机条目配置为相同的服务（比如审计），则用户配置的第二个主机条目会充当第一个主机条目的故障切换备份设备。举例来说，如果第一主机条目无法提供记帐服务了，则交换机会尝试使用同一台设备上提供审计服务的第二主机条目（按照配置的顺序尝试 RADIUS 上的主机条目）。

AAA 授权

AAA 授权能够限制用户可以使用的服务。当用户启用了 AAA 授权后，交换机会使用从用户配置文件中检索的信息来配置用户的会话，这个配置文件位于本地用户数据库中，或位于安全服务器。只有当用户配置文件中的信息允许时，用户才能够访问所请求的服务。

RADIUS 审计

AAA 审计功能会跟踪用户正在访问的服务，以及用户消耗的网络资源总量。当用户启用了 AAA 审计后，交换机以审计记录的形式向 RADIUS 安全服务器报告用户的活动。每个审计记录中都包含了审计属性值（AV）对，并且这些信息存储在安全服务器上。之后用户可以使用这些数据来对网络管理、客户端计费或审计进行分析。

厂商指定的 RADIUS 属性

Internet 工程任务组（IETF）草案标准中指定了一种通过使用厂商特定的属性（属性 26），在交换机和 RADIUS 服务器之间传送厂商特定信息的方法。厂商特定属性（VSA）能够使厂商支持自己的扩展属性，但不适合一般使用。Inspur RADIUS 通过使用规范中推荐的格式，来支持厂商特定的选项。Inspur 的厂商 ID 为 9，支持的选项具有供应商类型 1，名为 *inspur-avpair*。这个值是使用以下格式的字符串：
protocol : attribute sep value *

protocol 是用于特定授权类型的 Inspur 协议属性。*attribute* 和 *value* 是定义在 Inspur TACACS+规范中的适当属性值（AV）对，*sep* 是=，用来强制执行属性，*是可选属性。TACACS+授权中可用的完整特性集，都可用于 RADIUS 中。

举例来说，下面这个 AV 对会在 IP 授权期间（在 PPP 的 Internet 协议控制协议（IPCP）地址分配期间）激活 Inspur 的“多命名 IP 地址池”功能：

inspur-avpair= ”ip:addr-pool=first“

如果用户插入一个“*”，AV 对“ip:addr-pool=first”就会成为选项。需要注意的是，任何 AV 对都可以成为选项：

inspur-avpair= ”ip:addr-pool*first“

下面这个示例展示了如何让通过网络访问服务器登录用户，直接获得访问 EXEC 命令的权限：

inspur-avpair= ”shell:priv-lvl=15“

其他厂商也有它们各自唯一的厂商 ID、选项，以及相关联的 VSA。更多厂商 ID 和 VSA 的相关信息，用户可以参考 RFC 2138“Remote Authentication Dial-In User Service (RADIUS)”。

属性 26 中包含以下三个元素：

- 类型
- 长度
- 字符串（也称为数据）
  - Vendor-Id
  - Vendor-Type
  - Vendor-Length
  - Vendor-Data

下图展示了在属性 26“后面”封装的 VSA 数据包格式。

图 102：属性 26 后面封装的 VSA

Type	类型
Length	长度
Attributes-specific… (vendor-data)	指定属性 （厂商数据）

注释： 厂商负责指定它们的 VSA 所使用的格式。与属性相关的字段（也称为厂商数据）也取决于厂商对于属性的定义。

下面这个表格中描述了厂商指定 RADIUS IETF 属性表（下面第二个表格）中的重要字段，其中列出了支持的厂商指定 RADIUS 属性（IETF 属性 26）。

表 130：厂商指定的属性表字段描述

字段	**描述**
编号	下面表格中列出的所有属性都是 IETF 属性 26 的扩展
厂商指定的命令代码	用来标识具体厂商的代码。代码 9 定义了 Inspur VSA，311 定义了 Microsoft VSA，529 定义了 Ascend VSA
子类型编号	属性 ID 编号。这个编号与 IETF 属性的 ID 编号非常类似，只是它是封装在属性 26 后面的“第 2 层”ID 编号
属性	属性的 ASCII 字符串名称
描述	属性的描述

表 131：厂商指定的 RADIUS IETF 属性

编号	厂商指定的公司代码	子类型编号	属性	描述
MS-CHAP 属性				
26	311	1	MSCHAP-Response	在用于质询的响应消息中包含由 PPP MS-CHAP 用户提供的响应值。它仅用于 Access-Request（访问请求）数据包。这个属性与 PPP CHAP 标识符相同（RFC 2548）
26	311	11	MSCHAP-Challenge	包含由网络接入服务器向 MS-CHAP 用户发送的质询消息。它会用于 Access-Request（访问请求）和 Access-Challenge（访问质询）数据包中（RFC 2548）
VPDN 属性				
26	9	1	2tp-cm-local-window-size	为 L2TP 控制消息指定最大的接收窗口大小。这个值会在隧道建立期间通告给对等体
26	9	1	l2tp-drop-out-of-order	通过丢弃接收到的失序数据包，来遵从数据包的序列号。这并不保证数据包中一定会发送序列号，只是在接收到的时候进行控制
26	9	1	l2tp-hello-interval	为 Hello 存活间隔指定秒钟数。Hello 数据包会按照在这里配置的秒钟数，在没有数据发送时发送到隧道上
26	9	1	l2tp-hidden-avp	在启用后，L2TP 控制消息中的敏感 AVP 会被干扰或隐藏
26	9	1	l2tp-nosession-timeout	指定一个秒钟数，让隧道在这个时间值超时并关闭前，保持活跃
26	9	1	tunnel-tos-reflect	从每个负载数据包的 IP 头部复制 IP ToS 字段，并在 LNS 上进入隧道的数据包中，写入隧道数据包的 IP 头部
26	9	1	l2tp-tunnel-authen	如果设置了这个属性，就会执行 L2TP 隧道认证
26	9	1	l2tp-tunnel-password	用于 L2TP 隧道认证和 AVP 隐藏的共享秘密

26	9	1	l2tp-udp-checksum	这是一个授权属性，定义了 L2TP 是否应该为数据包执行 UDP 校验和。有效值是“yes”和“no”。默认值是 no
储存和转发传真属性				
26	9	3	Fax-Account-Id-Origin	指出系统管理员为 **mmoip aaa receive-id** 或 **mmoip aaa send-id** 命令定义的帐户 ID 源
26	9	4	Fax-Msg-Id=	指出由储存和转发传真分配的唯一传真消息识别编号
26	9	5	Fax-Pages	指出在这个传真会话中传输或接收的页数。页数中包含封面
26	9	6	Fax-Coverpage-Flag	指出出站网关是否为这个传真会话生成了封面。True 标识生成了封面；False 标识没有生成封面
26	9	7	Fax-Modem-Time	以秒为单位指出调制解调器发送传真数据（x）的时间，以及发送传真会话的总时长，其中包括传真邮件和 PSTN 时间，格式为 x/y。举例来说，10/15 标识传输时间花费了 10 秒钟，总传输会话花费了 15 秒钟
26	9	8	Fax-Connect-Speed	指出这个传真邮件初始化传输或接收时的调制解调器速率。可能的值包括 1200、4800、9600 和 14400
26	9	9	Fax-Recipient-Count	指出这个传真传输的接收方数量。直到电子邮件服务器支持会话模式，编号应该为 1
26	9	19	Fax-Process-Abort-Flag	指出传真会话的状态是终止还是成功。True 标识会话被终止；False 标识会话成功
26	9	11	Fax-Dsn-Address	指出要发送的 DSN 的地址
26	9	12	Fax-Dsn-Flag	指出是否启用了 DSN。True 表示已启用 DSN；False 表示未启用 DSN
26	9	13	Fax-Mdn-Address	指出要发送的 MDN 的地址

26	9	14	Fax-Mdn-Flag	指出是否启用了消息传递通知（MDN）。True 表示已启用 MDN；False 表示未启用 MDN
26	9	15	Fax-Auth-Status	指出这个传真会话的认证是否成功。这个字段的值可能是成功、失败、旁路或未知
26	9	16	Email-Server-Address	指出处理出站传真邮件消息的电子邮件服务器 IP 地址
26	9	17	Email-Server-Ack-Flag	指出入站网关已经从接收传真邮件消息的电子邮件服务器那里接收到了肯定的确认
26	9	18	Gateway-Id	指出处理传真会话的网关名称。名称显示为以下格式： 主机名.域名
26	9	19	Call-Type	描述传真活动的类型：传真接收或传真发送
26	9	20	Port-Used	指出传输或接收这个传真邮件所使用的 Inspur AS5300 上的插槽/端口编号
26	9	21	Abort-Cause	如果传真会话终结，则指出发出终结信令的系统组成部分。能够触发终结事件的系统组成部分包括 FAP（传真应用进程）、TIFF（TIFF 阅读器或 TIFF 写入器）、传真邮件客户端、传真邮件服务器、ESMTP 客户端或 ESMTP 服务器
H323 属性				
26	9	23	Remote-Gateway-ID （H323 远端地址）	指出远端网关的 IP 地址
26	9	24	Connection-ID （h323 会议 ID）	标识会议 ID
26	9	25	Setup-Time （h323 建立时间）	指出这个连接建立的时间，标识为协调世界时间（UTC），以前称为格林威治时间（GMT）和 Zulu 时间
26	9	26	Call-Origin （h323 呼叫源）	指出与网关相关联的呼叫源。可能出现的值是发起和

				终结（应答）
26	9	27	Call-Type （h323 呼叫类型）	指出呼叫线路类型。可能出现的值是 **telephony** 和 **VoIP**
26	9	28	Connect-Time （h323 连接时间）	以 UTC 时间指出这条呼叫线路的连接时间
26	9	29	Disconnect-Time （h323 断开连接时间）	以 UTC 时间指出这条呼叫线路断开的时间
26	9	30	Disconnect-Cause （h323 连接断开原因）	根据 Q.931 定义，指定连接离线的原因
26	9	31	Voice-Quality （h323 语音质量）	指定影响一通呼叫语音质量的损伤因子（ICPIF）
26	9	32	Gateway-ID （h323 网关 ID）	指出底层网关的名称
大范围拨出属性				
26	9	1	callback-dialstring	为回拨定义一组拨号字符串
26	9	1	data-service	无描述信息可用
26	9	1	dial-number	定义拨叫号码
26	9	1	force-56	确定网络访问服务器是否只使用隧道的 56 K 部分，即使所有 65 K 都是可用的
26	9	1	map-class	让用户配置文件能够调用配置在 map-class 中的信息，网络访问服务器使用相同名称的 map-class 进行拨出行为
26	9	1	send-auth	定义 CLID 认证后，用户名密码认证使用的协议（PAP 或 CHAP）
26	9	1	send-name	PPP 名称认证。要想应用 PAP，不能在接口配置 **ppp pap sent-name password** 命令。对于 PAP 来说，“preauth:send-name”和“preauth:send-secret”会作为出向认证的 PAP 用户名和 PAP 密码。对于 CHAP 来说，“preauth:send-name”不仅会被用于出向认证，还会被用于入向认证。在 CHAP 入向环境中，NAS 会使用发往主叫方的质询数据包中，“preauth:send-name”中定义的名称。

				注释： send-name 属性会发生变化：初始时，它提供现在由 send-name 和 remote-name 属性提供的功能。由于已经添加 remote-name 属性，因此 send-name 属性被限制为它当前的行为
26	9	1	send-secret	PPP 密码认证。厂商指定的属性（VSA）“preauth:send-name”和“preauth:send-secret”会被用于出向认证的 PAP 用户名和 PAP 密码。在 CHAP 出向环境中，“preauth:send-name”和“preauth:send-secret”会被用于响应数据包中
26	9	1	remote-name	提供用于大规模拨出的远程主机的名称。拨号程序会检查大规模拨出使用的远程名称是否与通过认证的名称相匹配，以防止发生用户 RADIUS 错误配置（举例来说，拨打有效的电话号码，但连接到错误的设备）
其他属性				
26	9	2	Inspur-NAS-Port	为 NAS 端口审计定义其他厂商指定的属性（VSA）信息。要想以属性值对（AV 对）格式指定其他 NAS 端口信息，用户要使用全局配置命令 **radius-server vsa send** **注释：** 这个 VSA 通常用于审计，但也可以用在认证（Access-Request）数据包中
26	9	1	min-links	为 MLP 设置最少链路数量
26	9	1	proxyacl#<n>	允许用户通过认证代理特性，配置下载用户配置文件（动态 ACL），使用户能够配置授权，来放行穿过指定接口的流量。
26	9	1	spi	承载归属代理（Home Agent）所需的认证信息，在

				注册期间对移动节点进行认证。这个信息与 **ip mobile secure host <addr>**配置命令的语法相同。基本上，它包含该字符串之后配置命令的其余部分。它提供了安全参数索引（SPI）、密钥、认证算法、认证模式和重放保护时间戳范围。

厂商私有的 RADIUS 服务器通信

尽管 IETF 在 RADIUS 标准草案中规定了交换机和 RADIUS 服务器之间传递厂商专有信息的方法，但一些厂商仍会以一种独特的方式对 RADIUS 属性集进行扩展。Inspur INOS 软件支持厂商私有 RADIUS 属性中的一个子集。

如前所述，要想配置 RADIUS（无论是厂商私有的或是符合 IETF 草案的），用户必须指定运行 RADIUS 服务器守护程序的主机及其与交换机共享的秘密文本字符串。用户可以使用全局配置命令 **radius server** 指定 RADIUS 主机和秘密文本字符串。

如何配置 RADIUS

标识 RADIUS 服务器主机

要想为与设备通信的所有 RADIUS 服务器设置全局配置，用户需要使用以下三个全局配置命令：**radius-server timeout**、**radius-server retransmit** 和 **radius-server key**。

用户可以配置设备，通过使用 AAA 服务器组，将现有的服务器主机汇总起来用于身份认证。更多详细信息，用户可以参考下面的相关主题。

用户还需要在 RADIUS 服务器上配置一些设置。这些设置包括设备的 IP 地址，以及服务器和设备共享的密钥字符串。更多详细信息，用户可以参考 RADIUS 服务器文档。

用户可以按照以下步骤配置基于服务器的 RADIUS 服务器通信。

在开始前

如果用户在设备上配置了全局功能，并针对每台服务器服务器配置了功能（超时、重传和密钥命令），那么针对每台服务器设置的定时器、重传和密钥值命令，会覆盖全局配置的定时器、重传和密钥值命令。在所有 RADIUS 服务器上配置这些设置的信息，用户可以参考下面的相关主题。

总步骤

1. enable

2. configure terminal

3. radius server *server name*

4. address {**ipv4 | ipv6**}*ip address*{ **auth-port** *port number* | **acct-port** *port number*}

5. key *string*

6. retransmit *value*

7. timeout *seconds*

8. end

9. show running-config

10. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式 • 在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**radius server** *server name* **示例：** `Device(config)# radius server rsim`	
步骤 4	**address** {**ipv4	ipv6**}*ip address*{ **auth-port** *port number*	**acct-port** *port number*} **示例：** `Device(config-radius-server)# address ipv4 124.2.2.12 auth-port 1612`	（可选）指定 RADIUS 服务器参数。在 **auth-port** *port-number* 部分为认证请求指定 UDP 目的端口。默认值是 1645，取值范围是 0 至 65535 在 **acct-port** *port-number* 部分指定认证请求的 UDP 目的端口。默认值是 1646
步骤 5	**key** *string* **示例：** `Device(config-radius-server)# key rad123`	（可选）在 **key** *string* 部分指定设备和运行了 RADIUS 守护程序的 RADIUS 服务器之间所使用的认证和加密密钥。 **注释：** 密钥的文本字符串必须与 RADIUS 服务器上使用的加密密钥相匹配。用户始终要把密钥配置为 **radius server** 命令中的最后一项。字符串前面的空格会被忽略，但密钥结尾处可以设置空格。如果在密钥中使用了空格，用户不要把密钥括在引号中，除非引号是密钥的一部分
步骤 6	**retransmit** *value* **示例：** `Device(config-radius-server)#`	（可选）指定当服务器没有响应或响应较慢时，重新发送 RADIUS 请求的次数。取值范围是 1 至 100。这个设置会覆盖全局配置命令 **radius-server retransmit** 中的设置

	retransmit 10	
步骤 7	**timeout** *seconds* **示例：** Device(config-radius-server)# **timeout 60**	（可选）指定设备等待 RADIUS 服务器的响应，等待指定的时间间隔后重新发送请求。取值范围是 1 至 1000。这个设置会覆盖全局配置命令 **radius-server timeout** 中的设置
步骤 8	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式
步骤 9	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 10	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

配置 RADIUS 登录认证

用户可以按照以下步骤来配置 RADIUS 登录认证：

在开始前

要想使用 AAA 方法来保护设备上的 HTTP 访问，用户必须使用全局配置命令 **ip http authentication aaa** 来配置设备。配置 AAA 认证并不会使用 AAA 方法来保护设备上的 HTTP 访问。

总步骤

1. enable

2. configure terminal

3. aaa new-model

4. aaa authentication login {**default** | *list-name*} *method1* [*method2...*]

5. line [**console** | **tty** | **vty**] *line-number* [*ending-line-number*]

6. login authentication {**default** | *list-name*}

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** Device> **enable**	进入特权 EXEC 模式。在提示时输入密码

步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**aaa new-model** **示例：** Device(config)# **aaa new-model**	启用 AAA
步骤 4	**aaa authentication login** {**default** | *list-name*} *method1* [*method2...*] **示例：** Device(config)# **aaa authentication login default local**	创建一个登录认证方法列表。 • 要想创建一个默认列表，当用户*没有*在 **login authentication** 命令中指定命名列表时就使用这个默认列表，用户需要在默认情况下使用的方法后面添加 **default** 关键字。默认方法列表会自动被应用到所有端口 • 在 *list-name* 部分指定一个字符串，用来命名用户创建的列表 • 在 *method1...*部分指定认证算法使用的实际方法。其他认证方法只有当前一个方法返回了错误响应消息时才会使用，而不是返回失败消息时使用。 用户可以选择以下方法之一： • *enable*——使用 enable 密码来进行认证。在用户可以使用这个认证方法前，必须使用全局配置命令 **enable** *password* 来定义一个 enable 密码 • *group radius*——使用 RADIUS 认证。在用户可以使用这个认证方法前，必须配置 RADIUS 服务器 • *line*——使用线路密码来进行认证。在用户可以使用这个认证方法前，必须先定义一个线路密码。用户可以使用线路配置命令 **password** *password* • *local*——使用本地用户名数据库进行认证。用户必须输入数据库中的用户名信息。

		需要使用全局配置命令 **username** *name* **password** 进行配置 • *local-case*——使用区分大小写的本地用户名数据库进行认证。用户必须使用全局配置命令 **username** *password*，把用户名信息输入到数据库中 • *none*——不为登录使用任何认证
步骤 5	**line** [**console** | **tty** | **vty**] *line-number* [*ending-line-number*] **示例：** `Device(config)# line 1 4`	进入线路配置模式，并对想要应用认证列表的线路进行配置
步骤 6	**login authentication** {**default** | *list-name*} **示例：** `Device(config-line)# login authentication default`	把认证列表应用在一条或多条线路上。 • 如果用户指定了 **default**，就使用命令 **aaa authentication login** 创建默认列表 • 在 *list-name* 部分指定 **aaa authentication login** 命令中创建的列表
步骤 7	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 8	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 9	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

定义 AAA 服务器组

用户可以使用 **server** 服务器组配置命令，把指定服务器关联到用户定义的服务器组中。用户可以使用服务器的 IP 地址来标识服务器，或使用可选关键字 **auth-port** 和 **acct-port** 来标识多个主机实例或条目。

用户可以按照以下步骤来定义 AAA 服务器组：

总步骤

1. enable

2. configure terminal

3. radius server *name*

4. address {**ipv4** | **ipv6**} {*ip-address* | *hostname*} **auth-port** *port-number* **acct-port** *port-number*

5. key *string*

6. end

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**radius server** *name* **示例：** `Device(config)# radius server ISE`	为受保护访问证书（PAC）的部署指定 RADIUS 服务器的名称，并进入 RADIUS 服务器配置模式。 设备也为 IPv6 支持 RADIUS
步骤 4	**address** {**ipv4**	**ipv6**} {*ip-address*	*hostname*} **auth-port** *port-number* **acct-port** *port-number* **示例：** `Device(config-radius-server)# address ipv4 10.1.1.1 auth-port 1645 acct-port 1646`	为 RADIUS 服务器审计和认证参数配置 IPv4 地址
步骤 5	**key** *string* **示例：** `Device(config-radius-server)# key inspur123`	指定设备和 RADIUS 服务器之间所使用的认证和加密密钥。
步骤 6	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 7	**show running-config**	检查用户输入的信息

	示例： Device# **show running-config**	
步骤 8	**copy running-config startup-config** 示例： Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

为用户特权访问和网络服务配置 RADIUS 授权

注释： 对于通过 CLI 登录且已通过了认证的用户，即使配置了授权，也会绕过授权。

用户可以按照以下步骤为特权访问和网络服务配置 RADIUS 授权：

总步骤

1. enable

2. configure terminal

3. aaa authorization network radius

4. aaa authorization exec radius

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**aaa authorization network tacacs+** 示例： Device(config)# **aaa authorization network radius**	配置交换机为所有与网络相关的服务请求使用 RADIUS 授权
步骤 4	**aaa authorization exec tacacs+** 示例： Device(config)# **aaa authorization exec radius**	配置交换机为特权 EXEC 的访问使用 RADIUS 授权。 **exec** 关键字可能会返回用户配置文件信息（比如 **autocommand** 信息）
步骤 5	**end**	返回特权 EXEC 模式

	示例： Device(config)# **end**	
步骤 6	**show running-config** 示例： Device# **show running-config**	检查用户输入的信息
步骤 7	**copy running-config startup-config** 示例： Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

接下来做什么？

用户可以使用全局配置命令 **aaa authorization** 和 **radius** 关键字设置指定参数，来限制用户访问特权 EXEC 模式的网络访问行为。

aaa authorization exec radius local 命令中可以设置以下三个授权参数：

- 如果使用 RADIUS 执行认证的话，用户可以使用 RADIUS 来提供特权 EXEC 访问的授权；
- 如果没有使用 RADIUS 执行认证的话，用户可以使用本地数据库来进行授权。

开始使用 RADIUS 审计

用户可以按照以下步骤开始使用 RADIUS 审计。

总步骤

1. enable

2. configure terminal

3. aaa accounting network start-stop radius

4. aaa accounting exec start-stop radius

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** 示例： Device# **configure terminal**	进入全局配置模式
步骤 3	**aaa accounting network start-stop radius**	为所有与网络相关的服务请求启用 RADIUS 审计

	示例： Device(config)# **aaa accounting network start-stop radius**	
步骤 4	**aaa accounting exec start-stop radius** 示例： Device(config)# **aaa accounting exec start-stop radius**	启用 RADIUS 审计，在开始特权 EXEC 处理时发送开始记录（start-record）审计通知，在结束时发送停止记录（stop-record）
步骤 5	**end** 示例： Device(config)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config** 示例： Device# **show running-config**	检查用户输入的信息
步骤 7	**copy running-config startup-config** 示例： Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

接下来做什么？

如果在 AAA 服务器不可达时，要与路由器建立会话，用户可以使用 **aaa accounting system guarantee-first** 命令。这条命令可以确保系统审计为第一条记录，这也是默认的条件。在有些情况下，这种设置可能会阻止用户在 Console 或终端连接上启动会话，直到系统重启才能解决问题，这可能需要 3 分钟以上的时间。

如果路由器重启时 AAA 服务器不可达，要想与路由器建立 Console 或 Telnet 会话，用户可以使用 **no aaa accounting system guarantee-first** 命令。

为所有 RADIUS 服务器配置相关设置

从特权 EXEC 模式开始，用户可以按照以下步骤为所有 RADIUS 服务器配置相关设置：

总步骤

1. configure terminal

2. radius-server key *string*

3. radius-server retransmit *retries*

4. radius-server timeout *seconds*

5. radius-server deadtime *minutes*

6. end

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 2	**radius-server key** *string* **示例：** Device(config)# **radius-server key your_server_key** Device(config)# **key your_server_key**	指定交换机和所有 RADIUS 服务器之间共享的秘密文本字符串。 **注释：** 密钥的文本字符串必须与 RADIUS 服务器上使用的加密密钥相匹配。字符串前面的空格会被忽略，但密钥结尾处可以设置空格。如果在密钥中使用了空格，用户不要把密钥括在引号中，除非引号是密钥的一部分
步骤 3	**radius-server retransmit** *retries* **示例：** Device(config)# **radius-server retransmit 5**	指定交换机发送 RADIUS 请求的次数，超出指定次数后交换机会放弃发送。默认值是 2；取值范围是 1 至 1000
步骤 4	**radius-server timeout** *seconds* **示例：** Device(config)# **radius-server timeout 3**	指定交换机在重新发送请求之前，等待 RADIUS 请求响应的时间。默认值是 5 秒钟；取值范围是 1 至 1000
步骤 5	**radius-server deadtime** *minutes* **示例：** Device(config)# **radius-server deadtime 0**	当 RADIUS 服务器么有响应认证请求时，用户使用这条命令来指定服务器停止发送请求的时间。这样做可以避免在尝试下一个配置的服务器之前，等待请求就超时了。默认值是 0；取值范围是 1 至 1440 分钟
步骤 6	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式
步骤 7	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 8	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

配置设备来使用厂商指定的 RADIUS 属性

用户可以按照以下步骤，配置设备来使用厂商指定的 RADIUS 属性：

总步骤

1. enable

2. configure terminal

3. radius-server vsa send [**accounting** | **authentication**]

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**radius-server vsa send** [**accounting**	**authentication**] **示例：** `Device(config)# radius-server vsa send accounting`	启用设备来识别并使用 RADIUS IETF 属性 26 中定义的 VSA。 • （可选）使用 **accounting** 关键字把一部分厂商指定的属性设置为只发送审计属性 • （可选）使用 **authentication** 关键字把一部分厂商指定的属性设置为只发送认证属性 如果用户在输入这条命令时没有设置关键字的话，会同时使用审计和认证厂商指定的属性
步骤 4	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 6	**copy running-config startup-config** **示例：** `Device# copy running-config`	（可选）把输入的命令保存到配置文件中

	startup-config	

配置设备来使用厂商私有的 RADIUS 服务器通信

用户可以按照以下步骤，来配置设备使用厂商私有的 RADIUS 服务器通信。

总步骤

1. enable

2. configure terminal

3. radius server *server name*

4. address { **ipv4 | ipv6** }*ip address*

5. non-standard

6. key *string*

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**radius server** *server name* **示例：** `Device(config)# radius server rsim`	指定 RADIUS 服务器
步骤 4	**address** { **ipv4	ipv6** }*ip address* **示例：** `Device(config-radius-server)# address ipv4 172.24.25.10`	（可选）指定 RADIUS 服务器的 IP 地址
步骤 5	**non-standard** **示例：** `Device(config-radius-server)# non-standard`	使用厂商私有的 RADIUS 部署方式来标识 RADIUS 服务器
步骤 6	**key** *string*	指定设备和厂商私有 RADIUS 服务器之间所使用的共享秘密文本字符

	示例： Device(config-radius-server)# **key rad123**	串。设备和 RADIUS 服务器会使用这个文本字符串来加密密码和交换响应信息
步骤 7	**end** 示例： Device(config-radius-server)# **end**	返回特权 EXEC 模式
步骤 8	**show running-config** 示例： Device# **show running-config**	检查用户输入的信息
步骤 9	**copy running-config startup-config** 示例： Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

在设备上配置 CoA

用户可以按照以下步骤在设备上配置 CoA，用户需要按序配置。

总步骤

1. enable
2. configure terminal
3. aaa new-model
4. aaa server radius dynamic-author
5. client {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]
6. server-key [**0** | **7**] *string*
7. port *port-number*
8. auth-type {**any** | **all** | **session-key**}
9. ignore session-key
10. ignore server-key
11. authentication command bounce-port ignore
12. authentication command disable-port ignore
13. end
14. show running-config
15. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： Device> **enable**	进入特权 EXEC 模式。在提示时输入密码

步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**aaa new-model** **示例：** `Device(config)# aaa new-model`	启用 AAA
步骤 4	**aaa server radius dynamic-author** **示例：** `Device(config)# aaa server radius dynamic-author`	把设备配置为认证、授权和审计（AAA）服务器，用来与外部策略服务器协同工作
步骤 5	**client** {*ip-address* | *name*} [**vrf** *vrfname*] [**server-key** *string*]	进入动态授权本地服务器配置模式，并指定 RADIUS 客户端，设备会从这个 RADIUS 客户端接受 CoA 请求和断开连接请求
步骤 6	**server-key** [**0** | **7**] *string* **示例：** `Device(config-sg-radius)# server-key your_server_key`	配置设备和 RADIUS 客户端之间共享的 RADIUS 密钥
步骤 7	**port** *port-number* **示例：** `Device(config-sg-radius)# port 25`	在设备上指定端口，让设备通过这个端口侦听 RADIUS 客户端发来的 RADIUS 请求
步骤 8	**auth-type** {**any** | **all** | **session-key**} **示例：** `Device(config-sg-radius)# auth-type any`	指定设备为 RADIUS 客户端使用的授权类型。 客户端必须匹配用户配置的所有属性，才能获得授权
步骤 9	**ignore session-key**	（可选）配置设备来忽略会话密钥。 有关 **ignore** 命令的更多信息，用户可以参考 icntnetworks.com 上的 *Inspur INOS Intelligent Services Gateway Command Reference*
步骤 10	**ignore server-key** **示例：** `Device(config-sg-radius)# ignore server-key`	（可选）配置设备来忽略服务器密钥。 有关 **ignore** 命令的更多信息，用户可以参考 icntnetworks.com 上的 *Inspur INOS Intelligent Services Gateway Command Reference*
步骤 11	**authentication command bounce-port**	（可选）配置设备来忽略 CoA 请求，

	ignore **示例：** `Device(config-sg-radius)#` **`authentication command bounce-port ignore`**	以便暂时禁止在端口上发起会话。暂时禁用端口的目的是为了当 VLAN 发生变化且终端设备无法检测这个变化时，触发 DHCP 重配置
步骤 12	**authentication command disable-port ignore** **示例：** `Device(config-sg-radius)#` **`authentication command disable-port ignore`**	（可选）配置设备来忽略非标准命令：该命令请求发起会话的端口变为管理失效模式。关闭端口的结果是会话终结。 用户需要使用标准 CLI 或 SNMP 命令来重新启用端口
步骤 13	**end** **示例：** `Device(config-sg-radius)#` **`end`**	返回特权 EXEC 模式
步骤 14	**show running-config** **示例：** `Device#` **`show running-config`**	检查用户输入的信息
步骤 15	**copy running-config startup-config** **示例：** `Device#` **`copy running-config startup-config`**	（可选）把输入的命令保存到配置文件中

监控 CoA 功能

表 132：特权 EXEC show 命令

命令	目的
show aaa attributes protocol radius	显示 RADIUS 命令的 AAA 属性

表 133：全局排错命令

命令	目的
debug radius	显示有关 RADIUS 排错的信息
debug aaa coa	显示有关 CoA 进程排错的信息
debug aaa pod	显示有关 POD 数据包排错的信息
debug aaa subsys	显示有关 POD 数据包排错的信息
debug cmdhd [**detail** | **error** | **events**]	显示排错命令的头部信息

有关这些命令显示信息中各个字段的详细信息，用户可以参考这个版本的命令参考文档。

其他参考资料

相关文档

相关主题	文档名称
为会话感知类网络配置身份控制策略和身份服务模版	Session Aware Networking Configuration Guide, Inspur INOS (Inspur 6850 Switches) http://www.icntnetworks.com
配置 RADIUS、TACACS+、SSH、802.1X 和 AAA	Securing User Services Configuration Guide Library, Inspur INOS (Inspur 6850 Switches) http://www.icntnetworks.com

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

配置 Kerberos

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置 Kerberos 控制交换机访问的先决条件

配置 Kerberos 来控制交换机访问的先决条件如下所示：

- 为了使远程用户能够向网络服务进行身份验证，用户必须在 Kerberos 域中配置主机和 KDC，以便使用户和网络服务能够进行通信和相互认证。为此，用户必须让它们能够识别出彼此。用户需要把主机的条目添加到 KDC 上的 Kerberos 数据库中，并将由 KDC 生成的 KEYTAB 文件添加到 Kerberos 域中的所有主机上。用户还可以在 KDC 数据库中为用户创建条目；
- Kerberos 服务器可以是一台交换机，用户需要把它配置为网络安全服务器，这样它就可以使用 Kerberos 协议来对用户进行认证了。

用户在为主机和用户添加或创建条目时，需要遵从以下指导：

- Kerberos 规则名称*必须*全都是小写字符；
- Kerberos 实例名称*必须*全都是小写字符；
- Kerberos 域名*必须*全都是大写字符。

Kerberos 的相关信息

这一部分提供了 Kerberos 的相关信息。

Kerberos 和交换机访问

这一部分描述了如何启用和配置 Kerberos 安全系统，它能够使用一个受信任的第三方，来为网络资源进行请求认证。

注释： 在 Kerberos 配置示例中个，受信任的第三方可以是任何支持 Kerberos 的交换机，这太交换机会被用户配置为网络安全服务器，并且它能够使用 Kerberos 协议来对用户进行认证。

Kerberos 概述

Kerberos 是一项秘密密钥网络认证协议，由麻省理工学院（MIT）开发。它使用数据加密标准（DES）加密算法进行加密和认证，并为网络资源的请求提供认证。Kerberos 使用受信任第三方的概念来对用户和服务提供安全认证。这个受信任的第三方称为*密钥分发中心*（KDC）。Kerberos 会验证用户是他们所声称的用户，以及他们所使用的网络服务是他们所声称要使用的服务。为此，KDC 或受信任的 Kerberos 服务器会向用户发送票据（ticket）。这些票据具有有限的生命周期，并储存在用户证书缓存中。Kerberos 服务器会使用这些票据，而不是使用用户名和密码来验证用户和网络服务。

注释： Kerberos 服务器可以是任何被用户配置为网络安全服务器的交换机，并且它能够使用 Kerberos 协议来对用户进行认证。

Kerberos 的证书机制使用称为*单一登录*（Single Logon）的进程。这个进程会一次性对用户进行认证，然后根据接受的用户证书范围，在这个范围中的任何地方用户都会通过安全验证（无需加密另一个密码）。

这个软件版本支持 Kerberos 5，已经在使用 Kerberos 5 的组织机构可以在 KDC 上，使用已在其他网络主机（比如 UNIX 服务器和 PC）上使用的相同 Kerberos 身份认证数据库。

Kerberos 支持以下网络服务：

- Telnet
- rlogin
- rsh

下面这个表格中列出了常见的 Kerberos 相关术语和定义。

表 134：Kerberos 术语

术语	定义
认证	用户或服务向另一个服务验证自己身份的过程。举例来说，客户端可以向交换机验证自己的身份，或者交换机可以向另一台交换机验证自己的身份
授权	交换机用来识别用户拥有网络中哪些特权的方式，或者交换机用来识别用户能够执行哪些行为的方式
证书	表示认证票据的通用术语，比如 TGT[9] 和服务证书。Kerberos 的证书验证了用户或服务器的身份。如果一个网络服务决定信任发放票据的 Kerberos 服务器，它就可以使用证书来代替用户名和密码。证书的默认生命周期是 8 个小时
实例	Kerberos 规则的授权等级标签。大多数 Kerberos 规则的格式都是 *user@REALM*（比如 smith@EXAMPLE.COM）。Kerberos 规则和

	Kerberos 实例一起表示为 *user/instance@REALM*（比如 smith/admin@EXAMPLE.COM）。Kerberos 示例可以用来指定授权等级，如果用户认证成功了的话。提供每个网络服务的服务器可能会实施并使用 Kerberos 示例的授权映射，但这并不是强制行为。 **注释：** Kerberos 规则和实例的名称*必须*全都是小写字符 **注释：** Kerberos 域名*必须*全都是大写字符
KDC[10]	密钥分发中心，由 Kerberos 服务器和运行在一台网络主机上的数据库程序构成
Kerberos 化的	这个术语用来描述已经被变更为能够支持 Kerberos 证书架构的应用和服务
Kerberos 域	由用户、主机和网络服务构成的域，这些组成元素都会注册到 Kerberos 服务器上。Kerberos 服务器是受信任的设备，用来向用户或网络服务认证另一个用户或网络服务的身份。 **注释：** Kerberos 域名*必须*全都是大写字符
Kerberos 服务器	运行在网络主机上的一个守护进程。用户和网络服务把它们各自的身份注册到 Kerberos 服务器上。网络服务向 Kerberos 服务器进行查询，以此来认证其他网络服务
KEYTAB[11]	网络服务与 KDC 共享的密码。在 Kerberos 5 和后续的 Kerberos 版本中，网络服务在认证一个加密的服务证书时，会使用 KEYTAB 对其进行解密。在 Kerberos 5 之前的版本中，KEYTAB 称为 SRVTAB[12]
规则	也称为 Kerberos 实体，这是根据 Kerberos 服务器指定的设备身份或服务器身份 **注释：** Kerberos 规则名称*必须*全都是小写字符
服务证书	网络服务的证书。KDC 颁发后，这个证书是使用网络服务和 KDC 之间共享的密码进行加密的。这个密码也会与用户 TGT 共享
SRVTAB	网络服务与 KDC 共享的密码。在 Kerberos 5 或后续的 Kerberos 版本中，SRVTAB 也称为 KEYTAB
TGT	承认的票据，这是 KDC 颁发给通过认证的用户的证书。用户接收到 TGT 时，它们可以在 KDC 代表的 Kerberos 域中对网络服务进行认证

9 承认的票据

10 密钥分发中心
11 密钥表
12 服务器表

Kerberos 的工作原理

Kerberos 服务器可以是配置为网络安全服务器的任何设备，它可以使用 Kerberos 协议来认证远端用户。尽管用户可以使用多种方式来自定义 Kerberos，但远端用户在尝试访问网络服务时，必须通过三层安全防范措施，才能访问网络服务。
要想使用 Kerberos 服务器设备来对网络服务进行认证，远端用户必须遵从以下步骤：

向边界交换机进行认证

这部分描述了远端用户必须通过的第一层安全防范措施。用户必须首先向边界交换机进行认证。这时会发生以下事件：

1. 用户向边界交换机开启未 Kerberos 化的 Telnet 连接；
2. 交换机向用户提示输入用户名和密码；
3. 交换机为用户向 KDC 请求 TGT；
4. KDC 向交换机发送加密的 TGT，其中包含用户的身份；
5. 交换机尝试使用用户输入的密码来解密 TGT：
   - 如果解密成功的话，交换机上的用户认证就成功了；
   - 如果解密没有成功的话，用户会重复步骤 2：重新输入用户名和密码（如果大写键或数字键已开启或关闭的话），或输入另一个用户名和密码。

初始化非 Kerberos 化的 Telnet 会话并向边界交换机进行身份认证的远程用户位于防火墙内，但在访问网络服务之前，用户仍必须直接向 KDC 进行身份验证。用户必须向 KDC 进行身份验证，因为交换机上储存着 KDC 颁发的 TGT，并且这个 TGT 无法在用户登录到交换机前，用于其他认证。

从 KDC 获得 TGT

这一部分介绍了远程用户必须通过的第 2 层安全防范措施。用户现在必须向 KDC 进行认证，并从 KDC 获得 TGT 来访问网络服务。
如何向 KDC 进行认证的指导，用户可以参考 *Inspur INOS Security Configuration Guide, Release 12.4* 中，“Security Server Protocols”一章中的“Obtaining a TGT from a KDC”部分。

向网络服务进行认证

这部分介绍了远程用户必须通过的第 3 层安全防范措施。持有 TGT 的用户现在必须向 Kerberos 域中的网络服务进行身份验证。
如何向网络服务进行认证的指导，用户可以参考 *Inspur INOS Security Configuration Guide, Release 12.4* 中，“Security Server Protocols”一章中的“Authenticating to Network Services”部分。

如何配置 Kerberos

要想建立一个由 Kerberos 进行认证的服务器-客户端系统，用户需要按照以下步骤进行配置：

- 使用 Kerberos 命令来配置 KDC；
- 配置交换机来使用 Kerberos 协议。

监控 Kerberos 的配置

要想查看 Kerberos 的配置，用户可以使用以下命令：

- **show running-config**
- **show kerberos creds**：列出当前用户证书缓存中的证书
- **clear kerberos creds**：清除当前用户证书缓存中的所有证书，其中包括转发的证书

其他参考资料

相关文档

相关主题	文档名称
Kerberos 命令	*Inspur INOS Security Command Reference*

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

配置本地认证和授权

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

如何配置本地认证和授权

配置交换机来执行本地认证和授权

用户可以通过设置交换机实施本地模式的 AAA，以此实现不使用服务器的 AAA 操作。交换机会负责处理认证和授权事宜。这种配置不支持审计。

注释： 为了使用 AAA 方式来确保 HTTP 访问交换机的安全性，用户必须使用全局配置命令 **ip http authentication aaa** 来配置交换机。配置 AAA 认证并不会对使用 AAA 方法的交换机提供 HTTP 访问保护。

用户可以按照以下步骤来配置 AAA 操作，以本地的方式（而不是用服务器）来设置交换机实施 AAA：

总步骤

1. enable

2. configure terminal

3. aaa new-model

4. aaa authentication login default local

5. aaa authorization exec local

6. aaa authorization network local

7. username *name* [**privilege** *level*] {**password** *encryption-type password*}

8. end

9. show running-config

10. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable**	进入特权 EXEC 模式。在提示时输入密

	示例： `Device> enable`	码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**aaa new-model** **示例：** `Device(config)# aaa new-model`	启用 AAA
步骤 4	**aaa authentication login default local** **示例：** `Device(config)# aaa authentication login default local`	设置使用本地用户名数据库来执行登录认证。使用 **default** 关键字为所有端口使用本地用户数据库进行认证
步骤 5	**aaa authorization exec local** **示例：** `Device(config)# aaa authorization exec local`	配置用户 AAA 授权、检查本地数据库，并允许用户使用 EXEC 命令
步骤 6	**aaa authorization network local** **示例：** `Device(config)# aaa authorization network local`	为所有与网络相关的服务请求配置 AAA 授权
步骤 7	**username** *name* [**privilege** *level*] {**password***encryption-type password*} **示例：** `Device(config)# username your_user_name privilege 1 password 7 secret567`	进入本地数据库，并建立基于用户名的认证系统。 用户需要为每个用户重复配置以下命令： • 在 *name* 部分指定用户 ID，可以指定一个单词。不能使用空格和引号 • （可选）在 *level* 部分指定用户能够获得的特权等级。取值范围是 0 至 15。等级 15 是特权 EXEC 模式的访问权限。等级 0 是用户 EXEC 模式的访问权限 • 在 *encryption-type* 部分输入 0 来指定未加密的密码。输入 7 来指定隐藏密码 • 在 *password* 部分指定用户必须输

		入并获得交换机访问权限的密码。密码必须在 1 至 25 字符之间，可以包含空格，并且必须是 **username** 命令中配置的最后一个选项
步骤 8	**end** **示例：** `Device(config-sg-radius)# end`	返回特权 EXEC 模式
步骤 9	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 10	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

监控本地认证和授权

要想查看本地认证和授权的配置，用户可以使用特权 EXEC 命令 **show running-config**。

其他参考资料

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息	http://www.icntnetworks.com

聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	

第 91 章 配置安全壳（SSH）

注释： 从 Inspur INOS 12.2 版本开始，已弃用安全壳版本 1（SSHv1）。

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置安全壳的先决条件

在交换机上配置安全壳（SSH）拥有以下先决条件：

- 要想让 SSH 正常工作，交换机上需要有 Rivest、Shamir 和 Adleman（RSA）公钥/私钥对。这与安全复制协议（SCP）相同，SCP 依赖于 SSH 来提供传输安全性；
- 在启用 SCP 之前，用户必须先在交换机上正确配置了 SSH、认证和授权；
- 因为 SCP 依赖于 SSH 来提供传输安全性，因此路由器上必须拥有 Rivest、Shamir 和 Adleman（RSA）公钥/私钥对；
- SCP 依赖于 SSH 来提供安全性；
- SCP 需要认证、授权和审计（AAA）的授权配置，这样路由器才能够确定用户应该获得的正确特权等级；
- 用户必须有正确的授权，才能使用 SCP；
- 拥有能够使用 SCP 正确授权的用户，可以使用 **copy** 命令把交换机 Inspur INOS 文件系统（IFS）中的文件复制进来或复制出去。授权的管理员也可以从工作站执行相同的工作；

- 安全壳（SSH）服务器需要使用 IPsec（数据加密标准[DES]或 3DES）加密软件镜像；SSH 客户端需要使用 IPsec（DES 或 3DES）加密软件镜像；
- 用户可以使用全局配置模式的命令 **hostname** 和 **ip domain-name** 来为设备配置用户名和主机域名。

配置安全壳的限制条件

在设备上配置安全壳拥有以下限制条件：

- 交换机需要支持 Rivest、Shamir 和 Adleman（RSA）认证；
- SSH 只支持可执行 Shell 应用；
- 只有数据加密标准（DES，56 比特）和 3DES（168 比特）数据加密软件能够支持 SSH 服务器和 SSH 客户端。在 DES 软件镜像中，DES 是唯一可用的加密算法。在 3DES 软件镜像中，可以使用 DES 和 3DES 加密算法；
- 设备支持高级加密算法（AES）加密算法：128 比特密钥、192 比特密钥或 256 比特密钥。但不支持使用对称密码 AES 来加密密钥；
- 这个软件版本不支持 IP 安全（IPsec）；
- 在使用 SCP 时，用户不能在 **copy** 命令中输入密码。用户必须在看到提示时输入密码；
- 安全壳版本 1 中不支持登录旗标消息。安全壳版本 2 中可以支持；
- 在配置 Console 访问的反向 SSH 备用方法时，-l 分隔符关键字和用户 ID：{number} {ip-address}参数是必需的。

SSH 的相关信息

安全壳（SSH）是一个用来为设备提供安全远程连接的协议。SSH 比 Telnet 为远程连接提供了更多的安全性，它能够在设备认证时提供强健的加密措施。这个软件版本支持 SSH 版本 1（SSHv1）和 SSH 版本 2（SSHv2）。

SSH 和交换机访问

安全壳（SSH）是一个用来为设备提供安全远程连接的协议。SSH 比 Telnet 为远程连接提供了更多的安全性，它能够在设备认证时提供强健的加密措施。这个软件版本支持 SSH 版本 1（SSHv1）和 SSH 版本 2（SSHv2）。

SSH 能够为 IPv6 提供与 IPv4 相同的功能。对于 IPv6 来说，SSH 能够支持 IPv6 地址，并且能够为使用 IPv6 传输的远端 IPv6 节点，提供启用了安全加密的连接。

SSH 服务器、集成客户端和支持的版本

安全壳（SSH）集成客户端特性是运行在 SSH 协议上的应用，用来提供设备认证和加密。SSH 客户端能够使 Inspur 设备与其他 Inspur 设备之间建立安全加密的连接，或者与其他运行 SSH 服务器的设备之间建立安全加密的连接。这条连接提供的功能与带外 Telnet 连接提供的功

能类似，只不过这条连接是加密的。通过使用认证和加密，SSH 客户端可以在不安全的网络上提供安全通信。

SSH 服务器和 SSH 集成客户端都是运行在交换机上的应用。SSH 服务器与这个版本中支持的 SSH 客户端和非 Inspur SSH 客户端一起工作。SSH 客户端能够与可用的公开和商业 SSH 服务器一起工作。SSH 客户端能够支持数据加密标准（DES）、3DES 和密码认证。

交换机上支持 SSHv1 或 SSHv2 服务器。

交换机上支持 SSHv1 客户端。

注释： 只有当用户启用了 SSH 服务器后，才能使用 SSH 客户端功能。

为用户提供的用户认证功能与 Telnet 会话中提供的用户认证类似。SSH 也支持下列用户认证方式：

- TACACS+
- RADIUS
- 本地认证和授权

SSH 的配置指导

在把交换机配置为 SSH 服务器或 SSH 客户端时，用户需要遵从以下指导：

- SSHv1 服务器生成的 RSA 密钥对，也可以由 SSHv2 服务器使用，反之亦然；
- 如果在堆栈主用设备上运行 SSH 服务器，并且堆栈主用设备失效了，新的堆栈主用设备会使用前一个堆栈主用设备所生成的 RSA 密钥对；
- 如果用户在输入全局配置命令 **crypto key generate rsa** 后看到了 CLI 错误消息，并且 RSA 密钥对没有生成。那么用户需要重新配置用户名和域名，然后再次输入 **crypto key generate rsa** 命令。更多信息，用户可以参考相关主题部分；
- 在生成 RAS 密钥对时，交换机可能会显示出没有指定主机名的消息。如果看到了这条消息，用户必须使用全局配置命令 **hostname** 来配置主机名；
- 生成 RAS 密钥对时，交换机可能会显示出没有指定域名的消息。如果看到了这条消息，用户必须使用全局配置命令 **ip comain-name** 来配置 IP 域名；
- 在配置本地认证和授权的认证方法时，用户要确保在 Console 端口上禁用了 AAA。

安全复制协议概述

安全复制协议（SCP）特性为复制交换机配置文件或交换机镜像文件提供了一种安全且能够执行认证的方法。SCP 依赖于安全壳（SSH），SSH 是能够为 Berkeley r-tools 提供安全性的应用和协议。

要想让 SSH 正常工作，交换机上需要有 RSA 公钥/私钥对。这与 SCP 相同，SCP 依赖于 SSH 来提供传输安全性。

由于 SSH 也依赖于 AAA 认证，因此 SCP 也依赖于 AAA 授权，因此用户需要正确配置相关信息。

- 在启用 SCP 之前，用户必须先在交换机上正确配置了 SSH、认证和授权；
- 因为 SCP 依赖于 SSH 来提供传输安全性，因此路由器上必须拥有 Rivest、Shamir 和 Adleman（RSA）公钥/私钥对

注释： 在使用 SCP 时，用户不能在 **copy** 命令中输入密码。用户必须在看到提示时输入密码。

安全复制协议

安全复制协议（SCP）特性为复制设备配置文件或交换机镜像文件提供了一种安全，且能够进行认证的方法。SCP 的行为与远程复制（rcp）类似，后者来自于 Berkeley 远程工具集，只不过 ACP 依赖于 SSH 提供安全性。SCP 也需要配置认证、授权和审计（AAA）的授权功能，这样设备才能够确定用户应该使用的正确特权等级。要想配置安全复制特性，用户应该理解 SCP 的概念。

如何配置 SSH

设置设备来运行 SSH

用户可以按照以下步骤，设置设备来运行 SSH：

在开始前

为本地或远程访问配置用户认证功能。用户需要按顺序进行配置。更多信息用户可以参考相关主题部分。

总步骤

1. enable

2. configure terminal

3. hostname *hostname*

4. ip domain-name *domain_name*

5. crypto key generate rsa

6. end

7. show running-config

8. copy running-config startup-config

具体配置

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**hostname** *hostname* **示例：** `Device(config)# hostname your_hostname`	为用户的设备配置主机名和 IP 域名。 **注释：** 只有在把设备配置为 SSH 服务器时，才使用这个步骤
步骤 4	**ip domain-name** *domain_name*	为用户的设备配置一个域名

	示例： Device(config)# **ip domain-name your_domain**	
步骤 5	**crypto key generate rsa** 示例： Device(config)# **crypto key generate rsa**	在设备上为本地和远程认证启用 SSH 服务器功能，并生成一个 RSA 密钥对。为设备生成 RSA 密钥对会自动启用 SSH。 我们建议使用的最小系数大小是 1024 比特。 在生成 RSA 密钥时，用户会看到输入系数长度的提示。更长的系数会提供更高的安全性，但也会需要更长的时间来进行生成和使用。 **注释：** 只有在把设备配置为 SSH 服务器时，才使用这个步骤
步骤 6	**end** 示例： Device(config)# **end**	返回特权 EXEC 模式
步骤 7	**show running-config** 示例： Device# **show running-config**	检查用户输入的信息
步骤 8	**copy running-config startup-config** 示例： Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

配置 SSH 服务器

用户可以按照以下步骤来配置 SSH 服务器：

注释： 只有在把设备配置为 SSH 服务器时，才使用这个步骤。

总步骤

1. enable

2. configure terminal

3. ip ssh version [**1** | **2**]

4. ip ssh {**timeout** *seconds* | **authentication-retries** *number*}

5. 使用以下命令之一：

- **line vty***line_number*[*ending_line_number*]
- **transport input ssh**

6. end

7. show running-config

8. copy running-config startup-config

具体配置

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**ip ssh version** [**1**	**2**] **示例：** `Device(config)# ip ssh version 1`	（可选）配置设备来运行 SSH 版本 1 或 SSH 版本 2。 • **1**——配置设备来运行 SSH 版本 1 • **2**——配置设备来运行 SSH 版本 2 如果用户没有输入这条命令，或者没有指定关键字，SSH 服务器会选择 SSH 客户端所支持的最新 SSH 版本。举例来说，如果 SSH 客户端支持 SSHv1 和 SSHv2，那么 SSH 服务器会选择使用 SSHv2
步骤 4	**ip ssh** {**timeout** *seconds*	**authentication-retries** *number*} **示例：** `Device(config)# ip ssh timeout 90 authentication-retries 2`	配置 SSH 控制参数： • 以秒为单位指定超时值；默认值为 120 秒钟。取值范围是 0 至 120 秒。这个参数应用在 SSH 协商阶段。在连接建立后，设备会使用基于 CLI 会话的默认超时值 默认情况下，设备为网络中多条 CLI 会话同时支持 5 条加密的 SSH 连接（会话 0 至会话 4）。在开始执行 Shell 后，基于 CLI 的会话超时值会返回到默认的 10 分钟 • 指定客户端可以向服务器进行重认证的次数。默认值是 3；取值范围是 0 至 5 用户需要重复这个步骤来同时配置两个参数
步骤 5	使用以下命令之一： • **line vty** *line_number* [*ending_line_number*] • **transport input ssh** **示例：**	（可选）配置虚拟终端线路设置： • 进入线路配置模式，来配置虚拟终端线路设置。在 *line_number* 和 *ending_line_number* 部分指定一对线路，取值范围是 0 至 5 • 指定设备阻止非 SSH 的 Telnet 连

	Device(config)# **line vty 1 10** 或者 Device(config-line)# **transport input ssh**	接。限制路由器只支持 SSH 连接
步骤 6	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式
步骤 7	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 8	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

监控 SSH 的配置和状态

下面这个表格中的命令显示了 SSH 服务器的配置和状态。

表 135：显示 SSH 服务器配置和状态的命令

命令	目的
show ip ssh	显示 SSH 服务器的版本和配置信息
show ssh	显示 SSH 服务器的状态

其他参考资料

相关主题

相关主题	文档名称
为会话感知类网络配置身份控制策略和身份服务模版	Session Aware Networking Configuration Guide, Inspur INOS (Inspur 6850 Switches) http://www.icntnetworks.com
配置 RADIUS、TACACS+、SSH、802.1X 和 AAA	Securing User Services Configuration Guide Library, Inspur INOS (Inspur 6850 Switches) http://www.icntnetworks.com

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

SSH 的特性信息

版本	特性信息
Inspur INOS 12.2	引入该特性

用于 SSH 认证的 X.509v3 证书

用于 SSH 认证的 X.509v3 证书

SSH 认证的 X.509v3 证书特性在安全壳（SSH）服务器侧的服务器和用户认证使用 X.509v3 数字证书。

这部分描述了如何为数字证书配置服务器和用户证书配置文件。

查询特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置用于 SSH 认证的 X.509v3 证书的先决条件

用于 SSH 认证的 X.509v3 证书特性引入了 **ip ssh server algorithm authentication** 命令，来代替 **ip ssh server authenticate user** 命令。如果用户使用了 **ip ssh server authenticate user** 命令，设备上会显示以下信息。

Warning: SSH command accepted but this CLI will be deprecated soon. Please move to new CLI “ip ssh server algorithm authentication”. Please configure “default ip ssh server authenticate user” to make CLI ineffective.

- 用户可以使用命令 **default ip ssh server authenticate user** 来移除 **ip ssh server authenticate user** 命令的作用。然后使用 **ip ssh server algorithm authentication** 命令来启用 INOS 安全壳（SSH）。

配置用于 SSH 认证的 X.509v3 证书的限制条件

- 用于 SSH 认证的 X.509v3 证书特性只能实施在 INOS 安全壳（SSH）服务器侧；
- INOS SSH 服务器在 INOS SSH 服务器侧的服务器和用户认证上，只支持基于 x509v3-ssh-rsa 算法的认证。

用于 SSH 认证的 X.509v3 证书的相关信息

数字证书

认证的有效性取决于公共签名密钥和签名者身份之间的联系强度。X.509v3 格式（RFC5280）的数字证书用来提供身份管理。受信任的根证书机构及其中间证书机构的签名链，会把指定的公共签名密钥绑定到指定的数字身份。
公钥基础设施（PKI）信任点有助于管理数字证书。证书和信任点之间的关联有助于跟踪证书。信任点包含有关证书颁发机构（CA）、不同的身份参数和数字证书的信息。用户可以创建多个信任点来与不同的证书相关联。

使用 X.509v3 的服务器和用户认证

对于服务器认证来说，INOS 安全壳（SSH）服务器会把自己的证书发送到 SSH 客户端进行验证。这个服务器证书与服务器证书配置文件（配置在 ssh-server-cert-profile-server 配置模式中）中配置的信任点相关联。
对于用户认证来说，SSH 客户端会把用户的证书发送到 INOS SSH 服务器进行验证。SSH 服务器会使用服务器证书配置文件（配置在 ssh-server-cert-profile-user 配置模式中）中配置的公钥基础设施（PKI）信任点来验证入站的用户证书。
默认情况下，用户需要在 INOS SSH 服务器端为服务器和用户启用基于证书的身份认证。

如何配置用于 SSH 认证的 X.509v3 证书

配置 INOS SSH 服务器来为服务器认证使用数字证书

总步骤

1. enable
2. configure terminal
3. ip ssh server algorithm hostkey {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]}
4. ip ssh server certificate profile
5. server
6. trustpoint sign *PKI-trustpoint-name*
7. ocsp-response include
8. end

具体步骤

	命令或操作	目的
步 骤 1	**enable** **示例：**	进入特权 EXEC 模式。在提示时输入密码

	Device> enable	
步骤 2	**configure terminal** **示例：** Device# configure terminal	进入全局配置模式
步骤 3	**ip ssh server algorithm hostkey {x509v3-ssh-rsa [ssh-rsa] | ssh-rsa [x509v3-ssh-rsa]}** **示例：** Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa	定义主机密钥算法的顺序。只有配置的算法会用于与安全壳（SSH）客户端的协商中。 **注释：** INOS SSH 服务器上必须至少配置一个主机密钥算法： • ssh-rsa——基于公钥的算法 • x509v3-ssh-rsa——基于证书的认证
步骤 4	**ip ssh server certificate profile** **示例：** Device(config)# ip ssh server certificate profile	配置服务器证书配置文件和用户证书配置文件，并进入 SSH 证书配置文件配置模式
步骤 5	**server** **示例：** Device(ssh-server-cert-profile)# server	配置服务器证书配置文件，并进入 SSH 服务器证书配置文件服务器配置模式
步骤 6	**trustpoint sign** *PKI-trustpoint-name* **示例：** Device(ssh-server-cert-profile-server)# trustpoint sign trust1	把公钥基础设施（PKI）信任点关联到服务器证书配置文件。SSH 服务器会使用这个 PKI 信任点关联的证书来对服务器进行认证
步骤 7	**ocsp-response include** **示例：** Device(ssh-server-cert-profile-server)# ocsp-response include	（可选）随服务器证书发送在线证书状态协议（OCSP）响应或 OCSP 闭合（Stapling）。 **注释：** 默认情况下配置的是这条命令的“no”形式，也就是不会随服务器证书发送 OCSP 响应
步骤 8	**end** **示例：** Device(ssh-server-cert-profile-	离开 SSH 服务器证书配置文件服务器配置模式，并进入特权 EXEC 模式

	server)# **end**	

配置 INOS SSH 服务器为用户认证验证用户数字证书

总步骤

1. enable

2. configure terminal

3. ip ssh server algorithm authentication {**publickey** | **keyboard** | **password**}

4. ip ssh server algorithm publickey {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]}

5. ip ssh server certificate profile

6. user

7. trustpoint verify *PKI-trustpoint-name*

8. ocsp-response required

9. end

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**ip ssh server algorithm authentication** {**publickey** | **keyboard** | **password**} **示例：** `Device(config)# ip ssh server algorithm authentication publickey`	定义用户认证算法的顺序。只有配置的算法会用于与安全壳（SSH）客户端的协商中。 **注释：** INOS SSH 服务器上必须至少配置一个用户认证算法 **注释：** 要想为用户认证使用证书方式，用户必须配置 **publickey** 关键字 **注释：** 使用 **ip ssh server algorithm authentication** 命令代替 **ip ssh server authenticate user** 命令
步骤 4	**ip ssh server algorithm publickey** {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]} **示例：** `Device(config)# ip ssh server algorithm publickey x509v3-ssh-rsa`	定义公钥算法的顺序。只有配置的算法可以被 SSH 客户端接受并用于用户认证。 **注释：** INOS SSH 客户端上必须至少配置一个公钥算法： • ssh-rsa——基于公钥的算法 • x509v3-ssh-rsa——基于证书的认证
步骤 5	**ip ssh server certificate profile**	配置服务器证书配置文件和用户证书

	示例： `Device(config)# ip ssh server certificate profile`	配置文件，并进入 SSH 证书配置文件配置模式中
步骤 6	**user** **示例：** `Device(ssh-server-cert-profile)# user`	配置用户证书配置文件，并进入 SSH 服务器证书配置文件用户配置模式
步骤 7	**trustpoint verify** *PKI-trustpoint-name* **示例：** `Device(ssh-server-cert-profile-user)# trustpoint verify trust2`	配置公钥基础设施（PKI）信任点，以此用来验证入站的用户证书。 **注释：** 用户需要多次执行相同的命令来配置多个信任点。用户最多可以配置 10 个信任点
步骤 8	**ocsp-response required** **示例：** `Device(ssh-server-cert-profile-user)# ocsp-response required`	（可选）使用在线证书状态协议（OCSP）来对入站用户的证书做出响应。 **注释：** 默认的配置是这条命令的“no”格式，无需 OCSP 响应就会接受用户证书
步骤 9	**end** **示例：** `Device(ssh-server-cert-profile-user)# end`	离开 SSH 服务器证书配置文件用户配置模式，并进入特权 EXEC 模式

验证使用数字证书的服务器和用户认证配置

总步骤

1. enable

2. show ip ssh

具体步骤

步骤 1 enable

进入特权 EXEC 模式

- 在提示时输入密码

示例：

```
Device> enable
```

步骤 2 show ip ssh

显示当前配置的认证方法。来确认使用的基于证书的认证，并确保配置的主机密钥算法是 x509v3-ssh-rsa 算法。

示例：

```
Device# show ip ssh
SSH Enabled - version 1.99
Authentication methods:publickey,keyboard-interactive,password
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa
Hostkey Algorithms:x509v3-ssh-rsa,ssh-rsa
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits
```

用于 SSH 认证的 X.509v3 证书配置示例

示例：配置 INOS SSH 服务器来为服务器认证使用数字证书

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa
Device(config)# ip ssh server certificate profile
Device(ssh-server-cert-profile)# server
Device(ssh-server-cert-profile-server)# trustpoint sign trust1
Device(ssh-server-cert-profile-server)# exit
```

示例：配置 INOS SSH 服务器来为用户认证验证用户的数字证书

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm authentication publickey
Device(config)# ip ssh server algorithm publickey x509v3-ssh-rsa
Device(config)# ip ssh server certificate profile
Device(ssh-server-cert-profile)# user
Device(ssh-server-cert-profile-user)# trustpoint verify trust2
Device(ssh-server-cert-profile-user)# end
```

用于 SSH 认证的 X.509v3 证书的其他参考资料

相关文档

相关主题	文档名称
Inspur INOS 命令	Inspur INOS 主命令列表，所有版本
安全命令	• Inspur INOS 安全命令参考：命令 A 至 C • Inspur INOS 安全命令参考：命令 D 至 L • Inspur INOS 安全命令参考：命令 M 至 R • Inspur INOS 安全命令参考：命令 S 至 Z
SSH 认证	*SecureShellConfigurationGuide*中的“Secure Shell-Configuring User Authentication Methods”一章
公钥基础设施（PKI）信任点	*Public Key Infrastructure ConfigurationGuide*中的“Configuring and Managing a Inspur INOS

	Certificate Server for PKI Deployment”一章

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

用于 SSH 认证的 X.509v3 证书的特性信息

下面这个表格提供了这部分内容中描述的特性版本信息。这个表格中只列出了指定软件版本系列中，引入该特性的软件版本。除非另行说明，否则这个软件版本的后续版本也支持该特性。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

表 136：用于 SSH 认证的 X.509v3 证书的特性信息

特性名称	版本	特性信息
用于 SSH 认证的 X.509v3 证书	Inspur INOS XE 3.14S 版本	用于 SSH 认证的 X.509v3 证书特性在安全壳（SSH）服务器侧，在服务器和用户认证中使用 X.509v3 数字证书。 这个版本中引入或更改了以下命令：**ip ssh server algorithm hostkey**、**ip ssh server algorithm authentication** 和 **ip ssh server certificate profile**

配置安全套接字层 HTTP

查寻特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

安全套接字层（SSL）HTTP 的相关信息

安全 HTTP 服务器和客户端概述

在安全的 HTTP 连接上，去往和来自 HTTP 服务器的数据在通过 Internet 进行传输之前，会先进行加密。使用 SSL 加密的 HTTP 能够提供安全连接，从而实现一些功能，比如从 Web 浏览器对交换机进行配置。Inspur 安全 HTTP 服务器和安全 HTTP 客户端的实现利用了提供应用层加密的 SSL 3.0 版本。HTTP over SSL 缩写为 HTTPS；提供安全连接的 URL 以 https://而不是 http://开头。

注释： 1999 年 SSL 演变为传输层安全（TLS）协议，但有些环境中仍会使用 SSL。

HTTP 安全服务器（交换机）的主要作用是侦听指定端口（默认的 HTTPS 端口是 443）上的 HTTPS 请求，并将请求传递给 HTTP 1.1 Web 服务器。HTTP 1.1 服务器会负责处理这个请求，并把响应（页面）传递回到 HTTP 安全服务器，接着 HTTP 安全服务器会对原始请求做出响应。

HTTP 安全客户端（Web 浏览器）的主要作用是为 HTTPS 用户代理服务，做出 INOS 应用请求响应、为应用执行 HTTPS 用户代理服务，并将响应传递回给应用。

注释： 从 Inspur INOS 12.2 版本开始，用户能够为 HTTP 服务器关联 IPv6 ACL。在 Inspur INOS 12.2 版本之前，用户只能为安全 HTTP 服务器配置 IPv4 ACL。用户可以为安全 HTTP 服务器使用配置 CLI 命令，把已经配置好的 IPv6 和 IPv4 ACL 关联到 HTTP 服务器。

证书授权中心信任点

证书授权中心（CA）负责管理证书请求，并向参与网络行为的设备颁发证书。这些服务为参与网络行为的设备提供了安全密钥和证书的集中管理。指定的 CA 服务器称为信任点。

当用户在尝试连接时，HTTPS 服务器会通过向客户端颁发 X.509v3 证书的方式来为客户端提供安全连接，这个证书是从指定的 CA 信任点获取的，且已经经过了认证。接着客户端（通常是 Web 浏览器）会使用公钥来对这个证书进行认证。

为了确保 HTTP 连接的安全性，我们强烈建议用户配置一个 CA 信任点。如果用户没有为运行 HTTPS 服务器的设备配置 CA 信任点，则服务器会对自己进行认证，并生成所需的 RSA 密钥对。由于自我认证（自签名）证书无法提供足够的安全性，因此服务器所连接的客户端上会生成一个通知消息，表明这个证书是自我认证的，并让用户有机会接受或拒绝这个连接。这个选项在内部网络拓扑（比如测试）中很有用。

如果用户没有配置 CA 信任点的话，则在启用安全 HTTP 连接时，设备会自动生成为安全 HTTP 服务器（或客户端）使用的临时或永久自签名证书。

- 如果交换机上没有配置主机名和域名，则它会生成一个临时的自签名证书。在交换机重新启动后，所有临时的自签名证书都会丢失，并且交换机会生成一个新的临时自签名证书；
- 如果交换机上已配置了主机和域名，则它会生成一个永久自签名证书。在交换机重新启动后，或者禁用了安全 HTTP 服务器，以便下次重新启用安全 HTTP 连接时，这个证书会保持可用状态。

注释： 用户必须在每个设备上单独配置证书授权中心和信任点信息。从其他设备上复制的信息是无效的。

在注册新证书时，新的配置变更不会立即应用于 HTTPS 服务器，直到服务器重新启动为止。用户可以使用 CLI 或者通过物理的方式，重新启动服务器。在重新启动服务器时，交换机将会使用新证书。

如果交换机生成了自签名证书，特权 EXEC 命令 **show running-config** 的输出内容中会包含以下信息。以下为命令输出中的部分内容，只显示了自签名证书的命令。

```
Device# show running-config
Building configuration...
<output truncated>
crypto pki trustpoint TP-self-signed-3080755072
enrollment selfsigned
subject-name cn=INOS-Self-Signed-Certificate-
3080755072 revocation-check none
rsakeypair TP-self-signed-3080755072
!
!
crypto ca certificate chain TP-self-signed-3080755072
certificate self-signed 01
3082029F  30820208  A0030201  02020101  300D0609  2A864886  F70D0101
04050030
59312F30  2D060355  04031326  494F532D  53656C66  2D536967  6E65642D
43657274
69666963  6174652D  33303830  37353530  37323126  30240609  2A864886
F70D0109
02161743  45322D33  3535302D  31332E73  756D6D30  342D3335  3530301E
170D3933
30333031  30303030  35395A17  0D323030  31303130  30303030  305A3059
```

```
312F302D
<output truncated>
```

要想删除这个自签名证书，用户可以禁用安全 HTTP 服务器，并且输入全局配置命令 **no crypto pki trustpoint TP-self-signed-30890755072**。如果用户之后再次启用安全 HTTP 服务器，交换机就会生成一个新的自签名证书。

注释： *TP-self-signed* 后面的数值取决于设备的序列号。

用户可以使用可选命令（**ip http secure-client-auth**）使 HTTPS 服务器能够从客户端请求 X.509v3 证书。认证客户端要比服务器的自身认证提供了更高的安全性。

更多有关认证授权中心的信息，用户可以查看*Inspur INOS Security Configuration Guide, Release 12.4*中的“Configuring Certification Authority Interoperability”一章。

加密套件

加密套件（CipherSuite）指定了在一个 SSL 连接上使用的加密算法和摘要算法。在与 HTTPS 服务器建立连接时，客户端 Web 浏览器会提供它所支持的加密套件列表，客户端和服务器会在它们都支持的列表中协商出最佳的加密算法来使用。举例来说，Netscape Communicator 4.76 能够支持使用 RSA 公钥加密的 U.S.安全性、MD2、MD5、RC2-CBC、RC4、DES-CBC 和 DES-EDE3-CBC。

为了实现尽可能好的加密措施，用户应该使用支持 128 比特加密算法的客户端浏览器，比如 Microsoft Internet Explorer V5.5（或更高版本）或 Netscape Communicator 4.76 版本（或更高版本）。SSL_RSA_WITH_DES_CBC_SHA 加密套件提供的安全性低于其他的加密套件，因为它不提供 128 比特加密。

使用更安全和更复杂的加密套件需要消耗更多的处理时间。这个列表定义了交换机所支持的加密套件，并按照路由器处理负载（速度），把它们按照从最快到最慢的顺序进行排列：

1. SSL_RSA_WITH_DES_CBC_SHA——在 RSA 密钥交换（RSA 公钥加密）中，为消息加密使用 DES-CBC 加密，为消息摘要使用 SHA；
2. SSL_RSA_WITH_NULL_SHA——在密钥交换中，为消息加密使用 NULL，为消息摘要使用 SHA（只用于 SSL 3.0）；
3. SSL_RSA_WITH_NULL_MD5——在密钥交换中，为消息加密使用 NULL，为消息摘要使用 MD5（只用于 SSL 3.0）；
4. SSL_RSA_WITH_RC4_128_MD5——在 RSA 密钥交换中，为消息加密使用 RC4 128 比特加密，为消息摘要使用 MD5；
5. SSL_RSA_WITH_RC4_128_SHA——在 RSA 密钥交换中，为消息加密使用 RC4 128 比特加密，为消息摘要使用 SHA；
6. SSL_RSA_WITH_3DES_EDE_CBC_SHA——在 RSA 密钥交换中，为消息加密使用 3DES 和 DES-EDE3-CBC 加密，为消息摘要使用 SHA；
7. SSL_RSA_WITH_AES_128_CBC_SHA——在 RSA 密钥交换中，为消息加密使用 AES 128 比特加密，为消息摘要使用 SHA（只适用于 SSL 3.0）；
8. SSL_RSA_WITH_AES_256_CBC_SHA——在 RSA 密钥交换中，为消息加密使用 AES 256 比特加密，为消息摘要使用 SHA（只适用于 SSL 3.0）；
9. SSL_RSA_WITH_DHE_AES_128_CBC_SHA——在 RSA 密钥交换中，为消息加密使用 AES 128 比特加密，为消息摘要使用 SHA（只适用于 SSL 3.0）；
10. SSL_RSA_WITH_DHE_AES_256_CBC_SHA——在 RSA 密钥交换中，为消息加密使用 AES 256 比特加密，为消息摘要使用 SHA（只适用于 SSL 3.0）

注释： 最新版本的 Chrome 浏览器不支持四个原始的加密套件，因此无法访问 Web GUI 和用户门户。
RSA（与指定的加密和摘要算法组合相结合）同时用于 SSL 连接上的密钥生成和认证。这种用法与用户是否配置了 CA 信任点无关。

默认的 SSL 配置

启用了标准 HTTP 服务器
启用了 SSL
未配置 CA 信任点
未生成自签名证书

SSL 的配置指导

当用户在交换机集群中使用 SSL 时，SSL 会话会终结在集群指挥官（Commander）上。集群成员交换机上必须运行标准 HTTP。
在用户配置 CA 信任点之前，应该确保已经设置了系统时钟。如果用户没有设置时钟的话，则交换机会因为日期不正确而拒绝证书。
在交换机堆栈中，SSL 会话会终结在堆栈主用设备上。

如何配置安全 HTTP 服务器和客户端

配置 CA 信任点

为了保障 HTTP 连接的安全性，我们建议用户配置一个官方的 CA 信任点。CA 信任点要比自签名证书更安全。
从特权 EXEC 模式开始，用户可以按照以下步骤来配置 CA 信任点：
总步骤
1. configure terminal
2. hostname *hostname*
3. ip domain-name *domain-name*
4. crypto key generate rsa
5. crypto ca trustpoint *name*
6. enrollment url *url*
7. enrollment http-proxy *host-name port-number*
8. crl query *url*
9. primary *name*
10. exit
11. crypto ca authentication *name*
12. crypto ca enroll *name*
13. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** 示例： `Device# configure terminal`	进入全局配置模式
步骤 2	**hostname** *hostname* 示例： `Device(config)# hostname your_hostname`	指定交换机的主机名（只有当用户之前没有配置过主机名时才需要配置）。使用安全密钥和证书需要有主机名
步骤 3	**ip domain-name** *domain-name* 示例： `Device(config)# ip domain-name your_domain`	指定交换机的 IP 域名（只有当用户之前没有配置过 IP 域名时才需要配置）。使用安全密钥和证书需要有域名
步骤 4	**crypto key generate rsa** 示例： `Device(config)# crypto key generate rsa`	（可选）生成一个 RSA 密钥对。交换机上先要拥有 RSA 密钥对，用户才能为交换机获取一个证书。RSA 密钥对是自动生成的。用户可以使用这条命令来按需生成密钥
步骤 5	**crypto ca trustpoint** *name* 示例： `Device(config)# crypto ca trustpoint your_trustpoint`	为 CA 信任点指定一个本地配置名称，并进入 CA 信任点配置模式
步骤 6	**enrollment url** *url* 示例： `Device(ca-trustpoint)# enrollment url http://your_server:80`	指定一个 URL，也就是交换机向其发送证书请求的 URL
步骤 7	**enrollment http-proxy** *host-name port-number* 示例： `Device(ca-trustpoint)# enrollment http-proxy your_host 49`	（可选）配置交换机通过 HTTP 代理服务器，从 CA 那里获得证书。 • 在 *host-name* 部分指定用来访问 CA 的代理服务器 • 在 *port-number* 部分指定用来访问 CA 的端口号
步骤 8	**crl query** *url* 示例： `Device(ca-trustpoint)# crl query ldap://your_host:49`	配置交换机来请求一个证书撤销列表（CRL），来确保对等体的证书未被撤销

步骤 9	**primary** *name* **示例：** `Device(ca-trustpoint)# primary your_trustpoint`	（可选）指定一个信任点，并将这个信任点用作处理 CA 请求的主用（默认）信任点。 • 在 *name* 部分指定用户刚配置的信任点
步骤 10	**exit** **示例：** `Device(ca-trustpoint)# exit`	退出 CA 信任点配置模式，并返回全局配置模式
步骤 11	**crypto ca authentication** *name* **示例：** `Device(config)# crypto ca authentication your_trustpoint`	通过获得 CA 的公钥来对 CA 进行认证。使用与步骤 5 相同的名称
步骤 12	**crypto ca enroll** *name* **示例：** `Device(config)# crypto ca enroll your_trustpoint`	从指定的 CA 信任点获取证书。这条命令会为每个 RSA 密钥对请求一个签名证书
步骤 13	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式

配置安全 HTTP 服务器

从特权 EXEC 模式开始，用户可以按照以下步骤来配置安全 HTTP 服务器。

在开始前

如果用户使用证书授权中心来提供认证服务，则应该在启用 HTTP 服务器之前，按照上述步骤在交换机上配置 CA 信任点。如果用户没有配置 CA 信任点，则在首次启用安全 HTTP 服务器时，交换机会生成自签名证书。在用户配置了服务器之后，用户可以有选择地应用适用于标准和安全 HTTP 服务器的选项（路径、要应用的访问列表、最大连接数量，或超时策略）。用户要想验证使用 Web 浏览器的安全 HTTP 连接，可以输入 https://*URL*，其中 URL 是服务器交换机的 IP 地址或主机名。如果用户配置了默认端口之外的端口，就还必须在 URL 后面指定端口号。举例来说：

注释： 不支持 AES256_SHA2。

`https://209.165.129:1026`

或者

`https://host.domain.com:1026`

用来指定访问列表（只适用于 IPv4 ACL）的现有命令 **ip http access-class access-list-number** 将被弃用。用户仍然可以使用这条命令来指定访问列表，来放行访问 HTTP 服务器的流量。现在用户可以使用两个新命令来指定 IPv4 和 IPv6 ACL。

命令 **ip http access-class ipv4** *access-list-name* | *access-list-number* 用来指定 IPv4 ACL，命令 **ip http access-class ipv6** *access-list-name* 用来指定 IPv6 ACL。我们建议用户使用新的 CLI 命令，避免收到警告消息。

在指定访问列表时，用户需要考虑以下考量因素：

- 如果用户在指定一个不存在的访问列表，配置会生效，同时用户会收到以下警告消息：
  `ACL being attached does not exist, please configure it`
- 如果用户使用命令 **ip http access-class** 为 HTTP 服务器指定访问列表，用户会看到以下警告消息：
  `This CLI will be deprecated soon, Please use new CLI ip http access-class ipv4/ipv6 <access-list-name>| <access-list-number>`
- 如果用户使用命令 **ip http access-class ipv4** *access-list-name* | *access-list-number* 或命令 **ip http access-class ipv6** *access-list-name* 时，已经使用命令 **ip http access-class** 配置了访问列表，用户就会看到以下警告消息：
  `Removing ip http access-class <access-list-number>`

命令 **ip http access-class** *access-list-number* 和命令 **ip http access-class ipv4** *access-list-name* | *access-list-number* 拥有相同的功能。每条命令会覆盖之前命令的配置。这两条命令的下列组合会对运行配置带来以下影响：

- 如果用户已经配置命令 **ip http access-class** *access-list-number*，之后再尝试配置 **ip http access-class ipv4** *access-list-number* 命令，那么命令 **ip http access-class** *access-list-number* 的配置会被移除，命令 **ip http access-class ipv4** *access-list-number* 的配置会被放入运行配置中；
- 如果用户已经配置了命令 **ip http access-class** *access-list-number*，之后再尝试配置 **ip http access-class ipv4** *access-list-name* 命令，那么命令 **ip http access-class** *access-list-number* 的配置会被移除，命令 **ip http access-class ipv4** *access-list-name* 的配置会被添加到运行配置中；
- 如果用户已经配置了命令 **ip http access-class ipv4** *access-list-number*，之后再尝试配置 **ip http access-class** *access-list-name* 命令，那么命令 **ip http access-class ipv4** *access-list-number* 的配置会被移除，命令 **ip http access-class** *access-list-name* 的配置会被添加到运行配置中；
- 如果用户已经配置了命令 **ip http access-class ipv4** *access-list-name*，之后再尝试配置 **ip http access-class** *access-list-number* 命令，那么命令 **ip http access-class ipv4** *access-list-name* 的配置会被移除，命令 **ip http access-class** *access-list-number* 的配置会被添加到运行配置中。

总步骤

1. show ip http server status

2. configure terminal

3. ip http secure-server

4. ip http secure-port *port-number*

5. ip http secure-ciphersuite {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}

6. ip http secure-client-auth

7. ip http secure-trustpoint *name*

8. ip http path *path-name*

9. ip http access-class { **ipv4** {*access-list-number* | *access-list-name*} | **ipv6** {*access-list-name*} }

10. ip http max-connections *value*

11. ip http timeout-policy idle *seconds* **life** *seconds* **requests** *value*

12. end

具体配置

	命令或操作	目的

步骤 1	**show ip http server status**
	
	示例：
	`Device# show ip http server status`
		（可选）显示 HTTP 服务器的状态，来确定软件中是否支持安全 HTTP 服务器特性。用户应该会在输出信息中看到以下输出内容：
		`HTTP secure server capability: Present`
		或者
		`HTTP secure server capability: Not present`

步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**ip http secure-server** **示例：** `Device(config)# ip http secure-server`	启用 HTTP 服务器（如果还未启用的话）。HTTPS 服务器默认是启用的
步骤 4	**ip http secure-port** *port-number* **示例：** `Device(config)# ip http secure-port 443`	（可选）指定 HTTPS 服务器使用的端口号。默认端口号是 443。有效选项是 443 或 1025 至 65535 之中的任意号码
步骤 5	**ip http secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]} **示例：** `Device(config)# ip http secure-ciphersuite rc4-128-md5`	（可选）指定在 HTTPS 连接上，为加密使用的加密套件（加密算法）。如果用户不想指定具体的加密套件，就应该允许服务器和客户端来协商出它们都支持的加密套件。这是默认设置
步骤 6	**ip http secure-client-auth** **示例：** `Device(config)# ip http secure-client-auth`	（可选）配置 HTTP 服务器在连接过程中，向客户端请求 X.509v3 证书用于认证。默认设置是客户端需要向服务器请求证书，但服务器无需向客户端请求证书用于认证
步骤 7	**ip http secure-trustpoint** *name* **示例：** `Device(config)# ip http`	指定用来获得 X.509v3 安全证书的 CA 信任点，并用来认证客户端认证连接。 **注释：** 使用这条命令的前提是用户已经按照之前的步骤配置了 CA 信任

	secure-trustpoint your_trustpoint	点
步骤 8	**ip http path** *path-name* **示例：** `Device(config)# ip http path /your_server:80`	（可选）为 HTML 文件设置一个基本 HTTP 路径。路径定义了本地系统上的 HTTP 服务器文件的位置（通常位于系统 flash 内存中）
步骤 9	**ip http access-class** { **ipv4** {*access-list-number* | *access-list-name*} | **ipv6** {*access-list-name*}} **示例：** `Device(config)# ip http access-class ipv4 4`	（可选）指定用来允许 HTTP 服务器访问的访问列表
步骤 10	**ip http max-connections** *value* **示例：** `Device(config)# ip http max-connections 4`	（可选）设置同时连接 HTTP 服务器的最大并发连接数量。我们建议设置不小于 10 的值。这可以保证 UI 功能正常运行
步骤 11	**ip http timeout-policy idle** *seconds* **life** *seconds* **requests** *value* **示例：** `Device(config)# ip http timeout-policy idle 120 life 240 requests 1`	（可选）指定 HTTP 服务器连接可以维持多长时间，并且定义以下情况： • **idle**——指定最大空闲时间周期，也就是没有收到数据，或者不能发送响应数据的时间段。取值范围是 1 至 600 秒。默认值为 180 秒钟（3 分钟） • **life**——指定连接建立后的最大时间周期。取值范围是 1 至 86400 秒（24 小时）。默认值为 180 秒钟 • **requests**——在持续连接上处理请求的最大数量。最大值为 86400，默认值为 1
步骤 12	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式

配置安全 HTTP 客户端

从特权 EXEC 模式开始，用户可以按照以下步骤来配置安全 HTTP 客户端：

在开始前

标准 HTTP 客户端和安全 HTTP 客户端总是启用的。证书授权中心需要为安全 HTTP 客户端提供证书。在配置以下命令的前提是用户已经在交换机上配置了 CA 信任点。如果用户没有配置 CA 信任点，并且远端 HTTPS 服务器请求客户端进行认证，那么与安全 HTTP 客户端之间的连接会失效。

总步骤

1. configure terminal

2. ip http client secure-trustpoint *name*

3. ip http client secure-ciphersuite {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]}

4. end

具体配置

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 2	**ip http client secure-trustpoint** *name* **示例：** `Device(config)# ip http client secure-trustpoint your_trustpoint`	（可选）指定远端 HTTP 服务器在请求客户端认证时使用的 CA 信任点。使用这条命令的前提是用户已经按照之前的步骤配置了 CA 信任点。当无需进行客户端认证，或者当用户已经配置了主用信任点时，这条命令是可选的
步骤 3	**ip http client secure-ciphersuite** {[**3des-ede-cbc-sha**] [**rc4-128-md5**] [**rc4-128-sha**] [**des-cbc-sha**]} **示例：** `Device(config)# ip http client secure-ciphersuite rc4-128-md5`	（可选）指定在 HTTPS 连接上，为加密使用的加密套件（加密算法）。如果用户不想指定具体的加密套件，就应该允许服务器和客户端来协商出它们都支持的加密套件。这是默认设置
步骤 4	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式

监控安全 HTTP 服务器和客户端状态

为了监控 SSL 安全服务器和客户端状态，用户可以使用以下表格中的特权 EXEC 命令。

表 137：显示 SSL 安全服务器和客户端状态的命令

命令	目的
show ip http client secure status	显示 HTTP 安全客户端的配置
show ip http server secure status	显示 HTTP 安全服务器的配置
show running-config	显示为安全 HTTP 连接生成的自签名证书

其他参考资料

相关主题

相关主题	文档名称
为会话感知类网络配置身份控制策略和身份服务模版	Session Aware Networking Configuration Guide, Inspur INOS XE Release 3SE (Inspur 6850 Switches) http://www.icntnetworks.com
配置 RADIUS、TACACS+、SSH、802.1X 和 AAA	Securing User Services Configuration Guide Library, Inspur INOS XE Release 3SE (Inspur 6850 Switches) http://www.icntnetworks.com

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

配置 IPv4ACL

查询特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置 IPv4 访问控制列表的先决条件

本节列出了在使用访问控制列表（ACL）配置网络安全性时的一些先决条件。

- 在运行 LAN 基本特性集的交换机上，不支持 VLAN map 特性。

配置 IPv4 访问控制列表的限制条件

通用网络安全

使用 ACL 配置网络安全性的限制条件如下所示：

- 并不是所有适用于编号 ACL 的命令都适用于命名 ACL。在接口上进行包过滤和路由过滤时，可以使用命名 ACL。实施 VLAN map 也可以使用命名 ACL；
- 标准 ACL 与扩展 ACL 不可以使用相同的名字；
- 尽管在命令行的帮助信息中能看到 **appletalk**，但在 MAC 访问列表配置模式中，配置 **deny** 和 **permit** 语句时不能把 **appletalk** 作为匹配条件；
- 在下游客户端策略中，不支持 ACL 通配符。

网络接口上的 IPv4 ACL

在网络接口上应用 IPv4 ACL 的限制条件如下所示：

- 当用户希望对接口实施访问控制时，既可以使用命名 ACL，也可以使用编号 ACL；
- 如果用户在一个二层接口上应用了 ACL，且该接口是某个 VLAN 中的成员，那么这个二

层（端口）ACL 优先于相应 VLAN 接口上应用的入向三层 ACL，也优先于这个 VLAN 上配置的 VLAN map；

- 如果用户在三层接口上应用了 ACL，但这个交换机并未启用路由功能，那么这个 ACL 仅过滤需要 CPU 处理的数据包，比如 SNMP、Telnet 或其他网页流量；
- 如果配置了 **preauth_ipv4_acl**ACL 用来过滤数据包，这个 ACL 会在身份验证后被清除；
- 用户在二层接口上应用 ACL 时，不必开启路由功能。

注释： 当三层接口上配置的访问控制列表拒绝了一个数据包时，路由器默认会发送 Internet 控制消息协议（ICMP）不可达消息。这些被访问控制列表拒绝了的数据包并不是在硬件中直接丢弃，而是交给交换机的 CPU 进行处理，因此设备会发送 ICMP 不可达消息。如果用户不希望交换机生成 ICMP 不可达消息，可以在使用 ACL 的同时实施接口配置命令 **no ipunreachables**，该命令能够禁用 ICMP 不可达消息。

二层接口上的 MAC ACL

用户在创建了一个 MAC ACL 后，就可以把它应用到二层接口上了，这种应用可以过滤进入该二层接口的非 IP 流量。当应用 MAC ACL 时，用户应该考虑如下的指导建议：

- 用户可以在一个二层接口上同时应用一个 IP 访问列表和一个 MAC 访问列表。IP 访问列表仅过滤 IP 数据包，但是 MAC 访问列表可以过滤所有非 IP 数据包；
- 一个二层接口上只能应用一个 MAC 访问列表。如果用户在一个已经配置了 MAC ACL 的二层接口上应用另一个新的 MAC 访问列表，那么这个新的 ACL 就会取代之前配置的那个 MAC ACL。

注释： 接口配置命令 **mac access-group** 仅在二层物理接口上有效。用户不能在 EtherChannel 端口上使用此命令。

IP 访问列表条目序号

- 这个特性不支持动态、自反、防火墙访问列表。

与 ACL 相关的网络安全信息

本章会介绍如何通过访问控制列表（ACL）在交换机上配置网络安全性，在一些命令和表格中，访问控制列表也被称为访问列表。

ACL 概述

实施数据包过滤有助于限制网络流量，同时还能够限制某些用户或设备使用网络的行为。ACL 能够过滤穿越路由器或交换机的流量，并允许或拒绝那些穿过指定接口或 VLAN 的数据包。ACL 会按顺序依次检查那些针对数据包设置的允许或拒绝的条件。当接口接收到数据包时，交换机会将数据包中的字段与访问列表中具体的条件进行比较，以判断是否转发此数据包。交换机会将数据包与访问列表中的条件逐一进行匹配。根据最先匹配的结果，交换机判断是否接收此数据包。因为一旦某个条件匹配成功，交换机就会停止检查，所以访问列表中

条件的先后顺序是至关重要的。如果访问列表中的所有条件都没有匹配成功，那么交换机就会拒绝这个数据包。如果访问列表中没有匹配的限制条件，那么交换机就会转发这个数据包，否则就会丢弃数据包。交换机可以使用 ACL 过滤它转发的所有数据包，也包括那些在 VLAN 中的数据包。

用户可以通过在路由器或三层交换机上配置访问列表，为其网络提供基本的安全性。如果用户不配置 ACL，那么通过用户交换机的所有数据包都可以被传递到网络的各个部分。用户可以使用 ACL 来控制不同的主机访问网络不同的部分，或是使用 ACL 来决定路由器接口转发哪些类型的流量、阻塞哪些类型的流量。比如，用户可以允许转发电子邮件的流量，但阻塞 Telnet 的流量。用户可以配置 ACL 来阻塞入向流量、出向流量或同时阻塞双向的流量。

访问控制条目

一个 ACL 中包含一个按序排列的访问控制条目（ACE）列表。每条 ACE 中都会指定 *permit* 或 *deny* 行为，以及一组条件，数据包必须满足这些条件才能匹配这条 ACE。*permit* 或 *deny* 的对象取决于这个 ACL 当时所在的配置环境。

支持的 ACL 类型

交换机支持 IP ACL 和以太网（MAC）ACL：

- IP ACL 能够过滤 IPv4 流量，其中包括 TCP、用户数据报协议（UDP）、Internet 组管理协议（IGMP），以及 Internet 控制消息协议（ICMP）；
- 以太网 ACL 能够过滤非 IP 流量。

交换机还支持服务质量（QoS）分类 ACL。

支持的 ACL

交换机支持使用以下三种 ACL，来过滤流量：

- 端口 ACL 负责对进入二层接口的流量实施访问控制。用户仅可以应用一个 IP 访问列表和一个 MAC 访问列表；
- 路由器 ACL 负责对 VLAN 间的路由流量实施访问控制，用户可以把它应用在三层接口的某个方向上（入向或出向）；
- VLAN ACL 或 VLAN map 负责对所有（桥接的和路由的）数据包实施访问控制。用户可以使用 VLAN map 过滤同一个 VLAN 中不同设备之间的流量。配置 VLAN map 可以为基于三层地址的 IPv4 提供访问控制。要想对那些不受支持的协议实施访问控制，用户可以凭借 MAC 地址使用以太网 ACE。当一个 VLAN 中应用了 VLAN map 之后，所有（桥接的和路由的）数据包在进入这个 VLAN 时，都会由 VLAN map 进行检查。数据包可以通过交换端口进入这个 VLAN，也可以通过路由端口被路由到这个 VLAN 中。

ACL 优先级

当用户在一台交换机上同时配置了 VLAN map、端口 ACL 和路由器 ACL 时，入向流量的过滤优先级从高到低依次是：端口 ACL、VLAN map、路由器 ACL，出向流量的过滤优先级从高到

低依次是：路由器 ACL、VLAN map、端口 ACL。
下列示例描述了一些简单的使用环境：

- 当用户在一台交换机上同时应用了入向端口 ACL 和入向 VLAN map 时，在那些应用了端口 ACL 的端口上，入站数据包会由端口 ACL 进行过滤。其他端口收到的入站数据包会由 VLAN map 进行过滤；
- 当用户在一个交换机虚拟接口（SVI）上同时应用了入向路由器 ACL 和入向端口 ACL 时，在那些应用了端口 ACL 的端口上，入站数据包会由端口 ACL 进行过滤。其他端口收到的入站路由 IP 数据包会由路由器 ACL 进行过滤。其他的数据包不会被过滤；
- 当用户在一个 SVI 接口上同时应用了出向路由器 ACL 和入向端口 ACL 时，在那些应用了端口 ACL 的端口上，入站数据包会由端口 ACL 进行过滤。出站路由 IP 数据包会由路由器 ACL 进行过滤。其他的数据包不会被过滤；
- 当用户在一个 SVI 接口上同时应用了 VLAN map、入向路由器 ACL 和入向端口 ACL 时，在那些应用了端口 ACL 的端口上，入站数据包只会由端口 ACL 进行过滤。其他端口收到的入站路由 IP 数据包会由 VLAN map 和路由器 ACL 同时过滤。其他的数据包仅会由 VLAN map 进行过滤；
- 当用户在一个 SVI 接口上同时应用了 VLAN map、出向路由器 ACL 和入向端口 ACL 时，在那些应用了端口 ACL 的端口上，入站数据包仅会由端口 ACL 进行过滤。出站路由 IP 数据包会同时由 VLAN map 和路由器 ACL 进行过滤。其他的数据包仅会由 VLAN map 进行过滤。

端口 ACL

端口 ACL 是一种应用在交换机二层接口上的 ACL。它只能用在物理接口上，不能用于 EtherChannel 接口。用户可以将端口 ACL 分别应用在接口的出方向和接口的入方向上。端口 ACL 能够支持下列三种访问列表：

- 标准 IP 访问列表，针对源地址进行过滤；
- 扩展 IP 访问列表，针对源地址、目的地址，以及可选协议类型信息进行过滤；
- 扩展 MAC 访问列表，针对源 MAC 地址、目的 MAC 地址，以及可选协议类型信息进行过滤。

交换机会检查接口上应用的 ACL，并根据数据包与 ACL 条目匹配的情况，来决定允许或是拒绝转发这个数据包。这样，ACL 就能够对整个网络或网络的某部分，实施访问控制了。
下面这个示例中介绍了，当所有工作站都属于同一个 VLAN 的时候，端口 ACL 是如何对网络实施访问控制的。用户在图 103 中这台交换机的二层接口上，应用了入向端口 ACL，使得主机 A 能够正常地访问人力资源网络，但主机 B 却无法访问这个网络。在本示例中，用户仅将端口 ACL 应用在了二层接口的入方向上。

图 103：使用 ACL 来控制网络中的流量

Human Resouces Network	人力资源 网络
Research & Development Network	研究和发展 网络
ACL denying traffic from Host B	ACL 拒绝从主机 B 发来的流量

and permitting traffic from Host A	允许从主机 A 发来的流量
Packet	数据包

当用户在一个 Trunk 端口上应用了端口 ACL 时，这个 ACL 会过滤该 Trunk 端口上所有 VLAN 的流量。

当用户在一个配置了语音 VLAN 的端口上应用了端口 ACL 时，这个 ACL 既可以过滤数据流量，也可以过滤语音 VLAN 的流量。

在端口 ACL 中，用户可以通过 IP 访问列表来过滤 IP 流量，通过 MAC 访问列表来过滤非 IP 流量。

用户可以在同一个二层接口上，同时应用一个 IP 访问列表和一个 MAC 访问列表，来同时过滤 IP 流量和非 IP 流量。

注释： 在一个二层接口上，用户只能应用一个 IP 访问列表和一个 MAC 访问列表。如果用户在一个已经配置了 IP 访问列表或 MAC 访问列表的二层接口上，应用另一个新的 IP 访问列表或 MAC 访问列表，那么这个新的 ACL 就会取代之前配置的那个 ACL。

路由器 ACL

用户可以分别在交换机虚拟接口（SVI；这是 VLAN 的三层接口）、三层物理接口，以及三层 EtherChannel 接口上应用路由器 ACL。用户可以在接口的某个方向上（入向或出向）应用路由器 ACL。在一个接口的每个方向上，用户只能应用一个路由器 ACL。

针对 IPv4 流量，交换机支持的访问列表如下所示：

- 标准 IP 访问列表，针对源地址进行匹配；
- 扩展 IP 访问列表，针对源地址、目的地址，以及可选的协议类型信息进行匹配。

与端口 ACL 一样，交换机会在查看 ACL 的同时，查看该接口上配置的特性。当数据包进入交换机接口的时候，交换机会检查该接口上配置的所有入向特性相关联的 ACL。那些被路由的数据包，在被转发至下一跳之前，交换机会检查与出接口上配置的所有出向特性相关联的 ACL。

根据数据包与 ACL 条目匹配的情况，ACL 来决定允许或是拒绝转发这个数据包，这样就能够对整个网络或网络的某部分，实施访问控制了。

VLAN Map

VLAN ACL 或 VLAN map 是一种用于控制 VLAN 内部网络流量的访问控制列表。用户可以把 VLAN map 应用在交换机或交换机堆栈中桥接的 VLAN 内部数据包上。VACL 能够严格地进行数据包的安全性过滤，并且可以将流量重定向到具体的物理接口上。VACL 是没有方向性的（入向或出向）。

所有非 IP 协议都是通过 MAC 地址和以太类型（EtherType）字段，使用 MAC VLAN map 进行访问控制的（IP 流量不是通过 MAC VLAN map 进行访问控制的）。用户只能对那些穿越交换机的数据包实施 VLAN map；用户不能对通过集线器相连的主机之间的流量，以及这台交换机直连的其他交换机的流量实施 VLAN map。

交换机会根据 VLAN map 中指定的动作，来决定允许或是拒绝转发这些数据包。

图 104 展示了一个应用示例，用户通过应用 VLAN map，不允许交换机转发来自 VLAN10 中主机 A 的特定流量。用户只能在一个 VLAN 上应用一个 VLAN map。

图 104：使用 VLAN Map 来控制流量

Host A	主机 A
Host B	主机 B
VLAN map denying specific type of traffic from Host A	VLAN map 拒绝了从主机 A 发来的指定类型流量
Packet	数据包

ACE 与分片和未分片流量

IP 数据包在穿越网络的时候可以进行分片处理。分片处理后，只有数据包的起始分片中会包含第 4 层信息，比如 TCP 或 UDP 的端口号、ICMP 类型和编码信息等。其他的所有分片中都不会包含上述信息。

有一些访问控制条目（ACE）不会检查四层信息，这样用户就可以把它们应用在数据包的所有分片上。而对于那些需要检查四层信息的 ACE，用户就不能直接把它们应用在 IP 数据包的非起始分片上了。当需要检查第 4 层信息的 ACE 遇到不包含四层信息的分片时，匹配规则会发生如下的改变：

- 那些检查分片中三层信息（包括协议类型，比如 TCP、UDP 等）的 ACEpermit 条目，对分片进行匹配时，并不检查该分片是否包含四层信息；
- 那些检查四层信息的 ACE 的 deny 条目，只会在分片中包含四层信息时，才会匹配这些分片。

ACE 与分片和未分片流量的示例

下列是配置在访问列表 102 中的命令，用户将访问列表 102 应用在三个数据包分片上：

```
Device(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Device(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Device(config)# access-list 102 permit tcp any host 10.1.1.2
Device(config)# access-list 102 deny tcp any any
```

注释： 在示例的第一个和第二个 ACE 中，目的地址后面的 *eq* 关键字表示要检查的 TCP 目的端口号。在这里，端口号分别由简单邮件传输协议（SMTP）和 Telnet 代替。

- 数据包 A 是一个 TCP 数据包，从主机 10.2.2.2 的 65000 端口发往主机 10.1.1.1 的 SMTP 端口。如果这个数据包经过了分片处理，那么第一个分片能够成功匹配上第一条 ACE（允许），因为第一个分片中包含了所有的四层信息，就像是一个完整的数据包一样。虽然其余的分片中并不包含 SMTP 端口信息，但是因为第一条 ACE 只检查这些分片的三层信息，所以这些分片也可以与第一条 ACE 成功匹配。本示例中匹配的信息是 TCP 数据包和目的地址 10.1.1.1；
- 数据包 B 是一个从主机 10.2.2.2 的 65001 端口发往主机 10.1.1.2 的 Telnet 端口的数据包。如果这个数据包经过了分片处理，那么第一个分片能够成功匹配上第二条 ACE（拒绝），因为第一个分片中包含了所有三层和四层的信息。因为其余的数据包分片中不包含四层信息，所以它们不会与第二条 ACE 相匹配，而是与第三条 ACE（允许）相匹配。因为第一个分片被拒绝，所有主机 10.1.1.2 无法把这些分片重新组合成数据包，所以数据包 B 被有效地拒绝了。但是，除了第一个分片之外的其余分片还是会被转发，这样不仅浪费了网络带宽资源，同时也会消耗主机 10.1.1.2 上的硬件资源来尝试重新组合这些

分片；

- 数据包 C 是一个从主机 10.2.2.2 的 65001 端口发往主机 10.1.1.3 的 FTP 端口的数据包。如果这个数据包经过了分片处理，那么第一个分片能够成功匹配上第四条 ACE（拒绝）。其余所有的分片也可以成功匹配上第四条 ACE，因为这条 ACE 不会检查四层信息，而这些分片中的三层信息又表明它们的目的主机是 10.1.1.3，与之前几条 ACE 允许条目中的目的地址无法匹配。

ACL 和交换机堆栈

交换机堆栈对 ACL 的支持与对单台交换机对 ACL 的支持是一样的。ACL 的配置信息会被传播到堆栈中的每一台交换机上。堆栈中的所有交换机，也包括主用交换机，都能够处理信息，以及管理它们的硬件功能。

主用交换机和 ACL 功能

主用交换机能够支持的 ACL 功能如下所示：

- 主用交换机可以处理 ACL 的配置，并将配置信息传播到堆栈中的每一台交换机；
- 主用交换机可以把 ACL 的配置信息发送给新加入到堆栈中的交换机；
- 如果由于某种原因（比如，硬件资源不足时），导致交换机必须通过软件来转发数据包，那么主用交换机仅在对这些数据包应用了 ACL 之后，才会转发它们；
- 主用交换机能够根据由自己处理过的 ACL 的配置信息，来配置自己的硬件功能。

堆栈成员和 ACL 功能

堆栈成员能够支持的 ACL 功能如下所示：

- 堆栈成员可以接收由主用交换机发来的 ACL 的配置信息，并根据这些信息来配置它们自己的硬件功能；
- 用户可以把一台堆栈成员配置为备用交换机，如果主用交换机发生了故障，这台备用交换机就会执行主用交换机的功能。

主用交换机故障和 ACL

主用交换机和备用交换机上都有 ACL 配置信息。当主用交换机发生故障时，备用交换机会接管主用交换机的角色。新的主用交换机会向所有堆栈成员发送 ACL 配置信息。

标准和扩展 IPv4 ACL

这一部分会介绍 IP ACL。

ACL 是一系列按顺序的，允许（permit）和拒绝（deny）条件的集合。交换机会把数据包与访问列表中的条件逐一进行匹配。根据最先匹配的结果，交换机会判断是接受还是拒绝这个数据包。因为一旦某个条件匹配成功，交换机就会停止检查，所以访问列表中匹配条件的先

后顺序是至关重要的。如果访问列表中的所有匹配条件都没有匹配成功，那么交换机就会拒绝这个数据包。

交换机操作系统支持的 ACL 或 IPv4 访问列表类型如下所示：

- 标准 IP 访问列表，针对源地址进行匹配；
- 扩展 IP 访问列表，针对源地址、目的地址，以及可选的协议类型信息进行匹配。

交换机 IPv4 ACL 不支持的特性

在交换机上配置 IPv4 ACL，与在其他 Inspur 交换机和路由器上配置 IPv4 ACL 的方法是一样的。

交换机无法支持下列与 ACL 相关的特性：

- 非 IP 协议的 ACL
- IP 审计
- 自反 ACL 和动态 ACL

访问列表编号

用户用来标记 ACL 的编号可以指明用户创建的访问列表的类型。

表 138 中列出了访问列表的编号，以及与编号相对应的访问列表的类型，并且注明了哪些类型是交换机所支持的。交换机能够支持 IPv4 标准访问列表和扩展访问列表，标准访问列表的编号是从 1 至 199 号，扩展访问列表的编号是从 1300 至 2699 号。

表 138：访问列表编号

访问列表编号	类型	是否支持
1-99	IP 标准访问列表	支持
100-199	IP 扩展访问列表	支持
200-299	协议类型代码访问列表	不支持
300-399	DECnet 访问列表	不支持
400-499	XNS 标准访问列表	不支持
500-599	XNS 扩展访问列表	不支持
600-699	AppleTalk 访问列表	不支持
700-799	48 比特 MAC 地址访问列表	不支持
800-899	IPX 标准访问列表	不支持
900-999	IPX 扩展访问列表	不支持
1000-1099	IPX SAP 访问列表	不支持
1100-1199	扩展的 48 比特 MAC 地址访问列表	不支持
1200-1299	IPX 汇总地址访问列表	不支持
1300-1399	IP 标准访问列表（延伸范围）	支持
2000-2699	IP 扩展访问列表（延伸范围）	支持

用户除了创建编号的标准和扩展 ACL 外，还可以通过使用相应的编号，来创建命名的标准和扩展 IP ACL。也就是说，标准 IP ACL 的名字可以是 1 至 99 中的任意数字；扩展 IP ACL 的名字可以是 100 至 199 中的任意数字。相比于编号的访问列表，命名的 ACL 的优势在于，用户可以在不删除整个 ACL 的情况下，单独删除其中的某些列表条目。

编号的标准 IPv4 ACL

请切记，当用户在创建一个 ACL 的时候，在这个 ACL 的结尾会默认自动生成一条隐藏的拒绝所有数据包的语句，也就是说，如果数据包不能与 ACL 中前面配置的条件匹配成功，那么，默认就会被拒绝。在配置标准访问列表条目的时候，如果用户在 IP 地址后面没有配置通配符掩码，那么 ACL 会默认把这个通配符掩码假定为 0.0.0.0。
交换机会自动重新排列标准访问列表中条目的顺序，这样做是为了让那些与 **host** 匹配的条目和使用*无所谓*掩码 0.0.0.0 的条目位于列表的顶端，使它们位于使用非零*无所谓*掩码的条目的前面。因此，在 **show** 命令的输出中，以及在配置文件中，ACE 并不会以用户输入的顺序显示出来。
创建完成后，用户可以将这个编号的标准 IPv4 ACL 应用在 VLAN、终端线路或接口上。

编号的扩展 IPv4 ACL

由于标准 ACL 只能根据源地址进行匹配，为了满足更精准的控制需求，用户可以使用扩展 ACL，因为扩展 ACL 是根据源地址、目的地址，以及可选的协议类型信息进行匹配操作的。
当用户创建了一个编号的扩展访问列表时，请切记，在这个 ACL 的结尾也会默认自动生成一条隐藏的，拒绝所有数据包的语句。此外，用户不能对编号的访问列表中的条目进行重新排序，也不能插入新的条目或单独移除已有条目。
交换机无法支持动态访问列表和自反访问列表，也无法支持基于服务类型（ToS）最低位的过滤操作。
当用户在 ACL 中配置某些协议的时候，需要为这些协议设置具体的参数和关键字。
用户可以定义一个扩展的 TCP、UDP、ICMP、IGMP 或其他 IP ACL。交换机能够支持下列的 IP 协议。
注释： ICMP echo-reply 消息不能被过滤，除此之外，所有的 ICMP 编码或类型都可以被过滤。
交换机支持的 IP 协议如下所示：

- 认证头部协议（**ahp**）
- 封装安全负载（**esp**）
- 增强型内部网关路由协议（**eigrp**）
- 通用路由封装（**gre**）
- Internet 控制消息协议（**icmp**）
- Internet 组管理协议（**igmp**）
- 所有的内部协议（**ip**）
- IP-in-IP 隧道协议（**ipinip**）
- 兼容 KA9Q NOS 的 IP-over-IP 隧道协议（**nos**）
- 开放式最短路径优先路由协议（**ospf**）
- 负载压缩协议（**pcp**）
- 协议无关多播（**pim**）
- 传输控制协议（**tcp**）
- 用户数据报协议（**udp**）

命名的 IPv4 ACL

用户可以通过字母和数字组成的字符串为 IPv4 ACL 命名，而不像之前介绍的那样为 ACL 进行编号。相比于编号的方式，用户使用命名的 ACL 可以在一台路由器上配置更多的 IPv4 访问列表。在配置的方式和命令的语法方面，命名的 ACL 也与编号的 ACL 不尽相同。此外，并不是所有应用于配置 IP 访问列表的命令，都可以适用于命名的访问列表。

注释： 用户也可以使用数字为标准或扩展 ACL 命名，所使用的数字范围要符合访问列表编号的规则。也就是说，如果用户希望使用数字为一个标准 IP ACL 命名的话，可以使用的数字范围就是 1 至 99。相比于编号的列表，命名的 ACL 的优势在于，用户可以单独地删除列表中的条目。

用户在配置命名的 ACL 之前，应该考虑如下的指导建议：

- 也可以使用编号的 ACL；
- 用户不能使用同一个名字来命名一个标准 ACL 和一个扩展 ACL。

ACL 日志

交换机的软件可以为用户提供，那些被标准 IP 访问列表允许或拒绝的数据包的日志消息。也就是说，任何与 ACL 成功匹配的数据包，交换机都会为这个数据包生成一条日志消息，并将该日志消息发送给 Console 接口。用户可以使用控制系统日志消息的命令 **logging concole**，来控制发往 Console 接口的日志消息的等级。

注释： 由于交换机的路由操作是由硬件实现的，而日志却是由软件产生的，因此如果有大量的数据包成功匹配上了包含 **log** 关键字的 *permit* 或 *deny* 的 ACE，有可能会出现软件处理速率无法跟上硬件处理速率的情况，那么此时就不是所有的数据包都具有日志消息了。

第一个与 ACL 成功匹配的数据包会立即触发 ACL 产生一个日志消息，软件会收集之后 5 分钟之内所有匹配成功的数据包的信息，以产生它们的日志消息。无论 ACL 允许还是拒绝数据包，日志消息中都会包含访问列表编号、数据包的源 IP 地址，以及 5 分钟内从这个源地址允许或拒绝的数据包数量。

注释： 如果有太多要处理的日志消息，或者如果在 1 秒钟之内有一个以上的日志消息要处理，日志记录工具可能就丢弃一些日志消息包。这种行为是为了防止路由器由于需要处理太多的日志记录数据包而崩溃。因此，用户不应该把日志记录工具作为审计工具，或者作为访问列表匹配数量的准确来源。

IP ACL 的硬件和软件处理

ACL 的处理是在硬件中执行的。如果硬件中到达了用来储存 ACL 配置的容量极限，那么指定接口上的所有数据包都会被丢弃。

注释： 如果由于交换机或堆叠成员上的资源不足，而无法在硬件中实施 ACL 配置，则只有交换机上接收到的指定 VLAN 中的流量才会受到影响。

在用户输入了特权 EXEC 命令 **show ip access-lists** 后，命令的输出信息中显示的匹配计数值不考虑在硬件中进行访问控制的数据包。用户可以使用特权 EXEC 命令来获取交换和路由数据包的一些基本硬件 ACL 统计信息。

VLAN map 的配置指导

VLAN map 是控制 VLAN 内部流量过滤的唯一方法。VLAN map 没有方向。用户如果要通过使用 VLAN map 来过滤特定方向的流量，就需要在 ACL 中指定具体的源或目标地址。如果对于一个类型的数据包（IP 或 MAC），在 VLAN map 中指定了这种类型数据包的匹配命令，则交换机会默认当数据包与 VLAN map 中的任何条目都不匹配时，丢弃该数据包。如果 VLAN map 中没有定义该类型数据包的匹配命令，则默认转发数据包。

用户可以在配置 VLAN map 是使用以下配置指导：

- 如果接口上没有配置 ACL 来拒绝流量，并且没有配置 VLAN map 的话，所有流量都会被放行；
- 每个 VLAN map 都由一系列条目构成。VLAN map 中的条目顺序至关重要。交换机接收到的数据包都会与 VLAN map 中的第 1 个条目进行匹配。如果相匹配，交换机就会对其应用 VLAN map 中这一部分下配置的行为。如果不相匹配，数据包会与 map 中的下一个条目进行匹配；
- 如果 VLAN map 中为某个类型的数据包（IP 或 MAC）配置了至少一条匹配命令，那么与这些匹配条件不相符的数据包默认都会被丢弃。如果 VLAN map 中没有某个类型的数据包匹配条目，那么默认交换机会转发这个类型的数据包；
- VLAN map 不支持日志消息；
- 当用户把一个 IP 访问列表或 MAC 访问列表应用在交换机的二层接口上，并且为这个端口所属的 VLAN 应用了 VLAN map，那么端口 ACL 的优先级会高于 VLAN map；
- 如果用户不能把 VLAN map 的配置应用在硬件中，那么指定 VLAN 中的数据包都会被丢弃。

VLAN map 和路由器 ACL

要想同时对桥接流量和路由流量应用访问控制，用户可以只使用 VLAN map，或者结合使用路由器 ACL 和 VLAN map。用户可以在路由 VLAN 接口上同时指定入向和出向的路由器 ACL，并且可以定义一个 VLAN map 来对桥接流量实施访问控制。

如果数据包流与 ACL 中的 VLAN map 拒绝语句相匹配，那么无论用户是否配置了路由器 ACL 配置，数据包流都会被拒绝。

注释： 当用户结合使用了路由器 ACL 和 VLAN map，当路由器 ACL 中需要生成日志消息的数据包，与 VLAN map 中的拒绝语句相匹配，那么这个数据包并不会被记录。

如果 VLAN map 中为某个类型的数据包（IP 或 MAC）配置了匹配命令，那么与这些匹配条件不相符的数据包默认都会被丢弃。如果 VLAN map 中没有配置匹配条目，并且没有指定任何行为，那么如果数据包不匹配任何 VLAN map 条目的话，交换机就会转发这个数据包。

VLAN map 和路由器 ACL 的配置指导

如果用户想要在相同的 VLAN 中配置路由器 ACL 和 VLAN map，可以使用以下配置指导。这些指导并不适用于用户想要把路由器 ACL 和 VLAN map 应用于不同 VLAN 的情况。

如果用户必须在同一个 VLAN 上同时配置路由器 ACL 和 VLAN map 的话，可以使用以下指导来实施路由器 ACL 和 VLAN map 配置：

- 用户可以在一个 VLAN 接口上，在每个方向上（入向/出向）只配置一个 VLAN map 和一个路由器 ACL；
- 如果可能的话，用户可以尝试为 ACL 中的所有条目都配置单独的行为，除了最后为其他类型应用的默认行为。也就是说，以下面两种格式之一来编写 ACL：
  permit… permit… permit… deny ip any any
  或者
  deny… deny… deny… permit ip any any
- 要想在一个 ACL 中定义多个行为（permit 或 deny），用户可以把每个行为类型结合起来，来减少条目数量；
- 用户要避免在 ACL 中包含第 4 层信息；添加这种信息会增加合并过程的难度。如果 ACL 中定义的是根据 IP 地址（源和目的）进行过滤，而不是根据完整的流（源 IP 地址、目的 IP 地址、协议和协议端口）进行过滤的话，会得到最好的合并结果；
  如果用户需要使用完整的流匹配模式，并且 ACL 中同时包含 IP ACE，以及携带四层信息的 TCP/UDP/ICMP CE 的话，用户要把四层 ACE 放到列表的最后。让列表优先基于 IP 地址进行过滤。

ACL 的时间范围

用户可以使用全局配置命令 **time-range**，基于一天中的时间和星期来有选择地应用扩展 ACL。首先，用户要定义一个时间范围名称，并设置时间和日期，或者设置具体的星期几。然后用户要在把它应用到 ACL 时输入指定的时间范围名称，以此来对访问列表的行为进行限制。用户可以使用时间范围来限定 ACL 中的 permit 或 deny 语句何时生效，举例来说，在指定时间周期中生效，或者在指定的星期几范围内生效。命名的和编号的扩展 ACL 任务表中可以调用关键字 **time-range** 和参数。
使用时间范围有以下好处：

- 用户可以对允许或拒绝一个用户访问资源的行为施加更多的控制，比如指定一个应用（通过 IP 地址/掩码对，以及端口号进行标识）；
- 用户可以控制日志消息的生成，可以使 ACL 条目只在指定的星期几对流量进行记录。这样一来，用户可以在高峰时段只简单地拒绝流量访问，而无需对生成的大量日志进行分析。

基于时间的访问列表会触发 CPU 的活动，因为新的访问列表配置必须与其他特性进行合并，结合后的配置会加载到硬件内存中。出于这个考虑，用户应该注意不要让多个访问列表连续生效（每个访问列表的生效时间只间隔短短几分钟）。
注释： 时间范围的工作依赖于交换机系统的时钟；因此用户需要设置一个可靠的时钟源。我们建议用户使用网络时间协议（NTP）来同步交换机时钟。

IPv4 ACL 接口的考量因素

当用户在一个三层接口（SVI 接口、三层 EtherChannel 接口或路由端口）上使用接口配置命令 **ip access-group** 时，接口上必须已经配置了 IP 地址。三层访问列表会过滤由 CPU 进行处理的三层路由出去的或接收到的数据包。它不会对 VLAN 内部的桥接数据包带来任何影响。对于出向 ACL 来说，在接收到数据包后，交换机会用 ACL 来检查这个数据包。如果 ACL 中允许这个数据包，交换机就会继续处理这个数据包。如果 ACL 中拒绝这个数据包，交换机就会

丢弃这个数据包。
在默认环境中，当数据包被丢弃时，入站接口会发送 ICMP 不可达消息，无论这个丢弃数据包的行为是由于入站接口上的 ACL 导致的，还是由于出站接口上的 ACL 导致的。每个入站接口每半秒钟只能发送一个 ICMP 不可达消息，但用户可以使用全局配置命令 **ip icmp rate-limit unreachable** 来改变这一限制。
当用户把一个还未定义的 ACL 应用在接口后，交换机会当作这个接口上还没有应用任何 ACL，并且会放行所有数据包。如果用户想要使用未定义的 ACL 来提供网络安全的话，要记得交换机的这种行为。

如何配置 ACL

配置 IPv4 ACL

用户可以按照以下步骤在交换机上配置 IP ACL。
总步骤
1. 通过指定访问列表编号或名称，以及指定访问条件，来创建一个 ACL；
2. 把这个 ACL 应用在接口或终端线路上。用户也可以在 VLAN map 中应用标准和扩展 IP ACL。

具体步骤

	命令或操作	目的
步骤 1	通过指定访问列表编号或名称，以及指定访问条件，来创建一个 ACL	
步骤 2	把这个 ACL 应用在接口或终端线路上。用户也可以在 VLAN map 中应用标准和扩展 IP ACL	

创建编号的标准 ACL

用户可以按照以下步骤来创建编号的标准 ACL。
总步骤
1. enable
2. configure terminal
3. access-list *access-list-number* {**deny** | **permit**} *source source-wildcard*]
4. end
5. show running-config
6. copy running-config startup-config
具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码

步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**access-list** *access-list-number* {**deny** | **permit**} *source source-wildcard*] **示例：** `Device(config)# access-list 2 deny your_host`	使用源地址和通配符掩码来定义一个标准的 IPv4 访问列表。 在 *access-list* 部分指定一个十进制数值，取值范围是 1 至 99，或 1300 至 1999。 输入 **deny** 和 **permit** 来指定当条件匹配时，拒绝或放行数据包。 在 *source* 部分指定源地址，也就是从指定网络或主机发出的这个数据包，用户可以指定以下信息： • 32 比特点分十进制数值； • 关键字 **any** 是缩写表达，表示 *source* 和 *source-wildcard* 分别是 0.0.0.0 255.255.255.255。用户无需输入完整的源和通配符掩码。 （可选）*source-wildcard* 会在源地址上应用通配符掩码比特。 **注释：** 用户只能够在三层接口上关联的 ACL 中应用日志功能
步骤 4	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 6	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

配置编号的扩展 ACL

用户可以按照以下步骤来创建编号的扩展 ACL。

总步骤

1. configure terminal

2. access-list *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** tos] [**fragments**] [**log** [**log-input**] [**time-range**

time-range-name] [**dscp** *dscp*]

3. access-list *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**established**] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*]

4. access-list *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]

5. access-list *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*]

6. access-list *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*]

7. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 2	**access-list** *access-list-number* {**deny** | **permit**} *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*] [**tos** tos] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] **示例：** `Device(config)#` **access-list 101 permit ip host 10.1.1.2 any precedence 0 tos 0 log**	定义一个扩展的 IPv4 访问列表和访问条件。 在 *access-list-number* 部分指定一个十进制数值，取值范围是 100 至 199，或 2000 至 2699。 输入 **deny** 和 **permit** 来指定当条件匹配时，拒绝或放行数据包。 在 *protocol* 部分输入一个 IP 协议的名称或编号：**ahp**、**eigrp**、**esp**、**gre**、**icmp**、**igmp**、**igrp**、**ip**、**ipinip**、**nos**、**ospf**、**pcp**、**pim**、**tcp** 或 **udp**，或者使用代表一个 IP 协议编号的 0 至 255 之间的整数值。要想匹配任何 Internet 协议（包括 ICMP、TCP 和 UDP），用户需要使用关键字 **ip**。 **注释：** 这个步骤中包含了大多数 IP 协议选项。有关 TCP、UDP、ECMP 和 IGMP 的特定参数，用户可参考以下步骤。 *source* 部分指定了发送数据包的网络号或主机号。 *source-wildcard* 部分指定了源的通配符掩码比特。 *destination* 部分指定了数据包发往的

		网络号或主机号。 *destination-wildcard* 部分指定了目的的通配符掩码比特。 在配置 *source*、*source-wildcard*、*destination* 和 *destination-wildcard* 时，用户可以指定以下信息： • 32 比特的点分十进制数值 • 使用关键字 **any** 来表示 0.0.0.0 255.255.255.255（任意主机） • 使用关键字 **host** 来表示单台主机 0.0.0.0 用户还可以在这条命令中配置以下关键字，这些关键字的含义如下所示： • **precedence——**输入用来匹配数据包的优先级，用户可以使用编号 0 至 7，也可以使用名称：**routine**（0）、**priority**（1）、**immediate**（2）、**flash**（3）、**flash-override**（4）、**critical**（5）、**internet**（6）、**network**（7）； • **fragments**——输入这个关键字来检查非初始分片； • **tos**——输入用来匹配数据包的服务类型级别，用户可以使用编号 0 至 15，也可以使用名称：**normal**（0）、**max-reliability**（2）、**max-throughput**（4）、**min-delay**（8）； • **log**——输入这个关键字来创建发送到 Console 的有关匹配数据包的消息，或者输入 **log-input** 在日志条目中包含入站接口； • **time-range**——指定时间范围名称； • **dscp**——输入数据包匹配的 DSCP 值，取值范围是 0 至 63，或者使用问号（?）来查看可用值列表。 **注释：** 如果用户输入了一个 **dscp** 值，就不能再输入 **tos** 或 **precedence** 值了。用户可以在不使用 **dscp** 值的情况下，同时使用 **tos** 和 **precedence**
步骤 3	**access-list** *access-list-number* {**deny** | **permit**} **tcp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**established**]	定义一个扩展 TCP 访问列表，并指定访问条件。 这条命令中的参数与扩展 IPv4 ACL 中描述的参数相同，除了以下内容：

	[**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] [*flag*] **示例：** `Device(config)# access-list 101 permit tcp any any eq 500`	（可选）输入 *operator* 和 *port* 来对比源端口（如果在 *source source-wildcard* 后面输入的话）或目的端口（如果在 *destination destination-wildcard* 后面输入的话）。可选的运算符包括 **eq**（等于）、**gt**（大于）、**lt**（小于）、**neq**（不等于）和 **range**（包含首尾数值的范围）。运算操作需要有一个对应的端口号（使用关键字 **range** 时需要配置两个端口号，中间以空格分隔）。 在 *port* 部分指定 TCP 端口的十进制数值（取值范围是 0 至 65535）或名称。在过滤 TCP 时，用户只能使用 TCP 端口号或名称。 用户可以配置其他可选关键字，其含义如下所示： • **established**——输入这个关键字来匹配已建立的连接。这个关键字的功能与匹配 **ack** 或 **rst** 标记的功能相同 • *flag*——输入以下标记来匹配指定的 TCP 头部比特：**ack**（确认）、**fin**（完成）、**psh**（推送）、**rst**（重置）、**syn**（同步）或 **urg**（紧急）
步骤 4	**access-list** *access-list-number* {**deny** | **permit**} **udp** *source source-wildcard* [*operator port*] *destination destination-wildcard* [*operator port*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] **示例：** `Device(config)# access-list 101 permit udp any any eq 100`	（可选）定义一个扩展的 UDP 访问列表，并指定访问条件。 UDP 参数与上一步骤中描述的 TCP 参数相同，除了[operator [port]]端口号或名称必须是 UDP 端口号或名称，并且关键字 **flag** 和 **established** 不适用于 UDP
步骤 5	**access-list** *access-list-number* {**deny** | **permit**} **icmp** *source source-wildcard destination destination-wildcard* [*icmp-type* | [[*icmp-type icmp-code*] | [*icmp-message*]] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**time-range** *time-range-name*] [**dscp** *dscp*]	定义一个扩展的 ICMP 访问列表，并指定访问条件。 ICMP 参数与扩展 IPv4 ACL 中大多数 IP 协议的参数相同，除了 ICMP 中还可以指定 ICMP 消息类型和代码参数。这些可选关键字的含义如下所示： • *tcmp-type*——输入这个参数来过滤 ICMP 消息类型，取值范围是 0

	示例： `Device(config)# access-list 101 permit icmp any any 200`	至 255 之间的数值 • *icmp-code*——输入这个参数来过滤 ICMP 数据包，并根据 ICMP 消息代码类型进行过滤，取值范围是 0 至 255 之间的数值 • *icmp-message*——输入这个参数来过滤 ICMP 数据包，并根据 ICMP 消息类型名称或 ICMP 消息类型和代码名称进行过滤
步骤 6	**access-list** *access-list-number* {**deny** | **permit**} **igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**tos** *tos*] [**fragments**] [**log** [**log-input**] [**time-range** *time-range-name*] [**dscp** *dscp*] **示例：** `Device(config)# access-list 101 permit igmp any any 14`	（可选）定义一个扩展的 IGMP 访问列表，并指定访问条件。 IGMP 参数与扩展 IPv4 ACL 中大多数 IP 协议的参数相同，同时还包括以下可选参数： *igmp-type*——为了匹配 IGMP 消息类型，用户可以输入 0 至 15 之间的数值，或者输入消息名称：**dvmrp**、**host-query**、**host-report**、**pim** 或 **trace**
步骤 7	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式

创建命名的标准 ACL

用户可以按照以下步骤来创建使用名称的标准 ACL：

总步骤

1. enable

2. configure terminal

3. ip access-list standard *name*

4. 使用以下命令之一：

• **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]

• **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**]

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：**	进入特权 EXEC 模式。在提示时输入密码

	Device> enable	
步骤 2	**configure terminal** **示例：** Device# configure terminal	进入全局配置模式
步骤 3	**ip access-list standard** *name* **示例：** Device(config)# **ip access-list standard 20**	使用名称定义一个标准 IPv4 访问列表，并进入 access-list 配置模式。 名称也可以是 1 至 99 之间的数值
步骤 4	使用以下命令之一： • **deny** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**] • **permit** {*source* [*source-wildcard*] | **host** *source* | **any**} [**log**] **示例：** Device(config-std-nacl)# **deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255** 或者 Device(config-std-nacl)# **permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0**	在 access-list 配置模式中，指定一个或多个条件，以及拒绝或允许行为，来决定转发数据包还是丢弃数据包。 • **host** *source*——表示的源和源通配符掩码为 *source* 0.0.0.0 • **any**——表示的源和源通配符掩码为 0.0.0.0 255.255.255.255
步骤 5	**end** **示例：** Device(config-std-nacl)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

创建命名的扩展 ACL

用户可以按照以下步骤来创建使用名称的扩展 ACL：

总步骤

1. enable

2. configure terminal

3. ip access-list extended *name*

4. {**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*]

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**ip access-list extended** *name* **示例：** `Device(config)# ip access-list extended 150`	使用名称定义一个扩展的 IPv4 访问列表，并进入 access-list 配置模式。 名称可以是 100 至 199 之间的数值
步骤 4	{**deny** | **permit**} *protocol* {*source* [*source-wildcard*] | **host** *source* | **any**} {*destination* [*destination-wildcard*] | host *destination* | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log**] [**time-range** *time-range-name*] **示例：** `Device(config-ext-nacl)# permit 0 any any`	在 access-list 配置模式中，指定允许或拒绝的条件。用户可以使用关键字 **log** 来获得访问列表的日志消息，其中包括违规行为。 • **host** *source*——表示的源和源通配符掩码为 *source* 0.0.0.0 • **host** *destination*——表示的目的和目的通配符掩码为 *destination* 0.0.0.0 • **any**——表示的源和源通配符掩码，或者目的和目的通配符掩码为 0.0.0.0 255.255.255.255
步骤 5	**end** **示例：** `Device(config-ext-nacl)# end`	返回特权 EXEC 模式
步骤 6	**show running-config**	检查用户输入的信息

	示例： `Device# show running-config`	
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

当用户在创建扩展 ACL 时，要记住在默认情况下，如果在到达 ACL 末尾之前都没有找到匹配条目，那么它就会与 ACL 末尾匹配所有数据包的隐含 deny 语句相匹配。对于标准 ACL 来说，如果用户在相关联的 IP 主机地址访问列表的定义中省略了掩码，则默认使用 0.0.0.0 为掩码。在创建 ACL 后，用户添加的所有条目都会被放置在列表的末尾。用户不能有选择地在 ACL 中的指定位置添加 ACL 条目。但用户可以使用 access-list 配置模式命令 **no permit** 和 **no deny** 从命名 ACL 中删除 ACL 条目。

能够有选择地从命名 ACL 中删除 ACL 条目，是用户使用命名 ACL 而不是编号 ACL 的一个理由。

接下来做什么？

在创建命名的 ACL 后，用户可以把它应用在接口或 VLAN 上。

为 ACL 配置时间范围

用户可以按照以下步骤来为 ACL 配置时间范围参数：

总步骤

1. enable

2. configure terminal

3. time-range *time-range-name*

4. 使用以下命令之一：

• **absolute** [**start** *time date*] [**end** *time date*]

• **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm*

• **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm*

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**time-range** *time-range-name*	为创建的时间范围指定一个有意义的

	示例： `Device(config)#` **`time-range workhours`**	名称（比如 *workhours*[工作时间]），并进入 time-range 配置模式。名称中不能包含空格或问号，并且必须以字母开头
步骤 4	使用以下命令之一： **absolute** [**start** *time date*] [**end** *time date*] **periodic** *day-of-the-week hh:mm to* [*day-of-the-week*] *hh:mm* **periodic** {**weekdays** | **weekend** | **daily**} *hh:mm to hh:mm* **示例：** `Device(config-time-range)#` **`absolute start 00:00 1 Jan 2006 end 23:59 1 Jan 2006`** 或者 `Device(config-time-range)#` **`periodic weekdays 8:00 to 12:00`**	指定何时使用配置的操作。 • 用户可以在时间范围中只使用一个 **absolute** 语句。如果用户配置了多个 absolute 语句，只有最后配置的会生效 • 用户可以输入多个 **periodic** 语句。举例来说，用户可以在不同的工作日和周末中配置不同的小时 用户可以参考示例配置
步骤 5	**end** **示例：** `Device(config-time-range)#` **`end`**	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** `Device#` **`show running-config`**	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** `Device#` **`copy running-config startup-config`**	（可选）把输入的命令保存到配置文件中

接下来做什么？

如果用户需要指定多个分别在不同时间运行的项目的话，就需要多次重复这个配置步骤。

在终端线路上应用 IPv4 ACL

用户可以使用编号 ACL 来控制一个或多个终端线路的访问行为。用户不能在线路上应用命名 ACL。用户必须对所有的虚拟终端线路设置相同的限制，因为用户可以尝试连接到其中任何一个线路。

用户可以按照以下步骤来限制虚拟终端线路和 ACL 中指定地址之间入站和出站的连接：

总步骤

1. enable

2. configure terminal

3. line [**console** | **vty**] *line-number*

4. access-class *access-list-number* {**in** | **out**}

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**line** [**console**	**vty**] *line-number* **示例：** `Device(config)# line console 0`	指明要配置的线路，并进入线路配置模式。 • **console**——指定 Console 终端线路。Console 端口是 DCE； • **vty**——指定用于远程 Console 访问的虚拟终端。 在指定了线路类型后，在 *line-number* 部分配置用户希望配置的第 1 个线路编号。取值范围是 0 至 16
步骤 4	**access-class** *access-list-number* {**in**	**out**} **示例：** `Device(config-line)# access-class 10 in`	限制指定的虚拟终端线路（进入一台设备）和访问列表中指定地址之间的入站和出站连接
步骤 5	**end** **示例：** `Device(config-line)# end`	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：**	（可选）把输入的命令保存到配置文件中

	Device# **copy running-config startup-config**	

在接口上应用 IPv4 ACL

这部分描述了如何在网络接口上应用 IPv4 ACL。
从特权 EXEC 模式开始，用户可以按照以下步骤来对接口实施访问控制：
总步骤
1. configure terminal
2. interface *interface-id*
3. ip access-group {*access-list-number* | *name*} {**in** | **out**}
4. end
5. show running-config
6. copy running-config startup-config
具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** Device# configure terminal	进入全局配置模式
步骤 2	**interface** *interface-id* **示例：** Device(config)# **interface gigabitethernet1/0/1**	指定用户想要配置的接口，并进入接口配置模式。 接口可以是二层接口（端口 ACL），或者三层接口（路由器 ACL）
步骤 3	**ip access-group** {*access-list-number*	*name*} {**in**	**out**} **示例：** Device(config-if)# **ip access-group 2 in**	对指定接口实施访问控制
步骤 4	**end** **示例：** Device(config-if)# **end**	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 6	**copy running-config startup-config** **示例：** Device# **copy running-config**	（可选）把输入的命令保存到配置文件中

	startup-config	

创建命名的 MAC 扩展 ACL

用户可以使用 MAC 地址和命名的 MAC 扩展 ACL，在 VLAN 或二层接口上实施非 IPv4 流量的过滤。这个配置过程与配置其他命名的扩展 ACL 类似。
用户可以按照以下步骤来创建命名的 MAC 扩展 ACL：

总步骤

1. enable

2. configure terminal

3. mac access-list extended *name*

4. {**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbINOS** | **vines-echo** | **vines-ip** | **xns-idp** | 0-*65535*] [**cos** *cos*]

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**mac access-list extended** *name* **示例：** `Device(config)# mac access-list extended mac1`	使用名称定义一个扩展 MAC 访问列表
步骤 4	{**deny** | **permit**} {**any** | **host** *source MAC address* | *source MAC address mask*} {**any** | **host** *destination MAC address* | *destination MAC address mask*} [*type mask* | **lsap** *lsap mask* | **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbINOS** | **vines-**	在扩展 MAC access-list 配置模式中，指定 **permit** 或 **deny** 任意源 MAC 地址、源 MAC 地址/掩码，或者指定 **host**（主机）源 MAC 地址和 **any**（任意）目的 MAC 地址、目的 MAC 地址/掩码，或者指定目的 MAC 地址。 （可选）用户也可以输入以下选项： • *type mask*——任意 Ethernet II 或 SNAP 封装类型的数据包

	echo | **vines-ip** | **xns-idp** | 0-*65535*] [**cos** *cos*] **示例：** **decnet-iv** 或者 Device(config-ext-macl)# **permit any any**	EtherType 编号，格式为十进制、十六进制，或八进制，也可以在进行匹配前在 EtherType 上应用可选的*无所谓*比特掩码 • **lsap** *lsap mask*——IEEE 802.2 封装类型的数据包 LSAP 编号，格式为十进制、十六进制，或八进制，还可选的使用*无所谓*比特掩码 • **aarp** | **amber** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbINOS** | **vines-echo** | **vines-ip** | **xns-idp**——非 IP 协议 • **cos** *cos*——IEEE 802.1Q 服务类别编号，取值范围是 0 至 7，用来设置优先级
步骤 5	**end** **示例：** Device(config-ext-macl)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** Device# **show running-config**	检查用户输入的信息
步骤 7	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

在二层接口上应用 MAC ACL

用户可以按照以下命令在二层接口上应用 MAC 访问列表来实施访问控制：

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. mac access-group {*name*} {**in** | **out** }

5. end

6. show mac access-group [**interface** *interface-id*]

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**interface** *interface-id* **示例：** `Device(config)#` **`interface gigabitethernet1/0/2`**	指定一个接口，并进入接口配置模式。这个接口必须是物理的二层接口（端口 ACL）
步骤 4	**mac access-group** {*name*} {**in** | **out** } **示例：** `Device(config-if)#` **`mac access-group mac1 in`**	通过使用 MAC 访问列表来对指定接口实施访问控制。 用户可以在出方向上和入方向上应用端口 ACL
步骤 5	**end** **示例：** `Device(config-if)#` **`end`**	返回特权 EXEC 模式
步骤 6	**show mac access-group** [**interface** *interface-id*] **示例：** `Device#` **`show mac access-group interface gigabitethernet1/0/2`**	显示应用在指定接口或所有二层接口上的 MAC 访问列表
步骤 7	**show running-config** **示例：** `Device#` **`show running-config`**	检查用户输入的信息
步骤 8	**copy running-config startup-config** **示例：** `Device#` **`copy running-config startup-config`**	（可选）把输入的命令保存到配置文件中

在接收到一个数据包后，交换机会使用入方向的 ACL 来检查这个数据包。如果 ACL 中允许数据包通过，交换机就会继续处理这个数据包。如果 ACL 中拒绝数据包，交换机就会丢弃这个

数据包。当用户在接口上应用了一个未定义的 ACL 后，交换机会当作这个接口上还没有应用任何 ACL，并且会放行所有数据包。如果用户想要使用未定义的 ACL 来提供网络安全的话，要记得交换机的这种行为。

配置 VLAN map

用户可以按照以下命令来创建 VLAN map 并把它应用在一个或多个 VLAN 上：

在开始前

用户需要先创建想要应用在 VLAN 上的标准或扩展 IPv4 ACL，或者命名的 MAC 扩展 ACL。

总步骤

1. vlan access-map *name* [**number**]

2. match {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]

3. 输入以下命令之一，来指定 IP 数据包或非 IP 数据包（只能以已知的 MAC 地址进行指定），并且使用一个或多个（标准或扩展）ACL 来匹配数据包：

• **action** { **forward**}

Device(config-access-map)# **action forward**

• **action** { **drop**}

Device(config-access-map)# **action drop**

4. vlan filter *mapname* **vlan-list** *list*

具体步骤

	命令或操作	目的
步骤 1	**vlan access-map** *name* [**number**] **示例：** `Device(config)# vlan access-map map_1 20`	创建一个 VLAN map，并指定一个名称和（可选）一个编号。编号是这个 map 中条目的序列号。 在用户使用相同的名称创建 VLAN map 时，条目的编号是以 10 递增的。在更改或删除 VLAN map 时，用户可以输入想要更改或删除的 map 条目编号。 VLAN map 中并不能指定 permit 或 deny 关键字。要想使用 VLAN map 拒绝一个数据包，用户需要创建一个 ACL 来进行数据包匹配，并设置丢弃行为。ACL 中的 permit 语句表示相匹配。ACL 中的 deny 语句表示不匹配。输入这条命令会进入 access-map 配置模式
步骤 2	**match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*] **示例：** `Device(config-access-map)# match ip address ip2`	（使用 IP 或 MAC 地址）通过一个或多个标准或扩展访问列表来匹配数据包。需要注意的是，数据包只会以正确的协议类型来匹配访问列表。用户需要使用标准或扩展 IP 访问列表来匹配 IP 数据包。用户需要使用命名的

		MAC 扩展访问列表来匹配非 IP 数据包。 **注释：** 如果用户配置 VLAN map 来匹配一类数据包（IP 或 MAC），并且 VLAN map 中的行为是丢弃，那么所有匹配这个类型的数据包都会被丢弃。如果 VLAN map 中没有配置匹配条件，并且配置了丢弃行为，那么所有 IP 和二层数据包都会被丢弃

步骤 3	输入以下命令之一，来指定 IP 数据包或非 IP 数据包（只能以已知的 MAC 地址进行指定），并且使用一个或多个（标准或扩展）ACL 来匹配数据包：
	• **action** {**forward**}
	`Device(config-access-map)# action forward`
	• **action** {**drop**}
	`Device(config-access-map)# action drop`
		为 map 条目设置行为

步骤 4	**vlan filter** *mapname* **vlan-list** *list* **示例：** `Device(config)# vlan filter map 1 vlan-list 20-22`	把 VLAN map 应用到一个或多个 VLAN ID。 *list* 可以是一个 VLAN ID（22）、一个连续的列表（10 - 22），或者多个 VLAN ID（12, 22, 30）。逗号和连字符前后的空格是可选的

创建一个 VLAN map

每个 VLAN map 都由一系列有序的条目组成。从特权 EXEC 模式开始，用户可以按照以下步骤来创建、添加或删除 VLAN map 条目：

总步骤

1. configure terminal

2. vlan access-map *name* [**number**]

3. match {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*]

4. action {**drop** | **forward**}

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：**	进入全局配置模式

	Device# configure terminal	
步骤 2	**vlan access-map** *name* [**number**] **示例：** Device(config)# **vlan access-map map_1 20**	创建一个 VLAN map，并指定一个名称和（可选）一个编号。编号是这个 map 中条目的序列号。 在用户使用相同的名称创建 VLAN map 时，条目的编号是以 10 递增的。在更改或删除 VLAN map 时，用户可以输入想要更改或删除的 map 条目编号。 VLAN map 中并不能指定 permit 或 deny 关键字。要想使用 VLAN map 拒绝一个数据包，用户需要创建一个 ACL 来进行数据包匹配，并设置丢弃行为。ACL 中的 permit 语句表示相匹配。ACL 中的 deny 语句表示不匹配。 输入这条命令会进入 access-map 配置模式
步骤 3	**match** {**ip** | **mac**} **address** {*name* | *number*} [*name* | *number*] **示例：** Device(config-access-map)# **match ip address ip2**	（使用 IP 或 MAC 地址）通过一个或多个标准或扩展访问列表来匹配数据包。需要注意的是，数据包只会以正确的协议类型来匹配访问列表。用户需要使用标准或扩展 IP 访问列表来匹配 IP 数据包。用户需要使用命名的 MAC 扩展访问列表来匹配非 IP 数据包。 **注释：** 如果用户配置 VLAN map 来匹配一类数据包（IP 或 MAC），并且 VLAN map 中的行为是丢弃，那么所有匹配这个类型的数据包都会被丢弃。如果 VLAN map 中没有配置匹配条件，并且配置了丢弃行为，那么所有 IP 和二层数据包都会被丢弃
步骤 4	**action** {**drop** | **forward**} **示例：** Device(config-access-map)# **action forward**	（可选）为 map 条目设置行为，默认行为是转发
步骤 5	**end** **示例：** Device(config-access-map)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config**	检查用户输入的信息

	示例： `Device# show running-config`	
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

在 VLAN 上应用 VLAN map

从特权 EXEC 模式开始，用户可以按照以下步骤在一个或多个 VLAN 上应用 VLAN map：

总步骤：

1. enable

2. configure terminal

3. vlan filter *mapname* **vlan-list** *list*

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**vlan filter** *mapname* **vlan-list** *list* **示例：** `Device(config)# vlan filter map 1 vlan-list 20-22`	在一个或多个 VLAN ID 上应用 VLAN map。 *list* 可以是一个 VLAN ID（22）、一个连续的列表（10 - 22），或者多个 VLAN ID（12, 22, 30）。逗号和连字符前后的空格是可选的
步骤 4	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 6	**copy running-config startup-config**	（可选）把输入的命令保存到配置文

	示例： Device# **copy running-config startup-config**	件中

监控 IPv4 ACL

用户可以通过查看交换机上配置的 ACL，以及查看应用在接口和 VLAN 上的 ACL，来监控 IPv4 ACL。

在用户使用接口配置命令 **ip access-group**，在二层或三层接口上应用 ACL 时，用户可以查看接口上的 access-group。用户也可以查看应用在二层接口上的 MAC ACL。用户可以使用下面这个表格中展示的特权 EXEC 命令。

表 139：显示 access-list 和 access-group 的命令

命令	**目的**
show access-lists [*number* | *name*]	显示一个或当前所有 IP 和 MAC 地址访问列表或一个指定访问列表（编号或命名）中的内容
show ip access-lists [*number* | *name*]	显示当前所有 IP 访问列表或指定 IP 访问列表（编号或命名）中的内容
show ip interface *interface-id*	显示一个接口的配置和状态。如果接口上启用了 IP 功能，并且用户使用接口配置命令 **ip access-group** 应用了 ACL，命令的输出内容中就会包含 access-group
show running-config [**interface** *interface-id*]	显示交换机或指定接口的配置文件内容，其中包括所有配置的 MAC 和 IP 访问列表，以及接口上应用的 access-group
show mac access-group [**interface** *interface-id*]	显示应用在所有二层接口或指定接口上的 MAC 访问列表。 二层接口

ACL 的配置示例

示例：在 ACL 中使用时间范围

这个示例展示了在用户配置了时间范围 *workhours*，并把 200 年 1 月 1 日设置为公司假期后，如何验证相关的配置内容。

```
Device# show time-range
time-range entry: new_year_day_2003 (inactive)
absolute start 00:00 01 January 2006 end 23:59 01 January 2006
time-range entry: workhours (inactive)
```

```
periodic weekdays 8:00 to 12:00
periodic weekdays 13:00 to 17:00
```

要想应用一个时间范围，用户需要在能够实施时间范围的扩展 ACL 中，输入时间范围名称。这个示例展示了如何创建和验证扩展访问列表 188 的信息，这个访问列表中拒绝了从任意源去往任意目的地的 TCP 流量，应用的时间范围是假期时间，但在工作时间允许所有 TCP 流量。

```
Device(config)# access-list 188 deny tcp any any time-range new_year_day_2006
Device(config)# access-list 188 permit tcp any any time-range workhours
Device(config)# end
Device# show access-lists
Extended IP access list 188
10 deny tcp any any time-range new_year_day_2006 (inactive)
20 permit tcp any any time-range workhours (inactive)
```

下面这个示例展示了使用命名的 ACL 来放行和拒绝相同的流量。

```
Device(config)# ip access-list extended deny_access
Device(config-ext-nacl)# deny tcp any any time-range new_year_day_2006
Device(config-ext-nacl)# exit
Device(config)# ip access-list extended may_access
Device(config-ext-nacl)# permit tcp any any time-range workhours
Device(config-ext-nacl)# end
Device# show ip access-lists
Extended IP access list lpip_default
10 permit ip any any
Extended IP access list deny_access
10 deny tcp any any time-range new_year_day_2006 (inactive)
Extended IP access list may_access
10 permit tcp any any time-range workhours (inactive)
```

示例：ACL 中包含的命令

用户可以使用 **remark** 关键字，在任意 IP 标准或扩展 ACL 中包含一些注释（备注）信息。备注信息能够让用户更容易理解和搜索 ACL。每个备注信息限制为 100 个字符。

用户可以在 permit 或 deny 语句的前后设置备注信息。用户应该总是在同样的位置设置备注信息，这样就能看得出来哪条备注是在描述哪条 permit 或 deny 语句了。举例来说，如果有些备注标记在 permit 或 deny 语句前，有些标记在语句后，就会让用户感到混乱。

要想在编号的 IP 标准或扩展 ACL 中包含备注信息，用户需要使用全局配置命令 **access-list** *access-list number* **remark** *remark*。要想移除备注，需要使用这条命令的 **no** 格式。

在这个示例中，用户放行了属于 Jones 的工作站，并拒绝了属于 Smith 的工作站：

```
Device(config)# access-list 1 remark Permit only Jones workstation through
Device(config)# access-list 1 permit 171.69.2.88
```

```
Device(config)# access-list 1 remark Do not allow Smith through
Device(config)# access-list 1 deny 171.69.3.13
```

对于命名 IP ACL 中的条目，用户需要使用 access-list 配置命令 **remark**。要想移除备注，需要使用这条命令的 **no** 格式。

在这个示例中，Jones 子网不允许使用出向 Telnet：

```
Device(config)# ip access-list extended telnetting
Device(config-ext-nacl)# remark Do not allow Jones subnet to telnet
out
Device(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

IPv4 ACL 配置示例

这部分提供了配置和应用 IPv4 ACL 的示例。配置 ACL 的具体信息，用户可以参考 *Inspur INOS Security Configuration Guide, Release 12.4*，以及 *Inspur INOS IP Configuration Guide, Release 12.4* 中“IP Addressing and Services”一章中的“Configuring IP Services”部分。

小型网络办公室中的 ACL

这一部分展示了一个小型网络办公室环境，其中路由端口 2 连接着服务器 A，服务器 A 上包含有效益信息和其他信息，所有雇员都可以访问。路由端口 1 连接着服务器 B，服务器 B 上包含保密的工资数据。所有用户都可以访问服务器 A，但服务器 B 是被限制访问的。

图 105：使用路由器 ACL 来控制流量

Server A Benefits	服务器 A 效益
Server B Payroll	服务器 B 工资
Port 2	端口 2
Port1	端口 1
Human Resources	人力资源
Accounting	审计

用户可以使用以下两种方式之一来应用路由器 ACL：

- 创建一个标准 ACL，过滤从端口 1 去往服务器的流量；
- 创建一个扩展 ACL，过滤从端口 1 进入的服务器流量。

示例：小型网络办公室中的 ACL

这个示例中使用了一个标准 ACL 来过滤从端口进入的服务器 B 流量，只允许审计部门的源地址 172.20.128.64 至 172.20.128.95。这个 ACL 用来匹配从指定源地址去往路由端口 1 的流量。

```
Device(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Device(config)# end
Device# show access-lists
```

```
Standard IP access list 6
10 permit 172.20.128.64, wildcard bits 0.0.0.31
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 6 out
```

下面这个示例中使用了一个扩展 ACL，来过滤从服务器 B 进入端口的流量，允许任意源地址（本例中是服务器 B）流量只能去往审计部门的目的地址 172.20.128.64 至 172.20.128.95。这个 ACL 用来匹配进入路由端口 1 的流量，并且只允许这些流量去往指定目的地。注意在配置扩展 ACL 时，用户必须在源和目的信息前输入协议（IP）信息。

```
Device(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Device(config)# end
Device# show access-lists
Extended IP access list 106
10 permit ip any 172.20.128.64 0.0.0.31
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 106 in
```

示例：编号的 ACL

在这个示例中，网络 36.0.0.0 上一个 A 类网络，它的第 2 个八位组用来表示子网；也就是说它的子网掩码是 255.255.0.0。网络地址 36.0.0.0 中的第 3 和第 4 八位组用来指定具体的主机。用户使用了访问列表 2，让交换机接受子网 48 上的 1 个地址，拒绝这个子网上的所有其他地址。列表中的最后一行显示出交换机要接受所有其他网络 36.0.0.0 的子网。用户把这个 ACL 应用在进入端口的数据包上。

```
Device(config)# access-list 2 permit 36.48.0.3
Device(config)# access-list 2 deny 36.48.0.0 0.0.255.255
Device(config)# access-list 2 permit 36.0.0.0 0.255.255.255
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group 2 in
```

示例：扩展 ACL

在这个示例中，第一行允许任意入站 TCP 连接，并且目的端口要大于 1023。第二行允许入站 TCP 连接，并且要去往主机 128.88.1.2 的简单邮件传输协议（SMTP）端口。第三行允许用于错误反馈的入站 ICMP 消息。

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Device(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Device(config)# access-list 102 permit icmp any any
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group 102 in
```

在这个示例中，假设用户的网络连接到 Internet，并且用户希望网络中的任意主机能够与

Internet 上的任意主机建立 TCP 连接。但是，用户不希望 IP 主机能够与自己网络中的主机建立 TCP 连接，除了指定邮件主机的邮件（SMTP）端口。

SMTP 在一端使用 TCP 端口 25，在另一端使用随机端口号。在连接建立后到断开前都会使用相同的端口号。从 Internet 进入的邮件数据包的目的端口号是 25。出向数据包的端口号正相反。因为网络的安全系统总是会接受端口 25 上的邮件连接，因此入站和出站服务是分别进行控制的。用户必须在出向接口上配置一个入站 ACL，并且在入向接口上配置一个出站 ACL。

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 102 in
```

在这个示例中的网络是一个 B 类网络，地址为 128.88.0.0，邮件主机地址为 128.88.1.2。用户只为 TCP 连接使用了 **established** 关键字，以此显示已建立的连接。当 TCP 数据包中设置了 ACK 或 RST 位，就认为数据包匹配，这表示数据包是属于一个已存在的连接。硬件号码 1 上的 GigabitEthernet 接口 1 就是连接去往 Internet 的路由器的接口。

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 established
Device(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 102 in
```

示例：命名的 ACL

创建命名的标准和扩展 ACL

这个示例中创建了一个标准 ACL，名称为 *Internet_filter*，并创建了一个扩展 ACL，名称为 *marketing_group*。*Internet_filter* ACL 放行了源地址为 1.2.3.4 的所有流量。

```
Device(config)# ip access-list standard Internet_filter
Device(config-ext-nacl)# permit 1.2.3.4
Device(config-ext-nacl)# exit
```

marketing_group ACL 允许去往目的地址和通配符掩码 171.69.0.0 0.0.255.255 的任意 TCP Telnet 流量，并拒绝所有其他 TCP 流量。ACL 允许 ICMP 流量，拒绝从任意源地址去往目的地址范围 171.69.0.0 至 172.69.255.255，且目的端口号小于 1024 的 UDP 流量，拒绝所有其他 IP 流量，并为匹配结果提供日志消息。

```
Device(config)# ip access-list extended marketing_group
Device(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Device(config-ext-nacl)# deny tcp any any
Device(config-ext-nacl)# permit icmp any any
Device(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Device(config-ext-nacl)# deny ip any any log
Device(config-ext-nacl)# exit
```

用户为三层端口的出站流量应用了 *Internet_filter* ACL，为三层端口的入站流量应用了 *marketing_group* ACL。

```
Device(config)# interface gigabitethernet3/0/2
Device(config-if)# no switchport
Device(config-if)# ip address 2.0.5.1 255.255.255.0
Device(config-if)# ip access-group Internet_filter out
Device(config-if)# ip access-group marketing_group in
```

从命名的 ACL 中删除指定的 ACE

这个示例展示了人如何从命名访问列表 *border-list* 中删除指定的 ACE：

```
Device(config)# ip access-list extended border-list
Device(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

示例：为 IP ACL 应用时间范围

用户在这个示例中拒绝了 HTTP 流量，执行时间为周一至周五，8:00 至 18:00。这个示例只有在周六和周日的 12:00 至 20:00 之间，才放行 UDP 流量。

```
Device(config)# time-range no-http
Device(config)# periodic weekdays 8:00 to 18:00
!
Device(config)# time-range udp-yes
Device(config)# periodic weekend 12:00 to 20:00
!
Device(config)# ip access-list extended strict
Device(config-ext-nacl)# deny tcp any any eq www time-range no-http
Device(config-ext-nacl)# permit udp any any time-range udp-yes
!
Device(config-ext-nacl)# exit
Device(config)# interface gigabitethernet2/0/1
Device(config-if)# ip access-group strict in
```

示例：配置备注 IP ACL 条目

在这个示例中用户配置了一个编号的 ACL，允许属于 Jones 的工作站的访问行为，并拒绝属于 Smith 的工作站的访问行为：

```
Device(config)# access-list 1 remark Permit only Jones workstation
through
Device(config)# access-list 1 permit 171.69.2.88
Device(config)# access-list 1 remark Do not allow Smith workstation
through
Device(config)# access-list 1 deny 171.69.3.13
```

这个示例中用户配置了一个编号的 ACL，其中拒绝 Winter 和 Smith 工作站使用浏览 Web：

```
Device(config)# access-list 100 remark Do not allow Winter to browse
the web
```

```
Device(config)# access-list 100 deny host 171.69.3.85 any eq www
Device(config)# access-list 100 remark Do not allow Smith to browse
the web
Device(config)# access-list 100 deny host 171.69.3.13 any eq www
```

在这个示例中用户配置了一个命名的 ACL，拒绝了 Jones 子网的访问行为：

```
Device(config)# ip access-list standard prevention
Device(config-std-nacl)# remark Do not allow Jones subnet through
Device(config-std-nacl)# deny 171.69.0.0 0.0.255.255
```

在这个示例中用户配置了一个命名的 ACL，拒绝了 Jones 子网使用出向 Telnet：

```
Device(config)# ip access-list extended telnetting
Device(config-ext-nacl)# remark Do not allow Jones subnet to telnet
out
Device(config-ext-nacl)# deny tcp 171.69.0.0 0.0.255.255 any eq
telnet
```

示例：ACL 日志

路由器 ACL 支持两种日志记录。**log** 关键字能够向 Console 发送与该条目匹配的数据包的信息性日志消息,；**log-input** 关键字会在日志条目中包含输入接口信息。

在这个示例中用户配置了一个命名的标准访问列表 *stan1*，拒绝了来自 10.1.1.0 0.0.0.255 的流量，允许来自所有其他源地址的流量，并在命令中包含了 **log** 关键字。

```
Device(config)# ip access-list standard stan1
Device(config-std-nacl)# deny 10.1.1.0 0.0.0.255 log
Device(config-std-nacl)# permit any log
Device(config-std-nacl)# exit
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group stan1 in
Device(config-if)# end
Device# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
Console logging: level debugging, 37 messages logged
Monitor logging: level debugging, 0 messages logged
Buffer logging: level debugging, 37 messages logged
File logging: disabled
Trap logging: level debugging, 39 message lines logged
Log Buffer (4096 bytes):
00:00:48: NTP: authentication delay calculation problems
<output truncated>
00:09:34:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
00:09:59:%SEC-6-IPACCESSLOGS:list stan1 denied 10.1.1.15 1 packet
00:10:11:%SEC-6-IPACCESSLOGS:list stan1 permitted 0.0.0.0 1 packet
```

在这个示例中用户配置了一个命名的扩展访问列表 *ext1*，允许从任意源去往 10.1.1.0 0.0.0.255 的 ICMP 数据包，并拒绝所有 UDP 数据包。

```
Device(config)# ip access-list extended ext1
```

```
Device(config-ext-nacl)# permit icmp any 10.1.1.0 0.0.0.255 log
Device(config-ext-nacl)# deny udp any any log
Device(config-std-nacl)# exit
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip access-group ext1 in
```

这个示例展示的是一个扩展 ACL 的日志消息：

```
01:24:23:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 ->
10.1.1.61 (0/0), 1
packet
01:25:14:%SEC-6-IPACCESSLOGDP:list ext1 permitted icmp 10.1.1.15 ->
10.1.1.61 (0/0), 7
packets
01:26:12:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) ->
255.255.255.255(0), 1 packet
01:31:33:%SEC-6-IPACCESSLOGP:list ext1 denied udp 0.0.0.0(0) ->
255.255.255.255(0), 8 packets
```

注意所有 IP ACL 条目的日志消息都是以%SEC-6-IPACCESSLOG 开头的，并且根据 ACL 的类别和匹配的访问条目，这些信息会有些许变化。

这个示例展示的是一个启用了 **log-input** 关键字的输出消息：

```
00:04:21:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp
10.1.1.10 (Vlan1 0001.42ef.a400)
->
10.1.1.61 (0/0), 1 packet
```

使用 **log** 关键字记录的相同数据包日志消息中不包含入站接口信息：

```
00:05:47:%SEC-6-IPACCESSLOGDP:list inputlog permitted icmp
10.1.1.10 -> 10.1.1.61 (0/0), 1
packet
```

ACL 和 VLAN map 的配置示例

示例：创建 ACL 和 VLAN map 来拒绝数据包

这个示例展示了如何创建一个 ACL 和一个 VLAN map，来拒绝数据包。在第一个 map 中，所有匹配 *ip1* ACL（TCP 数据包）的数据包都会被丢弃。用户首先创建 *ip1* ACL，在其中允许所有 TCP 数据包，并拒绝其他数据包。由于 VLAN map 中有一个匹配 IP 数据包的条目，默认行为是丢弃所有不匹配条件的 IP 数据包。

```
Device(config)# ip access-list extended ip1
Device(config-ext-nacl)# permit tcp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map_1 10
Device(config-access-map)# match ip address ip1
Device(config-access-map)# action drop
```

示例：创建 ACL 和 VLAN map 来允许数据包

这个示例展示了如何创建一个 VLAN map 来放行数据包。这个示例中使用了 ACL *ip2*，允许 UDP 数据包，并且所有匹配 *ip2* ACL 的数据包都会被转发。在这个 map 中，所有不匹配之前 ACL 的 IP 数据包（也就是那些既不是 TCP 数据包，也不是 UDP 数据包的数据包）都会被丢弃。

```
Device(config)# ip access-list extended ip2
Device(config-ext-nacl)# permit udp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map_1 20
Device(config-access-map)# match ip address ip2
Device(config-access-map)# action forward
```

示例：丢弃 IP 数据包和转发 MAC 数据包的默认行为

在这个示例中，VLAN map 中指定了丢弃 IP 数据包的默认行为，以及转发 MAC 数据包的默认行为。这个 VLAN map 与标准 ACL 101 和命名的扩展访问列表 **igmp-match** 和 **tcp-match** 结合使用，map 会执行以下行为：

- 转发所有 UDP 数据包
- 丢弃所有 IGMP 数据包
- 转发所有 TCP 数据包
- 丢弃所有其他 IP 数据包
- 转发所有非 IP 数据包

```
Device(config)# access-list 101 permit udp any any
Device(config)# ip access-list extended igmp-match
Device(config-ext-nacl)# permit igmp any any
Device(config-ext-nacl)# permit tcp any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map drop-ip-default 10
Device(config-access-map)# match ip address 101
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-ip-default 20
Device(config-access-map)# match ip address igmp-match
Device(config-access-map)# action drop
Device(config-access-map)# exit
Device(config)# vlan access-map drop-ip-default 30
Device(config-access-map)# match ip address tcp-match
Device(config-access-map)# action forward
```

示例：丢弃 MAC 数据包和转发 IP 数据包的默认行为

在这个示例中，VLAN map 中配置了丢弃 MAC 数据包的默认行为，以及转发 IP 数据包的默认行为。这个 VLAN map 与 MAC 扩展访问列表 **good-hosts** 和 **good-protocols** 结合使用，map 会执行以下行为：

- 转发来自主机 0000.0c00.0111 和 0000.0c00.0211 的 MAC 数据包
- 转发携带 decnet-iv 或 vines-ip 协议的 MAC 数据包
- 丢弃所有其他非 IP 数据包
- 转发所有 IP 数据包

示例：丢弃所有数据包的默认行为

在这个示例中，VLAN map 中配置了丢弃所有数据包（IP 和非 IP）的默认行为。这个 VLAN map 与示例 2 和 3 中的访问列表 **tcp-match** 和 **good-hosts** 结合使用，map 会执行以下行为：

- 转发所有 TCP 数据包
- 转发来自主机 0000.0c00.0111 和 0000.0c00.0211 的 MAC 数据包
- 丢弃所有其他 IP 数据包
- 丢弃所有其他 MAC 数据包

```
Device(config)# vlan access-map drop-all-default 10
Device(config-access-map)# match ip address tcp-match
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan access-map drop-all-default 20
Device(config-access-map)# match mac address good-hosts
Device(config-access-map)# action forward
```

在用户网络中使用 VLAN map 的配置示例

示例：配线柜的配置

在配线柜的配置中，交换机上可能没有启用路由功能。在这种配置中，交换机仍可以支持 VLAN map 和 QoS 分类 ACL。假设主机 X 和主机 Y 分别位于不同的 VLAN，并且分别连接到配线柜交换机 A 和 C。从主机 X 去往主机 Y 的流量最终会由交换机 B 进行路由，交换机 B 是启用了路由功能的三层交换机。从主机 X 去往主机 Y 的流量可以在流量进入交换机 A 时收到访问控制。

图 106：配线柜的配置

Switch B	交换机 B
Switch A	交换机 A
Switch C	交换机 C
VLAN map: Deny HTTP from X to Y	VLAN map：拒绝 HTTP 从 X 去往 Y

HTTP is dropped at entry point.	HTTP 是在进入位置被丢弃的
Host X	主机 X
Host Y	主机 Y
Packet	数据包

如果用户不希望交换机转发从主机 X 去往主机 Y 的 HTTP 流量，用户可以在交换机 A 上配置一个 VLAN map，丢弃从主机 X（IP 地址 10.1.1.32）去往主机 Y（IP 地址 10.1.1.34）的所有 HTTP 流量，并且不会把这些流量转发到交换机 B。

首先，用户需要定义 IP 访问列表 *http*，允许（匹配）HTTP 端口上的所有 TCP 流量。

```
Device(config)# ip access-list extended http
Device(config-ext-nacl)# permit tcp host 10.1.1.32 host 10.1.1.34
eq www
Device(config-ext-nacl)# exit
```

接下来，用户要创建 VLAN map *map2*，使它丢弃与访问列表 *http* 匹配的流量，并转发所有其他 IP 流量。

```
Device(config)# vlan access-map map2 10
Device(config-access-map)# match ip address http
Device(config-access-map)# action drop
Device(config-access-map)# exit
Device(config)# ip access-list extended match_all
Device(config-ext-nacl)# permit ip any any
Device(config-ext-nacl)# exit
Device(config)# vlan access-map map2 20
Device(config-access-map)# match ip address match_all
Device(config-access-map)# action forward
```

然后，把 VLAN map *map2* 应用到 VLAN 1。

```
Device(config)# vlan filter map2 vlan 1
```

示例：限制访问另一个 VLAN 上的服务器

用户可以丢弃访问另一个 VLAN 上服务器的流量。举例来说，VLAN 10 中的服务器 10.1.1.100 需要拒绝下列主机的访问：

- 应该拒绝 VLAN 20 中子网 10.1.2.0/8 中主机的访问；
- 应该拒绝 VLAN 10 中主机 10.1.1.4 和 10.1.1.8 的访问。

图 107：限制访问另一个 VLAN 上的服务器

Server (VLAN 10)	服务器（VLAN 10）
Host (VLAN 10)（共 2 处）	主机（VLAN 10）
Layer 3 switch	三层交换机
Subnet	子网
Host (VLAN 20)	主机（VLAN 20）

示例：拒绝访问另一个 VLAN 上的服务器

这个示例展示了用户如何通过创建 VLAN map SERVER1_ACL，来拒绝访问另一个 VLAN 上服务器的流量，这个 VLAN map 拒绝了去往子网 10.1.2.0/8、主机 10.1.1.4 和主机 10.1.1.8 的流量，并允许其他 IP 流量。最后一步用户把 map SERVER1_ACL 应用到 VLAN 10。
用户先定义 IP ACL 来匹配正确的数据包。

```
Device(config)# ip access-list extended SERVER1_ACL
Device(config-ext-nacl))# permit ip 10.1.2.0 0.0.0.255 host
10.1.1.100
Device(config-ext-nacl))# permit ip host 10.1.1.4 host 10.1.1.100
Device(config-ext-nacl))# permit ip host 10.1.1.8 host 10.1.1.100
Device(config-ext-nacl))# exit
```

用户定义一个 VLAN map，在其中调用这个 ACL，丢弃匹配 SERVER1_ACL 的 IP 数据包，转发不匹配这个 ACL 的 IP 数据包。

```
Device(config)# vlan access-map SERVER1_MAP
Device(config-access-map)# match ip address SERVER1_ACL
Device(config-access-map)# action drop
Device(config)# vlan access-map SERVER1_MAP 20
Device(config-access-map)# action forward
Device(config-access-map)# exit
```

用户把 VLAN map 应用到 VLAN 10。

```
Device(config)# vlan filter SERVER1_MAP vlan-list 10
```

在 VLAN 上应用路由器 ACL 和 VLAN map 的配置示例

在这部分展示的示例中，用户在 VLAN 上应用了路由器 ACL 和 VLAN map，应用于交换、桥接、路由和组播数据包。虽然在下面的展示中，数据包都被转发到了它们的目的地，但每次数据包路径上应用了 VLAN map 或 ACL 时，数据包也可能被丢弃，而不是被转发。

示例：ACL 和被交换的数据包

在这个示例中，展示了如何为在 VLAN 内部进行交换的数据包应用 ACL。在 VLAN 内部进行交换的数据包不会进行路由，或者由回退-桥接进行转发，因此它会受到入站 VLAN 的 VLAN map 影响。

图 108：在被交换的数据包上应用 ACL

Input router ACL	入站 路由器 ACL
Output router ACL	出站 路由器 ACL
Frame	数据帧

Host A	主机 A
Host C	主机 C
Packet	数据包

示例：ACL 和被桥接的数据包

这个示例中展示了如何在回退-桥接的数据包上应用 ACL。对于桥接的数据包，只能在入站 VLAN 上应用二层 ACL。只有非 IP、非 ARP 数据包可以进行回退-桥接。

图 109：在桥接的数据包上应用 ACL

Host A	主机 A
Frame	数据帧
Host B	主机 B
Fallback bridge	回退桥接
Packet	数据包

示例：ACL 和被路由的数据包

这个示例展示了如何在被路由的数据包上应用 ACL。ACL 是按照以下顺序应用的：

1. 入站 VLAN 的 VLAN map
2. 入站路由器 ACL
3. 出站路由器 ACL
4. 出站 VLAN 的 VLAN map

图 110：在被路由的数据包上应用 ACL

Input router ACL	入站 路由器 ACL
Output router ACL	出站 路由器 ACL
Frame	数据帧
Host A	主机 A
Host B	主机 B
Routing function	路由功能
Packet	数据包

示例：ACL 和组播数据包

这个示例中展示了如何在通过 IP 组播进行复制的数据包上应用 ACL。被路由的组播数据包有两种应用过滤的方式：一个用来匹配入站 VLAN 中其他端口的目的地，另一个用来匹配其他 VLAN（数据包被路由的 VLAN）中的每个目的地。数据包可能会被路由到多个出现出向 VLAN，在这种情况下，用户可以为每个目的 VLAN 应用不同的路由器出向 ACL 和 VLAN map。最终结果是有些出向 VLAN 可能会放行数据包，而在其他出向 VLAN 中则拒绝。数据包的副

本会被转发到被放行的那些目的地。但是，如果入向 VLAN map 丢弃了数据包，则没有目的地能够接收到这个数据包的副本。

图 111：在组播数据包上应用 ACL

Input router ACL	入站 路由器 ACL
Output router ACL	出站 路由器 ACL
Frame	数据帧
Host A	主机 A
Host B	主机 B
Routing function	路由功能
Host C	主机 C
Packet	数据包

其他参考资料

相关主题

相关主题	**文档名称**
IPv4 访问控制列表主题	Securing the Data Plane Configuration Guide Library, Inspur INOS XE Release 3SE (Inspur 6850 Switches) http://www.icntnetworks.com

错误消息解码器

描述	**链接**
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

技术助手

描述	**链接**
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。	http://www.icntnetworks.com

在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	

配置 IPv6 ACL

查询特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

IPv6 ACL 概述

用户可以通过创建 IPv6 访问控制列表（ACL），并把它们应用到接口，来过滤 IP 版本 6（IPv6）流量，就像创建并应用 IP 版本 4（IPv4）命名 ACL。在运行 IP Base 和 LAN Base 特性集的交换机上，用户也可以创建并应用入向路由器 ACL，来过滤三层管理流量。

交换机支持以下三种类型的 IPv6 ACL：

- 用户可以为三层接口上的出向或入向流量应用 IPv6 路由器 ACL，这个接口可以是路由端口、交换机虚拟接口（SVI），或三层 EtherChannel。IPv6 路由器 ACL 只应用在被路由的 IPv6 数据包上；
- 入向二层接口上支持 IPv6 端口 ACL。IPv6 端口 ACL 能够应用于进入接口的所有 IPv6 数据包上；
- VLAN ACL 或 VLAN map 能够对一个 VLAN 内部的所有数据包执行访问控制。用户可以使用 VLAN map 来过滤同一个 VLAN 不同设备之间的流量。ACL VLAN map 可以应用在二层 VLAN 上。VLAN map 中可以基于三层地址，对 IPv6 执行访问控制。用户要理解通过 MAC 地址，使用以太网 ACE 执行访问控制的协议。在把 VLAN map 应用到一个 VLAN 后，所有进入这个 VLAN 的数据包都会由 VLAN map 进行检查；

用户可以在一个接口上同时应用 IPv4 和 IPv6 ACL。与 IPv4 ACL 一样，IPv6 端口 ACL 的优先级

高于路由器 ACL。

交换机堆栈和 IPv6 ACL

主用交换机能够在硬件中支持 IPv6 ACL，并把这个 IPv6 ACL 分发到堆栈成员上。
如果备用交换机接管并成为了主用交换机，它会把 ACL 的配置分发到所有堆栈成员。成员交换机会与新的主用交换机分发的配置进行同步，并把不需要的条目移除。
在用户修改 ACL、在接口上关联或解除关联 ACL 时，主用交换机会把变更分发给所有堆栈成员。

ACL 优先级

当用户在同一台交换机上配置 VLAN map、端口 ACL 和路由器 ACL 时，对于入向流量来说，这些访问限制按照过滤优先级的从高到底排列为：端口 ACL、VLAN map，然后是路由器 ACL。
对于出向流量来说，过滤优先级排列为：路由器 ACL、VLAN map，然后是端口 ACL。
以下示例描述了简单的使用情况：

- 但用户同时应用了入向端口 ACL 和 VLAN map 时，应用了端口 ACL 的端口在接收到入站数据包时，这些入站数据包会由端口 ACL 进行过滤。其他数据包由 VLAN map 进行过滤；
- 当一个交换机虚拟接口（SVI）上同时应用了入向路由器 ACL 和入向端口 ACL 时，应用了端口 ACL 的端口在接收到入站数据包时，这些入站数据包是由端口 ACL 进行过滤的。其他端口上接收到的入站路由 IP 数据包是由路由器 ACL 进行过滤的。其他数据包不执行过滤；
- 当一个 SVI 接口上同时应用了出向路由器 ACL 和入向端口 ACL 时，应用了端口 ACL 的端口在接收到入站数据包时，这些入站数据包是由端口 ACL 进行过滤的。出向路由 IP 数据包是由路由器 ACL 进行过滤的。其他数据包不执行过滤；
- 当一个 SVI 接口上同时应用了 VLAN map、入向路由器 ACL 和入向端口 ACL 时，应用了端口 ACL 的端口在接收到入站数据包时，这些入站数据包只会由端口 ACL 进行过滤。其他端口上收到的入站路由 IP 数据包会同时由 VLAN map 和路由器 ACL 进行过滤。其他数据包只会由 VLAN map 进行过滤；
- 当一个 SVI 接口上同时应用了 VLAN map、出向路由器 ACL 和出向端口 ACL 时，应用了端口 ACL 的端口在接收到入站数据包时，这些入站数据包只会由端口 ACL 进行过滤。其他端口上收到的入站路由 IP 数据包会同时由 VLAN map 和路由器 ACL 进行过滤。其他数据包只会由 VLAN map 进行过滤。

VLAN map

用户可以使用 VLAN ACL 或 VLAN map 来对一个 VLAN 内部的网络流量实施控制。用户可以为一台交换机或一个交换机堆栈上，一个 VLAN 内部桥接的所有数据包应用 VLAN map。VACL 专门用来执行安全数据包过滤行为，并且用于把流量重定向到指定的物理接口。VACL 在定义时不涉及方向性（入向或出向）。
所有非 IP 协议都是使用 MAC VLAN map 进行访问控制的，需要匹配 MAC 地址和以太类型（IP 流量不会由 MAC VLAN map 来提供访问控制）。用户可以在穿越交换机的数据包上实施

VLAN map；用户不能在通过集线器或另一台交换机，连接到本地交换机的主机与本地交换机之间的流量上应用 VLAN map。

在使用 VLAN map 时，交换机会根据 map 中指定的行为，来允许或拒绝数据包的转发行为。下图中展示了如何应用 VLAN map 来执行过滤的情景，用户要拒绝来自 VLAN 10 中主机 A 的指定类型流量。用户只可以在一个 VLAN 上应用一个 VLAN map。

图 112：使用 VLAN map 来实施流量控制

Host A	主机 A
Host B	主机 B
VLAN map denying specific type of traffic from Host A	VLAN map 拒绝了来自主机 A 的特性类型流量
Packet	数据包

与其他特性和交换机的互操作

- 如果用户配置了一个 IPv6 路由器 ACL 来拒绝数据包，则数据包不能被路由。这个数据包的一个副本会被发送到 Internet 控制消息协议（ICMP）队列，以便为这个数据帧生成 ICMP 不可达消息；
- 如果由于端口 ACL 而丢弃了一个桥接的数据帧，则这个数据包无法被桥接；
- 用户可以在一台交换机或交换机堆栈上同时创建 IPv4 和 IPv6 ACL，用户也可以在同一个接口上同时应用 IPv4 和 IPv6 ACL。每个 ACL 必须有唯一的名称；如果用户尝试使用已经配置过的名称，就会看到一条错误消息。

  用户需要使用不同的命令来创建 IPv4 和 IPv6 ACL，以及在相同的二层或三层接口上关联 IPv4 或 IPv6 ACL。如果用户在关联一个 ACL 时使用了错误的命令（比如使用 IPv4 命令来关联 IPv6 ACL），用户就会看到一条错误消息；
- 用户不能使用 MAC ACL 来过滤 IPv6 数据帧。MAC ACL 只能用来过滤非 IP 数据帧；
- 如果设备的硬件内存满了，那么数据包会在接口上就被丢弃，并且设备会记录一条 Unload 错误消息。

配置 IPv6 ACL 的限制条件

在 IPv4 中，用户可以配置编号的标准和扩展 IP ACL、命名 IP ACL 和 MAC ACL。IPv6 只支持命名 ACL。

交换机上能够支持大多数 Inspur INOS 所支持的 IPv6 ACL，除了以下注意事项：

- 交换机不支持使用这些关键字进行匹配：**routing header** 和 **undetermined-transport**；
- 交换机不支持自反 ACL（使用关键字 **reflect**）；
- 交换机不能在 IPv6 数据帧上应用基于 MAC 的 ACL；
- 用户不能在二层 EtherChannel 上应用 IPv6 端口 ACL；
- 在配置 ACL 时，对于用户在 ACL 中输入的关键字并没有限制，除非设备平台不支持。当用户在需要执行硬件转发的接口（物理端口或 SVI 接口）上应用 ACL 时，交换机会检查并确认这个接口是否能够支持 ACL。如果不支持，关联 ACL 的行为被拒绝；
- 如果用户在接口上应用了一个 ACL，并且尝试在一个访问控制条目（ACE）中使用接口不支持的关键字，那么交换机不会允许用户在当前关联到接口上的这个 ACL 中添加这

条 ACE。

交换机上的 IPv6 ACL 具有以下特征：

- 支持分片的数据帧（与 IPv4 中 **fragments** 关键字相同）；
- IPv4 中支持的状态统计信息，在 IPv6 ACL 中也同样支持；
- 如果交换机的硬件空间不足，与 ACL 相关联的数据包会在接口上被丢弃；
- 路由器 ACL 能够使用日志功能，端口 ACL 不能使用日志功能；
- 交换机支持使用全范围的前缀长度进行 IPv6 地址匹配。

默认的 IPv6 ACL 配置

以下为默认的 IPv6 ACL 配置：

```
Switch# show access-lists preauth_ipv6_acl
IPv6 access list preauth_ipv6_acl (per-user)
permit udp any any eq domain sequence 10
permit tcp any any eq domain sequence 20
permit icmp any any nd-ns sequence 30
permit icmp any any nd-na sequence 40
permit icmp any any router-solicitation sequence 50
permit icmp any any router-advertisement sequence 60
permit icmp any any redirect sequence 70
permit udp any eq 547 any eq 546 sequence 80
permit udp any eq 546 any eq 547 sequence 90
deny ipv6 any any sequence 100
```

配置 IPv6 ACL

用户可以按照以下步骤来过滤 IPv6 流量：

总步骤

1. enable

2. configure terminal

3. [**no**]{**ipv6 access-list** *list-name*| **client permit-control-packets**| **log-update threshold**| **role-based** *list-name*}

4. [**no**]{**deny** | **permit**} protocol {*source-ipv6-prefix/|prefix-length*|**any threshold**| **host** *source-ipv6-address*} [operator [*port-number*]] { *destination-ipv6-prefix/ prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]][**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]

5. {**deny** | **permit**} **tcp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [**operator** [**port-number**]] {*destination-ipv6- prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [**ack**] [**dscp** *value*] [**established**] [**fin**] [**log**] [**log-input**] [**neq** {**port** | protocol}] [**psh**] [**range** {**port** | protocol}] [**rst**] [**routing**] [**sequence** *value*] [**syn**] [**time-range** *name*] [**urg**]

6. {**deny** | **permit**} **udp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [operator

[*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [**dscp** *value*] [**log**] [**log-input**] [**neq** {*port* | *protocol*}] [**range** {*port* | *protocol*}] [**routing**] [**sequence** *value*] [**time-range** *name*]]

7. {**deny** | **permit**} **icmp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [operator [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [*icmp-type* [*icmp-code*] | icmp-message] [**dscp** *value*] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]

8. end

9. show ipv6 access-list

10. show running-config

11. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	[**no**]{**ipv6 access-list** *list-name*| **client permit-control-packets**| **log-update threshold**| **role-based** *list-name*} **示例：** `Device(config)# ipv6 access-list example_acl_list`	定义一个 IPv6 ACL 名称，并进入 IPv6 访问列表配置模式
步骤 4	[**no**]{**deny**	**permit**} protocol {*source-ipv6-prefix/|prefix-length*|**any threshold**| **host** *source-ipv6-address*} [operator [*port-number*]] { *destination-ipv6-prefix/ prefix-length*	**any**	**host** *destination-ipv6-address*} [operator [*port-number*]][**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]	输入 **deny** 和 **permit** 来指定当条件匹配时，拒绝或放行数据包。 • 在 *protocol* 部分输入一个 IP 协议的名称或编号：**ahp**、**esp**、**icmp**、**ipv6**、**pcp**、**stcp**、**tcp** 或 **udp**，或者使用代表一个 IPv6 协议编号的 0 至 255 之间的整数值； • 在 *source-ipv6-prefix/prefix-length* 或 *destination-ipv6-prefix/prefix-length* 部分指定要为其设置拒绝和允许条件的源和目的 IPv6 网络或网络类，使用十六进制的 16 比特数值，以冒号分隔（详见 RFC 2373 文档）； • 使用 **any** 这个缩写来表示 IPv6 前

		缀::/0； • 在 **host** *source-ipv6-address* 或 *destination-ipv6-address* 部分输入要为其设置拒绝和允许条件的源或目的 IPv6 主机地址，使用十六进制的 16 比特数值，以冒号分隔； • （可选）在 **operator** 部分指定要进行对比运算的指定协议源或目的端口。可选的运算符包括 **lt**（小于）、**gt**（大于）、**eq**（等于）、**neq**（不等于）和 **range**。 如果 **operator** 后面跟着 *source-ipv6-prefix/prefix-length* 参数，则它必须匹配源端口。如果 **operator** 后面跟着 *destination-ipv6-prefix/prefix-length* 参数，则它必须匹配目的端口； • （可选）在 **port-number** 部分指定 TCP 或 UDP 端口的十进制数值，取值范围是 0 至 65535。在过滤 TCP 时，用户只可以使用 TCP 端口名称。在过滤 UDP 时，用户只可以使用 UDP 端口名称； • （可选）输入 **dscp** 值来匹配差分服务代码点值，来匹配 IPv6 数据包头部中的流量类别字段。取值范围是 0 至 63； • （可选）输入 **fragments** 来检查非初始分片。只有当 protocol 为 ipv6 时，用户才会看到这个关键字； • （可选）输入 **log** 关键字来创建发送到 Console 的有关匹配数据包的消息。输入 **log-input** 在日志条目中包含入站接口。只有路由器 ACL 可以支持日志功能； • （可选）输入 **routing** 来指定被路由的 IPv6 数据包； • （可选）输入 **sequence** *value* 来为访问列表条目指定序列号。可选范围是 1 至 4294967295； • （可选）输入 **time-range** 名称来指定要用来拒绝或允许数据包的时间范围

步骤 5	{**deny** | **permit**} **tcp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [**operator** [**port-number**]] {*destination-ipv6- prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [**ack**] [**dscp** *value*] [**established**] [**fin**] [**log**] [**log-input**] [**neq** {**port** | protocol}] [**psh**] [**range** {**port** | protocol}] [**rst**] [**routing**] [**sequence** *value*] [**syn**] [**time-range** *name*] [**urg**]	（可选）定义一个 TCP 访问列表和访问条件。 输入 **tcp** 来指定传输控制协议。这条命令中的参数与步骤3a中指定的参数相同，此外用户还可以配置以下可选参数： • **ack**——设置确认比特； • **established**——已建立的连接。如果TCP数据段中设置了ACK或RST比特，则认为匹配； • **fin**——设置完成比特；不会再发送任何数据； • **neq** {*port* | protocol}——只匹配携带非指定端口号的数据包； • **psh**——设置推送功能比特； • **range** {*port* | protocol}——只匹配携带端口号范围的数据包； • **rst**——设置重置比特； • **syn**——设置同步比特； • **urg**——设置紧急指针比特
步骤 6	{**deny** | **permit**} **udp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [operator [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [**dscp** *value*] [**log**] [**log-input**] [**neq** {*port* | *protocol*}] [**range** {*port* | *protocol*}] [**routing**] [**sequence** *value*] [**time-range** *name*]]	（可选）定义一个 UDP 访问列表和访问条件。 输入 **udp** 来指定用户数据报协议。UDP参数与 TCP 部分描述的参数相同，除了[operator [*port*]]部分的端口号或名称必须是 UDP 端口好或名称，已建立参数不适用于 UDP
步骤 7	{**deny** | **permit**} **icmp** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [operator [*port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [operator [*port-number*]] [*icmp-type* [*icmp-code*] | icmp-message] [**dscp** *value*] [**log**] [**log-input**] [**routing**] [**sequence** *value*] [**time-range** *name*]	（可选）定义一个 ICMP 访问列表和访问条件。 输入 **icmp** 来指定 Internet 控制消息协议。ICMP 参数与步骤 1 中描述的大多数 IP 协议相同，除了用户还可以设置 ICMP 消息类型和代码参数。用户可以使用的可选参数有如下含义： • *icmp-type*——输入这个参数来按照 ICMP 消息类型进行过滤，取值范围为 0 至 255 之间的数值； • *icmp-code*——输入这个参数来按照 ICMP 消息代码类型进行 ICMP 数据包过滤，取值范围为 0 至 255 之间的数值； • *icmp-message*——输入这个参数

		来按照 ICMP 消息类型名称或 ICMP 消息类型和代码名称进行 ICMP 数据包过滤。要想查看完整的 ICMP 消息类型名称和代码名称，用户可以使用?或查看这个版本的命令参考
步骤 8	**end**	返回特权 EXEC 模式
步骤 9	**show ipv6 access-list**	检查访问列表的配置
步骤 10	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 11	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

接下来做什么？

在接口上关联 IPv6 ACL

在接口上关联 IPv6 ACL

用户可以在三层接口的出方向上或入方向上应用 ACL，或者在二层接口的入方向上应用 ACL。用户也可以只在三层接口的入向管理流量上应用 ACL。

用户可以按照以下步骤，控制接口上的流量访问行为：

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. no switchport

5. ipv6 address *ipv6-address*

6. ipv6 traffic-filter *access-list-name* {**in** | **out**}

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal**	进入全局配置模式

	示例： `Device# configure terminal`	
步骤 3	**interface** *interface-id*	指定用户想要应用访问列表的二层接口（端口 ACL）或三层接口（路由器 ACL），并进入接口配置模式
步骤 4	**no switchport**	如果要应用路由器 ACL，用户需要使用这条命令把接口从二层模式（默认）更改为三层模式
步骤 5	**ipv6 address** *ipv6-address*	在三层接口（路由器 ACL）上配置 IPv6 地址
步骤 6	**ipv6 traffic-filter** *access-list-name* {**in** | **out**}	为接口上的入站或出站流量应用访问列表。 **注释：**
步骤 7	**end** **示例：** `Device(config-if)# end`	返回特权 EXEC 模式
步骤 8	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 9	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

配置 VLAN map

用户可以按照以下步骤，来创建一个 VLAN map，并将其应用在一个或多个 VLAN 上。

在开始前

用户需要创建想要应用在 VLAN 上的 IPv6 ACL。

总步骤

1. enable

2. configure terminal

3. vlan access-map *name* [**number**]

4. match {**ip** | **ipv6** | **mac**} **address** {*name* | *number*} [*name* | *number*]

5. 输入以下命令之一，来指定 IP 数据包或非 IP 数据包（只能以已知的 MAC 地址进行指定），并且使用一个或多个（标准或扩展）ACL 来匹配数据包：

• **action** { **forward**}

Device(config-access-map)# action forward

• **action** { **drop**}

Device(config-access-map)# action drop

6. vlan filter *mapname* **vlan-list** *list*

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** Device> enable	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** Device# configure terminal	进入全局配置模式
步骤 3	**vlan access-map** *name* [**number**] **示例：** Device(config)# **vlan access-map map_1 20**	创建一个 VLAN map，并指定一个名称和（可选）一个编号。编号是这个 map 中条目的序列号。 在用户使用相同的名称创建 VLAN map 时，条目的编号是以 10 递增的。在更改或删除 VLAN map 时，用户可以输入想要更改或删除的 map 条目编号。 VLAN map 中并不能指定 permit 或 deny 关键字。要想使用 VLAN map 拒绝一个数据包，用户需要创建一个 ACL 来进行数据包匹配，并设置丢弃行为。ACL 中的 permit 语句表示相匹配。ACL 中的 deny 语句表示不匹配。 输入这条命令会进入 access-map 配置模式
步骤 4	**match** {**ip** | **ipv6** | **mac**} **address** {*name* | *number*} [*name* | *number*] **示例：** Device(config-access-map)# **match ipv6 address ip_net**	通过一个或多个访问列表来匹配数据包。需要注意的是，数据包只会以正确的协议类型来匹配访问列表。用户需要使用 IP 访问列表来匹配 IP 数据包。用户需要使用命名的 MAC 访问列表来匹配非 IP 数据包。 **注释：** 如果用户配置 VLAN map 来匹配一类数据包（IP 或 MAC），并且 VLAN map 中的行为是丢弃，那么所有匹配这个类型的数据包都会被丢弃。如果 VLAN map 中没有配置匹配条件，并且配置了丢弃行为，那么所有 IP 和二层数据包都会被丢弃
步骤 5	输入以下命令之一，来指定 IP 数据包或	为 map 条目设置行为

	非 IP 数据包（只能以已知的 MAC 地址进行指定），并且使用一个或多个（标准或扩展）ACL 来匹配数据包：
	• **action {forward}**
	`Device(config-access-map)# action forward`
	• **action {drop}**
	`Device(config-access-map)# action drop`	

步骤 6	**vlan filter** *mapname* **vlan-list** *list* **示例：** `Device(config)# vlan filter map 1 vlan-list 20-22`	把 VLAN map 应用到一个或多个 VLAN ID。 *list* 可以是一个 VLAN ID（22）、一个连续的列表（10 - 22），或者多个 VLAN ID（12, 22, 30）。逗号和连字符前后的空格是可选的

在 VLAN 上应用 VLAN map

从特权 EXEC 模式开始，用户可以按照以下步骤在一个或多个 VLAN 上应用 VLAN map：

总步骤：

1. enable

2. configure terminal

3. vlan filter *mapname* **vlan-list** *list*

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**vlan filter** *mapname* **vlan-list** *list* **示例：** `Device(config)# vlan filter map 1 vlan-list 20-22`	在一个或多个 VLAN ID 上应用 VLAN map。 *list* 可以是一个 VLAN ID（22）、一个连续的列表（10 - 22），或者多个 VLAN ID（12, 22, 30）。逗号和连字符前后的空格是可选的

步骤 4	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 6	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

监控 IPv6 ACL

用户可以使用下面表格中展示的一条或多条特权 EXEC 命令，来查看所有配置的访问列表、IPv6 访问列表，或指定的访问列表。

命令	目的
show access-lists	显示交换机上配置的所有访问列表
show ipv6 access-lists [*access-list-name*]	显示所有配置的 IPv6 访问列表，或使用名称指定访问列表
show vlan access-map[*map-name*]	显示 VLAN 访问 map 的配置
showvlan filter[**access-map***access-map*|**vlan***vlan-id*]	显示 VACL 和 VLAN 之间的映射关系

以下示例展示了特权 EXEC 命令 **show access-list** 的输入信息。输出信息中会包含交换机或交换机堆栈上配置的所有访问列表。

```
Switch # show access-lists
Extended IP access list hello
10 permit ip any any
IPv6 access list ipv6
permit ipv6 any any sequence 10
```

以下示例展示了特权 EXEC 命令 **show ipv6 access-list** 的输出信息。输出信息中只包含交换机或交换机堆栈上配置的 IPv6 访问列表。

```
Switch# show ipv6 access-list
IPv6 access list inbound
permit tcp any any eq bgp (8 matches) sequence 10
permit tcp any any eq telnet (15 matches) sequence 20
permit udp any any sequence 30
IPv6 access list outbound
deny udp any any sequence 10
deny tcp any any eq telnet sequence 20
```

以下示例展示了特权 EXEC 命令 **show vlan access-map** 的输出信息。输出信息中展示了 VLAN 访问 map 的信息。

```
Switch# show vlan access-map
Vlan access-map "m1" 10
Match clauses:
ipv6 address: ip2
Action: drop
```

其他参考资料

相关主题

相关主题	文档名称
IPv6 安全配置主题	IPv6 Configuration Guide, Inspur INOS XE Release 3SE (Inspur 6850 Switches) http://www.icntnetworks.com
IPv6 命令参考	IPv6 Command Reference, Inspur INOS XE Release 3SE (Inspur 6850 Switches) http://www.icntnetworks.com

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。 要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	http://www.icntnetworks.com

配置 DHCP

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于 DHCP 的信息

DHCP 服务器

DHCP 服务器从交换机或路由器上指定的地址池中分配 IP 地址给 DHCP 客户端，并管理这些地址。如果 DHCP 服务器自己的数据库不能给 DHCP 客户端提供请求的配置参数，该服务器会把请求转发给网络管理员定义的一个或多个次级 DHCP 服务器。交换机可以被配置为 DHCP 服务器。

DHCP 中继代理

DHCP 中继代理是一个在 DHCP 客户端和服务器之间转发 DHCP 包的三层设备。当客户端和服务器不在相同的物理子网上时，中继代理需要在其间转发请求和响应。在正常的二层转发过程中，IP 数据包会透明地在网络之间进行交换，而中继代理的转发行为与正常的二层转发不同。中继代理接收到 DHCP 信息后，会生成新的 DHCP 消息并在出接口上发送。

DHCP 侦听

DHCP 侦听是一项提供网络安全性的 DHCP 安全特性，其过滤不可信的 DHCP 消息，构建并维护一个 DHCP 侦听绑定数据库，也称为 DHCP 侦听绑定表。
DHCP 侦听是不可信主机和 DHCP 服务器之间的防火墙。管理员可以使用 DHCP 侦听特性来区分连接到终端用户的不可信接口，以及连接到 DHCP 服务器或其他交换机的可信接口。
注释： 为使 DHCP 侦听特性正常工作，所有 DHCP 服务器必须通过可信接口连接到交换机。

不可信的 DHCP 消息是通过不可信接口收到的消息。默认情况下，交换机认为所有的接口都是不可信的。所以为了使用 DHCP 侦听，用户必须配置交换机的一些接口为可信接口。在服务提供商环境中使用 DHCP 侦听特性时，不可信消息是从服务提供商网络外的设备发来的，比如客户的交换机。来自未知设备的消息是不可信的，因为这些设备可能是流量攻击的源点。DHCP 侦听绑定数据库中包含 MAC 地址、IP 地址、租用时间、绑定类型、VLAN 编号以及对应于一个交换机本地不可信接口的接口信息。数据库中没有与可信接口互连的主机的信息。在服务提供商网络中，可能配置为可信接口的一个例子是与相同网络中主机端口相连的接口。不可信接口的例子是与网络中不可信接口相连的接口，或是与网络之外的设备相连的接口。

当交换机在一个不可信接口上收到一个包，且接口属于启用了 DHCP 侦听的 VLAN 时，交换机会对比源 MAC 地址以及 DHCP 客户端的硬件地址。如果两地址相同（默认情况），交换机会转发此包。如果两地址不同，交换机会丢弃此包。

交换机在以下情况之一发生时，会丢弃 DHCP 包：

- 从网络或防火墙外部收到了来自 DHCP 服务器的包，类型如 DHCPOFFER、DHCPACK、DHCPNAK 或 DHCPLEASEQUERY；
- 在不可信接口上接收到数据包，且源 MAC 地址与 DHCP 客户端的硬件地址不相同；
- 交换机接收到了 MAC 地址在 DHCP 侦听绑定数据库中的 DHCPRELEASE 或 DHCPDECLINE 广播消息，但绑定数据库中的接口信息与接收消息的接口不相同；
- DHCP 中继代理转发了中继代理 IP 地址不为 0.0.0.0 的 DHCP 包，或者中继代理将一个包含可选 82 信息的包转发给了不可信端口。

如果交换机是一台支持 DHCP 侦听的汇聚层交换机，且连接到了一台插入可选 82 信息的边界交换机，该交换机将丢弃从不可信接口接收到的带有可选 82 信息的包。如果启用了 DHCP 侦听且在可信端口上收到了包，该汇聚层交换机不会学习连接设备的 DHCP 侦听绑定信息，且不能构建完整的 DHCP 侦听绑定数据库。

当一台汇聚层交换机可以通过一个不可信接口连接到一台边界交换机，且用户输入了 **ip dhcp snooping information option allow-untrusted** 全局配置命令时，该汇聚层交换机会接受来自边界交换机的带有可选 82 信息的包。该汇聚层交换机将会学习通过不可信交换机接口连接的主机的绑定信息。当交换机通过主机连接的不可信入接口收到带有可选 82 信息的包时，在该汇聚层交换机上仍然可以启用 DHCP 安全特性，如动态 ARP 监测或 IP 源防护。连接到汇聚层交换机的边界交换机的端口必须被配置为可信接口。

插入可选 82 数据

在住宅及城域以太网接入环境中，DHCP 可以中心化地管理大量租户的 IP 地址分配。在交换机上启用 DHCP 可选 82 特性时，租户的设备（除 MAC 地址外）由将其连接至网络的交换机端口标识。租户 LAN 中的多台主机可以连接到接入交换机的相同端口上，且被唯一标识。

注释： 只有在使用可选 82 的租户设备分配的 VLAN 上全局启用 DHCP 侦听特性时，才支持使用 DHCP 可选 82 特性。

下面展示了一个城域以太网网络，其中有一个中心化的 DHCP 服务器给连接到接入层交换机的租户分配 IP 地址。因为 DHCP 客户端及相关联的 DHCP 服务器不在相同的 IP 网络或子网中，所以给一个 DHCP 中继代理（Inspur 交换机）配置了 helper 地址，使其可以转发广播包并在客户端和服务器之间传输 DHCP 消息。

图 113：城域以太网络中的 DHCP 中继代理

DHCP server	DHCP 服务器
Catalystswitch（DHCP relay agent）	Catalyst 交换机（DHCP 中继代理）
Access layer	接入层
Host A (DHCP client)	主机 A（DHCP 客户端）
Subscribers	租户
Host B (DHCP client)	主机 B（DHCP 客户端）
VLAN 10	VLAN 10

在交换机上启用 DHCP 侦听信息可选 82 时，将会发生以下一系列事件：

- 主机（DHCP 客户端）生成 DHCP 请求并广播到网络中；
- 当交换机收到 DHCP 请求时，它向包中添加可选 82 信息。默认情况下，远程 ID 子选项是交换机的 MAC 地址，电路 ID 子选项是端口标识符 **vlan-mod-port**，即接收包的端口。可以配置远程 ID 及电路 ID；
- 如果配置了中继代理的 IP 地址，交换机把此 IP 地址添加到 DHCP 包中；
- 交换机将包含可选 82 字段的 DHCP 请求转发给 DHCP 服务器；
- DHCP 服务器收到包。如果服务器可以处理可选 82 信息，它会使用远程 ID 或电路 ID 来分配 IP 地址并实施策略，比如限制可以分配给一个远程 ID 或电路 ID 的 IP 地址数量。随后 DHCP 服务器在 DHCP 应答中回复该可选 82 字段；
- 如果请求是通过交换机中继给服务器的，DHCP 服务器会把应答单播发给交换机。交换机通过检查远程 ID 或电路 ID 字段，证实该字段是自己插入的。交换机会移除可选 82 字段，并把包转发给连接发送 DHCP 请求的 DHCP 客户端的交换机端口。

在默认的子选项配置中，当上述的一系列事件发生时，这些字段中的值不会改变（见图示 *子选项包格式*）：

- 电路 ID 子选项字段
  - 子选项类型
  - 子选项类型长度
  - 电路 ID 类型
  - 电路 ID 类型长度
- 远程 ID 子选项自选
  - 子选项类型
  - 子选项类型长度
  - 远程 ID 类型
  - 远程 ID 类型长度

在电路 ID 子选项的端口字段中，端口编号从 3 开始。比如，在一台有 24 个 10/100/1000 端口和四个小型可插拔（small form-factor pluggable，SFP）模块插槽的交换机上，端口 3 是吉比特以太网 1/0/1 端口，端口 4 是吉比特以太网 1/0/2 端口，以此类推。端口 27 是 SFP 模块插槽吉比特以太网 1/0/25 端口，以此类推。

图示 *子选项包格式*展示了使用默认子选项配置时的远程 ID 子选项和电路 ID 子选项。对于电路 ID 子选项，模块编号对应于堆栈中的交换机编号。在全局启用 DHCP 侦听并输入 ip dhcp snooping information option 全局配置命令时，交换机使用此包格式。

图 114：子选项包格式

Circuit ID Suboption Frame Format	电路 ID 子选项帧格式
Suboption type	子选项类型
Length	长度
Circuit ID type	电路 ID 类型

Module	模块
Port	端口
1 byte	1 字节
Remote ID Suboption Frame Format	远程 ID 子选项帧格式
Remote ID type	远程 ID 类型
MAC address	MAC 地址

图示用户配置的子选项包格式展示了用户配置的远程 ID 和电路 ID 子选项的包格式。用户在全局启用 DHCP 侦听，且输入全局配置命令 **ip dhcp snooping information option format remote-id** 以及接口配置命令 **ip dhcp snooping vlan information option format-type circuit-id string** 时，交换机使用这些包格式。

用户在配置远程 ID 和电路 ID 子选项时，包中这些字段的值会从默认值变为配置值：

- 电路 ID 子选项字段
  - 电路 ID 类型为 1；
  - 长度值可变，取决于配置的字符串长度。
- 远程 ID 子选项字段
  - 远程 ID 类型为 1；
  - 长度值可变，取决于配置的字符串长度。

图 115：用户配置的子选项包格式

Circuit ID Suboption Frame Format(for user-configured string)	电路 ID 子选项帧格式（用户定义字符串）
Suboption type	子选项类型
Length	长度
Circuit ID type	电路 ID 类型
ASCII Circuit ID string	ASCII 电路 ID 字符串
1 byte	1 字节
Remote ID Suboption Frame Format(for user-configured string)	远程 ID 子选项帧格式（用户定义字符串）
Remote ID type	远程 ID 类型
MAC address	MAC 地址
ASCII Remote ID string or hostname	ASCII 远程 ID 字符换或 hostname

Inspur INOS DHCP 服务器数据库

在基于 DHCP 的自动配置过程中，指定的 DHCP 服务器会使用 Inspur INOS DHCP 服务器数据库。其中有 IP 地址、地址绑定以及配置的参数，如启动文件。

地址绑定是 Inspur INOS DHCP 服务器数据库中 IP 地址和 MAC 地址的一个映射。管理员可以手动指定客户端 IP 地址，DHCP 服务器也可以从 DHCP 地址池中分配 IP 地址。更多有关手动及自动进行地址绑定的信息，参见 *Inspur INOS IP 配置指南 12.4 版*的“配置 DHCP”章节。

有关启用并配置 Inspur INOS DHCP 服务器数据库过程的信息，参见 *Inspur INOS IP 配置指南 12.4 版*的“配置 DHCP”章节中的“DHCP 配置任务列表”一节。

DHCP 侦听绑定数据库

启用 DHCP 侦听时，交换机使用 DHCP 侦听绑定数据库存储不可信接口的相关信息。数据库可存储至多 64000 个绑定条目。

每个数据库条目（绑定）都包含一个 IP 地址，一个关联的 MAC 地址，租用时间（十六进制格式），绑定适用的接口以及接口所属的 VLAN 信息。数据库代理将绑定信息以文件的形式存储在配置的位置。每个条目结尾都有一个校验和，负责对从文件开始的所有与条目相关的字节进行校验。每个条目为 72 字节，后接一个空格，然后是校验和值。

为了在交换机重启时保留绑定信息，管理员必须使用 DHCP 侦听数据库代理。如果代理被禁用，动态 ARP 监测或 IP 源防护会被启用，且 DHCP 侦听绑定数据库有动态绑定条目，则交换机会失去连通性。如果代理被禁用且只启用了 DHCP 侦听，交换机不会使用连通性，但 DHCP 侦听可能无法阻住 DHCP 伪造攻击。

重启时，交换机会读取绑定文件以构建 DHCP 侦听绑定数据库。数据库变化时交换机会更新此文件。

当交换机学习到新的绑定信息或者丢失绑定时，它会立即更新数据库中的条目。交换机也会更新绑定文件中的条目。更新文件的频率是基于可配置的时延的，而且更新批量进行。如果文在在特定的时间内没有被更新（由写时延以及终止超时时延设置），更新停止。

以下是绑定文件的格式：

```
<initial-checksum>
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
<entry-1><checksum-1>
<entry-2><checksum-1-2>
...
...
<entry-n><checksum-1-2-..-n>
END
```

文件中的每个条目都标记有一个校验和值，交换机在读取文件时用此值验证条目。第一行的初始校验和条目区分了与最新的文件更新相关的条目和与上一个文件更新相关的条目。

绑定文件的示例如下：

```
2bb4c2a1
TYPE DHCP-SNOOPING
VERSION 1
BEGIN
192.1.168.1 3 0003.47d8.c91f 2BB6488E Gi1/0/4 21ae5fbb
192.1.168.3 3 0003.44d6.c52f 2BB648EB Gi1/0/4 1bdb223f
192.1.168.2 3 0003.47d9.c8f1 2BB648AB Gi1/0/4 584a38f0
END
```

当交换机启动且计算出的校验和值与存储的校验和值相同时，交换机从绑定文件中读取条目并将其添加到自己的 DHCP 侦听绑定数据库中。当以下情况之一发生时，交换机忽略条目：

- 交换机读取了条目，且计算出的校验和值与存储的校验和值不同。该条目及其后的条目会被忽略。

- 一个条目有过期的租用时间（交换机可能不会在租用时间过期时移除绑定）。
- 系统中不再有条目中的接口。
- 接口是被路由接口或 DHCP 侦听可信接口。

DHCP 侦听及交换机堆栈

DHCP 侦听在堆栈主用设备上进行管理。新的交换机加入堆栈时，该交换机从堆栈主用设备接收 DHCP 侦听配置。当成员离开堆栈时，所有与该交换机关联的 DHCP 侦听地址绑定都会超时。

所有的侦听统计信息都在堆栈主用设备上产生。如果选出了新的堆栈主用设备，统计计数器会被重置。

当堆栈合并发生时，如果堆栈主用设备不再是新堆栈的主用设备，其上的所有 DHCP 侦听绑定都会被丢弃。对于使用堆栈分区的情况，现有堆栈主用设备不变，属于分区交换机的绑定都会超时。分区堆栈的新主用设备开始处理新进的 DHCP 包。

如何配置 DHCP 特性

默认的 DHCP 侦听配置

表 140：默认的 DHCP 配置

特性	**默认设置**
DHCP 服务器	在 Inspur INOS 软件中启用，需要配置 [13]
DHCP 中继代理	启用 [14]
DHCP 包转发地址	无配置
检查中继代理信息	启用（非法信息被丢弃）
DHCP 中继代理转发策略	替代现有的中继代理信息
全局启用 DHCP 侦听	禁用
DHCP 侦听信息选项	启用
接受不可信入端口包的 DHCP 侦听选项 [15]	禁用
DHCP 侦听限速	无配置
DHCP 侦听可信	不可信
DHCP 侦听 VLAN	禁用
DHCP 侦听 MAC 地址认证	启用
Inspur INOS DHCP 服务器绑定数据库	在 Inspur INOS 软件中启用，需要配置。 **注释：**交换只通过配置为 DHCP 服务器的设备获取网络地址及配置参数
DHCP 侦听绑定数据库代理	在 Inspur INOS 软件中启用，需要配置。此特性仅在配置了目的时可用

[13] 交换机仅被配置为 DHCP 服务器时才会响应 DHCP 请求。

[14] 交换机仅在 DHCP 客户端的 SVI 中配置了 DHCP 服务器的 IP 地址时才会中继 DHCP 包。

[15] 当交换机是汇聚层交换机且从边界交换机接收带有可选 82 信息的包时，使用此特性。

DHCP 侦听配置指南

如果一个交换机端口连接到 DHCP 服务器，输入配置命令 **ip dhcpsnooping trust interface** 配置该端口为可信。
如果一个交换机端口连接到 DHCP 客户端，用户需要输入接口配置命令 **no ip dhcpsnooping trust** 把该端口配置为不可信。
管理员可以输入用户 EXEC 命令 **show ip dhcp snooping statistics** 显示 DHCP 侦听统计信息，也可以输入 **clear ip dhcp snoopingstatistics** 特权 EXEC 命令清除侦听统计计数器。

配置 DHCP 服务器

交换机可以作为 DHCP 服务器使用。
关于配置交换机作为 DHCP 服务器使用的过程，参见 *Inspur INOS IP 配置指南 12.4 版*的“IP 编址及服务”章节中的“配置 DHCP”一节。

DHCP 服务器及交换机堆栈

DHCP 绑定数据库由堆栈主用设备管理。指定新的堆栈主用设备时，新的主用设备通过 TFTP 服务器下载存储的绑定数据库。如果堆栈主用设备故障，所有未保存的绑定都会丢失。与丢失的绑定相关的 IP 地址会被释放。管理员应用使用全局配置命令 **ip dhcp database** *url* [**timeout** *seconds* | **write-delay** *seconds*]配置自动备份。
堆栈合并发生时，变为堆栈成员设备的堆栈主用设备丢失所有的 DHCP 租用绑定。当堆栈分区发生时，分区中的新主用设备会成为新的 DHCP 服务器，不含有任何现有的 DHCP 租用绑定。

配置 DHCP 终极代理

按照以下步骤在交换机上启用 DHCP 中继代理：
总步骤
1. enable
2. configure terminal
3. service dhcp
4. end
5. show running-config
6. copy running-config startup-config
具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal**	进入全局配置模式

	示例： `Device#` **configure terminal**	
步骤 3	**service dhcp** **示例：** `Device(config)#` **service dhcp**	在交换机上启用 DHCP 服务器器及中继代理。此特性默认被启用
步骤 4	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** `Device#` **show running-config**	验证配置的条目
步骤 6	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中

接下来做什么？

查看 *Inspur INOS IP 配置指南 12.4 版*的“IP 编址及服务”章节“配置 DHCP”部分的以下步骤：

- 检查（验证）中继代理信息
- 配置中继代理转发策略

指定包转发地址

如果 DHCP 服务器和 DHCP 客户端不再相同的网络或子网上，管理员必须使用 **ip helper-address** *address* 接口配置命令配置交换机。一般规则是在接近客户端的三层接口上配置该命令。命令 **ip helper-address** 中使用的地址可以是特定的 DHCP 服务器 IP 地址，也可以是其他 DHCP 服务器所在目的网段的网络地址。使用网络地址允许任意 DHCP 服务器响应请求。

从特权 EXEC 模式开始，按照以下步骤指定包转发地址：

总步骤

1. enable

2. configure terminal

3. interface vlan *vlan-id*

4. ip address *ip-address subnet-mask*

5. ip helper-address *address*

6. end

7. 使用以下命令之一:

• interface range *port-range*

• interface *interface-id*

8. switchport mode access

9. switchport access vlan *vlan-id*

10. end

11. show running-config

12. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**interface vlan** *vlan-id* **示例：** `Device(config)# interface vlan 1`	输入 VLAN ID 创建交换机虚接口，并进入接口配置模式
步骤 4	**ip address** *ip-address subnet-mask* **示例：** `Device(config-if)# ip address 192.108.1.27 255.255.255.0`	给接口配置 IP 地址和 IP 子网
步骤 5	**ip helper-address** *address* **示例：** `Device(config-if)# ip helper-address 172.16.1.2`	指定 DHCP 包的转发地址。 helper 地址可以是一个特定的 DHCP 服务器地址，也可以是其他 DHCP 服务器所在目的网段的网络地址。使用网络地址允许其他服务器应答 DHCP 请求。 如果有多个服务器，可以为每个服务器配置一个 helper 地址
步骤 6	**end** **示例：** `Device(config-if)# end`	返回全局配置模式
步骤 7	使用以下命令之一: • **interface range** *port-range* • **interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet1/0/2`	配置多个连接到 DHCP 客户端的物理端口，并进入端口范围配置模式。 或 配置一个连接到 DHCP 客户端的物理端口，并进入接口配置模式
步骤 8	**switchport mode access** **示例：** `Device(config-if)# switchport mode access`	为端口定义 VLAN 成员模式
步骤 9	**switchport access vlan** *vlan-id* **示例：** `Device(config-if)# switchport access vlan 1`	指定端口到第 3 步配置的 VLAN 中
步骤 10	**end** **示例：** `Device(config-if)# end`	返回特权 EXEC 模式
步骤 11	**show running-config**	验证配置的条目

	示例： `Device#` **show running-config**	
步骤 12	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中

配置 DHCP 侦听及可选 82 的前提

配置 DHCP 侦听及可选 82 特性的前提如下：

- 管理员必须在交换机上全局启用 DHCP 侦听；
- 全局启用 DHCP 侦听之前，确保配置了作为 DHCP 服务器工作的设备，且 DHCP 中继代理配置并启用；
- 如果希望交换机响应 DHCP 请求，必须配置交换机作为 DHCP 服务器；
- 在交换机上配置 DHCP 侦听信息选项之前，确保配置了作为 DHCP 服务器工作的设备。管理员必须指定该 DHCP 服务器可以分配或排除的 IP 地址，或者必须为这些设备配置 DHCP 选项；
- 为了让 DHCP 侦听能正常工作，所有的 DHCP 服务器必须通过可信接口连接到交换机。在服务提供商网络中，可信接口是连接到位于相同网络的设备端口上的接口；
- 为使用 DHCP 侦听特性，必须配置交换机使用 Inspur INOS DHCP 服务器绑定数据库；
- 为使用 DHCP 侦听选项接受在不可信入端口上收到的包，交换机必须是汇聚层交换机，且从边界交换机上接收带有可选 82 信息的包；
- DHCP 侦听绑定数据库配置的前提如下：
  - 为使用 DHCP 侦听特性，必须在 DHCP 侦听绑定数据库中配置目的；
  - 因为 NVRAM 和闪存的存储空间都很有限，建议将绑定文件存储在 TFTP 服务器上；
  - 对于使用基于网络的 URL 来说（如 TFTP 以及 FTP），管理员必须为配置的 URL 创建一个空文件，以使交换机可以写入绑定信息到该 URL 的绑定文件中。请查看使用的 TFTP 服务器的文档，确定是否必须现在服务器上创建空文件；一些 TFTP 服务器无法这样配置；
  - 为了确保数据库中的租用时间是准确的，建议管理员启用并配置网络时间协议（Network Time Protocol，NTP）；
  - 如果配置了 NTP，交换机只会在交换机的系统时间与 NTP 同步后才会将绑定更新写入到绑定文件中。
- 在交换机上配置 DHCP 中继代理之前，确保配置了工作为 DHCP 服务器的设备。管理员必须指定该 DHCP 服务器可以分配或排除的 IP 地址，配置该设备的 DHCP 选项，或设置 DHCP 数据库代理；
- 如果希望交换机中继 DHCP 包，必须在 DHCP 客户端的交换机虚接口（SVI）上必须配置 DHCP 服务器的 IP 地址；
- 如果交换机端口连接到DHCP服务器，需输入配置命令ip dhcpsnooping trust interface将端口配置为可信；
- 如果端口连接到DHCP客户端，需使用接口配置命令no ip dhcpsnooping trust将端口配置为不可信。

启用 Inspur INOS DHCP 服务器数据库

有关启用及配置 Inspur INOS DHCP 服务器数据库的过程，参见 *Inspur INOS IP 配置指南 12.4 版*的“配置 DHCP”章节的“DHCP 配置任务列表”部分。

监控 DHCP 侦听信息

表 141：显示 DHCP 信息的命令

命令	描述
show ip dhcp snooping	显示交换机的 DHCP 侦听配置。
show ip dhcp snooping binding	只显示 DHCP 侦听绑定数据库（绑定表）中动态配置的绑定信息。
show ip dhcp snooping database	显示 DHCP 侦听绑定数据库的状态及统计信息。
show ip dhcp snooping statistics	显示 DHCP 侦听统计信息的汇总详情。
show ip source binding	显示动态及静态配置的绑定信息。

注释： 如果启用了 DHCP 侦听，且接口变为 down 状态，交换机不会删除静态配置的绑定条目。

配置 DHCP 服务器进行基于端口的地址分配

配置 DHCP 服务器进行基于端口地址分配的相关信息

DHCP 服务器的基于端口地址分配特性允许 DHCP 在一个以太网交换机端口上保持使用相同的 IP 地址，无论连接的设备客户端标识符或客户端硬件地址如何变化。

在网络中部署使用以太网交换机时，它们为直连设备提供连通性。在比如工厂车间这样的环境中，如果设备发生故障，替换的设备必须能立刻在现有网络中工作。当前的 DHCP 部署方式无法保证 DHCP 能给替换的设备提供相同的 IP 地址。控制、监控及其他软件希望每台设备能有关联的稳定的 IP 地址。如果设备被替换，即便 DHCP 客户端改变了，地址分配过程也应保持稳定。

配置了 DHCP 服务器的基于端口的地址分配特性后，就能确保给相同的连接端口分配相同的 IP 地址，即使从该端口收到的 DHCP 消息中客户端标识符或客户端硬件地址发生了改变。DHCP 协议通过 DHCP 包中的客户端标识符选项识别 DHCP 客户端。不包含客户端标识符选项的客户端通过其硬件地址进行标识。配置了此特性后，接口的名称覆盖客户端标识符或硬件地址，而交换机端口这个实际的连接点成为了客户端标识符。

所有情况中，通过以太网线缆连接到相同端口的设备都能通过 DHCP 获取相同的 IP 地址。

DHCP 服务器的基于端口的地址分配特性只在 Inspur INOS DHCP 服务器上支持，第三方服务器不支持此特性。

默认的基于端口地址分配配置

默认情况下，DHCP 服务器的基于端口地址分配特性被禁用。

基于端口的地址分配配置指南

- 默认情况下，DHCP 服务器的基于端口地址分配特性被禁用；
- 为把地址的分配从 DHCP 地址池限制到预配置的预留地址中（非预留地址不会提供给客户端，其他客户端不由该地址池服务），管理员可以输入 **reserved-only** DHCP 地址池配置命令。

启用 DHCP 侦听绑定数据库代理

总步骤

1. enable

2. configure terminal

3. ip dhcp snooping database {**flash**[*number*]**:**/*filename* | **ftp://***user***:***password***@***host***/***filename* | **http://**[[*username***:***password*]**@**]{*hostname | host-ip*}[/*directory*] **/***image-name***.tar** | **rcp://***user***@***host***/***filename*}| **tftp://***host***/***filename*

4. ip dhcp snooping database timeout *seconds*

5. ip dhcp snooping database write-delay *seconds*

6. end

7. ip dhcp snooping binding *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id* **expiry** *seconds*

8. show ip dhcp snooping database [**detail**]

9. show running-config

10. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 3	**ip dhcp snooping database** {**flash**[*number*]**:**/*filename* | **ftp://***user***:***password***@***host***/***filename* | **http://**[[*username***:***password*]**@**]{*hostname |* *host-ip*}[/*directory*] **/***image-name***.tar** | **rcp://***user***@***host***/***filename*}| **tftp://***host***/***filename* **示例：**	使用以下命令之一指定数据库代理或绑定文件的 URL： • **flash**[*number*]**:/***filename* （可选）使用 *number* 参数指定堆栈主用设备的堆栈成员编号。*number* 的范围从 1 到 9。 • **ftp://***user***:***password***@***host***/***filename* •

	Device(config)# **ip dhcp snooping database tftp://10.90.90.90/snooping-rp2**	**http://**[[*username*:*password*]**@**]{*hostname* | *host-ip*}[/*directory*] **/***image-name***.tar** • **rcp://***user***@***host***/***filename* • **tftp://***host***/***filename*
步骤 4	**ip dhcp snooping database timeout** *seconds* **示例：** Device(config)# **ip dhcp snooping database timeout 300**	指定在停止数据库传输过程前等待多久（单位秒）。 默认值是 300 秒，范围从 0 到 86400。使用 0 指定无限期间，表示无限期的尝试传输
步骤 5	**ip dhcp snooping database write-delay** *seconds* **示例：** Device(config)# **ip dhcp snooping database write-delay 15**	指定绑定数据库变化后应该延迟传输的时间长度。范围从 15 到 86400 秒，默认值是 300 秒（5 分钟）
步骤 6	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式
步骤 7	**ip dhcp snooping binding** *mac-address* **vlan** *vlan-id* *ip-address* **interface** *interface-id* **expiry** *seconds* **示例：** Device# **ip dhcp snooping binding 0001.1234.1234 vlan 1 172.20.50.5 interface gi1/1 expiry 1000**	（可选）向 DHCP 侦听绑定数据库添加绑定条目。*vlan-id* 的范围从 1 到 4904。*seconds* 的范围从 1 到 4294967295。 添加每个条目都要输入此命令。 测试或调试交换机时使用此命令
步骤 8	**show ip dhcp snooping database** [**detail**] **示例：** Device# show ip dhcp snooping database detail	显示 DHCP 侦听绑定数据库代理的状态和统计信息
步骤 9	**show running-config** **示例：** Device# **show running-config**	验证配置的条目
步骤 10	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把配置保存在配置文件中

启动 DHCP 服务器的基于端口地址分配特性

按照以下步骤全局启用基于端口的地址分配特性，并在接口上自动生成租户标识符。

总步骤

1. enable

2. configure terminal

3. ip dhcp use subscriber-id client-id

4. ip dhcp subscriber-id interface-name

5. interface *interface-id*

6. ip dhcp server use subscriber-id client-id

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 3	**ip dhcp use subscriber-id client-id** **示例：** `Device(config)#` **ip dhcp use subscriber-id client-id**	配置 DHCP 服务器，将租户标识符作为客户端标识符全局应用在所有进入的 DHCP 消息上
步骤 4	**ip dhcp subscriber-id interface-name** **示例：** `Device(config)#` **ip dhcp subscriber-id interface-name**	基于接口的短名称自动生成租户标识符。 特定接口上的租户标识符配置优先于此命令
步骤 5	**interface** *interface-id* **示例：** `Device(config)#` **interface gigabitethernet1/0/1**	指定需要配置的接口，进入接口配置模式
步骤 6	**ip dhcp server use subscriber-id client-id** **示例：** `Device(config-if)#` **ip dhcp server use subscriber-id client-id**	配置 DHCP 服务器使用租户标识符作从该接口进入的所有 DHCP 消息的客户端标识符
步骤 7	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式
步骤 8	**show running-config** **示例：** `Device#` **show running-config**	验证配置的条
步骤 9	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中

接下来做什么？

在交换机上启用 DHCP 基于端口的地址分配之后，使用 **ip dhcp pool** 全局配置命令预分配 IP 地址并将其与客户端关联。

监控 DHCP 服务器基于端口的地址分配

表 142：显示 DHCP 基于端口的地址分配信息的命令

命令	目的
show interface *interface id*	显示特定接口的状态和配置
show ip dhcp pool	显示 DHCP 地址池
show ip dhcp binding	显示 Inspur INOS DHCP 服务器上的地址绑定信息

其他参考资源

相关文档

相关主题	文档题目
DHCP 配置信息及过程	IP 编址：DHCP 配置指南，Inspur INOS XE 3S 版 http://www.icntnetworks.com

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

配置 IP 源防护

IP 源防护（IP Source Guard，IPSG）是一项在非路由的二层接口上限制 IP 流量的安全特性。该特性基于 DHCP 侦听绑定数据库和手动配置的 IP 源绑定信息进行流量过滤。

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

有关 IP 源防护的信息

IP 源防护

如果一台主机尝试使用邻居的 IP 地址，可以使用 IP 源防护来避免流量攻击。当 DHCP 侦听在不可信接口上启用时也可以启用 IP 源防护。
在接口上启用 IPSG 之后，交换机会阻隔除了 DHCP 侦听允许的 DHCP 包之外的所有在接口上收到的 IP 流量。
交换机使用硬件的源 IP 查找表来绑定 IP 地址到端口。对于 IP 和 MAC 过滤，交换机会组合进行源 IP 以及源 MAC 的查找。源 IP 地址在绑定表中 IP 流量会被允许，而所有其他流量会被拒绝。
IP 源绑定表中的绑定条目是通过 DHCP 侦听学习到的或者是手工配置的（静态 IP 源绑定）。表中的一个条目包含 IP 地址，关联的 MAC 地址以及关联 VLAN 编号。交换机只在启用 IP 源防护的时候使用 IP 源绑定表。
只有包括接入端口和中继端口这样的二层端口才支持 IPSG。可以配置 IPSG 进行源 IP 地址过滤或源 IP 及 MAC 地址过滤。

静态主机的 IP 源防护

注释： 不要对上行链路端口上的静态主机或中继端口使用 IPSG（IP 源防护）。
静态主机的 IP 源防护将 IPSG 的能力扩展到了非 DHCP 的静态环境中。以前的 IPSG 使用 DHCP 侦听创建的条目来验证连接到交换机的主机。任何从主机收到的没有对应合法 DHCP 绑定条目的流量都会被丢弃。这项安全特性限制了非路由二层接口上的 IP 流量。它基于 DHCP 侦听绑定数据库以及手动配置的 IP 源绑定信息过滤流量。以前的 IPSG 版本要求有 DHCP 环境才能工作。
静态主机的 IPSG 允许在不使用 DHCP 的情况下工作。静态主机的 IPSG 依赖 IP 设备追踪表的

条目来安装端口 ACL。交换机根据 ARP 请求或者其他的 IP 包来创建条目，维护特定端口的合法主机列表。管理员也可以指定允许给特定端口发送流量的主机数量。这项操作等同于三层的端口安全特性。

静态主机的 IPSG 也支持动态主机。如果一台动态主机接收了一个 DHCP 分配的 IP 地址，且这个地址同时在 IP DHCP 侦听表中可用，IP 设备追踪表也会学习到相同的条目信息。在堆叠环境中，当主用设备故障切换发生时，连接到成员端口的静态主机的 IP 源防护条目将被保留。当管理员输入 EXEC 命令 **show ip device tracking all** 时，IP 设备追踪表会显示这些条目状态的为 ACTIVE。

注释： 一些有多个网络接口的 IP 主机可能会向网络接口发送非法的数据包。非法的数据包会以该主机其他网络接口的 IP 或 MAC 地址作为源。这些非法的数据包可以导致静态主机的 IPSG 连接到该主机，获知非法的 IP 或 MAC 地址绑定信息，并拒绝合法的绑定。请咨询对应操作系统及网路接口的提供商，避免主机发送非法的数据包。

静态主机的 IPSG 开始时通过基于 ACL 的侦听机制动态地学习 IP 或 MAC 绑定。IP 或 MAC 的绑定是通过 ARP 和 IP 包从静态主机上学习来的。这些信息被存储在设备追踪数据库中。当特定端口上动态学习或者静态配置的 IP 地址数量达到最大值时，交换机硬件会丢弃任何使用新的 IP 地址的包。为了解决主机因故移动或移除的问题，静态主机的 IPSG 会利用 IP 设备追踪功能来对动态获知的 IP 地址绑定信息进行超时处理。这项特性可以与 DHCP 侦听特性一起使用。对于同时连接了 DHCP 主机和静态主机的端口，将会有多个绑定条目被创建。例如，绑定信息会同时存储在设备追踪数据库和 DHCP 侦听绑定数据库中。

IP 源防护配置指南

- 管理员只能在非路由端口上配置静态IP绑定特性。如果在被路由接口上输入了**ip source binding***mac-address***vlan***vlan-id ip-address***interface** *interface-id*全局配置命令，将出现以下错误信息：
  - `Static IP source binding can only be configured on switch port.`
- 在接口上启用过滤源 IP 的 IP 源防护时，必须在该接口的接入 VLAN 上启用 DHCP 侦听；
- 如果在有多个 VLAN 的中继端口上启用了 IP 源防护，且对所有 VLAN 都启用了 DHCP 侦听，源 IP 地址过滤会被应用到所有 VLAN 上；

注释： 如果启用了 IP 源防护且管理员对中继端口上的 VLAN 启用或禁用了 DHCP 侦听，交换机可能无法正常过滤流量。

- 可以在启用 802.1x 基于端口认证特性的同时启用此特性。

如何配置 IP 源防护

启用 IP 源防护

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. ip verify source [**mac-check**]

5. exit

6. ip source binding *mac-address* **vlan***vlan-id ip-address* **interface** *interface-id*

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet 1/0/1**	指定要配置的接口，进入接口配置模式
步骤 4	**ip verify source** [**mac-check**] **示例:** `Device(config-if)#` **ip verify source**	启用进行源 IP 地址过滤的 IP 源防护。 （可选）**mac-check**——启用进行源 IP 地址过滤机 MAC 地址过滤的 IP 源防护
步骤 5	**exit** **示例:** `Device(config-if)#` **exit**	返回全局配置模式
步骤 6	**ip source binding** *mac-address* **vlan***vlan-id ip-address* **interface** *interface-id* **示例:** `Device(config)#` **ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface gigabitethernet1/0/1**	添加静态 IP 源绑定条目。 每个静态绑定条目都需输入此命令
步骤 7	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式
步骤 8	**show running-config** **示例：** `Device#` **show running-config**	验证配置的条目
步骤 9	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中

在二层接入端口上配置静态主机的 IP 源防护

为使静态主机的 IP 源防护特性工作，必须全局配置接口配置命令 **ip device tracking maximum***limit-number*。如果只在端口上配置了此命令而没有全局启用 IP 设备追踪，或者对该端口设置了最大的 IP 设备追踪数量，静态主机的 IPSG 会拒绝所有来自该接口的 IP 流量。

总步骤

1. enable

2. configure terminal

3. ip device tracking

4. interface *interface-id*

5. switchport mode access

6. switchport access vlan*vlan-id*

7. ip verify source[**tracking**] [**mac-check**]

8. ip device tracking maximum *number*

9. end

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 3	**ip device tracking** **示例:** `Device(config)#` **ip device tracking**	开启 IP 主机表，并全局启用 IP 设备追踪
步骤 4	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet 1/0/1**	进入接口配置模式
步骤 5	**switchport mode access** **示例:** `Device(config-if)#` **switchport mode access**	将端口配置为 Access 模式
步骤 6	**switchport access vlan***vlan-id* **示例:** `Device(config-if)#`**switchport access vlan 10**	为此端口配置 VLAN
步骤 7	**ip verify source**[**tracking**] [**mac-check**] **示例:** `Device(config-if)#`**ip verify source tracking** **mac-check**	启用进行源 IP 地址过滤的 IP 源防护。 （可选）为静态主机启用 IP 源防护。 （可选）启用 MAC 地址过滤。 命令 **ip verify source tracking mac-check**enables 启用进行 MAC 地址过滤

		的静态主机 IP 源防护
步骤 8	**ip device tracking maximum** *number* **示例:** `Device(config-if)#`**ip device tracking maximum 8**	设置 IP 设备追踪表允许端口拥有的最大静态 IP 数量。范围从 1 到 10，最大值是 10。 **注释**必须配置接口配置命令**ip device tracking maximum***limit-number*
步骤 9	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式

监控 IP 源防护

表 143：特权 EXEC show 命令

命令	目的
show ip verify source [**interface** *interface-id*]	显示交换机或者特定接口的 IP 源防护配置
showip device tracking { **all**| **interface** *interface-id* | **ip***ip-address* | **mac** *imac-address*}	显示 IP 设备追踪表中的条目信息

表 144：接口配置命令

命令	目的
ip verify source tracking	验证数据源

有关显示输出字段的详细信息，参见此版本的命令参考手册。

其他参考资料

错误信息解释

描述	**链接**
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

配置动态 ARP 监测

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

动态 ARP 监测的限制

本节列出了在交换机上配置动态 ARP 监测的限制条件和配置指南。

- 动态 ARP 监测是一种入向的安全特性，它不执行出向检查；
- 动态 ARP 监测对于连接到交换机上的不支持动态 ARP 监测或未启用此特性的主机无效。因为中间人攻击被限制在一个二层的广播域，而这个广播域被执行动态 ARP 监测检查的主机和不进行检查的主机分隔开。该特性的操作保护了域中启用动态 ARP 监测主机的 ARP 缓存；
- 动态 ARP 监测依靠 DHCP 侦听绑定数据库中的条目来验证入向 ARP 请求和响应的 IP-MAC 地址映射。确保启用了 DHCP 侦听特性，放行使用动态分配 IP 地址的 ARP 包。

在禁用 DHCP 侦听或非 DHCP 的环境中，使用 ARP ACL 允许或拒绝数据包。

- 动态 ARP 监测支持接入端口、中继端口以及 EtherChannel 端口。

注释： 不要在 RSPAN VLAN 上启用动态 ARP 监测特性。如果在 RSPAN VLAN 上启用了动态 ARP 监测，动态 ARP 监测包可能无法到达 RSPAN 目的端口。

- 只有当物理端口的可信状态与通道端口的可信状态相同时，物理端口才可以加入 EtherChannel 端口通道。否则，物理端口会在端口通道中保持挂起。端口通道从第一个加入通道的物理端口上继承可信状态。因此，第一个物理端口的可信状态无需与通道的可信状态相同。

  相反的，更改端口通道的可信状态时，交换机会在组成通道的所有物理端口上配置新的可信状态；
- 限速在交换机堆栈中的每个交换机上独立计算。对于跨堆栈的 EtherChannel，实际的限速值可能比配置的值高。例如，对于一个端口在交换机 1 上而另一个端口在交换机 2 上的 EtherChannel，如果管理员设置了限速为 30pps，每个端口在不造成 EtherChannel 错误禁用的情况下可以接收数据包的速度是 29pps；
- 端口通道的运行速率是通道内所有物理端口速率的累加和。例如，如果配置端口通道对

ARP 的限速为 400pps，通道中所有端口总共接收速率为 400pps。EtherChannel 端口的入向 ARP 包速率是所有通道成员的入向包速率的和。请在检查通道端口成员的入向 ARP 包速率之后再配置 EtherChannel 端口的限速速率。

物理端口上入向包的速率与端口通道的配置对比，不与物理端口的配置进行对比。端口通道的限速配置独立于物理端口的配置。

如果 EtherChannel 接收了超过配置速率的 ARP 包，信道（包括所有物理端口）会被置为错误禁用状态；

- 确保对入向中继端口上的 ARP 包进行限速。应给中继端口配置较高的速率以反映其聚合性，以便处理多个启用了动态 ARP 监测 VLAN 的数据包。也可以使用接口配置命令 **ip arp inspection limit none** 设置速率为不限制。当系统将端口置为错误禁用状态时，一个 VLAN 上的高限速设置可能会造成对另一个的 VLAN 的拒绝服务攻击；
- 在交换机上启用动态 ARP 监测时，配置用于管理 ARP 流量的策略器将不再生效。启用的结果是所有 ARP 流量都会被送往 CPU。

理解动态 ARP 监测

ARP 通过进行 IP 地址与 MAC 地址的映射在二层广播域间提供 IP 通信。例如，主机 B 希望给主机 A 发送信息，但是它的 ARP 缓存中没有主机 A 的 MAC 地址。主机 B 生成一个发往广播域中所有主机的消息，以获取与主机 A 的 IP 地址关联的 MAC 地址。广播域中的所有主机都会收到此 ARP 请求，而主机 A 使用自己的 MAC 地址来应答。然而，因为 ARP 允许主机在没有收到 ARP 请求的情况下发出无故应答，ARP 伪造攻击和 ARP 缓存的毒化就可以发生。在攻击发生后，被攻击设备的所有流量都会流经攻击者的计算机，然后再发往路由器、交换机或者主机。

恶意用户可以攻击连接到二层网络的主机、交换机和路由器，毒化连接到子网的系统的 ARP 缓存，并截获发往子网上其他主机的流量。图 26-1 展示了 ARP 缓存毒化的示例。

图 116：APR 缓存毒化

Host A	主机 A
Host B	主机 B
Host C(man-in-the-middle)	主机 C（中间人）

主机 A、B 和 C 连接到交换机的接口 A、B 和 C，这些端口在相同的子网中。主机的 IP 和 MAC 在括号中标出，如主机 A 使用 IP 地址 IA 和 MAC 地址 MA。当主机 A 需要与主机 B 在 IP 层通信时，它会广播一个 ARP 消息请求与 IP 地址 IB 关联的 MAC 地址。当交换机和主机 B 收到此 ARP 请求时，它们会填充自己的 ARP 缓存，产生主机 IP 地址为 IA，MAC 地址为 MA 的 ARP 绑定信息，如 IP 地址 IA 被绑定到 MAC 地址 MA 上。当主机 B 应答时，交换机和主机 A 填充其 ARP 缓存，产生主机 IP 地址为 IB，MAC 地址为 MB 的绑定。

主机 C 可以广播伪造的 ARP 应答，将主机 IP 地址 IA（或 IB）与 MAC 地址 MC 地址绑定，进而毒化交换机、主机 A 和主机 B 的 ARP 缓存。ARP 缓存被毒化的主机会使用 MAC 地址 MC 作为发往 IA 或 IB 的流量的目的 MAC 地址。这意味着主机 C 截获了这些流量。因为主机 C 知道与 IA 和 IB 关联的真正 MAC 地址，它可以使用正确的 MAC 地址作为目的把截获的流量转发给这些主机。主机 C 把自己插入在主机 A 到主机 B 的流量之间，这就是*典型的中间人（man-in-the middle）*攻击。

动态 ARP 监测是一项验证网络中 ARP 包的安全特性，它截获、记录并丢弃 IP-MAC 地址绑定非法的 ARP 包。这项特性能保护网络免于特定的中间人攻击。

动态 ARP 监测确保只有合法的 ARP 请求和应答被转发。交换机执行这些行为：

- 截获不可信端口上的所有 ARP 请求和应答
- 在更新本地 ARP 缓存或者把包转发给正确目的之前验证每个截获的数据包有合法的 IP-MAC 地址绑定
- 丢弃非法 ARP 包

动态 ARP 监测根据存储在可信数据库（DHCP 侦听绑定数据库）中的合法 IP-MAC 地址绑定来确定一个 ARP 包的合法性。如果交换机和 VLAN 上启用了 DHCP 侦听，这个数据库由 DHCP 侦听构建。如果在可信接口上收到 ARP 包，交换机不做检查转发此数据包。在不可信接口上，交换机只在数据包合法时才进行转发。

管理员可以使用全局配置命令 **ip arp inspection vlan** *vlan-range* 基于每个 VLAN 启用动态 ARP 监测。

在非 DHCP 环境中，动态 ARP 监测可以对比用户配置的 ARP 访问控制列表（access control lists，ACLs）验证静态配置 IP 地址的主机。管理员可以使用全局配置命令 **arp access-list** *acl-name* 定义 ARP ACL。

可以配置动态 ARP 监测在 ARP 包中的 IP 地址非法或者 ARP 包中的 MAC 地址与以太网报头中的地址不同时就丢弃数据包。使用全局配置命令 **ip arp inspection validate** {[**src-mac**] [**dst-mac**] [**ip**]}进行配置。

接口可信状态及网络安全性

动态 ARP 监测会把可信状态与交换机上的每个接口关联。可信接口上到达的数据包会绕过所有的动态 ARP 监测验证检查，不可信接口上到达的数据包会经历动态 ARP 监测验证过程。

典型的网络配置中，可以把所有连接到主机端口的交换机端口配置为不可信，把所有连接到交换机的交换机端口配置为可信。在此配置中，从特定交换机进入网络的所有 ARP 包会绕过安全检查，在 VLAN 或网络的任何其他地方都无需进行验证。可以使用接口配置命令 **iparp inspection trust** 设置可信状态。

注意： 请小心使用可信状态配置。在接口应该被信任时把它配置成不可信可能会导致丢失连通性。

下图中，假设交换机 A 和交换机 B 都在包含主机 1 和主机 2 的 VLAN 上运行动态 ARP 监测。如果主机 1 和主机 2 都通过连接到交换机 A 的 DHCP 服务器获取 IP 地址，只有交换机 A 会进行主机 1 的 IP-MAC 地址绑定。因此，如果交换机 A 和交换机 B 之间的接口是不可信的，来自主机 1 的 ARP 包会被交换机 B 丢弃。主机 1 和主机 2 之间的连通性会丢失。

图 117：启用动态 ARP 监测 VLAN 上的 ARP 包验证

DHCP server	DHCP 服务器
Switch A	交换机 A
Switch B	交换机 B
Host 1	主机 1
Port 1	端口 1

当接口实际不可信时配置其为可信会在网络中留下安全漏洞。如果交换机 A 不运行动态 ARP 监测，主机 1 可以容易的毒化交换机 B 的 ARP 缓存（如果交换机间的链路配置为可信，也会毒化主机 2 的缓存）。即使交换机 B 运行动态 ARP 监测也可以发生这种情况。

动态 ARP 监测确保连接到交换不可信接口上的主机不能毒化网络中其他主机的 ARP 缓存。然而，对于连接到运行动态 ARP 监测的交换机上的主机，动态 ARP 监测不能防止在网络其

他部分的主机毒化的这些主机的缓存。
在 VLAN 里一些交换机运行动态 ARP 监测而一些交换机不运行的情况中，应配置连接到这些交换机的接口为不可信。然而，为了验证来自非动态 ARP 监测交换机的数据包的绑定，可以配置运行动态 APR 监测的交换机使用 ARP ACL。在不能确定这样的绑定信息时，应在三层隔离运行动态 ARP 监测的交换机和不运行的交换机。
注释： 根据 DHCP 服务器以及网络设置的不同，可能无法在 VLAN 中的所有交换机上验证特定 ARP 包。

ARP 包的限速

交换机的 CPU 执行动态 ARP 监测检查。因此，为了避免拒绝服务攻击，要对入向 ARP 包的数量进行限速。默认情况下，不可信接口的速率是 15 包每秒（packets per second，pps）。可信接口不被限速。可以使用接口配置命令 **ip arp inspection limit** 更改此设置。
当入向 ARP 包的速率超过配置的限制时，交换机会把端口置为错误禁用状态。端口在管理员干预之前保持此状态。可以使用全局配置命令 **errdisable recovery** 启用错误禁用恢复，这样端口就可以在指定的超时周期之后自动摆脱此状态。
注释： 对于 EtherChannel 的限速会独立地应用到堆栈中的每个交换机上。比如，如果在 EtherChannel 上配置了限速 20pps，EtherChannel 中的每个交换机端口可以承载至多 20pps。如果任意交换机超过了限制，整个 EtherChannel 都会被置为错误禁用状态。

ARP ACL 和 DHCP 侦听条目的相对优先级

动态 ARP 监测使用 DHCP 侦听数据库作为合法 IP-MAC 地址的映射表。
ARP ACL 优先于 DHCP 侦听绑定数据库中的条目。交换机仅在使用全局配置命令 **ip arp inspection filter vlan** 配置时才使用 ACL。交换机首先把 ARP 包和用户配置的 ARP ACL 进行比较。如果 ARP ACL 拒绝此 ARP 包，即使 DHCP 侦听填充的数据库中有对应合法的绑定存在，交换机也会拒绝此数据包。

记录丢弃的数据包

当交换机丢弃一个数据包时，交换机会在日志缓存中放置一个条目，然后在控制速率的基础上生成系统消息。在消息生成后，交换机会从日志缓存中清除条目。每个日志条目都包含流信息，如接收 VLAN、端口号、源目 IP 地址以及源目 MAC 地址。
可以使用全局配置命令 **ip arp inspection log-buffer** 配置缓存的条目数量以及特定间隔内生成系统消息所需的条目数量。可以使用全局配置命令 **ip arp inspection vlan logging** 指定记录的数据包类型。

默认的动态 ARP 监测配置

特性	默认设置

动态 ARP 监测	在所有 VLAN 上禁用
接口可信状态	所有接口都是不可信
入向 ARP 包限速	不可信接口的速率是 15pps，假定网络是被交换网络，其中主机每秒连接多达 15 个新主机。 对所有可信接口不限速。 突发间隔是 1 秒
非 DHCP 环境的 ARP ACL	无 ARP ACL 定义
验证检查	不执行检查
日志缓存	启用动态 ARP 检查时，所有拒绝或丢弃的 ARP 包都被记录。 日志的条目数量是 32。 系统消息数量限制为 5 个每秒。 日志速率间隔是 1 秒
基于 VLAN 的日志	所有拒绝或丢弃的 ARP 包都被记录

ARP ACL 和 DHCP 侦听条目的相对优先级

动态 ARP 监测特性会为有效的 IP 到 MAC 地址绑定表，使用 DHCP 侦听（Snooping）绑定数据库。

ARP ACL 的优先级高于 DHCP Snooping 绑定数据库中的条目。只有用户使用全局配置命令 ip arp inspection filter vlan 进行了配置后，交换机才会使用这个 ACL。交换机会首先用 ARP 数据包与用户配置的 ARP ACL 进行比较。如果 ARP ACL 中拒绝 ARP 数据包，交换机也就会拒绝数据包，即使 DHCP Snooping 生成的数据库中存在有效的绑定关系。

为非 DHCP 环境配置 ARP ACL

以下展示了当图 2 中的交换机 B 不支持动态 ARP 监测或 DHCP 侦听时如何配置动态 ARP 监测特性。

如果配置交换机 A 的端口 1 为可信，就产生了一个安全漏洞，因为交换机 A 和主机 1 都可能被交换机 B 或主机 2 攻击。为了避免这种可能性，用户必须把交换机 A 上的端口 1 配置成不可信。要允许来自主机 2 的 ARP 包通过，必须设置 ARP ACL 并把它应用在 VLAN 1 上。如果主机 2 的 IP 地址不是静态的（不可能在交换机 A 上应用 ACL 配置），必须在三层隔离交换机 A 和交换机 B，并使用路由器在它们之间进行路由。

以下是在交换机 A 上配置 ARP ACL 的步骤。此过程应在非 DHCP 环境中执行。

总步骤

1. enable

2. configure terminal

3. arp access-list *acl-name*

4. permit ip host *sender-ip* **mac host** *sender-mac*

5. exit

6. ip arp inspection filter *arp-acl-name* **vlan** *vlan-range* [**static**]

7. interface *interface-id*

8. no ip arp inspection trust

9. end

10. 使用以下show命令:

- **show arp access-list** acl-name
- **show ip arp inspection vlan** *vlan-range*
- **show ip arp inspection interfaces**

11. show running-config

12. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 3	**arp access-list** *acl-name*	定义一个 ARP ACL，然后进入 ARP 访问列表配置模式。默认情况下，没有定义 ARP 访问列表。 **注释：**在ARP访问列表的末尾，有一个隐含的**deny ip anymac any**命令
步骤 4	**permit ip host** *sender-ip* **mac host** *sender-mac*	允许特定主机（主机 2）的 ARP 包。 • 对于 *sender-ip,*，输入主机 2 的 IP 地址。 • 对于 *sender-mac*，输入主机 2 的 MAC 地址
步骤 5	**exit**	返回全局配置模式。
步骤 6	**ip arp inspection filter** *arp-acl-name* **vlan** *vlan-range* [**static**]	把 ARP ACL 应用到 VLAN 上。默认情况下，没有应用到 VLAN 上的 ARP ACL。 • 对于 *arp-acl-name*，指定第 3 步中创建的 ACL 名称。 • 对于 *vlan-range*，指定交换机以及主机所属的 VLAN。 可以使用 VLAN ID 编号指定一个 VLAN，可以指定由连字符分隔的 VLAN 范围，也可以指定由逗号分隔的一组 VLAN。VLAN 范围从 1 到 4096。 • （可选）指定 **static** 字段，把 ARP ACL 中隐含的拒绝条目当作显式条目对待，丢弃与 ACL 中之前任意条目都不匹配的数据包。DHCP 绑定

		不被使用。 如果不指定此关键字，意味着 ACL 中没有显式的拒绝数据包的条目存在，DHCP 绑定就会在数据包不匹配 ACL 中任意行的时候决定其被允许还是被拒绝。 只包含 IP-MAC 地址映射的 ARP 包与 ACL 进行比较。只有在访问列表允许时这些数据包才被允许转发
步骤 7	**interface** *interface-id*	指定交换机 A 连接到交换机 B 的接口，并进入接口配置模式
步骤 8	**no ip arp inspection trust**	把交换机 A 连接到交换机 B 的接口配置为不可信。 默认情况下，所有接口都是不可信的。 对于不可信接口，交换机会截获所有的ARP请求和应答。交换机在更新本地缓存并把数据包转发到目的之前会验证截获的包是否有合法的IP-MAC地址绑定。交换机会丢弃非法的数据包，并根据全局配置命令**ip arpinspection vlan logging**指定的记录配置把这些包记录在日志缓存中
步骤 9	**end**	返回特权 EXEC 模式
步骤 10	使用以下show命令: • **show arp access-list** acl-name • **show ip arp inspection vlan** *vlan-range* • **show ip arp inspection interfaces**	验证配置的条目
步骤 11	**show running-config** **示例：** `Device#` **show running-config**	验证配置的条目
步骤 12	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中

在 DHCP 环境中配置动态 ARP 监测

在开始前

以下展示了两台交换机都支持动态 ARP 监测特性时如何进行配置。主机 1 连接到交换机 A，主机 2 连接到交换机 B。两台交换机都在主机所属的 VLAN 1 上运行动态 ARP 监测。有一台

DHCP 服务器连接到交换机 A。两台主机都通过相同的 DHCP 服务器获取 IP 地址。因此，交换机 A 有主机 1 和主机 2 的绑定信息，交换机 B 有主机 2 的绑定信息。

注释： 动态 ARP 监测根据 DHCP 侦听数据库中的条目验证入向 ARP 请求和 ARP 应答中的 IP-MAC 地址绑定信息。确保让 DHCP 侦听允许动态分配 IP 地址的 ARP 包。

按照以下步骤配置动态 ARP 监测。管理员必须在两台交换机上都进行此配置过程。此过程是必需的。

总步骤

1. enable

2. show cdp neighbors

3. configure terminal

4. ip arp inspection vlan *vlan-range*

5. interface*interface-id*

6. ip arp inspection trust

7. end

8. show ip arp inspection interfaces

9. show ip arp inspection vlan *vlan-range*

10. show ip dhcp snooping binding

11. show ip arp inspection statistics vlan *vlan-range*

~~**12. configure terminal**~~

~~**13. configure terminal**~~

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**show cdp neighbors** **示例:** `Device(config-if)#`**show cdp neighbors**	验证交换机之间的连接
步骤 3	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 4	**ip arp inspection vlan** *vlan-range* **示例:** `Device(config)#` **ip arp inspection vlan 1**	基于 VLAN 启用动态 ARP 监测。默认情况下，动态 ARP 监测在所有 VLAN 上禁用。对于 *vlan-rang*e 字段，可以使用 VLAN ID 编号指定一个 VLAN，可以指定由连字符分隔的 VLAN 范围，也可以指定由逗号分隔的一组 VLAN。VLAN 范围从 1 到 4096。请对两台交换机指定相同的 VLAN ID
步骤 5	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/1**	指定连接到其他交换机的接口，并进入接口配置模式
步骤 6	**ip arp inspection trust**	把交换机之间的连接配置为可信。默认情况

	示例: `Device(config-if)#`**ip arp inspection trust**	下，所有的接口都是不可信的。 交换机不会检查来自可信接口上其他交换机的 ARP 包。它会直接转发这些数据包。 对于不可信接口，交换机会截获所有的 ARP 请求和应答。交换机在更新本地缓存并把数据包转发到目的之前会验证截获的包是否有合法的 IP-MAC 地址绑定。交换机会丢弃非法的数据包，并根据全局配置命令 **ip arpinspection vlan logging** 指定的记录配置把这些包记录在日志缓存中
步骤 7	**end** **示例:** `Device(config-if)#`**end**	返回特权 EXEC 模式
步骤 8	**show ip arp inspection interfaces**	验证接口上的动态 ARP 监测配置
步骤 9	**show ip arp inspection vlan** *vlan-range* **示例:** `Device(config-if)#`**show ip arp inspection vlan 1**	验证 VLAN 上的动态 ARP 监测配置
步骤 10	**show ip dhcp snooping binding** **示例:** `Device(config-if)#`**show ip dhcp snooping binding**	验证 DHCP 绑定信息
步骤 11	**show ip arp inspection statistics vlan** *vlan-range* **示例:** `Device(config-if)#`**show ip arp inspection statistics vlan 1**	检查 VLAN 上的动态 ARP 监测统计信息

对入向 ARP 包限速

动态 ARP 监测验证检查由交换机的 CPU 执行；因此，为避免拒绝服务攻击，入向 ARP 包的数量应被限速。

当入向 ARP 包的速率超过了配置的限制时，交换机会把接口置为错误禁用状态。端口会已知保持在此状态中，直到管理员启用了错误禁用恢复，允许端口在指定的超时间隔后自动脱离此状态。

注释： 除非在接口上配置了限速，否则改变接口的可信状态也会把该可信状态的限速设置变为默认值。配置限速之后，即使接口的可信状态变化，接口也保持限速设置。如果输入接口配置命令 **no ip arp inspection limit**，接口会恢复到默认的限速设置。

按照以下步骤设置入向 ARP 包的限速值。此步骤是可选的。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. ip arp inspection limit {rate pps [burst interval seconds] | none}

5. exit

6. 使用以下show命令:

- **errdisable detect cause arp-inspection**
- **errdisable recovery cause arp-inspection**
- **errdisable recovery interval** *interval*

7. exit

8. 使用以下show命令:

- **show ip arp inspection interfaces**
- **show errdisable recovery**

9. show running-config

10. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 3	**interface** *interface-id*	指定要被限速的接口，进入接口配置模式
步骤 4	**ip arp inspection limit {rate pps [burst interval seconds] | none}**	对接口的入向 ARP 请求和应答进行限速。不可信接口上的默认速率是 15pps，可信接口无限制。突发间隔是 1 秒。 关键字的含义如下： • 对于 **rate** *pps*，指定每秒处理的入向数据包数量的上限。范围从 0 到 2048pps。 • （可选）对于 **burst interval** *seconds*，指定连续的间隔秒数，在此期间接口因高的 ARP 包速率被监控。范围从 1 到 15。 • 对于 **rate none**，指定对处理的入向 ARP 数据包数量不设上限
步骤 5	**exit**	返回全局配置模式

步骤 6	使用以下 show 命令: **• errdisable detect cause arp-inspection** **• errdisable recovery cause arp-inspection** **• errdisable recovery interval** *interval*	（可选）启用动态 ARP 监测错误禁用状态的错误恢复，配置动态 ARP 监测恢复机制的参数。 默认情况下，恢复被禁用，恢复间隔是 300 秒。 对于 **interval** *interval*，以秒的形式指定从错误禁用状态恢复的时间。范围从 30 到 86400
步骤 7	**exit**	返回特权 EXEC 模式
步骤 8	使用以下 show 命令: **• show ip arp inspection interfaces** **• show errdisable recovery**	验证设置
步骤 9	**show running-config** **示例：** `Device#` **show running-config**	验证配置的条目
步骤 10	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中

执行动态 ARP 监测验证检查

动态 ARP 监测截获、记录并丢弃含有非法 IP-MAC 地址映射信息的 ARP 包。管理员可以配置交换机对目的 MAC 地址、发送方和目标的 IP 地址以及源 MAC 地址进行额外的检查。
按照以下步骤配置对入向 ARP 包的特定检查。此过程是可选的。

总步骤

1. enable

2. configure terminal

3. ip arp inspection validate {[src-mac] [dst-mac] [ip]}

4. exit

5. show ip arp inspection vlan *vlan-range*

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 3	**ip arp inspection validate {[src-mac] [dst-mac] [ip]}**	对入向 ARP 包执行特定的检查。默认情况下，这些检查不执行。

		关键字的含义如下： • 对于 **src-mac**，把以太网报头中的源 MAC 地址与 ARP 报文中的发送方 MAC 地址进行比较。此检查对 ARP 请求和 ARP 应答都会进行。启用时，MAC 地址不同的数据包会被分类为非法数据包并被丢弃。 • 对于 **dst-mac**，把以太网报头中的目的 MAC 地址与 ARP 报文中的目标 MAC 地址进行比较。此检查对 ARP 应答进行。启用时，MAC 地址不同的数据包会被分类为非法数据包并被丢弃。 • 对于 **ip**，检查 ARP 报文体中的非法 IP 地址。这些地址包括 0.0.0.0,255.255.255.255 以及所有的 IP 组播地址。对于所有的 ARP 请求和应答发送方 IP 地址都会被检查，对于 ARP 应答检查目标 IP 地址。 管理员必须至少指定一个关键字。每条命令都会覆盖之前命令的配置，即如果一条命令启用了 src 和 dstmac 验证，而第二条命令仅启用了 IP 验证，src 和 dstmac 的验证都会因第二条命令而被禁用
步骤 4	**exit**	返回特权 EXEC 模式
步骤 5	**show ip arp inspection vlan** *vlan-range*	验证设置
步骤 6	**show running-config** **示例：** `Device#` **show running-config**	验证配置的条目
步骤 7	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中

监控 DAI

使用以下命令监控 DAI：

命令	描述

clear ip arp inspection statistics	清除动态 ARP 监测统计数据。
show ip arp inspection statistics [**vlan** *vlan-range*]	显示特定 VLAN 的统计数据，包括转发、丢弃、MAC 验证失败、IP 验证失败、ACL 允许及拒绝以及 DHCP 允许及拒绝的数据包信息。如果不指定 VLAN 或者指定了 VLAN 范围，则只显示启用了动态 ARP 监测（active）的 VLAN 的信息
clear ip arp inspection log	清除动态 ARP 监测日志缓存
show ip arp inspection log	显示动态 ARP 监测日志缓存的配置及内容

对于 **show ip arp inspection statistics** 命令，交换机会递增每个可信动态 ARP 监测端口上转发的 ARP 请求和应答包的数量。交换机会递增被 ACL 或 DHCP 允许的数据包数量。对于每个被源 MAC、目的 MAC 或 IP 验证检查拒绝的数据包，交换机会增加对应的计数。

验证 DAI 配置

使用以下命令显示及验证 DAI 配置：

命令	描述
show arp access-list [*acl-name*]	显示 ARP ACL 的详细信息
show ip arp inspection interfaces [interface-id]	显示特定接口或者所有接口的可信状态以及 ARP 包限速设置
show ip arp inspection vlan *vlan-range*	显示特定 VLAN 的动态 ARP 监测配置及运行状态。如果不指定 VLAN 或者指定了 VLAN 范围，则只显示启用了动态 ARP 监测（active）的 VLAN 的信息

其他参考资料

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS	http://www.icntnetworks.com

源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	

配置 IEEE 802.1x 基于端口的认证

本章描述如何配置 IEEE 802.1x 基于端口的认证。IEEE 802.1x 认证阻止未认证的设备（客户端）获取网络访问权限。除非另有说明，交换机一词表示独立交换机或交换机堆栈。

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于 802.1x 基于端口认证的信息

802.1x 标准定义了一个基于客户端服务器的访问控制及认证协议，可以阻止未被授权的客户端通过可公开访问的端口连接 LAN。在提供交换机或者 LAN 的服务之前，认证服务器会对每个连接到交换机端口的客户端进行认证。

注释： TACACS 不支持 802.1x 认证。

在客户端被认证之前，802.1x 访问控制只允许基于 LAN 的可扩展认证协议（Extensible Authentication Protocol overLAN，EAPOL）、Inspur 发现协议（Inspur Discovery Protocol，IDP）以及生成树协议（Spanning Tree Protocol，STP）的流量通过客户端连接的端口。认证成功后，正常流量才可以通过端口。

注释： 使用 **show platform software trace message smd** 命令查看 RADIUS 以及 AAA 的调试日志信息。更多信息参见 *Cisco IOS XE Denali 16.1.1 命令参考指南*的 Trace 命令章节。

基于端口的认证过程

要配置 IEEE 802.1x 基于端口的认证功能，必须启用认证、授权和审计（authentication, authorization, accounting，AAA）并指定认证方式列表。方式列表描述了认证用户时查询认证方式的顺序。

AAA 过程从认证开始。当启用了 802.1x 基于端口的认证且客户端支持兼容 802.1x 的客户端软件时，会发生以下事件：

- 如果客户端身份合法且 802.1x 认证成功，交换机允许客户端访问网络；
- 如果 802.1x 认证等待 EAPOL 消息交换超时且启用了 MAC 旁路认证，交换机可以使用客户端的 MAC 地址进行认证。如果客户端的 MAC 地址合法且认证成功，交换机允许客户端网络。如果因客户端的 MAC 地址非法而认证失败，在配置了访客 VLAN 的情况下交换机会给客户端分配一个访客 VLAN，提供有限的访问服务；
- 如果交换机在 802.1x 兼容的客户端上得到了非法的身份，在配置了首先 VLAN 的情况下交换机会给客户端分配一个受限 VLAN，提供有限的访问服务；
- 如果 RADIUS 认证服务器不可用且启用了不可访问旁路认证，交换机会允许客户端访问网络，并把置为临界认证状态的端口放在 RADIUS 配置或用户指定的接入 VLAN 中。

注释： 不可访问旁路认证也被称为临界认证或 AAA 失败策略。

如果在端口上启用了多域认证（Multi Domain Authentication，MDA），也可以使用以上流程加上适用于语音认证的例外情况进行认证。

下图显示了认证的过程。

图 120：认证流程图

Start	开始
Done	结束
Yes	是
No	否
Is the client IEEE 802.1x capable?	客户端是否兼容 IEEE 802.1x？
IEEE 802.1x authentication process times out.	IEEE 802.1x 认证过程超时。
Is MAC authentication bypass enabled?	是否启用了 MAC 旁路认证？
Use MAC authentication bypass.[1]	使用 MAC 旁路认证。[1]
Client MAC address identity is valid.	客户端 MAC 地址身份合法。
Client MAC address identity is invalid.	客户端 MAC 地址身份非法。
Assign the port to a VLAN.	分配端口给 VLAN。
Assign the port to a guest VLAN.[1]	分配端口给访客 VLAN。[1]
The switch gets an EAPOL message, and the EAPOL message exchange begins.	交换机收到了 EAPOL 消息，EAPOL 消息交换开始。
Start IEEE 802.1x port-based authentication.	开始 IEEE 802.1x 基于端口的认证。
User does not have a certificate but the system previously logged on to the network using a computer certificate.	用户没有证书，但是系统之前使用计算机证书登录过网络。
Assign the port to a restricted VLAN.	分配端口给受限 VLAN。
Client identity is invalid.	客户端身份非法。
Client identity is valid.	客户端身份合法。
All authentication servers are down.	所有认证服务器均故障。

Use inaccessible authentication bypass(critical authentication) to assign the critical port to a VLAN.	使用不可访问旁路认证（临界认证）分配临界端口给 VLAN。
1 = This occurs if the switch does not detect EAPOL packets from the client.	1 =在交换机没有从客户端检测到 EAPOL 包时发生。

以下情况之一发生时交换机重新认证客户端：

- 启用了周期性重新认证，且重新认证计时器超时。
  用户可以配置重新认证计时器使用交换机特定的值或者基于 RADIUS 服务器的值计时。配置使用 RADIUS 服务器进行 802.1x 认证后，交换机基于会话超时 RADIUS 属性（属性[27]）以及终止操作 RADIUS 属性（属性[29]）设置计时器。
  会话超时 RADIUS 属性（属性[27]）指定了进行重新认证的经过时间。
  终止操作 RADIUS 属性（属性[29]）指定了在重新认证过程中需要采取的操作。这些操作是*初始化*以及*重新认证*。设置*初始化*操作时（该属性值为*默认*），802.1x 会话结束，且重新认证过程中连接性丢失。设置*重新认证*操作时（该属性值为 RADIUS 请求），会话在重新认证过程中不受影响；
- 可以输入特权 EXEC 命令 **dot1x re-authenticate interface** *interface-id* 手动重新认证客户端。

基于端口的认证初始化及消息交换

在 802.1x 认证期间，交换机或客户端可以发起认证。如果使用接口配置命令 **authentication port-control auto** 启用了端口认证，交换机会在链路状态从 down 变为 up 时发起认证，或在端口保持 up 且未认证状态时周期性地发起认证。交换机给客户端发送 EAP 请求/身份数据帧请求其身份。接收到数据帧之后，客户端会使用 EAP 应答/身份帧回应。

然而，如果在启动期间客户端没有从交换机上收到 EAP 请求/身份帧，客户端可通过发送 EAP 开始帧发起认证，这会促使交换机请求客户端的身份。

注释： 如果 802.1x 认证未在网络接入设备上启用或设备不支持，来自客户端的 EAPOL 帧会被丢弃。如果客户端三次尝试接收 EAP 请求/身份失败，客户端会当作端口在已认证的状态来发送数据帧。处于已认证状态的端口相当于客户端已经被成功认证。

客户端提供自己的身份信息后，交换机开始作为中介的角色，在客户端和认证服务器之间传递 EAP 帧直到认证成功或者失败。如果认证成功，交换机端口变为已认证状态。如果认证失败，可以重新尝试认证，端口可能被分配给提供有限的接入服务的 VLAN，也可能被授予网络访问权限。

具体的 EAP 交换过程取决于使用的认证方式。

下图显示了客户端和 RADIUS 服务器使用一次性密码（One-Time-Password，OTP）认证方式时由客户端发起的消息交换过程。

图 121：消息交换

Client	客户端
Authentication server(RADIUS)	认证服务器（RADIUS）
EAPOL-Start	EAPOL 开始
EAPOL-Request/Identity	EAPOL 请求/身份
EAPOL-Response/Identity	EAPOL 应答/身份
EAPOL-Request/OTP	EAPOL 请求/OTP
EAPOL-Response/OTP	EAPOL 应答/OTP

EAPOL-Success	EAPOL 成功
EAPOL-Logoff	EAPOL 登出
RADIUS Access-Request	RADIUS 访问请求
RADIUS Access-Challenge	RADIUS 访问挑战
RADIUS Access-Request	RADIUS 访问请求
RADIUS Access-Accept	RADIUS 访问接受
Port Authorized	端口已认证
Port Unauthorized	端口未认证

如果等待 EAPOL 消息交换时 802.1x 认证超时且启用了 MAC 旁路认证，交换机可以在从客户端检测到以太网数据包时授权客户端。交换机使用客户端的 MAC 地址作为其身份，并把此信息包含在发往 RADIUS 服务器的 RADIUS 访问请求帧中。在服务器给交换机发送了 RADIUS 访问接受帧之后（授权成功），端口变为已认证。如果授权失败且指定了访客 VLAN，交换机会把端口分配给访客 VLAN。如果交换机等待以太网数据包时检测到了 EAPOL 包，交换机会停止 MAC 旁路认证过程并开始 802.1x 认证。

下图显示了 MAC 旁路认证过程中的消息交换过程。

图 122：MAC 旁路认证过程中的消息交换

Client	客户端
Authentication server(RADIUS)	认证服务器（RADIUS）
Switch	交换机
EAPOL-Request/Identity	EAPOL 请求/身份
RADIUS Access-Request	RADIUS 访问请求
RADIUS Access-Accept	RADIUS 访问接受
Ethernetpacket	以太网数据包

基于端口认证的认证管理器

基于端口的认证方式

表 145:802.1x 特性

认证方式	模式			
	单主机	多主机	MDA	多认证
802.1x	VLAN 分配 基于用户的 ACL 过滤 ID 属性 可下载的 ACL[16] 重定向 URL	VLAN 分配	VLAN 分配 基于用户的 ACL 过滤 ID 属性 可下载的 ACL 重定向 URL	VLAN 分配 基于用户的 ACL 过滤 ID 属性 可下载的 ACL 重定向 URL
MAC 旁路认证	VLAN 分配 基于用户的 ACL 过滤 ID 属性 可下载的 ACL 重定向 URL	VLAN 分配	VLAN 分配 基于用户的 ACL 过滤 ID 属性 可下载的 ACL 重定向 URL	VLAN 分配 基于用户的 ACL 过滤 ID 属性 可下载的 ACL 重定向 URL

独立网页认证	代理 ACL，过滤 ID 属性，可下载的 ACL			
NAC 二层 IP 验证	过滤 ID 属性 可下载的 ACL 重定向 URL	过滤 ID 属性 可下载的 ACL 重定向 URL	过滤 ID 属性 可下载的 ACL 重定向 URL	过滤 ID 属性 可下载的 ACL 重定向 URL
备用的网页认证[17]	代理 ACL，过滤 ID 属性，可下载的 ACL	代理 ACL，过滤 ID 属性，可下载的 ACL	代理 ACL，过滤 ID 属性，可下载的 ACL	代理 ACL，过滤 ID 属性，可下载的 ACL

[16] InspurINOS 12.2(50)SE 及之后版本支持。

[17] 对于不支持 802.1x 认证的客户端使用。

基于用户的 ACL 和过滤 ID 属性

注释： 在 ACL 中源只能设置为 **any**。

注释： 对于为多主机模式配置的 ACL，声明的源部分必须是 *any*（比如 **permit icmp** *any* **host 10.10.1.1**）。

对于定义的任意 ACL，源部分必须指定为 **any**。否则，ACL 不能被应用且认证会失败。单主机模式是唯一的例外，可以向后兼容。

在启用 MDA 或多认证的端口上可以认证多台主机。应用于一台主机的 ACL 策略不会影响其他主机的流量。如果一台主机在一个多主机端口上进行了认证，而其他主机没有认证就获得了网络访问权限，通过设置源地址为 any 可以把用于第一台主机的 ACL 策略应用到其他连接的主机上。

基于端口认证管理器的 CLI 命令

认证管理器的接口配置命令管理所有的认证方式，比如 802.1x、MAC 旁路认证以及网页认证。认证管理器的命令决定应用到连网主机上的认证方式的优先级以及顺序。

认证管理器的命令控制通用的特征特性，比如主机模式、违反模式以及认证计时器。通用的认证命令包括接口配置命令 **authentication host-mode**， **authentication violation**， 以及 **authentication timer**。

802.1x特性的命令以**dot1x**关键字开始。例如，接口配置命令**authentication port-controlauto**在接口上启用认证。然而，全局配置命令**dot1xsystem-authentication control**只能全局地启用或禁用802.1x认证。

注释： 如果全局禁用了802.1x认证，其他认证方式仍会在端口上启用，如网页认证。

authentication manager 命令与以前的 802.1x 命令功能相同。

当过滤掉由认证管理器生成的详细系统消息时，被过滤的内容通常与认证成功有关。也可以过滤 802.1x 认证以及 MAB 认证的详细消息。每种认证方式都有独立的配置命令：

- 全局配置命令 **no authentication logging verbose** 过滤来自认证管理器的详细消息；
- 全局配置命令 **no dot1x logging verbose** 过滤 802.1x 认证的详细消息；
- 全局配置命令 **no mab logging verbose** 过滤 MAC 旁路认证（MAB）的详细消息。

表 146：认证管理器命令及以前的 802.1x 命令

Inspur	**INOSRelease**	**Inspur**	**INOSRelease**	**描述**

12.2(50)SE 及之后版本的认证管理器命令	12.2(46)SE 及之前版本的等同 802.1x 命令	
authentication control-direction {both | in}	**dot1x control-direction {both | in}**	启用 802.1x 认证以及局域网唤醒（wake-on-LAN，WoL）特性，并配置单向或者双向端口控制
authentication event	**dot1x auth-fail vlan** **dot1x critical (interface configuration)** **dot1x guest-vlan6**	在端口上启用受限 VLAN。 启用不可访问旁路认证特性。 指定一个活跃的 VLAN 作为 802.1x 的访客 VLAN
authentication fallback *fallback-profile*	**dot1x fallback** *fallback-profile*	配置端口对于不支持 802.1x 认证的客户端使用网页认证作为备用方式
authentication host-mode [multi-auth | multi-domain | multi-host | single-host]	**dot1x host-mode {single-host | multi-host | multi-domain}**	允许 802.1x 授权的端口上有一个或多个主机（客户端）
authentication order	**mab**	提供灵活定义使用的认证方式的功能
authentication periodic	**dot1x reauthentication**	启用客户端的周期性重新认证
authentication port-control {auto | force-authorized | force-un authorized}	**dot1x port-control {auto | force-authorized | force-unauthorized}**	允许人工控制端口的认证状态
authentication timer	**dot1x timeout**	设置 802.1x 的计时器
authentication violation {protect | restrict | shutdown}	**dot1x violation-mode {shutdown | restrict | protect}**	配置新设备连接到端口或者在端口已连接了最大数量的设备之后有新设备连接到端口时的违反模式

已认证以及未认证状态的端口

在 802.1x 认证过程中，交换机可以根据交换机端口的状态授予一个客户端访问网络的权限。端口开始的状态是*未认证*。在此状态中的未被配置为语音 VLAN 的端口会禁止除 802.1x 认证、CDP 以及 STP 包之外的所有入向及出向流量。当客户端被成功授权时，端口的状态变为*已认证*状态，允许客户端的所有流量正常流通。如果端口被配置为语音 VLAN 端口，在客户端被成功认证之前，端口允许 VoIP 流量以及 802.1x 协议数据包通过。

注释： 不支持 CDP 旁路功能，这可能导致端口进入错误禁用状态。

如果不支持 802.1x 认证的客户端连接到了一个未认证的 802.1x 端口，交换机会请求客户端的身份。在此情况下，客户端不会响应请求，端口会保持未认证状态，而客户端不被授予网络访问权限。

相反，当启用 802.1x 的客户端连接到了不运行 802.1x 标准的端口时，客户端会发送 EAPOL 开始帧以发起认证过程。在未收到应答的情况下，客户端会发送固定次数的请求。因为仍未

接收到应答，客户端会把端口当作是已认证的状态并开始发送数据帧。

管理员可以使用接口配置命令 **authentication port-control** 及以下关键字控制端口的认证状态：

- **force-authorized**——禁用 802.1x 认证，使端口无需经过认证交换过程而直接变为已认证状态。端口不对客户端进行基于 802.1x 的认证并正常收发流量。这是默认的设置；
- **force-unauthorized**——使端口保持未认证状态，忽略客户端的所有认证尝试。交换机不能通过该端口为客户端提供认证服务；
- **auto**——启用 802.1x 认证，让端口的开始状态是未认证，只允许 EAPOL 数据帧通过端口收发。认证过程在端口链路状态从 down 变为 up 时或者在收到 EAPOL 开始帧的时候开始。交换机请求客户端的身份并在客户端和认证服务器之间传递认证消息。每个尝试访问网络的客户端都会被交换机使用客户端的 MAC 地址唯一标识。

如果客户端认证成功（从认证服务器收到了接受帧），端口状态会变为已认证，且所有来自已认证客户端的数据帧都被允许通过端口。如果认证失败，端口会保持在未认证的状态，但可以重新尝试认证。如果认证服务器不可达，交换机可以重新发送请求。如果在特定的尝试次数之后还是没有接收到来自服务器的应答，认证失败，且不授予网络访问权限。

客户端登出时，会发送一个 EAPOL 登出消息，使交换机的端口变为未认证状态。

如果端口的链路状态从 up 变为 down，或者收到了 EAPOL 登出帧，端口都会返回未认证的状态。

基于端口的认证以及交换机堆栈

如果一台交换机被添加进交换机堆栈或被从交换机堆栈移除，只要堆栈与 RADIUS 服务器保持 IP 连通性，802.1x 认证就不受影响。以上说明也适用于堆栈主用设备被从交换机堆栈移除的情况。如果堆栈主用设备故障，堆栈成员会通过选举过程成为新的堆栈主用设备，且 802.1x 认证过程照常进行。

如果因为连接到服务器的交换机被移除或故障，导致了与 RADIUS 服务器的 IP 连通性中断，会发生以下事件：

- 已经被认证且没有启用周期性重新认证的端口会保持已认证的状态。无需与 RADIUS 进行通信；
- 已经被认证且启用了周期性重新认证（使用全局配置命令**dot1xre-authentication**）的端口会在重新认证发生时认证失败。在重新认证过程中端口会返回未认证状态。需要与 RADIUS服务器进行通信。

  对于正在进行的认证，认证会因为与服务器无连通性而立即失败。

如果发生故障的交换机启动并重新加入了交换机堆栈，认证可能失败也可能成功，这取决于交换机的启动时间以及尝试认证的时候与RADIUS服务器的连通性是否已经重建。

为了避免失去与RADIUS服务器的连通性，应确保存在冗余连接。例如，管理员可以让 RADIUS服务器有到堆栈主用设备和堆栈成员的冗余连接，这样如果堆栈主用设备故障，交换机堆栈仍然有到RADIUS服务器的连通性。

802.1x 主机模式

管理员可以把端口配置成单主机模式或多主机模式。在单主机模式中，只有一台客户端可以连接到启用了 802.1x 的交换机端口。交换机可以通过发送 EAPOL 帧或者在端口链路状态变为 up 状态时发现客户端。如果客户端离开或者被另一个客户端代替，交换机会把端口的链

路状态变为 down，且端口返回未认证状态。
在多主机模式中，可以把多台主机连接到一个启用了 802.1x 的端口。在此模式中，只需有一台连接的客户端被认证，所有客户端都可以被授予网络访问权限。如果端口变为未认证状态（重新认证失败或收到了 EAPOL 登出消息），交换机会拒绝所有连接主机的网络访问。
图 123：多主机模式示例

Wireless clients	无线客户端
Access point	接入点
Authentication server(RADIUS)	认证服务器（RADIUS）

注释： 对于所有主机模式，配置基于端口的认证时，认证之前链路协议保持为 up。
交换机支持多域认证（multidomain authentication，MDA），允许数据设备和语音设备（如 Inspur 或非 Inspur 的 IP 电话）同时连接到相同的交换机端口。

802.1x 多认证模式

多认证（multiple-authentication，multiauth）模式允许数据 VLAN 中有多个被认证的客户端。每台主机被独立认证。如果配置了语音 VLAN，此模式也允许 VLAN 上有一个客户端（如果端口检测到了其他的语音客户端，它们会被端口丢弃，但此时不会发生违反错误）。
如果启用了 802.1x 的端口连接了集线器或者无线接入点，每个连接的设备都必须进行认证。对于非 802.1x 设备，可以使用 MAC 旁路认证或者网页认证作为备用的主机认证方式，对单个端口上的不同主机使用不同的认证方式。
多认证端口可以认证的数据主机数量没有限制。然而，如果配置了语音 VLAN，只允许有一台语音设备。因为没有定义主机数量限制，也就不会触发违反操作，如果发现了第二台语音设备，其流量会被静默地丢弃。对于语音 VLAN 上的 MDA 功能，多认证模式会根据从认证服务器收到的 VSA 把已认证的设备分配到数据 VLAN 或者语音 VLAN。
注释： 端口在多认证模式时，访客 VLAN 以及认证失败的 VLAN 特性不被激活。
在以下情况中，多认证模式可以分配 RADIUS 服务器提供的 VLAN：

- 主机是端口上认证的第一台主机，且 RADIUS 服务器提供了 VLAN 信息；
- 后续认证主机使用的 VLAN 与运行的 VLAN 相同；
- 端口上被认证的主机没有 VLAN 分配信息，且后续主机也没有 VLAN 分配，或者其 VLAN 信息与运行的 VLAN 相同；
- 端口上第一台被认证的主机有一组 VLAN 分配信息，且后续主机没有 VLAN 分配，或者其 VLAN 组信息与端口的 VLAN 组信息相同。后续主机必须与第一台主机使用 VLAN 组中相同的 VLAN。如果使用了 VLAN 列表，所有主机都要服从 VLAN 列表中定义的条件；
- 多认证端口上只支持一个语音 VLAN 的分配；
- 在给端口上的主机分配了 VLAN 之后，后续主机必须有相同的 VLAN 信息，否则就会被拒绝访问端口；
- 在多认证模式中不能配置访客 VLAN 或者认证失败 VLAN；
- 多认证模式下临界认证 VLAN 的行为不变。当主机尝试认证而服务器不可达时，所有已认证的主机都会被重新初始化到配置的 VLAN 中。

基于用户 VLAN 分配的多认证

注释： 此特性只在运行 LAN Base 镜像的 Inspur 2960X 交换机上支持。

基于用户 VLAN 分配的多认证特性允许在拥有一个配置的接入 VLAN 的端口上，根据分配给端口上客户端的 VLAN 创建多个运行的接入 VLAN。配置为接入端口的交换机端口不进行 dot1q 标记，其上的所有 VLAN 的流量都与数据域关联，且这些 VLAN 被当做本征 VLAN。

每个多认证端口的主机数量是 8 个，然而也可以有更多的主机。

注释： 基于用户 VLAN 分配的多认证特性不支持语音 VLAN。端口上所有语音域中的客户端都是用一个 VLAN。

以下是基于用户 VLAN 分配的多认证情景：

情景一

集线器连接到接入端口，且端口配置了接入 VLAN（V0）。

主机（H1）通过集线器分配了 VLAN（V1）。端口的运行 VLAN 被改为 V1。此行为与单主机或多域认证端口相似。

当第二台主机（H2）连接集线器且被分配了 VLAN（V2），端口将有两个运行的 VLAN（V1 和 V2）。如果 H1 和 H2 发送了未打标记的入向流量，H1 的流量会被映射到 VLAN（V1），而 H2 的流量会被映射到 VLAN（V2），所有该端口的出向 VLAN（V1）和 VLAN（V2）的流量都不会被打标记。

如果两台主机 H1 和 H2 都登出，或者会话因故被移除，VLAN（V1）和 VLAN（V2）会被从端口上移除，且端口恢复为配置的 VLAN（V0）。

情景二

集线器连接到接入端口，且端口配置了接入 VLAN（V0）。

主机（H1）通过集线器分配了 VLAN（V1）。端口的运行 VLAN 被改为 V1。

当第二台主机（H2）连接到集线器，被授权且没有显式的 VLAN 策略，H2 希望使用恢复到端口上的配置的 VLAN（V0）。所有从 VLAN（V0）和 VLAN（V1）发出的出向流量都不被打标记。

如果主机（H2）登出或因故会话被移除，配置的 VLAN（V0）会被从端口上移除，而 VLAN（V1）会成为端口上的唯一运行 VLAN。

情景三

集线器连接到开放模式的接入端口，且端口配置了接入 VLAN（V0）。

主机（H1）通过集线器分配了 VLAN（V1）。端口的运行 VLAN 被改为 V1。当第二台主机（H2）连接上且保持未认证时，因为使用开放模式，其仍然可以访问运行 VLAN（V1）。

如果主机（H1）登出或因故会话被移除，VLAN（V1）被从端口上移除，且主机（H2）被分配到 VLAN（V0）。

注释： 开放模式以及 VLAN 分配的组合对主机（H2）有负面影响，因为其 IP 地址子网对应于 VLAN（V1）。

基于用户 VLAN 分配的多认证的限制

在基于用户 VLAN 分配的多认证特性中，一个端口上来自多个 VLAN 的出向流量不会被打标记，主机会收到发给其他主机的流量。这可能对广播以组播流量造成问题。

- **IPv4 ARP**：主机会接收到来自其他子网的 ARP 包。如果端口上有两个活跃的在不同虚拟路由转发（Virtual Routing and Forwarding，VRF）表的子网，且子网使用了重叠的 IP 地址范围，就会发生问题。主机的 ARP 缓存可能有非法的条目；
- **IPv6 控制包**：在 IPv6 中，路由器通告（Router Advertisements，RA）会被不应接收的主机

处理。当 VLAN 中的一台主机接收到了来自不同 VLAN 的 RA，主机会给自己分配不正确的 IPv6 地址。这样的主机无法访问网络。

解决方法是启用 IPv6 首跳安全功能，让广播的 ICMPv6 包转化为单播包并从启用了多认证的端口发出。此时的数据包会复制给多认证端口上属于 VLAN 的每个客户端，且目的 MAC 地址会被设置为每个客户端的地址。如果端口有一个 VLAN，则 ICMPv6 包正常广播；

- **IP 组播：**如果 VLAN 中的主机加入了组播组，发往组播组的流量会被复制给不同的 VLAN。如果一个多认证端口上的两个不同 VLAN 的主机加入了一个组播组，每个组播包会从这个端口上发出两份。

MAC 移动

如果一个 MAC 地址在一个交换机端口上被认证，这个地址就不被允许出现该交换机上另一个启用了认证管理器的端口上。如果交换机在另一个启用了认证管理器的端口上检测到了相同的 MAC 地址，该地址不被允许。

有一些情况下 MAC 地址可能需要从一个端口移动到相同交换机的另一个端口。比如，当认证的主机和交换机端口之间有另一台设备时（如集线器或者 IP 电话），管理员可能希望断开主机与另一个台设备的连接并直接连接到相同交换机的另一个端口上。

可以全局启用 MAC 移动特性，设备会在新端口上重新被认证。当主机移动到第二个端口上时，第一个端口上的会话会被删除，而主机会在新端口上重新被认证。MAC 移动在所有主机模式中都支持（被认证的主机可以移动到交换机的任意端口上，无论该端口启用了何种主机模式）。当 MAC 地址从一个端口移动到另一个，交换机会结束原始端口上的认证会话，并在新端口上发起新的认证过程。MAC 移动特性对于语音和数据主机都适用。

注释： 在开发认证模式中，MAC 地址可以立即从原始端口移动到新端口上，而无需在新端口上进行认证。

MAC 替换

MAC 替换特性可以用来解决主机尝试连接到之前认证了另一台主机的端口的违规情况。

注释： 此特性不适用于多认证模式的端口，因为违规情况在该模式中不会被触发。此特性不适用于多主机模式的端口，因为在该模式中，只要求认证第一台主机。

如果配置了接口配置命令 **authentication violation** 以及 **replace** 关键字，多域模式端口的认证过程如下：

- 在有已认证 MAC 地址的端口上收到了新的 MAC 地址；
- 认证管理器会用新的 MAC 地址替换端口上当前数据主机的 MAC 地址；
- 认证管理器会发起新 MAC 地址的认证过程；
- 如果认证管理器确定新主机是语音主机，原始的语音主机会被移除；

如果端口为开放认证模式，新的 MAC 地址会立即被加入 MAC 地址表中。

802.1x 审计

802.1x 标准定义了如何对用户的网络访问进行认证和授权，但不记录网络的使用情况。802.1x

审计默认被禁用。可以启用 802.1x 审计功能监控启用了 802.1x 的端口活动：

- 用户成功认证
- 用户登出
- 链路 down
- 重新认证成功
- 重新认证失败

交换机不会记录 802.1x 的审计信息。它会把这些信息发给 RADIUS 服务器，必须配置服务器记录审计消息。

802.1x 审计属性-值对

发送给 RADIUS 服务器的信息以属性-值（Attribute-Value，AV）对的形式展示。这些 AV 对给不同的应用提供数据（比如，审计程序可能需要 RADIUS 包中 Acct-Input-Octets 或 Acct-Output-Octets 属性的信息）

AV 对由配置了 802.1x 审计的交换机自动发送。交换机会发送三种类型的 RADIUS 审计包：

- 开始——在新用户会话开始时发送
- 中间——在现有会话更新时发送
- 停止——在会话终止时发送

注释： 使用命令 **show platform software trace message smd** 查看 RADIUS 和 AAA 的调试信息。更多信息参见 *Cisco IOS XE Denali 16.1.1 命令参考指南*的 Trace 命令章节。

下表列出了 AV 对及何时由交换机发出。

表 147：审计 AV 对

属性名	AV 对名	开始	中间	停止
属性[1]	User-Name	总是	总是	总是
属性[4]	NAS-IP-Address	总是	总是	总是
属性[5]	NAS-Port	总是	总是	总是
属性[8]	Framed-IP-Address	从不	有时 [18]	有时
属性[30]	Called-Station-ID	总是	总是	总是
属性[31]	Calling-Station-ID	总是	总是	总是
属性[40]	Acct-Status-Type	总是	总是	总是
属性[41]	Acct-Delay-Time	总是	总是	总是
属性[42]	Acct-Input-Octets	从不	总是	总是
属性[43]	Acct-Output-Octets	从不	总是	总是
属性[47]	Acct-Input-Packets	从不	总是	总是
属性[48]	Acct-Output-Packets	从不	总是	总是
属性[44]	Acct-Session-ID	总是	总是	总是
属性[45]	Acct-Authentic	总是	总是	总是
属性[46]	Acct-Session-Time	从不	总是	总是
属性[49]	Acct-Terminate-Cause	从不	从不	总是
属性[61]	NAS-Port-Type	总是	总是	总是

[18] Framed-IP-Address AV 对在配置了合法的静态 IP 地址或当 DHCP 侦听绑定表中存在主机的 DHCP 绑定时发送。

802.1x 就绪状态检查

802.1x 就绪状态检查监控交换机所有端口上的 802.1x 活动，且显示连接到端口的支持 802.1x 的设备的信息。可以使用此特性确定连接到交换机端口的设备是否兼容 802.1x。对于不支持 802.1x 的设备可以使用如 MAC 旁路认证以及网页认证等其他认证方式。
此特性只在客户端支持使用 NOTIFY EAP 通知包查询时有效。客户端必须在 802.1x 超时时间之内应答。

交换机到 RADIUS 服务器的通信

RADIUS 安全服务器的标识方式有主机名或 IP 地址、主机名及特定的 UDP 端口号以及 IP 地址以及特定的 UDP 端口号。IP 地址和 UDP 端口号的组合是唯一的标识符，允许把 RADIUS 请求发给相同 IP 地址服务器上的多个 UDP 端口。可以为相同 RADIUS 服务器的相同服务（如认证）配置两个不同的主机条目，第二个条目作为第一个条目的备用项。RADIUS 主机条目会按照配置的顺序被尝试访问。

进行 VLAN 分配的 802.1x 认证

交换机支持进行 VLAN 分配的 802.1x 认证。在端口 802.1x 认证成功后，RADIUS 服务器会发送 VLAN 分配信息来配置交换机端口。RADIUS 服务器数据库维护着用户名到 VLAN 的映射，基于连接到交换机端口的客户端用户名分配 VLAN。可以使用此特性限制特定用户的网络访问。
Inspur INOS 12.2(37)SE 版本支持多域主机模式中的语音设备认证。在 Inspur INOS 12.2(40)SE 及之后版本中，当语音设备被授权且 RADIUS 服务器返回了授权的 VLAN 时，会配置端口上的语音 VLAN 为指定的 VLAN 并收发数据包。在启用多域认证（MDA）的端口上，语音 VLAN 的分配过程与数据 VLAN 相同。
在交换机和 RADIUS 服务器上配置时，进行 VLAN 分配的 802.1x 认证特征如下：

- 如果 RADIUS 服务器没有提供 VLAN 或者禁用了 802.1x 认证，认证成功后端口被配置在其所属的接入 VLAN 中。接入 VLAN 是分配给接入端口的 VLAN。在这个端口上收发的所有数据包都属于此 VLAN；
- 如果启用了 802.1x 认证但是来自 RADIUS 服务器的 VLAN 信息不合法，认证失败且配置的 VLAN 保持使用。这避免了端口因为配置错误意外地出现在不合适的 VLAN 中。
  配置错误的情况可能包括为被路由端口指定了 VLAN、异常的 VLAN ID、不存在或内部（被路由端口）VLAN ID、RSPAN VLAN 以及关闭或停用的 VLAN。在多域主机端口上，配置错误也可能包括尝试分配与配置或指定的语音 VLAN ID 相同的数据 VLAN（反之亦然）；
- 如果启用了 802.1x 认证，且所有来自 RADIUS 服务器的信息都是合法的，被授权的设备会在认证后被置于指定的 VLAN 中；
- 如果在 802.1x 端口上启用了多主机模式，所有主机都会被置入与第一台认证主机相同的 VLAN（有 RADIUS 服务器指定）中；
- 启用端口安全特性不会影响 RADIUS 服务器分配的 VLAN 行为；
- 如果端口上禁用了 802.1x 认证，端口会恢复配置的接入 VLAN 以及配置的语音 VLAN。
- 如果 802.1x 端口被认证且被置入了 RADIUS 服务器分配的 VLAN 中，任何对端口接入

VLAN 配置的更改都不会生效。在多域主机的场景中，以上规则适用于完全授权语音设备，但包含例外情况。

- 如果一台设备的配置变化导致其 VLAN 与其他设备配置或分配的 VLAN 相同，那么该端口上所有设备的授权都会被终止，且多域主机模式被禁用，直到恢复了数据和语音设备配置的 VLAN 不相同的合法配置；
- 如果一台语音设备被授权且使用下载的语音 VLAN，移除语音 VLAN 配置或者将配置值修改为 dot1p 或未标记都会导致语音设备变为未授权且多域主机模式被禁用。

当端口在强制授权、强制未授权、未授权或关闭状态时，端口会被置入配置的接入 VLAN。

进行 VLAN 分配的 802.1x 认证特性不支持中继端口、动态端口以及通过 VLAN 成员策略服务器（VLAN Membership Policy Server，VMPS）进行动态接入分配的端口。

要配置 VLAN 分配，用户需要执行以下操作：

- 使用 **network** 关键字启用 AAA 认证，允许 RADIUS 服务器配置接口；
- 启用 802.1x 认证（在接入端口上配置 802.1x 认证时 VLAN 分配特性会被自动启用）；
- 指定 RADIUS 服务器厂商特定的隧道属性。RADIUS 服务器必须给交换机返回以下属性：
  - [64] 隧道类型（Tunnel-Type） = VLAN
  - [65] 隧道介质类型（Tunnel-Medium-Type） = 802
  - [81] 隧道私有组 ID（Tunnel-Private-Group-ID）= VLAN 名或 VLAN ID
  - [83] 隧道偏好（Tunnel-Preference）

  属性[64]必须包含 *VLAN*（类型 13）值。属性[65]必须包含值 *802*（类型 6）。属性[81]指定分配给 IEEE 802.1x 认证用户的 *VLAN 名称或 VLAN ID*。

使用基于用户 ACL 的 802.1x 认证

可以启用基于用户的访问控制列表（ACL），为经过 802.1x 认证的用户提供不同等级的网络访问及服务。当 RADIUS 服务器认证了一个连接到 802.1x 端口的用户时，服务器会基于用户的身份获取 ACL 属性并将其发送给交换机。交换机会把这些属性在用户会话的持续时间内应用到 802.1x 端口上。当会话结束，认证失败或者链路 down 发生时，交换机会移除基于用户的 ACL 配置。交换机不会在运行配置中保存 RADIUS 指定的 ACL。端口为未授权状态时，交换机会把 ACL 从端口移除。

可以在相同的交换机上同时配置路由器 ACL 及输入端口 ACL。然而，端口 ACL 优先于路由器 ACL。如果把输入端口 ACL 应用到属于某个 VLAN 的端口，端口的 ACL 会优先于应用在 VLAN 接口上啊的输入路由器 ACL。在应用了端口 ACL 的端口上收到的入向数据包会被端口 ACL 进行过滤。在其他端口上收到的入向被路由数据包会被路由器 ACL 过滤。出向的被路由数据包会被路由器 ACL 过滤。为了避免配置冲突，管理员应该小心规划存储在 RADIUS 服务器上的用户配置。

RADIUS 支持基于用户的属性，包括厂商特定的属性。这些厂商特定的属性（vendor-specific attribute，VSA）在认证过程中以八位字节格式传递给交换机。对于 VSA 中基于用户的 ACL，入方向是 inacl#<n>，出方向是 outacl#<n>。MAC ACL 仅在入方向支持。交换机仅在入方向支持 VSA，不支持二层端口的出方向端口 ACL。

应只使用扩展 ACL 语法风格定义存储在 RADIUS 服务器上的基于用户的配置。当收到 RADIUS 服务器传输的这些属性时，交换机会按照扩展的命名方式创建 ACL。然而，如果使用 Filter-Id 属性，则可以指向一个标准的 ACL。

可以使用 Filter-Id 属性指定一个已经在交换机上配置了的入向或出向 ACL。此属性包含 ACL 编号，以及入向过滤的*.in* 或者出向过滤的*.out*。如果 RADIUS 服务器不支持*.in* 或*.out* 语法，

访问列表默认被应用为出向 ACL。因为交换机上的 Inspur INOS 仅支持有限数量的访问列表，所以只支持 Filter-Id 属性编号从 1 到 199 以及 1300 到 2699 的 IP ACL（IP 标准及 IP 扩展 ACL）。基于用户 ACL 的最大尺寸的 4000 个 ASCII 字符，但受限于 RADIUS 服务器基于用户 ACL 的最大尺寸。

要配置基于用户的 ACL：

- 启用 AAA 认证
- 启用 AAA 授权，使用 **network** 关键字允许 RADIUS 服务器进行接口配置
- 启用 802.1x 认证
- 在 RADIUS 服务器上配置用户配置及 VSA
- 将 802.1x 端口为单主机模式

注释： 基于用户的 ACL 仅在单主机模式中支持。

使用可下载 ACL 以及重定向 URL 的 802.1x 认证

可以在 802.1x 认证或者 MAC 旁路认证期间从 RADIUS 服务器向交换机下载 ACL 或者重定向 URL。也可以在网页认证期间下载 ACL。

注释： 可下载的 ACL 也被称为 *dACL*。

如果有多台主机被认证且主机在单主机、MDA 或多认证模式中，交换机会把 ACL 中的源地址改为主机的 IP 地址。

可以把 ACL 以及重定向 URL 应用到连接到 802.1x 端口的所有设备上。

如果 802.1x 认证期间没有下载 ACL，交换机会为主机在端口上应用静态默认 ACL。在配置为多认证或 MDA 模式的语音 VLAN 端口上，交换机只会把 ACL 当作授权策略的一部分应用给电话。

从 Inspur INOS 12.2(55)SE 版开始，如果端口上没有静态 ACL，交换机会创建一个动态的认证默认 ACL，在可下载 ACL 应用之前执行策略。

注释： 认证默认 ACL 不会出现在运行配置中。

当在端口上检测到至少有一台主机有授权策略时，授权默认 ACL 会被创建。

当最后一个认证的会话结束时，授权默认 ACL 会被移除。可以使用全局配置命令 **ip access-list extended auth-default-acl** 配置授权默认 ACL。

注释： 单主机模式中的认证默认 ACL 不支持 Inspur 发现协议（CDP）旁路模式。为了支持 CDP 旁路，必须在接口上配置静态 ACL。

802.1x 和 MAB 认证方式支持两种认证模式，*开放（open）*及*闭合（closed）*。如果*闭合*认证模式的端口上没有静态 ACL：

- 认证默认 ACL 会被创建；
- 在执行策略之间，认证默认 ACL 只允许 DHCP 流量；
- 当第一台主机认证时，授权策略被应用且不插入 IP 地址；
- 当检测到第二台主机时，用于第一台主机的策略被刷新，首个及后续会话的策略会插入 IP 地址并执行。

如果*开放*认证模式的端口上没有静态 ACL：

- 开放认证默认 ACL 会被创建，允许所有流量通过；
- 为避免安全漏洞，将执行插入了 IP 地址的策略；
- 网页认证受制于开放认证默认 ACL。

为了控制没有授权策略的主机的访问，可以配置指令。支持的指令值是*开放（open）*和*默认（default）*。配置*开放*指令时，所有流量都被允许。*默认*指令让流量受限于端口提供的接

入权限。可以在AAA服务器的用户配置中配置指令，也可以在交换机上配置。在AAA服务器上配置指令，请使用全局命令**authz-directive =<open/default>**。在交换机上配置指令，请使用全局配置命令**epmaccess-control open**。

注释： 指令的默认值是*默认*。

如果主机在没有配置ACL的端口上使用备用的网页认证：

- 如果端口是开放认证模式，交换机会创建开放认证默认ACL；
- 如果端口时闭合认证模式，交换机会创建认证默认ACL。

备用ACL中的访问控制条目（access control entries，ACE）会被转换为基于用户的条目。如果配置的备用配置不包括备用ACL，主机会受限于与端口关联的认证默认ACL。

注释： 如果网页认证使用了自定义的logo且存储在外部服务器上，端口的ACL必须允许在认证之前访问外部服务器。管理员必须配置静态端口ACL或者更改认证默认ACL，以提供到外部服务器的连接。

用于重定向 URL 的 Inspur 安全 ACS 及属性-值对

交换机使用以下*inspur-av-pair* VSA：

- URL重定向（url-redirect）是HTTP或HTTPS URL；
- URL重定向ACL（url-redirect-acl）是交换机ACL名称或编号。

交换机使用Inspur安全定义ACL属性-值（AV）对来截获终端的HTTP或HTTPS请求。交换机之后将客户端的网页浏览器跳转到特定的重定向地址。Inspur安全ACS上的url-redirect AV包含浏览器被重定向到的URL。url-redirect-acl属性值对包含要进行特定HTTP或HTTPS流量重定向的ACL名称或编号。

注释：

- 匹配ACL中permit ACE的流量被重定向；
- 在交换机上定义URL重定向ACL以及默认端口ACL。

如果认证服务器上为客户端配置了重定向URL，必须在客户端连接的交换机端口上配置默认端口ACL。

用于可下载 ACL 的 Inspur 安全 ACS 及属性-值对

用户可以在 Inspur 安全 ACS 上设置 Inspur 安全定义的 ACL 属性-值（AV）对，使用厂商特定属性（VSA）：RADIUS Inspur AV 对。这一对值使用#ACL#-IP-name-number 属性，指定了 Inspur 安全 ACS 上的可下载 ACL 的名称。

- *name* 是 ACL 的名称
- *number* 是版本号（如 3f783768）

如果认证服务器上为客户端配置了可下载ACL，必须在客户端连接的交换机端口上配置默认端口ACL。

如果在交换机上配置了默认的ACL，且Inspur安全ACS给交换机发送了主机访问策略，交换机会把策略应用到来自交换机端口连接的主机的流量上。如果不应用策略，交换机会应用默认ACL。如果Inspur安全ACS给交换机发送了可下载的ACL，此ACL优先于交换机端口上配置的默认ACL。然而，如果交换机从Inspur安全ACS接收了一个主机访问策略，但没有配置默认ACL，交换机会声明授权失败。

基于 VLAN ID 的 MAC 认证

如果希望基于静态的 VLAN ID 而不是可下载的 VLAN 来认证主机，可以使用基于 VLAN ID 的 MAC 认证特性。在交换机上配置静态 VLAN 策略时，VLAN 信息会和每台请求认证的主机的 MAC 地址一同发给 IAS（Microsoft）RADIUS 服务器。配置在连接端口上的 VLAN ID 会被用来进行 MAC 认证。通过同时使用基于 VLAN ID 的 MAC 认证和 IAS 服务器，网络中可以有固定数量的 VLAN。

此特性也限制了 STP 监控及处理的 VLAN 数量。可以把网络中的 VLAN 当作固定的来管理。

注释： Inspur ACS 服务器不支持此特性（ACS 服务器会忽略发来的新主机的 VLAN ID，并仅基于 MAC 地址进行认证）。

使用访客 VLAN 的 802.1x 认证

可以为交换机上的每个 802.1x 端口配置一个访客 VLAN，给客户端提供有限的服务，比如下载 802.1x 客户端软件。这些客户机可以升级系统以进行 802.1x 认证，而一些主机可能不兼容 IEEE 802.1x，如运行 Windows 98 系统的主机。

在 802.1x 端口上启用了访客 VLAN 时，交换机没有收到发送的 EAP 请求/身份帧的应答，或者客户端没有发送 EAPOL 包时，交换机会把访客 VLAN 分配给客户端。

交换机会维护 EAPOL 包的历史。如果在链路的生存时间内在接口上检测到了 EAPOL 包，交换机会认为连接到该接口的设备是兼容 IEEE 802.1x 的，接口也就不会变为访客 VLAN 的状态。如果接口的链路状态变为 down，EAPOL 历史会被清空。如果未在接口上检测到 EAPOL 包，接口会变为访客 VLAN 状态。

如果交换机尝试授权一台兼容 802.1x 的语音设备，而此时 AAA 服务器不可用，授权尝试会失败，但检测到 EAPOL 包的事件会被保存在 EAPOL 历史中。当 AAA 服务器可用时，交换机会授权该语音设备。然而，交换机不再允许其他设备接入访客 VLAN。为了避免这样的情况发生，可以使用以下命令之一：

- 输入接口配置命令 **authentication event no-response action authorize vlan** *vlan-id*，允许访问访客 VLAN；
- 输入接口配置命令 **shutdown**，接着再输入接口配置命令 **no shutdown** 以重启端口。

如果在链路的生存时间内设备给交换机发送了 EAPOL 包，交换机不再允许认证失败的客户端访问访客 VLAN。

注释： 如果在接口更改为访客 VLAN 之后检测到了 EAPOL 包，接口会返回到未授权状态，而 802.1x 认证会重启。

当交换机端口变为访客 VLAN 后，会允许任意数量的不兼容 802.1x 的客户端进行访问。如果一台兼容 802.1x 的客户端加入了配置了访客 VLAN 的端口，端口会变为未授权状态并被置入用户配置的接入 VLAN，而认证过程会重启。

802.1x 端口的访客 VLAN 在单主机、多主机、多认证以及多域模式中支持。

可以把除了 RSPAN VLAN、私有 VLAN 或语音 VLAN 之外的任意活跃 VLAN 配置为 802.1x 访客 VLAN。访客 VLAN 特性不被内部 VLAN（被路由端口）或中继端口支持，仅被接入端口支持。

交换机支持 MAC 旁路认证。在 802.1x 端口上启用了 *MAC 旁路认证*时，交换机会在 IEEE 802.1x 认证等待 EAPOL 消息交换超时的情况下基于 MAC 地址对客户端进行授权。在 802.1x 端口上检测到客户端时，交换机会等待来自客户端的以太网数据包。交换机会给认证服务器发送一个 RADIUS 访问/请求帧，其中带有基于 MAC 地址生成的用户名和密码。如果授权成

功，交换机会允许客户端访问网络。如果授权失败，交换机会把端口分配到指定的访客 VLAN 中。

使用受限 VLAN 的 802.1x 认证

可以为每个交换机堆栈或者交换机的 IEEE 802.1x 端口配置受限 VLAN（也称为*认证失败* VLAN），给不能访问访客 VLAN 的客户端提供有限的服务。这些客户端兼容 802.1x，但因为认证失败而不能访问其他的 VLAN。受限 VLAN 允许在认证服务器上没有合法凭据的用户（通常是企业的访客）访问有限的服务。管理员可以控制对受限 VLAN 可用的服务。

注释： 如果希望给访客 VLAN 的用户以及受限 VLAN 的用户提供相同的服务，可以配置一个 VLAN 同时作为两种 VLAN 使用。

不使用此特性时，客户端会无限次地尝试认证并失败，而交换机端口会保持在生成树的阻塞状态。使用此特性时，可以让交换机端口在指定次数的认证尝试（默认值是 3 次）之后进入受限 VLAN 中。

认证程序会记录客户端认证失败的次数。当次数超过了配置的最大尝试次数，端口会被移动至受限 VLAN 中。当 RADIUS 服务器回复了 *EAP 失败*包或者不使用 EAP 包的空应答时，失败尝试计数会增加。当端口移动至受限 VLAN 时，失败尝试计数重置。

认证失败的用户会保持在受限 VLAN 中，直到下一次重新尝试认证。受限 VLAN 中的端口会按照配置的间隔（默认为 60 秒）重新尝试认证。如果重新认证失败，端口会保留在受限 VLAN 中。如果重新认证成功，端口会被移动到配置的 VLAN 或者 RADIUS 服务器发来的 VLAN 中。

可以禁用重新认证功能。如果执行了此操作，重启认证过程的唯一方式是在端口上接收到*链路 down* 或 *EAP 登出*事件。建议在客户端可能通过集线器连接的情况下保持重新认证功能启用。因为当客户端断开到集线器的连接时，端口可能无法收到*链路 down* 或 *EAP 登出*事件。

在端口移动到受限 VLAN 之后，交换机会给客户端发送一个假的 EAP 成功消息。此行为会防止客户端无限期地尝试认真就。一些客户端（如运行 Windows XP 的设备）收不到 EAP 成功消息就无法进行 DHCP 的操作。

受限 VLAN 在所有主机模式的 802.1x 端口以及二层端口上支持。

可以把除了 RSPAN VLAN、主私有 VLAN 或语音 VLAN 之外的任意活跃 VLAN 配置为 802.1x 受限 VLAN。受限 VLAN 特性不被内部 VLAN（被路由端口）或中继端口支持，仅被接入端口支持。

其他的端口安全特性，如动态 ARP 监测、DHCP 侦听以及 IP 源防护，可以在受限 VLAN 独立配置。

使用不可访问旁路认证的 802.1x 认证

当交换机无法连通配置的 RADIUS 服务器且新主机无法被认证时，可以使用不可访问旁路认证特性（也称*临界认证*或 *AAA 失败策略*）。可以配置交换机把这些主机连接到临界端口。

当新主机尝试连接到临界端口时，该主机会被移动至用户特定的接入 VLAN，即*临界 VLAN* 中。管理员可以给这些主机授予有限的认证。

当交换机尝试认证连接到临界端口的主机时，交换机会检查配置的 RADIUS 服务器状态。如果服务器可用，交换机就可以认证主机。然而，如果所有的 RADIUS 服务器都不可用，交换机会授予主机网络访问权限，并把交换机端口置入*临界认证*状态中，这是认证状态的一种特

殊情况。

注释： 如果在接口上配置了临界认证，交换机用于临界授权的 VLAN（*临界 VLAN*）应该是活跃的。如果*临界 VLAN* 的状态为不活跃或 down，*临界认证*会话会不断尝试启用不活跃的 VLAN 并一直失败。这可能导致大量的内存占用。

多认证端口对不可访问旁路认证的支持

当端口被配置在任意的主机模式且AAA服务器不可用时，端口会被配置为多主机模式，并被移动至临界VLAN中。要在多认证模式的端口上支持不可访问旁路认证，可以使用**authentication event server dead action reinitialize vlan** *vlan-id*命令。当有新主机尝试连接临界端口时，该端口会被重新初始化，所有连接的主机都会被移动到用户指定的接入VLAN中。此命令在所有主机模式中都支持。

不可访问旁路认证的认证结果

不可访问旁路认证特性的行为取决于端口的授权状态：

- 当连接到临界端口的主机尝试进行认证而所有服务器都不可用时，如果端口是未授权状态，交换机会把端口置为临界认证状态，并放在 RADIUS 配置的或用户指定的接入 VLAN 中；
- 如果端口已经是授权状态且发生了重新认证，交换机会把临界端口置为临界认证状态并放在当前 VLAN 中，此 VLAN 可能是之前由 RADIUS 服务器指定的 VLAN；
- 如果在认证交换期间 RADIUS 服务器变为不可用状态，当前的交换过程会超时，交换机会在进行下一次认证尝试时把临界端口置为临界认证状态。

可以配置临界端口在 RADIUS 服务器重新可用时重新初始化主机，并把它们从临界 VLAN 中移出。配置此操作时，所有在临界认证状态中的临界端口都会自动重新进行认证。

不可访问旁路认证的特性相互影响

不可访问旁路认证会与以下特性相互影响：

- 访客 VLAN——不可访问旁路认证兼容访客 VLAN。在 802.1x 端口上启用访客 VLAN 时，特性间的相互作用如下：
  - 如果至少有一台 RADIUS 服务器可用，在交换机没有收到对其发送的 EAP 请求/身份帧的应答或者客户端没有发送 EAPOL 包时，交换机会把客户端分配到访客 VLAN 中；
  - 如果所有的 RADIUS 服务器都不可用且客户端连接到临界端口，交换机会认证客户端，把临界端口置于临界认证状态，并放在 RADIUS 配置或用户指定的接入 VLAN 中；
  - 如果所有的 RADIUS 服务器都不可用且客户端未连接到临界端口，交换机可能不会把客户端分配到访客 VLAN 中；
  - 如果所有的 RADIUS 服务器都不可用且客户端连接到之前分配到访客 VLAN 的临界端口，交换机会把端口保留在访客 VLAN 中。
- 受限 VLAN——如果 RADIUS 服务器不可用且端口已经被授权在受限 VLAN 中，交换机会

把临界端口置为临界认证状态，并放在受限 VLAN 中；

- 802.1x 审计——如果 RADIUS 服务器不可用，审计不受影响；
- 私有 VLAN——可以在私有 VLAN 主机端口上配置不可访问旁路认证。接入 VLAN 必须是次级私有 VLAN；
- 语音 VLAN——不可访问旁路认证与语音 VLAN 兼容，但是 RADIUS 配置的或用户指定的接入 VLAN 必须与语音 VLAN 不同；
- 远程交换端口分析器（Remote Switched Port Analyzer，RSPAN）——不要把 RSPAN VLAN 配置为 RADIUS 或用户为不可访问旁路认证配置的接入 VLAN。

在交换机堆栈中：

- 堆栈主用设备会通过发送保活包检查 RADIUS 服务器的状态。当 RADIUS 服务器的状态改变时，堆栈主用设备会把此信息发送给堆栈成员。堆栈成员可以在重新认证临界端口的时候检查 RADIUS 服务器的状态；
- 如果选举出了新的堆栈主用设备，交换机堆栈与 RADIUS 服务器之间的链路可能改变，新的堆栈主用设备会立即发送保活包来更新 RADIUS 服务器的状态。如果服务器的状态从 *dead* 变为 *alive*，交换机会重新认证所有临界认证状态的端口。

当成员添加到堆栈时，堆栈主用设备会给成员发送服务器的状态。

注释： 交换机堆栈只在运行 LAN Base 镜像的 Inspur 2960X 交换机上支持。

802.1x 临界语音 VLAN

当一台 IP 电话连接到一个已被访问控制服务器（access control server，ACS）认证的端口上，IP 电话会被放在语音域中。如果 ACS 不可达，交换机不能确定设备是否是语音设备，电话也就不能访问语音网络，进而也不能工作。

对于数据流量，可以配置不可访问旁路认证或临界认证，当服务器不可达的时候允许流量通过本征 VLAN。如果 RADIUS 服务器不可用且启用了不可用旁路认证，交换机会授予客户端访问网络的权利，并把临界认证状态的端口放在 RADIUS 配置或者用户指定的接入 VLAN 中。当交换机不能连通配置的 RADIUS 服务器时，新的主机不能被认证，交换机会把这些主机连接到临界端口上。尝试连接临界端口的新主机会被移动到用户指定的接入 VLAN（临界 VLAN）中，并被授予有限的认证权限。

可以输入接口配置命令 **authentication event server dead action authorize voice** 来配置临界语音 VLAN 特性。当 ACS 不响应时，端口会进入临界认证模式。当来自主机的流量打了语音 VLAN 的标签时，连接的设备（IP 电话）会被置入为端口配置的语音 VLAN 中。IP 电话会通过 CDP（Inspur 设备）、LLDP 或 DHCP 学习语音 VLAN 的标识信息。

可以输入接口配置命令 **switchport voice vlan** *vlan-id* 为端口配置语音 VLAN。

此特性在多域及多认证主机模式上支持。虽然也可以在单主机或多主机模式的交换机上输入此命令，但是除非交换机改为多域或多认证主机模式，否则命令不会生效。

802.1x 用户分配

可以配置 802.1x 用户分配特性，把群组名相同用户在多个不同的 VLAN 上进行负载均分。这些 VLAN 可以由 RADIUS 服务器提供，也可以通过交换机的 CLI 配置在一个 VLAN 群组名下：

- 配置 RADIUS 服务器为用户发送多个 VLAN 名称。多个 VLAN 名称可以作为发往用户的

应答的一部分。802.1x 用户分配特性会追踪特定 VLAN 中的所有用户，并把已授权的用户移动到流量最少的 VLAN 中。

- 配置 RADIUS 服务器为用户发送一个 VLAN 群组名。VLAN 群组名可以作为发往用户的应答的一部分。可以使用交换机的 CLI 查询在配置的 VLAN 群组名中哪个群组名被选用。如果交换机找到了这样的 VLAN 群组名，就会查询这个 VLAN 群组名下的流量最少的 VLAN。负载均分通过把对应的已授权用户移动到这个 VLAN 实现。

注释： RADIUS 服务器发送的 VLAN 信息可以是任意 VLAN ID、VLAN 名称及 VLAN 群组的组合。

802.1x 用户分配配置指南

- 确认至少有一个 VLAN 映射到了 VLAN 群组；
- 可以把多个 VLAN 映射到一个 VLAN 群组中；
- 可以添加或删除 VLAN 群组中的 VLAN；
- 当管理员清除一个 VLAN 群组中已有的 VLAN 时，该 VLAN 中已认证的端口不会被清除，但映射会被从现有的 VLAN 群组中移除；
- 如果清除了 VLAN 群组中的最后一个 VLAN，VLAN 群组也会被清除；
- 即使有活跃的 VLAN 映射到 VLAN 群组，也可以清除该群组。清除群组时，群组内任意 VLAN 中已认证状态的端口或用户都不会被清除，但是 VLAN 到 VLAN 群组的映射会被清除。

语音 VLAN 端口与 IEEE 802.1x 认证

一个语音 VLAN 端口是特殊的接入端口，它关联了两个 VLAN 标识符：

- VVID，承载 IP 电话收发的语音流量。VVID 被用于配置连接到端口的 IP 电话；
- PVID，承载通过 IP 电话连接到交换机的工作站收发的数据流量。PVID 是端口的本征 VLAN。

无论端口授权状态如何，IP 电话都会使用 VVID 传输其语音流量。这使得电话可以独立于 IEEE 802.1x 认证工作。

在单主机模式中，语音 VLAN 上只允许有 IP 电话。在多主机模式中，请求者在 PVID 上认证后其他客户端可以在语音 VLAN 上发送流量。启用多主机模式时，请求者的认证会同时影响 PVID 以及 VVID。

当存在链路，且在来自 IP 电话的第一个 CDP 消息后出现了设备的 MAC 地址时，语音 VLAN 端口变为活跃状态。Inspur IP 电话不会中继来自其他设备的 CDP 包。因此，如果有多台 IP 电话串联在一起，交换机只能识别直接相连的一台。在语音 VLAN 的端口上启用 IEEE 802.1x 认证后，交换机会丢弃来自一跳之外的未识别 IP 电话发来的数据包。

在交换机端口上启用 IEEE 802.1x 认证时，可以把接入端口的 VLAN 同时配置为语音 VLAN。当 IP 电话连接到单主机模式的 802.1x 交换机端口时，交换机无需认证 IP 电话就会授予其网络访问的权利。建议在既认证数据设备也认证语音设备的端口上使用多域认证（MDA）。

注释： 如果在接入端口上启用了 IEEE 802.1x 认证，且该端口已经配置了语音 VLAN 并有 Inspur IP 点相连，Inspur IP 电话会失去与交换机的连通性至多 30 秒。

端口安全与 IEEE 802.1x 认证

通常来说，Inspur 不建议在启用了 IEEE 802.1x 的端口上启用端口安全特性。因为 IEEE 802.1x 强制一个端口只有一个 MAC 地址（为 IP 电话配置 MDA 时，强制一个 VLAN 只有一个 MAC 地址），端口安全特性就冗余了，而且有时还可能干扰 IEEE 802.1x 的操作。

LAN 唤醒与 IEEE 802.1x 认证

使用 IEEE 802.1x 认证以及 LAN 唤醒（wake-on-LAN，WoL）特性，可以在交换机收到特定的以太网帧时启动休眠的主机，这样的数据帧也被称为*魔力包（magic packet）*。可以在管理员需要连接到已经关机的系统时使用此特性。

当一台使用 WoL 的主机连接到 IEEE 802.1x 端口上，且主机已关机，IEEE 802.1x 端口变为未授权状态。这样的端口只能收发 EAPOL 包，所以 WoL 的魔力包就不能到达主机。PC 关机，不被授权，则交换机端口不开放。

当交换机使用 IEEE 802.1x 认证以及 WoL 时，交换机会向未授权端口转发流量，其中就包含魔力包。因为端口仍是未授权状态，交换机会继续阻塞除了 EAPOL 包之外的入向流量。这时的主机可以接收数据包，但不能向网络上的其他设备发送数据包。

注释： 如果在端口上启用了 PortFast，端口强制为双向状态。

使用接口配置命令 **authentication control-direction in** 把端口配置为单向时，端口会变为生成树的转发状态。该端口可以给主机发送数据包，但是不能接收来自主机的数据包。

使用接口配置命令 **authentication control-direction both** 把端口配置为双向时，端口的两个方向都会进行访问控制。该端口不会向主机收发数据包。

MAC 旁路认证与 IEEE 802.1x 认证

可以配置交换机使用 MAC 旁路认证特性，让交换机基于客户端的 MAC 地址进行授权。例如，可以在连接了打印机的 IEEE 802.1x 端口上启用此特性。

如果 IEEE 802.1x 认证等待客户端 EAPOL 应答超时，交换机会尝试使用 MAC 旁路认证特性授权客户端。

在启用了 IEEE 802.1x 的端口上启用 MAC 旁路认证特性时，交换机会把 MAC 地址作为客户端的身份。认证服务器上的数据库中有允许访问网络的客户端的 MAC 地址。在 IEEE 802.1x 端口上检测到客户端之后，交换机会等待来自客户端的以太网数据包。交换机会给认证服务器发送一个 RADIUS 访问/请求帧，带有基于 MAC 地址生成的用户名和密码。如果授权成功，交换机会授予客户端访问网络的权限。如果授权失败，交换机会把端口分配给配置的访客 VLAN。此过程适用于多数客户端设备，但不适用于其他 MAC 地址格式的客户端。当客户端 MAC 地址与标准格式不同，或者 RADIUS 配置要求用户名与密码不同时，可以配置使用 MAB 认证。

如果在链路的生存时间内在接口上检测到了 EAPOL 包，交换机可以确定连接到接口上的设备兼容 802.1x，就会使用 802.1x 认证方式（而不是 MAC 旁路认证）来授权接口。如果接口的链路状态变为 down，EAPOL 历史会被清除。

交换机已经使用 MAC 旁路认证授权了一个端口，如果此时检测到了 IEEE 802.1x 认证请求者，交换机不会授权连接到端口的客户端。重新认证发生时，如果之前的会话因为终止操作

RADIUS 属性的值是 DEFAULT 而结束，交换机会使用端口配置的认证或重新认证方式执行操作。

使用MAC旁路认证方式授权的客户端可以被重新认证。该客户端的重新认证过程与使用IEEE 802.1x 认证的客户端相同。在重新认证期间，端口会保持在之前分配的 VLAN 中。如果重新认证成功，交换机会把端口保留在相同 VLAN 中。如果重新认证失败，交换机会把端口分配给配置的访客 VLAN。

如果重新认证基于会话超时 RADIUS 属性（属性[27]）以及终止操作 RADIUS 属性（属性[29]），而且终止操作 RADIUS 属性的操作是*初始化（Initialize）*（属性默认值为 *DEFAULT*），MAC 旁路认证的会话将结束，且重新认证期间的连通性会丢失。如果启用了 MAC 旁路认证且 IEEE 802.1x 认证超时，交换机会使用 MAC 旁路认证特性来初始化重新认证过程。更多有关 AV 对的信息，参见 RFC 3580“IEEE 802.1X 远程验证拨入用户服务（RemoteAuthentication Dial In User Service，RADIUS）使用指南”。

MAC 旁路认证会与以下特性相互影响：

- IEEE 802.1x 认证——只有在端口上启用 802.1x 认证时才可以启用 MAC 旁路认证
- 访客 VLAN——如果客户端 MAC 地址身份非法，交换机会把客户端分配到配置的访客 VLAN 中
- 受限 VLAN——当连接到 IEEE 802.1x 端口的客户端使用 MAC 旁路认证时，此特性不被支持
- 端口安全
- 语音 VLAN
- 私有 VLAN——可以将客户端分配给私有 VLAN。
- 网络边缘接入拓扑（Network Edge Access Topology，NEAT）——MAB 和 NEAT 特性是互斥的。在接口上启用 NEAT 时不能启用 MAB，反之亦然

Inspur INOS 12.2（55）SE 以及之后版本支持过滤详细的 MAB 系统消息。

网络接入控制二层 IEEE 802.1x 验证

交换机支持网络接入控制（Network Admission Control，NAC）二层 IEEE 802.1x 验证特性，会在授予设备网络访问权限之前检查终端系统或客户端的防病毒状态或*态势（posture）*。使用 NAC 二层 IEEE 802.1x 验证时，可以执行以下操作：

- 从认证服务器上下载会话超时 RADIUS 属性（属性[27]）以及终止操作 RADIUS 属性（属性[29]）；
- 把进行重新认证尝试之间的秒数设置为会话超时 RADIUS 属性（属性[27]）值，并通过 RADIUS 服务器获取客户端的访问策略；
- 使用终止操作 RADIUS 属性（属性[29]）设置交换机尝试重新认证客户端时采取的操作。如果此值为 *DEFAULT* 或未设置，重新认证时会话结束。如果值是 RADIUS 请求，重新认证过程开始；
- 把 VLAN 编号或名称的列表、VLAN 群组名称设置为隧道组私有 ID（属性[81]）的值，并让隧道偏好（属性[83]）值使用这些 VLAN。如果不配置隧道偏好，首个隧道组私有 ID（属性[81]）会从列表中选取；
- 使用特权 EXEC 命令 **show authentication** 查看客户端的 NAC 态势令牌，获知客户端的态势；
- 把次级私有 VLAN 配置为访客 VLAN。

配置 NAC 二层 IEEE 802.1x 验证的过程与配置 IEEE 802.1x 基于端口认证的过程相似，除了必

须要在 RADIUS 服务器上配置态势令牌。

灵活的认证顺序

可以使用灵活认证顺序功能配置端口认证新主机时采用的方法的顺序。IEEE 802.1x 灵活认证（Flexible Authentication）特性支持三种认证方式：

- dot1X——IEEE 802.1x 认证时二层认证方式
- mab——MAC 旁路认证时二层认证方式
- webauth——网页认证是三层认证方式

使用此特性时，可以控制哪些端口使用哪些认证方式，而且可以控制这些端口上认证方式故障转移的顺序。例如，MAC 旁路认证以及 802.1x 可以是认证的主要方式或次要方式，如果尝试这些认证方式失败，网页认证可以作为备用方式使用。

IEEE 802.1x 灵活认证特性支持以下主机模式：

- 多认证——多认证允许在一个语音 VLAN 上进行一次认证，在数据 VLAN 上进行多次认证
- 多域认证——多域认证允许进行两次认证：一次在语音 VLAN 上，一次在数据 VLAN 上

Open1x 认证

Open1x 认证允许设备在被认证之前访问端口。配置开放认证时，新主机可以根据端口上定义的访问控制列表（ACL）传输流量。主机被认证之后，在 RADIUS 服务器上配置的策略会应用给主机。

可以在以下场景中配置开放认证：

- 单主机模式——认证前后只允许一个用户访问网络
- MDA 模式——只允许语音域中有一个用户，数据域中有一个用户
- 多认证模式——与 MDA 相似，但可以认证多台主机

注释： 如果配置了开放认证，该方式优先于其他的认证控制特性。这意味着如果使用了接口配置命令**authentication open**，无论接口配置命令**authentication port-control**配置如何，端口都会允许主机访问。

多域认证

交换机支持进行多域认证（multidomain authentication，MDA），允许在一个交换机端口上同时认证一台数据设备和一台语音设备（如 Inspur 或非 Inspur 的 IP 电话）。端口被分为一个数据域和一个语音域。

注释： 对于所有的主机模式，配置基于端口的认证时，认证之前线路协议保持 up 状态。MDA 不强制设备认证的顺序。然而在启用 MDA 的端口上，建议在认证数据设备之前认证语音设备。

按照以下指南配置 MDA：

- 必须把交换机端口配置为 MDA；
- 当主机模式设置为多域时，必须配置 IP 电话使用的语音 VLAN；

- 启用 MDA 端口上的语音 VLAN 分配功能在 Inspur INOS 12.2（40）SE 及之后版本上支持；
- 如需认证语音设备，必须配置 AAA 服务器发送属性值为 *device-traffic-class=voice* 的 Inspur 属性值（AV）对。否则，交换机会把语音设备当作数据设备；
- 访客 VLAN 和受限 VLAN 特性只适用于 MDA 端口上的数据设备。交换机会把授权失败的语音设备当作数据设备；
- 如果有多台设备尝试在端口的语音域或数据域上进行授权，端口会被错误禁用；
- 在设备被授权之前，端口会丢弃其流量。允许在语音以及数据 VLAN 中使用非 Inspur 的 IP 电话或语音设备。数据 VLAN 允许该设备联系 DHCP 服务器，获取 IP 地址以及语音 VLAN 的信息。当语音设备开始在语音 VLAN 上发送数据后，其对数据 VLAN 的访问将被阻止；
- 对于端口安全特性的 MAC 地址数量限制，绑定到语音 VLAN 上的语音设备的 MAC 地址不会被计数；
- MDA 可以使用 MAC 旁路认证作为备用的认证机制，使不支持 IEEE 802.1x 认证的设备可以连通交换机端口；
- 在端口上检测到一个*数据*设备或*语音*设备时，在认证成功前其 MAC 地址会被阻塞。如果认证失败，该 MAC 地址会保持阻塞 5 分钟；
- 未授权的端口上，如果在*数据* VLAN 中检测到了超过五台设备，或者在*语音* VLAN 上检测到超过一台设备，该端口会错误禁用；
- 当端口主机模式从单主机或多主机变为多域模式时，端口上已授权的数据设备仍保持授权状态。然而，端口语音 VLAN 已经允许的 Inspur IP 电话会被自动移除，且必须在端口上进行重新认证；
- 当端口主机模式从单主机或多主机变为多域模式时，如访客和受限 VLAN 这样的主动回退机制配置保持不变；
- 把端口主机模式从多域模式改为单主机或多主机模式会移除端口上所有已授权的设备；
- 如果数据域先被认证，且被置入访客 VLAN 中，不兼容 IEEE 802.1x 的语音设备需要把自己的数据包标记为语音 VLAN 数据包才能触发认证过程；
- 不建议在启用 MDA 的端口上使用基于用户的 ACL。使用基于用户 ACL 策略的已授权设备可能会同时影响端口上的语音 VLAN 和数据 VLAN。若使用这样的配置，只应给端口上的一台设备执行基于用户的 ACL。

802.1x 请求交换机、认证交换机以及网络边缘接入拓扑（NEAT）

网络边缘接入拓扑（Network Edge Access Topology，NEAT）特性把身份认证扩展到配线间之外的区域（比如会议室）。

- 802.1x 请求交换机：可以使用 802.1x 请求者特性，配置一台交换机作为另一台交换机的请求者。在配线间外的交换机通过中继端口连接到上行交换机这类场景中，此配置很有用。配置了 802.1x 请求者特性的交换机会与上行交换机认证以进行安全连接。当请求交换机认证成功时，认证交换机上的端口模式会从接入变为中继。启用 CISP 时，必须在请求交换机上手动配置中继；
- 如果在认证交换机上配置了接入 VLAN，该 VLAN 在成功认证之后会成为中继端口的本征 VLAN。

默认状态下，如果请求交换机连接到了一台启用了BPDU防护的认证交换机，且认证交换机的端口在请求交换机被认证之前收到了生成树协议（Spanning Tree Protocol，STP）的网桥协议数据单元（BridgeProtocol Data Unit，BPDU）数据包，该端口可能被错误禁用。从

Inspur INOS 15.0(1) SE版开始，可以控制认证期间从请求端口上发出的流量。输入全局配置命令**dot1x supplicant controlled transient**会在认证期间临时阻塞请求交换机的端口，以保证认证交换机端口不会在认证完成之前被关闭。如果认证失败，请求交换机端口会开放。输入全局配置命令**no dot1x supplicant controlledtransient**可以在认证期间打开请求交换机端口。这是交换机的默认行为。

当认证交换机上通过接口配置命令**spanning-tree bpduguard enable**启用了BPDU防护时，强烈建议在请求交换机上使用**dot1x supplicant controlled transient**命令。

注释： 如果在认证交换机上使用全局配置命令**spanning-tree portfastbpduguard default**启用了BPDU防护，输入**dot1x supplicant controlled transient**命令不能防止BPDU违规发生。

可以在认证交换机连接到多台请求交换机的接口上启用 MDA 或多认证模式。认证交换机接口不支持多主机模式。

当重启接口上使用单主机模式的认证交换机时，该接口可能在认证前变为错误禁用状态。要从错误禁用状态恢复，请重启认证交换机的接口以激活接口并发起认证过程。

要让网络边缘接入拓扑（NEAT）特性在所有主机模式中都能工作，请在请求交换机上配置全局配置命令 **dot1x supplicant force-multicast**。

- 主机授权：确保网络上只允许通过被授权主机（连接到请求交换机）的流量。交换机使用客户端信息信令协议（Client Information Signalling Protocol，CISP）把连接到请求交换机的 MAC 地址发送给认证交换机；
- 自动启用：在认证交换机上自动启用中继配置，允许传输来自请求交换机上多个 VLAN 的用户流量。请在 ACS 上把 inspur-av-pair 配置为 *device-traffic-class=switch*（可以在*组*或*用户*设置下配置此项）

图 124：使用 CISP 的认证交换机及请求交换机

Workstations (clients)	工作站（客户端）
Supplicant switch (outside wiring closet)	请求交换机（在配线间外）
Authenticator switch	认证交换机
Access control server (ACS)	访问控制服务器（ACS）
Trunk port	中继端口

注释： 使用 NEAT 的请求交换机和认证交换机上不支持 **switchport nonegotiate** 命令。此命令不应配置在拓扑的请求端。如果配置在认证端，交换机的内部宏会自动把此命令从端口移除。

语音感知的 802.1x 安全特性

注释： 要使用语音感知 IEEE 802.1x 认证功能，交换机必须运行 LAN Base 的镜像。

可以使用语音感知的 802.1x 安全特性，配置交换机在数据 VLAN 或语音 VLAN 安全违规事件发生时只禁用相关 VLAN。之前，当尝试认证的数据客户端造成了安全违规事件时，整个端口会被关闭，到了连通性完全丢失。

当 PC 连接到 IP 电话时，可以使用此特性。数据 VLAN 上的安全违规事件只会导致数据 VLAN 被关闭。经过交换机传输的语音 VLAN 流量不会被干扰。

通用会话 ID

无论采取何种认证方式，认证管理器都会对客户端使用一个会话 ID（称为通用会话 ID）。此 ID 被用于所有的报告功能，如 show 命令以及 MIB。此会话 ID 出现在会话前的 syslog 消息中。

会话 ID 包含：

- 网络接入设备（Network Access Device，NAD）的 IP 地址
- 一个单调递增的唯一的 32 位整数
- 会话开始时间戳（一个 32 位整数）

以下示例显示了命令 show authentication 输出中会话 ID 的显示方式。此例中的会话 ID 是 160000050000000B288508E5：

```
Device# show authentication sessions
Interface MAC Address Method Domain Status Session ID
Fa4/0/4 0000.0000.0203 mab DATA Authz Success 160000050000000B288508E5
```

以下是 syslog 输出中会话 ID 的显示方式。此例中的会话 ID 是 160000050000000B288508E5：

```
1w0d: %AUTHMGR-5-START: Starting 'mab' for client (0000.0000.0203) on Interface
Fa4/0/4
AuditSessionID 160000050000000B288508E5
1w0d: %MAB-5-SUCCESS: Authentication successful for client (0000.0000.0203) on
Interface
Fa4/0/4 AuditSessionID 160000050000000B288508E5
1w0d: %AUTHMGR-7-RESULT: Authentication result 'success' from 'mab' for client
(0000.0000.0203) on Interface Fa4/0/4 AuditSessionID 160000050000000B288508E5
```

此会话 ID 被 NAD、AAA 服务器以及其他汇报分析程序用来标识客户端。此 ID 自动生成，无需配置。

如何配置 802.1x 基于端口的认证

默认的 802.1x 认证配置

表 148：默认的 802.1x 认证配置

特性	默认设置
交换机上 802.1x 的启用状态	禁用
每个端口的 802.1x 启用状态	禁用（强制授权）。 端口会收发正常流量，无需对客户端进行基于 802.1x 的认证
AAA	禁用
RADIUS 服务器 • IP 地址 • UDP 认证端口 • 默认审计端口	• 未指定 • 1645 • 1646 • 未指定

• 秘钥	
主机模式	单主机模式
控制方向	双向控制
周期性重新认证	禁用
尝试重新认证的时间间隔	3600 秒
重新认证次数	2 次（在端口变为未授权状态之前交换机重启认证过程的次数）
静默时长	60 秒(在与客户端认证交换失败之后交换机保持静默状态的秒数）
重传时间	30 秒（交换机在重新发送 EAP 请求/身份帧之前等待客户端应答的秒数）
最大重传次数	2 次（交换机在重启认证过程之前发送 EAP 请求/身份帧的次数）
客户端超时时间	30 秒（在中继认证服务器到客户端的请求时，交换机重发请求给客户端之前等待的时间）
认证服务器超时时间	30 秒（在中继客户端给认证服务器的应答时，交换机重发应答给服务器之前等待的时间）
不活跃超时	禁用
访客 VLAN	未指定
不可访问旁路认证	禁用
受限 VLAN	未指定
认证交换机模式	未指定
MAC 旁路认证	禁用
语音感知安全	禁用

802.1x 认证配置指南

802.1x 认证

以下是 802.1x 配置指南：

- 启用 802.1x 认证时，端口在任何其他二层或三层特性启用之前被认证；
- 如果启用了 802.1x 的端口分配的 VLAN 变化，变化对交换机透明且不会影响交换机配置。比如，端口可能分配给了 RADIUS 服务器指定的 VLAN，而重新认证之后又分配给了不同的 VLAN，此变化对交换机透明；
- 如果 802.1x 端口分配的 VLAN 关闭、禁用或被移除，端口会变为未授权状态。比如，端口分配的接入 VLAN 被关闭或移除之后，端口变为未授权；
- 802.1x 协议支持二层静态接入端口、语音 VLAN 端口以及三层被路由端口，但不支持以下端口类型：
  - 动态端口——动态模式中的端口可以与邻居协商并成为中继端口。如果尝试在动态端口上启用 802.1x 认证，会出现错误消息，且 802.1x 认证不会被启用。如果尝试

把启用了 802.1x 的端口更改为动态模式，会出现错误消息，且端口模式不会变化；
  - EtherChannel 端口——不要把 EtherChannel 中的活跃成员或尚未活跃的成员配置为 802.1x 端口。如果尝试在一个 EtherChannel 端口上启用 802.1x 认证，会出现错误消息，且 802.1x 认证不会被启用；
  - 交换端口分析器（SPAN）和远程 SPAN（RSPAN）目的端口——可以在 SPAN 或 RSPAN 目的端口上启用 802.1x 认证。然而，直到端口从 SPAN 或 RSPAN 目的端口中移除，802.1x 认证才会启用。可以在 SPAN 或 RSPAN 源端口上启用 802.1x 认证。
- 在交换机上使用全局配置命令全局启用 802.1x 认证之前，应移除同时配置了 802.1x 认证和 EtherChannel 的端口上的 EtherChannel 配置；
- Inspur INOS 12.2(55)SE 以及之后版本支持过滤与 802.1x 认证相关的系统消息。

VLAN 分配、访客 VLAN、受限 VLAN 以及不可访问旁路认证

以下是 VLAN 分配、访客 VLAN、受限 VLAN 以及不可访问旁路认证的配置指南：
- 在端口上启用 802.1x 认证时，不能把端口 VLAN 配置为语音 VLAN；
- 使用 VLAN 分配的 802.1x 认证特性不支持中继端口、动态端口或是使用 VMPS 分配的动态端口；
- 可以把除了 RSPAN VLAN 或语音 VLAN 之外的任意 VLAN 配置为 802.1x 访客 VLAN。访客 VLAN 特性不支持内部 VLAN（被路由端口）或中继端口，只支持接入端口；
- 为连接了 DHCP 客户端的 802.1x 端口配置访客 VLAN 时，客户端可能需要通过 DHCP 服务器获取主机 IP 地址。可以更改设置，在客户端上的 DHCP 进程超时并尝试从 DHCP 服务器获取 IP 地址之前，让交换机重启 802.1x 认证过程。可以减少对 802.1x 认证过程的设置（接口配置命令 authentication timer inactivity 和 authentication timer reauthentication）。需减少的设置数量取决于连接的 802.1x 设备类型；
- 按照以下指南配置不可访问旁路认证特性：
  - 此特性支持单主机模式和多主机模式的 802.1x 端口；
  - 如果客户端运行 Windows XP，且客户端连接到的端口在临界认证状态中，Windows XP 可能报告接口未被认证；
  - 如果 Windows XP 客户端配置进行 DHCP 且拥有来自 DHCP 服务器的 IP 地址，在临界端口上接收 EAP 成功消息可能不会重启 DHCP 配置过程；
  - 可以在 802.1x 端口上配置不可访问旁路特性以及受限 VLAN。如果交换机尝试重新认证受限 VLAN 中的临界端口，且所有 RADIUS 服务器都不可用，交换机会把端口状态改为临界认证状态，并保持在受限 VLAN 中；
  - 如果 CTS 链路在临界认证模式中且主用设备重启，SGT 在设备上配置的策略在新主用设备上不可用。这是因为 3750-X 交换机堆栈中的内部绑定不会被同步到备用交换机上。
- 可以把除了 RSPAN VLAN 或语音 VLAN 之外的任意 VLAN 配置为 802.1x 受限 VLAN。受限 VLAN 特性不支持内部 VLAN（被路由端口）或中继端口，只支持接入端口。

MAC 旁路认证

802.1x 端口上允许的最大设备数量如下：
- 在单主机模式中，接入 VLAN 上只允许有一台设备。如果端口也配置了语音 VLAN，无

限数量的 Inspur IP 电话可以通过语音 VLAN 收发流量；

- 在多域认证（MDA）模式中，接入 VLAN 允许有一台设备，语音 VLAN 允许有一台设备；
- 在多主机模式中，端口上只允许有一个 802.1x 请求者，但是接入 VLAN 中允许有无限数量的非 802.1x 主机。语音 VLAN 允许有无限数量的设备。

配置 802.1x 就绪状态检查

802.1x 就绪状态检查特性会监控交换机所有端口上的 802.1x 活动，并显示端口连接的支持 802.1x 的设备信息。可以使用此特性确定连接到交换机端口的设备是否兼容 802.1x。

802.1x 就绪状态检查允许在配置 802.1x 的所有端口上使用。就绪状态检查在配置为 **dot1x force-unauthorized** 的端口上不可用。

按照以下步骤在交换机上启用 802.1x 就绪状态检查：

在开始前

以下是启用就绪状态检查的指南：

就绪状态检查通常在 802.1x 启用之前使用。

如果使用特权 EXEC 命令 **dot1x test eapol-capable** 且没有指定接口，交换机堆栈的所有端口都会被测试。

如果在启用 802.1x 的端口上配置了 the **dot1x test eapol-capable** 命令，链路启用时，端口会查询连接客户端的 802.1x 兼容性。客户端使用通知包应答，则其兼容 802.1x。如果客户端在超时时间内进行响应，交换机会产生 syslog 消息。如果客户端没有响应查询消息，则其不兼容 802.1x，此时不会产生 syslog 消息。

可以在处理多台主机的端口上（比如 PC 通过 IP 电话连接端口）发送就绪状态检查消息。对于每一个在超时时间内响应的客户端，交换机都会产生一条 syslog 消息。

总步骤

1. enable

2. configure terminal

3. dot1x test eapol-capable [**interface** *interface-id*]

4. dot1x test timeout *timeout*

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 3	**dot1x test eapol-capable** [**interface** *interface-id*] **示例:** `Device#` **dot1x test eapol-capable interface**	在交换机上启用就绪状态检查， （可选）使用 *interface-id* 指定检查哪个端口的 IEEE 802.1x 就绪状态。 **注释：**如果省略可选的 **interface** 关键

	gigabitethernet1/0/13 DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on gigabitethernet1/0/13 is EAPOL capable	字，交换机上的所有端口都会被测试
步骤 4	**dot1x test timeout** *timeout*	（可选）配置等待 EAPOL 应答的超时时间。范围从 1 到 65535 秒，默认值是 10 秒
步骤 5	**end** **示例:** Device(config)# **end**	返回特权 EXEC 模式
步骤 6	**show running-config** **示例：** Device# **show running-config**	验证配置的条目
步骤 7	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把配置保存在配置文件中

配置语音感知的 802.1x 安全特性

注释： 要使用语音感知 IEEE 802.1x 认证功能，交换机必须运行 LAN Base 的镜像。

可以使用语音感知的 802.1x 安全特性，配置交换机在数据 VLAN 或语音 VLAN 安全违规事件发生时只禁用相关 VLAN。当 PC 连接到 IP 电话时，可以使用此特性。数据 VLAN 上的安全违规事件只会导致数据 VLAN 被关闭。经过交换机传输的语音 VLAN 流量不会被干扰。

按照以下指南在交换机上配置语音感知 802.1x 安全特性：

- 输入全局配置命令**errdisable detect cause security-violationshutdown vlan**启用语音感知802.1x安全特性。输入命令的**no**形式禁用语音感知802.1x安全。此命令会应用到交换机上所有配置了802.1x的端口。

  注释: 如果不包括**shutdown vlan**关键字，进入错误禁用状态时整个端口都会被关闭。
- 如果使用全局配置命令**errdisable recovery cause security-violation**配置错误禁用恢复，端口会被自动重启。如果没有为端口配置错误禁用恢复，可以使用**shutdown**和**no shutdown**接口配置命令重启端口。
- 可以使用特权EXEC命令**clear errdisable interface** *interface-id* **vlan** [*vlan-list*]重启单个VLAN。如果不指定范围，端口上的所有VLAN都会被启用。

在特权 EXEC 模式中按照以下步骤启用语音感知 802.1x 安全特性。

总步骤

1. **configure terminal**
2. **errdisable detect cause security-violation shutdown vlan**
3. **errdisable recovery cause security-violation**
4. **clear errdisable interface***interface-id* **vlan** *[vlan-list]*
5. 输入以下命令：

• shutdown

• no shutdown

6. end

7. show errdisable detect

具体步骤

	命令或操作	目的
步骤 1	**configure terminal**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**errdisable detect cause security-violation shutdown vlan**	关闭发生了安全违规错误的 VLAN。 **注释**：如果未包含 **shutdown vlan** 关键字，整个端口都会进入错误禁用状态并被关闭
步骤 3	**errdisable recovery cause security-violation**	配置错误禁用恢复。
步骤 4	**clear errdisable interface***interface-id* **vlan** *[vlan-list]*	（可选）重新启用单个错误禁用的 VLAN。 • 使用 interface-id 指定要重新启用 VLAN 的端口。 • （可选）使用 vlan-list 指定要重新启用的 VLAN 列表。如果 vlan-list 不指定，所有 VLAN 都被重启
步骤 5	输入以下命令： **• shutdown** **• no shutdown**	（可选）重新启用错误禁用的 VLAN，清除错误禁用标志
步骤 6	**end**	返回特权 EXEC 模式
步骤 7	**show errdisable detect**	验证配置的条目

以下示例展示了如何配置交换机，关闭发生了安全违规错误的 VLAN：

```
Switch(config)# errdisable detect cause security-violation shutdown vlan
```

以下示例展示了如何在 Gigabit Ethernet 40/2 端口上启用所有错误禁用的 VLAN。

```
Switch# clear errdisable interface gigabitethernet4/0/2vlan
```

可以输入特权 EXEC 命令 **show errdisable detect** 验证设置。

配置 802.1x 违规模式

可以配置 802.1x 端口，使其在在以情况发生时关闭端口、生成 syslog 消息或者丢弃来自新设备的包：

- 设备连接到启用 802.1x 的端口
- 端口被认证的设备到达最大数量

在特权 EXEC 模式中按照以下步骤在交换机上配置安全违规操作。

总步骤

1. configure terminal

2. aaa new-model

3. aaa authentication dot1x {**default**} *method1*

4. interface *interface-id*

5. switchport mode access

6. authentication violation {**shutdown** | **restrict** | **protect** | **replace**}

7. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device#` **configure terminal**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**aaa new-model** **示例:** `Device(config)#` **aaa new-model**	启用 AAA
步骤 3	**aaa authentication dot1x** {**default**} *method1* **示例:** `Device(config)#` **aaa authentication dot1x default group radius**	创建 802.1x 认证方式列表。 当*没有*在 **authentication** 命令中指定命名的列表时，可以在方法之后加上 **default** 关键字来创建默认列表，以在默认情况下使用。默认方式列表会自动应用到所有端口上。 对于 *method1* 字段，输入关键字，使用所有 RADIUS 服务器列表进行认证
步骤 4	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/4**	指定连接到客户端的将要启用 IEEE 802.1x 认证的端口，并进入接口配置模式
步骤 5	**switchport mode access** **示例:** `Device(config-if)#` **switchport mode access**	设置端口为接入模式
步骤 6	**authentication violation** {**shutdown**	**restrict** | **protect**	**replace**} **示例:** `Device(config-if)#` **authentication violation restrict**	配置违规模式。关键字含义如下： • **shutdown**——错误禁用端口 • **restrict**——生成 syslog 错误 • **protect**——丢弃任何新设备发给端口的流量 • **replace**——移除当前会话，对新主机进行认证
步骤 7	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式

配置 802.1x 认证

为使用基于用户的 ACL 或 VLAN 分配，必须配置交换机为所有网络相关的服务请求启用 AAA

授权。

以下是 802.1x AAA 过程。

在开始前

要配置 802.1x 基于端口的认证，必须启用认证、授权以及审计（AAA）并指定认证方式列表。方式列表描述了向认证服务器查询的认证方式顺序。

总步骤

1. **用户连接到交换机端口。**
2. **进行认证。**
3. **基于 RADIUS 服务器配置进行 VLAN 分配。**
4. **交换机给审计服务器发送开始消息。**
5. **按需执行重新认证。**
6. **基于重新认证结果，交换机给审计服务器发送中间审计更新消息。**
7. **用户断开端口。**
8. **交换机给审计服务器发送停止消息。**

具体步骤

	命令或操作	目的

配置 802.1x 基于端口的认证

在特权 EXEC 模式中按照以下步骤配置 802.1x 基于端口的认证。

总步骤

1. **configure terminal**
2. **aaa new-model**
3. **aaa authentication dot1x** {**default**} *method1*
4. **dot1x system-auth-control**
5. **aaa authorization network** {**default**} **group radius**
6. **radius server** *server name*
7. **key** *string*
8. **interface** *interface-id*
9. **switchport mode access**
10. **authentication port-control auto**
11. **dot1x pae authenticator**
12. **end**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device# configure terminal`	进入特权 EXEC 模式
步骤 2	**aaa new-model** **示例:** `Device(config)# aaa new-model`	启用 AAA
步骤 3	**aaa authentication dot1x** {**default**} *method1*	创建 802.1x 认证方式列表。 当*没有*在 **authentication** 命令中指定命

	示例: `Device(config)#` **aaa authentication dot1x default group radius**	名的列表时，可以在方法之后加上 **default** 关键字来创建默认列表，以在默认情况下使用。默认方式列表会自动应用到所有端口上。 对于 *method1* 字段，输入关键字，使用所有 RADIUS 服务器列表进行认证。 **注释：** 虽然命令行帮助字符串中还有其他可见的关键字，但只支持 **group radius** 关键字
步骤 4	**dot1x system-auth-control** **示例:** `Device(config)#` **dot1x system-auth-control**	在交换机上全局启用 802.1x 认证
步骤 5	**aaa authorization network {default} group radius** **示例:** `Device(config)#` **aaa authorization network default group radius**	（可选）配置交换机使用用户 RADIUS 授权所有网络相关的服务请求，如基于用户的 ACL 或 VLAN 分配
步骤 6	**radius server** *server name* **示例:** `Device(config)#` **radius server rsim address ipv4 124.2.2.12**	（可选）指定 RADIUS 服务器的 IP 地址
步骤 7	**key** *string* **示例:** `Device(config-radius-server)#` **key rad123**	（可选）指定交换机和 RADIUS 服务器上 RADIUS 后台进程之间使用的认证及加密密钥
步骤 8	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/2**	指定连接到客户端的将要启用 IEEE 802.1x 认证的端口，并进入接口配置模式
步骤 9	**switchport mode access** **示例:** `Device(config-if)#` **switchport mode access**	（可选）如果在步骤 6、7 中配置了 RADIUS 服务器，设置端口为接入模式
步 骤 10	**authentication port-control auto** **示例:** `Device(config-if)#` **authentication port-control auto**	在端口上启用 802.1x 认证
步 骤 11	**dot1x pae authenticator** **示例:** `Device(config-if)#` **dot1x pae authenticator**	设备端口接入实体（Port Access Entity）只作为认证者，忽略所有发往请求者的消息
步 骤 12	**end** **示例:**	返回特权 EXEC 模式

	Device(config-if)# **end**	

配置交换机到 RADIUS 服务器的通信

管理员也需要在 RADIUS 服务器上进行一些设置，包括交换机的 IP 地址，以及服务器和交换机共享的密钥穿。更多信息参见 RADIUS 服务器文档。

按照以下步骤在交换机上配置 RADIUS 服务器参数。此步骤是必须的。

在开始前

必须启用认证、授权以及审计（AAA）并指定认证方式列表。方式列表描述了向认证服务器查询认证方式的顺序。

总步骤

1. enable

2. configure terminal

3. radius server *server name*

4. address {**ipv4 | ipv6**} *ip address* **auth-port** *port number* **acct-port** *port number*

5. key *string*

6. end

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** Device>**enable**	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**radius server** *server name* **示例:** Device(config)# **radius server rsim**	指定 RADIUS 服务器名称，并进入 RADIUS 服务器配置模式
步骤 4	**address** {**ipv4	ipv6**} *ip address* **auth-port** *port number* **acct-port** *port number* **示例:** Device(config-radius-server)# **address ipv4 124.2.2.12**	指定 RADIUS 服务器的 IP 地址。 使用 **auth-port** *port-number*,指定认证请求的 UDP 目的端口。默认值是 1645，范围从 0 到 65536。 使用 **acct-port** *port-number* 指定认证请求的 UDP 的目的端口。默认值是 1646
步骤 5	**key** *string* **示例:** Device(config-radius-server)# **key rad123**	指定设备以及 RADIUS 服务器运行的 RADIUS 后台程序之间使用的认证以及加密密钥。 **注释：**此密钥是明文密钥，且必须与 RADIUS 服务器使用的加密密钥相同。把密钥配置在 **radius server** 命令的最后一项。密钥前的空格会被忽略，但是密钥中间以及之后的空格会被

		使用。如果密钥使用空格，不要把密钥放在引号之间，除非引号是密钥的一部分
步骤 6	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式

配置主机模式

对于接口配置命令 **authentication port-control** 设置为 **auto** 的 IEEE 802.1x 授权端口，可以在特权 EXEC 模式中按照以下步骤配置，以允许有多台主机。配置关键字 **multi-domain** 可以启用多域认证（MDA），允许在一个交换机端口上同时有主机和语音设备（如 Inspur 或非 Inspur 的 IP 电话）。此步骤是可选的。

总步骤

1. configure terminal

2. interface *interface-id*

3. authentication host-mode [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]

4. end

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/1**	指定多台主机直连的端口，并进入接口配置模式
步骤 3	**authentication host-mode** [**multi-auth**	**multi-domain**	**multi-host**	**single-host**] **示例:** `Device(config-if)#` **authentication host-mode multi-host**	允许 802.1x 授权的端口上有多台主机。关键字含义如下： • **multi-auth**——允许语音 VLAN 有一台客户端，数据 VLAN 有多台客户端被认证。 **注释：** **multi-auth** 关键字只在 **authentication host-mode** 命令中可用。 • **multi-host**——在一台主机被认证之后，允许 802.1x 授权端口上存在多台主机。 • **multi-domain**——允许 IEEE 802.1x 授权端口上有一台主机以及一台语音设备。 **注释：**当主机模式设置为时必须为 IP 电话配置语音 VLAN。

		确保特定接口的接口配置命令 **authentication port-control** 设置为 **auto**
步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式

配置周期性重新认证

可以启用周期性的 802.1x 客户端重新认证功能并指定执行频率。如果不指定时间周期，尝试重新认证的周期是 3600 秒。

在特权 EXEC 模式中，按照以下步骤启用客户端的周期性重新认证，并指定尝试重新认证的周期。此过程是可选的。

总步骤

1. configure terminal

2. interface *interface-id*

3. authentication periodic

4. authentication timer {{[**inactivity** | **reauthenticate** | **restart**]} {*value*}}

5. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/1**	指定要配置的端口，并进入接口配置模式
步骤 3	**authentication periodic** **示例:** `Device(config-if)#` **authentication periodic**	启用客户端的重新认证，该设置默认被禁用。 **注释：**默认周期是3600秒。如需更改重新认证计时器值或者使用RADIUS服务器提供的会话超时时间，输入命令 **authenticationtimer reauthenticate**
步骤 4	**authentication timer** {{[**inactivity** | **reauthenticate** | **restart**]} {*value*}} **示例:** `Device(config-if)#` **authentication timer reauthenticate 180**	设置尝试重新认证的周期描述。 **authentication timer** 关键字含义如下： • **inactivity**——在设置的间隔秒数后如果客户端无活动，则其变为未授权。 • **reauthenticate**——尝试进行自动重新认证的秒数间隔。 • **restart** *value*——尝试认证未授权端口的秒数间隔。 如果启用了周期性重新认证，此命令

		会影响交换机的行为
步骤 5	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式

更改静默周期

当交换机无法认证客户端时，交换机会保持闲置一段时间然后再次尝试认证。接口配置命令 **authentication timer inactivity** 控制着闲置时长。认证失败可能是因为客户端提供了非法的密码。可以设置比默认值更小的时长，为用户提供更快的响应时间。
在特权 EXEC 模式中按照以下步骤更改静默周期。此步骤是可选的。

总步骤

1. configure terminal

2. interface *interface-id*

3. authentication timer inactivity *seconds*

4. end

5. show authentication sessions interface *interface-id*

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/1**	指定要配置的端口，并进入接口配置模式
步骤 3	**authentication timer inactivity** *seconds* **示例:** `Device(config-if)#` **authentication timer inactivity** **30**	设置交换机与客户端认证交换失败之后保持静默状态的秒数。 范围从1到65536秒，默认值是60秒
步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式
步骤 5	**show authentication sessions interface** *interface-id* **示例:** `Device#` **show authentication sessions interface gigabitethernet2/0/1**	验证配置的条目
步骤 6	**copy running-config startup-config** **示例:**	（可选）把配置保存在配置文件中

	Device# **copy running-config startup-config**	

更改交换机到客户端的重传时间

客户端会使用 EAP 应答/身份帧响应来自交换机的 EAP 请求/身份帧。如果交换机没收收到应答，它会等待一定的时长（称为重传时间）然后重新发送数据帧。

注释： 只应在不寻常的情况下更改此命令的默认值，比如存在不可靠的链路或者客户端、认证服务器存在特定的行为问题。

在特权 EXEC 模式中，按照以下步骤更改交换机等待客户端通知的时长。此过程是可选的。

总步骤

1. **configure terminal**
2. **interface** *interface-id*
3. **authentication timer reauthenticate** *seconds*
4. **end**
5. **show authentication sessions interface** *interface-id*
6. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 2	**interface** *interface-id* **示例:** Device(config)# **interface gigabitethernet2/0/1**	指定要配置的端口，并进入接口配置模式
步骤 3	**authentication timer reauthenticate** *seconds* **Example:** Device(config-if)# **authentication timer reauthenticate 60**	设置交换机在重新发送请求之前等待客户端回应EAP请求/身份帧的时长。范围从1到65535秒，默认值是5秒
步骤 4	**end** **示例:** Device(config)# **end**	返回特权 EXEC 模式
步骤 5	**show authentication sessions interface** *interface-id* **示例:** Device# **show authentication sessions interface gigabitethernet2/0/1**	验证配置的条目
步骤 6	**copy running-config startup-config** **示例:** Device# **copy running-config startup-config**	（可选）把配置保存在配置文件中

设置交换机到客户端的帧重传次数

除了更改交换机到客户端的重传时间，还可以更改交换机在重启认证过程之前向客户端发送EAP 请求/身份帧的次数（假设未收到响应）。

注释： 只应在不寻常的情况下更改此命令的默认值，比如存在不可靠的链路或者客户端、认证服务器存在特定的行为问题。

总步骤

1. configure terminal

2. interface *interface-id*

3. dot1x max-reauth-req *count*

4. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/1**	指定要配置的端口，并进入接口配置模式
步骤 3	**dot1x max-reauth-req** *count* **示例:** `Device(config-if)#` **dot1x max-reauth-req 5**	设置交换机在重启认证过程之前向客户端发送EAP请求/身份帧的次数。 范围从1到10，默认值是2
步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式

设置重新认证次数

也可以更改端口变成未授权状态之前交换机重启认证过程的次数。

注释： 只应在不寻常的情况下更改此命令的默认值，比如存在不可靠的链路或者客户端、认证服务器存在特定的行为问题。

总步骤

1. configure terminal

2. interface *interface-id*

3. switchport mode access

4. dot1x max-req *count*

5. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal**	进入全局配置模式。

	示例： `Device#` **configure terminal**	
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/1**	指定要配置的端口，并进入接口配置模式。
步骤 3	**switchport mode access** **示例:** `Device(config-if)#` **switchport mode access**	仅在之前配置了 RADIUS 服务器的情况下把端口设置为接入模式。
步骤 4	**dot1x max-req** *count* **示例:** `Device(config-if)#` **dot1x max-req 4**	设置端口变成未授权状态之前交换机重启认证过程的次数。范围从0到10，默认值是2。
步骤 5	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。

启用 MAC 移动

MAC 移动特性允许已认证的主机从交换机上的一个端口移动到另一个端口。
在特权 EXEC 模式中按照以下步骤在交换机上全局启用 MAC 移动特性。此过程是可选的。
总步骤

1. configure terminal

2. authentication mac-move permit

3. end

4. show running-config

5. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**authentication mac-move permit** **示例:** `Device(config)#` **authentication mac-move permit**	在交换机上启用 MAC 移动特性，该特性默认被禁用。 在会话感知网络模式中，默认设置是 **access-session mac-move deny**.。要在会话感知网络中启用MAC移动，使用全局配置命令**no access-sessionmac-move**。
步骤 3	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。

步骤 4	**show running-config** **示例：** `Device#` **show running-config**	验证配置的条目。
步骤 5	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

启用 MAC 替换

MAC 替换特性允许一台主机替代端口上另一台已认证的主机。
在特权 EXEC 模式中，按照以下步骤在接口上启用 MAC 替换。此过程是可选的。

总步骤

1. configure terminal

2. interface *interface-id*

3. authentication violation {**protect** | **replace** | **restrict** | **shutdown**}

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/2**	指定要配置的端口，并进入接口配置模式。
步骤 3	**authentication violation** {**protect** | **replace** | **restrict** | **shutdown**} **示例:** `Device(config-if)#` **authentication violation replace**	使用 **replace** 关键字在接口上启用 MAC 替换。 端口会移除当前会话并发起对新主机的认证。 其他关键词的效果如下： • **protect**：端口丢弃非预期 MAC 地址的数据包，且不会生成系统消息。 • **restrict**：违规的数据包会被 CPU 丢弃，且会生成系统消息。 • **shutdown**：端口在收到非预期 MAC 地址的数据包时会被错误禁用。
步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。

步骤 5	**show running-config** **示例：** `Device#` **show running-config**	验证配置的条目。
步骤 6	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

配置 802.1x 审计

配置 AAA 系统审计功能进行 802.1x 的审计，会让系统重载发送给审计 RADIUS 服务器的事件。服务器进而可以推测所有活跃的 802.1x 会话都已关闭。

因为 RADIUS 使用不可靠的 UDP 传输协议，审计消息可能因为网络状况差而丢失。在可配置次数的审计请求重传之后，如果交换机没有收到来自 RADIUS 服务器的审计应答消息，会显示以下系统消息：

```
Accounting message %s for session %s failed to receive Accounting Response.
```

当停止消息没有成功发送时，会出现以下消息：

```
00:09:55: %RADIUS-4-RADIUS_DEAD: RADIUS server 172.20.246.201:1645,1646 is not responding.
```

注释： 必须配置 RADIUS 服务器执行审计任务，比如审计开始、停止以及中间更新消息和时间戳。要启用这些功能，可以在 RADIUS 服务器网络配置页中启用“更新/Watchdog 来自此 AAA 客户端的数据包”记录功能。接着，在 RADIUS 服务器的系统配置页启用“CVS RADIUS 审计”。

在交换机上启用 AAA 之后，在特权 EXEC 模式中按照以下步骤配置 802.1x 审计功能。此过程是可选的。

总步骤

1. **configure terminal**
2. **interface** *interface-id*
3. **aaa accounting dot1x default start-stop group radius**
4. **aaa accounting system default start-stop group radius**
5. **end**
6. **show running-config**
7. **copy running-config startup-config**

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/3**	指定要配置的端口，并进入接口配置模式。
步骤 3	**aaa accounting dot1x default start-stop group radius**	使用 RADIUS 服务器列表启用 802.1x 审计。

	示例: `Device(config-if)#` **aaa accounting dot1x default** **start-stop group radius**	
步骤 4	**aaa accounting system default start-stop group radius** **示例:** `Device(config-if)#` **aaa accounting system default** **start-stop group radius**	（可选）启用系统审计功能（使用 RADIUS 服务器列表）并在交换机重启时生成系统审计重启事件消息。
步骤 5	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 6	**show running-config** **示例：** `Device#` **show running-config**	验证配置的条目。
步骤 7	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

配置访客 VLAN

配置访客 VLAN 时，如果服务器没收到 EAP 请求/身份帧的应答，不兼容 802.1x 的客户端会被置于访客 VLAN 中。兼容 802.1x 但是认证失败的客户端不被授权网络访问权限。交换机在单主机或多主机模式中支持使用访客 VLAN。

在特权 EXEC 模式中，按照以下步骤配置访客 VLAN。此过程是可选的。

总步骤

1. **configure terminal**
2. **interface** *interface-id*
3. 使用以下命令之一：
   - **switchport mode access**
   - **switchport mode private-vlan host**
4. **authentication event no-response action authorize vlan** *vlan-id*
5. **end**

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface** **gigabitethernet1/0/3**	指定要配置的端口，并进入接口配置模式。

步骤 3	使用以下命令之一： **• switchport mode access** **• switchport mode private-vlan host** **示例:** `Device(config-if)#` **switchport mode private-vlan host**	• 设置端口为接入模式。 • 配置二层端口为私有 VLAN 主机端口。
步骤 4	**authentication event no-response action authorize vlan** *vlan-id* **示例:** `Device(config-if)#` **authentication event no-response action authorize vlan 2**	指定一个活跃的 VLAN 为 802.1x 访客 VLAN，范围从 1 到 4094。 可以把除了内部 VLAN（被路由端口）、RSPAN VLAN 或语音 VLAN 之外的任意活跃 VLAN 配置为 802.1x 访客 VLAN。
步骤 5	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。

配置受限 VLAN

在交换机堆栈或者交换机上配置受限 VLAN 时，如果认证服务器没有收到合法的客户端用户名及密码，这些兼容 IEEE 802.1x 的客户端会被移动到受限 VLAN 中。交换机只在单主机模式中支持使用受限 VLAN。

在特权 EXEC 模式中，按照以下步骤配置受限 VLAN。此过程是可选的。

总步骤

1. configure terminal

2. interface *interface-id*

3. 使用以下命令之一：

- **switchport mode access**
- **switchport mode private-vlan host**

4. authentication port-control auto

5. authentication event fail action authorize vlan *vlan-id*

6. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/3**	指定要配置的端口，并进入接口配置模式。
步骤 3	使用以下命令之一： **• switchport mode access**	• 设置端口为接入模式。 • 配置二层端口为私有 VLAN 主机端

	• **switchport mode private-vlan host** **示例:** `Device(config-if)#` **switchport mode private-vlan host**	口。
步骤 4	**authentication port-control auto** **示例:** `Device(config-if)#` **authentication port-controlauto**	在端口上启用 802.1x 认证。
步骤 5	**authentication event fail action authorize vlan** *vlan-id* **示例:** `Device(config-if)#` **authentication event fail action authorize vlan 2**	指定一个活跃的 VLAN 为 802.1x 受限 VLAN，范围从 1 到 4094。 可以把除了内部 VLAN（被路由端口）、RSPAN VLAN 或语音 VLAN 之外的任意活跃 VLAN 配置为 802.1x 受限 VLAN。
步骤 6	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。

配置受限 VLAN 上的认证尝试次数

使用接口配置命令 **authentication event retry** *retry count*，可以配置给用户分配受限 VLAN 之前允许的最大认证尝试次数。允许的认证尝试次数范围从 1 到 3，默认值是 3。
在特权 EXEC 模式中，按照以下步骤配置允许的最大认证尝试次数。此过程是可选的。
总步骤

1. **configure terminal**
2. **interface** *interface-id*
3. 使用以下命令之一：
   - **switchport mode access**
   - **switchport mode private-vlan host**
4. **authentication port-control auto**
5. **authentication event fail action authorize vlan** *vlan-id*
6. **authentication event retry** *retry count*
7. **end**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/3**	指定要配置的端口，并进入接口配置模式。

步骤 3	使用以下命令之一： **• switchport mode access** **• switchport mode private-vlan host** **示例:** `Device(config-if)#` **switchport mode private-vlan host**	• 设置端口为接入模式。 • 配置二层端口为私有 VLAN 主机端口。
步骤 4	**authentication port-control auto** **示例:** `Device(config-if)#` **authentication port-controlauto**	在端口上启用 802.1x 认证。
步骤 5	**authentication event fail action authorize vlan** *vlan-id* **示例:** `Device(config-if)#` **authentication event fail action authorize vlan 8**	指定一个活跃的 VLAN 为 802.1x 受限 VLAN，范围从 1 到 4094。 可以把除了内部 VLAN（被路由端口）、RSPAN VLAN 或语音 VLAN 之外的任意活跃 VLAN 配置为 802.1x 受限 VLAN。
步骤 6	**authentication event retry** *retry count* **示例:** `Device(config-if)#` **authentication event retry 2**	配置把用户移动到受限 VLAN 之前允许的最大认证尝试次数。范围从 1 到 3，默认值是 3。
步骤 7	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。

配置 802.1x 不可访问旁路认证以及临界语音 VLAN

在特权 EXEC 模式中，按照以下步骤在端口上配置临界语音 VLAN 并启用不可访问旁路认证特性。

总步骤

1. configure terminal

2. aaa new-model

3. radius-server dead-criteria{**time** *seconds* } [**tries** *number*]

4. radius-serverdeadtime*minutes*

5. radius-server host ip-address *address*[**acct-port** *udp-port*][**auth-port** *udp-port*] [**testusername** *name*[**idle-time** *time*] [**ignore-acct-port**][**ignore auth-port**]] [**key** *string*]

6. dot1x critical {**eapol** | **recovery delay** *milliseconds*}

7. interface *interface-id*

8. authentication event server dead action {**authorize** | **reinitialize**} **vlan** *vlan-id*]

9. switchport voice vlan *vlan-id*

10. authentication event server dead action authorize voice

11. show authentication interface *interface-id*

12. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**aaa new-model** **示例:** `Device(config)#` **aaa new-model**	启用 AAA。
步骤 3	**radius-server dead-criteria**{**time** *seconds* } [**tries** *number*] **示例:** `Device(config)#` **radius-server dead-criteria time 20 tries 10**	设置决定 RADIUS 服务器不可用或 down 的条件。 • **time**——1 到 120 秒。交换机会动态决定一个在 10 到 60 之间的默认 *seconds* 值。 • **number**——1 到 100 次尝试。交换机会动态决定一个在 10 到 100 之间的默认 *number* 值。
步骤 4	**radius-server deadtime** *minutes* **示例:** `Device(config)#` **radius-server deadtime 60**	（可选）设置 RADIUS 服务器不发送请求的分钟数。范围从 0 到 1440 分钟（24 小时）。默认值是 0 分钟。
步骤 5	**radius-server host ip-address** *address*[**acct-port** *udp-port*][**auth-port** *udp-port*] [**testusername** *name*[**idle-time** *time*] [**ignore-acct-port**][**ignore auth-port**]] [**key** *string*] **示例:** `Device(config)#` **radius-server host 1.1.1.2 acct-port 1550 auth-port 1560 test username user1 idle-time 30 key abc1234**	（可选）使用以下关键字配置 RADIUS 服务器参数： • **acct-port** *udp-port*——指定 RADIUS 审计服务器的 UDP 端口。UDP 端口号范围从 0 到 65536，默认值是 1646。 • **auth-port** *udp-port* —— 指定 RADIUS 认证服务器的 UDP 端口。UDP 端口号范围从 0 到 65536，默认值是 1645。 **注释：**可以把 RADIUS 认证服务器及审计服务器的 UDP 端口配置为非默认值。 • **test username** *name*——自动测试 RADIUS 服务器状态，并指定使用的用户名。 • **idle-time** *time*——设置交换机给服务器发送测试包的间隔分钟数。范围从 1 到 35791 分钟，默认值是 60 分钟（1 小时）。 • **ignore-acct-port**——禁用 RADIUS 服务器审计端口的测试。 • **ignore-auth-port**——禁用 RADIUS

		服务器认证端口的测试。 • 使用 **key** *string* 指定交换机和 RADIUS 服务器上的 RADIUS 后台程序之间使用的认证及加密密钥。此密钥是明文密钥，且必须与 RADIUS 服务器使用的加密密钥相同。 **注释：** 把密钥配置在 radius serverhost 命令的最后一项。密钥前的空格会被忽略，但是密钥中间以及之后的空格会被使用。如果密钥使用空格，不要把密钥放在引号之间，除非引号是密钥的一部分。 也可以使用全局配置命令 **radius-server key** {**0**string | **7***string* | *string*}配置认证及加密密钥。
步骤 6	**dot1x critical** {**eapol** | **recovery delay** *milliseconds*} **示例:** `Device(config)#` **dot1x critical eapol** `(config)#` **dot1x critical recovery delay 2000**	（可选）配置不可访问旁路认证参数： • **eapol**——指定交换机在成功认证临界端口时发送一个 EAPOL 成功消息。 • **recovery delay** *milliseconds*——指定在不可用的 RADIUS 服务器变为可用时交换机重新初始化临界端口要等待的恢复时延。范围从 1 到 10000 毫秒，默认值是 1000 毫秒（端口每秒都可以被重新初始化）。
步骤 7	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet 1/0/1**	指定要配置的端口，并进入接口配置模式。
步骤 8	**authentication event server dead action** {**authorize** | **reinitialize**} **vlan** *vlan-id*] **示例:** `Device(config-if)#` **authentication event server dead action reinitialicze vlan 20**	使用以下关键字在 RADIUS 服务器不可用时把端口上主机移动到临界 VLAN： • **authorize**——把任意尝试认证的新主机移动到用户指定的临界 VLAN 中。 • **reinitialize**——把端口上所有已授权的主机移动到用户指定的临界 VLAN 中。
步骤 9	**switchport voice vlan** *vlan-id* **Example:** `Device(config-if)#` **switchport voice vlan**	为端口指定语音 VLAN。语音 VLAN 不能与配置的临界 VLAN 相同。

步骤 10	**authentication event server dead action authorize voice** **示例:** `Device(config-if)#` **authentication event server dead action authorize voice**	配置临界语音 VLAN，如果 RADIUS 服务器不可用时，把端口上的数据流量移动到语音 VLAN 中。
步骤 11	**show authentication interface** *interface-id* **示例:** `Device(config-if)#` **do show authentication interface gigabit 1/0/1**	（可选）验证配置的条目。
步骤 12	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

要返回RADIUS服务器的默认设置，使用全局配置命令**no radius-server dead-criteria**，**radius-serverdeadtime**，和**no radius-server host**。要禁用不可访问旁路认证，使用接口配置命令**no authentication event server dead action**。要禁用临界语音VLAN，使用接口配置命令**noauthentication event server dead action authorize voice**。

配置不可访问旁路认证的示例

以下示例展示了如果配置不可访问旁路认证特性：

```
Device(config)# radius-server dead-criteria time 30 tries 20
Device(config)# radius-server deadtime 60
Device(config)# radius-server host 1.1.1.2 acct-port 1550 auth-port 1560 test username user1
idle-time 30 key abc1234
Device(config)# dot1x critical eapol
Device(config)# dot1x critical recovery delay 2000
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# dot1x critical
Device(config-if)# dot1x critical recovery action reinitialize
Device(config-if)# dot1x critical vlan 20
Device(config-if)# end
```

配置 802.1x 认证以及 WoL

在特权 EXEC 模式中，按照以下步骤启用 802.1x 以及 WoL。此过程是可选的。

总步骤

1. configure terminal

2. interface *interface-id*

3. authentication control-direction {**both** | **in**}

4. end

5. show authentication sessions interface *interface-id*

6. copy running-config startup-config

具体步骤

	命令或操作	目的

步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/3**	指定要配置的端口，并进入接口配置模式。
步骤 3	**authentication control-direction** {**both** | **in**} **示例:** `Device(config-if)#` **authentication control-direction both**	在端口上启用 802.1x 认证以及 WoL，使用以下关键字配置端口执行单向操作或双向操作。 • **both**——设置端口为双向。端口无法收发主机的数据包。默认情况下端口为双向。 • **in**——设置端口为单向。端口可以向主机发送数据包，但不能接收来自主机的数据包。
步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**show authentication sessions interface** *interface-id* **示例:** `Device#` **show authentication sessions interface gigabitethernet2/0/3**	验证配置的条目。
步骤 6	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

配置 MAC 旁路认证

在特权 EXEC 模式中，按照以下步骤启用 MAC 旁路认证。此过程是可选的。

总步骤

1. configure terminal

2. interface *interface-id*

3. authentication port-control auto

4. mab [**eap**]

5. end

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。

步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/1**	指定要配置的端口，并进入接口配置模式。
步骤 3	**authentication port-control auto** **示例:** `Device(config-if)#` **authentication port-control auto**	在端口上启用 802.1x 认证。
步骤 4	**mab** [**eap**] **示例:** `Device(config-if)#` **mab**	启用 MAC 旁路认证。 （可选）使用关键字 **eap** 配置交换机使用 EAP 进行认证。
步骤 5	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。

配置 802.1x 用户分配

在特权 EXEC 模式中，按照以下步骤配置 VLAN 群组并映射 VLAN 到其中。

总步骤

1. configure terminal

2. vlan group *vlan-group-name* **vlan-list** *vlan-list*

3. end

4. no vlan group *vlan-group-name* **vlan-list** *vlan-list*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**vlan group** *vlan-group-name* **vlan-list** *vlan-list* **示例:** `Device(config)#` **vlan group eng-dept vlan-list 10**	配置 VLAN 群组，并把一个 VLAN 或一个范围的 VLAN 映射到其中。
步骤 3	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**no vlan group** *vlan-group-name* **vlan-list** *vlan-list* **示例:** `Device(config)#` **no vlan group eng-dept vlan-list**	清理 VLAN 群组配置或 VLAN 群组配置中的某个元素。

	10	

配置 VLAN 群组的示例

以下示例显示了如何配置 VLAN 群组，映射 VLAN 到群组以及验证 VLAN 群组的配置及映射。

```
Device(config)# vlan group eng-dept vlan-list 10
Device(config)# show vlan group group-name eng-dept
Group Name Vlans Mapped
------------ --------------
eng-dept 10
Device(config)# show dot1x vlan-group all
Group Name Vlans Mapped
------------ --------------
eng-dept 10
hr-dept
```

以下示例显示了如何把 VLAN 添加到现有的 VLAN 群组，并验证操作。

```
Device(config)# vlan group eng-dept vlan-list 30
Device(config)# show vlan group eng-dept
Group Name        Vlans Mapped
-------------     --------------
eng-dept          10,30
```

以下示例显示了如何从 VLAN 群组中移除 VLAN。

```
Device# no vlan group eng-dept vlan-list 10
```

以下示例显示了当 VLAN 群组中的所有 VLAN 都被清除时，VLAN 群组也会被清除。

```
Device(config)# no vlan group eng-dept vlan-list 30
Vlan 30 is successfully cleared from vlan group eng-dept.
Device(config)# show vlan group group-name eng-dept
```

以下示例显示了如何清除所有 VLAN 群组。

```
Device(config)# no vlan group end-dept vlan-list all
Device(config)# show vlan-group all
```

更多有关以上命令的信息，参见 *Inspur INOS 安全性命令手册*。

配置 NAC 二层 802.1x 验证

可以配置 NAC 二层 802.1x 验证特性 。

在特权 EXEC 模式中，按照以下步骤配置 NAC 二层 802.1x 验证。此过程是可选的。

总步骤

1. **configure terminal**
2. **interface** *interface-id*
3. **switchport mode access**
4. **authentication event no-response action authorize vlan** *vlan-id*
5. **authentication periodic**
6. **authentication timer reauthenticate**
7. **end**
8. **show authentication sessions interface** *interface-id*
9. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/3**	指定要配置的端口，并进入接口配置模式。
步骤 3	**switchport mode access** **示例:** `Device(config-if)#` **switchport mode access**	仅在配置了 RADIUS 服务器时设备端口为接入模式。
步骤 4	**authenticationevent no-response actionauthorize vlan** *vlan-id* **示例:** `Device(config-if)#` **authentication event no-response action authorize vlan 8**	指定一个活跃的 VLAN 为 802.1x 访客 VLAN，范围从 1 到 4094。 可以把除了内部 VLAN（被路由端口）、RSPAN VLAN 或语音 VLAN 之外的任意活跃 VLAN 配置为 802.1x 访客 VLAN。
步骤 5	**authentication periodic** **示例:** `Device(config-if)#` **authentication periodic**	启用对客户端的周期性重新认证，该特性默认禁用。
步骤 6	**authentication timer reauthenticate** **示例:** `Device(config-if)#` **authentication timer reauthenticate**	设置尝试重新认证客户端（设置为 1 小时）。 如果启用周期性重新认证，此命令会影响交换机行为。
步骤 7	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。
步骤 8	**show authentication sessions interface** *interface-id* **示例:** `Device#` **show authentication sessions interface gigabitethernet2/0/3**	验证配置的条目。
步骤 9	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

配置认证交换机以及 NEAT

配置此特性要求配线间外的一台交换机被配置为请求者，且连接到认证交换机。

注释： 必须在 ACS 上把 inspur-av-pair 配置为 *device-traffic-class=switch*，此配置会在请求者认证成功后把连接接口设置为中继。

在特权 EXEC 模式中，按照以下步骤配置交换机为认证者。

总步骤

1. configure terminal

2. cisp enable

3. interface *interface-id*

4. switchport mode access

5. authentication port-control auto

6. dot1x pae authenticator

7. spanning-tree portfast

8. end

9. show running-config interface *interface-id*

10. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**cisp enable** **示例:** `Device(config)#` **cisp enable**	启用 CISP。
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/3**	指定要配置的端口，并进入接口配置模式。
步骤 4	**switchport mode access** **示例:** `Device(config-if)#` **switchport mode access**	设置端口模式为接入模式。
步骤 5	**authentication port-control auto** **示例:** `Device(config-if)#` **authentication port-control auto**	设置端口认证模式为 auto。
步骤 6	**dot1x pae authenticator** **示例:** `Device(config-if)#` **dot1x pae authenticator**	配置接口为端口接入实体（PAE）认证者。
步骤 7	**spanning-tree portfast** **示例:** `Device(config-if)#` **spanning-tree portfast trunk**	在连接到一台工作站或服务器的接入端口上启用 Port Fast。
步骤 8	**end** **示例:**	返回特权 EXEC 模式。

	Device(config-if)# **end**	
步骤 9	**show running-config interface** *interface-id* **示例:** Device# **show running-config interface gigabitethernet2/0/1**	验证配置的条目。
步骤 10	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把配置保存在配置文件中。

配置请求交换机以及 NEAT

在特权 EXEC 模式中，按照以下步骤配置交换机为请求者。

总步骤

1. configure terminal

2. cisp enable

3. dot1x credentials *profile*

4. username *suppswitch*

5. password *password*

6. dot1x supplicant force-multicast

7. interface *interface-id*

8. switchport trunk encapsulation dot1q

9. switchport mode trunk

10. dot1x pae supplicant

11. dot1x credentials *profile-name*

12. end

13. show running-config interface *interface-id*

14. copy running-config startup-config

15. 配置 NEAT 以及自动智能端口宏（Auto Smartports Macros）

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式。
步骤 2	**cisp enable** **示例:** Device(config)# **cisp enable**	启用 CISP。
步骤 3	**dot1x credential**s *profile* **示例:** Device(config)# **dot1x credentials test**	创建 802.1x 凭据配置。此配置必须挂接到配置为请求者的端口上。
步骤 4	**username** *suppswitch* **示例:** Device(config)# **username suppswitch**	创建用户名。
步骤 5	**password** *password*	为新用户名创建密码。

	示例: `Device(config)#` **password myswitch**	
步骤 6	**dot1x supplicant force-multicast** **示例:** `Device(config)#` **dot1x supplicant force-multicast**	强制交换机只发送组播 EAPOL 包，无论其收到了单播包还是组播包。 这也使 NEAT 可以适用于所有主机模式的请求交换机。
步骤 7	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的端口，并进入接口配置模式。
步骤 8	**switchport trunk encapsulation dot1q** **示例:** `Device(config-if)#` **switchport trunk encapsulation dot1q**	甚至端口为中继模式。
步骤 9	**switchport mode trunk** **示例:** `Device(config-if)#` **switchport mode trunk**	配置接口为 VLAN 中继端口。
步 骤 10	**dot1x pae supplicant** **示例:** `Device(config-if)#` **dot1x pae supplicant**	配置接口为端口接入实体（PAE）请求者。
步 骤 11	**dot1x credentials** *profile-name* **示例:** `Device(config-if)#` **dot1x credentials test**	把 802.1x 凭据配置挂接到接口上。
步 骤 12	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。
步 骤 13	**show running-config interface** *interface-id* **示例:** `Device#` **show running-config interface gigabitethernet2/0/1**	验证配置的条目。
步 骤 14	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。
步 骤 15	配置NEAT以及自动智能端口宏	配置授权交换机时，也可以使用自动智能端口用户定义宏来代替交换机的 VSA。更多信息，参见此版本的*自动智能端口配置指南*。

配置 802.1x 认证、可下载的 ACL 以及重定向 URL

除了需要在交换机上配置 802.1x 认证之外，还需要配置 ACS。更多信息参见 *Inspur 安全 ACS 4.2 配置指南*:

http://www.icntnetworks.com

注释： 在交换机下载 ACL 之前，必须在 ACS 上配置一个可下载的 ACL。
在端口认证之后，可以使用特权 EXEC 命令 **show ip access-list** 显示端口下载的 ACL。

配置可下载的 ACL

当客户端完成认证且客户端的 IP 地址被加入 IP 设备追踪表之后，该策略生效。交换机随后会把可下载 ACL 应用到端口上。

总步骤

1. **configure terminal**
2. **ip device tracking**
3. **aaa new-model**
4. **aaa authorization network default local group radius**
5. **radius-server vsa send authentication**
6. **interface** *interface-id*
7. **ip access-group** *acl-id* **in**
8. **show running-config interface** *interface-id*
9. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**ip device tracking** **示例:** `Device(config)#` **ip device tracking**	设置 IP 设备追踪表。
步骤 3	**aaa new-model** **示例:** `Device(config)#` **aaa new-model**	启用 AAA。
步骤 4	**aaa authorization network default local group radius** **示例:** `Device(config)#` **aaa authorization network default local group radius**	设备本地授权方式。要移除授权方式，使用命令**no aaa authorizationnetwork default local group radius**。
步骤 5	**radius-server vsa send authentication** **示例:** `Device(config)#` **radius-server vsa send authentication**	配置 RADIUS VSA 发送认证。
步骤 6	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/4**	指定要配置的端口，并进入接口配置模式。
步骤 7	**ip access-group** *acl-id* **in** **示例:**	在端口输入方向配置默认 ACL。 **注释：** *acl-id* 是访问列表的命令或编号。

	Device(config-if)# **ip access-group default_acl in**	
步骤 8	**show running-config interface** *interface-id* **示例:** Device# **show running-config interface gigabitethernet2/0/1**	验证配置。
步骤 9	**copy running-config startup-config** **示例:** Device# **copy running-config startup-config**	（可选）把配置保存在配置文件中。

配置可下载的策略

在特权 EXEC 模式中执行以下配置：

总步骤

1. configure terminal

2. access-list *access-list-number* **{ deny | permit } { hostname | any | host } log**

3. interface *interface-id*

4. ip access-group *acl-id* **in**

5. exit

6. aaa new-model

7. aaa authorization network default group radius

8. ip device tracking

9. ip device tracking probe [**count** | **interval** | **use-svi**]

10. radius-server vsa send authentication

11. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** Device# **configure terminal**	进入全局配置模式。
步骤 2	**access-list** *access-list-number* **{ deny | permit } { hostname | any | host } log** **示例:** Device(config)# **access-list 1 deny any log**	定义默认的端口 ACL。 *access-list-number* 是十进制数，从 1 到 99 或 1300 到 1999。 输入 **deny** 或 **permit** 指定匹配后拒绝或允许访问。 源是发送数据包的网络或主机源地址，如： • **hostname**：32 位的点分十进制数。 • **any**：此关键字是源以及源通配值 0.0.0.0 255.255.255.255 的缩写。无需输入源通配值。 • **host**：此关键字是源以及源通配 0.0.0.0 的缩写。

		（可选）把源通配应用到源。 （可选）输入 log 把数据包匹配条目的通知信息发送给控制台。
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet2/0/2**	进入接口配置模式。
步骤 4	**ip access-group** *acl-id* **in** **示例:** `Device(config-if)#` **ip access-group default_acl in**	配置端口进入方向的默认ACL。 **注释：***acl-id*是访问列表的名称或编号。
步骤 5	**exit** **示例:** `Device(config-if)#` **exit**	返回全局配置模式。
步骤 6	**aaa new-model** **示例:** `Device(config)#` **aaa new-model**	启用 AAA。
步骤 7	**aaa authorization network default group radius** **示例:** `Device(config)#` **aaa authorization network default group radius**	设置本地授权方式。要移除授权方式，使用命令 **no aaa authorization network default group radius**。
步骤 8	**ip device tracking** **示例:** `Device(config)#` **ip device tracking**	启用 IP 设备追踪表。 要禁用 IP 设备追踪表，使用全局配置命令 **no ip device tracking**。
步骤 9	**ip device tracking probe** [**count** | **interval** | **use-svi**] **示例:** `Device(config)#` **ip device tracking probe count**	（可选）配置 IP 设备追踪表： • **count** *count*——设置交换机发送 ARP 探测帧的次数。范围从 1 到 5，默认值是 3。 • **interval** *interval*——设置交换机重发 ARP 探测帧之前等待的秒数。范围从 30 到 300 秒，默认值是 30 秒。 • **use-svi**——使用交换机虚接口（SVI）的 IP 地址作为 ARP 探测帧的源地址。
步骤 10	**radius-server vsa send authentication** **示例:** `Device(config)#` **radius-server vsa send authentication**	配置网络接入服务器识别并使用厂商特定的属性。 **注释：**可下载的 ACL 必须可操作。
步骤 11	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。

配置基于 VLAN ID 的 MAC 认证

在特权 EXEC 模式中按照以下步骤进行配置。

总步骤

1. configure terminal

2. mab request format attribute 32 vlan access-vlan

3. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**mab request format attribute 32 vlan access-vlan** **示例:** `Device(config)#` **mab request format attribute 32 vlan access-vlan**	启用基于 VLAN ID 的 MAC 认证。
步骤 3	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

配置灵活认证顺序

以下示例的命令改变了灵活认证排序的顺序，让 MAB 在 IEEE 802.1x 认证（dot1x）之前尝试。MAB 被配置为第一个认证方式，所以将优先于所有其他的认证方式。

注释： 在更改默认认证方式的顺序以及优先级时，应该理解更改操作的潜在后果。详情参见 http://www.icntnetworks.com

在特权 EXEC 模式中按照以下步骤进行配置。

总步骤

1. configure terminal

2. interface *interface-id*

3. switchport mode access

4. authentication order [**dot1x** | **mab**] | {**webauth**}

5. authentication priority [**dot1x** | **mab**] | {**webauth**}

6. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id*	指定要配置的端口，并进入接口配置

	示例: `Device(config)#` **interface gigabitethernet 1/0/1**	模式。
步骤 3	**switchport mode access** **示例:** `Device(config-if)#` **switchport mode access**	仅在配置了 RADIUS 服务器后把端口设置为接入模式。
步骤 4	**authentication order [dot1x | mab] | {webauth}** **示例:** `Device(config-if)#` **authentication order mab dot1x**	（可选）设置端口上使用的认证方式顺序。
步骤 5	**authentication priority [dot1x | mab] | {webauth}** **示例:** `Device(config-if)#` **authentication priority mab** **dot1x**	（可选）为端口优先级列表添加认证方式。
步骤 6	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。

配置 Open1x

在特权 EXEC 模式中，按照以下步骤手工控制端口的授权状态。

总步骤

1. configure terminal

2. interface *interface-id*

3. switchport mode access

4. authentication control-direction {**both** | **in**}

5. authentication fallback *name*

6. authentication host-mode [**multi-auth** | **multi-domain** | **multi-host** | **single-host**]

7. authentication open

8. authentication order [**dot1x** | **mab**] | {**webauth**}

9. authentication periodic

10. authentication port-control {**auto** | **force-authorized** | **force-un authorized**}

11. end

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id*	指定要配置的端口，并进入接口配置

	示例: `Device(config)#` **interface gigabitethernet 1/0/1**	模式。
步骤 3	**switchport mode access** **示例:** `Device(config-if)#` **switchport mode access**	仅在配置了 RADIUS 服务器后把端口设置为接入模式。
步骤 4	**authentication control-direction {both | in}** **示例:** `Device(config-if)#` **authentication control-direction both**	（可选）配置单向或双向的端口控制。
步骤 5	**authentication fallback** *name* **示例:** `Device(config-if)#` **authentication fallback profile1**	（可选）配置端口为不支持 802.1x 认证的客户端使用网页认证作为备用方式。
步骤 6	**authentication host-mode [multi-auth | multi-domain | multi-host | single-host]** **示例:** `Device(config-if)#` **authentication host-mode multi-auth**	（可选）设置端口上的授权管理器模式。
步骤 7	**authentication open** **示例:** `Device(config-if)#` **authentication open**	（可选）在端口上启用或禁用开放访问。
步骤 8	**authentication order [dot1x | mab] | {webauth}** **示例:** `Device(config-if)#` **authentication order dot1x webauth**	（可选）设置端口使用的认证方式顺序。
步骤 9	**authentication periodic** **示例:** `Device(config-if)#` **authentication periodic**	（可选）在端口上启用或禁用重新认证。
步 骤 10	**authentication port-control {auto | force-authorized | force-un authorized}** **示例:** `Device(config-if)#` **authentication port-control auto**	（可选）启用端口授权状态的手工控制。
步 骤	**end**	返回特权 EXEC 模式。

11	**示例:** `Device(config-if)#` **end**	

在端口上禁用 802.1x 认证

可以使用接口配置命令 **no dot1x pae** 在端口上禁用 802.1x 认证。
在特权 EXEC 模式中，按照以下步骤在端口上禁用 802.1x 认证。此过程是可选的。
总步骤

1. configure terminal

2. interface *interface-id*

3. switchport mode access

4. no dot1x pae authenticator

5. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet 2/0/1**	指定要配置的端口，并进入接口配置模式。
步骤 3	**switchport mode access** **示例:** `Device(config-if)#` **switchport mode access**	（可选）仅在配置了 RADIUS 服务器后把端口设置为接入模式。
步骤 4	**no dot1x pae authenticator** **示例:** `Device(config-if)#` **no dot1x pae authenticator**	在端口上禁用 802.1x 认证。
步骤 5	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。

把 802.1x 认证配置重置为默认值

在特权 EXEC 模式中，按照以下步骤把 802.1x 认证配置重置为默认值。此过程是可选的。
总步骤

1. configure terminal

2. interface *interface-id*

3. dot1x default

4. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet 1/0/2**	指定要配置的端口，并进入接口配置模式。
步骤 3	**dot1x default** **示例:** `Device(config-if)#` **dot1x default**	重置 802.1x 参数为默认值。
步骤 5	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。

监控 802.1x 统计信息及状态

表 149：特权 EXECshow 命令

命令	目的
show dot1x all statistics	显示所有端口的 802.1x 统计信息。
show dot1x interface *interface-id* **statistics**	显示特定端口的 802.1x 统计信息。
show dot1x all [**count** | **details** | **statistics** | **summary**]	显示交换机的 802.1x 管理状态及运行状态。
show dot1x interface *interface-id*	显示特定端口的 802.1x 管理状态及运行状态。

表 150：全局配置命令

命令	目的
no dot1x logging verbose	过滤详细的 802.1x 认证消息（从 Inspur INOS 12.2(55)SE 版本之后支持）。

有关显示字段的详细信息，查看此版本的命令手册。

其他参考资料

相关文档

相关主题	文档标题
为会话感知网络配置身份控制策略以及身份服务模板。	Inspur INOS 会话感知网络配置指南（Inspur 6850 交换机） http://www.icntnetworks.com
配置 RADIUS、TACACS+、安全 Shell、802.1x 以及 AAA。	Inspur INOS 保护用户服务配置指南库（Inspur 6850 交换机）

	http://www.icntnetworks.com

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

IPv4 访问控制列表的特性信息

版本	特性信息
Inspur INOS 12.2	IPv4 访问控制列表执行数据包过滤，控制数据包在网络之间的移动。其控制功能限制网络流量，约束用户及设备的网络访问，并阻止流量离开网络，进而提供安全性。
Inspur INOS 12.2	命名的 ACL 允许访问控制条目使用非连续的端口号，让管理员可以在一个访问控制条目中指定非连续的端口，极大地减少了访问列表中源地址、目的地址以及协议都相同，但是端口不同的条目数量。

配置基于端口的流量控制

基于端口的流量控制概述

基于端口的流量控制是 Inspur 交换机上的一组二层特性，可以用来在端口级别过滤或阻塞满足特定流量条件的数据包。本配置指南所述的 Inspur INOS 版本支持的基于端口的流量控制特性如下：

- 风暴控制
- 保护端口
- 端口阻塞
- 端口安全
- 协议风暴保护

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于风暴控制的信息

风暴控制

风暴控制特性防止 LAN 上的流量因为物理端口上的广播、组播或者单播风暴而中断。LAN 风暴发生时，数据包在 LAN 中被泛洪，产生了过量的流量，并导致网络性能降级。造成风暴的原因可能是协议栈实现中的错误，网络配置的错误，还可能是用户发起了拒绝服务攻击。

风暴控制（或称流量抑制）监控从接口发往交换总线的数据包，并确定数据包是单播、组播、还是广播的。交换机会记录 1 秒的间隔时间内接收的特定类型的数据包数量，并将其与预定义的抑制等级门限值进行比较。

如何测量流量活动

风暴控制特性使用以下方式之一来测量流量活动：

- 广播、组播以及单播流量能够使用的端口总可用带宽的百分比。
- 接收广播、组播以及单播流量的流量速率，单位是数据包每秒。

- 接收广播、组播以及单播流量的流量速率，单位是比特每秒。
- 小数据帧的流量速率，单位是数据包每秒。此特性全局启用。会为每个接口配置小数据帧的门限值。

对于每种方式，速率达到上升门限值时，端口会阻塞流量。端口保持阻塞，直到流量速率下降到下降门限值以下，端口恢复正常的转发。如果没有指定下降抑制等级，在流量速率下降到上升抑制等级以下之前，交换机会阻塞所有流量。通常来说，抑制等级越高，对广播风暴的防护效果越小。

注释： 达到组播流量的风暴控制门限值时，除了如网桥协议数据单元（BPDU）以及 Inspur 发现协议（CDP）这样的控制流量，所有的组播流量都会被阻塞。然而，交换机不会区分如 OSPF 这样的路由更新与常规的组播数据流量，所以这两类流量都会被阻塞。

流量模式

以下示例展示了特定时间段内接口上的广播流量模式。

图 130：广播风暴控制示例

Forwarded traffic	转发流量
Blocked traffic	阻塞流量
Threshold	门限值
Time	时间
Total number of broadcast packets or bytes	广播的数据包或字节总数

在 T1、T2 以及 T4、T5 时间间隔内，转发的广播流量超过了配置的门限值。当特定流量总量超过门限值时，该类型的所有流量在下一个时间段内都会被丢弃。因此，在之后的 T2 和 T5 的间隔内，广播流量被阻塞。在下一个时间间隔中（如 T3），如果广播流量没有超过门限值，它将再次被转发。

风暴控制抑制等级与 1 秒时间间隔组合起来控制了风暴控制算法的工作方式。更高的门限值允许更多的数据包通过。门限值 100%表示不对流量进行限制，0.0 表示端口上的所有广播、组播或单播流量都会被阻塞。

注释： 因为数据包不会按照均匀的间隔到达，在测量流量活动时 1 秒的时间间隔就可以影响风暴控制的行为。

可以使用接口配置命令 **storm-control** 设置每类流量的门限值。

如何配置风暴控制

配置风暴控制及门限值等级

可以在端口上配置风暴控制并输入希望为特定类型流量使用的门限值等级。

然而，因为硬件的限制以及对不同尺寸的数据包统计方式不同，门限值的百分比是近似值。根据组成入向流量的数据包大小不同，实际使用的门限值可与配置的等级有几个百分点的差别。

注释： 风暴控制支持在物理端口上使用。也可以在 EtherChannel 上配置风暴控制。对

EtherChannel 进行配置时，风暴控制设置会传播到 EtherChannel 物理接口上。

按照以下步骤配置风暴控制以及门限值等级。

在开始前

风暴控制支持在物理端口上使用。也可以在 EtherChannel 上配置风暴控制。对 EtherChannel 进行配置时，风暴控制设置会传播到 EtherChannel 物理接口上。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. storm-control {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps*[*bps-low*] | **pps***pps*[*pps-low*]}

5. storm-control action {**shutdown** | **trap**}

6. end

7. show storm-control [*interface-id*] [**broadcast** | **multicast** | **unicast**]

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/1**	指定要被配置的接口，并进入接口配置模式。
步骤 4	**storm-control** {**broadcast** | **multicast** | **unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps***pps*[*pps-low*]} **示例:** `Device(config-if)#`**storm-control unicast level 87 65**	配置广播、组播或单播风暴控制。默认情况下，风暴控制被禁用。 关键字含义如下： • *level* 字段以带宽百分比的形式（最多到小数点后两位）指定广播、组播或单播流量的上升门限值等级。速率达到上升门限值时端口阻塞流量。范围从 0.00 到 100.00。 • （可选）*level-low* 字段以带宽百分比的形式（最多到小数点后两位）指定下降门限值等级。该值必须小于或等于上升抑制值。端口在流量降到该等级以下时进行流量转发。如果不配置下降抑制等级，其被设置为上升抑制等级。范围从 0.00 到 100.00。 如果设置门限值为最大值

		（100%），对流量不进行限制。如果设置门限值为 0.0，端口上的所有广播、组播以及单播流量都会被阻塞。 • **bps** *bps* 字段以比特每秒（最多到小数点后一位）指定广播、组播或单播流量的上升门限值等级。速率达到上升门限值时端口阻塞流量。范围从 0.0 到 10000000000.0。 • （可选）*bps-low* 字段以比特每秒（最多到小数点后一位）指定下降门限值等级。该值必须小于或等于上升抑制值。端口在流量降到该等级以下时进行流量转发。范围从 0.0 到 10000000000.0。 • **pps***pps* 字段以数据包每秒（最多到小数点后一位）指定广播、组播或单播流量的上升门限值等级。速率达到上升门限值时端口阻塞流量。范围从 0.0 到 10000000000.0。 • （可选）*pps-low* 以数据包每秒（最多到小数点后一位）指定下降门限值等级。该值必须小于或等于上升抑制值。端口在流量降到该等级以下时进行流量转发。范围从 0.0 到 10000000000.0。 对于 BPS 及 PPS 设置，较大的门限值数可以使用度量后缀，如 k、m 和 g。
步骤 5	**storm-control action** {**shutdown** | **trap**} **示例:** `Device(config-if)#`**storm-control action trap**	指定检测到风暴时要采取的行为。默认行为是过滤流量且不发送陷阱（trap）。 • 选择 **shutdown** 关键字，在风暴期间设置端口为错误禁用。 • 选择 **trap** 关键字，在检测到风暴时生成 SNMP 陷阱。
步骤 6	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。
步骤 7	**show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**] **示例:** `Device#` **show storm-control**	验证在端口上为特定的流量类型设置的风暴控制抑制等级。如果不输入流量类型，会显示广播风暴控制的设置。

	gigabitethernet1/0/1 unicast	
步骤 8	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

配置小数据帧的到达速率

入向有 VLAN 标记且小于 67 字节的数据包被认为是小数据帧。它们会被交换机转发，但不会使交换机风暴控制计数器增加。

可以在交换机上全局启用小数据帧到达特性，为每个接口上的数据包配置小数据帧门限值。小于最小大小且按照特定速率（门限值）到达的数据包会被丢弃，因为此时端口会被错误禁用。

总步骤

1. enable

2. configure terminal

3. errdisable detect cause small-frame

4. errdisable recovery interval *interval*

5. errdisable recovery cause small-frame

6. interface *interface-id*

7. small-frame violation-rate *pps*

8. end

9. show interfaces *interface-id*

10. show running-config

11. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**errdisable detect cause small-frame** **示例:** `Device(config)#` **errdisable detect cause small-frame**	在交换机上启用小数据帧到达速率特性。
步骤 4	**errdisable recovery interval** *interval* **示例:** `Device(config)#` **errdisable recovery interval60**	（可选）指定从错误禁用状态恢复的时间。
步骤 5	**errdisable recovery cause small-frame** **示例:** `Device(config)#` **errdisable recovery cause**	（可选）配置错误禁用端口的恢复时间，使端口因到达的小数据帧被错误禁用后能自动重新启用。

	small-frame	风暴控制支持在物理端口上使用。也可以在 EtherChannel 上配置风暴控制。对 EtherChannel 进行配置时，风暴控制设置会传播到 EtherChannel 物理接口上。
步骤 6	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/2**	指定要配置的接口，并进入接口配置模式。
步骤 7	**small-frame violation-rate** *pps* **示例:** `Device(config-if)#`**small-frame violation rate 10000**	配置接口丢弃入向数据包并被错误禁用的门限值。范围从 1 到 10000 包每秒（pps）。
步骤 8	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 9	**show interfaces** *interface-id* **示例:** `Device#` **show interfaces gigabitethernet1/0/2**	验证配置。
步骤 10	**show running-config** **示例:** `Device#` **show running-config**	验证配置的条目。
步骤 11	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

关于保护端口的信息

保护端口

一些应用会要求二层流量不在相同交换机的端口之间进行转发，使得一个邻居看不到另一个邻居产生的流量。在这样的环境中，使用保护端口能保证交换机的这些端口之间没有单播、组播或广播流量交换。

保护端口的特性如下：

- 保护端口不会把任何流量（单播、组播或广播）转发给其他的保护端口。数据流量在二层上不能在保护端口之间转发；只有如 PIM 包这样的控制流量会被转发，因为这些包会被 CPU 处理并在软件中转发。所有保护端口之间的数据流量必须通过三层设备进行转发。
- 保护端口和非保护端口之间的转发行为照常进行。

因为一个交换机堆栈代表一台逻辑交换机，所以二层流量不会在交换机堆栈中的保护端口之间转发，无论保护端口是否在堆栈中相同的交换机上。

保护端口的默认配置

默认无保护端口定义。

保护端口指南

可以在物理接口（如吉比特以太网端口 1）或者 EtherChannel 群组（如端口通道 5）上配置保护端口。对端口通道启用保护端口，该特性会为端口通道组中的所有端口启用。

如何配置保护端口

配置保护端口

在开始前

没有预定义的保护端口。以下将配置一个保护端口。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport protected

5. end

6. show interfaces *interface-id* **switchport**

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的接口，并进入接口配置模式。
步骤 4	**switchport protected** **示例:** `Device(config-if)#` **switchport protected**	配置接口为保护端口。
步骤 5	**end**	返回特权 EXEC 模式。

	示例: `Device(config)#` **end**	
步骤 6	**show interfaces** *interface-id* **switchport** **示例:** `Device#` **show interfaces gigabitethernet1/0/1 switchport**	验证配置。
步骤 7	**show running-config** **示例:** `Device#` **show running-config**	验证配置的条目。
步骤 8	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

监控保护端口

表 153：显示保护端口设置

命令	目的
show interfaces [*interface-id*] **switchport**	显示所有交换（非路由）端口或特定端口的管理状态以及运行状态，包括端口阻塞和端口保护设置。

有关端口阻塞的信息

端口阻塞

默认情况下，交换机会把带有未知目的 MAC 地址的数据包从所有端口泛洪出去。如果未知的单播或组播流量被转发到了保护端口上，可能会造成安全问题。为了避免未知的单播或组播流量被从一个端口转发到另一个端口，管理员可以阻止一个端口（保护或非保护端口）把未知的单播或组播数据包发往其他端口。

注释： 对于组播流量，端口阻塞特性只会阻止纯二层的数据包。报头中包含 IPv4 或 IPv6 信息的组播数据包不会被阻塞。

如何配置端口阻塞

阻塞端口上泛洪流量

在开始前

接口可以是物理接口，也可以是 EtherChannel 组。配置阻塞端口通道上的组播或单播流量

时，端口通道组中的所有端口都会进行阻塞。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport block multicast

5. switchport block unicast

6. end

7. show interfaces *interface-id* **switchport**

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的接口，并进入接口配置模式。
步骤 4	**switchport block multicast** **示例:** `Device(config-if)#` **switchport block multicast**	阻塞从端口转发出的未知组播流量。 **注释：**只阻塞纯二层组播流量。报头中包含 IPv4 或 IPv6 信息的组播数据包不会被阻塞。
步骤 5	**switchport block unicast** **示例:** `Device(config-if)#` **switchport block unicast**	阻塞从端口转发出的未知单播流量。
步骤 6	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 7	**show interfaces** *interface-id* **switchport** **示例:** `Device#` **show interfaces gigabitethernet1/0/1 switchport**	验证配置。
步骤 8	**show running-config** **示例:** `Device#` **show running-config**	验证配置的条目。
步骤 9	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

监控端口阻塞

表 154：显示端口阻塞特性设置的命令

命令	目的
show interfaces [*interface-id*] **switchport**	显示所有交换（非路由）端口或特定端口的管理状态以及运行状态，包括端口阻塞和端口保护设置。

端口安全的前提条件

注释： 如果尝试设置的最大数值小于已经在端口上配置的安全地址数量，配置的命令会被拒绝。

端口安全的限制

可以在交换机或者交换机堆栈上配置的最大安全 MAC 地址数量由系统中允许的最大可用 MAC 地址数量决定。此数值由活跃的交换机数据库管理（Switch Database Management，SDM）模板决定。此数值是总的可用 MAC 地址数量，包括用于其他二层功能的地址以及接口上配置的任何其他安全 MAC 地址。

有关端口安全的信息

端口安全

使用端口安全特性，可以限制并标识允许访问端口的工作站的 MAC 地址，进而限制对端口的输入。在为安全端口分配安全 MAC 地址时，该端口不会转发源地址在定义的地址组之外的数据包。如果把安全 MAC 地址的数量限制为 1 并分配了一个安全 MAC 地址，能够确保连接到端口的工作站使用端口的全部带宽。

如果端口被配置为安全端口，且到达了其配置的最大安全 MAC 地址数量，当尝试访问端口的工作站的 MAC 地址与已标识的安全 MAC 地址都不同时，会发生安全违规事件。同时，如果在一个安全端口上配置或者学习到的工作站安全 MAC 地址尝试访问另一个安全端口时，违规事件会被标记。

安全 MAC 地址类型

交换机支持以下类型的安全 MAC 地址：

- 静态安全MAC地址——使用接口配置命令**switchport port-securitymac-address** *mac-*

*address*手动配置，保存在地址表中，且被添加到交换机的运行配置中。

- 动态安全 MAC 地址——动态配置，仅保存在地址表中，且在交换机重启时会被移除。
- 粘性安全 MAC 地址——可以动态学习或手工配置，保存在地址表中，且被添加到运行配置。如果这些地址被保存在配置文件中，当交换机重启时，接口无需重新动态配置这些地址。

粘性安全 MAC 地址

可以启用粘性学习功能，配置接口把动态 MAC 地址转换为粘性安全 MAC 地址，并将其添加到运行配置中。接口会把所有动态安全 MAC 地址，包括启用粘性学习之前动态学习到的地址转换为粘性安全 MAC 地址。所有粘性安全 MAC 地址都会被添加到运行配置中。

粘性安全 MAC 地址不会自动保存到交换机重启时使用的启用配置文件中。如果把粘性安全 MAC 地址保存在配置文件中，当交换机重启时，接口无需重新学习这些地址。如果不保存粘性安全地址，它们会丢失。

如果禁用了粘性学习功能，粘性安全 MAC 地址会被转换为动态安全地址，且会被从运行配置中移除。

安全违规

以下情况之一发生时会造成安全违规：

- 已经向地址表中添加了最大数量的安全 MAC 地址，而有 MAC 地址不在地址表中的工作站尝试访问接口。
- 在一个安全接口上学习到或配置的地址在相同 VLAN 中的另一个安全接口上被发现。

可以基于违规发生时要采取的行为，将接口配置为以下三种违规模式之一：

- 保护——当安全 MAC 地址的数量达到了端口允许的最大限制时，源地址未知的数据包会被丢弃，直到移除了足量的安全 MAC 地址或者增加了允许的最大地址数量。安全违规事件发生时不会通知用户。

注释： 不建议在中继端口上配置保护违规模式。当任意 VLAN 达到了其最大限制时，即使端口没有达到最大限制，保护模式也会禁用学习功能。

- 限制——当安全 MAC 地址的数量达到了端口允许的最大限制时，源地址未知的数据包会被丢弃，直到移除了足量的安全 MAC 地址或者增加了允许的最大地址数量。在此模式中，安全违规事件发生时会通知用户。会发送 SNMP 陷阱，记录 syslog 消息，且违规计数器会增加。
- 关闭——端口安全违规事件会导致接口成为错误禁用状态或被立即关闭，且端口的 LED 灯会关闭。当安全端口在错误禁用状态时，可以输入全局配置命令 **errdisable recovery cause** *psecure-violation* 将端口移出此状态，也可以输入接口配置命令 **shutdown** 及 **no shutdown** 手动重新启用接口。这是默认的模式。
- 关闭 VLAN——基于 VLAN 设置安全违规模式。在此模式中，违规事件发生时 VLAN 会被错误禁用，而端口不会被禁用。

下表显示了端口安全配置的违规模式以及采取的行为。

表 155：安全违规模式行为

违规模式	转发流量[19]	发送 SNMP 陷阱	发送 syslog 消息	显示错误消息	增加违规计数器	关闭端口

				[20]		
保护	否	否	否	否	否	否
限制	否	是	是	否	是	否
关闭	否	否	否	否	是	是
关闭 VLAN	否	否	是	否	是	否 [21]

[19] 源地址未知的数据包会被丢弃，直到移除了足量的安全 MAC 地址。

[20] 如果手动配置了可能造成安全违规的地址，交换机会返回错误消息。

[21] 只关闭违规发生的 VLAN。

端口安全老化特性

可以使用端口安全老化特性设置端口上所有安全地址的老化时间。每个端口上支持两种类型的老化方式：

- 绝对——在特定的老化时间过后端口上的安全地址会被删除。
- 不活跃——如果在特定的老化时间内安全地址不活跃，端口上的该安全地址会被删除。

端口安全与交换机堆栈

当一台交换机加入堆栈时，新的交换机会获取配置的安全地址。新交换机会从其他堆栈成员上下载所有的动态安全地址。

当一台交换机离开堆栈时（活跃交换机或堆栈成员），其余的堆栈成员会被通知，且该交换机配置或学习到的安全 MAC 地址会被从安全 MAC 地址表中删除。

默认的端口安全配置

表 156：默认的端口安全配置

特性	**默认设置**
端口安全	在端口上禁用。
粘性地址学习	禁用。
每个端口的最大安全地址数量	1。
违规模式	关闭。超过最大安全 MAC 地址数量时端口会被关闭。
端口安全老化	禁用。老化时间是 0。 静态老化被禁用。 类型为绝对老化。

端口安全配置指南

- 只能在静态端口或中继端口上配置端口安全特性。安全端口不能是动态接入端口。
- 安全端口不能是交换端口分析器（Switched Port Analyzer，SPAN）的目的端口。

注释： 虽然允许进行配置，但语音 VLAN 支持接入端口，不支持中继端口。

- 在配置了语音 VLAN 的接口上启用端口安全时，把端口的最大安全地址数量设置为 2。当端口连接到 Inspur IP 电话时，IP 电话需要使用一个 MAC 地址。Inspur IP 电话的地址会在语音 VLAN 上学习到，不会在接入 CLAN 上学习到。如果把一台 PC 连接到 Inspur IP 电话上，无需额外的 MAC 地址。如果把多台 PC 连接到 Inspur IP 电话上，必须配置足够的安全地址数量，满足每台 PC 机一台电话使用。
- 当中继端口配置了端口安全，且为数据流量分配了一个接入 VLAN，为语音流量分配了一个语音 VLAN 时，输入 **switchport voice** 以及 **switchport priority extend** 接口配置命令没有效果。

  当连接的设备使用相同的 MAC 地址在接入 VLAN 中请求 IP 地址，又在语音 VLAN 中请求 IP 地址时，只会给接入 VLAN 分配 IP 地址。
- 输入接口的最大安全地址数量，且新输入的值大于之前的值时，新值会覆盖之前配置的值。如果接口上已配置的安全地址数量大于新值时，命令被拒绝。
- 交换机不支持对粘性安全 MAC 地址进行端口安全老化操作。

下表总结了端口安全与其他基于端口特性的兼容性。

端口类型或端口特性	与端口安全的兼容性
DTP[22] 端口 [23]	否
中继端口	是
动态接入端口 [24]	否
被路由端口	否
SPAN 源端口	是
SPAN 目的端口	否
EtherChannel	是
隧道端口	是
保护端口	是
IEEE 802.1x 端口	是
语音 VLAN 端口 [25]	是
IP 源防护	是
动态地址解析协议（ARP）监测	是
灵活链路	是

[22] DTP=动态中继协议（Dynamic Trunking Protocol）

[23] 使用接口配置命令 **switchport mode dynamic** 配置的端口。

[24] 使用接口配置命令 **switchport access vlan dynamic** 配置的 VLAN 查询协议（VLAN Query Protocol，VQP）端口。

[25] 必须把端口允许的最大安全地址数量设置为 2 加上接入 VLAN 允许的最大安全地址数量。

如何配置端口安全

启用并配置端口安全

在开始前

此设置通过限制并标识允许访问端口的工作站的 MAC 地址来限制接口的输入。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport mode {**access** | **trunk**}

5. switchport voice vlan*vlan-id*

6. switchport port-security

7. switchport port-security [maximum *value* [**vlan**{*vlan-list* | {**access** | **voice**}}]]

8. switchport port-security violation {**protect** | **restrict** | **shutdown** | **shutdown vlan**}

9. switchport port-security [mac-address *mac-address*[**vlan**{*vlan-id* | {**access** | **voice**}}]

10. switchport port-security mac-address sticky

11. switchport port-security mac-address sticky [*mac-address* | **vlan**{*vlan-id* | {**access** | **voice**}}]

12. end

13. show port-security

14. show running-config

15. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的接口，并进入接口配置模式。
步骤 4	**switchport mode** {**access**	**trunk**} **示例:** `Device(config-if)#`**switchport mode access**	设置接口的模式为接入或中继；默认模式（dynamic auto）中的接口不能配置为安全端口。
步骤 5	**switchport voice vlan***vlan-id* **示例:** `Device(config-if)#`**switchport voice vlan 22**	在端口上启用语音 VLAN。 *vlan-id*——指定用于语音流量的 VLAN。
步骤 6	**switchport port-security** **示例:** `Device(config-if)#`**switchport port-security**	在接口上启用端口安全。
步骤 7	**switchport port-security [maximum** *value* [**vlan**{*vlan-list*	{**access**	**voice**}}]] **示例:**	（可选）设置端口的最大安全 MAC 地址数量。可以在交换机或者交换机堆栈上配置的最大安全 MAC 地址数量由系统中允许的最大可用 MAC 地址数量

	Device(config-if)#**switchport port-security maximum 20**	决定。此数值由活跃的交换机数据库管理（Switch Database Management，SDM）模板决定。此数值是总的可用 MAC 地址数量，包括用于其他二层功能的地址以及接口上配置的任何其他安全 MAC 地址。 （可选）**vlan**——基于 VLAN 设置最大值。 输入 **vlan** 关键字之后输入以下选项： • *vlan-list*——在中继端口上可以基于 VLAN 设置最大值，输入连字符分隔的VLAN范围或一组由逗号分隔的 VLAN。对于为指定的 VLAN，将使用基于 VLAN 的最大值。 • **access**——在接入端口上指定 VLAN 为接入 VLAN。 • **voice**——在接入端口上指定 VLAN 为语音 VLAN。 **注释：**只有在端口上配置了语音 VLAN 且端口不在接入 VLAN 中时，**voice** 关键字才可用。如果接口配置了语音 VLAN，应配置最大 2 个安全 MAC 地址。
步骤 8	**switchport port-security violation {protect | restrict | shutdown | shutdown vlan}** **示例:** Device(config-if)#**switchport port-security violation restrict**	（可选）设置违规模式以及检测到安全违规时要采取的行为： • **protect**——当安全 MAC 地址的数量达到了端口允许的最大限制时，源地址未知的数据包会被丢弃，直到移除了足量的安全 MAC 地址或者增加了允许的最大地址数量。安全违规事件发生时不会通知用户。 **注释：** 不建议在中继端口上配置保护违规模式。当任意 VLAN 达到了其最大限制时，即使端口没有达到最大限制，保护模式也会禁用学习功能。 • **restrict**——当安全 MAC 地址的数量达到了端口允许的最大限制时，源地址未知的数据包会被丢弃，直到移除了足量的安全 MAC 地址或者增加了允许的最大地址数量。在此模式中，安全违规事件发生时会通知用户。会发送 SNMP

		陷阱，记录 syslog 消息，且违规计数器会增加。 • **shutdown**——端口安全违规事件会导致接口成为错误禁用状态，且端口的 LED 灯会关闭。会发送 SNMP 陷阱，记录 syslog 消息，且违规计数器会增加。 • **shutdown vlan**——基于 VLAN 设置安全违规模式。在此模式中，违规事件发生时 VLAN 会被错误禁用，而端口不会被禁用。 **注释：**安全端口在错误禁用状态时，可以输入全局配置命令**errdisable recovery causepsecure-violation**将端口移出此状态。可以使用接口配置命令**shutdown**及**no shutdown**手动重启端口，或使用特权EXEC命令**clear errdisableinterface vlan**。
步骤 9	**switchport port-security [mac-address** *mac-address*[**vlan** {*vlan-id* | {**access** | **voice**}}] **示例:** `Device(config-if)#`**switchport port-security mac-address 00:A0:C7:12:C9:25 vlan 3 voice**	（可选）为接口输入安全 MAC 地址。可以使用此命令输入源 MAC 地址的最大数量。如果输入的值小于最大安全 MAC 地址数，其余的 MAC 地址可以动态学习。 **注释：**如果输入此命令后启用了粘性学习，动态学习到的安全地址会被转换为粘性安全 MAC 地址并被添加到运行配置中。 （可用）**vlan**——基于 VLAN 设置最大值。 输入 **vlan** 关键字之后输入以下选项： • *vlan-list*——在中继端口上可以指定 VLAN ID 以及 MAC 地址。如果不指定 VLAN ID，将使用本征 VLAN。 • **access**——在接入端口上指定 VLAN 为接入 VLAN。 • **voice**——在接入端口上指定 VLAN 为语音 VLAN。 **注释：**只有在端口上配置了语音 VLAN 且端口不在接入 VLAN 中时，**voice** 关键字才可用。如果接口配置了语音 VLAN，应配置最大 2 个安全 MAC 地址。

步 骤 10	**switchport port-security mac-address sticky** **示例:** `Device(config-if)#`**switchport port-security mac-address sticky**	（可选）在接口上启用粘性学习。
步 骤 11	**switchport port-security mac-address sticky** [*mac-address* | **vlan**{*vlan-id* | {**access** | **voice**}}] **示例:** `Device(config-if)#`**switchport port-security mac-address sticky 00:A0:C7:12:C9:25 vlan voice**	（可选）输入粘性安全 MAC 地址，按需重复输入命令。如果输入的数量小于最大安全 MAC 地址数量，其余 MAC 地址会自动学习，并被转换为粘性安全 MAC 地址，且被添加到运行配置中。 **注释：**如果在输入此命令前未启用粘性学习，会显示错误消息，且无法输入粘性安全 MAC 地址。 （可选）**vlan**——基于 VLAN 设置最大值。 输入 **vlan** 关键字之后输入以下选项： • *vlan-list*——在中继端口上可以指定 VLAN ID 以及 MAC 地址。如果不指定 VLAN ID，将使用本征 VLAN。 • **access**——在接入端口上指定 VLAN 为接入 VLAN。 • **voice**——在接入端口上指定 VLAN 为语音 VLAN。 **注释：**只有在端口上配置了语音 VLAN 且端口不在接入 VLAN 中时，**voice** 关键字才可用。
步 骤 12	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步 骤 13	**show port-security** **示例:** `Device#` **show port-security**	验证配置的条目。
步 骤 14	**show running-config** **示例:** `Device#` **show running-config**	验证配置的条目。
步 骤 15	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

启用并配置端口安全老化

使用此特性可以在安全端口上移除并添加设备，无需手动删除现有的安全 MAC 地址，且仍

可以限制端口上安全地址的数量。可以基于端口启用或禁用安全地址的老化功能。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport port-security aging {**static** | **time** *time*| **type** {**absolute** | **inactivity**}}

5. end

6. show port-security [**interface** *interface-id*] [**address**]

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/1**	指定要配置的接口，并进入接口配置模式。
步骤 4	**switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}} **示例:** `Device(config-if)#` **switchport port-security aging time 120**	启用或禁用安全端口的静态老化，或设置老化时间及类型。 **注释：**交换机不支持粘性安全端口的端口安全老化。 在端口上输入 **static** 来启用静态配置的安全地址的老化。 *time* 字段指定端口的老化时间。合法的范围从 0 到 1440 分钟。 **type** 字段可以选择以下关键字： • **absolute**——设置老化类型为绝对老化。端口上的所有安全地址在指定的时间后会老化且被从安全地址列表中移除。 • **inactivity**——设置老化类型为不活跃老化。只有特定时间内没有来自安全源地址的数据流量，安全地址才会老化。
步骤 5	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 6	**show port-security** [**interface** *interface-id*] [**address**]	验证配置。

	示例: `Device#` **show port-security interface gigabitethernet1/0/1**	
步骤 7	**show running-config** **示例:** `Device#` **show running-config**	验证配置的条目。
步骤 8	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）把配置保存在配置文件中。

端口安全配置示例

以下示例显示了如何在端口上启用端口安全，并设置最大安全地址数量为 50。违规模式为默认，未配置静态安全 MAC 地址，且启用了粘性学习。

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)#switchport mode access
Device(config-if)#switchport port-security
Device(config-if)#switchport port-security maximum 50
Device(config-if)#switchport port-security mac-address sticky
```

以下示例显示了如何在端口的 VLAN 3 上配置静态的安全 MAC 地址：

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)#switchport mode trunk
Device(config-if)#switchport port-security
Device(config-if)#switchport port-security mac-address 0000.0200.0004 vlan 3
```

以下示例显示了如何在端口上启用粘性端口安全特性，手动为数据 VLAN 以及语音 VLAN 配置 MAC 地址，并把安全地址的最大数量设置为 20（数据 VLAN 10 个，语音 VLAN10 个）。

```
Device(config)# interface tengigabitethernet1/0/1
Device(config-if)#switchport access vlan 21
Device(config-if)#switchport mode access
Device(config-if)#switchport voice vlan 22
Device(config-if)#switchport port-security
Device(config-if)#switchport port-security maximum 20
Device(config-if)#switchport port-security violation restrict
Device(config-if)#switchport port-security mac-address sticky
Device(config-if)#switchport port-security mac-address sticky 0000.0000.0002
Device(config-if)#switchport port-security mac-address 0000.0000.0003
Device(config-if)#switchport port-security mac-address sticky 0000.0000.0001 vlan voice
Device(config-if)#switchport port-security mac-address 0000.0000.0004 vlan voice
Device(config-if)#switchport port-security maximum 10 vlan access
Device(config-if)#switchport port-security maximum 10 vlan voice
```

其他参考资料

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

配置 IPv6 第一跳安全

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

IPv6 第一跳安全的前提

- 已配置了必要的 IPv6 SDM 模板。
- 熟悉 IPv6 的邻居发现特性。

IPv6 第一跳安全的限制

- 把 FHS 策略应用到 EtherChannel 接口（端口通道）时存在以下限制：
  - 配置了 FHS 策略的物理端口不能加入 EtherChannel 组。
  - 物理端口是 EtherChannel 组成员时，不能为其配置 FHS 策略。
- 默认情况下，侦听策略有防护的安全等级。在接入层交换机上配置这样的侦听策略时，即使面向路由器或 DHCP 服务器/中继的上行链路端口被配置为可信端口，外部的 IPv6 路由器通告（Router Advertisement，RA）或 IPv6 的动态主机配置协议（Dynamic Host Configuration Protocolfor IPv6，DHCPv6）服务器数据包也会被阻塞。要允许 IPv6 RA 或 DHCPv6 服务器消息，需执行以下操作：
  - 在上行链路端口上应用 IPv6 RA 防护策略（对于 RA）或 IPv6 DHCP 防护策略（对于 DHCP 服务器消息）。
  - 配置较低安全等级的侦听策略，如收集或监测。然而，不建议为这样的安全策略配置较低的安全等级，因为这时第一跳安全特性就不再有效。

关于 IPv6 第一跳安全的信息

IPv6 第一跳安全（First Hop Security in IPv6，FHS IPv6）是一组 IPv6 的安全特性，其策略可以配置到物理接口或 VLAN 上。IPv6 软件策略数据库服务存储并访问这些策略。配置或更新策略时，策略的属性会被存储或更新到软件策略数据库中，然后再按配置进行应用。当前支持以下 IPv6 策略：

- IPv6 侦听策略——IPv6 侦听策略作为策略容器，让 FHS IPv6 中大多数特性可以使用。
- IPv6 FHS 绑定表内容——通过来自邻居发现（Neighbor Discovery，ND）协议侦听等信息源的信息，可以创建一个连接到交换机的 IPv6 邻居数据库。这个数据库（或称绑定表）会被多种 IPv6 防护特性（如 IPv6 ND 监测）用来验证链路层地址（link-layeraddress，LLA）、IPv4 或 IPv6 地址以及邻居的前缀绑定信息，以防止伪造或重定向攻击。
- IPv6 邻居发现监测——IPv6 ND 监测特性会学习并保护二层邻居表中的无状态自动配置地址绑定信息。IPv6 ND 监测会分析邻居发现消息以构建可信绑定表数据库，而不合规的 IPv6 邻居发现消息会被丢弃。如果可以验证一个 ND 消息的 IPv6 到介质访问控制（Media AccessControl，MAC）映射信息，则该消息被认为是可信的。
  此特性缓解了 ND 机制的一些固有弱点，比如可能产生对 DAD、地址解析、路由器发现以及邻居缓存的攻击。
- IPv6 路由器通告防护——IPv6 路由器通告（RA）防护特性让管理员可以阻塞或拒绝到达网络交换机平台上的不希望的流氓 RA 消息。RA 被路由器用来在链路上通告自己的存在。RA 防护特性会分析 RA 消息并过滤掉未授权路由器发送的伪造 RA。在主机模式中，

所有的路由器通告以及路由器重定向消息都不在端口上被允许。RA 防护特性会在二层设备上对比配置信息与在接收的 RA 帧中发现的信息。一旦二层设备对比配置验证了 RA 帧以及路由器重定向帧的内容，它会把 RA 转发给其单播或组播目的地址。如果 RA 帧的内容未被验证，RA 会被丢弃。

- IPv6 DHCP 防护——IPv6 DHCP 防护特性会阻塞来自未授权 DHCPv6 服务器及中继代理的应答及通告消息。IPv6 DHCP 防护特性可以防止伪造的消息被输入到绑定表，并且可以阻塞在没有显式配置为面向 DHCPv6 服务器或 DHCP 中继的端口上接收的 DHCPv6 服务器消息。要使用此特性，需配置策略并将其配置到接口或 VLAN 上。要显示 DHCP 防护数据包的调试信息，使用特权 EXEC 命令 **debug ipv6 snooping dhcp-guard**。
- IPv6 源防护——与 IPv4 源防护相似，IPv6 源防护会验证源地址或前缀信息，以避免源地址伪造。

  源防护程序会基于源或目的地址信息，控制硬件允许或拒绝流量。该特性只处理数据包流量。

  IPv6 源防护特性允许把条目存储在硬件 TCAM 表中，以防止主机发送带有非法 IPv6 源地址的数据包。

  要显示源防护数据包的调试信息，使用特权 EXEC 命令 **debug ipv6 snooping source-guard**。

  注释： IPv6 源防护及前缀防护特性仅在入方向支持，不支持在出方向执行。

  该特性有以下限制：
  - 物理端口是 EtherChannel 组成员时，不能为其配置 FHS 策略。
  - 在交换机端口上启用 IPv6 源防护时，必须在交换机端口所属的接口上启用 NDP 或 DHCP 侦听。否则，来自该端口的所有数据流量都会被阻塞。
  - IPv6 源防护策略不能配置到 VLAN 上，进在接口级别支持。
  - 不能同时使用 IPv6 源防护以及前缀防护特性。把策略配置到接口上时，策略应该“验证地址”或“验证前缀”，而不能同时验证。
  - PVLAN 以及源/前缀防护不能同时应用。
  - IPv6 源防护及前缀防护支持在 EtherChannel 上使用。

  有关 IPv6 源防护的更多信息，参见 icntnetworks.com 网站 Inspur INOS IPv6 配置指南库的“IPv6 源防护”一章。
- IPv6 前缀防护——IPv6 前缀防护特性在 IPv6 源防护特性之下工作，让设备拒绝源自非拓扑正确地址的流量。IPv6 前缀防护通常在使用 DHCP 前缀授权功能给设备授权 IPv6 前缀时使用（如家庭网关）。该特性能够发现分配给链路的地址范围，并阻塞源地址在范围外的流量。

  有关 IPv6 前缀防护的更多信息，参见 icntnetworks.com 网站 Inspur INOS IPv6 配置指南库的“IPv6 前缀防护”一章。
- IPv6 目的防护——IPv6 目的防护特性在 IPv6 邻居发现之下工作，确保设备只对链路上已知的活跃地址进行地址解析。其依赖地址收集功能把链路上活跃的目的填充到绑定表中，且在解析绑定表中未发现的地址发生之前进行阻塞。

  注释： 建议把 IPv6 目的防护特性应用在配置了 SVI 的二层 VLAN 中。

  有关 IPv6 目的防护的更多信息，参见 icntnetworks.com 网站 Inspur INOS IPv6 配置指南库的“IPv6 目的防护”一章。

关于基于 SISF 的 IPv4 及 IPv6 设备追踪的信息

基于交换机集成安全特性（Switch Integrated Security Features based，SISF-based）的 IP 设备追踪作为容器策略，支持在 IPv4 和 IPv6 中通过 IP 诊断 CLI 命令使用 FHS 提供的侦听及设备追踪特性。

所有现有的 IPv6 侦听命令都有对应的基于 SISF 的设备追踪命令，可以把配置同时应用在 IPv4 和 IPv6 地址族上。

对于设备上存在的传统 IP 设备追踪以及 IPv6 侦听配置，新 **device-tracking upgrade-cli** 的允许管理员把现有配置迁移成新的基于 SISF 的设备追踪 CLI 命令。更多信息参见*迁移 IPDT 以及 IPv6 侦听命令到基于 SISF 的设备追踪命令*。

迁移到基于 SISF 的设备追踪 CLI 时的限制

- 如果设备上没有传统的 IP 设备追踪（IPDT）或 IPv6 侦听 CLI 配置，对于未来的配置可以仅使用新的基于 SISF 的设备追踪 CLI 命令。老的 IP 设备追踪 CLI 以及 IPv6 侦听 CLI 不可用。
- 如果设备上配置了 IPv6 侦听，对于未来配置可以继续使用传统的 IPv6 侦听 CLI，也可以使用 **device-tracking upgrade-cli** 命令将其迁移到新的基于 SISF 的设备追踪 CLI。在所有传统的 IPv6 侦听命令都被转换之后，设备上只能运行新的设备追踪命令。如果不使用 **device-tracking upgrade-cli** 命令，设备上只可使用传统的 IPv6 侦听命令。
- 如果在设备上配置了 IPDT，可以继续使用传统的 IPDT 命令以及 IPv6 侦听命令。此选项限制用户使用传统模式，设备上只可以使用传统的 IPDT 以及 IPv6 侦听命令。然而，建议管理员将传统配置迁移到新的基于 SISF 的设备追踪命令。
- 要把传统的 IPDT 以及 IPv6 侦听配置迁移到新的基于 SISF 的设备追踪命令，需运行 **device-tracking upgrade-cl** 命令。在运行此命令后，设备上只可以使用新的设备追踪命令，且传统的 IPDT 或 IPv6 侦听命令都不被支持。
- 不能混用旧的 IPDT 和 IPv6 侦听 CLI 以及新的基于 SISF 的设备追踪 CLI。
- 如果在传统模式中启用了 **ip dhcp snooping vlan** 命令，当传统配置迁移到新的基于 SISF 的设备追踪配置时，一个称为 WL-DEV-TRACK-DHCP 的设备追踪策略会被自动创建，用来追踪启用了 IP 设备追踪的 IPv4 以及 IPv6 客户端。如果未启用 **ip dhcp snooping vlan**，确保在设备上启用设备追踪特性，以支持其他依赖于设备追踪的特性。

迁移 IPDT 以及 IPv6 侦听命令到基于 SISF 的设备追踪命令

建议使用 **device-tracking upgrade-cli** 命令，迁移传统的 IP 设备追踪（IPDT）以及 IPv6 侦听命令到新的设备追踪命令 CLI 命令。

配置情景及迁移结果

基于设备上现有的传统配置，**device-tracking upgrade-cli** 命令会使用不同方式升级 CLI。在迁移现有配置时，请考虑以下情景及对应的迁移信息。

只存在 IPDT 配置

如果设备只有 IP 设备追踪（IPDT）配置，运行 **device-tracking upgrade-cli** 命令会在设备内部把配置翻译成新的 SISF 策略并配置在接口上。可以之后更新此 SISF 策略。

只存在 IPv6 侦听配置

在有 IPv6 侦听配置的设备上，老的 IPv6 侦听命令可以用作以后的配置。存在以下选项：

- （推荐）使用 **device-tracking upgrade-cli** 命令把传统配置迁移到新的基于 SISF 的设备追踪命令。在所有传统命令都被转换后，在设备上只能使用新的设备追踪命令。
- 在未来配置中使用传统的 IPv6 侦听命令，不运行 **device-tracking upgrade-cli** 命令。在此选项中，设备上只可以使用传统的 IPv6 侦听命令，且不能使用新的基于 SISF 的设备追踪 CLI 命令。

一个名为 Default 的设备追踪策略会在转换过程中被创建。无法手动把此策略配置到其他接口上。

同时存在 IPDT 以及 IPv6 侦听配置

在同时存在传统 IPDT 配置以及 IPv6 侦听配置的设备上，可以使用 **device-tracking upgrade-cli** 命令把传统命令转换为新的设备追踪 CLI 命令。然而，要注意只能给接口配置一个侦听策略，且 IPv6 侦听策略的参数会覆盖 IPDT 的设置。

注释： 如果不迁移到新的基于 SISF 的命令，且继续使用传统的 IPv6 侦听或 IPDT 命令，设备上的 IPv4 设备追踪配置信息可能会在 IPv6 侦听命令的输出中显示，因为该命令作为统一的特性，会同时处理 IPv4 和 IPv6 的配置。为了避免这样的情况，建议迁移传统配置并使用新的设备追踪命令。

不存在 IPDT 或 IPv6 侦听配置

如果设备上没有传统的 IP 设备追踪或 IPv6 侦听配置，管理员在以后的配置中只能使用基于 SISF 的 **device-tracking** 命令。传统的 IPDT 命令以及 IPv6 侦听命令不可用。

IPDT、IPv6 侦听以及设备追踪 CLI 兼容性

下表显示了新的基于 SISF 的设备追踪命令以及对应的 IPDT 和 IPv6 侦听命令。

表 x

IP 设备追踪（IPDT）	**IPv6 侦听**	**基于 SISF 的设备追踪**
ip device tracking probe count	不支持	不支持
ip device tracking probe delay	**ipv6 neighbor bindingreachable-lifetime**	**device-tracking policyreachable-lifetime**
ip device tracking probe interval	**ipv6 snooping trackingretry-interval**	**device-tracking policyretry-interval**
ip device tracking probe use-svi	接受，解释为**ipdevice tracking probeauto-source override**	接受，解释为 **ipdevice tracking probeauto-source override**
ip device tracking probeauto-source fallback	不支持	不支持
ip device tracking probeauto-source override	不支持	不支持
ip device tracking tracebuffer	不支持	不支持

ip device tracking maximum	**ipv6 snooping policy** <*name*>**limit**	**device-tracking snooping policy**<*name*>**limit**
ip device tracking probe count	不支持	不支持
ip device tracking probe interval	不支持	不支持
clear ip device tracking all	不支持	不支持

如何创建基于 SISF 的设备追踪及侦听策略

在特权 EXEC 模式中，按照以下步骤配置设备追踪策略。

总步骤

1. configure terminal

2. device-tracking policy *policy-name*

3. {[**device-role** {**node** | **switch**}] | [**limit address-count** *value*] | [**no**] | [**destination-glean**{**recovery**|**log-only**[**dhcp**]] | [**data-glean**{**recovery**|**log-only**{**dhcp** | **ndp**}] | **prefix-glean**] | [**security-level** {**glean** | **guard** | **inspect**}] | [**tracking** {**disable** [**stale-lifetime** [*seconds* | **infinite**] | **enable** [**reachable-lifetime** [*seconds* | **infinite**] }] | [**trusted-port**] }

4. end

5. show device-tracking policy *policy-name*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**device-tracking policy** *policy-name* **示例:** `Device(config)#` **device-tracking policy example_policy**	进入设备追踪配置模式。
步骤 3	{[**device-role** {**node**	**switch**}]	[**limit address-count** *value*]	[**no**]	[**destination-glean**{**recovery**|**log-only**[**dhcp**]]	[**data-glean**{**recovery**|**log-only**{**dhcp**	**ndp**}]	**prefix-glean**]	[**security-level** {**glean**	**guard**	**inspect**}]	[**tracking** {**disable** [**stale-lifetime** [*seconds*	**infinite**]	**enable** [**reachable-lifetime** [*seconds*	**infinite**] }]	[**trusted-port**] } **示例:**	为 IPv4 和 IPv6 启用以下选项： • （可选）**device-role**{**node**	**switch**}——指定连接到端口的设备角色。默认是 **node**。 • （可选）**limit address-count** *value*——限制每个目标允许的地址数量。 • （可选）**no**——取消命令或将其设置为默认配置。 • （可选）**destination-glean**{**recovery**|**log-only**}[**dhcp**]}——通过数据流量源地址收集来恢复绑定表。 • （可选）**data-glean**{**recovery**| **log-**

	Device(config-device-tracking)# security-level inspect **示例:** Device(config-device-tracking)# trusted-port	**only**}[**dhcp**	**ndp**]}——使用源或数据地址收集来恢复绑定表。 • （可选）**security-level**{**glean**|**guard**|**inspect**} — — 指定特性执行的安全等级。默认是**guard**。 **glean**——从消息中收集地址，且无需认证就可以填充到绑定表。 **guard**——收集并监测消息。此外，拒绝路由器通告（RA）以及 DHCP 服务器消息。这是默认的选项。 **inspect**——收集地址，验证消息的一致性，并进行地址从属检查。 • （可选）**tracking** {**disable**	**enable**}——指定追踪选项。 • （可选）**trusted-port**——设置可信端口。禁用对适用目标的防护。通过可信端口学习到的绑定优先于通过其他端口学习到的绑定。在创建表条目发生冲突时，可信端口的条目有高优先级。
步骤 4	**end** **示例:** Device(config-device-tracking)# **exit**	推出配置模式。
步骤 5	**show device-tracking policy** *policy-name* **示例:** Device#**show device-tracking policy example_policy**	显示设备追踪策略配置。

如何将设备追踪策略配置到接口

在特权 EXEC 模式中，按照以下步骤将设备追踪策略配置到接口。

总步骤

1. configure terminal

2. interface *interface*

3. device-tracking attach-policy *policy name*

4. show device-tracking policies [**interface***interface*]

具体步骤

	命令或操作	目的

步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface***interface* **示例:** `Device(config)#` **interface gigabitethernet 1/1/4**	指定接口并进入接口配置模式。
步骤 3	**device-tracking attach-policy** *policy name* **示例:** `Device(config-if)#` **device-tracking attach-policy example_policy**	将设备追踪策略配置到接口或接口的指定 VLAN 上。
步骤 4	**show device-tracking policies** [**interface** *interface*] **示例:** `Device#(config-if)#` **do show running-config**	显示匹配特定接口类型及编号的策略。

如何将设备追踪策略配置到 VLAN

在特权 EXEC 模式中，按照以下步骤将设备追踪策略配置到 VLAN 上。

总步骤

1. configure terminal

2. vlan configuration *vlan_list*

3. device-tracking [**attach-policy** *policy_name*]

4. do show running-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**vlan configuration** *vlan_list* **示例：** `Device(config)#` **vlan configuration 333**	指定要配置设备追踪策略的 VLAN，进入 VLAN 接口配置模式。
步骤 3	**device-tracking** [**attach-policy** *policy_name*] **示例：** `Device(config-vlan-config)#`**device-tracking attach-policy example_policy**	将设备追踪策略配置到指定的 VLAN 上。
步骤 4	**do show running-config** **示例：** `Device#(config-if)#` **do show running-**	在接口配置模式中验证策略应用到了指定的 VLAN 上。

	config	

如何向绑定表中添加设备范围的条目

在特权 EXEC 模式中，按照以下步骤配置绑定表内容。

总步骤

1. configure terminal

2. [**no**] **device-trackingDefault**| [**down-lifetime** *value*] | [**logging**] | [**max entries***value*] | [**reachable-lifetime**

seconds | **retry-interval** *seconds*] | [**stale-lifetime**[*seconds*]

3. exit

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	[**no**] **device-tracking Default**| [**down-lifetime***value*]	[**logging**]	[**max entries***value*] ||[**reachable-lifetime** *seconds*	**retry-interval** *seconds*]	[**stale-lifetime**[*seconds*] **示例：** `Device(config)#` **device-tracking Default**	使用以下选项创建设备范围的默认设备追踪策略，并将绑定表中添加条目。 • **down-lifetime**——设置条目在被删除之前保持 DOWN 状态的默认最大时间。 • **logging**——对绑定表事件启动系统日志记录。 • **max-entries**——定义绑定表的最大条目数量。 • **reachable-lifetime**——定义在无需证明可达性的情况下，一个可达的条目被认为是直接或间接可达的最长时间。 • **retry-interval**——定义两次探测的间隔。 • **stale-lifetime**——定义条目在被删除之前保持 Stale 状态的最大时间。
步骤 3	**exit** **示例：** `Device(config)#` **exit**	退出全局配置模式，返回特权 EXEC 配置模式。

如何配置 IPv6 侦听策略

在特权 EXEC 模式中，按照以下步骤配置 IPv6 侦听策略。

总步骤

1. configure terminal

2. ipv6 snooping policy *policy-name*

3. {[**default**] | [**device-role** {**node** | **switch**}] | [**limit address-count** *value*] | [**no**] | [**protocol** {**dhcp** | **ndp**}] | [**security-level** {**glean** | **guard** | **inspect**}] | [**tracking** {**disable** [**stale-lifetime** [*seconds* | **infinite**] |**enable** [**reachable-lifetime** [*seconds* | **infinite**] }] | [**trusted-port**] }

4. end

5. show ipv6 snooping policy *policy-name*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**ipv6 snooping policy** *policy-name* **示例：** `Device(config)#` **ipv6 snooping policyexample_policy**	创建侦听策略并进入 IPv6 侦听策略配置模式。
步骤 3	{[**default**]	[**device-role** {**node**	**switch**}]	[**limit address-count** *value*]	[**no**]	[**protocol**{**dhcp**	**ndp**}]	[**security-level** {**glean**	**guard**| **inspect**}]	[**tracking** {**disable** [**stale-lifetime**[*seconds*	**infinite**]	**enable**[**reachable-lifetime** [*seconds*	**infinite**] }] |[**trusted-port**] } **示例：** Device(config-ipv6-snooping)# security-levelinspect **示例：** Device(config-ipv6-snooping)# trusted-port	启用数据地址收集，对比多项条件验证消息，指定消息的安全等级。 • （可选）**device-role**{**node	switch**}——指定连接到端口的设备角色。默认是 **node**。 • （可选）**limit address-count** *value*——限制每个目标允许的地址数量。 • （可选）**no**——取消命令或将其设置为默认配置。 • （可选）**protocol**{**dhcp	ndp**}——指定哪种协议应被重定向到侦听特性进行分析。默认设置是 **dhcp** 和 **ndp**。要更改默认设置，使用命令 **no protocol**。 • （可选）**security-level**{**glean**|**guard**|**inspect**}——指定特性执行的安全等级。默认等级是 **guard**。 **glean**——从消息中收集地址，且无需认证就可以填充到绑定表。 **guard**——收集并监测消息。此外，拒绝路由器通告（RA）以及 DHCP 服务器消息。这是默认的选项。 **inspect**——收集地址，验证消息的一致性，并进行地址从

		属检查。 • （可选）**tracking** {**disable** | **enable**}——覆盖默认追踪行为并指定追踪选项。 • （可选）**trusted-port**——设置可信端口。禁用对适用目标的防护。通过可信端口学习到的绑定优先于通过其他端口学习到的绑定。在创建表条目发生冲突时，可信端口的条目有高优先级。
步骤 4	**end** **示例：** `Device(config-ipv6-snooping)#` **exit**	返回特权 EXEC 模式。
步骤 5	**show ipv6 snooping policy** *policy-name* **示例：** `Device#`**show ipv6 snooping policy example_policy**	显示侦听策略配置。

接下来做什么？

配置 IPv6 侦听策略到接口或 VLAN。

如何把 IPv6 侦听策略配置到接口上

在特权 EXEC 模式中，按照以下步骤配置 IPv6 侦听策略到接口或接口上的 VLAN。

总步骤

1. configure terminal

2. interface Interface_type *stack/module/port*

3. switchport

4. ipv6 snooping [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]

5. do show running-config

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** Interface_type *stack/module/port* **示例：** `Device(config)#` **interface gigabitethernet 1/1/4**	指定接口类型及标识符，进入接口配置模式。
步骤 3	**switchport** **示例：** `Device(config-if)#` **switchport**	进入 switchport 模式。 **注释：**配置二层参数时，如果接口在三层模式，必须输入不带参数的 switchport 接口配置命令，将接口置为

		二层模式。此操作会关闭并重启接口，可能在设备上生成接口已连接的消息。把三层模式的接口置为二层模式时，接口之前的配置信息可能丢失，且接口会返回默认配置。交换机端口配置模式的命令提示符为(config-if)#。
步骤 4	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan**{*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** |**remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids*| **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] **示例：** `Device(config-if)#` **ipv6 snooping** 或 `Device(config-if)#` **ipv6 snooping attach-policy example_policy** 或 `Device(config-if)#` **ipv6 snooping vlan 111,112** 或 `Device(config-if)#` **ipv6 snooping attach-policy example_policy vlan 111,112**	将定义的IPv6侦听策略配置到接口或接口上的VLAN。要给接口配置默认策略，使用不带**attach-policy**关键字的**ipv6 snooping**命令。要给接口上的VLAN配置默认策略，使用**ipv6snooping vlan**命令。默认策略中，安全等级是**guard**，设备角色是**node**，协议是**ndp**和**dhcp**。
步骤 5	**do show running-config** **示例：** `Device#(config-if)#` **do show running-config**	在接口配置模式中验证策略应用到了指定的接口上。

如何把 IPv6 侦听策略配置到二层 EtherChannel 接口上

在特权 EXEC 模式中，按照以下步骤配置 IPv6 侦听策略到 EtherChannel 接口或 VLAN。

总步骤

1. configure terminal

2. interface range *Interface_name*

3. ipv6 snooping [**attach-policy** *policy_name* [**vlan** {*vlan_ids* \| **add** *vlan_ids* \| **except** *vlan_ids* \| **none** \|**remove** *vlan_ids* \| **all**}] \| **vlan** [{*vlan_ids* \| **add** *vlan_ids* \| **except***vlan_ids* \| **none** \| **remove** *vlan_ids* \|**all**}]

4. do show running-config interface *portchannel_interface_name*

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。

步骤 2	**interface range** *Interface_name* **示例：** `Device(config)#` **interface Po11**	指定创建 EtherChannel 时分配的端口通道接口名称。进入接口范围配置模式。 提示：输入 **do show interfaces summary** 命令快速查询接口名称以及类型。
步骤 3	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* |**add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** |**remove** *vlan_ids* | **all**}] **示例：** `Device(config-if-range)#` **ipv6 snooping attach-policyexample_policy** 或 `Device(config-if-range)#` **ipv6 snooping attach-policyexample_policy vlan 222,223,224** **或** `Device(config-if-range)#`**ipv6 snooping vlan 222,223,224**	将 IPv6 侦听策略配置到接口或接口上的 VLAN。如果不使用 **attach-policy** 选项，则配置默认策略。
步骤 4	**do show running-config interface** *portchannel_interface_name* **示例：** `Device#(config-if-range)#` **do show running-configint po11**	在当前模式中确认策略已配置在指定的接口上。

如何把 IPv6 侦听策略全局配置到 VLAN 上

在特权 EXEC 模式中，按照以下步骤把 IPv6 侦听策略配置到 VLAN 上。

总步骤

1. configure terminal

2. vlan configuration *vlan_list*

3. ipv6 snooping [**attach-policy** *policy_name*]

4. do show running-config

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**vlan configuration** *vlan_list* **示例：** `Device(config)#` **vlan configuration 333**	指定要配置 IPv6 侦听策略的 VLAN，进入 VLAN 接口配置模式。
步骤 3	**ipv6 snooping** [**attach-policy** *policy_name*] **示例：**	将 IPv6 侦听策略配置到 VLAN 上，VLAN 可以覆盖所有交换机以及堆栈接口。

	Device(config-vlan-config)#**ipv6 snooping attach-policy example_policy**	如果不使用 **attach-policy** 选项，默认策略会被配置。默认策略中，安全等级是 **guard**，设备角色是 **node**，协议是 **ndp** 和 **dhcp**。
步骤 4	**do show running-config** **示例：** Device#(config-if)# **do show running-config**	在当前模式中确认策略已配置在指定的 VLAN 上。

如何配置 IPv6 绑定表内容

在特权 EXEC 模式中，按照以下步骤配置 IPv6 绑定表内容。

总步骤

1. configure terminal

2. [**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** interface_type *stack/module/porthw_address* [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**tracking**{ [default | disable] [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**enable** [**reachable-lifetimevalue** [*seconds* |**default** | **infinite**] | [**retry-interval** {*seconds*| **default** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**]}]

3. [**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number* | **port-limit** *number* [**mac-limit***number*] | **vlan-limit** *number* [[**mac-limit** *number*] | [**port-limit** *number* [**mac-limit***number*]]]]

4. ipv6 neighbor binding logging

5. exit

6. show ipv6 neighbor binding

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式。
步骤 2	[**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface**interface_type *stack/module/port hw_address* [**reachable-lifetimevalue**[*seconds*	**default**	**infinite**]	[**tracking**{ [default	disable] [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**enable**[**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**retry-interval**{*seconds*| **default** [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]}] **示例：** Device(config)# **ipv6 neighbor binding**	向绑定表中添加静态条目。

步骤 3	[**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number* |**port-limit** *number* [**mac-limit** *number*] | **vlan-limit** *number* [[**mac-limit***number*] | [**port-limit** *number* [**mac-limit***number*]]]] **示例：** `Device(config)#` **ipv6 neighbor binding max-entries 30000**	指定允许添加到绑定表缓存的最大条目数量。
步骤 4	**ipv6 neighbor binding logging** **示例：** `Device(config)#` **ipv6 neighbor binding logging**	记录绑定表主要事件。
步骤 5	**exit** **示例：** `Device(config)#` **exit**	退出全局配置模式，返回特权 EXEC 模式。
步骤 6	**show ipv6 neighbor binding** **示例：** `Device#` **show ipv6 neighbor binding**	显示绑定表内容。

如何配置 IPv6 邻居发现监测策略

在特权 EXEC 模式中，按照以下步骤配置 IPv6 ND 监测策略。

总步骤

1. configure terminal

2. [**no**]**ipv6 nd inspection policy** *policy-name*

3. device-role {**host** | **monitor** | **router** | **switch**}

4. drop-unsecure

5. limit address-count *value*

6. sec-level minimum *value*

7. tracking {**enable** [**reachable-lifetime** {*value* | **infinite**}] | **disable** [**stale-lifetime** {*value* | **infinite**}]}

8. trusted-port

9. validate source-mac

10. no {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port** |**validate source-mac**}

11. default {**device-role** | **drop-unsecure** | **limit address-count** | **sec-level minimum** | **tracking** | **trusted-port**| **validate source-mac**}

12. do show ipv6 nd inspection policy *policy_name*

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：**	进入全局配置模式。

	Device# **configure terminal**	
步骤 2	[**no**]**ipv6 nd inspection policy** *policy-name* **示例：** Device(config)# **ipv6 nd inspection policyexample_policy**	指定 ND 监测策略名称，进入 ND 监测策略配置模式。
步骤 3	**device-role** {**host** | **monitor** | **router** | **switch**} **示例：** Device(config-nd-inspection)# **device-role switch**	指定连接到端口的设备角色。默认角色是 **host**。
步骤 4	**drop-unsecure** **示例：** Device(config-nd-inspection)# **drop-unsecure**	丢弃无选项、选项不合法或签名不合法的消息。
步骤 5	**limit address-count** *value* **示例：** Device(config-nd-inspection)# **limit address-count 1000**	输入 1 到 10000 的值。
步骤 6	**sec-level minimum** *value* **示例：** Device(config-nd-inspection)# **limit address-count 1000**	指定使用加密生成的地址（Cryptographically Generated Address，CGA）选项时的最小安全等级参数。
步骤 7	**tracking** {**enable** [**reachable-lifetime** {*value* | **infinite**}] |**disable** [**stale-lifetime** {*value* | **infinite**}]} **示例：** Device(config-nd-inspection)# **tracking disablestale-lifetime infinite**	覆盖端口上的默认追踪策略。
步骤 8	**trusted-port** **示例：** Device(config-nd-inspection)# **trusted-port**	配置端口为可信端口。
步骤 9	**validate source-mac** **示例：** Device(config-nd-inspection)# **validate source-mac**	对比源介质访问控制（MAC）地址与链路层地址。
步骤 10	**no** {**device-role** | **drop-unsecure** | **limit address-count** |**sec-level minimum** | **tracking** | **trusted-port** | **validatesource-mac**} **示例：** Device(config-nd-inspection)# **no validate source-mac**	使用命令的 **no** 形式移除当前配置的参数。
步骤	**default** {**device-role** | **drop-unsecure** |	恢复配置为默认设置。

11	**limit address-count** |**sec-level minimum** | **tracking** | **trusted-port** | **validatesource-mac**} **示例：** `Device(config-nd-inspection)#` **default limit address-count**	
步 骤 12	**do show ipv6 nd inspection policy** *policy_name* **示例：** `Device(config-nd-inspection)#` **do show ipv6 ndinspection policy example_policy**	在当前配置模式中验证配置的 ND 监测配置。

如何把 IPv6 邻居发现监测策略配置到接口上

在特权 EXEC 模式中，按照以下步骤配置 IPv6 ND 监测策略到接口或接口上的 VLAN。

总步骤

1. configure terminal

2. interface Interface_type *stack/module/port*

3. ipv6 nd inspection [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none**| **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* |**all**}]

4. do show running-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** Interface_type *stack/module/port* **示例：** `Device(config)#` **interface gigabitethernet 1/1/4**	指定接口类型及标识符，进入接口配置模式。
步骤 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids*| **add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids*	**all**}]	**vlan** [{*vlan_ids*	**add** *vlan_ids*	**except***vlan_ids*	**none** |**remove** *vlan_ids*	**all**}] **示例：** `Device(config-if)#` **ipv6 nd inspection attach-policyexample_policy** 或 `Device(config-if)#` **ipv6 nd inspection**	配置邻居发现监测策略到接口或接口上的指定 VLAN。如果不使用 **attach-policy** 选项，默认策略会被配置。

	attach-policyexample_policy vlan 222,223,224 **或** `Device(config-if)#` **ipv6 nd inspection vlan 222, 223,224**	
步骤 4	**do show running-config** **示例：** `Device#(config-if)#` **do show running-config**	在接口配置模式中验证配置到接口的策略。

如何把 IPv6 邻居发现监测策略配置到二层 EtherChannel 接口上

在特权 EXEC 模式中，按照以下步骤配置 IPv6 邻居发现监测策略到 EtherChannel 接口或 VLAN。

总步骤

1. configure terminal

2. interface range *Interface_name*

3. ipv6 nd inspection [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none**| **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* |**all**}]

4. do show running-config interface *portchannel_interface_name*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface range** *Interface_name* **示例：** `Device(config)#` **interface Po11**	指定创建 EtherChannel 时分配的端口通道接口名称。进入接口范围配置模式。 **提示：** 输入 **do show interfaces summary** 命令快速查看接口名称及类型。
步骤 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan**{*vlan_ids*	**add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove***vlan_ids*	**all**}]	**vlan** [{*vlan_ids*	**add** *vlan_ids* |**except***vlan_ids*	**none**	**remove** *vlan_ids*	**all**}] **示例：** `Device(config-if-range)#` **ipv6 nd inspectionattach-policy example_policy** 或 `Device(config-if-range)#` **ipv6 nd**	配置 ND 监测策略到接口或接口上的指定 VLAN。如果不使用 **attach-policy** 选项，默认策略会被配置。

	inspectionattach-policy example_policy vlan 222,223,224 或 `Device(config-if-range)#`**ipv6 nd inspection vlan 222,223,224**	
步骤 4	**do show running-config interface***portchannel_interface_name* **示例：** `Device#(config-if-range)#` **do show running-config int po11**	在当前配置模式中确认指定接口的策略配置。

如何把 IPv6 邻居发现监测策略全局配置到 VLAN 上

在特权 EXEC 模式中，按照以下步骤把 IPv6 ND 监测策略配置到覆盖多个接口的 VLAN 上。

总步骤

1. configure terminal

2. vlan configuration *vlan_list*

3. ipv6 nd inspection [**attach-policy** *policy_name*]

4. do show running-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**vlan configuration** *vlan_list* **示例：** `Device(config)#` **vlan configuration 334**	指定要配置 IPv6 侦听策略的 VLAN，进入 VLAN 接口配置模式。
步骤 3	**ipv6 nd inspection** [**attach-policy** *policy_name*] **示例：** `Device(config-vlan-config)#`**ipv6 ndinspection attach-policy example_policy**	把 IPv6 邻居发现策略配置到覆盖所有交换机及堆栈接口的指定 VLAN 上。如果不使用 **attach-policy** 选项，默认策略会被使用。 默认策略中，主机角色是 **host**，无 drop-unsecure 设置，禁用地址计数限制，禁用最小安全等级设置，禁用追踪，无可信端口，不验证源 MAC 地址。
步骤 4	**do show running-config** **示例：** `Device#(config-if)#` **do show running-config**	在当前配置模式中确认指定 VLAN 的策略配置。

如何配置 IPv6 路由器通告防护策略

在特权 EXEC 模式中，按照以下步骤配置 IPv6 路由器通告防护策略。

总步骤

1. configure terminal

2. [**no**]**ipv6 nd raguard policy** *policy-name*

3. [**no**]**device-role** {**host** | **monitor** | **router** | **switch**}

4. [**no**]**hop-limit** {**maximum** | **minimum**} *value*

5. [**no**]**managed-config-flag** {**off** | **on**}

6. [**no**]**match** {**ipv6 access-list** *list* | **ra prefix-list** *list*}

7. [**no**]**other-config-flag** {**on** | **off**}

8. [**no**]**router-preference maximum** {**high** | **medium** | **low**}

9. [**no**]**trusted-port**

10. default {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list**| **ra prefix-list** } | **other-config-flag** | **router-preference maximum**| **trusted-port**}

11. do show ipv6 nd raguard policy *policy_name*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	[**no**]**ipv6 nd raguard policy** *policy-name* **示例：** `Device(config)#` **ipv6 nd raguard policy example_policy**	指定 RA 防护策略名称并进入 RA 防护策略配置模式。
步骤 3	[**no**]**device-role** {**host**	**monitor**	**router** |**switch**} **示例：** `Device(config-nd-raguard)#` **device-roleswitch**	指定连接到端口的设备角色。默认角色是 **host**。
步骤 4	[**no**]**hop-limit** {**maximum**	**minimum**} *value* **示例：** `Device(config-nd-raguard)#` **hop-limit maximum 33**	最大和最小跳数限制的范围（1-255）。使用跳数限制值过滤路由器通告消息。流氓 RA 消息可能有较低的跳数限制值（等同于 IPv4 的生存时间），该消息被主机接受时，能阻止主机生成发往流氓 RA 产生者以外的流量。跳数限制未指定的 RA 消息会被阻塞。 如果不进行配置，此项过滤会被禁用。配置 **minimum** 来阻塞跳数限制值低于配置值的 RA 消息。配置 **maximum** 来阻塞跳数限制值高于配置值的 RA 消息。
步骤 5	[**no**]**managed-config-flag** {**off**	**on**}	使用管理地址配置标识

	示例： `Device(config-nd-raguard)#` **managed-config-flag on**	（ManagedAddress Configuration，也称“M”标识字段）过滤路由器通告消息。M 字段为 1 的流氓 RA 消息可能导致主机使用流氓 DHCPv6 服务器。如果不进行配置，此项过滤被禁用。 **On**——接受并转发 M 值为 1 的 RA 消息，并阻塞 M 值为 0 的消息。 **Off**——接受并转发 M 值为 0 的 RA 消息，并阻塞 M 值为 1 的消息。
步骤 6	[**no**]**match** {**ipv6 access-list** *list* | **ra prefix-list***list*} **示例：** `Device(config-nd-raguard)#` **match ipv6access-list example_list**	匹配指定的前缀列表或访问列表。
步骤 7	[**no**]**other-config-flag** {**on** | **off**} **示例：** `Device(config-nd-raguard)#`**other-config-flag on**	使用其他配置（OtherConfiguration，也称“O”标识字段）过滤路由器通告消息。O 字段为 1 的流氓 RA 消息可能导致主机使用流氓 DHCPv6 服务器。如果不进行配置，此项过滤被禁用。 **On**——接受并转发 O 值为 1 的 RA 消息，并阻塞 O 值为 0 的消息。 **Off**——接受并转发 O 值为 0 的 RA 消息，并阻塞 O 值为 1 的消息。
步骤 8	[**no**]**router-preference maximum** {**high** |**medium** | **low**} **示例：** `Device(config-nd-raguard)#`**router-preference maximum high**	使用路由器优先级标志过滤路由器通告消息。如果不进行配置，此项过滤被禁用。 **high**——接受路由器优先级设置为高、中或低的 RA 消息。 **medium**——阻塞路由器优先级设置为高的 RA 消息。 **low**——阻塞路由器优先级设置为中和高的 RA 消息。
步骤 9	[**no**]**trusted-port** **Example:** `Device(config-nd-raguard)#` **trusted-port**	端口配置为可信时，所有连接的设备都被信任，且不再对其进行消息验证。
步 骤 10	**default** {**device-role** | **hop-limit** {**maximum** |**minimum**} | **managed-config-flag** | **match** {**ipv6access-list** | **ra prefix-list** } | **other-config-flag** |**router-preference maximum**| **trusted-port**} **示例：** `Device(config-nd-raguard)#` **defaulthop-limit**	恢复命令为默认值。
步 骤	**do show ipv6 nd raguard policy**	（可选）在 RA 防护策略配置模式中显

11	*policy_name* **示例：** `Device(config-nd-raguard)#` **do show ipv6nd raguard policy example_policy**	示 ND 防护策略配置。

如何把 IPv6 路由器通告防护策略配置到接口上

在特权 EXEC 模式中，按照以下步骤配置 IPv6 路由器防护策略到接口或接口上的 VLAN。

总步骤

1. configure terminal

2. interface Interface_type *stack/module/port*

3. ipv6 nd raguard [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** |**remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* |**all**}]

4. do show running-config

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** Interface_type *stack/module/port* **示例：** `Device(config)#` **interface gigabitethernet 1/1/4**	指定接口类型及标识符，进入接口配置模式。
步骤 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* |**add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids*	**all**}]| **vlan** [{*vlan_ids*	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove***vlan_ids*	**all**}] **示例：** `Device(config-if)#` **ipv6 nd raguard attach-policyexample_policy** 或 `Device(config-if)#` **ipv6 nd raguard attach-policyexample_policy vlan 222,223,224** **或** `Device(config-if)#` **ipv6 nd raguard vlan 222, 223,224**	把 IPv6 邻居发现监测策略配置到接口或接口的指定 VLAN 上。如果不使用 **attach-policy** 选项，默认策略会被配置。
步骤 4	**do show running-config** **示例：** `Device#(config-if)#` **do show running-config**	在当前配置模式中确认策略配置到指定接口上。

如何把 IPv6 路由器通告防护策略配置到二层 EtherChannel 接口上

在特权 EXEC 模式中，按照以下步骤配置 IPv6 路由器通告防护策略到 EtherChannel 接口或 VLAN 上。

总步骤

1. configure terminal

2. interface range *Interface_name*

3. ipv6 nd raguard [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** |**remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* |**all**}]

4. do show running-config interface *portchannel_interface_name*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface range** *Interface_name* **示例：** `Device(config)#` **interface Po11**	指定创建 EtherChannel 时分配的端口通道接口名称。进入接口范围配置模式。 提示：输入 **do show interfaces summary** 命令快速查询接口名称以及类型。
步骤 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids*| **add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids* |**all**}]	**vlan** [{*vlan_ids*	**add** *vlan_ids*	**except***vlan_ids*	**none**| **remove** *vlan_ids*	**all**}] **示例：** `Device(config-if-range)#` **ipv6 nd raguardattach-policy example_policy** 或 `Device(config-if-range)#` **ipv6 nd raguardattach-policy example_policy vlan 222,223,224** **或** `Device(config-if-range)#`**ipv6 nd raguard vlan 222,223,224**	把 IPv6 邻居发现监测策略配置到接口或接口的指定 VLAN 上。如果不使用 **attach-policy** 选项，默认策略会被配置。
步骤 4	**do show running-config interface** *portchannel_interface_name* **示例：** `Device#(config-if-range)#` **do show**	在当前配置模式中确认策略配置到指定接口上。

	running-config **int po11**	

如何把 IPv6 路由器通告防护策略全局配置到 VLAN 上

在特权 EXEC 模式中，按照以下步骤把 IPv6 路由器通告防护策略配置到 VLAN 上。

总步骤

1. configure terminal

2. vlan configuration *vlan_list*

3. ipv6 dhcp guard [**attach-policy** *policy_name*]

4. do show running-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**vlan configuration** *vlan_list* **示例：** `Device(config)#` **vlan configuration 335**	指定要配置 IPv6 RA 防护策略的 VLAN，进入 VLAN 接口配置模式。
步骤 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] **示例：** `Device(config-vlan-config)#`**ipv6 nd raguardattach-policy example_policy**	把 IPv6 邻居发现监测策略配置到指定的覆盖所有交换机和堆栈接口 VLAN 上。如果不使用 **attach-policy** 选项，默认策略会被配置。
步骤 4	**do show running-config** **示例：** `Device#(config-if)#` **do show running-config**	在当前配置模式中确认策略配置到指定 VLAN 上。

如何配置 IPv6 DHCP 防护策略

在特权 EXEC 模式中，按照以下步骤配置 IPv6 DHCP（DHCPv6）防护策略。

总步骤

1. configure terminal

2. [**no**]**ipv6 dhcp guard policy** *policy-name*

3. [**no**]**device-role** {**client** | **server**}

4. [**no**] **match server access-list** *ipv6-access-list-name*

5. [**no**] **match reply prefix-list** *ipv6-prefix-list-name*

6. [**no**]**preference**{ **max** *limit* | **min** *limit* }

7. [**no**] **trusted-port**

8. default {**device-role** | **trusted-port**}

9. do show ipv6 dhcp guard policy *policy_name*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	[**no**]**ipv6 dhcp guard policy** *policy-name* **示例：** `Device(config)#` **ipv6 dhcp guard policy example_policy**	指定 DHCPv6 防护策略名称，并进入 DHCPv6 防护策略配置模式。
步骤 3	[**no**]**device-role** {**client** | **server**} **示例：** `Device(config-dhcp-guard)#` **device-role server**	（可选）过滤掉端口上来自指定角色设备以外的 DHCPv6 应答和 DHCPv6 通告。默认角色是 **client**。 • **client**——默认值，指定连接的设备是客户端。此端口上的服务器消息会被丢弃。 • **server**——指定连接的设备是一台 DHCPv6 服务器。此端口上的服务器消息被允许。
步骤 4	[**no**] **match server access-list** *ipv6-access-list-name* **示例：** `;;`假设预配置的`IPv6`访问列表如下： `Device(config)#` **ipv6 access-list my_acls** `Device(config-ipv6-acl)#` **permit hostFE80::A8BB:CCFF:FE01:F700 any** `;;`配置`DHCPv6`防护，匹配允许的访问列表。 `Device(config-dhcp-guard)#` **match serveraccess-list my_acls**	（可选）验证被通告的 DHCPv6 服务器或中继地址在授权的服务器访问列表中（访问列表中的目的地址是“any”）。如果未配置，此检查会被略过。空访问列表被当作允许地址处理。
步骤 5	[**no**] **match reply prefix-list** *ipv6-prefix-list-name* **示例：** `;;`假设预配置的`IPv6`前缀列表如下： `Device(config)#` **ipv6 prefix-list my_prefixpermit 2001:0DB8::/64 le 128** `;;` 配置`DHCPv6`防护匹配前缀 `Device(config-dhcp-guard)#` **match replyprefix-list my_prefix**	（可选）验证 DHCPv6 应答消息通告的前缀在配置的授权前缀列表中。如果未配置，此检查会被略过。空前缀列表被当作允许处理。
步骤 6	[**no**]**preference**{ **max** *limit* | **min** *limit* } **示例：** `Device(config-dhcp-guard)#` **preference max250** `Device(config-dhcp-guard)#`**preference min 150**	**device-role** 是 **server** 时，配置 **max** 和 **min** 来使用服务器优先级值过滤 DHCPv6 服务器通告。默认设置允许所有的通告。 **max** *limit*——（0 到 255）（可选）验证通告的优先级（优先级选项中）小于指定的限制。默认值是 255。如果未指定，

		该检查会被略过。 **min** *limit*——（0 到 255）（可选）验证通告的优先级（优先级选项中）大于指定的限制。默认值是 0。如果未指定，该检查会被略过。
步骤 7	**[no] trusted-port** **示例：** `Device(config-dhcp-guard)# trusted-port`	（可选）**trusted-port**——设置端口为可信模式。端口上不再执行策略。 **注释：**如果配置了可信端口，则 device-role 选项不可用。
步骤 8	**default {device-role | trusted-port}** **示例：** `Device(config-dhcp-guard)#` **defaultdevice-role**	（可选）**default**——设置命令为默认值。
步骤 9	**do show ipv6 dhcp guard policy** *policy_name* **示例：** `Device(config-dhcp-guard)# do show` **ipv6 dhcpguard policy example_policy**	（可选）在配置子模式中显示 IPv6 DHCP 防护策略配置。省略 *policy_name* 变量会显示所有 DHCPv6 策略。

DHCPv6 防护配置示例

```
enable
configure terminal
ipv6 access-list acl1
permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy pol1
device-role server
match server access-list acl1
match reply prefix-list abc
preference min 0
preference max 255
trusted-port
interface GigabitEthernet 0/2/0
switchport
ipv6 dhcp guard attach-policy pol1 vlan add 1
vlan 1
ipv6 dhcp guard attach-policy pol1
show ipv6 dhcp guard policy pol1
```

如何把 IPv6 DHCP 防护策略配置到接口或接口的 VLAN 上

在特权 EXEC 模式中，按照以下步骤配置 IPv6 DHCP 防护策略到接口或接口的 VLAN 上。

总步骤

1. configure terminal

2. interface Interface_type *stack/module/port*

3. ipv6 dhcp guard [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** |**remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* |**all**}]

4. do show running-config interface Interface_type *stack/module/port*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** Interface_type *stack/module/port* **示例：** `Device(config)#` **interface gigabitethernet 1/1/4**	指定接口类型及标识符，进入接口配置模式。
步骤 3	**ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* |**add** *vlan_ids*	**except** *vlan_ids*	**none**	**remove** *vlan_ids*	**all**}] |**vlan** [{*vlan_ids*	**add** *vlan_ids*	**except***vlan_ids*	**none**	**remove***vlan_ids*	**all**}] **示例：** `Device(config-if)#` **ipv6 dhcp guard attach-policyexample_policy** 或 `Device(config-if)#` **ipv6 dhcp guard attach-policyexample_policy vlan 222,223,224** **或** `Device(config-if)#` **ipv6 dhcp guard vlan 222, 223,224**	把 DHCP 防护策略配置到接口或接口上的指定 VLAN。如果不使用 **attach-policy** 选项，默认策略会被配置。
步骤 4	**do show running-config interface** Interface_type *stack/module/port* **示例：** `Device#(config-if)#` **do show running-config gig 1/1/4**	在当前配置模式中确认策略配置到指定接口上。

如何把 IPv6 DHCP 防护策略配置到二层 EtherChannel 接口上

在特权 EXEC 模式中，按照以下步骤配置 IPv6 的 DHCP 防护策略到 EtherChannel 接口或 VLAN 上。

总步骤

1. configure terminal

2. interface range *Interface_name*

3. ipv6 dhcp guard [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* |

none |**remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* |**all**}]

4. do show running-config interface*portchannel_interface_name*

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface range** *Interface_name* **示例：** `Device(config)#` **interface Po11**	指定创建 EtherChannel 时分配的端口通道接口名称。进入接口范围配置模式。 **提示：** 输入 **do show interfaces summary** 命令快速查看接口名称及类型。
步骤 3	**interface range** *Interface_name* **示例：** `Device(config)#` **interface Po11**	把 DHCP 防护策略配置到接口或接口上的指定 VLAN。如果不使用 **attach-policy** 选项，默认策略会被配置。
步骤 4	**do show running-config interface***portchannel_interface_name* **示例：** `Device#(config-if-range)#` **do show running-config** **int po11**	在当前配置模式中确认策略配置到指定接口上。

如何把 IPv6 DHCP 防护策略全局配置到 VLAN 上

在特权 EXEC 模式中，按照以下步骤配置 IPv6 DHCP 防护策略到覆盖多个接口的 VLAN 上。

总步骤

1. configure terminal

2. vlan configuration *vlan_list*

3. ipv6 dhcp guard [**attach-policy** *policy_name*]

4. do show running-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**vlan configuration** *vlan_list* **示例：** `Device(config)#` **vlan configuration 334**	指定要配置 IPv6 侦听策略的 VLAN，并进入 VLAN 接口配置模式。
步骤 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] **示例：**	把 IPv6 DHCP 防护策略配置到指定的覆盖所有交换机以及堆栈接口的 VLAN。如果不使用 **attach-policy** 选项，默认策

	Device(config-vlan-config)#**ipv6 dhcp guard attach-policy example_policy**	略会被配置。默认策略中，设备角色是 **client**，无可信接口。
步骤 4	**do show running-config** **示例：** Device#(config-if)# **do show running-config**	在当前配置模式中确认策略配置到指定 VLAN 上。

如何配置 IPv6 源防护

总步骤

1. enable

2. configure terminal

3. [**no**] **ipv6 source-guard policy** *policy_name*

4. [**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }]

5. end

6. show ipv6 source-guard policy *policy_name*

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** Device>**enable**	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式。
步骤 3	[**no**] **ipv6 source-guard policy** *policy_name* **示例：** Device(config)# **ipv6 source-guard policy example_policy**	指定 IPv6 源防护策略的名称，并进入 IPv6 源防护策略配置模式。
步骤 4	[**deny global-autoconf**] [**permit link-local**][**default**{. . . }] [**exit**] [**no**{. . . }] **示例：** Device(config-sisf-sourceguard)# **deny global-autoconf**	（可选）定义 IPv6 源防护策略。 • **deny global-autoconf**——拒绝来源于自动配置的全局地址的数据流量。当链路上的所有全局地址都由 DHCP 分配，且管理员希望阻塞自动配置地址的主机发送流量时，此特性很有用。 • **permit link-local**——允许来源于链路本地地址的数据流量。 **注释：**源防护策略下不支持可信选项。
步骤 5	**end** **示例：** Device(config-sisf-sourceguard)# **end**	退出 IPv6 源防护策略配置模式。

步骤 6	**show ipv6 source-guard policy** *policy_name* **示例：** `Device#` **show ipv6 source-guard policy example_policy**	显示策略配置以及应用策略的所有端口。

接下来做什么？

把 IPv6 源防护策略应用到接口。

如何把 IPv6 源防护策略配置到接口上

总步骤

1. enable

2. configure terminal

3. interface Interface_type *stack/module/port*

4. ipv6 source-guard [**attach-policy** <*policy_name*>]

5. show ipv6 source-guard policy *policy_name*

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** Interface_type *stack/module/port* **示例：** `Device(config)#` **interface gigabitethernet 1/1/4**	指定接口类型及标识符，并进入接口配置模式。
步骤 4	**ipv6 source-guard** [**attach-policy** <*policy_name*>] **示例：** `Device(config-if)#` **ipv6 source-guard attach-policy example_policy**	把 IPv6 源防护策略配置到接口上。如果不使用 **attach-policy** 选项，默认策略将被配置。
步骤 5	**show ipv6 source-guard policy** *policy_name* **示例：** `Device#(config-if)#` **show ipv6 source-guard policy example_policy**	显示策略配置及应用策略的所有接口。

如何把 IPv6 源防护策略配置到二层 EtherChannel 接口上

总步骤

1. enable

2. configure terminal

3. interface port-channel *port-channel-number*

4. ipv6 source-guard [**attach-policy** <*policy_name*>]

5. show ipv6 source-guard policy *policy_name*

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface port-channel** *port-channel-number* **示例：** `Device (config)#` **interface Po4**	指定接口类型及端口号，并进入端口通道配置模式。
步骤 4	**ipv6 source-guard** [**attach-policy** <*policy_name*>] **示例：** `Device(config-if) #` **ipv6 source-guard attach-policy example_policy**	把 IPv6 源防护策略配置到接口上。如果不使用 **attach-policy** 选项，默认策略将被配置。
步骤 5	**show ipv6 source-guard policy** *policy_name* **示例：** `Device(config-if) #`**show ipv6 source-guard policy example_policy**	显示策略配置及应用策略的所有接口。

如何配置 IPv6 前缀防护

注释： 在应用前缀防护特性时，为了让路由协议能控制源自链路本地地址的数据包，应在源防护策略配置模式中启用 permit link-local 命令。

总步骤

1. enable

2. configure terminal

3. [**no**] **ipv6 source-guard policy** *source-guard-policy*

4. [**no**] **validate address**

5. validate prefix

6. exit

7. show ipv6 source-guard policy [*source-guard-policy*]

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	[**no**] **ipv6 source-guard policy** *source-guard-policy* **示例：** `Device (config)#` **ipv6 source-guard policy my_snooping_policy**	定义 IPv6 源防护策略名称，并进入交换机集成安全特性源防护策略配置模式。
步骤 4	[**no**] **validate address** **示例：** `Device (config-sisf-sourceguard)#` **no validateaddress**	禁用验证地址特性，让 IPv6 前缀防护特性可以配置。
步骤 5	**validate prefix** **示例：** `Device (config-sisf-sourceguard)#` **validate prefix**	启用 IPv6 源防护特性，执行 IPv6 前缀防护操作。
步骤 6	**exit** **示例：** `Device (config-sisf-sourceguard)#` **exit**	退出交换机集成安全特性源防护策略配置模式，返回特权 EXEC 模式。
步骤 7	**show ipv6 source-guard policy** [*source-guard-policy*] **示例：** `Device #` **show ipv6 source-guard policy policy1**	显示 IPv6 源防护策略配置。

如何把 IPv6 前缀防护策略配置到接口上

总步骤

1. enable

2. configure terminal

3. interface Interface_type *stack/module/port*

4. ipv6 source-guard attach-policy *policy_name*

5. show ipv6 source-guard policy *policy_name*

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** Interface_type *stack/module/port* **示例：** `Device(config)#` **interface gigabitethernet 1/1/4**	指定接口类型及标识符，并进入接口配置模式。
步骤 4	**ipv6 source-guard attach-policy** *policy_name* **示例：** `Device(config-if)#` **ipv6 source-guard attach-policyexample_policy**	把 IPv6 源防护策略配置到接口上。如果不使用 **attach-policy** 选项，默认策略将被配置。
步骤 5	**show ipv6 source-guard policy** *policy_name* **示例：** `Device(config-if)#` **show ipv6 source-guard policyexample_policy**	显示策略配置及应用策略的所有接口。

如何把 IPv6 前缀防护策略配置到二层 EtherChannel 接口上

总步骤

1. enable

2. configure terminal

3. interface port-channel *port-channel-number*

4. ipv6 source-guard [**attach-policy** <*policy_name*>]

5. show ipv6 source-guard policy *policy_name*

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface port-channel** *port-channel-number* **示例：** `Device (config)#` **interface Po4**	指定接口类型及端口号，并进入端口通道配置模式。
步骤 4	**ipv6 source-guard** [**attach-policy**	把 IPv6 源防护策略配置到接口上。如

	<policy_name>] **示例：** `Device(config-if)#` **ipv6 source-guardattach-policy example_policy**	果不使用 **attach-policy** 选项，默认策略将被配置。
步骤 5	**show ipv6 source-guard policy** *policy_name* **示例：** `Device(config-if)#` **show ipv6 source-guard policyexample_policy**	显示策略配置及应用策略的所有接口。

IPv6 第一跳安全配置示例

示例：如何把 IPv6 源防护策略配置到二层 EtherChannel 接口上

以下示例展示了如何把 IPv6 源防护策略配置到二层 EtherChannel 接口上。

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch(config-sisf-sourceguard) # validate address
switch(config-sisf-sourceguard)# exit
Switch(config)# interface Po4
Switch(config)# ipv6 snooping
Switch(config-if)# ipv6 source-guard attach-policy POL
Switch(config-if)# exit
switch(config)#
```

示例：如何把 IPv6 前缀防护策略配置到二层 EtherChannel 接口上

以下示例展示了如何把 IPv6 前缀防护策略配置到二层 EtherChannel 接口上。

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch (config-sisf-sourceguard)# no validate address
Switch((config-sisf-sourceguard)# validate prefix
Switch(config)# interface Po4
Switch(config-if)# ipv6 snooping
Switch(config-if)# ipv6 source-guard attach-policy POL
```

其他参考资料

相关文档

相关主题	文档标题
部署 IPv6 编址以及基本的连通性	http://www.icntnetworks.com
IPv6 网络管理以及安全主题	IPv6 配置库，Inspur INOS（Inspur 6850 交换机） http://www.icntnetworks.com
IPv6 命令参考手册	IPv6 命令参考手册，Inspur INOS（Inspur 6850 交换机） http://www.icntnetworks.com

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

配置 InspurTrustSec

关于 InspurTrustSec 的信息

InspurTrustSec 对网络中的用户、主机以及网络设备有强大的识别能力，能够提升 Inspur 网络设备的安全性。TrustSec 能够唯一地区分特定角色的数据流量，进行拓扑无关且可扩展的访问控制。该特性能够为被认证的对端建立信任关系，并加密对端之间的链路，进而确保数据保密性及完整性。

InspurTrustSec 的关键组件是 Inspur 身份服务引擎（Inspur Identity Services Engine，ISE）。可以使用 Inspur ISE 的 TrustSec 身份及安全组 ACL（Security Group ACL，SGACL）规划交换机策略，也可以手动进行交换机配置。

查询特性信息

要在交换机上配置 InspurTrustSec，请在以下 URL 查阅“InspurTrustSec 交换机配置指南”：

http://www.icntnetworks.com

InspurTrustSec 通用可用性版的版本注释 URL 如下：

http://www.icntnetworks.com

有关在 Inspur 6850 以及 6650 上的限制，请通过以下 URL 查看注释：

http://www.icntnetworks.com

有关 InspurTrustSec 方案的概览、数据表、平台特性矩阵以及示例学习等其他信息，请查看以下 URL：

http://www.icntnetworks.com

InspurTrustSec 特性

下表列出了最终会在启用 TrustSec 的 Inspur 交换机上实现的 TrustSec 特性。以后的 TrustSec 通用可用性版本会增加支持的交换机数量，并扩展每种交换机支持的特性数量。

InspurTrustSec 特性	**描述**
802.1AE 标记(MACsec)	基于 IEEE 802.1AE 的线速率逐跳二层加密协议。 在支持 MACsec 的设备之间，数据包在传输设备的出方向进行加密，在接收设备的入方向进行解密，在设备中的形式是明文。 此特性仅在硬件支持 TrustSec 的设备之间可用。 **注释：**此特性不支持在 Inspur 6850 以及 Inspur 6650 的 Inspur INOS 上使用。 **注释：**此特性不支持 Inspur 5960x。
终端准入控制（Endpoint Admission Control，EAC）	EAC 是对连接到 TrustSec 域的终端用户或设备的认证过程。EAC 通常在接入层交换机上进行。如果 EAC 认证及授权过程成功，会给

	用户或设备分配安全组标签。当前的 EAC 方式可以是 802.1x、MAC 旁路认证（MAB）以及 Web 认证代理（WebAuth）。
网络设备准入控制（Network Device Admission Control，NDAC）	TrustSec 域中的每台网络设备都可以使用 NDAC 验证对端设备的凭据以及可信度。NDAC 使用 IEEE 802.1x 基于端口认证的认证框架，并使用 EAP-FAST 作为 EAP 方式。NDAC 认证及授权过程成功后，安全关联协议会协商 IEEE 802.1AE 的加密方式。 **注释：**此特性不支持 Inspur 2960x。
安全组访问控制列表（Security Group Access Control List，SGACL）	安全组访问控制列表（SGACL）对安全组标签以及策略进行关联。对 TrustSec 域出方向有 SGT 标签的流量执行策略。
InspurTrustSec SGACL 高可用性	在支持 InspurStackWise 技术的交换机上，InspurTrustSec 安全组访问控制列表（SGACL）支持高可用性功能。InspurStackWise 技术提供了状态化的冗余性，允许交换机堆栈执行并处理访问控制条目。 启用此功能没有特定的 InspurTrustSec 配置。此特性仅支持 Inspur 6850 以及 6650 系列交换机。
安全关联协议（Security Association Protocol，SAP）	在 NDAC 认证之后，安全关联协议（SAP）会自动协商密钥及加密套件，为后续 TrustSec 对端之间的 MACsec 链路加密使用。SAP 在 IEEE 802.11i 中定义。 **注释：**此特性不支持在 Inspur 6850 以及 Inspur 6650 的 Inspur INOS 上使用。 **注释：**此特性不支持 Inspur 5960x。
安全组标签（Security Group Tag，SGT）	SGT 是一个 16 位的标签，表示 TrustSec 域中源的安全等级。该标签会被附加到以太网帧或 IP 数据包之后。
SGT 交换协议（SGT Exchange Protocol，SXP）	使用 SXP 时，硬件上不支持 TrustSec 的设备可以接收 Inspur 身份服务引擎（Inspur Identity ServicesEngine，ISE）或 Inspur 安全访问控制系统（Inspur Secure Access ControlSystem，ACS）发给被认证用户或设备的 SGT 属性。该设备随后可以给硬件支持 TrustSec 的设备发送源 IP 到 SGT 的绑定信息，支持的设备可以由此标记源流量，以执行 SGACL 策略。

当链路两端都支持 802.1AE MACsec 时，SAP 协商过程会发生。请求者与认证者之间会进行 EAPOL 密钥交换，以协商加密套件，交换安全参数并管理密钥。这些任务成功完成后，安全关联（SA）会被建立。

根据软件版本、授权以及链路硬件支持的不同，SAP 协商可以使用以下操作模式之一：

- 伽罗瓦计数器模式（Galois Counter Mode，GCM）——认证及加密
- GCM 认证（GCMA）——GCM 认证，无加密
- 无封装——无封装（明文）
- 空——封装，无认证或加密

InspurTrustSec 的特性信息

表 160：Inspur TrustSec 的特性信息

特性名称	版本	特性信息
• NDAC • SXPv1、SXPv2 • SGT • 二层执行 SGACL • 接口到 SGT 映射以及 VLAN 到 SGT 映射 • 子网到 SGT 映射 • 三层端口映射（Port Mapping，PM） • 三层身份端口映射（Identity PortMapping，IPM） • 安全组名称下载 • SXP 环路检测 • 基于策略的 CoA	Inspur INOS 12.2	这些特性在 Inspur 6850 以及 6650 交换机上引入。
SXPv1 以及 SXPv2	Inspur INOS 12.2	SXR 在 Inspur 2960-X 交换机上引入。
SXPv1 以及 SXPv2	Inspur INOS 12.2	SXR 在 Inspur 2960-XR 交换机上引入。

配置控制层限速

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和

特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

CoPP 的限制

控制层限速（control plane policing，CoPP）的限制包含如下几点：
- 仅支持入向 CoPP。**system-cpp-policy** 策略映射仅在控制层接口的入向可用。
- 仅可以把 **system-cpp-policy** 策略映射安装在控制层接口上。
- **system-cpp-policy** 策略映射以及 17 个系统定义的类不能被修改或删除。
- **system-cpp-policy** 策略映射下仅允许有 **police** 行为。而且，**police rate** 仅能按照数据包每秒（packets per second，pps）来配置。
- 每个类映射有一个或多个 CPU 队列。对于多个 CPU 队列属于一个类映射的情况，更改一个类映射的限速器速率会影响术语该类映射的所有 CPU 队列。相似的，禁用一个类映射会禁用所有属于该类映射的队列。

关于控制层限速的信息

本章描述了控制层限速（CoPP）如何在设备上工作，以及如何对其进行配置。

CoPP 概述

CoPP 特性通过优先处理控制层及管理流量，保护 CPU 不受不必要的流量或 DoS 流量的影响，进而提升设备的安全性。
设备通常被划分为三个操作层，每层目标不同：
- 数据层，转发数据包。
- 控制层，正确路由数据。
- 管理层，管理网元。

可以使用 CoPP 来保护多数 CPU 处理的流量，以确保路由的稳定性、可达性，保证数据包正常送达。更重要的是，可以使用 CoPP 来保护 CPU 免受 DoS 攻击。
CoPP 使用模块化的 QoS 命令行界面（MQC）以及 CPU 队列来实现这些目标。不同类型的控制层流量会被基于特定的条件分为一组，并分配给一个 CPU 队列。可以通过配置专用的硬件限速器来管理这些 CPU 队列。例如，可以修改特定 CPU 队列（流量类型）的限速器速率，也可以禁用特定类型流量的限速器。
虽然限速器在硬件上配置，但 CoPP 不会影响 CPU 性能或数据层性能。然而，因为其限制了进入 CPU 的数据包数量，所以 CPU 的负载被控制了。这意味着等待来自硬件的数据包的服务能处理的入向数据包速率会更受控制（该速率可以由用户配置）。

CoPP 的系统定义功能

第一次启动设备时，系统会自动执行以下操作：

- 查找策略映射 **system-cpp-policy**。如果未检测到此策略映射，系统会创建并将其安装到控制层。
- 系统在 **system-cpp-policy** 之下创建 17 个类映射。
  下一次启动设备时，系统会检测到已经创建了策略及类映射。
- 安装策略后，默认会启用（32 个队列中的）16 个 CPU 队列，各使用自己的默认速率。
  默认启用的 CPU 队列及其默认速率在表 161：CoPP 的系统定义值中列出。

下表列出了启动设备时系统创建的类映射。表中列出了每个类映射对应的限速器以及分组在每个类映射下的一个或多个 CPU 队列。类映射与限速器之间是一对一映射；类映射与 CPU 队列之间是一对多映射。

表 161：CoPP 的系统定义值

类映射名称	限速器索引（限速器编号）	CPU 队列（队列编号）	是否默认启用 CPU 队列？	默认限速器速率（数据包每秒，pps）
system-cpp-police-data	WK_CPP_POLICE_DA TA(0)	WK_CPU_Q_ICMP_GEN(3) WK_CPU_Q_BROADCAST(12)	是	200
system-cpp-police-l2-control	WK_CPP_POLICE_L2_CONTROL(1)	WK_CPU_Q_L2_CONTROL(1)	否	500
system-cpp-police-routing-control	WK_CPP_POLICE_ROUTING_CONTROL(2)	WK_CPU_Q_ROUTING_CONTROL(4)	是	500
system-cpp-police-control-low-	WK_CPP_POLICE_CONTROL_LOW_PRI(3)	WK_CPU_Q_ICMP_REDIRECT(6) WK_CPU_Q_GENERAL_PUNT(25)	否	500

priority				
system-cpp-police-punt-webauth	WK_CPP_POLICE_PUNT_WEBAUTH(7)	WK_CPU_Q_PUNT_WEBAUTH(22)	否	1000
system-cpp-police-topology-control	WK_CPP_POLICE_TOPOLOGY_CONTROL(8)	WK_CPU_Q_TOPOLOGY_CONTROL(15)	否	13000
system-cpp-police-multicast	WK_CPP_POLICE_MULTICAST(9)	WK_CPU_Q_TRANSIT_TRAFFIC(18) WK_CPU_Q_MCAST_DATA(30)	是	500
system-cpp-police-sys-data	WK_CPP_POLICE_SYS _DATA(10)	WK_CPU_Q_LEARNING_CACHE_OVFL(13) WK_CPU_Q_CRYPTO_CONTROL(23) WK_CPU_Q_EXCEPTION(24) WK_CPU_Q_EGR_EXCEPTION(28) WK_CPU_Q_NFL_SAMPLED_DATA(26) WK_CPU_Q_GOLD_PKT(31) WK_CPU_Q_RPF_FAILED(19)	是	100
system-cpp-police-dot1x-auth	WK_CPP_POLICE_DOT1X(11)	WK_CPU_Q_DOT1X_AUTH(0)	否	1000
system-cpp-police-protocol-snooping	WK_CPP_POLICE_PR	WK_CPU_Q_PROTO_SNOOPING(16)	否	500
system-cpp-police-sw-	WK_CPP_POLICE_SW_FWD(13)	WK_CPU_Q_SW_FORWARDING_Q(14) WK_CPU_Q_SGT_CACHE_FULL(27)	是	1000

forward		WK_CPU_Q_LOGGING(21)		
system-cpp-police-forus	WK_CPP_POLICE_FORUS(14)	WK_CPU_Q_FORUS_ADDR_RESOLUTION(5) WK_CPU_Q_FORUS_TRAFFIC(2)	否	1000
system-cpp-police-multicast-end-station	WK_CPP_POLICE_MULTICAST_SNOOPING(15)	WK_CPU_Q_MCAST_END_STATION_SERVICE(20)	是	2000
system-cpp-default	WK_CPP_POLICE_DEFAULT_POLICER	WK_CPU_Q_DHCP_SNOOPING WK_CPU_Q_SHOW_FORWARD	否	1000

CoPP 的用户可配置功能

可以执行以下操作来管理控制层流量：

- 启用或禁用 CPU 队列。
  要启用 CPU 队列，需配置策略映射 **system-cpp-policy** 之下对应类映射的限速器行为（数据包每秒）。
  要禁用 CPU 队列，需移除策略映射 **system-cpp-policy** 之下对应类映射的限速器行为（数据包每秒）。
- 要更改限速器速率，需配置策略映射 **system-cpp-policy** 之下对应类映射的限速器行为（数据包每秒）。
- 要设置 CPU 队列为默认值，需在全局配置模式中输入 **cpp system-default** 命令。

如何配置 CoPP

启用 CPU 队列或更改限速器速率

启用 CPU 队列与更改 CPU 队列的限速器速率的过程相同，按如下步骤进行。

总步骤

1. enable

2. configure terminal

3. policy-map *policy-map-name*

4. class *class-name*

5. police rate *rate* **pps**

6. end

7. show running-config | begin system-cpp-policy

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**policy-map** *policy-map-name* **示例:** `Device(config)#` **policy-map system-cpp-policy** `Device(config-pmap)#`	进入策略映射配置模式。
步骤 4	**class** *class-name* **示例:** `Device(config-pmap)#` **class system-cpp-police-protocol-snooping** `Device(config-pmap-c)#`	进入类行为配置模式。输入希望启用的 CPU 队列对应的类名。参见表 161：CoPP 的系统定义值。
步骤 5	**police rate** *rate* **pps** **示例:** `Device(config-pmap-c)#` **police rate 100 pps**	指定特定类型流量每秒处理的入向数据包数量上限。 **注释：**指定的速率会应用到属于该类映射的所有 CPU 队列上。
步骤 6	**end** **示例:** `Device(config-pmap-c)#` **end**	返回特权 EXEC 模式。
步骤 7	**show running-config | begin system-cpp-policy** **示例:** `Device#` **show running-config | begin system-cpp-policy**	显示为不同流量类型配置的速率。

禁用 CPU 队列

按照以下步骤禁用 CPU 队列。

总步骤

1. enable

2. configure terminal

3. policy-map *policy-map-name*

4. class *class-name*

5. no police rate *rate* **pps**

6. end

7. show running-config | begin system-cpp-policy

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**policy-map** *policy-map-name* **示例:** `Device(config)#` **policy-map system-cpp-policy** `Device(config-pmap)#`	进入策略映射配置模式。
步骤 4	**class** *class-name* **示例:** `Device(config-pmap)#` **class system-cpp-police-protocol-snooping** `Device(config-pmap-c)#`	进入类行为配置模式。输入希望启用的 CPU 队列对应的类名。参见表 161：CoPP 的系统定义值。
步骤 5	**no police rate** *rate* **pps** **示例:** `Device(config-pmap-c)#` **no police rate 100 pps**	禁用对特定类型流量的入向数据包处理。 **注释：**此操作会禁用属于指定类映射的所有 CPU 队列。
步骤 6	**end** **示例:** `Device(config-pmap-c)#` **end**	返回特权 EXEC 模式。
步骤 7	**show running-config | begin system-cpp-policy** **示例:** `Device#` **show running-config | begin system-cpp-policy**	显示为不同流量类型配置的速率。

为所有 CPU 队列配置默认限速器速率

按照以下步骤把所有 CPU 队列的限速器速率设置为默认值。

总步骤

1. enable

2. configure terminal

3. cpp system-default

4. end

5. show platform hardware fed switch *switch-number* **qos que stat internal cpu policer**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：**	进入特权 EXEC 模式。在提示时输入密码。

	Device>**enable**	
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式。
步骤 3	**cpp system-default** **示例:** Device(config)# **cpp system-default** Defaulting CPP : Policer rate for all classes willbe set to their defaults	把所有类的限速器速率设置为默认速率。
步骤 4	**end** **示例:** Device(config-pmap-c)# **end**	返回特权 EXEC 模式。
步骤 5	**show platform hardware fed switch** *switch-number* **qos questat internal cpu policer** **示例:** Device# **show platform hardware fed switch 1 qos questat internal cpu policer**	显示为不同流量类型配置的速率。

CoPP 配置示例

示例：启用 CPU 队列或更改 CPU 队列的限速器速率

此示例展示了如何启用 CPU 队列或更改 CPU 队列的限速器速率。示例中 CPU 队列 system-cpp-police-protocol-snooping 的限速器速率被设置为 100pps。

```
Device>enable
Device# configure terminal
Device(config)# policy-map system-cpp-policy
Device(config-pmap)# class system-cpp-police-protocol-snooping
Device(config-pmap-c)# police rate 100 pps
Device(config-pmap-c)# end
Device# show running-config | begin system-cpp-policy
policy-map system-cpp-policy
class system-cpp-police-data
police rate 200 pps
class system-cpp-police-sys-data
police rate 100 pps
class system-cpp-police-sw-forward
police rate 1000 pps
class system-cpp-police-multicast
police rate 500 pps
class system-cpp-police-multicast-end-station
```

```
police rate 2000 pps
class system-cpp-police-punt-webauth
class system-cpp-police-l2-control
class system-cpp-police-routing-control
police rate 500 pps
class system-cpp-police-control-low-priority
class system-cpp-police-topology-control
class system-cpp-police-dot1x-auth
class system-cpp-police-protocol-snooping
police rate 100 pps
class system-cpp-police-forus
class system-cpp-default
<输出已删节>
```

示例：禁用 CPU 队列

此示例展示了如何禁用 CPU 队列。示例中 CPU 队列 system-cpp-police-protocol-snooping 被禁用。

```
Device>enable
Device# configure terminal
Device(config)# policy-map system-cpp-policy
Device(config-pmap)# class system-cpp-police-protocol-snooping
Device(config-pmap-c)# no police rate 100 pps
Device(config-pmap-c)# end
Device# show running-config | begin system-cpp-policy
policy-map system-cpp-policy
class system-cpp-police-data
police rate 200 pps
class system-cpp-police-sys-data
police rate 100 pps
class system-cpp-police-sw-forward
police rate 1000 pps
class system-cpp-police-multicast
police rate 500 pps
class system-cpp-police-multicast-end-station
police rate 2000 pps
class system-cpp-police-punt-webauth
class system-cpp-police-l2-control
class system-cpp-police-routing-control
police rate 500 pps
class system-cpp-police-control-low-priority
class system-cpp-police-topology-control
class system-cpp-police-dot1x-auth
class system-cpp-police-protocol-snooping
```

```
class system-cpp-police-forus
class system-cpp-default
<输出已删节>
```

示例：为所有 CPU 队列配置默认限速器速率

此示例展示了如何为所有 CPU 队列配置默认限速器速率并验证设置。

```
Device>enable
Device# configure terminal
Device(config)# cpp system-default
Defaulting CPP : Policer rate for all classes will be set to their defaults
Device(config)# end
Device# show platform hardware fed switch 1 qos queue stats internal cpu policer
                                          (default) (set)
QId PlcIdx   Queue Name                Enabled      Rate     Rate     Drop
------------------------------------------------------------------------------
0   11       DOT1X Auth                   No       1000     1000     0
1   1        L2 Control                   No       500      500      0
2   14       Forus traffic                No       1000     1000     0
3   0        ICMP GEN                     Yes      200      200      0
4   2        Routing Control              Yes      1800     1800     0
5   14       Forus Address resolution     No       1000     1000     0
6   3        ICMP Redirect                No       500      500      0
7   6        WLESS PRI-5                  No       1000     1000     0
8   4        WLESS PRI-1                  No       1000     1000     0
9   5        WLESS PRI-2                  No       1000     1000     0
10  6        WLESS PRI-3                  No       1000     1000     0
11  6        WLESS PRI-4                  No       1000     1000     0
12  0        BROADCAST                    Yes      200      200      0
13  10       Learning cache ovfl          Yes      100      100      0
14  13       Sw forwarding                Yes      1000     1000     0
15  8        Topology Control             No       13000    13000    0
16  12       Proto Snooping               No       500      500      0
17  16       DHCP Snooping                No       1000     1000     0
18  9        Transit Traffic              Yes      500      500      0
19  10       RPF Failed                   Yes      100      100      0
20  15       MCAST END STATION            Yes      2000     2000     0
21  13       LOGGING                      Yes      1000     1000     0
22  7        Punt Webauth                 No       1000     1000     0
23  10       Crypto Control               Yes      100      100      0
24  10       Exception                    Yes      100      100      0
25  3        General Punt                 No       500      500      0
26  10       NFL SAMPLED DATA             Yes      100      100      0
27  2        SGT Cache Full               Yes      1800     1800     0
```

```
28   10       EGR Exception                Yes      100      100      0
29   16       Show frwd                    No       1000     1000     0
30   9        MCAST Data                   Yes      500      500      0
31   10        Gold Pkt                    Yes      100      100      0
```

监控 CoPP

按照以下步骤显示限速器设置，如流量类型以及 CPU 队列的限速器速率（用户配置及默认速率）。

总步骤

1. enable

2. show platform hardware fed switch *switch-number* **qos que stat internal cpu policer**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**show platform hardware fed switch** *switch-number***qos que stat internal cpu policer**	显示为不同流量类型配置的速率。

```
Device>enable
Device# show platform hardware fed switch 3 qos queue stats internal cpu policer
(default) (set)
QId PlcIdx Queue Name Enabled Rate Rate Drop
------------------------------------------------------------------------
0 11 DOT1X Auth No 1000 1000 0
1 1 L2 Control No 500 500 0
2 14 Forus traffic No 1000 1000 0
3 0 ICMP GEN Yes 200 200 0
4 2 Routing Control Yes 1800 1800 0
5 14 Forus Address resolution No 1000 1000 0
6 3 ICMP Redirect No 500 500 0
7 6 WLESS PRI-5 No 1000 1000 0
8 4 WLESS PRI-1 No 1000 1000 0
9 5 WLESS PRI-2 No 1000 1000 0
10 6 WLESS PRI-3 No 1000 1000 0
11 6 WLESS PRI-4 No 1000 1000 0
12 0 BROADCAST Yes 200 200 0
13 10 Learning cache ovfl Yes 100 100 0
14 13 Sw forwarding Yes 1000 1000 0
15 8 Topology Control No 13000 13000 0
16 12 Proto Snooping No 500 500 0
17 16 DHCP Snooping No 1000 1000 0
```

```
18 9 Transit Traffic Yes 500 500 0
19 10 RPF Failed Yes 100 100 0
20 15 MCAST END STATION Yes 2000 2000 0
21 13 LOGGING Yes 1000 1000 0
22 7 Punt Webauth No 1000 1000 0
23 10 Crypto Control Yes 100 100 0
24 10 Exception Yes 100 100 0
25 3 General Punt No 500 500 0
26 10 NFL SAMPLED DATA Yes 100 100 0
27 2 SGT Cache Full Yes 1800 1800 0
28 10 EGR Exception Yes 100 100 0
29 16 Show frwd No 1000 1000 0
30 9 MCAST Data Yes 500 500 0
31 10 Gold Pkt Yes 100 100 0
```

其他参考资料

相关文档

相关主题	**文档标题**
MQC QoS 命令及 CoPP**show** 命令	统一平台命令参考手册，Inspur INOS（Inspur 6650 交换机）

错误信息解释

描述	**链接**
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

标准以及 RFC

标准/RFC	**标题**
无	-

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

CoPP 的特性历史与信息

下表提供了本模块描述特性的版本信息。表中仅列出了引入特定特性支持的软件版本。除非另有说明，否则后续软件版本同样支持该特性。

特性名称	版本	特性信息
控制层限速（CoPP）或 CPP	Inspur INOS 12.2	此特性被引入。
CoPP 的 CLI 配置		此特性可由用户配置。可以使用 CLI 配置选项来启用或禁用 CPU 队列，更改限速器速率，并把限速器速率设置为默认值。

配置服务等级协定

本章描述了如何在交换机上使用 Inspur INOS IP 服务等级协定（SLA）。除非另行说明，否则术语*交换机*在这里表示单台交换机或交换机堆栈。

查询特性信息

用户的软件版本可能无法支持这部分文档所提到的全部特性。想要查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

SLA 的限制条件

这部分列出了 SLA 的限制条件。

IP SLA 网络性能测量的限制条件如下所示：

- 设备不支持使用网守注册延迟探针评估的 VoIP 服务等级；
- 只有 Inspur INOS 设备可以作为目的 IP SLA 响应方的源；
- 用户不能在非 Inspur 设备上配置 IP SLA 响应方，Inspur INOS IP SLA 只能向那些设备的原生服务发送探针数据包。

SLA 的相关信息

Inspur INOS IP 服务等级协定（SLA）

Inspur INOS IP SLA 会向网络中发送数据，以此来评估多个网络站点或多条网络路径的性能。它会模拟网络数据和 IP 服务，并实时收集网络性能信息。Inspur INOS IP SLA 可以在 Inspur INOS 设备之间生成并分析流量，也可以从 Inspur INOS 设备向远端 IP 设备（比如网络应用服务器）发送并分析流量。用户通过使用各种 Inspur INOS IP SLA 探针提供的评估结果，可以进行排错、问题分析，以及设计网络拓扑。

根据具体的 Inspur INOS IP SLA 探针，Inspur 设备中的各种网络性能状态统计信息都可以实现监控，并且这些状态统计信息会同时保存在命令行界面（CLI）和简单网络管理协议（SNMP）MIB 中。IP SLA 数据包拥有可配置的 IP 地址和应用层选项，比如源和目的 IP 地址、用户数据报协议（UDP）/TCP 端口号、服务类型（ToS）字节（其中包括差分服务代码点[DSCP]和 IP 前缀比特）、虚拟专用网（VPN）路由/转发实例（VRF），以及 URL 网页地址。

由于 Inspur IP SLA 与二层传输无关，因此用户可以在相互分离的网络上配置端到端探针，以便更好地反映出度量结果，也就是终端用户最有可能经历的环境。IP SLA 会收集并分析下列性能度量值：

- 延迟（往返和单向）
- 抖动（有方向性）
- 丢包（有方向性）
- 数据包序列（数据包排序）
- 路径（逐跳）
- 连通性（有方向性）
- 服务器或网站下载时间

由于 Inspur INOS IP SLA 可以由 SNMP 进行访问，因此性能监控应用也能使用它，比如 Inspur Prime 互联网络性能监控器（IPM）和其他第三方 Inspur 合作伙伴的性能管理产品。

使用 IP SLA 可以获得以下好处：

- 服务等级协定监控、评估和验证；
- 网络性能监控：
  - 评估网络中的抖动、延迟或丢包
  - 可持续性、可靠性和可预测性评估
- IP 服务网络健康评估特性能够验证现有的 QoS 是否能够满足新的 IP 服务；
- 边界到边界网络可用性监控特性能够提供主动的网络资源验证和连通性测试（举例来说，从远端站点查看用来存储业务重要数据的 NFS 服务器的网络可用性）；
- 通过提供持续且可靠的评估结果，可以立即指出问题并节省排错时间，以此实现网络探针排错；

- 多协议标签交换（MPLS）性能监控和网络验证（如果设备支持 MPLS 的话）。

使用 Inspur INOS IP SLA 评估网络性能

用户可以使用 IP SLA 来监控网络中各个区域之间的性能——核心层、分布层和边界——而无需部署物理探针。它会通过生成的流量来测量两台网络通信设备之间的网络性能。
下图展示了当源色湖北向目的设备发送了它所生成的数据包后，IP SLA 是如何展开工作的。在目的设备收到数据包后，根据 IP SLA 探针的类型，它会向源设备反悔带有时间戳的信息，以便计算性能度量值。IP SLA 探针可以使用某种具体协议（比如 UDP）来评估网络中源设备与目的设备之间的网络性能。

图 74：Inspur INOS IP SLA 探针

Any IP device	任意 IP 设备
IP SLA measurement and IP SLA responder to IP SLA responder（共 2 处）	IP SLA 评估以及 IP SLA 响应方到 IP SLA 响应方
IP SLA responder	IP SLA 响应方
IP network	IP 网络
IP SLA source	IP SLA 源
Performance management application	性能管理应用

IP SLA 响应方和 IP SLA 控制协议

IP SLA 响应方是内嵌在目的 Inspur 设备中的一个组成部分，它使系统能够参与并响应 IP SLA 请求数据包。响应方能够提供精确的评估，无需部署专用的探针。响应方使用 Inspur INOS IP SLA 控制协议，使其能够知道应该监听哪个端口以及使用哪个端口进行响应。
注释： IP SLA 响应方可以是 Inspur INOS 二层（可配置响应方的）设备。响应方无需支持完整的 IP SLA 功能。
下图展示了 IP 网络中能够使用 Inspur INOS IP SLA 响应方的位置。响应方会在指定端口上监听由 IP SLA 探针发来的控制协议消息。根据收到的控制消息，它能够在指定时间段内启用指定的 UDP 或 TCP 端口。在这段时间内，响应方会接受请求并对其进行响应。在它对 IP SLA 数据包做出响应后，或者当指定时间超时后，它会禁用相应的端口。为了增加安全性，设备可以为控制消息实施 MD5 认证

图 75：Inspur INOS IP SLA 探针

Any IP device	任意 IP 设备
IP SLA measurement and IP SLA responder to IP SLA responder（共 2 处）	IP SLA 评估以及 IP SLA 响应方到 IP SLA 响应方
IP SLA responder	IP SLA 响应方
IP network	IP 网络
IP SLA source	IP SLA 源

Performance management application	性能管理应用

用户无需在目的设备上为所有 IP SLA 探针启用响应方。举例来说，对于目的路由器上已经提供的服务就无需启用响应方（比如 Telnet 或 HTTP）。

IP SLA 响应时间的计算

交换机、控制器和路由器可以通过使用其他高优先级进程，花费几十毫秒来处理入站数据包。这个延迟会影响响应时间，因为测试数据包的响应可能会在等待处理时进入队列。在这种情况中，响应时间可能无法精确反映出真实的网络延迟。IP SLA 会把源设备和目标设备上（如果使用了响应方）的这些处理延迟最小化，以此来确定真实的往返时间。IP SLA 测试数据包会使用时间戳来把处理延迟最小化。

当启用了 IP SLA 响应方后，它使目标设备在数据包到达接口时，在中断级（Interrupt Level）设置时间戳，并在测试数据包离开时也设置时间戳，以此消除处理时间。这个时间戳精确到亚毫秒级。

下图展示了响应方的工作。其中共有四个时间戳用来计算往返时间。在目标路由器上，当启用了响应方功能后，用时间戳 3（TS3）减去时间戳 2（TS2）就能够得到花费在处理测试数据包上的时间，也就是图中 Δ 表示的内容。最后 IP SLA 会从整体往返时间中减去这个 Δ 值。源路由器上的 IP SLA 也会执行相同的行为，也就是会在中断级设置入站时间戳 4（TS4），来达到更高的精确度。

图 76：Inspur INOS IP SLA 响应方时间戳

Source router	源路由器
Target router	目标路由器
Responder	响应方
RTT (Round-trip time) = T4 (Time stamp 4) – T1 (Time stamp 1) - Δ	RTT（往返时间）= T4（时间戳 4）– T1（时间戳 1）– Δ

在目标设备上设置两个时间戳带来的另一个好处是：能够追踪单向延迟、抖动和单向丢包。由于很多网络行为都不是同步发生的，因此很有必要记录这些统计数据。然而，在进行单向延迟评估时，用户必须同时在源路由器和目标路由器上配置网络时间协议（NTP），这样源和目标才能与相同的时钟源进行同步。单向抖动的评估不需要同步时钟。

IP SLA 探针计划

当用户在配置 IP SLA 探针时，必须对探针开始捕获状态统计信息的时间，以及收集错误信息的时间有所计划。用户可以设置让探针立即开始工作，也可以指定具体的月、日、小时。用户可以使用 *pending（待定）*选项让探针稍后开始工作。待定选项是探针的一种内部状态，可以通过 SNMP 进行查看。当一个探针的响应（门限值）行为有待触发时，探针的状态也是待定。用户可以一次性规划一个 IP SLA 探针的工作时间，也可以同时规划一组探针的工作时间。

用户可以在 Inspur INOS CLI 或 INSPUR RTTMON-MIB 中使用一条命令来规划多个 IP SLA 探针的工作时间。通过规划让探针平均地分多次运行，可以让用户控制 IP SLA 监控流量的总量。这种分布 IP SLA 探针的工作方式有助于最小化 CPU 利用率，从而提高网络可扩展性。

更多有关 IP SLA 多探针计划功能的详细信息，用户可以参考 *Inspur INOS IP SLA 配置指南（Inspur INOS IP SLAs Configuration Guide）*中“IP SLA——多探针计划”一章。

IP SLA 探针门限值监控

为了成功地进行服务等级协定监控，必须有某种机制能够及时向用户通知网络中可能发生的违规行为。IP SLA 可以发送 SNM Trap 消息，下列这些事件可以作为触发机制：

- 连接断开
- 超时
- 往返时间门限值
- 平均抖动门限值
- 单向丢包
- 单向抖动
- 单向平均意见得分（MOS）
- 单向延迟

一个 IP SLA 门限值检测到的违规行为同时也会触发另一个 IP SLA 探针进行深入分析。比如提高测试频率，或者开始使用 Internet 控制消息协议（ICMP）路径应答或 ICMP 路径抖动探针来进行排错。

ICMP Echo

ICMP Echo（应答）探针能够评估 Inspur 设备与其他 IP 设备之间端到端的响应时间。响应时间是由测量源设备向目的设备发出 ICMP Echo 请求消息，再到源设备收到 ICMP Echo Reply 之间所花费的时间计算出来的。很多客户都会使用 IP SLA 中基于 ICMP 的探针，进行内部 Ping 测试，或者基于 Ping 的专用探针来评估响应时间。IP SLA ICMP Echo 探针符合 ICMP Ping 测试的定义，这两种方式都可以得到相同的响应时间。

UDP 抖动

抖动与数据包之间延迟的变化是同义术语。当源设备向目的设备以 10 毫秒为间隔连续发送多个数据包时，目的设备应该每隔 10 毫秒收到一个数据包（如果网络行为完全正常的话）。但如果网络中存在延迟（比如队列延迟、通过替代路径到达目的地等），数据包的到达时间间隔可能会小于或大于 10 毫秒。正抖动值表示数据包的到达时间间隔大于 10 毫秒。负抖动值表示数据包的到达时间间隔小于 10 毫秒。如果数据包的到达时间间隔是 12 毫秒，正抖动就是 2 毫秒；如果数据包的到达时间间隔是 8 毫秒，负抖动就是 2 毫秒。对于延迟敏感的网络，正抖动是不可容忍的，抖动值为 0 是理想情况。

除了监控抖动外，IP SLA UDP 抖动探针还能当作多目的数据收集探针。由 IP SLA 生成的数据包中会携带序列号信息，以及从源到目的的时间戳，其中还包含数据包的发送和接收数据。根据这些数据，UDP 抖动探针可以评估下列度量值：

- 有方向性的抖动（源到目的，以及目的到源）
- 有方向性的丢包
- 有方向性的延迟（单向延迟）
- 往返延迟（平均往返时间）

由于发送和接收数据的路径可能有多条（不对称路径），用户可以使用有方向性的数据测试，更有效地识别网络中拥塞的位置，或者网络中发生的其他问题。

UDP 抖动探针能够产生合成（模拟）UDP 流量，并发送大量 UDP 数据包，并且每个数据包都有指定的大小，发送指定的毫秒数，从源路由器到目标路由器以固定的频率发送。默认情况下，UDP 抖动探针会发送 10 个数据包-数据帧，每个负载大小为 10 字节，每 10 毫秒生成一个，每 60 秒重复进行测试。用户可以对这些参数一一进行配置，精确模拟网络中的 IP 服务。
为了提供精确的单向延迟评估结果，源设备和目标设备之间需要进行时间同步（比如 NTP 提供的服务）。评估单向抖动和丢包不必需进行时间同步。如果源设备和目标设备之间的时间没有同步，UDP 抖动探针在反回单向延迟和丢包的评估数据时，单向延迟的评估结果为 0。

如何配置 IP SLA 探针

这部分中不包含所有可用的探针配置信息，*Inspur INOS IP SLA 配置指南（Inspur INOS IP SLAs Configuration Guide）*中包含更多详细配置信息。这部分中会包含多个探针配置示例，其中包括配置响应方、配置 UDP 抖动探针（需要配置响应方），以及配置 ICMP Echo 探针（不需要配置响应方）。有关配置其他探针的详细信息，用户可以参考 *Inspur INOS IP SLA 配置指南（Inspur INOS IP SLAs Configuration Guide）*。

默认配置

设备中没有配置 IP SLA 探针。

配置指导

有关 IP SLA 命令的信息，用户可以查看 *Inspur INOS IP SLA 命令参考，版本 12.4T（Inspur IP SLAs Command Reference, Release 12.4T）*命令参考手册。
有关描述和配置步骤的细节信息，用户可以查看 *Inspur INOS IP SLA 配置指南，版本 12.4TL（Inspur INOS IP SLAs Configuration Guide, Release 12.4TL）*。
并不是参考指南中的所有 IP SLA 命令或探针设备都能够支持。设备支持的 IP 服务等级分析功能包括使用 UDP 抖动、UDP Echo、HTTP、TCP 连接、ICMP Echo、ICMP 路径 Echo、ICMP 路径抖动、TFP、DNS 和 DHCP，以及多种探针计划和主动门限值监控。设备不支持使用网守注册延迟探针评估的 VoIP 服务等级。
在配置任意 IP SLA 应用之前，用户可以使用特权 EXEC 命令 **show ip sla application**，来确认软件版本所支持的探针类型。这条命令的输出示例如下所示：

```
Device# show ip slaapplication
IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III
Supported OperationTypes:
icmpEcho, path-echo, path-jitter, udpEcho,tcpConnect, http
dns, udpJitter, dhcp,ftp, udpApp, wspApp
Supported Features:
IPSLAs Event Publisher
```

```
IP SLAs low memory water mark: 33299323
Estimated system maxnumber of entries: 24389
Estimated number ofconfigurable operations:24389
Number of Entriesconfigured : 0
Number of active Entries: 0
Number of pending Entries: 0
Number of inactiveEntries : 0
Time of last change inwhole IP SLAs: *13:04:37.668 UTC Wed Dec 19
2012
```

配置 IP SLA 响应方

只有运行 Inspur INOS 软件的设备上能够使用 IP SLA 响应方功能，其中包括不支持完整 IP SLA 功能的一些二层设备。

用户可以按照以下步骤，在目标设备（探针目标）上配置 IP SLA 响应方。

总步骤

1. enable

2. configure terminal

3. ip sla responder {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number*

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**ip sla responder** {**tcp-connect** | **udp-echo**} **ipaddress** *ip-address* **port** *port-number* **示例：** `Device(config)# ip sla responder udp-echo 172.29.139.134 5000`	把设备配置为 IP SLA 响应方。 关键字的解释如下所示： • **tcp-connect**——为 TCP 连接探针启用响应方 • **udp-echo**——为用户数据报协议（UDP）Echo 或抖动探针启用响应方 • **ipaddress** *ip-address*——输入目的 IP 地址 • **port** *port-number*——输入目的端口号

		注释： IP 地址和端口号必须与源设备上为 IP SLA 探针配置的 IP 地址和端口号相同
步骤 4	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 5	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 6	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

实施 IP SLA 网络性能评估

用户可以按照以下步骤，在设备上实施 IP SLA 网络性能评估。

在开始前

用户可以使用特权 EXEC 命令 **show ip sla application**，来确认软件版本所支持的探针类型。

总步骤

1. enable

2. configure terminal

3. ip sla *operation-number*

4. udp-jitter {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]

5. frequency *seconds*

6. threshold *milliseconds*

7. exit

8. ipsla schedule *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]

9. end

10. show running-config

11. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码

步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式
步骤 3	**ip sla** *operation-number* **示例：** Device(config)# **ip sla 10**	创建 IP SLA 探针，并进入 IP SLA 配置模式
步骤 4	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*] **示例：** Device(config-ip-sla)# **udp-jitter 172.29.139.134 5000**	配置 IP SLA 探针，用户可以任意选择探针类型（示例中使用了 UDP 抖动探针），并进入这个探针的配置模式（示例中进入了 UDP 抖动配置模式）。 • *destination-ip-address* | *destination-hostname*——指定目的 IP 地址或主机名 • *destination-port*——指定目的端口，取值范围是 1 至 65535 • （可选）**source-ip** {*ip-address* | *hostname*}——指定源 IP 地址或主机名。如果用户没有指定源 IP 地址或主机名，IP SLA 会选择距离目的地最近的 IP 地址 • （可选）**source-port** *port-number*——指定源端口号，取值范围是 1 至 65535。当用户没有指定源端口号时，IP SLA 会选择一个可用端口 • （可选）**control**——启用或禁用设备向 IP SLA 响应方发送 IP SLA 控制消息的功能。默认情况下，设备会向目的设备发送 IP SLA 控制消息，以便与 IP SLA 响应方建立连接 • （可选）**num-packets** *number-of-packets*——输入需要生成的数据包数量。取值范围是 1 至 6000；默认值为 10 • （可选）**interval** *interpacket-interval*——以毫秒为单位输入发送数据包的时间间隔。取值范围是 1 至 6000；默认值为 20 毫秒
步骤 5	**frequency** *seconds* **示例：**	（可选）为 SLA 探针配置选项信息。示例中指定了 IP SLA 探针重复执行的速率。取值范围是 1 至 604800；默认

	Device(config-ip-sla-jitter)# **frequency 45**	值为 60 秒
步骤 6	**threshold** *milliseconds* **示例：** Device(config-ip-sla-jitter)# **threshold 200**	（可选）配置门限值条件。示例中把指定 IP SLA 探针的门限值设置为 200。取值范围是 1 至 60000 毫秒
步骤 7	**exit** **示例：** Device(config-ip-sla-jitter)# **exit**	退出 SLA 探针配置模式（也就是示例中的 UDP 抖动配置模式），并返回全局配置模式
步骤 8	**ipsla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] **示例：** Device(config)# **ip sla schedule 10 start-time now life forever**	为某个 IP SLA 探针配置计划参数。 • *operation-number*——输入 RTR 条目数量 • （可选）**life**——设置探针永远运行（**forever**）或以秒 *seconds* 为单位指定时长。取值范围是 0 至 2147483647；默认值为 3600 秒（1 小时） • （可选）**start-time**——输入这个探针开始收集信息的时间： 要想在指定时间开始，输入小时、分钟、秒（以 24 小时格式），以及月份日期。如果没有输入月份，默认为当前月。 输入 **pending** 表示不收集信息，直到设置了开始时间。 输入 **now** 表示马上开始运行探针。 输入 **after** *hh:mm:ss* 表示在指定时间过后开始执行探针。 • （可选）**ageout** *seconds*——以秒为单位输入当探针并没有收集到信息使，把探针保存在内存中的时间。取值范围是 0 至 2073600 秒，默认值为 0 秒（永不超时） • （可选）**recurring**——设置让探针每天自动运行
步骤 9	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式
步骤 10	**show running-config**	检查用户输入的信息

	示例： Device# **show running-config**	
步骤 11	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）把输入的命令保存到配置文件中

UDP 抖动配置

以下示例展示了如何配置 UDP 抖动 IP SLA 探针：

```
Device(config)# ip sla 10
Device(config-ip-sla)# udp-jitter 172.29.139.134 5000
Device(config-ip-sla-jitter)# frequency 30
Device(config-ip-sla-jitter)# exit
Device(config)# ip sla schedule 5 start-time now life forever
Device(config)# end
Device# show ip sla configuration 10
IP SLAs, InfrastructureEngine-II.
Entry number: 10
Owner:
Tag:
Type of operation toperform: udp-jitter
Target address/Sourceaddress: 1.1.1.1/0.0.0.0
Target port/Source port: 2/0
Request size (ARR dataportion): 32
Operation timeout(milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
Operation frequency(seconds): 30
Next Scheduled StartTime: Pending trigger
Group Scheduled : FALSE
Randomly Scheduled: FALSE
Life (seconds): 3600
Entry Ageout (seconds):never
Recurring (StartingEveryday): FALSE
Status of entry (SNMPRowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
Number of statistic hours kept:2
Number of statisticdistribution buckets kept: 1
```

```
Statistic distributioninterval (milliseconds): 20
Enhanced History:
```

使用 UDP 抖动探针分析 IP 服务等级

用户可以按照以下步骤，在源设备上配置 UDP 抖动探针。

在开始前

用户必须在目标设备（探针目标）上启用 IP SLA 响应方功能，以便在源设备上配置 UDP 抖动探针。

总步骤

1. enable

2. configure terminal

3. ip sla *operation-number*

4. udp-jitter {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interparket-interval*]

5. frequency *seconds*

6. exit

7. sla schedule *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]

8. end

9. show running-config

10. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**ip sla** *operation-number* **示例：** `Device(config)# ip sla 10`	创建 IP SLA 探针，并进入 IP SLA 配置模式
步骤 4	**udp-jitter** {*destination-ip-address* | *destination-hostname*} *destination-port* [**source-ip** {*ip-address* | *hostname*}] [**source-port** *port-number*] [**control** {**enable** | **disable**}] [**num-packets** *number-of-packets*] [**interval** *interpacket-interval*]	配置 IP SLA 探针，用户可以任意选择探针类型（示例中使用了 UDP 抖动探针），并进入这个探针的配置模式（示例中进入了 UDP 抖动配置模式）。 • *destination-ip-address* |

	示例： `Device(config-ip-sla)# udp-jitter 172.29.139.134 5000`	*destination-hostname*——指定目的 IP 地址或主机名 • *destination-port*——指定目的端口，取值范围是 1 至 65535 • （可选）**source-ip** {*ip-address* | *hostname*}——指定源 IP 地址或主机名。如果用户没有指定源 IP 地址或主机名，IP SLA 会选择距离目的地最近的 IP 地址 • （可选）**source-port** *port-number*——指定源端口号，取值范围是 1 至 65535。当用户没有指定源端口号时，IP SLA 会选择一个可用端口 • （可选）**control**——启用或禁用设备向 IP SLA 响应方发送 IP SLA 控制消息的功能。默认情况下，设备会向目的设备发送 IP SLA 控制消息，以便与 IP SLA 响应方建立连接 • （可选）**num-packets** *number-of-packets*——输入需要生成的数据包数量。取值范围是 1 至 6000；默认值为 10 • （可选）**interval** *interpacket-interval*——以毫秒为单位输入发送数据包的时间间隔。取值范围是 1 至 6000；默认值为 20 毫秒
步骤 5	**frequency** *seconds* **示例：** `Device(config-ip-sla-jitter)# frequency 45`	（可选）为 SLA 探针配置选项信息。示例中指定了 IP SLA 探针重复执行的速率。取值范围是 1 至 604800；默认值为 60 秒
步骤 6	**exit** **示例：** `Device(config-ip-sla-jitter)# exit`	退出 UDP 抖动配置模式，并返回全局配置模式
步骤 7	**ipsla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]	为某个 IP SLA 探针配置计划参数。 • *operation-number*——输入 RTR 条目数量 • （可选）**life**——设置探针永远运行（**forever**）或以秒 *seconds*

	示例： `Device(config)# ip sla schedule 10 start-time now life forever`	为单位指定时长。取值范围是 0 至 2147483647；默认值为 3600 秒（1 小时） • （可选）**start-time**——输入这个探针开始收集信息的时间： 要想在指定时间开始，输入小时、分钟、秒（以 24 小时格式），以及月份日期。如果没有输入月份，默认为当前月。 输入 **pending** 表示不收集信息，直到设置了开始时间。 输入 **now** 表示马上开始运行探针。 输入 **after** *hh:mm:ss* 表示在指定时间过后开始执行探针。 • （可选）**ageout** *seconds*——以秒为单位输入当探针并没有收集到信息使，把探针保存在内存中的时间。取值范围是 0 至 2073600 秒，默认值为 0 秒（永不超时） • （可选）**recurring**——设置让探针每天自动运行
步骤 8	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 9	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 10	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

配置 UDP 抖动 IP SLA 探针

以下示例展示了如何配置 UDP 抖动 IP SLA 探针：

```
Device(config)# ip sla 10
Device(config-ip-sla)# udp-jitter 172.29.139.134 5000
Device(config-ip-sla-jitter)# frequency 30
Device(config-ip-sla-jitter)# exit
Device(config)# ip sla schedule 5 start-time now life forever
Device(config)# end
```

```
Device# show ip sla configuration 10
IP SLAs, InfrastructureEngine-II.
Entry number: 10
Owner:
Tag:
Type of operation toperform: udp-jitter
Target address/Sourceaddress: 1.1.1.1/0.0.0.0
Target port/Source port: 2/0
Request size (ARR dataportion): 32
Operation timeout(milliseconds): 5000
Packet Interval (milliseconds)/Number of packets: 20/10
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Control Packets: enabled
Schedule:
Operation frequency(seconds): 30
Next Scheduled StartTime: Pending trigger
Group Scheduled : FALSE
Randomly Scheduled: FALSE
Life (seconds): 3600
Entry Ageout (seconds):never
Recurring (StartingEveryday): FALSE
Status of entry (SNMPRowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
Number of statistic hours kept:2
Number of statisticdistribution buckets kept: 1
Statistic distributioninterval (milliseconds): 20
Enhanced History:
```

使用 ICMP Echo 探针分析 IP 服务等级

用户可以按照以下步骤，在源设备上配置 ICMP Echo 探针：

在开始前

这个探针不需要启用 IP SLA 响应方。

总步骤

1. enable

2. configure terminal

3. ip sla *operation-number*

4. icmp-echo {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*]

5. frequency *seconds*

6. exit

7. sla schedule *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm* [*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**]

8. end

9. show running-config

10. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	进入特权 EXEC 模式。在提示时输入密码
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式
步骤 3	**ip sla** *operation-number* **示例：** `Device(config)# ip sla 10`	创建 IP SLA 探针，并进入 IP SLA 配置模式
步骤 4	**icmp-echo** {*destination-ip-address* | *destination-hostname*} [**source-ip** {*ip-address* | *hostname*} | **source-interface** *interface-id*] **示例：** `Device(config-ip-sla)# icmp-echo 172.29.139.134`	配置 ICMP Echo 探针，并进入 ICMP Echo 配置模式。 • *destination-ip-address* | *destination-hostname*——指定目的 IP 地址或主机名 • （可选）**source-ip** {*ip-address* | *hostname*}——指定源 IP 地址或主机名。如果用户没有指定源 IP 地址或主机名，IP SLA 会选择距离目的地最近的 IP 地址 • （可选）**source-interface** *interface-id*——为探针指定源接口
步骤 5	**frequency** *seconds* **示例：** `Device(config-ip-sla-echo)# frequency 45`	（可选）为 SLA 探针配置选项信息。示例中指定了 IP SLA 探针重复执行的速率。取值范围是 1 至 604800；默认值为 60 秒
步骤 6	**exit** **示例：** `Device(config-ip-sla-echo)# exit`	退出 UDP 抖动配置模式，并返回全局配置模式
步骤 7	**ipsla schedule** *operation-number* [**life** {**forever** | *seconds*}] [**start-time** {*hh:mm*	为某个 IP SLA 探针配置计划参数。 • *operation-number*——输入 RTR

	[*:ss*] [*month day* | *day month*] | **pending** | **now** | **after** *hh:mm:ss*] [**ageout** *seconds*] [**recurring**] **示例：** `Device(config)# ip sla schedule 10 start-time now life forever`	条目数量 • （可选）**life**——设置探针永远运行（**forever**）或以秒 *seconds* 为单位指定时长。取值范围是 0 至 2147483647；默认值为 3600 秒（1 小时） • （可选）**start-time**——输入这个探针开始收集信息的时间： 要想在指定时间开始，输入小时、分钟、秒（以 24 小时格式），以及月份日期。如果没有输入月份，默认为当前月。 输入 **pending** 表示不收集信息，直到设置了开始时间。 输入 **now** 表示马上开始运行探针。 输入 **after** *hh:mm:ss* 表示在指定时间过后开始执行探针。 • （可选）**ageout** *seconds*——以秒为单位输入当探针并没有收集到信息使，把探针保存在内存中的时间。取值范围是 0 至 2073600 秒，默认值为 0 秒（永不超时） • （可选）**recurring**——设置让探针每天自动运行
步骤 8	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 9	**show running-config** **示例：** `Device# show running-config`	检查用户输入的信息
步骤 10	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把输入的命令保存到配置文件中

配置 ICMP Echo IP SLA 探针

以下示例展示了如何配置 ICMP Echo IP SLA 探针：

```
Device(config)# ip sla 12
Device(config-ip-sla)# icmp-echo 172.29.139.134
Device(config-ip-sla-jitter)# frequency 30
```

```
Device(config-ip-sla-jitter)# exit
Device(config)# ip sla schedule 5 start-time now life forever
Device(config)# end
Device# show ip sla configuration 12
Entry number: 12
Owner:
Tag:
Type of operation toperform: echo
Target address: 2.2.2.2
Source address: 0.0.0.0
Request size (ARR dataportion): 28
Operation timeout(milliseconds): 5000
Type Of Service parameters: 0x0
Verify data: No
Vrf Name:
Schedule:
Operation frequency(seconds): 60
Next Scheduled StartTime: Pending trigger
Group Scheduled : FALSE
Randomly Scheduled: FALSE
Life (seconds): 3600
Entry Ageout (seconds):never
Recurring (StartingEveryday): FALSE
Status of entry (SNMPRowStatus): notInService
Threshold (milliseconds): 5000
Distribution Statistics:
Number of statistic hours kept:2
Number of statisticdistribution buckets kept: 1
Statistic distributioninterval (milliseconds): 20
History Statistics:
Number of history Liveskept: 0
Number of history Buckets kept: 15
History Filter Type: None
Enhanced History:
```

监控 IP SLA 探针

下面这个表格中描述了用来查看 IP SLA 探针配置和结果的命令。

表 76：监控 IP SLA 探针

命令	目的
show ip sla application	显示有关 INOS IP SLA 的全局信息
show ip sla authentication	显示 IP SLA 认证信息
show ip sla configuration [*entry-number*]	显示配置值，其中包括所有 IP SLA 探针或指

	定探针的所有默认值
show ip sla enhanced-history {**collection-statistics** | **distribution-statistics**} [*entry-number*]	以收集的历史桶或分布统计信息形式，为所有 IP SLA 探针或指定探针显示高级历史统计状态信息
show ip sla ethernet-monitor configuration [*entry-number*]	显示 IP SLA 自动以太网配置
show ip sla group schedule [*schedule-entry-number*]	显示 IP SLA 组计划配置及其详细信息
show ip sla history [*entry-number* | **full** | **tabular**]	显示为所有 IP SLA 探针收集的历史信息
show ip sla mpls-lsp-monitor {**collection-statistics** | **configuration** | **ldp operational-state** | **scan-queue** | **summary** [*entry-number*] | **neighbors**}	显示 MPLS 标签交换路径（LSP）健康监控器探针
show ip sla reaction-configuration [*entry-number*]	显示为所有 IP SLA 探针或指定探针设置的正门限值监控设置
show ip sla reaction-trigger [*entry-number*]	显示为所有 IP SLA 探针或指定探针设置的响应触发器信息
show ip sla responder	显示有关 IP SLA 响应方的信息
show ip sla statistics [*entry-number* | **aggregated** | **details**]	显示当前或汇集的探针状态和统计状态信息

监控 IP SLA 探针示例

以下示例展示了所有 IP SLA 应用的相关信息：

```
Device# show ip slaapplication
IP Service Level Agreements
Version: Round Trip Time MIB 2.2.0, Infrastructure Engine-III
Supported OperationTypes:
icmpEcho, path-echo, path-jitter, udpEcho,tcpConnect, http
dns, udpJitter, dhcp,ftp, udpApp, wspApp
Supported Features:
IPSLAs Event Publisher
IP SLAs low memory water mark: 33299323
Estimated system maxnumber of entries: 24389
Estimated number ofconfigurable operations:24389
Number of Entriesconfigured  : 0
Number  of  active  Entries: 0
Number  of  pending  Entries: 0
Number of inactiveEntries  : 0
Time of last change inwhole IP SLAs: *13:04:37.668 UTC Wed Dec 19
2012
```

以下示例展示了所有 IP SLA 分布统计信息：

```
Device# show ip slaenhanced-history distribution-statistics
Point by point EnhancedHistory
Entry  = Entry Number
Int  = Aggregation Interval
BucI  = Bucket Index
StartT  = Aggregation Start Time
Pth  = Path index
Hop  = Hop in path index
Comps  = Operations completed
OvrTh  = Operations completedover thresholds
SumCmp  = Sum of RTT (milliseconds)
SumCmp2L = Sum of RTTsquared low 32 bits (milliseconds)
SumCmp2H = Sum of RTTsquared high 32 bits(milliseconds)
TMax  = RTT maximum (milliseconds)
TMin  = RTT minimum (milliseconds)
Entry Int BucI StartT  Pth Hop Comps OvrThSumCmp  SumCmp2L  SumCmp2H
T
Max  TMin
```

其他参考资料

相关文档

相关主题	文档名称
Inspur Medianet Metadata Guide	*http://wwwicntnetworks.com*
Inspur Media Services Proxy Configuration Guide	*http://www.icntnetworks.com*
Inspur Mediatrace and Inspur Performance Monitor Configuration Guide	*http://www.icntnetworks.com*

错误消息解码器

描述	链接
为了帮助用户查找并解决于这个版本相关的系统错误消息，用户可以使用错误消息解码器（Error Message Decoder）工具	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网页中提供了大量在线资源，其中的文档和工具能够用来帮助用户排查和解决与 Inspur 产品和技术相关技术问题。	http://www.icntnetworks.com

要想收到与用户自己产品相关的安全和技术信息，用户可以订阅多种服务，比如产品告警工具（Product Alert Tool；从 Field Notices 中进行访问）、Inspur 技术服务时事（Technical Services Newsletter）和简易信息聚合（RSS）消息。 在 Inspur 支持网页上访问大多数工具都需要用户在 icntnetworks.com 上注册用户 ID 和密码。	

系统管理

管理交换机

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于管理设备的信息

系统时间及日期管理

可以使用自动配置方式（RTC 以及 NTP）或者手动配置方式来管理设备上的系统时间及日期。

注释： 有关本节使用命令的完整语法以及使用信息，参见 *icntnetworks.com* 上的 *Inspur INOS 配置基本命令参考*。

系统时钟

时间服务的基础是系统时钟。时钟从系统启动就开始运行，跟踪记录着日期和时间。
系统时钟可以通过以下来源设置：

- NTP
- 手动配置

系统时钟可以给以下服务提供时间：

- 用户 **show** 命令
- 日志以及调试信息

系统时钟内部基于世界协调时间（Coordinated Universal Time，UTC）（也称为格林威治标准时间，Greenwich Mean Time，GMT）来记录时间。可以配置本地时区以及夏令时的信息，以正确显示本地时区的时间。
系统时钟会记录时间是否*权威*（即是否由被认为权威的时间源设置）。如果不权威，该时间只用于显示信息，不会被分发其他设备。

网络时间协议

NTP 的目的是对网络中的设备进行时间同步。NTP 在用户数据报协议（User Datagram Protocol，UDP）之上运行，并通过 IP 运行。NTP 在 RFC 1305 中说明。
NTP 网络通常通过权威时间源获取时间，比如连接到时间服务器的无线电时钟或原子钟，之后 NTP 会在网络中分发时间信息。NTP 极其高效，要把两台设备的时间差同步在一毫秒以内，每分钟最多只需要一个数据包。

NTP 层

NTP 使用*层（stratum）*的概念来描述一台设备距离权威时间源有多个 NTP 跳远。第 1 层的时间服务器有直连的无线电时钟或原子钟，第 2 层的时间服务器通过 NTP 接收第 1 层时间服务器的时间，以此类推。运行 NTP 的设备会在与其通过 NTP 通信的时间源中选择层数最低的设备作为自己的时间源。这样的策略能够高效地构建自组的 NTP 通告者树。
NTP 不会从一个未同步的设备上同步时间，进而避免了同步的设备时间不准确的问题。NTP 也会比较几台设备汇报的时间，如果一台设备层数较低，但是它提供的时间与其他设备的时间明显不同，NTP 也不会从该设备同步时间。

NTP 关联

运行 NTP 的设备之间的通信（也称为关联）通常是静态配置的；每台设备都配置了应该建立关联的所有设备的 IP 地址。每对关联设备之间交换 NTP 消息使得可以进行精确的计时。然而，在 LAN 环境中，可以配置 NTP 使用 IP 广播消息。这种方式减少了配置复杂性，因为可以直接配置每台设备发送或接受广播消息。不过此时的信息流仅能是单向的。

NTP 安全

设备上记录的时间是一种关键资源，应该启用 NTP 安全特性，避免意外或被恶意设置了不正确的时间。有两种安全机制可以使用：基于访问列表的限制机制以及加密认证机制。

NTP 实现

Inspur 的 NTP 实现不支持第 1 层服务，设备不能连接无线电时钟或原子钟。建议通过 IP Internet 上的公共 NTP 服务器获取网络时间。

下图展示了一个典型的使用 NTP 的网络。设备 A 是 NTP master，设备 B、C 和 D 都配置为 NTP 服务器模式，与设备 A 进行关联。设备 E 分别是上游和下游设备 B 和设备 F 的 NTP 对端。

图 132：典型的 NTP 网络配置

Switch A	交换机 A
Localworkgroup servers	本地工作组服务器
Workstations	工作站

如果网络与 Internet 隔离，Inspur 的 NTP 实现允许设备在通过其他方式学习时间时，当作是通过 NTP 同步的来进行操作。之后其他设备可以与该设备通过 NTP 进行同步。

当有多个时间源可用时，NTP 总被认为是更权威的。NTP 时间会覆盖由其他方式设置的时间。

一些厂商的主机系统中带有 NTP 软件，UNIX 及众多衍生版本系统也有公开可用的 NTP 软件。这些软件允许主机系统进行时间同步。

NTP 第 4 版

本设备上实现了 NTP 第 4 版。NTPv4 是 NTP 第 3 版的扩展。NTPv4 支持 IPv4 以及 IPv6，并向后兼容 NTPv3。

NTPv4 提供以下功能：

- 支持 IPv6。
- 比 NTPv3 提升了安全性。NTPv4 使用基于公有密钥加密以及标准 X509 证书的安全框架。
- 能够自动计算网络的时间分发层级。使用特定的组播组，NTPv4 能够自动配置服务器层级，以通过最低的带宽开销实现最高的时间精确性。此特性使用 IPv6 站点本地组播地址。

配置 NTPv4 的更多信息参见 *Inspur INOS IPv6 配置指南 12.4T 版*中的*在 IPv6 中部署 NTPv4* 章节。

系统名称及提示符

可以在设备上配置系统名称以进行标识。默认情况下，系统名称及提示符是 Device。

如果没有配置系统提示符，系统名称的前 20 个字符被用于系统提示符，之后附带一个大于号“>”。提示符会在系统名改变时更新。

有关本节使用命令的完整语法以及使用信息，参见 *Inspur INOS 配置基本命令参考 12.4 版*以及 *Inspur INOS IP 命令参考第 2 卷：路由协议 12.4 版*。

堆栈系统名称及提示符

如果通过活跃交换机访问堆栈成员，必须使用特权 EXEC 命令 **session** *stack-member-number*。堆栈成员编号范围是 1 到 9。使用此命令时，堆栈成员编号会被附加在系统提示符之后。例如，Switch-2#是堆栈成员 2 的特权 EXEC 模式提示符，而交换机堆栈的系统提示符是 Switch。

默认系统名称及提示符配置

默认的交换机系统名称及提示符是 *Switch*。

DNS

DNS 协议控制着域名系统（Domain Name System，DNS），这是可以把主机名映射到 IP 地址的分布式数据库。在设备上配置 DNS 时，可以使用主机名替换所有 IP 命令中的 IP 地址，如 **ping**、**telnet**、**connect** 以及相关的 Telnet 支持操作。

IP 地址定义了一种层级化的命名机制，允许通过地址或域名标识设备。域名使用英文句号（.）作为分隔符拼接在一起。例如，Inspur Systems 是一家商业机构，IP 地址通过 *com* 域名标识，所以其域名是 *icntnetworks.com*。此域名中的一台特定设备，如文件传输协议（File Transfer Protocol，FTP）系统由 *ftp.icntnetworks.com* 标识。

为了记录域名，IP 定义了域名服务器的概念，该服务器会保存名称到 IP 地址映射的缓存（或数据库）。要把域名映射到 IP 地址，必须先标识出主机名，然后指定网络上现有的名称服务器，并且要启用 DNS。

默认 DNS 设置

表 174：默认 DNS 设置

特性	默认设置
DNS 启用状态	启用。
DNS 默认域名	未配置。
DNS 服务器	未配置名称服务器地址。

登录标语

可以配置当日消息（message-of-the-day，MOTD）以及登录标语。MOTD 标语会在所有连接终端登录时显示，可以用来发送会影响所有网络用户的消息（如即将进行的系统关机）。

登录标语也会在所有连接终端上显示。它出现在 MOTD 标语之后，登录提示之前。

注释： 有关本节使用命令的完整语法以及使用信息，参见 *Inspur INOS 配置基本命令参考 12.4 版*。

默认标语配置

MOTD 以及登录标语未被配置。

MAC 地址表

MAC 地址表包含了设备用来在端口之间转发流量使用的地址信息。地址表中的所有 MAC 地址都与一个或者多个端口关联。地址表包含以下几种类型的地址：

- 动态地址——设备学习到的源 MAC 地址，不使用时会被老化。
- 静态地址——手动输入的单播地址，不会老化且设备重置时不会丢弃。

地址表中列出了目的 MAC 地址，关联的 VLAN ID，与地址关联的端口号以及地址类型（静态或动态）。

注释：

注释： 有关本节使用命令的完整语法以及使用信息，参见此版本的命令参考手册。

MAC 地址表创建

所有端口上都支持有多个 MAC 地址，可以把设备端口连接到其他网络设备上。设备拥有动态学习地址的能力，可以学习每个端口收到数据包的源地址，并将学习到的地址以及关联的端口号添加到地址表中。在网中添加或移除设备时，设备会更新地址表，添加新的动态地址，并老化不使用的地址。

老化间隔是全局配置的。然而，设备会为每个 VLAN 维护一张地址表，且 STP 可以加速每个 VLAN 的老化间隔时间。

设备基于收到的数据包的目的地址，在任意端口组合之间发送数据包。通过使用 MAC 地址表，设备可以把数据包只转发到与目的地址关联的端口。如果目的地址在发送数据包的端口上，此数据包会被过滤而不被转发。设备总是会使用存储转发方式：在传输之前会存储完整的数据包并进行错误检查。

MAC 地址以及 VLAN

所有地址都与 VLAN 关联。一个地址可以存在于多个 VLAN 中，且在每个 VLAN 中可以有不同的目的端口。例如，单播地址数据包可以转发到 VLAN1 中的端口 1，以及 VLAN 5 中的端口 9、10 和 1 上。

每个 VLAN 都维护自己的逻辑地址表。一个 VLAN 中的已知地址在另一个 VLAN 中是未知的，除非在另一个 VLAN 中学习到或与某端口进行了静态关联。

MAC 地址以及设备堆栈

所有堆栈成员上的 MAC 地址表都是同步的。任意时刻，每个堆栈成员都有每个 VLAN 地址表的相同拷贝。地址老化后，会被从所有堆栈成员的地址表中移除。设备加入交换机堆栈时，该设备会接收到其他堆栈成员学习的每个 VLAN 的地址信息。当一个堆栈成员离开交换机堆栈时，其余堆栈成员会老化或移除通过之前的堆栈成员学习到的所有地址。

默认的 MAC 地址表设置

下表显示了 MAC 地址表的默认设置。

表 175：MAC 地址默认设置

特性	默认设置
老化时间	300 秒
动态地址	自动学习
静态地址	无配置

ARP 表管理

与一台设备进行通信时（如通过以太网通信），软件必须学习到该设备的 48 位 MAC 地址或者本地数据链路地址。这个通过 IP 地址学习本地数据链路地址的过程被称为*地址解析（address resolution）*。

地址解析协议（Address Resolution Protocol，ARP）会把 IP 地址与对应的介质或 MAC 地址以及 VLAN ID 进行关联。使用 IP 地址，ARP 查询关联的 MAC 地址。当找到 MAC 地址时，IP-MAC 地址关联会被存储在 ARP 缓存中，以便快速提取。随后 IP 数据报会被封装在数据链路

帧中，并通过网络发送。在以太网以外的 IEEE 802 网络上，IP 数据包封装以及 ARP 请求应答通过子网访问协议（Subnetwork Access Protocol，SNAP）进行了规范。默认情况下，在 IP 接口上启用标准以太网方式的 ARP 封装（由 **arpa** 关键字表示）。
手动添加到表中的 ARP 条目不会过期，且必须被手动移除。
相关 CLI 配置过程，参见 *icntnetworks.com* 上的 Inspur INOS 12.4 版文档。

如何管理设备

手动配置时间及日期

在重启过程之后，系统时间能保持准确，然而也可以在系统重启后手动配置时间以及日期。建议仅在必要时使用手工配置的方式。如果有设备可以同步的外部时间源，则无需手动设置系统时钟。
注释： 如果手动配置了时间，必须在活跃交换机故障且另一个堆栈成员接替了活跃交换机角色之前手动重新配置。

设置系统时钟

如果网络上有能提供时间服务的外部时间源，如 NTP 服务器，则无需手动设置系统时钟。按照以下步骤设置系统时钟：

总步骤

1. enable
2. 使用以下命令之一：
• **clock set** *hh:mm:ss day month year*
• **clock set** *hh:mm:ss month day year*

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式。在提示时输入密码。
步骤 2	使用以下命令之一： • **clock set** *hh:mm:ss day month year* • **clock set** *hh:mm:ss month day year* **示例：** `Device#` **clock set 13:32:00 23 March 2013**	使用以下一种格式设置系统时钟： • *hh:mm:ss* — — 按照小时、分钟一集秒的形式指定时间。指定的时间相对于配置的时区。 • *day*——指定一个月中的日期。 • *month*——指定月名称。 • *year*——指定年份（不缩写）。

配置时区

按照以下步骤手动配置时区。

总步骤

1. enable

2. configure terminal

3. clocktimezone*zone hours-offset* [*minutes-offset*]

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**clock timezone***zone hours-offset* [*minutes-offset*] **示例:** `Device(config)#` **clock timezone AST -3 30**	设置时区。 系统内部时间按照世界协调时间（UTC）记录，此命令仅用于在手动设置时间的时候展示时间。 • *zone*——输入标准时间生效时要显示的时区名。默认是 UTC。 • *hours-offset*——输入距 UTC 的偏移小时数。 • （可选）*minutes-offset*——输入距 UTC 的偏移分钟数。此选项在本地时区距 UTC 的时间是一小时的百分比时可用。
步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**show running-config** **示例:** `Device#` **show running-config**	验证条目。
步骤 6	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

配置夏令时

要在一些地区配置夏令时（日光节约时制）在每年特定某一周的某天起止，需执行以下操作：

总步骤

1. enable

2. configure terminal

3. clock summer-time *zone* **date** *date month year hh:mm date month year hh:mm*[*offset*]]

4. clock summer-time *zone* **recurring** [*week day month hh:mm week day month hh:mm*[*offset*]]

5. end

6. show running-config

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**clock summer-time** *zone* **date** *date month year* *hh:mm date month year hh:mm*[*offset*]] **示例:** `Device(config)#` **clock summer-time PDTdate10 March 2013 2:00 3 November 2013 2:00**	配置夏令时在每年特定的日期起止。
步骤 4	**clock summer-time** *zone* **recurring** [*week daymonth hh:mm week day month hh:mm*[*offset*]] **示例:** `Device(config)#` **clock summer-timePDT recurring 10 March 2013 2:00 3November 2013 2:00**	配置夏令时在每年特定的日期起止。所有时间都相对于本地时区。开始时间相对于标准时间。 结束时间相对于夏令时。夏令时默认被禁用。如果使用 **clock summer-time** *zone* **recurring** 且不指定参数，夏令时规则默认为美国规则。 如果开始月份在结束月份之后，系统假定用户位于南半球。 • *zone*——指定夏令时生效时显示的时区名称（如 PDT）。 • （可选）*week*——指定一个月中的一个星期（**1** 到 **4**，**first** 或 **last**）。 • （可选）*day*——指定一周中的

		一天（Sunday、Monday 等） • （可选）*month*——指定月份（January、February 等） • （可选）*hh:mm*——按小时和分钟指定时间。 • （可选）*offset*——指定夏令时期间要增加的分钟数。默认值是 60 分钟。
步骤 5	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 6	**show running-config** **示例:** `Device#` **show running-config**	验证条目。
步骤 7	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

如果所处地区的夏令时不是循环出现的，按照以下步骤进行配置（配置下一个夏令时的确切日期与时间）：

总步骤

1. enable

2. configure terminal

3. clock summer-time *zone* **date**[*month date year hh:mm month date year hh:mm*[*offset*]]or**clocksummer-time** *zone* **date** [*date month year hh:mm date month year hh:mm*[*offset*]]

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**clock summer-time** *zone* **date**[*month date yearhh:mm month date year hh:mm*[*offset*]]or**clocksummer-time** *zone*	配置夏令时在第一个日期开始，在第二个日期结束。 夏令时默认被禁用。

	date [*date month year hh:mmdate month year hh:mm*[*offset*]]	• 使用 *zone* 指定夏令时生效时显示的时区名（如 PDT）。 • （可选）使用 *week* 指定一个月中的一周（1 到 5 或 last）。 • （可选）使用 *day* 指定一周中的一天（Sunday、Monday 等）。 • （可选）使用 *month* 指定月份（January、February 等） • （可选）使用 *hh:mm* 按小时和分钟指定时间。 • （可选）使用 *offset* 指定夏令时期间要增加的分钟数。默认值是 60 分钟。
步骤 4	**end** **示例:** `Device(config)# end`	返回特权 EXEC 模式。
步骤 5	**show running-config** **示例:** `Device# show running-config`	验证条目。
步骤 6	**copy running-config startup-config** **示例:** `Device# copy running-config startup-config`	（可选）将条目保存在设备启动配置文件中。

配置系统名称

按照以下步骤配置系统名称：

总步骤

1. enable
2. configure terminal
3. hostname*name*
4. end
5. show running-config
6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例:** `Device>enable`	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal**	进入全局配置模式。

	示例: `Device#` **configure terminal**	
步骤 3	**hostname** *name* **Example:** `Device(config)#` **hostname remote-users**	配置系统名称。设置的系统名也会被用作系统提示符。 默认设置是 Device。 名称必须遵循 ARPANET 主机名规则，必须以字母开始，以字母或数字结束，且之间的字符只能是字母、数字或连字符。名称至多可以有 64 个字符。
步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**show running-config** **示例:** `Device#` **show running-config**	验证条目。
步骤 6	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

设置 DNS

如果使用设备 IP 地址作为主机名，IP 地址会被使用且不进行 DNS 查询。如果配置的主机名不包含英文句号（.），英文句号以及默认域名会被附加到主机名之后，然后进行 DNS 查询，将名称映射为 IP 地址。默认域名值由全局配置命令 **ip domain-name** 设置。如果域名中有英文句号（.），Inspur INOS 软件会查询 IP 地址且不会给主机名附加默认域名。
按照以下步骤设置交换机使用 DNS：

总步骤

1. enable
2. configure terminal
3. ip domain-name *name*
4. ip name-server *server-address1* [*server-address2 ... server-address6*]
5. ip domain-lookup [**nsap**| **source-interface** *interface*]
6. end
7. show running-config
8. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例:**	进入特权 EXEC 模式。在提示时输入密码。

	Device>**enable**	
步骤 2	**configure terminal** **示例:** Device# **configure terminal**	进入全局配置模式。
步骤 3	**ip domain-name** *name* **示例:** Device(config)# **ipdomainnameicntnetworks.com**	指定默认域名，系统使用该域名补全不合规的主机名（没有点分十进制域名的名称）。 不要包含分隔不合规名称与域名的句点。 设备启动时，无域名配置，然而如果设备通过 BOOTP 或动态主机配置协议（DHCP）服务器配置，可以使用 BOOTP 或 DHCP 服务器设置默认域名（如果服务器配置了此信息）。
步骤 4	**ip name-server** *server-address1*[*server-address2 ... server-address6*] **示例:** Device(config)# **ipname-server** **192.168.1.100192.168.1.200 192.168.1.300**	指定名称地址解析使用的一个或多个名称服务器。 可以指定至多六个名称服务器。每个服务器地址之间用空格分开。指定的第一个服务器是主服务器。设备会先给主服务器发送 DNS 查询。如果查询失败，会查询备用服务器。
步骤 5	**ip domain-lookup** [**nsap**| **source-interface***interface*] **示例:** Device(config)# **ip domain-lookup**	（可选）在设备上启用基于 DNS 的名称-地址翻译。该特性默认被启用。 如果网络设备要求与不能控制名称分配的网络设备有连通性，可以使用全局 Internet 命名机制（DNS）给设备动态分配唯一标识设备的名称。
步骤 6	**end** **示例:** Device(config)# **end**	返回特权 EXEC 模式。
步骤 7	**show running-config** **示例:** Device# **show running-config**	验证条目。
步骤 8	**copy running-config startup-config** **示例:** Device# **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

接下来做什么？

配置当日消息登录标语

可以配置单行或多行的消息标语，有人登录设备时会在屏幕上显示。
按照以下步骤配置 MOTD 登录标语：

总步骤

1. **enable**
2. **configure terminal**
3. **bannermotd***c message c*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。

步骤 3	**banner motd***c message c*
	示例:
	`Device(config)#` **banner motd #**
	`This is a secure site. Only authorized users are allowed. For access, contact technical support.`
	`#`
		指定当日消息。
		c——输入定界字符，如井号（#），并输入回车键。定界字符标示了标语文本的开始和结束。结束定界符之后的字符会被丢弃。
		message——输入标语消息，至多 255 字符。消息中不能使用定界字符。

步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**show running-config** **示例:** `Device#` **show running-config**	验证条目。
步骤 6	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

配置登录标语

可以配置在所有连接的终端上显示的登录标语。此标语会在 MOTD 标语之后登录提示符之前显示。

按照以下步骤配置登录标语：

总步骤

1. **enable**
2. **configure terminal**
3. **banner login** *c message c*
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>enable`	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device# configure terminal`	进入全局配置模式。

步骤 3	**banner login** *c message c*
	示例
	`Device(config)# banner login $`
	`Access for authorized users only.`
	`Please enter your username and password.`
	`$`
		指定登录消息。
		c——输入定界字符，如井号（#），并输入回车键。定界字符标示了标语文本的开始和结束。结束定界符之后的字符会被丢弃。
		message——输入标语消息，至多 255 字符。消息中不能使用定界字符。

步骤 4	**end** **示例:** `Device(config)# end`	返回特权 EXEC 模式。
步骤 5	**show running-config** **示例:** `Device# show running-config`	验证条目。
步骤 6	**copy running-config startup-config** **示例:** `Device# copy running-config startup-config`	（可选）将条目保存在设备启动配置文件中。

管理 MAC 地址表

更改地址老化时间

按照以下步骤配置动态地址表老化时间：

总步骤

1. enable

2. configure terminal

3. mac address-table aging-time [*0* | *10-1000000*] [**routed-mac** | **vlan***vlan-id*]

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**mac address-table aging-time** [*0* | *10-1000000*][**routed-mac** | **vlan***vlan-id*] **示例:** `Device(config)#` **mac address-tableaging-time 500 vlan 2**	设置 MAC 地址表中条目使用或更新之后动态条目保留的时长。 范围从 10 到 1000000 秒。默认值是 300 秒，也可以输入 0 来禁用老化。静态地址条目不会被老化也不会被从表中移除。 *vlan-id*——合法 ID 是 1 到 4094。
步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**show running-config** **示例:** `Device#` **show running-config**	验证条目。
步骤 6	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

配置 MAC 地址更改通知陷阱

按照以下步骤配置交换机给 NMS 主机发送 MAC 地址更改通知陷阱：

总步骤

1. enable

2. configure terminal

3. snmp-server host *host-addr community-string notification-type* { **informs**| **traps** } {**version** {**1** | **2c** | **3**}}{**vrf***vrf instance name*}

4. snmp-server enable traps mac-notification change

5. mac address-table notification change

6. mac address-table notification change [**interval** *value*] [**history-size** *value*]

7. interface*interface-id*

8. snmp trap mac-notification change {**added** | **removed**}

9. end

10. show running-config

11. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**snmp-server host** *host-addr community-stringnotification-type* { **informs** | **traps** } {**version** {**1**| **2c** | **3**}} {**vrf***vrf instance name*} **示例:** `Device(config)#` **snmp-server host172.20.10.10 traps private mac-notification**	指定陷阱消息的接收者。 • *host-addr*——指定 NMS 的名称或地址。 • **traps**（默认设置）——把 SNMP 陷阱发送给主机。 • **informs**——把 SNMP 通知发送给主机。 • **version**——指定支持的 SNMP 版本。默认是版本 1，该版本不可使用通知。 • *community-string*——指定通知操作发送的字符串。虽然可以使用 **snmp-server host** 命令设置此字符串，建议在使用 **snmp-server host** 命令之前使用 **snmp-server community** 命令定义此字符串。 • *notification-type*——使用 **mac-notification** 关键字。 • **vrf***vrfinstance name*——指定主机的 VPN 路由/转发实例。
步骤 4	**snmp-server enable traps mac-notificationchange** **示例:** `Device(config)#` **snmp-server enable**	让设备给 NMS 发送 MAC 地址更改通知陷阱。

	trapsmac-notification change	
步骤 5	**mac address-table notification change** **示例:** `Device(config)#` **mac address-tablenotification change**	启用 MAC 地址更改通知特性。
步骤 6	**mac address-table notification change** [**interval***value*] [**history-size** *value*] **示例:** `Device(config)#` **mac address-tablenotification change interval 123** `Device(config)#`**mac address-tablenotification change history-size 100**	输入陷阱间隔时间以及历史表大小。 • （可选）**interval** *value*——指定生成给 NMS 的每组通知陷阱的间隔秒数。范围从 0 到 2147483647 秒，默认是 1 秒。 • （可选）**history-size** *value*——指定 MAC 通知历史表的最大条目数量。范围从 0 到 500，默认值是 1。
步骤 7	**interface** *interface-id* **示例:** `Device(config)#` **interfacegigabitethernet1/0/2**	进入接口配置模式，指定要启用 SNMP MAC 地址通知陷阱的二层接口。
步骤 8	**snmp trap mac-notification change {added |removed}** **示例:** `Device(config-if)#` **snmp trapmac-notification change added**	在接口上启用 MAC 地址更改通知陷阱。 • 当 MAC 地址 **added**（添加）到接口时启用陷阱。 • 当 MAC 地址从接口 **removed**（移除）时启用陷阱。
步骤 9	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 10	**show running-config** **示例:** `Device#` **show running-config**	验证条目。
步骤 11	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

配置 MAC 地址移动通知陷阱

配置 MAC 移动通知后，当一个 MAC 地址从一个端口移动到相同 VLAN 的另一个端口时会生成 SNMP 通知并发往网络管理系统。

按照以下步骤配置设备给 NMS 主机发送 MAC 地址移动通知陷阱：

总步骤

1. enable

2. configure terminal

3. snmp-server host *host-addr*{**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*

4. snmp-server enable traps mac-notification move

5. mac address-table notification mac-move

6. end

7. show running-config

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**snmp-server host** *host-addr*{**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-stringnotification-type* **示例:** `Device(config)#` **snmp-server host 172.20.10.10 traps private mac-notification**	指定陷阱消息的接收者。 • *host-addr*——指定 NMS 的名称或地址。 • **traps**（默认设置）——把 SNMP 陷阱发送给主机。 • **informs**——把 SNMP 通知发送给主机。 • **version**——指定支持的 SNMP 版本。默认是版本 1，该版本不可使用通知。 • *community-string*——指定通知操作发送的字符串。虽然可以使用 **snmp-server host** 命令设置此字符串，建议在使用 **snmp-server host** 命令之前使用 **snmp-server community** 命令定义此字符串。 • *notification-type*——使用 **mac-notification** 关键字。
步骤 4	**snmp-server enable traps mac-notification move** **示例:** `Device(config)#` **snmp-server enable traps mac-notification move**	让设备给 NMS 发送 MAC 地址移动通知陷阱。
步骤 5	**mac address-table notification mac-move** **示例:**	启用 MAC 地址移动通知特性。

	Device(config)# **mac address-tablenotification mac-move**	
步骤 6	**end** **示例:** Device(config)# **end**	返回特权 EXEC 模式。
步骤 7	**show running-config** **示例:** Device# **show running-config**	验证条目。
步骤 8	**copy running-config startup-config** **示例:** Device# **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

接下来做什么？

要禁用MAC地址移动通知陷阱，需使用全局配置命令**no snmp-server enable traps mac-notificationmove**。要禁用MAC地址移动通知特性，需使用全局配置命令**no macaddress-table notification mac-move**。

可以输入特权EXEC命令**show mac address-table notification mac-move**验证设置。

配置MAC门限值通知陷阱

配置 MAC 门限值通知后，当达到或超过 MAC 地址表门限值时会生成 SNMP 通知并发送给网络管理系统。

按照以下步骤配置交换机给 NMS 主机发送 MAC 地址表门限值通知陷阱：

总步骤

1. enable

2. configure terminal

3. snmp-server host *host-addr*{**traps** | **informs**} {**version** {**1** | **2c** | **3**}} *community-string notification-type*

4. snmp-server enable traps mac-notification threshold

5. mac address-table notification threshold

6. mac address-table notification threshold [**limit** *percentage*] | [**interval** *time*]

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例:** Device>**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** Device# **configure terminal**	进入全局配置模式。

步骤 3	**snmp-server host** *host-addr*{**traps** | **informs**}{**version** {**1** | **2c** | **3**}} *community-stringnotification-type* **示例:** `Device(config)#` **snmp-server host 172.20.10.10 traps privatemac-notification**	指定陷阱消息的接收者。 • *host-addr*——指定 NMS 的名称或地址。 • **traps**（默认设置）——把 SNMP 陷阱发送给主机。 • **informs**——把 SNMP 通知发送给主机。 • **version**——指定支持的 SNMP 版本。默认是版本 1，该版本不可使用通知。 • *community-string*——指定通知操作发送的字符串。虽然可以使用 **snmp-server host** 命令设置此字符串，建议在使用 **snmp-server host** 命令之前使用 **snmp-server community** 命令定义此字符串。 • *notification-type*——使用 **mac-notification** 关键字。
步骤 4	**snmp-server enable traps mac-notificationthreshold** **示例:** `Device(config)#` **snmp-server enable trapsmac-notification threshold**	让设备给 NMS 发送 MAC 门限值通知陷阱。
步骤 5	**mac address-table notification threshold** **示例:** `Device(config)#` **mac address-table notification threshold**	启用 MAC 地址门限值通知特性。
步骤 6	**mac address-table notification threshold** [**limit***percentage*] | [**interval** *time*] **示例:** `Device(config)#` **mac address-tablenotification threshold interval 123** `Device(config)#` **mac address-tablenotification threshold limit 78**	输入 MAC 地址门限值使用监控的门限值。 • （可选）**limit** *percentage*——指定 MAC 地址表使用的百分比，合法值范围从 1 到 100，默认是 50%。 • （可选）**interval** *time*——指定通知间隔，合法值应大于或等于 120 秒。默认值是 120 秒。
步骤 7	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 8	**show running-config** **示例:** `Device#` **show running-config**	验证条目。
步骤 9	**copy running-config startup-config**	（可选）将条目保存在设备启动配

	示例: `Device#` **copy running-config startup-config**	置文件中。

接下来做什么？

添加或移除静态地址条目

按照以下步骤添加静态地址：

总步骤

1. enable

2. configure terminal

3. mac address-table static *mac-addr* **vlan** *vlan-id* **interface** *interface-id*

4. end

5. show running-config

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**mac address-table static** *mac-addr* **vlan** *vlan-id* **interface** *interface-id* **示例:** `Device(config)#` **mac address-tablestatic c2f3.220a.12f4 vlan 4 interfacegigabitethernet 1/0/1**	向 MAC 地址表添加静态地址。 • *mac-addr*——指定要添加到地址表的目的 MAC 单播地址。在指定 VLAN 中接收的使用该目的地址的数据包会被转发到指定的接口。 • *vlan-id*——指定接收指定 MAC 地址数据包的 VLAN。合法 VLAN ID 从 1 到 4094。 • *interface-id*——指定收到的数据包要被转发到的接口。合法的接口包括物理端口或端口通道。对于静态组播地址，可以输入多个接口 ID。对于静态单播地址，一次仅可以输入一个接口，但是可以多次输入相同 MAC 地址以及 VLAN ID 的命令。

步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。此外，也可以按下 **Ctrl-Z** 退出全局配置模式。
步骤 5	**show running-config** **示例:** `Device#` **show running-config**	验证条目。
步骤 6	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

配置单播 MAC 地址过滤

按照以下步骤配置设备丢弃源或目的单播静态地址：

总步骤

1. **enable**
2. **configure terminal**
3. **mac address-table static** *mac-addr* **vlan** *vlan-id* **drop**
4. **end**
5. **show running-config**
6. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**mac address-table static** *mac-addr* **vlan** *vlan-id* **drop** **示例:** `Device(config)#` **mac address-tablestatic c2f3.220a.12f4 vlan 4 drop**	启用单播 MAC 地址过滤，并配置设备丢弃使用指定源或目的单播静态地址的数据包。 • *mac-addr*——指定源或目的单播 MAC 地址（48 位）。使用此 MAC 地址的数据包会被丢弃。 • *vlan-id*——指定接收带有指定 MAC 地址数据包的 VLAN。合法 VLAN ID 从 1 到 4094。
步骤 4	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**show running-config**	验证条目。

	示例: `Device#` **show running-config**	
步骤 6	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

监控并维护设备管理

命令	目的
clear mac address-table dynamic	移除所有动态条目。
clear mac address-table dynamic address*mac-address*	移除指定的 MAC 地址。
clear mac address-table dynamic interface*interface-id*	移除指定物理端口或端口通道上的所有地址。
clear mac address-table dynamic vlan*vlan-id*	移除指定 VLAN 上的所有地址。
show clock [*detail*]	显示时间及日期配置。
show ipigmp snooping groups	显示所有 VLAN 或指定 VLAN 的二层组播条目。
show mac address-table address *mac-address*	显示指定 MAC 地址的 MAC 地址表信息。
show mac address-table aging-time	显示所有 VLAN 或指定 VLAN 中的老化时间。
show mac address-table count	显示所有 VLAN 或指定 VLAN 中的地址数量。
show mac address-table dynamic	仅显示动态 MAC 地址表条目。
show mac address-table interface *interface-name*	显示指定接口的 MAC 地址表信息。
show mac address-table move update	显示 MAC 地址表移动更新信息。
show mac address-table multicast	显示组播 MAC 地址列表。
show mac address-table notification {change |mac-move | threshold}	显示 MAC 通知参数及历史表。
show mac address-table secure	显示安全 MAC 地址。
show mac address-table static	仅显示静态 MAC 地址表条目。
show mac address-table vlan*vlan-id*	显示指定 VLAN 的 MAC 地址表信息。

设备管理的配置示例

示例：设置系统时钟

此示例展示了如何手动设置系统时钟：

```
Device# clock set 13:32:00 23 July 2013
```

示例：配置夏令时

此示例展示了如何指定从 3 月 10 日 02:00 开始并在 11 月 3 日 02:00 结束的夏令时：

```
Device(config)# clock summer-time PDT recurring PST date
10 March 2013 2:00 3 November 2013 2:00
```

This example shows how to set summer time start and end dates:

```
Device(config)#clock summer-time PST date
20 March 2013 2:00 20 November 2013 2:00
```

示例：配置 MOTD 标语

此示例展示了如何配置 MOTD 标语并使用井号（#）作为起始定界字符：

```
Device(config)# banner motd #
This is a secure site. Only authorized users are allowed.
For access, contact technical support.
#
Device(config)#
```

此示例展示了以上配置显示的标语：

```
Unix>telnet 192.0.2.15
Trying 192.0.2.15...
Connected to 192.0.2.15.
Escape character is '^]'.
This is a secure site. Only authorized users are allowed.
For access, contact technical support.
User Access Verification
Password:
```

示例：配置登录标语

此示例展示了如何配置登录标语并使用井号（#）作为起始定界字符：

```
Device(config)# banner login $
Access for authorized users only. Please enter your username and password.
$
Device(config)#
```

示例：配置 MAC 地址更改通知陷阱

此示例展示了如何指定 172.20.10.10 作为 NMS，启用发往 NMS 的 MAC 地址通知陷阱，启用 MAC 地址更改通知特性，设置时间间隔为 123 秒，设置历史大小为 100 个条目，并在 MAC 地址添加到指定端口时启用陷阱：

```
Device(config)# snmp-server host 172.20.10.10 traps private mac-notification
Device(config)# snmp-server enable traps mac-notification change
Device(config)# mac address-table notification change
Device(config)# mac address-table notification change interval 123
Device(config)# mac address-table notification change history-size 100
Device(config)# interface gigabitethernet1/2/1
Device(config-if)# snmp trap mac-notification change added
```

示例：配置 MAC 门限值通知陷阱

此示例展示了如何指定 172.20.10.10 作为 NMS，启用 MAC 地址门限值通知特性，设置时间间隔为 123 秒，并设置限制为 78%：

```
Device(config)# snmp-server host 172.20.10.10 traps private mac-notification
Device(config)# snmp-server enable traps mac-notification threshold
Device(config)# mac address-table notification threshold
Device(config)# mac address-table notification threshold interval 123
Device(config)# mac address-table notification threshold limit 78
```

示例：向 MAC 地址表添加静态地址

此示例展示了如何把静态地址 c2f3.220a.12f4 添加到 MAC 地址表。在 VLAN 4 中收到使用此 MAC 地址作为目的地址的数据包时，数据包会被转发到指定的端口：

```
Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 interface gigabitethernet1/1/1
```

示例：配置单播 MAC 地址过滤

此示例展示了如何启用单播 MAC 地址过滤以及如何配置交换机丢弃源或目的地址为 c2f3.220a.12f4 的数据包。在 VLAN 4 上收到源或目的地址为此 MAC 地址的数据包时，数据包会被丢弃：

```
Device(config)# mac address-table static c2f3.220a.12f4 vlan 4 drop
```

其他参考资料

相关文档

相关主题	文档标题
系统管理命令	*系统管理命令参考（Inspur 6650 交换机）*
网络管理配置	*网络管理配置指南（Inspur 6650 交换机）*
二层配置	*2/三层配置指南（Inspur 6650 交换机）*
VLAN 配置	*VLAN 配置指南（Inspur6650 交换机）*
平台无关的命令参考	*配置基础命令参考，Inspur INOS（Inspur 3850 交换机）*
平台无关的配置信息	*配置基础配置指南，Inspur INOS（Inspur 3850 交换机）* *IP 编址配置指南库，Inspur INOS（Inspur 3850 交换机）*

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

设备管理的特性历史与信息

版本	修订
Inspur INOS 12.2	引入了此特性。

引导完整性可视化

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用 Inspur 特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问 Inspur 特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于引导完整性可视化的信息

引导完整性可视化让 Inspur 的平台身份以及软件完整性信息变得可见且可操作。平台身份提供了平台的制造安装身份，而软件完整性显示了引导完整性度量信息，可以被用来评估平

台是否引导了可信的代码。在启动过程中，软件会在引导程序活动的每个阶段创建校验和记录。可以获取此记录并与 Inspur 认证的记录进行比较，判断软件镜像是否真实。如果校验和值不相同，则用户可能运行了未被 Inspur 认证的软件镜像，或是镜像被未授权的机构修改过。

验证软件镜像及硬件

此任务描述了如何获取交换机启动期间创建的校验和记录。在特权 EXEC 模式中输入以下命令：

注释： 在执行以下命令时，管理员可能会看到 CLI 上显示消息**% Please Try After Few Seconds**。这不表示 CLI 出错，而是表明正在设置所需的底层基础设施来获取所需的输出。建议等待几分钟再重试此命令。

消息**% Error retrieving SUDI certificate** and **% Error retrieving integrity data** 表示真正的 CLI 出错情况。

总步骤

1. **show platform sudi certificate** [**sign** [**nonce** *nonce*]]
2. **show platform integrity** [**sign** [**nonce** *nonce*]]

具体步骤

	命令或操作	目的
步骤 1	**show platform sudi certificate** [**sign** [**nonce** *nonce*]] **示例：** `Device#` **show platform sudi certificate signnonce 123**	显示指定 SUDI 的校验和记录。 • （可选）**sign**——显示签名 • （可选）**nonce**——输入随机值
步骤 2	**show platform integrity** [**sign** [**nonce** *nonce*]] **示例：** `Device#` **show platform integrity sign nonce 123**	显示启动阶段的校验和记录。 • （可选）**sign**——显示签名 • （可选）**nonce**——输入随机值

验证平台身份及软件完整性

验证平台身份

以下示例展示了 PEM 格式的安全唯一设备标识（Secure Unique Device Identity，SUDI）。第一个证书是 Inspur 根 CA 2048，第二个是 Inspur 下属 CA（ACT2 SUDI CA）。两个证书都可以在 http://www.icntnetworks.com 上进行验证。第三个是 SUDI 证书。

```
Device#show platform sudi certificate sign nonce 123
-----BEGIN CERTIFICATE-----
```

MIIDQzCCAiugAwIBAgIQX/h7KCtU3I1CoxW1aMmt/zANBgkqhkiG9w0BAQUFADA1
MRYwFAYDVQQKEw1DaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENB
IDIwNDgwHhcNMDQwNTE0MjAxNzEyWhcNMjkwNTE0MjAyNTQyWjA1MRYwFAYDVQQK
Ew1DaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENBIDIwNDgwggEg
MA0GCSqGSIb3DQEBAQUAA4IBDQAwggEIAoIBAQCwmrmrp68Kd6ficba0ZmKUeIhH
xmJVhEAyv8CrLqUccda8bnuoqrpu0hWISEWdovyD0My5jOAmaHBKeN8hF570YQXJ
FcjPFto1YYmUQ6iEqDGYeJu5Tm8sUxJszR2tKyS7McQr/4NEb7Y9JHcJ6r8qqB9q
VvYgDxFUl4F1pyXOWWqCZe+36ufijXWLbvLdT6ZeYpzPEApk0E5tzivMW/VgpSdH
jWn0f84bcN5wGyDWbs2mAag8EtKpP6BrXruOIIt6keO1aO6g58QBdKhTCytKmg9l
Eg6CTY5j/e/rmxrbU6YTYK/CfdfHbBcl1HP7R2RQgYCUTOG/rksc35LtLgXfAgED
o1EwTzALBgNVHQ8EBAMCAYYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUJ/PI
FR5umgIJFq0roIlgX9p7L6owEAYJKwYBBAGCNxUBBAMCAQAwDQYJKoZIhvcNAQEF
BQADggEBAJ2dhISjQal8dwy3U8pORFBi71R803UXHOjgxkhLtv5MOhmBVrBW7hmW
Yqpao2TB9k5UM8Z3/sUcuuVdJcr18JOagxEu5sv4dEX+5wW4q+ffy0vhN4TauYuX
cB7w4ovXsNgOnbFp1iqRe6lJT37mjpXYgyc81WhJDtSd9i7rp77rMKSsH0T8lasz
Bvt9YAretIpjsJyp8qS5UwGH0GikJ3+r/+n6yUA4iGe0OcaEb1fJU9u6ju7AQ7L4
CYNu/2bPPu8Xs1gYJQk0XuPL1hS27PKSb3TkL4Eq1ZKR4OCXPDJoBYVL0fdX4lId
kxpUnwVwwEpxYB5DC2Ae/qPOgRnhCzU=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIEPDCCAySgAwIBAgIKYQlufQAAAAAADDANBgkqhkiG9w0BAQUFADA1MRYwFAYD
VQQKEw1DaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENBIDIwNDgw
HhcNMTEwNjMwMTc1NjU3WhcNMjkwNTE0MjAyNTQyWjAnMQ4wDAYDVQQKEwVDaXNj
bzEVMBMGA1UEAxMMQUNUMiBTVURJIENBMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8A
MIIBCgKCAQEA0m5l3THIxA9tN/hS5qR/6UZRpdd+9aE2JbFkNjht6gfHKd477AkS
5XAtUs5oxDYVt/zEbslZq3+LR6qrqKKQVu6JYvH05UYLBqCj38s76NLk53905Wzp
9pRcmRCPuX+a6tHF/qRuOiJ44mdeDYZo3qPCpxzprWJDPclM4iYKHumMQMqmgmg+
xghHIooWS80BOcdiynEbeP5rZ7qRuewKMpl1TiI3WdBNjZjnpfjg66F+P4SaDkGb
BXdGj13oVeF+EyFWLrFjj97fL2+8oauV43Qrvnf3d/GfqXj7ew+z/sXlXtEOjSXJ
URsyMEj53Rdd9tJwHky8neapszS+r+kdVQIDAQABo4IBWjCCAVYwCwYDVR0PBAQD
AgHGMB0GA1UdDgQWBBRI2PHxwnDVW7t8cwmTr7i4MAP4fzAfBgNVHSMEGDAWgBQn
88gVHm6aAgkWrSugiWBf2nsvqjBDBgNVHR8EPDA6MDigNqA0hjJodHRwOi8vd3d3
LmNpc2NvLmNvbS9zZWN1cml0eS9wa2kvY3JsL2NyY2EyMDQ4LmNybDBQBggrBgEF
BQcBAQREMEIwQAYIKwYBBQUHMAKGNGh0dHA6Ly93d3cuY2lzY28uY29tL3NlY3Vy
aXR5L3BraS9jZXJ0cy9jcmNhMjA0OC5jZXIwXAYDVR0gBFUwUzBRBgorBgEEAQkV
AQwAMEMwQQYIKwYBBQUHAgEWNWh0dHA6Ly93d3cuY2lzY28uY29tL3NlY3VyaXR5
L3BraS9wb2xpY2llcy9pbmRleC5odG1sMBIGA1UdEwEB/wQIMAYBAf8CAQAwDQYJ
KoZIhvcNAQEFBQADggEBAGh1qclr9tx4hzWgDERm371yeuEmqcIfi9b9+GbMSJbi
ZHc/CcC10lJu0a9zTXA9w47H9/t6leduGxb4WeLxcwCiUgvFtCa51Iklt8nNbcKY
/4dw1ex+7amATUQO4QggIE67wVIPu6bgAE3Ja/nRS3xKYSnj8H5TehimBSv6TECi
i5jUhOWryAK4dVo8hCjkjEkzu3ufBTJapnv89g9OE+H3VKM4L+/KdkUO+52djFKn
hyl47d7cZR4DY4LIuFM2P1As8YyjzoNpK/urSRI14WdIlplR1nH7KNDl5618yfVP
0IFJZBGrooCRBjOSwFv8cpWCbmWdPaCQT2nwIjTfY8c=
-----END CERTIFICATE-----

```
-----BEGIN CERTIFICATE-----
MIIDhzCCAm+gAwIBAgIEAJT3DDANBgkqhkiG9w0BAQsFADAnMQ4wDAYDVQQKEwVD
aXNjbzEVMBMGA1UEAxMMQUNUMiBTVURJIENBMB4XDTE1MTExNDA5MzMzN1oXDTI1
MTExNDA5MzMzN1owczEsMCoGA1UEBRMjUElEOldTLUMzNjUwLTEyWDQ4VVEgU046
RkRPMTk0NkJHMDUxDjAMBgNVBAoTBUNpc2NvMRgwFgYDVQQLEw9BQ1QtMiBMaXRl
IFNVREkxGTAXBgNVBAMTEFdTLUMzNjUwLTEyWDQ4VVEwggEiMA0GCSqGSIb3DQEB
AQUAA4IBDwAwggEKAoIBAQC6SARWyImWrRV/x7XQogAE+02WmzKki+4arMVBvl9o
GgvJfkoJDdaHOROSUkEE3qXtd8N3lfKy3TZ+jtHD85m2aGz6+IRx/e/lLsQzi6dl
WIB+N94pgecFBONPR9wJriox1IGD3B43b0hMLkmro4R5Zrs8XFkDo9k1tBU7F2O7
GEzb/WkO5NLexznef2Niglx9fCDL0HC27BbsR5+03p8jhG0+mvrp8M9du1HKiGin
ZIV4XgTMp1/k/TVaIepEGZuWM3hxdUZjkNGG1c1m+oB8vLX3UlSL76sDBBoiaprD
rjXBgBIozyFW8tTjh50jMDG84hKD5s31ifOe4KpqEcnVAgMBAAGjbzBtMA4GA1Ud
DwEB/wQEAwIF4DAMBgNVHRMBAf8EAjAAME0GA1UdEQRGMESgQgYJKwYBBAEJFQID
oDUTM0NoaXBJRD1VWUpOTlZJMENBUkhVM1Z1SUVSbFl5QXlPQ0F4TXpvek5Ub3lN
U0EwS0NnPTANBgkqhkiG9w0BAQsFAAOCAQEADjtM8vdlf+p1WKSKX1C1qQ4aEnD5
p8T5e4iTer7Y1fbCrHIEEm3mnip+568j299z0H8V7PDp1ljuLHyMFTC+945F9RfA
eAuVWVb5A9dnGL8MssBJe2lVSnZwrWkT1EIdxLYrTiPAQHtll6CN77S4u/f71oYE
tzPE5AGfyGw7ro1MEPVGffaQmYUDAwKFNBH1uI7c2S1qlwk4WWZ6xxci+lhaQnIG
pWzapaiAYL1XrcBz4KwFc1ZZpQT6hHw24jzYaYimvCo+/kSKuA9xNdtSu18ycox0
zKnXQ17s6aChMMt7Y8Nh4iz9BDejoOF6/b3sM0wRi+2/4j+6/GhcMRs0Og==
-----END CERTIFICATE-----
Signature version: 1
Signature:
405C770D802B73947EDBF8DD0D2C8180F10D4B3EF9699444514219C579D2ED52F7D5
83E0F4408133FC4E9F549B2EB1C21725F7CB1C79F98271E47E780E703E674723880F
B52D4963E1D1FB9787B38E28B8E696570A180B7A2F1311B1F174EAA79F55DB4765DF
67386126D899E07EDF6C26E0A81272EAA114437DD03F26992937082756AE1F1BFAFB
BFACD6BE9CF9C84C961FACE9FA0FEE64D85AE4FA0086969D0702C536ABDB8FBFDC47
C14C17D02FEBF4F7F5BB24D2932FA876F56B4C07816270AA0B4195C53D975C85AEAE
3A74F2DBF293F52423ECB7B8539667080A9C57DA3E4B08B2B2CA623B2CBAF7080A0A
EB09B222E5B756970A3AA27E0F1D17C8A243
```

可以对三个证书、签名版本以及用户提供的随机数值进行 RSA 2048 签名。

```
RSA PKCS#1v1.5 Sign {<Nonce (UINT64)> || <Signature Version (UINT32)> || <Inspur Root
CA2048 cert (DER)> ||<Inspur subordinate CA (DER)> || <SUDI certificate (DER)> }
```

Inspur 管理解决方案能够解释以上输出。不过也可以通过使用 OpenSSL 命令的简单脚本来显示平台身份并验证签名，进而确认 Inspur 唯一设备身份。

```
[linux-host:~]openssl x509 -in sudicert.pem -subject -noout
subject= /serialNumber=PID:WS-C6650-12X48UQ SN:FDO1946BG05/O=Inspur/OU=ACT-2
Lite SUDI/CN=WS-C6650-12X48UQ
```

验证软件完整性

以下示例展示了启动阶段的校验和值。对于软件成功启动的三个阶段中的每个阶段都会显示一个哈希度量值。这些哈希值可以用来与 Inspur 提供的参考值进行比较。可以选择对输出进行签名，这让验证者可以确认这些输出是真实且未被修改的。可以提供随机数来对抗重放攻

击。

```
Device #show platform integrity sign nonce 456
Platform: WS-C6650-12X48UQ
Boot Loader Version: CAT3K_CAA Boot Loader (CAT3K_CAA-HBOOT-M) Version 4.16,
engineering
software (D)
Boot Loader Hash: DB5A686E9F4CE358481DE3AF8B9C762F0A604E3B4764DF2A351F176E3D7
D3C60EB85C02906BD8CF28228C0DFC2AA8960CAFE6675D696E4ABA0CD687C0609E7E2
Boot 0 Version: F01062R15.0508d68fa2015-09-15
Boot 0 Hash: 6EF15CD54D3C66A8B644194A67B7ED57044C8C2E0EECB69736A7FFEC1F6D0EAD
OS Version: 2016-10-18_10.57_mundru
OS Hash: 4C85AECC88DAA49D940BBF65B1F17269F55C8D98DEFB4140F981923AA961140293E1
3B3E6E68CE3F8ED7F596CD858ACDD4BEF6538F59C1E243C351353026E6CD
PCR0: 90214167AAF35C06B2AC97292596E5669EAB72578FCDAD0B91746683BAA7B2B0
PCR8: FC2CE1BAC397F97008936DF372A2218BB16A798222B8FF55A7B6AEDA8018EDF5
Signature version: 1
Signature:
632A724F1AB6ADE134F6B0E8724D2052B3157F45B47E547763EE224A848E807CD737600587FF68
2526A8FE354A116CC9EDEBD9C659B9927336542EE4295084368327D01BD22AB4849BB3C007B6EB
B67708685FD6BC85DD045431E19A389FEB358894D4FBCF7C0FC960AC9133B61099DFD507F316C1
BF82F7F98687C7E7E8F99355DC1A95BD511B0B8DCB0CA909828F9EFBDF18847930392A8E3D072D
F3D90536880BAE9B7D7CF0E301D3F5AF16E7517FC2700E2F75911B836D6559A18E15B4CF452555
91656DF22DFF73392F777AEB796BCF9AC046C581ADEF19CA48A98F620BB58A79B32DA8B3BFB1CF
8399468A096E2F0C54B8B3ECD15EE3FE2C5ABDB5A029
```

可选的 RSA 2048 签名使用 SUDI 的私有密钥生成，并可以通过 SUDI 证书中包含的 SUDI 公有密钥进行验证。系统会显示对 PCR 值、签名版本以及用户提供的随机数进行的签名。

```
RSA PKCS# 1 v1.5 Sign { <Nonce (UINT64)> || <Signature Version (UINT32)> || <PCR0 (32
bytes)>|| <PCR8 (32 bytes)> }
```

Inspur 管理解决方案能够解释以上输出，与发布的 Inspur 值进行比较，并验证签名。

执行设备设置配置

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个

特性，可以查看文档最后的特性信息表。
用户可以使用 Inspur 特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问 Inspur 特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于执行设备设置配置的信息

在执行包括 IP 地址分配以及 DHCP 自动配置等初始设备配置任务之前，请查看本模块的内容。

设备引导过程

要启用设备，用户需遵循硬件安装指南中的步骤，安装设备，给设备供电，并设置初始设备配置（IP 地址、子网掩码、默认网关、密码、Telnet 密码等等）。
正常的启动过程涉及到引导加载程序的以下活动：

- 执行低级 CPU 初始化。初始化控制物理内存映射到何处，以及其数量、速度等信息的 CPU 寄存器。
- 对 CPU 子系统执行加电自检（power-on self-test，POST）并测试系统的 DRAM。
- 初始化系统主板上的文件系统。
- 把默认的操作系统软件镜像加载到内存中并起动设备。

引导加载程序提供了在系统加载前访问文件系统的能力。正常情况下，引导加载程序仅被用来加载、解压并启动操作系统。在引导加载程序把 CPU 的控制权交给操作系统之后，直到下一次系统重置或启动之前它都不会活动。
如果操作系统的问题严重了到无法被使用的情况，引导加载程序也提供了隐藏的访问系统方式。隐藏机制提供了足够的系统访问权限，在必要时管理员可以使用 Xmodem 协议重新安装操作系统软件，在丢失或遗忘密码时恢复使用，并最终重启操作系统。
在指定设备信息之前，确保把 PC 或终端连接到了控制台端口，或者把 PC 连接到了以太网管理端口，并确保配置 PC 或终端模拟软件的波特率及字符格式满足如下设备控制台端口的设置：

- 波特率默认为 9600。
- 数据位默认为 8。
  注释： 如果设置数据位选项为 8，请设置奇偶校验和选项为无设置。
- 停止位默认为 2。
- 奇偶校验和默认为无设置。

软件安装程序特性

交换机上支持以下软件安装器特性：

- 在单独的交换机、交换机堆栈或交换机堆栈的交换机子集中安装软件包。如果配置了交换机堆栈，默认在其中所有交换机上进行安装。
- 在交换机堆栈中，Inspur 建议所有交换机都使用安装模式。
- 软件回滚的之前安装的软件包集合。
- 在引导闪存中没有合法的已安装软件包时进行紧急安装。

- 对加入交换机堆栈的使用不兼容软件的交换机进行自动升级。
- 使用一台交换机上的软件包作为源，并安装到交换机堆栈中的另一台交换机上。

注释： 软件安装及回滚必须仅在运行安装模式时进行。可以使用 EXEC 命令 **software expand** 来把软件包启动模式切换为安装模式。

软件引导模式

设备支持两种引导软件包的模式：

- 安装模式
- 捆绑包模式

安装引导模式

可以使用闪存中的软件包规划文件把设备引导为安装模式：

```
device: boot flash:packages.conf
```

规划文件中包含了要引导、挂载与运行的软件包列表。每个安装的软件包中的 ISO 文件系统会直接从闪存中挂载到根文件系统上。

注释： 用于引导安装模式的软件包以及规划文件必须位于闪存中。不支持通过 usbflash0: 或 tftp:引导为安装模式。

捆绑包引导模式

可以使用软件捆绑包（.bin）文件把设备引导为软件包引导模式：

```
switch: boot flash:cat3850-universalk9.SSA.03.08.83.EMD.150-8.83.EMD.bin
```

软件捆绑包中含有的规划文件决定要引导、挂载并运行哪些软件包。软件包会被从捆绑包中提取并拷贝到 RAM 中。每个软件包中的 ISO 文件系统会被挂载到根文件系统上。

与安装引导模式不同，在软件捆绑包模式中进行引导时，会使用与所用捆绑包大小相同的额外内存空间。

与安装引导模式不同，捆绑包引导模式可以在以下几个位置上使用：

- flash:
- usbflash0:
- tftp:

注释： 捆梆包引导模式不支持自动安装以及智能安装功能。

注释： 捆梆包引导模式中不支持 AP 镜像预下载特性。更多关于预下载特性的信息，参见 Inspur WLC 5700 系列的 *预加载镜像到接入点*一章。

交换机堆栈的引导模式

堆栈中的所有交换机都必须运行在安装模式或者捆梆包引导模式中。堆栈不支持混合使用引导模式。如果一台新的交换机尝试加入的堆栈，而其引导模式与活跃交换机不同，新交换机会进入 V-mismatch 的状态。

如果使用混合模式的交换机堆栈同时启动，则除活跃交换机之外的所有交换机会有 V-mismatch 的状态。如果引导模式不支持自动升级，则交换机堆栈成员必须在与活跃交换机

相同的引导模式中进行重新引导。
如果堆栈运行在安装模式中，可以使用自动升级特性来升级一台尝试加入交换机堆栈的交换机。
自动升级特性会把新交换机的引导模式改为安装模式。如果堆栈运行在捆梆包引导模式中，则自动升级特性不可用。此时管理员应使用捆梆包模式引导新的交换机，让它能够加入交换机堆栈。
以下示例展示了引导模式与活跃交换机不匹配时，尝试加入交换机堆栈的交换机状态：

```
Device# show switch
Switch/Stack Mac Address : 6400.f125.1100 - Local Mac Address
Mac persistency wait time: Indefinite
H/W Current
Switch# Role Mac Address Priority Version State
------------------------------------------------------------
1 Member 6400 f125.1a00 1 0 V-Mismatch
*2 Active 6400.f125.1100 1 V01 Ready
Device
```

指定设备信息

可以通过设备设置程序、DHCP 服务器或通过手动方式指定 IP 地址。
如果希望按照提示输入特定的 IP 信息，可以使用设备设置程序。通过此程序还可以配置主机名以及使能密码，指定 Telnet 密码（为远程管理提供安全性），把交换机配置为集群中的命令交换机或成员交换机，或配置为单独的交换机。
配置 DHCP 服务器后，可以使用 DHCP 服务器集中化控制并自动分配 IP 信息。
注释： 如果使用 DHCP，在设备接收动态分配的 IP 地址并读取配置文件之前，不要回答设置程序中的任何问题。
熟悉设备配置步骤的有经验用户可以手动配置设备。否则，请使用*引导过程*一节中描述的设置程序。

默认的交换机信息

表 176：默认的交换机信息

特性	默认设置
IP 地址以及子网掩码	未定义 IP 地址或子网掩码。
默认网关	未定义默认网关。
使能密码	未定义密码。
主机名	工厂分配的默认主机名是 Device。
Telnet 密码	未定义密码。
集群命令交换机功能	禁用。
集群名称	未定义集群名称。

基于 DHCP 的自动配置概述

DHCP 为 Internet 主机以及网络互连设备提供配置信息。该协议由两个组件组成：一个用于

从 DHCP 服务器向设备传送配置参数，另一个用来给设备分配网络地址。DHCP 构建在客户端-服务器模型上，指定的 DHCP 服务器会给动态配置的主机分配网络地址并传送配置参数。设备可以同时作为 DHCP 客户端和 DHCP 服务器。

在基于 DHCP 的自动配置过程中，设备（DHCP 客户端）在启动过程中会使用 IP 地址信息以及配置文件进行自动配置。

使用基于 DHCP 的自动配置时，设备上的 DHCP 客户端无需进行配置。然而，需要配置 DHCP 服务器使用与 IP 地址相关的多个租用选项。

如果希望使用 DHCP 来中继网络位置上的配置文件，可能还需要配置简单文件传输协议（Trivial File Transfer Protocol，TFTP）服务器以及域名系统（Domain Name System，DNS）服务器。

注释： 建议在交换机堆栈以及 DHCP、DNS 和 TFTP 服务器之间使用冗余连接。这有助于保证在一个连接的堆栈成员从交换机堆栈移除之后仍能访问这些服务器。

设备的 DHCP 服务器可以与设备在同一个 LAN 上，也可以在不同的 LAN 中。如果 DHCP 服务器运行在不同的 LAN 中，应该在设备与 DHCP 服务器之间配置 DHCP 中继设备。中继设备可以在两个直连的 LAN 之间转发广播流量。路由器不会转发广播数据包，只会基于接收包的目的 IP 地址转发数据包。

基于 DHCP 的自动配置会代替设备上的 BOOTP 客户端功能。

DHCP 客户端请求过程

启动设备时，DHCP 客户端会被调用，并在设备上没有配置文件时向 DHCP 服务器请求配置信息。如果存在配置文件，且配置中特定的被路由接口上包含接口配置命令 **ip address dhcp**，则会调用 DHCP 客户端为这些接口请求 IP 地址信息。

以下是 DHCP 客户端与 DHCP 服务器之间交换的消息序列。

图 134：DHCP 客户端及服务器消息交换

Switch A	交换机 A
DHCP server	DHCP 服务器
DHCPDISCOVER（broadcast）	DHCPDISCOVER（广播）
DHCPOFFER（unicast）	DHCPOFFER（单播）
DHCPREQUEST（broadcast）	DHCPREQUEST（广播）
DHCPACK（unicast）	DHCPACK（单播）

客户端交换机 A 会广播发送 DHCPDISCOVER 消息来定位 DHCP 服务器。DHCP 服务器会在 DHCPOFFER 单播消息中给客户端提供配置参数（比如 IP 地址、子网掩码、网关 IP 地址、DNS IP 地址、IP 地址的租期等等）。

在 DHCPREQUEST 广播消息中，客户端为提供的配置信息给 DHCP 服务器返回一个正式的请求。正式请求会广播发送，使所有其他收到这个客户端发送的 DHCPDISCOVER 消息的 DHCP 服务器收回分配给该客户端的 IP 地址。

DHCP 服务器给客户端发送一个 DHCPACK 单播消息，确认 IP 地址已经分配给了该客户端。通过这条消息，客户端和服务器被绑定在一起，客户端会使用从服务器收到的配置信息。设备接收的信息数量取决于如何配置 DHCP 服务器。

如果在 DHCPOFFER 单播消息中发送给客户端的配置参数是非法的（存在配置错误），客户端会给 DHCP 服务器发送 DHCPDECLINE 广播消息。

DHCP 服务器会给客户端发送 DHCPNAK 拒绝广播消息，表示提供的配置参数未被分配，在参

数协商过程中发生了错误，或者客户端回应 DHCPOFFER 消息过慢（DHCP 服务器已经把参数分配给了另一个客户端）。

DHCP 客户端可能会收到来自多个 DHCP 或 BOOTP 服务器的提议，且可以接收任意一个提议，然而，客户端通常接受收到的第一个提议。来自 DHCP 服务器的提议不能保证 IP 地址被分配给了客户端，然而 DHCP 服务器通常会预留该地址，直到客户端有机会正式请求地址。如果设备接受了来自 BOOTP 服务器的应答并且进行了自我配置，它会广播 TFTP 请求来获取设备配置文件。

DHCP 主机名选项允许一组设备通过中心化管理的 DHCP 服务器获取主机名以及标准配置。客户端会在 DHCPDISCOVER 消息中包含一个可选的 12 字段，向 DHCP 服务器请求主机名以及其他配置参数。除了通过 DHCP 获取的主机名，所有客户端的配置文件都相同。

如果客户端有默认的主机名（未配置全局配置命令 **hostname** *name* 或输入全局配置命令 **no hostname** 移除了主机名），输入接口配置命令 **ip address dhcp** 时，DHCP 主机名选项不会包含在数据包中。此时，如果客户端在为接口获取 IP 地址时通过 DHCP 交互接收到了 DHCP 主机名选项，客户端会接受 DHCP 用户名选项，并设置标志位以表示系统配置了主机名。

基于 DHCP 的自动配置及镜像更新

可以使用 DHCP 镜像升级特性，配置 DHCP 服务器给网络中的一台或多台设备下载新镜像以及新配置文件。对网络中的所有交换机同时进行镜像以及配置的升级能够确保加入到网络的每台新设备可以接收到相同的镜像及配置。

有两种类型的 DHCP 镜像升级：DHCP 自动配置以及 DHCP 自动镜像升级。

基于 DHCP 自动配置的限制

- 如果网络中没有处于 up 状态且未分配 IP 地址的三层接口，有保存配置的基于 DHCP 的自动配置过程会停止。
- 除非配置超时，否则有保存配置的基于 DHCP 的自动配置会无限次尝试下载 IP 地址。
- 如果配置文件不能被下载或配置文件损坏，自动安装过程会停止。
- 通过 TFTP 下载的配置文件会与运行配置中的现有配置合并，但是不会保存到 NVRAM 中，除非输入了特权 EXEC 命令 **write memory** 或 **copy running-configuration startup-configuration**。如果下载的配置被保存到启动配置中，在后续系统重启过程中该特性不会被触发。

DHCP 自动配置

DHCP 自动配置会从 DHCP 服务器给网络中的一台或多台设备下载配置文件。下载的配置文件会成为设备的运行配置。它不会覆盖保存在闪存中的启动配置，直到重启设备。

DHCP 自动镜像更新

可以使用 DHCP 自动镜像更新以及 DHCP 自动配置，给网络中的一台或多台设备同时下载配置及镜像。下载新配置以及新镜像的配置可以是空白的（或者只加载了默认工厂配置）。

如果新配置下载到了已经存在配置的交换机上，下载的配置会被附加到交换机存储的配置文件（现有配置不会被下载配置覆盖）。

要在设备上启用 DHCP 自动镜像更新，镜像以及配置所在的 DNS 服务器必须配置了正确的选项 67（配置文件名），选项 66（DHCP 服务器主机名），选项 150（TFTP 服务器地址）以及选项 125（Inspur INOS 镜像文件描述）设置。

把设备安装到网络中之后，自动镜像更新特性开始执行。下载的配置文件会被保存到设备的运行配置中，且新的镜像会下载并在设备上安装。重启设备时，配置会被存储在设备的保存配置中。

DHCP 服务器配置指南

按照以下指南把设备配置为 DHCP 服务器：

- 应该给 DHCP 服务器配置预留的租用信息，将租用地址与每台设备的硬件地址绑定。
- 如果希望设备接收 IP 地址信息，必须给 DHCP 服务器配置以下租用选项：
  - 客户端 IP 地址（必要）
  - 客户端子网掩码（必要）
  - DNS 服务器 IP 地址（可选）
  - 路由器 IP 地址（设备应使用的默认网关地址）（必要）
- 如果希望设备通过 TFTP 服务器接收配置文件，必须给 DHCP 服务器配置以下租用选项：
  - TFTP 服务器名（必要）
  - 引导文件名（客户端所需的配置文件名称）（建议）
  - 主机名（可选）
- 根据 DHCP 服务器的设置不同，设备可以接收 IP 地址信息或配置文件，也可以同时接收两者。
- 如果不给 DHCP 服务器配置之前上述租用选项，它会只给客户端回应配置了的参数。如果应答中没有 IP 地址以及子网掩码，设备不会被配置。如果未找到路由器 IP 地址或 TFTP 服务器名称，设备可能会发送广播 TFTP 请求。其他租用选项不可用不会影响自动配置过程。

设备可以作为 DHCP 服务器使用。默认情况下，Inspur INOS DHCP 服务器以及中继代理特性在设备上启用但未被配置（这些特性不会工作）。

TFTP 服务器的目的

基于 DHCP 服务器的配置，设备尝试从 TFTP 服务器下载一个或多个配置文件。如果配置 DHCP 服务器给设备回复 IP 连通 TFTP 服务器的所有必要选项，而且为 DHCP 服务器配置了 TFTP 服务器名称、地址以及配置文件，设备会尝试通过指定的 TFTP 服务器下载指定配置文件。

如果没有指定配置文件及 TFTP 服务器，或者配置文件不能被下载，设备会尝试使用多种文件名和 TFTP 服务器地址的组合下载配置文件。文件包含指定的配置文件或这些文件：network-config、inspurnet.cfg、hostname.config 或 hostname.cfg，其中 *hostname* 是设备当前的主机名。使用的 TFTP 地址包括指定的 TFTP 服务器地址或者广播地址（255.255.255.255）。

要使设备能成功下载配置文件，TFTP 服务器的基本目录中必须包含一个或多个配置文件。这些文件可以有：

- DHCP 应答中说明的配置文件（实际设备的配置文件）。
- network-confg 或 inspurnet.cfg 文件（默认配置文件）。
- router-confg 或 inspurrtr.cfg 文件（这些文件包含所有设备通用的命令。正常情况下，如果正确配置了 DHCP 以及 TFTP 服务器，这些文件不会被访问）。

如果在 DHCP 服务器租用数据库中指定了 TFTP 服务器名称，必须也在 DNS 服务器数据库中配置 TFTP 服务器名称到 IP 地址的映射。

如果使用的 TFTP 服务器与设备在不同的 LAN 中，或者要通过广播地址访问（如果 DHCP 服务器应答不含有之前所述的所有必须信息，会发生此情况），必须配置使用中继把 TFTP 数据

包转发到 TFTP 服务器上。首选方案是给 DHCP 服务器配置所有必须的信息。

DNS 服务器的目的

DHCP 服务器使用 DNS 服务器把 TFTP 服务器的名称解析为 IP 地址。必须在 DNS 服务器上配置 TFTP 服务器名称到 IP 地址的映射。TFTP 服务器包含着设备的配置文件。

可以在 DHCP 服务器的租用数据库中配置 DNS 服务器的 IP 地址，DHCP 应答会读取这些信息。在租用数据库中最多可以输入两个 DNS 服务器 IP 地址。

DNS 服务器可以与设备在相同 LAN 上，也可以在不同的 LAN 上。如果在不同 LAN 上，设备必须可以通过路由器访问该服务器。

如何获取配置文件

根据 IP 地址可用性以及 DHCP 预留租用信息中配置文件名的不同，设备会通过以下方式获取配置信息：

- 为设备预留了 IP 地址以及配置文件名，且通过 DHCP 应答提供给设备（读取一个文件的方式）。

  设备从 DHCP 服务器接收 IP 地址、子网掩码、TFTP 服务器地址以及配置文件名。设备给 TFTP 服务器发送单播消息，提取服务器基本目录上对应名称的配置文件，接收完毕后，设备完成启动过程。
- 为设备预留了 IP 地址以及配置文件名，但是 DHCP 应答中没有提供 TFTP 服务器地址（读取一个文件的方式）。

  设备从 DHCP 服务器接收 IP 地址、子网掩码以及配置文件名。设备给 TFTP 服务器发送广播消息，提取服务器基本目录上对应名称的配置文件，接收完毕后，设备完成启动过程。
- 仅为设备预留了 IP 地址并通过 DHCP 应答提供给设备，未提供配置文件名（读取两个文件的方式）。

  设备从 DHCP 服务器接收 IP 地址、子网掩码以及 TFTP 服务器地址。设备给 TFTP 服务器发送单播消息，提取默认配置文件 network-confg 或 inspurnet.cfg（如果无法读取 network-confg，设备会读取 inspurnet.cfg 文件）。

  默认配置文件包含设备主机名到 IP 地址的映射。设备使用文件中的信息填写自己的主机表，并获取其主机名。如果文件中未找到主机名，设备会使用 DHCP 应答中的主机名。如果 DHCP 应答中未指定主机名，设备会使用默认的 *Switch* 作为主机名。

  在通过默认配置文件或者 DHCP 应答获取主机名之后，设备会从 TFTP 服务器上读取与自己主机名相同的配置文件（根据之前读取的是 network-confg 或 inspurnet.cfg，可能读取(*hostname*-confg 或 *hostname*.cfg）。如果读取了 inspurnet.cfg 文件，主机的文件名会被截取为八个字符。

  如果设备不能读取 network-confg、inspurnet.cfg 或 hostname 文件，它会读取 router-confg 文件。如果设备不能读取 router-confg 文件，它会读取 inspurrtr.cfg 文件。

注释： 如果没能从 DHCP 应答中获取 TFTP 服务器，所有通过单播传输读取配置文件的尝试都失败，或者 TFTP 服务器名称不能被解析为 IP 地址，则设备会广播发送 TFTP 服务器请求。

如何控制环境变量

对于正常运行的设备，仅可以通过配置为 9600 bps 的控制台连接来进入引导加载程序模式。拔掉设备的电源线，重连电源线时按住 **Mode** 键。在系统所有琥珀色的 LED 灯都打开并保持常亮后，可以放开 **Mode** 键。此后会出现引导加载程序设备提示符。

设备引导加载程序提供对非易失性环境变量的支持，可以由此控制引导加载程序或系统运行的任何其他软件的运行方式。引导加载程序的环境变量与可以在 UNIX 或 DOS 系统上设置的环境变量类似。

有值的环境变量会被存储在闪存文件系统之外的闪存内存中。

这些文件的每一行都包含一个环境变量名，一个等号，接着是变量值。未出现的变量没有值；如果变量在文件中列出，就算值是空字符串，该变量也有值。设置为空字符串（如“”）的变量是有值的变量。许多环境变量都有预定义的默认值。

可以通过访问引导加载程序或使用 Inspur INOS 的命令来更改环境变量的设置。在正常情况下，没有必要更改环境变量的设置。

常用的环境变量

下表描述了众多常用环境变量的功能。

表 177：常用环境变量

变量	引导加载程序命令	Inspur INOS 全局配置命令
BOOT	**set BOOT** *filesystem* **:/** *file-url …* 设置自动引导时尝试加载并执行的可执行文件列表，用分号隔离。	**boot system** {*filesystem* **:** */file-url …* |**switch** {*number*	**all**}} 指定下一次引导循环期间要加载的 Inspur INOS 镜像以及要加载镜像的堆栈成员。此命令常用来更高 BOOT 环境变量的设置。 软件包规划文件，也称为 *packages.conf* 文件，被系统用来决定在启动过程中要激活哪些软件包。 在安装模式进行引导时，**boot**命令指定的软件包规划文件会被用来决定激活哪些软件包。比如 **bootflash:packages.conf**。 在捆梆包模式进行引导时，引导的捆梆包中的软件包规划文件会被用来激活捆梆包中的软件包。比如 **boot flash:image.bin**。
MANUAL_BOOT	**set MANUAL_BOOT yes** 决定交换机是自动引导还是手动引导。 合法的值是 1、yes、0 和 no。如果设置为 0 或 no，引导加载程序会尝试自动引导系	**boot manual** 在下一次引导循环期间手动引导交换机，并更改环境变量 MANUAL_BOOT 的设置。下次重启系统时，交换机会在引导加载程序模式中。要

	统。如果设置为其他值，必须在引导加载程序模式中手动引导交换机。	引导系统，需使用引导加载程序命令**boot flash:***filesystem* **:/** *file-url*，并指定可引导镜像的名称。
CONFIG_FILE	**set CONFIG_FILE flash:/***file-url* 更改 Inspur INOS 用来读写系统配置的非易失性拷贝的文件名。	**boot config-file flash:/** *file-url* 指定 Inspur INOS 用来读写系统配置的非易失性拷贝的文件名。此命令会更改 CONFIG_FILE 环境变量。
SWITCH_NUMBER	**set SWITCH_NUMBER***stack-member-number* 更改堆栈成员的成员编号。	**switch** *current-stack-member-number***renumber***new-stack-member-number* 更改堆栈成员的成员编号。
SWITCH_PRIORITY	**set SWITCH_PRIORITY***stack-member-number* 更改堆栈成员的优先级值。	**switch** *stack-member-number* **priority***priority-number* 更改堆栈成员的优先级值。
BAUD	**set BAUD** *baud-rate*	**line console 0** **speed** *speed-value* 配置波特率。
ENABLE_BREAK	**set ENABLE_BREAK yes/no**	**boot enable-break switch yes/no** 允许打断自动引导循环。用户有 5 秒中的时间来输入 **break** 命令。

TFTP 的环境变量

当交换机通过以太网管理端口连接到 PC 时，可以使用 TFTP 给引导加载程序下载或上传配置文件。确保配置了下表中的环境变量。

表 178：TFTP 的环境变量

变量	描述
MAC_ADDR	指定交换机的 MAC 地址。 **注释：**建议不修改此变量。 然而，如果在引导加载程序启用之后修改此变量，或者变量值与保存的值不同，请在使用 TFTP 之前输入此命令。
IP_ADDR	为交换机指定关联 IP 子网中的 IP 地址及子网掩码。
DEFAULT_ROUTER	指定默认网关的 IP 地址以及子网掩码。

计划重新加载软件镜像

可以计划设备在之后的时间重新加载软件镜像（如在深夜或在设备使用较少的周末），也可以在网络范围内同步进行重新加载（如对网络中的所有设备执行软件更新）。

注释： 计划的重新加载必须在大约 24 天内发生。

有以下重新加载选项：

- 重新加载软件在指定的分钟或小时分钟后生效。重新加载必须在约 24 小时内发生。可以使用至多 255 字符的字符串来指明重新加载的原因。
- 重新加载在指定的时间发生（使用 24 小时制）。如果指定了月份和日期，重新加载会计划在指定的时间和日期发生。如果未指定月份和日期，重新加载会在当天（如果指定的时间比当前时间晚）或下一天（如果指定的时间比当前时间早）指定的时间发生。指定 00:00 会计划在午夜进行重新加载。

reload 命令会让系统停机。如果系统未设置成手动引导，则设备会自行重启。

如果设备配置了手动引导，请不要通过虚拟终端进行重启。此限制避免了设备进入引导加载程序模式后夺取了远程用户的控制权。

如果修改了配置文件，设备会在重新加载之前提示用户保存配置。在保存操作过程中，如果环境变量 CONFIG_FILE 指向的启动配置文件不存在，系统会请求用户确认是够继续保存。此时如果继续操作，系统会在重新加载时进入设置模式。

要取消之前计划的重新加载，使用特权 EXEC 命令 **reload cancel**。

如何执行设备设置配置

使用 DHCP 给设备下载新镜像以及新配置要求管理员至少要配置两台设备。一台设备作为 DHCP 以及 TFTP 服务器，第二台设备（客户端）要配置下载新配置文件或是同时下载新配置文件以及新镜像文件。

配置 DHCP 自动配置（仅下载配置文件）

总步骤

1. configure terminal
2. ip dhcp pool *poolname*
3. boot *filename*
4. network *network-number mask prefix-length*
5. default-router *address*
6. option 150 *address*
7. exit
8. tftp-server flash:*filename.text*
9. interface *interface-id*
10. no switchport
11. ip address *address mask*
12. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device# configure terminal`	进入全局配置模式。

步骤 2	**ip dhcp pool** *poolname* **示例:** `Device(config)#` **ip dhcp pool pool**	创建 DHCP 服务器地址池名称，并进入 DHCP 地址池配置模式。
步骤 3	**boot** *filename* **示例:** `Device(dhcp-config)#` **boot config-boot.text**	指定用作引导镜像的配置文件名称。
步骤 4	**network** *network-number mask prefix-length* **示例:** `Device(dhcp-config)#` **network 10.10.10.0 255.255.255.0**	指定 DHCP 地址池的网络地址以及掩码。 **注释：**前缀长度指定了组成地址前缀的比特数。前缀是指定客户端网络掩码的另一种方式。前缀长度必须在正斜线（/）之后。
步骤 5	**default-router** *address* **示例:** `Device(dhcp-config)#` **default-router 10.10.10.1**	为 DHCP 客户端指定默认路由器的 IP 地址。
步骤 6	**option 150** *address* **示例:** `Device(dhcp-config)#` **option 150 10.10.10.1**	指定 TFTP 服务器的 IP 地址。
步骤 7	**exit** **示例:** `Device(dhcp-config)#` **exit**	返回全局配置模式。
步骤 8	**tftp-server flash**:*filename.text* **示例:** `Device(config)#` **tftp-serverflash:config-boot.text**	指定 TFTP 服务器上的配置文件。
步骤 9	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/4**	指定要接收配置文件的客户端地址。
步骤 10	**no switchport** **示例:** `Device(config-if)#` **no switchport**	将接口设置为三层模式。
步骤 11	**ip address** *address mask* **示例:** `Device(config-if)#` **ip address 10.10.10.1 255.255.255.0**	指定接口的 IP 地址以及掩码。
步骤 12	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。

配置 DHCP 自动镜像更新（下载配置文件及镜像）

此任务描述了对现有设备进行 TFTP 以及 DHCP 设置的过程，以使用 DHCP 自动配置来进行新交换机的安装设置。

在开始前

必须先创建一个要被上传到设备上的 text 文件（如 autoinstall_dhcp）。在 text 文件中输入希望下载的镜像名（如 c3750e-ipservices-mz.122-44.3.SE.tarc3750x-ipservices-mz.122-53.3.SE2.tar）。此镜像必须是 tar 文件，而不能是 bin 文件。

总步骤

1. **configure terminal**
2. **ip dhcp pool** *poolname*
3. **boot** *filename*
4. **network** *network-number mask prefix-length*
5. **default-router** *address*
6. **option 150** *address*
7. **option 125** *hex*
8. **copy tftp flash** *filename.txt*
9. **copy tftp flash** *imagename.bin*
10. **exit**
11. **tftp-server flash:** *config.text*
12. **tftp-server flash:** *imagename.bin*
13. **tftp-server flash:** *filename.txt*
14. **interface** *interface-id*
15. **no switchport**
16. **ip address** *address mask*
17. **end**
18. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device# configure terminal`	进入全局配置模式。
步骤 2	**ip dhcp pool** *poolname* **示例:** `Device(config)# ip dhcp pool pool`	创建 DHCP 服务器地址池名称，并进入 DHCP 地址池配置模式。
步骤 3	**boot** *filename* **示例:** `Device(dhcp-config)# boot config-boot.text`	指定用作引导镜像的配置文件名称。
步骤 4	**network** *network-number mask prefix-length* **示例:** `Device(dhcp-config)# network 10.10.10.0 255.255.255.0`	指定 DHCP 地址池的网络地址以及掩码。 **注释：**前缀长度指定了组成地址前缀的比特数。前缀是指定客户端网络掩码的另

		一种方式。前缀长度必须在正斜线（/）之后。
步骤 5	**default-router** *address* **示例:** `Device(dhcp-config)#` **default-router 10.10.10.1**	为 DHCP 客户端指定默认路由器的 IP 地址。
步骤 6	**option 150** *address* **示例:** `Device(dhcp-config)#` **option 150 10.10.10.1**	指定 TFTP 服务器的 IP 地址。
步骤 7	**option 125** *hex* **示例:** `Device(dhcp-config)#` **option 125 hex 0000.0009.0a05.08661.7574.6f69.6e73.7461.6c6c.5f64.686370**	指定 test 文件的路径，该文件描述了镜像文件的路径。
步骤 8	**copy tftp flash** *filename.txt* **示例:** `Device(config)#` **copy tftp flash image.bin**	上传 text 文件到设备上。
步骤 9	**copy tftp flash** *imagename.bin* **示例:** `Device(config)#` **copy tftp flash image.bin**	上传新镜像的 tar 文件到设备上。
步 骤 10	**exit** **示例:** `Device(dhcp-config)#` **exit**	返回全局配置模式。
步 骤 11	**tftp-server flash:** *config.text* **示例:** `Device(config)#` **tftp-server flash:config-boot.text**	指定 TFTP 服务器上的 Inspur INOS 配置文件。
步 骤 12	**tftp-server flash:** *imagename.bin* **示例:** `Device(config)#` **tftp-server flash:image.bin**	指定 TFTP 服务器上的镜像名称。
步 骤 13	**tftp-server flash:** *filename.txt* **示例:** `Device(config)#` **tftp-server flash:boot-config.text**	指定包含要下载的镜像文件名称的 text 文件。
步 骤 14	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitEthernet1/0/4**	指定要接收配置文件的客户端地址。
步 骤 15	**no switchport** **示例:** `Device(config-if)#` **no switchport**	把接口设置为三层模式。
步 骤 16	**ip address** *address mask* **示例:** `Device(config-if)#` **ip address 10.10.10.1 255.255.255.0**	为接口指定 IP 地址及掩码。
步 骤 17	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。
步 骤 18	**copy running-config startup-config** **示例:**	（可选）把配置的条目保存在配置文件中。

	`Device(config-if)#` **end**	

配置客户端从 DHCP 服务器下载文件

注释： 管理员应该只配置启用三层接口。不要指定 IP 地址或使用有保存配置的基于 DHCP 的自动配置。

总步骤

1. **configure terminal**
2. **boot host dhcp**
3. **boot host retry timeout** *timeout-value*
4. **banner config-save ^C** *warning-message* **^C**
5. **end**
6. **show boot**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**boot host dhcp** **示例:** `Device(conf)#` **boot host dhcp**	启用有保存配置的自动配置。
步骤 3	**boot host retry timeout** *timeout-value* **示例:** `Device(conf)#` **boot host retry timeout 300**	（可选）设置系统尝试下载配置文件的总时间。 **注释：**如果不设置超时，系统会无限次地尝试从 DHCP 服务器获取 IP 地址。
步骤 4	**banner config-save ^C** *warning-message* **^C** **示例:** `Device(conf)#` **banner config-save ^C Caution -Saving Configuration Fileto NVRAM May Cause You to No longer Automatically Download Configuration Files at Reboot^C**	（可选）创建警告消息，在尝试把配置文件保存到 NVRAM 时显示。
步骤 5	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。
步骤 6	**show boot** **示例:** `Device#` **show boot**	验证配置。

为多个 SVI 手动指定 IP 信息

此任务描述了如何手动给多个交换虚拟接口（SVI）指定 IP 信息。

总步骤

1. **configure terminal**
2. **interface vlan** *vlan-id*
3. **ip address** *ip-address subnet-mask*
4. **exit**
5. **ip default-gateway** *ip-address*
6. **end**
7. **show interfaces vlan** *vlan-id*
8. **show ip redirects**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface vlan** *vlan-id* **示例:** `Device(config)#` **interface vlan 99**	进入接口配置模式，输入分配 IP 信息的 VLAN。范围从 1 到 4094。
步骤 3	**ip address** *ip-address subnet-mask* **示例:** `Device(config-vlan)#` **ip address10.10.10.2 255.255.255.0**	输入 IP 地址以及子网掩码。
步骤 4	**exit** **示例:** `Device(config-vlan)#` **exit**	返回全局配置模式。
步骤 5	**ip default-gateway** *ip-address* **示例:** `Device(config)#` **ip default-gateway10.10.10.1**	输入下一跳路由器接口的 IP 地址，该接口直接连接配置了默认网关的设备。默认网关会接收设备发来的带有未解析目的 IP 地址的 IP 数据包。 配置默认网关之后，设备就有了主机通信所需的远程网络连通性。 **注释：**当配置设备进行 IP 路由时，不需要设置默认网关。 **注释：**设备的 CAPWAP 依靠默认网关配置来支持加入设备的被路由接入点。
步骤 6	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 7	**show interfaces vlan** *vlan-id* **示例:**	验证配置的 IP 地址。

	Device# **show interfaces vlan 99**	
步骤 8	**show ip redirects** **示例:** Device# **show ip redirects**	验证配置的默认网关。

修改设备设置配置

指定读写系统配置的文件名

默认情况下，Inspur INOS 软件使用 config.text 文件来读写系统配置的非易失性拷贝。然而可以指定不同的文件名，在下一个引导循环加载使用。

在开始前

使用单独的交换机进行此任务。

总步骤

1. **configure terminal**
2. **boot flash**:*/file-url*
3. **end**
4. **show boot**
5. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** Device# **configure terminal**	进入全局配置模式。
步骤 2	**boot flash**:*/file-url* **示例:** Device(config)# **boot flash:config.text**	指定下一个引导循环要加载的配置文件。 *file-url*——路径（目录）以及配置文件名。文件名以及目录名区分大小写。
步骤 3	**end** **示例:** Device(config)# **end**	返回特权 EXEC 模式。
步骤 4	**show boot** **示例:** Device# **show boot**	验证配置的条目。 全局配置命令 **boot** 会更改环境变量 CONFIG_FILE 的设置。
步骤 5	**copy running-config startup-config** **示例:** Device# **copy running-config startup-config**	（可选）把配置的条目保存到配置文件中。

手动引导交换机

默认时，交换机会自动引导启动，然而也可以配置交换机手动引导启动。

在开始前

使用单独的交换机进行此任务。

总步骤

1. **configure terminal**
2. **boot manual**
3. **end**
4. **show boot**
5. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device# configure terminal`	进入全局配置模式。
步骤 2	**boot manual** **Example:** `Device(config)# boot manual`	使交换机在一下个引导循环中手动引导启动。
步骤 3	**end** **示例:** `Device(config)# end`	返回特权 EXEC 模式。
步骤 4	**show boot** **示例:** `Device# show boot`	验证配置的条目。 全局配置命令 **boot manual** 会更改环境变量 MANUAL_BOOT 的设置。下一次重启系统时，交换机会进入引导加载程序模式，如提示符 *switch:*所示。要引导系统，请使用 **boot***filesystem:/file-url*引导加载程序命令。 • *filesystem*:——使用flash:作为系统主板闪存设备。 `device: boot flash:` • *file-url*——指定可引导镜像的路径（目录）及名称。 文件名及目录名区分大小写。
步骤 5	**copy running-config startup-config** **示例:** `Device# copy running-config startup-config`	（可选）把配置的条目保存到配置文件中。

引导安装模式中的设备

总步骤

1. **cp** *source_file_path destination_file_path*
2. **software expand file** *source_file_path*
3. **reload**
4. **boot flash:packages.conf**

5. show version

具体步骤

	命令或操作	目的
步骤 1	**cp** *source_file_path destination_file_path* **示例：** Device# **copy** **tftp://10.0.0.6/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin** **flash:**	进入全局配置模式。
步骤 2	**software expand file** *source_file_path* **示例：** 展开TFTP服务器的bin文件： Switch# **software expand file** **tftp://10.0.0.2/cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin** **to flash:** Preparing expand operation ... [1]: Downloading file tftp://10.0.0.2/cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin to active switch 1 [1]: Finished downloading file tftp://10.0.0.2/cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37. EXP.bin to active switch 1 [1]: Copying software from active switch 1 to switch 2 [1]: Finished copying software to switch 2 [1 2]: Expanding bundle cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin [1 2]: Copying package files [1 2]: Package files copied [1 2]: Finished expanding bundle cat3k_caa-universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin 18 -rw- 74387812 Dec 7 2012 05:55:43	使交换机在一下个引导循环中手动引导启动。

	+00:00 cat3k_caa- base.SSA.03.09.37.EXP.pkg 19 -rw- 2738868 Dec 7 2012 05:55:44 +00:00 cat3k_caa- drivers.SSA.03.09.37.EXP.pkg 20 -rw- 32465772 Dec 7 2012 05:55:44 +00:00 cat3k_caa-infra.SSA.03.09.37.EXP.pkg 21 -rw- 30389036 Dec 7 2012 05:55:44 +00:00 cat3k_caa-INOSd-universalk9.SSA.150- 9.37.EXP.pkg 22 -rw- 18342624 Dec 7 2012 05:55:44 +00:00 cat3k_caa- platform.SSA.03.09.37.EXP.pkg 23 -rw- 63374028 Dec 7 2012 05:55:44 +00:00 cat3k_caa- wcm.SSA.10.0.10.14.pkg 17 -rw- 1239 Dec 7 2012 05:56:29 +00:00 packages.conf	
步骤 3	**reload** **示例：** Device: **reload**	返回特权 EXEC 模式。
步骤 4	**boot flash:packages.conf** **示例：** switch: **boot flash:packages.conf**	验证配置的条目。 全局配置命令 **boot manual** 会更改环境变量 MANUAL_BOOT 的设置。 下一次重启系统时，交换机会进入引导加载程序模式，如提示符*switch:*所示。要引导系统，请使用**boot***filesystem:/file-url*引导加载程序命令。 • *filesystem*:——使用flash:作为系统主板闪存设备。 device: **boot flash:** • *file-url*——指定可引导镜像的路径（目录）及名称。 文件名及目录名区分大小写。
步骤 5	**show version** **示例：** switch# **show version** Switch Ports Model SW Version SW Image Mode	（可选）把配置的条目保存到配置文件中。

	------ ----- ----- ---------- ------- --- ---- 1 6 WS-C3850-6DS-S 03.09.26.EXP ct3850-ipservicesk9 INSTALL	

在捆梆包模式中引导设备

有几种可以引导设备的方式——可以从 TFTP 服务器拷贝 bin 文件然后引导设备，也可以使用 **boot flash:<image.bin>**或 **boot usbflash0:<image.bin>**命令直接从闪存或 USB 闪存引导设备。

总步骤

1. cp *source_file_path destination_file_path*

2. switch:BOOT=<source path of .bin file>

3. boot

4. show version

具体步骤

	命令或操作	目的
步骤 1	**cp** *source_file_path destination_file_path* **示例：** Device# **copy** **tftp://10.0.0.6/cat3k_caa-** **universalk9.SSA.03.12.02.EZP.150-** **12.02.EZP.150-12.02.EZP.bin** **flash:**	（可选）从 FTP 或 TFTP 服务器上把 bin 文件（image.bin）拷贝到闪存或 USB 闪存上。
步骤 2	**switch:BOOT=<source path of .bin file>** **示例：** Device: **switch:BOOT=tftp://10.0.0.2/cat3k_caa-** **universalk9.SSA.03.09.37.EXP.150-9.37.EXP.bin**	设置引导参数。
步骤 3	**boot** **示例：** switch: **boot**	引导设备。
步骤 4	**show version** **示例：** switch# **show version** Switch Ports Model SW Version SW Image Mode ------ ----- ----- ---------- --------- - ---- 1 6 WS-C3850-6DS-S 03.09.40.EXP ct3850- ipservicesk9 BUNDLE	验证设备在 **BUNDLE** 模式中。

在交换机堆栈上引导特定的软件镜像

默认情况下，交换机会使用 BOOT 环境变量中的信息自动引导启动系统。如果此变量未设置，

交换机会对闪存文件系统进行递归的深度优先搜索，加载并执行第一个找到的可执行镜像。在深度优先搜索目录的过程中，继续搜索原始目录之前每个遇到的子目录都会被完全搜索。然而，也可以指定特定的镜像来进行引导启动。

总步骤

1. **configure terminal**
2. **boot system switch** {*number* | **all**}
3. **end**
4. **show boot system**
5. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**boot system switch** {*number* | **all**} **示例：** `Switch(config)#` **boot system switch 2 flash:cat3850-universalk9.SSA.03.08.83.EMD.150-8.83.EMD.b**	（可选）对于堆栈中的交换机，需指定下一个引导循环时要加载系统镜像的交换机成员： • 使用 *number* 指定一个堆栈成员（只能指定一个堆栈成员）。 • 使用 **all** 指定所有堆栈成员。 如果进入了一个 Inspur 3750-X 堆栈的 master 或成员，则只为其他 Inspur 3750-X 堆栈成员指定交换机镜像。 如果进入了一个 Inspur 3750-E 堆栈的 master 或成员，则只为其他 Inspur 3750-E 堆栈成员指定交换机镜像。 如果希望为 Inspur 3750 交换机指定镜像，请在 Inspur 3750 堆栈成员上输入此命令。
步骤 3	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 4	**show boot system** **示例：** `Device#` **show boot system**	验证配置的条目。 全局配置命令 **boot system** 会更改 BOOT 环境变量的设置。 在下一个引导循环中，交换机会尝使用 BOOT 环境变量中的信息自动引导启动系统。
步骤 5	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）把配置的条目保存到配置文件中。

配置计划的软件镜像重载

此任务描述了如何配置设备在之后重新加载软件镜像。

总步骤

1. **configure terminal**
2. **copy running-config startup-config**
3. **reload in** [*hh*:]*mm* [*text*]
4. **reload at** *hh*: *mm* [*month day* | *day month*] [*text*]
5. **reload cancel**
6. **show reload**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**copy running-config startup-config** **示例：** **copy running-config startup-config**	在使用 **reload** 命令之前把设备的配置信息保存到启动配置中。
步骤 3	**reload in** [*hh*:]*mm* [*text*] **示例：** `Device(config)#` **reload in 12** `System configuration has been modified. Save? [yes/no]:` **y**	计划软件重新加载在指定的分钟或小时分钟以后生效。重新加载必须在约 24 日以内进行。可以使用一个含有至多 255 个字符的字符串说明重新加载的原因。
步骤 4	**reload at** *hh*: *mm* [*month day* | *day month*] [*text*] **示例：** `Device(config)#` **reload at 14:00**	按照小时分钟的方式指定重新加载的时间。 注释：仅在设置了设备时钟（通过网络时间协议（NTP）、硬件日历或手动设置）的情况下使用 **at** 关键字。此时间相对于设备上配置的时区。要计划多台设备同时进行重载，每台设备的时间必须使用 NTP 进行同步。
步骤 5	**reload cancel** **示例：** `Device(config)#` **reload cancel**	取消之前计划的重载。
步骤 6	**show reload** **示例：** **show reload**	显示之前计划的重载信息，或者显示设备上是否配置了重载计划。

监控设备设置配置

示例：验证设备的运行配置

```
Device# show running-config
Building configuration...
Current configuration: 1363 bytes
!
version 12.4
no service pad
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Stack1
!
enable secret 5 $1$ej9.$DMUvAUnZOAmvmgqBEzIxE0
!
.
<output truncated>
.
interface gigabitethernet6/0/2
mvr type source
<output truncated>
...!
interface VLAN1
ip address 172.20.137.50 255.255.255.0
no ip directed-broadcast
!
ip default-gateway 172.20.137.1 !
!
snmp-server community private RW
snmp-server community public RO
snmp-server community private@es0 RW
snmp-server community public@es0 RO
snmp-server chassis-id 0x12
!
end
```

示例：显示安装模式中的软件引导过程

此示例展示了安装模式中的软件引导过程。

```
switch: boot flash:packages.conf
Getting rest of image
Reading full image into memory....done
Reading full base package into memory...: done = 74596432
Nova Bundle Image
--------------------------------------
Kernel Address : 0x6042f354
Kernel Size : 0x318412/3245074
Initramfs Address : 0x60747768
Initramfs Size : 0xdc08e8/14420200
Compression Format: .mzip
Bootable image at @ ram:0x6042f354
Bootable image segment 0 address range [0x81100000, 0x81b80000] is in range
[0x80180000,
0x90000000].
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@boot
_system:
377
Loading Linux kernel with entry point 0x811060f0 ...
Bootloader: Done loading app on core_mask: 0xf
### Launching Linux Kernel (flags = 0x5)
All packages are Digitally Signed
Starting System Services
Nov 7 09:57:05 %INOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_START: Switch 2
is
starting stack discovery
######################################################################################
#################################
Nov 7 09:59:07 %INOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_DONE: Switch 2
has
finished stack discovery
Nov 7 09:59:07 %INOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-SWITCH_ADDED: Switch
2
has been added to the stack
Nov 7 09:59:14 %INOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-ACTIVE_ELECTED:
Switch 2
has been elected ACTIVE
Restricted Rights Legend
Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.
```

```
inspur Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706
Inspur INOS Software, Inspur L3 Switch Software (CAT3K_CAA-UNIVERSALK9-M),
Version 03.09.12.EMD EARLY DEPLOYMENT ENGINEERING NOVA_WEEKLY BUILD, synced to
DSGS_PI2_POSTPC_FLO_DSBU7_NG3K_1105
Copyright (c) 1986-2012 by Inspur Systems,
Inc. Compiled Sun 04-Nov-12 22:53 by gereddy
License level to INOSd is ipservices
```

此示例显示了捆梆包模式中的软件引导过程：

```
switch: boot flash:cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin
Reading full image into
memory..................................................................done
Nova Bundle Image
--------------------------------------
Kernel Address : 0x6042ff38
Kernel Size : 0x318412/3245074
Initramfs Address : 0x6074834c
Initramfs Size : 0xdc08e8/14420200
Compression Format: .mzip
Bootable image at @ ram:0x6042ff38
Bootable image segment 0 address range [0x81100000, 0x81b80000] is in range
[0x80180000,
0x90000000].
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
File "flash:cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin" uncompressed and
installed, entry point: 0x811060f0
Loading Linux kernel with entry point 0x811060f0 ...
Bootloader: Done loading app on core_mask: 0xf
### Launching Linux Kernel (flags = 0x5)
All packages are Digitally Signed
Starting System Services
Nov 7 09:45:49 %INOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_START: Switch 2
is
starting stack discovery
######################################################################################
#################################
Nov 7 09:47:50 %INOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-DISC_DONE: Switch 2
has
finished stack discovery
Nov 7 09:47:50 %INOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-SWITCH_ADDED: Switch
2
has been added to the stack
Nov 7 09:47:58 %INOSXE-1-PLATFORM: process stack-mgr: %STACKMGR-1-ACTIVE_ELECTED:
```

```
Switch 2
has been elected ACTIVE
Restricted Rights Legend
Use, duplication, or disclosure by the Government is
subject to restrictions as set forth in subparagraph
(c) of the Commercial Computer Software - Restricted
Rights clause at FAR sec. 52.227-19 and subparagraph
(c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.
inspur Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706
Inspur INOS Software, Inspur L3 Switch Software (CAT3K_CAA-UNIVERSALK9-
M), Version 03.09.12.EMD
EARLY DEPLOYMENT ENGINEERING NOVA_WEEKLY BUILD, synced to
DSGS_PI2_POSTPC_FLO_DSBU7_NG3K_1105
Copyright (c) 1986-2012 by Inspur Systems,
Inc. Compiled Sun 04-Nov-12 22:53 by gereddy
License level to INOSd is ipservices
```

示例：紧急安装

此示例展示了输入 **emergency-install** 引导命令时的示例输出：

```
switch: emergency-install
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin
The bootflash will be erased during install operation, continue (y/n)?y
Starting emergency recovery
(tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin)...
Reading full image into memory......................done
Nova Bundle Image
--------------------------------------
Kernel Address : 0x6042e5cc
Kernel Size : 0x318261/3244641
Initramfs Address : 0x60746830
Initramfs Size : 0xdb0fb9/14356409
Compression Format: .mzip
Bootable image at @ ram:0x6042e5cc
Bootable image segment 0 address range [0x81100000, 0x81b80000] is in range
[0x80180000,
0x90000000].
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
File "sda9:c3850-recovery.bin" uncompressed and installed, entry point: 0x811060f0
Loading Linux kernel with entry point 0x811060f0 ...
```

```
Bootloader: Done loading app on core_mask: 0xf
### Launching Linux Kernel (flags = 0x5)

Initiating Emergency Installation of bundle
tftp://172.19.211.47/cstohs/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin
Downloading bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin...
Validating bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin...
Installing bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin...
Verifying bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.09.12.EMD.150-9.12.EMD.bin...
Package cat3k_caa-base.SSA.03.09.12.EMD.pkg is Digitally Signed
Package cat3k_caa-drivers.SSA.03.09.12.EMD.pkg is Digitally Signed
Package cat3k_caa-infra.SSA.03.09.12.EMD.pkg is Digitally Signed
Package cat3k_caa-INOSd-universalk9.SSA.150-9.12.EMD.pkg is Digitally Signed
Package cat3k_caa-platform.SSA.03.09.12.EMD.pkg is Digitally Signed
Package cat3k_caa-wcm.SSA.03.09.12.EMD.pkg is Digitally Signed
Preparing flash...
Syncing device...
Emergency Install successful... Rebooting
Restarting system.
Booting...(use     DDR     clock     667     MHz)Initializing     and     Testing     RAM
+++@@@@####...++@@++@@++@@++@
```

执行设备设置的配置示例

示例：把设备配置为 DHCP 服务器

```
Device# configure terminal
Device(config)# ip dhcp pool pool1
Device(dhcp-config)# network 10.10.10.0 255.255.255.0
Device(dhcp-config)# boot config-boot.text
Device(dhcp-config)# default-router 10.10.10.1
Device(dhcp-config)# option 150 10.10.10.1
Device(dhcp-config)# exit
Device(config)# tftp-server flash:config-boot.text
Device(config)# interface gigabitethernet1/0/4
Device(config-if)# no switchport
Device(config-if)# ip address 10.10.10.1 255.255.255.0
Device(config-if)# end
```

示例：配置 DHCP 自动镜像更新

此示例在 VLAN 99 上使用一个三层 SVI 接口，启用了使用保存配置的 DHCP 自动配置。

```
Device# configure terminal
Device(config)# boot host dhcp
Device(config)# boot host retry timeout 300
Device(config)# banner config-save ^C Caution - Saving Configuration File to NVRAM May Cause
You to No longer Automatically Download Configuration Files at Reboot^C
Device(config)# vlan 99
Device(config-vlan)# interface vlan 99
Device(config-if)# no shutdown
Device(config-if)# end
Device# show boot
BOOT path-list:
Config file: flash:/config.text
Private Config file: flash:/private-config.text
Enable Break: no
Manual Boot: no
HELPER path-list:
NVRAM/Config file
buffer size: 32768
Timeout for Config
Download: 300 seconds
Config Download
via DHCP: enabled (next boot: enabled)
Device#
```

示例：计划软件镜像重载

此示例展示了如何在当前日期的 7:30 pm 重新加载设备上的软件。

```
Device# reload at 19:30
Reload scheduled for 19:30:00 UTC Wed Jun 5 2013 (in 2 hours and 25 minutes)
Proceed with reload? [confirm]
```

此示例展示了如何在未来某个时间重新加载设备上的软件。

```
Device# reload at 02:00 jun 20
Reload scheduled for 02:00:00 UTC Thu Jun 20 2013 (in 344 hours and 53 minutes)
Proceed with reload? [confirm]
```

其他参考资料

相关文档

相关主题	文档标题
设备设置命令 启动引导程序命令	*系统管理命令参考（Inspur 6650 交换机）*
预下载特性	*系统管理配置指南*
INOS DHCP 配置	*IP 编址配置指南库（Inspur 6650 交换机）*
硬件安装	*Inspur 6650 交换机硬件安装指南*
平台无关的命令参考	*配置基础命令参考，Inspur INOS（Inspur 6650 交换机）*
平台无关的配置信息	*配置基础配置指南（Inspur 6650 交换机）* *IP 编址配置指南库（Inspur 6650 交换机）*

标准和 RFC

标准/RFC	标题
	无-

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

配置 SDM 模板

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用 Inspur 特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问 Inspur 特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于配置 SDM 模板的信息

SDM 模板

可以使用 SDM 模板配置系统资源，并根据网络中的设备使用情况来优化对特定特性的支持。可以选用模板来为一些功能提供最大的系统使用情况。
设备上支持的模板如下：

- 高级模板——此版本的所有支持镜像中均可用。对于 netflow、组播组、安全 ACE、QoS ACE 等特性，模板能最大化系统资源使用。
- VLAN 模板——VLAN 模板仅对于 LAN Base 许可证可用。VLAN 模板禁用路由，支持最大数量的单播 MAC 地址。该模板通常对二层设备选用。

更改模板并重启系统之后，可以使用特权 EXEC 命令 **show sdm prefer** 来验证新的模板配置。如果在输入 **reload** 特权 EXEC 命令之前输入 **show sdm prefer** 命令，输出会显示当前在用的模板以及重启后会激活的模板。
默认使用高级模板，

表 198：模板允许的特性资源大致数量

资源	**高级**	**VLAN**
VLAN 数量	4094	4094
单播 MAC 地址	32K	32K
溢出的单播 MAC 地址	512	512
IGMP 组以及组播路由	4K	4K
溢出的 IGMP 组以及组播路由	512	512
直连路由	32K	32K
非直连 IP 主机	8K	8K
基于策略路由 ACE	1024	0
QoS 分类 ACE	3K	3K
安全 ACE	1.5K	1.5K
Netflow ACE	1024	1024
输入 Microflow 限速器 ACE	256K	256K

输出 Microflow 限速器 ACE	256K	256K
FSPAN ACE	256	256
隧道	256	0
控制层条目	512	512
输入 Netflow 流	8K	8K
输出 Netflow 流	16K	16K
SGT/DGT 条目	4K	4K
SGT/DGT 溢出条目	0	512

注释： 当交换机被用作无线移动性代理（Wireless Mobility Agent）时，仅允许使用高级模板。

表中展示了选用模板时的大致硬件限制。如果某部分硬件资源占满，所有在处理的过载都会被送往 CPU，这会严重影响交换机性能。

SDM 模板与交换机堆栈

在交换机堆栈中，所有堆栈成员都必须使用存储在活跃交换机上的相同 SDM 模板。当新交换机被添加到堆栈时，存储在活跃交换机上的 SDM 配置会覆盖独立交换机上配置的模板。可以使用特权 EXEC 命令 **show switch** 查看堆栈成员是否在 SDM 不匹配模式中。

如何配置 SDM 模板

配置 SDM 模板

配置交换机 SDM 模板

设置 SDM 模板

按照以下步骤使用 SDM 模板最大化特性使用情况：

总步骤

1. enable

2. configure terminal

3. sdm prefer { **advanced** | **vlan** }

4. end

5. reload

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device# configure terminal`	进入全局配置模式。
步骤 2	**configure terminal** **示例:** `Device# configure terminal`	进入全局配置模式。
步骤 3	**sdm prefer** { **advanced** | **vlan** } **示例:** `Device(config)# sdm prefer advanced`	指定要在交换机上使用的 SDM 模板。关键字含义如下： • **advanced**——支持高级特性，

		如 Netflow。 • **vlan**——最大化交换机上的 VLAN 配置，在硬件上不支持路由。 **注释：no sdm prefer** 命令以及默认模板不被支持。
步骤 4	**end** **示例:** `Device(config)# end`	返回特权 EXEC 模式。
步骤 5	**reload** **示例:** `Device# reload`	重新加载操作系统。

监控并维护 SDM 模板

命令	目的
show sdm prefer	显示使用的 SDM 模板。
reload	重新加载交换机，激活新配置的 SDM 模板。
no sdm prefer	设置默认 SDM 模板。

SDM 模板的配置示例

示例：配置 SDM 模板

示例：显示 SDM 模板

此示例输出显示了高级模板信息：

```
Device# show sdm prefer
Showing SDM Template Info
This is the Advanced template.
Number of VLANs: 4094
Unicast MAC addresses: 32768
Overflow Unicast MAC addresses: 512
IGMP and Multicast groups: 8192
Overflow IGMP and Multicast groups: 512
Directly connected routes: 32768
Indirect routes: 8192
Security Access Control Entries: 3072
QoS Access Control Entries: 2816
Policy Based Routing ACEs: 1024
Netflow ACEs: 1024
Input Microflow policer ACEs: 256
```

```
Output Microflow policer ACEs: 256
Flow SPAN ACEs: 256
Tunnels: 256
Control Plane Entries: 512
Input Netflow flows: 8192
Output Netflow flows: 16384
These numbers are typical for L2 and IPv4 features.
Some features such as IPv6, use up double the entry size;
so only half as many entries can be created.
Device#
```

此示例输出显示了 VLAN 模板信息：

```
Device# show sdm prefer vlan
Showing SDM Template Info
This is the VLAN template for a typical Layer 2 network.
Number of VLANs: 4094
Unicast MAC addresses: 32768
Overflow Unicast MAC addresses: 512
IGMP and Multicast groups: 8192
Overflow IGMP and Multicast groups: 512
Directly connected routes: 32768
Indirect routes: 8192
Security Access Control Entries: 3072
QoS Access Control Entries: 3072
Policy Based Routing ACEs: 0
Netflow ACEs: 1024
Input Microflow policer ACEs: 0
Output Microflow policer ACEs: 0
Flow SPAN ACEs: 256
Tunnels: 0
Control Plane Entries: 512
Input Netflow flows: 16384
Output Netflow flows: 8192
These numbers are typical for L2 and IPv4 features.
Some features such as IPv6, use up double the entry size;
so only half as many entries can be created.
Device#
```

配置 SDM 模板的特性历史与信息

版本	修改
Inspur INOS 12.2	此特性被引入。

配置系统消息日志

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用 Inspur 特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问 Inspur 特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于配置系统消息日志的信息

系统消息日志

默认情况下，交换机会把系统消息以及特权 EXEC 命令 **debug** 的输出发送给日志进程。堆栈成员可以触发系统消息。生成系统消息的堆栈成员会把按照“hostname-n”的形式（n 是交换机编号）把自己的主机名附加到消息中，并把输出重定向给活跃交换机的日志进程。虽然活跃交换机是堆栈成员，但是它不会把自己的主机名附加到系统消息上。日志进程控制着日志消息的分发，根据配置可以把消息发送到众多目的，如日志缓冲区、终端线路或 UNIXsyslog 服务器。此进程也可以把消息发送到控制台。

日志进程被禁用时，消息只会被发往控制台。消息生成时就会被发出，所以日志消息、调试输出以及提示或其他命令的输出会穿插在一起。生成消息的进程完成后消息会出现在活跃的控制台上。

可以设置消息的严重性等级，控制在控制台上展示以及发送给每个目的的消息类型。可以给日志消息加时间戳，或者设置 syslog 源地址，来增强实时调试与管理能力。有关可能出现的消息的信息，参见此版本的系统消息指南。

可以使用交换机的命令行界面（CLI）来访问记录的系统消息，也可以把消息保存到配置好的 syslog 服务器上。在单独的交换机中，交换机软件会把 syslog 消息保存到内部缓冲区，而在交换机堆栈中，消息会保存在活跃交换机上。如果单独交换机或堆栈 master 故障，除非保存到了闪存中，否则日志会丢失。

可以通过查看 syslog 服务器上的日志来远程监控系统消息，也可以通过 Telnet、控制台端口或以太网管理端口访问交换机。在交换机堆栈上，所有堆栈成员的控制台都有相同的控制台输出。

注释： syslog 的格式与 4.3 BSD UNIX 兼容。

系统日志消息格式

系统日志消息可以包含至多 80 个字符以及一个百分号（%），之前是可选配置的序号或时间戳信息。根据交换机不同，消息会按以下格式显示：

- seq no:timestamp: %facility-severity-MNEMONIC:description (hostname-n)
- seq no:timestamp: %facility-severity-MNEMONIC:description

消息百分号之前的内容取决于以下全局配置命令的设置：

- **service sequence-numbers**
- **service timestamps log datetime**
- **service timestamps log datetime** [**localtime**] [**msec**] [**show-timezone**]
- **service timestamps log uptime**

表 199：系统日志消息元素

元素	描述
seq no:	如果配置了全局配置命令 **service sequence-numbers**，会给消息标记上序号。
timestamp formats: *mm/dd h h:mm:ss* 或 *hh:mm:ss* (短形式启用时间) 或 *d h* (长形式启用时间)	消息或事件的日期与时间。此信息只在配置了全局配置命令**service timestampslog** [**datetime** | **log**]时出现。
facility	消息涉及的设备（如 SNMP、SYS 等）。
severity	表示消息严重性的数字代码，从 0 到 7。
MNEMONIC	唯一描述消息的文本串。
description	包含报告事件详细信息的文本串。
hostname-n	堆栈成员的主机名及其在堆栈中的交换机编号。虽然活跃交换机是堆栈成员，但它*不会*把自己的主机名附加到系统消息中。

默认系统消息日志设置

表 200：默认系统消息日志设置

特性	默认设置
系统消息日志输出到控制台	启用。
控制台严重级别	调试（Debugging）。
日志文件配置	未指定文件名。
日志缓冲区大小	4096 字节。
日志历史大小	1 个消息。
时间戳	禁用。
同步记录	禁用。
日志服务器	禁用。
syslog 服务器 IP 地址	未配置。
服务器设备	Local7
服务器严重级别	信息（Informational）。

Syslog 消息限制

如果使用全局配置命令 **snmp-server enable trap** 配置给 SNMP 网络管理工作站发送 syslog 消息陷阱，可以更改发送消息的等级以及储存在交换机历史表中的消息等级。也可以更改储存在历史表中的消息数量。

因为 SNMP 陷阱不保证能送达目的，所以消息会被保存在历史表中。默认情况下，即使没有启用 syslog 陷阱，历史表中也会储存一个 **warning** 级别以及数值更低级别的消息。

当历史表存满时（表中存储了全局配置命令**logginghistory size**指定的最大数量消息条目），最老的消息条目会被从表中删除，以允许存储新消息。

历史表中列出了消息的级别关键字以及安全性等级。对于SNMP，安全性等级值以1递增。例如，*emergencies*等于1，*critical*等于3。

如何配置系统消息日志

设置显示消息的目的设备

如果启用消息日志功能，可以把消息发送给控制台以外的其他位置。此任务是可选的。

总步骤

1. **configure terminal**
2. **logging buffered** [*size*]
3. **logging** *host*
4. **logging file flash:** *filename* [*max-file-size* [*min-file-size*]] [*severity-level-number* | *type*]
5. **end**
6. **terminal monitor**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**logging buffered** [*size*] **示例：** `Device(config)#` **logging buffered8192**	对于单独交换机，日志消息记录在交换机的内部缓冲区；对于交换机堆栈，日志消息记录在活跃交换机的内部缓冲区。范围从 4096 到 2147483647 字节。默认缓冲区大小是 4096 字节。 如果单独交换机或者活跃交换机故障，日志文件会丢失，除非之前把它保存在闪存内存中，见步骤 4。 **注释：**不要把缓冲区大小设置的太大，因为交换机可能耗尽其他任务所需的内存。使用特权 EXEC 命令

		show memory 查看交换机处理器的空闲内存空间。不过显示的值是最大可用值，所以缓冲区大小不应该设置为此值。
步骤 3	**logging** *host* **示例：** `Device(config)#` **logging125.1.1.100**	把日志消息发送到 UNIX syslog 服务器主机。 *host* 指定了用作 syslog 服务器主机的名称或 IP 地址。要创建接收日志消息的 syslog 服务器列表，请多次输入此命令。
步骤 4	**logging file flash:** *filename*[*max-file-size* [*min-file-size*]][*severity-level-number* | *type*] **示例：** `Device(config)#` **logging file flash:log_msg.txt 40960 4096 3**	对于单独交换机，日志消息记录在交换机的内存文件中；对于交换机堆栈，日志消息记录在活跃交换机的内存文件中。 • *filename*——输入日志消息文件名。 • （可选）**max-file-size**——指定最大日志文件大小。范围从 4096 到 2147483647 字节，默认值是 4096 字节。 • （可选）*min-file-size*——指定最小日志文件大小。范围从 1024 到 2147483647 字节，默认值是 2048 字节。 • （可选）*severity-level-number* | *type*——指定日志安全性等级或日志类型。安全性等级范围从到 7。
步骤 5	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 6	**terminal monitor** **示例：** `Device#` **terminal monitor**	在当前会话期间把日志消息发送到非控制台的终端。 终端参数设置命令在本地进行设置，且会话结束后失效。要查看调试消息，必须为每个会话执行此步骤。

同步日志消息

对于特定的控制台端口线路或虚拟终端线路，可以对非请求消息、特权 EXEC 命令 **debug** 输出以及请求设备消息、提示进行同步。可以基于安全性等级，指定要异步输出的消息类型。也可以为终端配置存储异步消息的最大缓冲区数量，超过此数量后消息会被丢弃。

对非请求的日志消息以及 **debug** 命令输出进行同步控制时，控制台上的非请求设备输出会在请求设备输出之后出现。控制台上的非请求消息以及 **debug** 命令输出会出现在用户输入提示返回之后。因此，非请求消息以及 **debug** 命令输出就不会与请求命令输出及提示穿插在一起。在非请求消息显示之后，控制台会再次显示用户提示符。
此任务是可选的。

总步骤

1. configure terminal

2. line [**console** | **vty**] *line-number* [*ending-line-number*]

3. logging synchronous [**level** [*severity-level* | **all**] | **limit** *number-of-buffers*]

4. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**line** [**console** | **vty**] *line-number*[*ending-line-number*] **示例:** `Device(config)#` **line console**	指定要配置的同步输出日志消息的线路。 • **console**——指定消息通过交换机控制台端口或以太网管理端口费阿松。 • **line vty** *line-number*——指定哪个 vty 线路要启用同步日志控制。可以对 Telnet 会话进行的配置使用 vty 连接。线路编号范围从 0 到 15。 输入以下命令可以一次更所全部 16 个 vty 线路的设置： **line vty 0 15** 也可以更改当前连接使用的 vty 线路设置。比如，要更改 vty 线路 2 的设置，输入： **line vty 2** 输入此命令时，配置模式更改为线路配置模式。
步骤 3	**logging synchronous** [**level**[*severity-level* | **all**] | **limit***number-of-buffers*] **示例:** `Device(config)#` **loggingsynchronous level 3 limit 1000**	启用日志消息同步控制。 • （可选）**level** *severity-level*——指定消息安全性等级。安全性等级等于或大于此值的消息会异步打印。数字越小安全性等级越高。默认值是 2。 • （可选）**level all**——指定所有消息都异步输出，无论安全性等级如何。 • （可选）**limit** *number-of-buffers*——

		指定终端的缓冲区数量，超过此数量的消息会被丢弃。范围从 0 到 2147483647，默认值是 20。
步骤 4	**end** **示例:** `Device(config)# end`	返回特权 EXEC 模式。

禁用消息日志

消息日志默认被启用。要把消息发送给控制台以外的目的地，消息日志必须被启用。启用时，日志消息会被发送给日志进程，该进程会把消息异步发送给指定的位置。
禁用日志进程可能会减慢交换机速度，因为进程必须等待消息写到控制台之后才能继续运行。禁用日志进程后，消息产生后会立刻显示在控制台上，通常会出现在命令输出之间。
全局配置命令 **logging synchronous** 也会影响控制台的消息显示。启用个该命令时，消息仅在用户输入**回车**之后显示。
要在禁用消息日志后重新启用，需使用全局配置命令 **logging on**。此任务是可选的。

总步骤

1. configure terminal
2. no logging console
3. end

具体步骤

	命令或操作	**目的**
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 2	**no logging console** **示例：** `Device(config)# no logging console`	禁用消息日志。
步骤 3	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式。

启用及禁用日志消息的时间戳

默认情况下，日志消息不加时间戳。
此任务是可选的。

总步骤

1. configure terminal
2. 使用以下命令之一：

- **service timestamps log uptime**
- **service timestamps log datetime**[**msec** | **localtime** | **show-timezone**]

3. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	使用以下命令之一： **• service timestamps log uptime** **• service timestamps log datetime**[**msec** | **localtime** | **show-timezone**] **示例：** `Device(config)#` **service timestamps log uptime** 或 `Device(config)#` **service timestamps log datetime**	启用日志时间戳。 • **log uptime**——在日志消息上加时间戳，显示系统重启后经过的时间。 • **log datetime**——在日志消息上加时间戳。根据选择选项不同，时间戳可以包含日期、相对于本地时区的毫秒形式时间以及时区名。
步骤 3	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。

启用及禁用日志消息序号

如果有多个时间戳相同的消息，可以显示带有序号的消息。默认情况下日志消息的序号不被显示。
此任务是可选的。

总步骤

1. configure terminal

2. service sequence-numbers

3. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**service sequence-numbers** **示例：** `Device(config)#` **service sequence-numbers**	启用序号。
步骤 3	**end**	返回特权 EXEC 模式。

	示例： `Device(config)#` **end**	

定义消息安全性等级

可以通过指定消息的等级来限制显示到所选设备的消息。
此任务是可选的。

总步骤

1. **configure terminal**
2. **logging console** *level*
3. **logging monitor** *level*
4. **logging trap** *level*
5. **end**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**logging console** *level* **示例：** `Device(config)#` **logging console 3**	限制显示到控制台的日志消息。 默认时，控制台接收 debugging 消息以及更低数值等级的消息。
步骤 3	**logging monitor** *level* **示例：** `Device(config)#` **logging monitor 3**	限制输出到终端线路的日志消息。 默认时，终端线路接收 debugging 消息以及更低数值等级的消息。
步骤 4	**logging trap** *level* **示例：** `Device(config)#` **logging trap 3**	限制输出到 syslog 服务器的日志消息。 默认时，syslog 服务器接受 informational 消息以及更低数值等级的消息。
步骤 5	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。

限制发送到历史表以及 SNMP 的 Syslog 消息

此任务展示了如何限制发送到历史表以及 SNMP 的 syslog 消息。

总步骤

1. **configure terminal**
2. **logging history** *level*

3. logging history size *number*

4. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**logging history** *level* **示例：** `Device(config)#` **logging history 3**	更改存储到历史表以及发送到 SNMP 服务器的 syslog 消息等级。默认发送 **warnings**、**errors**、**critical**、**alerts** 及 **emergencies** 的消息。
步骤 3	**logging history size** *number* **示例：** `Device(config)#` **logging history size 200**	指定存储在历史表中的消息数量。默认存储 1 条消息，配置范围从 0 到 500。
步骤 4	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。

发送日志消息给 UNIX Syslog 守护进程

此任务是可选的。

注释： 一些新版本的 UNIX syslog 守护进程不再接受来自网络的默认 syslog 数据包。如果使用的系统是这种情况，请使用 **man syslogd** UNIX 命令来决定为了启用远程 syslog 消息记录需要添加或删除哪些 syslog 命令。

在开始前

- 登录为 root。
- 在给 UNIX syslog 服务器发送系统日志消息之前，必须在 UNIX 服务器上配置 syslog 守护进程。

总步骤

1. 在/etc/syslog.conf 文件中添加一行命令。
2. 在 UNIX shell 中输入相关命令。
3. 确保 syslog 守护进程能读到新的配置变化。

具体步骤

	命令或操作	目的
步骤 1	在/etc/syslog.conf 文件中添加一行命令。 **示例：** **local7.debug /usr/adm/logs/*inspur.log***	• **local7**——指定日志记录设置。 • **debug**——指定 syslog 等级。此文件必须已经存在，且 syslog 守护进程必须有权限进行写文件。

步骤 2	在 UNIX shell 中输入相关命令。 **示例：** $ **touch /var/log/*inspur.log*** $ **chmod 666 /var/log/*inspur.log***	创建日志文件。syslog 守护进程会把此等级或更严重等级的消息发送到此文件中。
步骤 3	确保 syslog 守护进程能读到新的配置变化。 **示例：** $ **kill -HUP `cat /etc/syslog.pid`**	更多信息参见UNIX系统的**man syslog.conf**以及**mansyslogd**命令。

监控并维护系统消息日志

监控存档日志的配置

命令	目的
show archive log config {**all** | **number**[*end-number*] | **user** *username* [**session***number*] *number* [*end-number*] | **statistics**}[**provisioning**]	显示整个配置 log 或参数指定的日志。

系统消息日志配置示例

示例：堆栈系统消息

此示例展示了活跃交换机以及一个堆栈成员（*Switch-2*）的部分交换机系统消息：

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet1/0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet1/0/1,
changedstate to down 2
*Mar 1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar 1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2
(10.34.195.36)
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up (Switch-2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/1, changed state to up (Switch-
2)
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet2/0/2, changed state to up (Switch-
2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
(Switch-2)
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet2/0/1,
```

```
changedstate to down 2 (Switch-2)
```

示例：交换机系统消息

此示例展示了交换机上的部分交换机系统消息：

```
00:00:46: %LINK-3-UPDOWN: Interface Port-channel1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/1, changed state to up
00:00:47: %LINK-3-UPDOWN: Interface GigabitEthernet0/2, changed state to up
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan1, changed state to down
00:00:48: %LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed
stateto down 2
*Mar 1 18:46:11: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
18:47:02: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
*Mar 1 18:48:50.483 UTC: %SYS-5-CONFIG_I: Configured from console by vty2 (10.34.195.36)
```

其他参考资料

相关文档

相关主题	文档标题
系统管理命令	*系统管理命令参考（Inspur 6650 交换机）*
平台无关的命令参考	*配置基础命令参考，Inspur INOS（Inspur 6650 交换机）*
平台无关的配置信息	*IP 编址配置指南库，Inspur INOS（Inspur 6650 交换机）* *配置基础配置指南（Inspur 6650 交换机）*

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

系统消息日志的特性历史与信息

版本	特性信息
Inspur INOS 12.2	引入了此特性。

配置在线诊断

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用 Inspur 特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问 Inspur 特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于配置在线诊断的信息

在线诊断

在设备连接到现有网络时，可以通过在线诊断功能测试并验证设备的硬件功能。
在线诊断包含数据包交换测试，能检查不同的硬件组件并验证数据通路以及控制信令。
在线诊断能检测以下方面的内容：

- 硬件组件
- 接口（以太网端口等）
- 焊接接头

在线诊断分为按需诊断、计划诊断以及健康监测三类。按需诊断通过 CLI 运行；计划诊断在设备连接到现有网络时按照用户指定的间隔运行或在特定的时间运行；健康监测在后台按照用户定义的间隔运行。默认时，健康监测测试每 30 秒运行一次。
配置了在线诊断之后，可以手动开始诊断测试或显示测试结果。也可以看到已经为设备或交换机堆栈配置了哪些测试，以及已经运行过哪些诊断测试。

如何配置在线诊断

开始在线诊断测试

在设备上配置运行诊断测试之后，可以使用特权 EXEC 命令 **diagnostic start** 执行诊断测试。测试开始后，用户不能停止测试进程。
使用特权 EXEC 命令手动开始在线诊断测试：

总步骤

1. diagnostic start switch *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **complete** | **minimal** |**non-disruptive** | **per-port**}

具体步骤

	命令或操作	目的
步骤 1	**diagnostic start switch** *number* **test** {*name*| *test-id* | *test-id-range* | **all** | **basic** | **complete**| **minimal** | **non-disruptive** | **per-port**} **示例：** `Device# diagnostic start switch 2test basic`	开始诊断测试。 **switch** *number* 关键字仅在堆栈设备上支持。范围从 1 到 4。 可以使用以下选项之一指定测试参数： • *name*——输入测试名称。 • *test-id*——输入测试 ID 编号。 • *test-id-range*——输入测试 ID 范围，由逗号及连字符分隔的整数表示。 • **all**——开始所有测试。 • **basic**——开始基本测试集。 • **complete**——开始完整测试集。 • **minimal**——开始最小启动测试集。 • **non-disruptive**——开始无干扰测试集。 • **per-port**——开始基于端口的测试集。

配置在线诊断

在启用诊断监控之前，必须配置失败门限值以及测试间隔。

计划在线诊断

可以计划设备在线诊断的运行时间，在一天中指定的时间运行测试，或每天、每周或每月运行一次。使用命令的 **no** 形式移除计划。

总步骤

1. configure terminal

2. diagnostic schedule switch *number* **test** {*name* | *test-id* | *test-id-range* | **all** | **basic** | **complete** | **minimal**| **non-disruptive** | **per-port**} {**daily** | **on** *mm dd yyyy hh:mm* | **port** *inter-port-number port-number-list* |**weekly** *day-of-week hh:mm*}

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**diagnostic schedule switch** *number* **test**{*name* | *test-id* | *test-id-range* | **all** | **basic** |**complete** | **minimal** | **non-disruptive** |**per-port**} {**daily** | **on** *mm dd yyyy hh:mm* |**port** *inter-port-number port-number-list* |**weekly** *day-of-week hh:mm*} **示例：** `Device(config)#` **diagnostic scheduleswitch 3 test 1-5 on July 3 2013 23:10**	计划在指定的日期和时间执行按需诊断测试。 **switch** *number* 关键字仅在堆栈设备上支持。范围从 1 到 4。 可以使用以下选项之一指定测试计划参数： • *name*——命令 **show diagnostic content** 输出显示的测试名称。 • *test-id*——命令 **show diagnostic content** 输出显示的测试 ID 编号。 • *test-id-range* ——命令 **show diagnostic content** 输出显示的测试 ID 编号。 • **all**——所有测试 ID。 • **basic**——开始基本按需诊断测试。 • **complete**——开始完整测试集。 • **minimal**——开始最小启动测试集。 • **non-disruptive**——开始无干扰测试集。 • **per-port**——开始基于端口的测试集。 可以计划测试执行时间： • 每天——使用 **daily** *hh:mm* 参数。 • 指定日志与时间——使用 **on** *mm dd yyyy hh:mm* 参数。 • 每周——使用 **weekly** *day-of-week hh:mm* 参数。

配置健康监测诊断

可以在设备连接到现有网络时配置设备进行健康监测诊断测试。可以配置每个健康监测测试的执行间隔，让设备产生测试失败 syslog 消息，并启用特定的测试。
使用此命令的 **no** 形式来禁用测试。
默认情况下，健康监测被禁用，但是设备会在测试失败时产生 syslog 消息。
按照以下步骤配置并启用健康监测诊断测试：

总步骤

1. enable

2. configure terminal

3. diagnostic monitor interval switch *number* **test** {*name* | *test-id* | *test-id-range* | **all**} *hh:mm:ss milliseconds day*

4. diagnostic monitor syslog

5. diagnostic monitor threshold switch *number* **test** {*name* | *test-id* | *test-id-range* | **all**} **failure count** *count*

6. diagnostic monitor switch *number* **test** {*name* | *test-id* | *test-id-range* | **all**}

7. end

8. show running-config

9. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**diagnosticmonitor interval switch** *number***test** {*name* | *test-id* | *test-id-range* | **all**}*hh:mm:ss milliseconds day* **示例：** `Device(config)#` **diagnostic monitorinterval switch 2 test 1 12:30:00 7505**	配置指定健康监测测试的执行间隔。 **switch** *number* 关键字仅在堆栈设备上支持。范围从 1 到 4。 可以使用以下选项之一指定测试计划参数： • *name*——命令 **show diagnostic content** 输出显示的测试名称。 • *test-id*——命令 **show diagnostic content** 输出显示的测试 ID 编号。 • *test-id-range* ——命令 **show diagnostic content** 输出显示的测试 ID 编号。 • **all**——所有诊断测试。 指定间隔请设置以下参数： • *hh:mm:ss*——按照小时、分钟、秒的形式设置监控间隔。*hh* 的

		范围从 0 到 24，*mm* 和 *ss* 的范围从 0 到 60。 • *milliseconds*——毫秒格式（ms）的监控间隔。范围从 0 到 999。 • *day*——监控间隔天数。范围从 0 到 20。
步骤 4	**diagnostic monitor syslog** **示例：** `Device(config)#` **diagnostic monitorsyslog**	（可选）配置交换机在健康监测测试失败时生成 syslog 消息。
步骤 5	**diagnostic monitor threshold switch***number* **test** {*name* | *test-id* | *test-id-range* |**all**} **failure count** *count* **示例：** `Device(config)#` **diagnostic monitorthreshold switch 2 test 1 failurecount 20**	（可选）设置健康监测测试的故障门限值。 **switch** *number* 关键字仅在堆栈交换机上支持。范围从 1 到 9。 可以使用以下选项之一指定测试计划参数： • *name*——命令 **show diagnostic content** 输出显示的测试名称。 • *test-id*——命令 **show diagnostic content** 输出显示的测试 ID 编号。 • *test-id-range* ——命令 **show diagnostic content** 输出显示的测试 ID 编号。 • **all**——所有诊断测试。 失败门限值 *count* 范围从 0 到 99。
步骤 6	**diagnostic monitor switch** *number* **test**{*name* | *test-id* | *test-id-range* | **all**} **示例：** `Device(config)#` **diagnostic monitorswitch 2 test 1**	启用指定的健康监测测试。 **switch** *number* 关键字仅在堆栈交换机上支持。范围从 1 到 9。 可以使用以下选项之一指定测试计划参数： • *name*——命令 **show diagnostic content** 输出显示的测试名称。 • *test-id*——命令 **show diagnostic content** 输出显示的测试 ID 编号。 • *test-id-range* ——命令 **show diagnostic content** 输出显示的测试 ID 编号。 • **all**——所有诊断测试。
步骤 7	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 8	**show running-config** **示例：**	验证配置的条目。

	Device# **show running-config**	
步骤 9	**copy running-config startup-config** **示例：** Device# **copy running-configstartup-config**	（可选）把配置的条目保存到配置文件中。

接下来做什么？

使用全局配置命令**no diagnostic monitor interval test***test-id* | *test-id-range* }把间隔更改为默认值或者0。使用**no diagnostic monitor syslog**命令禁止在健康监测测试失败时产生syslog消息。使用**diagnostic monitor threshold test***test-id* |*test-id-range* }**failure count**命令移除失败门限值。

监控及维护在线诊断

显示在线诊断测试及测试结果

可以显示为设备或设备堆栈配置的在线诊断测试，并可以使用下表中的特权 EXEC **show** 命令检查测试结果：

表 201：诊断测试配置及结果的命令

命令	目的
show diagnostic content switch [*number* | **all**]	显示为交换机配置的在线诊断测试。 **switch** [*number* | **all**]参数仅支持在堆栈交换机上使用。
show diagnostic status	显示当前运行的诊断测试。
show diagnostic result switch [*number* | **all**] [**detail**| **test** {*name* | *test-id* | *test-id-range* | **all**} [**detail**]]	显示在线诊断测试结果。 **switch** [*number* | **all**]参数仅支持在堆栈交换机上使用。
show diagnostic switch [*number* | **all**] [**detail**]	显示在线诊断测试结果。 **switch** [*number* | **all**]参数仅支持在堆栈交换机上使用。
show diagnostic schedule switch [*number* | **all**]	显示计划在线诊断测试结果。 **switch** [*number* | **all**]参数仅支持在堆栈交换机上使用。
show diagnostic post	显示 POST 结果（输出与 **show post** 命令输出相同）。

在线诊断测试配置示例

示例：开始诊断测试

此示例展示了如何通过测试名称开始诊断测试：

Device# **diagnostic start switch 2 test TestInlinePwrCtlr**

此示例展示了如何开始所有基本诊断测试：

```
Device# diagnostic start switch 1 test all
```

示例：配置健康监测测试

此示例展示了如何配置健康监测测试：

```
Device(config)# diagnostic monitor threshold switch 1 test 1 failure count 50
Device(config)# diagnostic monitor interval switch 1 test TestPortAsicStackPortLoopback
```

示例：计划诊断测试

此示例展示了如何在特定交换机上配置计划诊断测试在指定的日期与时间运行：

```
Device(config)# diagnostic schedule test DiagThermalTest on June 3 2013 22:25
```

此示例展示了如何在特定交换机上配置计划诊断测试每周在特定时间运行：

```
Device(config)# diagnostic schedule switch 1 test 1,2,4-6 weekly saturday 10:30
```

示例：显示在线诊断

此示例展示了如何展示按需诊断设置：

```
Device# show diagnostic ondemand settings
Test iterations = 1
Action on test failure = continue
```

此示例展示了如何显示诊断事件错误：

```
Device# show diagnostic events event-type error
Diagnostic events (storage for 500 events, 0 events recorded)
Number of events matching above criteria = 0
No diagnostic log entry exists.
```

此示例展示了如何显示诊断测试的描述：

```
Device# show diagnostic description switch 1 test all
DiagGoldPktTest :
The GOLD packet Loopback test verifies the MAC level loopback
functionality. In this test, a GOLD packet, for which doppler
provides the support in hardware, is sent. The packet loops back
at MAC level and is matched against the stored packet. It is a non
-disruptive test.
DiagThermalTest :
This test verifies the temperature reading from the sensor is below the yellow
temperature threshold. It is a non-disruptive test and can be run as a health
monitoring test.
DiagFanTest :
This test verifies all fan modules have been inserted and working properly on the
board
It is a non-disruptive test and can be run as a health monitoring test.
DiagPhyLoopbackTest :
The PHY Loopback test verifies the PHY level loopback
functionality. In this test, a packet is sent which loops back
```

```
at PHY level and is matched against the stored packet. It is a
disruptive test and cannot be run as a health monitoring test.
DiagScratchRegisterTest :
The Scratch Register test monitors the health of application-specific
integrated circuits (ASICs) by writing values into registers and reading
back the values from these registers. It is a non-disruptive test and can
be run as a health monitoring test.
DiagPoETest :
This test checks the PoE controller functionality. This is a disruptive test
and should not be performed during normal switch operation.
DiagStackCableTest :
This test verifies the stack ring loopback functionality
in the stacking environment. It is a disruptive test and
cannot be run as a health monitoring test.
DiagMemoryTest :
This test runs the exhaustive ASIC memory test during normal switch operation
NG3K utilizes mbist for this test. Memory test is very disruptive
in nature and requires switch reboot after the test.
Device#
```

此示例展示了如何显示启动等级：

```
Device# show diagnostic bootup level
Current bootup diagnostic level: minimal
Device#
```

其他参考资料

相关文档

相关主题	文档标题
系统管理命令	*系统管理命令参考（Inspur 6650 交换机）*
平台无关的命令参考	*配置基础命令参考，Inspur INOS（Inspur 6650 交换机）*
平台无关的配置信息	*IP 编址配置指南库，Inspur INOS（Inspur 6650 交换机）* *配置基础配置指南（Inspur 6650 交换机）*

标准和 RFC

标准/RFC	标题
无	-

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文	http://www.icntnetworks.com

档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	

配置在线诊断的特性历史与信息

版本	特性信息
Inspur INOS 12.2	引入了此特性。

软件配置故障排除

本章介绍了如何发现并解决交换机上与 Inspur INOS 系统相关软件问题。根据问题的性质，用户可以使用命令行界面（CLI）、设备管理器或网络助手来发现并解决问题。
至于其他的故障排除信息，如 LED 说明，会在硬件安装指南中提供相关的说明。

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

关于软件配置故障排除的信息

交换机上的软件故障

交换机在升级期间如果下载了错误的文件或者删除了镜像文件，那么可能会导致交换机软件的损坏。在所有类似的情况下，交换机将无法通过开机自检（power-on self-test，POST），并且无法提供连通性。

设备密码遗忘或丢失

在交换机通电启动期间，设备的默认配置允许那些对设备具有物理访问权限的终端用户通过中断引导程序并输入新密码而恢复使用。这些恢复过程要求用户具有对设备的物理访问权限。

注释： 在这些设备上，系统管理员可以禁用某些特性的一些功能，只有当终端用户同意设备恢复出厂设置，终端用户才能为设备重设密码。如果终端用户想在密码恢复功能被禁用的时候重设密码，系统会有一个状态消息来提醒用户，设备在恢复的过程中会回到出厂设置。

以太网供电端口

如果交换机检测到电路上没有电，以太网供电（Power over Ethernet，PoE）的交换机端口会自动为以下这些连接的设备供电：

- Inspur 准标准化用电设备（例如 Inspur IP 电话或 Inspur Aironet 接入点）
- 兼容 IEEE 802.3af 的用电设备
- 兼容 IEEE 802.3at 的用电设备

当用电设备连接到以太网供电的交换机端口以及 AC（交流）电源时，设备可以有冗余电力。只连接到 PoE 端口时，设备没有冗余电力。

在交换机检测到用电设备之后，交换机会去确定用电设备功率需求，然后决定是否准许为设备供电。交换机还可以通过监视和限制电量的使用情况，来检测设备的实时功耗。

如要查看更多详细信息，请参阅*接口和硬件组件配置指南（Inspur 6650 交换机）*中的“配置 PoE”一章。

断电导致端口被禁用

如果一个连接到 PoE 设备端口并且由 AC 电源供电的用电设备（例如 Inspur IP 电话 7910）失去了 AC 电源供电，该设备可能会进入错误禁用的状态。要从错误禁用状态中恢复，请输入 **shutdown** 接口配置命令，然后输入 **no shutdown** 接口命令。管理员还可以在设备上配置自动恢复，使得设备恢复正常运行。

在一个设备上，**errdisable recovery cause loopback** 和 **errdisable recovery interval***seconds* 全局配置命令支持在特定的时间段后，使得端口自动从错误禁用状态中恢复。

错误 Link-Up 导致端口被禁用

在用电设备已经连接到某端口的情况下，如果用户使用 **power inline never** 接口配置命令配置此端口，可能会出现错误 link-up 状态，端口也将进入错误禁用状态。要使端口从错误禁用状态中恢复，输入 **shutdown** 和 **no shutdown** 接口配置命令。

请勿将 Inspur 用电设备连接到已使用 **power inline never** 命令进行配置的端口。

Ping

设备支持 IP ping，用户可以使用该命令测试与远程主机的连接性。Ping 向目的地址发送一个回显请求包，并等待回复。Ping 会返回的响应如下：

- 正常响应——正常响应（*主机名*是存活的）发生在 1 到 10 秒内，具体取决于网络流量状况。
- 目标不响应——如果主机没有响应，则返回*无应答*消息。
- 未知主机——如果主机不存在，则返回*未知主机*消息。
- 目的地不可达——如果默认网关不能到达指定的网络，则返回*目的地不可达*消息。
- 网络或主机不可达——如果主机或网络的路由表中没有路由表，则返回*网络或主机不可达*的消息。

二层 Traceroute

第 2 层 traceroute 特性允许交换机识别数据包从源设备到目标设备的物理路径。二层 traceroute 只支持单播的源目 MAC 地址。

traceroute 使用路径上设备的 MAC 地址表查找路径。当设备检测到路径中有不支持第 2 层 traceroute 的设备时，设备继续发送第 2 层 trace 查询请求并使其超时。

设备只能识别从源设备到目的设备的路径。它不能识别从源主机到源设备或从目的设备到目的主机的路径。

二层 Traceroute 指南

- 网络中的所有设备必须启用 Inspur 发现协议（Cisco Discovery Protocol，CDP）。如果要使二层 traceroute 正常工作，请不要禁用 CDP。
  如果物理路径中的某些设备对 CDP 透明，交换机则不能识别经过这些设备的路径。
- 当管理员使用特权 EXEC**ping** 命令可检测到两设备间的连通性，则说明一个设备到另一个设备具有可达性。物理路径中的所有设备必须彼此可达。
- 路径中的最大跳数为 10。
- 在不位于从源设备到目的设备的物理路径的交换机上，管理员可输入特权 EXEC**traceroute mac** 或 **traceroute mac ip** 命令。其中，这个交换机必须可以连通路径中的所有设备。
- 只有当指定的源目 MAC 地址属于同一 VLAN 时，**traceroute mac** 命令才会输出二层的路径。如果指定的源目 MAC 地址属于不同 VLAN，则无法识别第 2 层路径，并输出错误消

息。

- 如果指定了组播源 MAC 地址或目的 MAC 地址，路径不会被标识出，且会输出错误消息。
- 如果源或目的 MAC 地址属于多个 VLAN，必须指定源目 MAC 地址同时属于的 VLAN。如果不指定此 VLAN，路径不会被标识出，且会输出错误消息。
- 当指定的源目 IP 地址属于同一子网时，**traceroute mac ip** 命令输出二层路径。当管理员指定 IP 地址时，设备使用地址解析协议（Address Resolution Protocol，ARP）将 IP 地址与相应的 MAC 地址和 VLAN ID 关联起来。
    - 如果指定 IP 地址存在 ARP 表项，设备将使用关联的 MAC 地址并标识该物理路径。
    - 如果 ARP 表项不存在，设备将发送 ARP 查询并尝试解析 IP 地址。如果无法解析 IP 地址，则无法识别路径，并显示错误消息。
- 当多个设备通过集线器连接到一个端口时（例如，在端口上检测到多个 CDP 邻居），无法支持二层 traceroute 特性。当端口检测到多个 CDP 邻居时，无法识别二层路径，并显示错误消息。
- 令牌环 VLAN 中不支持二层 traceroute 特性。

IP Traceroute

管理员可以使用 IP traceroute 追踪数据包通过网络时的逐跳路径。该命令的输出会显示数据包在到达目的地的途中经过的所有网络层（第 3 层）设备，例如路由器。

管理员的设备可以作为 **traceroute** 特权 EXEC 命令的源或目的设备，并且可选择是否作为其中一跳在 **traceroute** 命令输出中显示。如果该设备是 traceroute 的目的设备，在 traceroute 输出中它将显示为最终目的设备。如果中间设备仅将数据包从某端口桥接到同 VLAN 中的另一端口，该设备将不会在 traceroute 输出中显示。但是，如果中间设备是一个多层的、为特定数据包进行路由的设备，该设备将会作为一跳在 traceroute 输出中显示。

traceroute 特权 EXEC 命令通过使用 IP 头中的生存时间（Time to live，TTL）字段，让路由器和服务器生成特定的返回消息。Traceroute 首先向目的主机发送 TTL 字段置为 1 的用户数据报协议（User Datagram Protocol，UDP），如果路由器发现 TTL 值为 1 或 0，则丢弃该数据报并向发送端发送网络控制消息协议（Internet Control Message Protocol，ICMP）生存时间超时的消息。Traceroute 通过检查 ICMP 生存时间超时消息的源地址字段来找出第一跳的地址。

为了识别下一跳，traceroute 发送一个 TTL 值为 2 的 UDP 包。第一个路由器将 TTL 字段减 1，并将数据报送至下一个路由器。第二个路由器看到 TTL 值为 1，丢弃数据报，并将生存时间超时消息返给源。这个过程将持续到 TTL 值增到足以使得数据报到达目的主机（或者 TTL 值增到最大）。

为了了解数据报何时到达其目的地，traceroute 将数据报中 UDP 的目的端口号设置为目的主机不太可能使用的超大值。当主机接收到包含本地未使用的目的端口号的数据报时，它会向源端发送 ICMP *端口不可达*错误。因为除了端口不可达之外的所有错误都来自中间跳，所以接收到端口不可到达错误意味着该消息由目的端口发送。

时域反射器指南

用户可以使用时域反射器（Time Domain Reflector，TDR）特性来诊断并解决布线问题。当运

行 TDR 时，本地设备通过电缆发送信号，并将反射信号与初始信号进行比较。

TDR 支持 10/100/1000 的铜线以太网端口和千兆位以太网（100Mbps / 1 / 2.5 / 5/10 Gbps）端口。 SFP 模块端口不支持 TDR。

TDR 可以检测到以下线路问题：

- 未闭合的、断开的或切断的双绞线——电线未连接到远程设备的电线。
- 双绞线短路——电线彼此接触或与远程设备的电线接触。例如，如果双绞线的一条电线焊接到另一条电线上，则双绞线发生短路。

  如果双绞线中的一条电线未闭合，TDR 可找到未闭合点位置。

注释： 当千兆位以太网端口使用此特性时，仅当检测到电线未闭合或短路情况时才显示故障位置。

以下情况中可使用 TDR 诊断并解决线路问题：

- 设备替换
- 建立配线柜
- 当链接无法建立或不正常运行时，对两个设备之间的连接进行故障排除

运行 TDR 时，设备会在以下情况报告准确的信息：

- 用于千兆位链路的电缆是实芯电缆。
- 末端未闭合的电缆无终点。

运行 TDR 时，设备不会在以下情况报告准确的信息：

- 用于千兆位链路的电缆是双绞线电缆或与实芯电缆串联。
- 链路是 10 兆位或 100 兆位的链路。
- 电缆是绞合电缆。
- 链接伙伴是 Inspur IP 电话。
- 链接伙伴不兼容 IEEE 802.3 标准。

调试命令

注意： 由于调试的输出在 CPU 进程中被分配了较高优先级，它可能导致系统不可用。因此，仅在排除特定问题或与 Inspur 技术支持人员进行故障排除会话期间使用 **debug** 命令。最好在网络流量较低和用户较少时使用调 **debug** 命令。在此期间的调试可减少增加的 **debug** 命令处理开销影响系统使用的可能性。

所有 **debug** 命令都在特权 EXEC 模式下输入，大多数 **debug** 命令不带参数。

系统报告

系统报告或 crashinfo 文件保存的信息有助于 Inspur 技术代表人员调试 Inspur INOS 镜像失败（崩溃）的问题。因此，有必要快速地、可靠地收集具有高保真和完整性的关键崩溃信息。此外，也有必要收集并打包该信息，让其可以关联或标识特定的崩溃事件。

系统报告会在以下情况产生：

- 在交换机故障的情况下，系统报告会在故障的成员上生成；堆栈中的其他成员不会生成报告。
- 在交换机切换的情况下，仅在高可用性的（High Available，HA）成员交换机上生成系统报告；non- HA 成员交换机不会生成报告。

系统不会在重载的情况下生成报告。
在进程崩溃期间，交换机从本地收集以下信息：

1 完整进程的核心
2 跟踪日志
3 INOS 系统日志（在非活动崩溃的情况下不保证有该日志）
4 系统进程信息
5 启动日志
6 重载日志
7 某些类型的/ proc 信息

这些信息分别存储在单独的文件中，然后压缩并存到一个包中。这样方便管理员通过一个文件得到崩溃快照，然后将其移出进行分析。这个报告会在交换机切换到 rommon / bootloader 模式之前生成。
除了完整的核心和跟踪日志，其他文件都以文本形式存储。

Crashinfo 文件

默认情况下，生成的系统报告文件都将保存到/crashinfo 目录中。如果 crashinfo 分区空间不足，文件将被保存到/flash 目录。
如果要显示文件，请输入 **dir crashinfo：**命令。下面是 crashinfo 目录的输出示例：

```
Switch#dir crashinfo:
Directory of crashinfo:/
46553 drwx 1024 Jun 29 2015 14:52:09 +00:00 ap_crash
12 -rw- 0 Jan 1 1970 00:00:11 +00:00 koops.dat
11 -rw- 0 Mar 22 2013 07:50:30 +00:00 deleted_crash_files
13 -rwx 594269 Mar 22 2013 07:50:30 +00:00 crashinfo_platform_mgr_20130322-075017-UTC
14 -rw- 44 Sep 9 2015 09:28:47 +00:00 last_crashinfo
15 -rw- 355 Sep 9 2015 09:29:31 +00:00 last_systemreport_log
16 -rw- 105753 Mar 22 2013 07:50:47 +00:00 system-report_1_20130322-075017-UTC.gz
17 -rw- 39 Sep 9 2015 09:29:31 +00:00 last_systemreport
18 -rwx 585996 Mar 22 2013 08:01:58 +00:00 crashinfo_platform_mgr_20130322-080144-UTC
19 -rw- 105065 Mar 22 2013 08:02:15 +00:00 system-report_1_20130322-080144-UTC.gz
20 -rwx 3426209 Sep 9 2015 06:49:12 +00:00 crashinfo_INOSd_20150909-064754-UTC
21 -rwx 9540376 Sep 9 2015 06:49:13 +00:00 fullcore_INOSd_20150909-064754-UTC
22 -rw- 469476 Sep 9 2015 06:49:56 +00:00 system-report_1_20150909-064754-UTC.gz
23 -rwx 3425350 Sep 9 2015 09:28:47 +00:00 crashinfo_INOSd_20150909-092728-UTC
24 -rwx 9535535 Sep 9 2015 09:28:47 +00:00 fullcore_INOSd_20150909-092728-UTC
25 -rw- 459709 Sep 9 2015 09:29:28 +00:00 system-report_1_20150909-092728-UTC.gz
26 -rw- 0 Sep 22 2015 11:11:33 +00:00 tracelogs.J8C
50601 drwx 10240 Oct 28 2015 22:42:50 +00:00 tracelogs
248354816 bytes total (204800000 bytes free)
```

系统报告位于 crashinfo 目录中，格式如下：

```
system-report_[switch number]_[date]-[timestamp]-UTC.gz
```

交换机崩溃后，请检查系统报告文件。最新生成的系统报告文件名存储在 crashinfo 目录下的 last_systemreport 文件中。系统报告和 crashinfo 文件可以帮助 TAC 解决故障问题。

生成的系统报告可以使用 TFTP、HTTP 或其他选项进行复制。

```
Switch#copy crashinfo: ?
crashinfo: Copy to crashinfo: file system
flash: Copy to flash: file system
ftp: Copy to ftp: file system
http: Copy to http: file system
https: Copy to https: file system
null: Copy to null: file system
nvram: Copy to nvram: file system
rcp: Copy to rcp: file system
running-config Update (merge with) current system configuration
scp: Copy to scp: file system
startup-config Copy to startup configuration
syslog: Copy to syslog: file system
system: Copy to system: file system
tftp: Copy to tftp: file system
tmpsys: Copy to tmpsys: file system
```

复制到 TFTP 服务器的一般语法如下：

```
Switch#copy crashinfo: tftp:
Source filename [system-report_1_20150909-092728-UTC.gz]?
Address or name of remote host []? 1.1.1.1
Destination filename [system-report_1_20150909-092728-UTC.gz]?
```

可以通过发出跟踪存档命令收集堆栈中所有成员的跟踪日志。该命令提供时间段选项。命令语法如下：

```
Switch#request platform software trace archive ?
last Archive trace files of last x days
target Location and name for the archive file
```

可以收集过去 6650 天内存储在 crashinfo：或 flash：目录中的跟踪日志。

```
Switch# request platform software trace archive last ?
<1-6650> Number of days (1-6650)
Switch#request platform software trace archive last 6650 days target ?
crashinfo: Archive file name and location
flash: Archive file name and location
```

注意： 为了有足够的空间存储跟踪日志或用作其他目的，一旦系统报告或跟踪存档被拷贝出去，及时从 flash 或 crashinfo 目录下清除它们很重要。

交换机上的板载故障记录

用户可使用板载故障日志记录（OnBoard Failure Logging，OBFL）功能来收集有关设备的信息。这些信息包括正常的运行时间，温度和电压信息，它们可以帮助 Inspur 技术支持人员排除设备故障。我们建议用户保持 OBFL 启用状态，不要擦除闪存中存储的数据。

默认情况下，OBFL 处于启用状态。OBFL 收集了关于该设备和小型可插拔模块（Small Form-factor Pluggable，SFP）的信息。这些信息存储在设备中的闪存中：

- CLI 命令——在独立设备或交换机堆叠成员上输入的 OBFL CLI 命令的记录。

- 环境数据——独立设备或交换机堆栈成员以及所有连接的 FRU 设备的唯一设备标识符（Unique Device Identifier，UDI）信息：产品标识（Product Identifier，PID），版本标识（Version Identifier，VID）以及序列号。
- 消息——由独立设备或交换机堆栈成员生成的与硬件相关的系统消息的记录。
- 以太网供电（Power over Ethernet，PoE）——独立设备或交换机堆栈成员上 PoE 端口的功耗记录。
- 温度——独立设备或交换机堆栈成员的温度。
- 正常运行时间的数据——独立设备或交换机堆栈成员启动用时，设备重新启动的原因，以及设备自上次重启以来运行的时长。
- 电压——独立设备或交换机堆叠成员的系统电压。

用户应手动设置系统时钟或使用网络时间协议（Network Time Protocol，NTP）配置。
当设备运行时，用户可以使用 **show logging onboard** 特权 EXEC 命令获取 OBFL 数据。如果设备出现故障，请联系您的 Inspur 技术支持人员了解如何获取数据。
当重新启动已启用 OBFL 的设备时，在开始记录新数据之前会有 10 分钟的延迟。

风扇故障

默认情况下，该特性被禁用。当现场可更换单元（Field-Replaceable Unit，FRU）或电源中有多个风扇出现故障时，设备不会关闭，并显示以下错误消息：

Multiple fan(FRU/PS) failure detected. System may get overheated. Change fan quickly.

设备可能会过热并关闭。
要启用风扇故障特性，请输入 **system env fan-fail-action shut** 特权 EXEC 命令。如果设备中有多个风扇出现故障，设备将自动关闭，并显示以下错误消息：

Faulty (FRU/PS) fans detected, shutting down system!

在第一个风扇关闭后，如果设备检测到有第二个风扇故障，设备将会等待 20 秒后再关闭。若要重新启动设备，必须关闭后再打开。

高 CPU 使用率的可能征兆

CPU 利用率过高可能会导致以下症状，但这些症状也可能由其他原因引起：

- 生成树拓扑变化
- 由于通信中断导致的 EtherChannel 链路关闭
- 停止响应管理请求（ICMP ping，SNMP 超时，Telnet 或 SSH 会话速度变慢）
- UDLD 抖动
- 由于 SLA 响应超过可接受的门限值导致的 IP SLA 失效
- DHCP 或 IEEE802.1x 失败（如果交换机不能转发或响应请求）

如何进行软件配置故障排除

从软件故障中恢复

在开始前

恢复过程要求用户具有对交换机的物理访问权限。

此过程使用引导程序命令和 TFTP 从损坏或不正确的镜像文件中恢复。

步骤 1 在 PC 端从 incntnetworks.com 下载软件镜像文件（image.bin）。

步骤 2 将软件的镜像加载到 TFTP 服务器中。

步骤 3 将 PC 连接到交换机以太网管理端口。

步骤 4 拔下交换机电源线。

步骤 5 按下 **Mode** 按钮的同时将电源线重新连接到交换机。

步骤 6 在引导程序 (ROMMON)提示符处，确保可以 ping 通 TFTP 服务器。

a） 设置 IP 地址 **switch: set IP_ADDR**ip_address subnet_mask

示例：

```
switch: set IP_ADDR 192.0.2.123/255.255.255.0
```

b） 设置默认路由 IP 地址 **switch: set DEFAULT_ROUTER**ip_address

示例：

```
switch: set DEFAULT_ROUTER 192.0.2.1
```

c） 请确认您可以 ping 通 TFTP 服务器 **switch: ping** ip_address_of_TFTP_server

示例：

```
switch: ping 192.0.2.15
ping 192.0.2.1 with 32 bytes of data...
Host 192.0.2.1 is alive.
switch:
```

步骤 7 请确认您的恢复分区（sda9 :)中是否有恢复镜像。

在使用紧急安装特性进行恢复时需要该恢复镜像。

示例：

```
switch: dir sda9:
Directory of sda9:/
    2 drwx 1024 .
    2 drwx 1024 ..
   11 -rw- 18923068 c3850-recovery.bin
36939776 bytes available (20830208 bytes used)
switch:
```

步骤 8 在引导程序（ROMMON）提示符处，启动紧急安装特性，帮助您恢复交换机上的软件镜像。

警告：紧急安装命令将擦除整个引导的闪存！

示例：

```
Switch#
emergency-install
tftp://192.0.2.47/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin

The bootflash will be erased during install operation, continue (y/n)?y
Starting emergency recovery
(tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SPA.03.02.00.SE.150-1.EX.bin)...
Reading full image into memory.....done
Nova Bundle Image
--------------------------------------
```

```
Kernel Address : 0x6042e5cc
Kernel Size : 0x318261/3244641
Initramfs Address : 0x60746830
Initramfs Size : 0xdb0fb9/14356409
Compression Format: .mzip
Bootable image at @ ram:0x6042e5cc
Bootable image segment 0 address range [0x81100000, 0x81b80000] is in range [0x80180000, 0x90000000].
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
File "sda9:c3850-recovery.bin" uncompressed and installed, entry point: 0x811060f0
Loading Linux kernel with entry point 0x811060f0 ...
Bootloader: Done loading app on core_mask: 0xf
### Launching Linux Kernel (flags = 0x5)
Initiating Emergency Installation of bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin
Downloading bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin...
Validating bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin...
Installing bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin...
Verifying bundle
tftp://192.0.2.47/cat3k/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin...
Package cat3k_caa-base..pkg is Digitally Signed
Package cat3k_caa-drivers.SPA.03.02.00.SE.pkg is Digitally Signed
Package cat3k_caa-infra.SPA.03.02.00.SE.pkg is Digitally Signed
Package cat3k_caa-INOSd-universalk9.SPA.03.02.00.SE.pkg is Digitally Signed
Package cat3k_caa-platform.SPA.03.02.00.SE.pkg is Digitally Signed
Package cat3k_caa-wcm.SPA.03.02.00.SE.pkg is Digitally Signed
Preparing flash...
Syncing device...
Emergency Install successful... Rebooting

Restarting system.
Booting...(use DDR clock 667 MHz)Initializing and Testing RAM
+++@@@@####...++@@++@@++@@++@
```

恢复丢失或忘记的密码

在交换机通电启动期间，设备的默认配置允许那些对设备具有物理访问权限的终端用户通过中断引导程序并输入新密码而恢复使用。这些恢复过程要求用户具有对设备的物理访问权限。

注释： 在这些设备上，系统管理员可以禁用某些特性的一些功能，只有当终端用户同意设

备恢复出厂设置，终端用户才能为设备重设密码。如果终端用户想在密码恢复功能被禁用的时候重设密码，系统会发出一个状态消息来提醒用户。

总步骤

1. 将终端或 PC 连接到交换机。
2. 将仿真软件上的线路速度设置为 9600 波特。
3. 关闭独立交换机或整个交换机堆栈。
4. 将电源线重新连接到交换机或活跃交换机上。并在 15 秒内按下 **Mode** 按钮，同时系统 LED 仍然闪烁绿色。继续按下 **Mode** 按钮，直到所有系统 LED 指示灯亮起并保持不变；然后释放 **Mode** 按钮。
5. 在恢复密码后，重新加载交换机或活跃交换机。
6. 打开堆栈中其余交换机的电源。

具体步骤

步骤 1 将终端或 PC 连接到交换机。

- 将终端或 PC 与终端仿真软件连接到交换机控制台端口。如果用户要为一个交换机堆栈恢复密码，请连接到活跃交换机的控制台端口。
- 将 PC 连接到以太网管理端口。如果用户要为一个交换机堆栈恢复密码，请连接到堆栈成员的以太网管理端口。

步骤 2 将仿真软件上的线路速度设置为 9600 波特。

步骤 3 关闭独立交换机或整个交换机堆栈。

步骤 4 将电源线重新连接到交换机或活跃交换机上。并在 15 秒内按下 **Mode** 按钮，同时系统 LED 仍然闪烁绿色。继续按下 **Mode** 按钮，直到所有系统 LED 指示灯亮起并保持不变；然后释放 **Mode** 按钮。

```
Switch:
Xmodem file system is available.
Base ethernet MAC Address: 20:37:06:4d:e9:80
Verifying bootloader digital signature.
The system has been interrupted prior to loading the operating
system software, console will be reset to 9600 baud rate.
```

请继续执行*启用密码恢复的过程*部分，然后按照步骤操作。

步骤 5 在恢复密码后，重新加载交换机或活跃交换机。

在交换机上：

```
Switch>reload
Proceed with reload? [confirm] y
```

在活跃交换机上：

```
Switch>reload slot <stack-active-member-number>
Proceed with reload? [confirm] y
```

步骤 6 打开堆栈中其余交换机的电源。

启用密码恢复时的过程

如果启用了密码恢复机制，则会显示该信息：

步骤 1 初始化闪存文件系统。

```
Device: flash_init
```

步骤 2 使用以下命令忽略启动配置：

```
Device: SWITCH_IGNORE_STARTUP_CFG=1
```

步骤 3 使用闪存中的 *packages.conf* 文件启动交换机。

```
Device: boot flash:packages.conf
```

步骤 4 通过回答 **NO** 终止初始配置对话。

```
Would you like to enter the initial configuration dialog? [yes/no]: No
```

步骤 5 在交换机提示符下，进入特权 EXEC 模式。

```
Device>enable

Switch#
```

步骤 6 将启动配置复制到正在运行的配置中。

```
Device# copy startup-config running-config Destination filename [running-config]?
```

确认提示符下按回车键。现在配置文件被重新加载，用户可以修改密码。

步骤 7 输入全局配置模式并修改 **enable** 密码。

```
Device# configure terminal

Device(config)#
```

步骤 8 将正在运行的配置写入启动配置文件中。

```
Device# copy running-config startup-config
```

步骤 9 确认已启用手动引导模式。

```
Device# show boot

BOOT variable = flash:packages.conf;

Manual Boot = yes

Enable Break = yes
```

步骤 10 重新加载交换机。

```
Device# reload
```

步骤 11 将 Bootloader 参数（之前在步骤 2 和 3 中更改的）返回到其原始值。

```
Device: switch: SWITCH_IGNORE_STARTUP_CFG=0
```

步骤 12 使用闪存中的 *packages.conf* 文件启动交换机。

```
Device: boot flash:packages.conf
```

步骤 13 设备启动后，在设备上禁用手动引导。

```
Device(config)# no boot manual
```

禁用密码恢复时的过程

如果禁用了密码恢复机制，则会显示该信息：

```
The password-recovery mechanism has been triggered, but
is currently disabled. Access to the boot loader prompt
through the password-recovery mechanism is disallowed at
this point. However, if you agree to let the system be
reset back to the default system configuration, access
to the boot loader prompt can still be allowed.
Would you like to reset the system back to the default configuration (y/n)?
```

注意： 将交换机返回默认配置会导致丢失所有当前配置。建议用户联系系统管理员确认是否具有备用设备和 VLAN 配置文件。

- 如果输入 **n**（否），将正常执行引导进程，就像没有按 **Mode** 按钮；用户无法进入引导加载提示符，也无法输入新密码。用户将看到以下信息：

  Press Enter to continue.....

- 如果输入 **y**（是），将删除闪存中的配置文件和 VLAN 数据库文件。
  加载默认配置时，用户可以重置密码。

步骤 1 选择继续执行密码恢复并删除当前配置：

Would you like to reset the system back to the default configuration (y/n)? Y

步骤 2 显示闪存内容：

Device: dir flash:

设备文件系统显示：

Directory of flash:/

.

.

.i'

15494 drwx 4096 Jan 1 2000 00:20:20 +00:00 kirch

15508 –rw– 258065648 Sep 4 2013 14:19:03 +00:00

cat3k_caa–universalk9.SSA.03.12.02.EZP.150–12.02.EZP.150–12.02.EZP.bin

162196684

步骤 3 启动系统：

Device: boot

提示用户开始设置程序。如要继续执行密码恢复，请在提示符后输入 **N**：

Continue with the configuration dialog? [yes/no]: N

步骤 4 在设备提示符下，进入特权 EXEC 模式：

Device>enable

步骤 5 进入全局配置模式：

Device# configure terminal

步骤 6 更改密码：

Device(config)# enable secret password

密码可以是 1-25 位数字和字母组合的字符，可以以数字开头，区分大小写，可以使用空格但会忽略空格。

步骤 7 返回特权 EXEC 模式：

Device(config)# exit

Device#

注释： 继续执行步骤 9 之前，请开启所有连接的堆栈成员，直到它们完全初始化。

步骤 8 将正在运行的配置写入启动配置文件中：

Device# copy running–config startup–config

现在，启动配置中包含新的密码。

步骤 9 现在，用户必须重新配置交换机。如果系统管理员拥有可用的备份设备和 VLAN 配置文件，用户应该使用。

防止交换机堆栈问题

为了防止交换机堆栈问题，用户应该按以下操作执行：

- 请确保向交换机堆栈添加或从中移除的设备已关闭电源。交换机堆栈中的所有供电注意事项，请参阅硬件安装指南中的“交换机安装”一章。
- 按下堆栈成员上的 **Mode** 按钮，直到堆栈模式的 LED 亮起。设备上最后两个端口的 LED 应为绿色。根据设备型号，最后两个端口是 10/100/1000 端口或小型可插拔（Small Form-factor Pluggable，SFP）模块。如果最后两个端口的 LED 有一个或都不是绿色，说明堆栈操作没有占用全部带宽。
- 建议在管理交换机堆栈时仅使用一个 CLI 会话。在活跃交换机上使用多个 CLI 会话时要小心。在一个会话中输入的命令不会显示在其他会话中。因此，用户可能无法分别所输入命令的对话。
- 根据设备在堆栈中的位置手动分配堆栈成员编号，可以更方便地对交换机堆栈进行远程故障排除。但是，在以后用户要添加、移除或重新排列设备时，用户要记住已经为设备手动分配了编号。使用 **switch** current-stack-member-number **renumber** new-stack-member-number 全局配置命令来手动分配堆栈成员编号。

如果使用相同型号的交换机替换堆栈成员，假设新设备与被替换设备使用相同的成员编号，那么新设备使用与被替换设备完全相同的配置进行工作。

移除已打开电源的堆栈成员会导致交换机堆栈划分（分割）为两个或多个交换机堆栈，每个交换机堆栈具有相同的配置。如果用户希望交换机堆栈保持分离，请更改新创建的交换机堆栈的 IP 地址。要从交换机堆栈分区恢复，请按照以下步骤操作：

1. 将新创建的交换机堆栈的电源关闭。
2. 通过 StackWise Plus 端口将它们重新连接到原始交换机堆栈。
3. 将设备电源打开。

有关可用于监控交换机堆栈及其成员的命令，请参阅*显示交换机堆栈信息*部分。

防止自动协商不匹配

IEEE802.3ab 自动协商协议管理设备速度（10Mb/s、100Mb/s 和 1000Mb/s，不包括 SFP 模块端口）和双工（半/全双工）的设置。有些情况下，此协议未能匹配这些设置，这降低了性能。在以下情况中会发生不匹配：

- 手动设置的速度或双工参数不同于所连接端口上手动设置的速度或双工参数。
- 某端口设置为自动协商，但其连接的端口设置为全双工无自动协商。

为了最大限度地提高设备性能并保证链路通信，请在更改双工和速度设置时遵循其中一个指导方案：

- 让两个端口自动协商速度和双工信息。
- 手动设置连接两端端口的速度和双工参数。

注释： 如果远程设备未自动协商，请将两个端口上的双工设置设为匹配。即使连接的端口未自动协商，速度参数也可以自行调整。

SFP 模块安全及标识的故障排除

Inspur 小型可插拔（Small Form-factor Pluggable，SFP）模块装有 EEPROM，其中包含模块序列号、供应商名称和 ID、唯一安全代码和循环冗余校验（Cyclic redundancy check，CRC）。当 SFP 模块插入设备时，设备软件会读取 EEPROM 来确认序列号、供应商名称和供应商 ID，并重新计算安全代码和 CRC。如果序列号、供应商名称或供应商 ID、安全代码或 CRC 无效，软件将生成安全错误消息并将接口置于错误禁用状态。

注释： 安全错误消息提及到 GBIC_SECURITY 功能。设备支持 SFP 模块，不支持 GBIC 模块。从字面上看安全错误消息指的是 GBIC 接口和模块，但实际上是指 SFP 模块和模块接口。

如果用户正在使用非 Inspur 的 SFP 模块，请从设备中移除该 SFP 模块，并用 Inspur 模块替换该模块。在插入 Inspur SFP 模块后，请使用 **errdisable recovery cause gbic-invalid** 全局配置命令确认端口状态，并输入从错误禁用状态恢复的时间间隔。在经过此时间间隔后，接口将从错误禁用状态中恢复，并重新运行。有关 **errdisable recovery** 命令的更多信息，请参阅此版本的命令参考。

如果模块被识别为 Inspur SFP 模块，但系统无法读取供应商数据信息以确认其准确性，则会生成 SFP 模块错误消息。在这种情况下，用户应移除并重新插入 SFP 模块。如果仍然出现故障，则说明 SFP 模块本身可能有故障。

监控 SFP 模块状态

用户可以使用 **show interface transceiver** 特权 EXEC 命令检查 SFP 模块的物理状态或运行状态。该命令显示了运行状态，例如特定接口上 SFP 模块的温度和电流以及警报状态。用户还可以使用该命令检查 SFP 模块上的速度设置和双工设置。有关更多信息，请参阅此版本命令参考中的 **show interfaces transceiver** 命令。

运行 Ping

如果用户试图 ping 不同 IP 子网中的主机，那么必须为网络定义静态路由，或者配置 IP 路由以帮助数据包在这些子网之间选路。

在默认情况下，所有设备上的 IP 路由处于禁用状态。

注释： 尽管 **ping** 命令可以使用其他协议的关键字，但此版本不支持这些关键字。

使用此命令从设备上 ping 网络中的其他设备：

命令	目的
ping ip*host* l*address* Device# ping 172.20.52.3	通过 IP，或者通过使用主机名或网络地址来 ping 远程主机。

温度监控

设备会监控温度，并使用温度信息来控制风扇。

使用 **show env temperature status** 特权 EXEC 命令显示温度值、状态和门限值。温度值是设备内部的温度（而不是外部温度）。用户只能使用 **system env temperature threshold yellow***value* 全局配置命令配置黄色门限的基准（摄氏度），以便设置黄色和红色门限值之间的差值。用户不能配置绿色或红色门限值。有关更多信息，请参阅此版本的命令参考。

物理路径监控

用户可以使用以下特权 EXEC 命令来监控数据包从源设备到目的设备的物理路径：

表格 204：物理路径监控

命令	目的
tracetroute mac [**interface***interface-id*] {*source-mac-address*} [**interface***interface-id*] {*destination-mac-address*} [**vlan***vlan-id*] [**detail**]	显示从指定的源 MAC 地址到指定的目的 MAC 地址的报文所经过的二层路径。
tracetroute mac ip{*source-ip-address*| *source-hostname*}{*destination-ip-address*| *destination-hostname*} [**detail**]	显示从指定的源 IP 地址或主机名到指定的目的 IP 地址或主机名的数据包所采用的第 2 层路径。

执行 IP Traceroute

注释： 尽管 **traceroute** 特权 EXEC 命令可以使用其他协议的关键字，但此版本不支持这些关键字。

命令	目的
traceroute ip*host* Device# traceroute ip 192.51.100.1	跟踪数据包通过网络时所走路径。

运行 TDR 及显示结果

要运行 TDR，请输入 **test cable-diagnostics tdr interface***interface-id* 特权 EXEC 命令。

要显示结果，请输入 **show cable-diagnostics tdr interface***interface-id* 特权 EXEC 命令。

重定向调试和错误消息输出

默认情况下，网络服务器将 **debug** 命令和系统错误消息的输出发送到控制台。如果使用此默认值，用户可以使用虚拟终端连接来监视调试的输出，而不必连接到控制台端口或以太网管理端口。

目的地包括控制台、虚拟终端、内部缓冲区和运行系统日志服务器的 UNIX 主机。系统日志格式与 4.3 版 Berkeley 标准分发（Berkeley Standard Distribution，BSD）UNIX 及其衍生版本兼容。

注释： 请注意，用户使用的调试目标会影响系统开销。当您将消息记录到控制台时，会产生非常高的开销。当您将消息记录到虚拟终端时，会产生较少的开销。将消息记录到系统日志服务器可以产生更少的开销，如果是记录到内部缓冲器产生的开销将最小。

了解更多有关系统消息记录的详细信息，请参阅*配置系统消息记录*。

使用 show platform forward 命令

如果通过系统将数据包发送到接口，**show platform forward** 特权 EXEC 命令的输出可以提供一些有关转发结果的有用信息。根据输入的有关数据包的参数，输出可提供用于计算转发目的地、位图和出口信息的查找表结果和端口映射。

命令输出中的大多数信息主要给技术支持人员使用，他们可以访问有关设备专用集成电路（Application-Specific Integrated Circuits，ASIC）的详细信息。然而，数据包转发信息也有助于故障排除。

使用 show debug 命令

在特权 EXEC 模式下输入 **show debug** 命令。此命令显示交换机上所有可用的调试选项。
要查看所有条件调试选项，请运行命令 **show debug condition**。可以通过选择条件标识符*<1-1000>*或 *all* 条件来列出这些命令。
要禁用调试，请使用 **no debug all** 命令。

注意： 由于调试的输出在 CPU 进程中被分配了较高优先级，它可能导致系统不可用。因此，仅在排除特定问题或与 Inspur 技术支持人员进行故障排除会话期间使用 **debug** 命令。最好在网络流量较低和用户较少时使用调 **debug** 命令。在此期间的调试可减少增加的 **debug** 命令处理开销影响系统使用的可能性。

有关详细信息，请参阅 *Inspur INOS 配置基础命令参考，Inspur INOS（Inspur 3850 交换机）*。

配置 OBFL

注意： 建议用户不要禁用 OBFL，并且不要删除存储在闪存中的数据。

- 要启用 OBFL，请使用 **hw-switch switch** [*switch-number*] **logging onboard** [**message level***level*]全局配置命令。在交换机上，*switch-number* 的范围在 1 到 9 之间。使用 **message level***level* 参数可以指定交换机生成并存储在闪存中的有关硬件信息的重要性。
- 要将 OBFL 数据复制到本地网络或指定文件系统，请使用 **copy onboard switch** *switch-number***url***url-destination* 特权 EXEC 命令。
- 要禁用 OBFL，请使用 **no hw-switch switch** [*switch-number*] **logging onboard** [**message level**] 全局配置命令。
- 要清除闪存中除了正常运行时间和 CLI 命令信息之外的所有 OBFL 数据，请使用 **clear onboard switch***switch-number* 特权 EXEC 命令。
- 在交换机堆栈中，用户可以使用 **hw-switch switch** [*switch-number*] **logging onboard** [**message level***level*]全局配置命令在独立交换机或所有堆栈成员上启用 OBFL。
- 用户可以从活跃交换机上启用或禁用成员交换机上的 OBFL。

有关本节中的命令的更多信息，请参阅此版本的命令参考。

验证软件配置的故障排除

显示 OBFL 信息

表格 205：显示 OBFL 信息的命令

命令	目的
show onboard switch*switch-number***clilog** Device# show onboard switch 1 clilog	显示独立交换机或指定堆栈成员上输入的 OBFL CLI 命令。

show onboard switch*switch-number***environment** Device# show onboard switch 1 environment	显示独立交换机或指定堆栈成员及所有连接的 FRU 设备上的 UDI 信息，包括：PID、VID 和序列号。
show onboard switch*switch-number***message** Device# show onboard switch 1 message	显示独立交换机或指定堆栈成员生成的有关硬件的消息。
show onboard switch*switch-number***counter** Device# show onboard switch 1 counter	显示独立交换机或指定堆栈成员的计数器信息。
show onboard switch*switch-number***temperature** Device# show onboard switch 1 temperature	显示独立交换机或指定交换机堆栈成员的温度。
show onboard switch *switch-number***uptime** Device# show onboard switch 1 uptime	显示独立交换机或指定堆栈成员启动的时间，重新启动的原因，以及自上次重新启动以来运行的时间。
show onboard switch *switch-number***voltage** Device# show onboard switch 1 voltage	显示独立交换机或指定堆栈成员的系统电压。
show onboard switch *switch-number***status** Device# show onboard switch 1 status	显示独立交换机或指定堆栈成员的状态。

示例：确认高 CPU 使用率的问题及原因

要确定高 CPU 使用率是否有问题，请输入 **show processes cpu sorted** 特权 EXEC 命令。请注意输出示例第一行中带下划线的信息。

```
Device# show processes cpu sorted
CPU utilization for five seconds: 8%/0%; one minute: 7%; five minutes: 8%
PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process
309 42289103 752750 56180 1.75% 1.20% 1.22% 0 RIP Timers
140 8820183 4942081 1784 0.63% 0.37% 0.30% 0 HRPC qos request
100 3427318 16150534 212 0.47% 0.14% 0.11% 0 HRPC pm-counters
192 3093252 14081112 219 0.31% 0.14% 0.11% 0 Spanning Tree
143 8 37 216 0.15% 0.01% 0.00% 0 Exec
...
<输出已删节>
```

示例显示了正常的 CPU 利用率。输出显示最近 5 秒的使用率是 8%/0%，具有以下含义：

- CPU 总的利用率为 8%，包括运行 Inspur INOS 进程和处理中断所花费的时间。
- 处理中断所花费的时间为 0%。

表格 206：排除 CPU 利用率问题

问题类型	原因	修正措施
中断百分比值几乎与 CPU 总利用率值一样高。	CPU 从网络中接收到过多数据包。	确定网络数据包的来源。停止网络流量，或更改交换机配置。 请参阅“分析网络流量”部分。

在中断用时最少的情况下，CPU 总利用率大于 50%。	一个或多个 Inspur INOS 进程消耗过多的 CPU 时间。这通常由激活进程的事件触发。	识别异常事件，并解决根本原因。请参阅“调试活跃进程”部分。

软件配置故障排除场景

以太网供电故障排除场景

表格 207：以太网供电故障排除场景

症状或问题	可能的原因和解决方法
只有一个端口没有 PoE。 故障仅出现在一个交换机端口上。PoE 和非 PoE 设备在此端口不工作，但在其他端口工作。	请确认用电设备是否在另一个 PoE 端口上工作。 使用 **show run** 或 **show interface status** 用户 EXEC 命令确认端口未关闭或处于错误禁用状态。 **注意：** 当端口关闭时，大多数交换机会关闭端口电源，即使在 IEEE 规范中也仅将其列为可选项。 确认用电设备到交换机端口的以太网电缆是否正常：将已知良好的非 PoE 以太网设备连接到以太网电缆，并确保用电设备可与另一主机建立链路并交换流量。 确认从交换机前面板到用电设备的电缆总长度不超过 100 米。 断开以太网电缆与交换机端口的连接。使用短以太网电缆将已知良好的以太网设备直接连接到交换机前面板（不是接线板）上的此端口。确认它可以与另一台主机建立以太网链路并交换流量，或 ping 端口 VLAN SVI。 接下来，将用电设备连接到此端口，并确认是否接通电源。 如果用电设备使用跳线连接到交换机端口时未接通电源，请将连接的用电设备总数与交换机功率预算（PoE 可用）进行比较。使用 **show inline power** 命令确认可用的电量。
在所有端口或一组端口上没有 PoE。 故障发生在所有交换机端口上。 未通电的以太网设备不能在任何端口上建立以太网链路，并且 PoE 设备也无法通电。	如果电源出现连续的、间歇的或重复的警报，若电源是一个现场可更换单元，请更换电源。否则，请更换交换机。 如果问题出现在一组连续的端口上，但并非所有端口，说明电源可能没有问题，问题可能与交换机中的 PoE 调节器有关。 使用 **show log** 特权 EXEC 命令查看之前报告的 PoE 条件或状态更改的警报或系统消息。 如果没有警报，请使用 **show interface status** 命令确认端口是否关闭或处于错误禁用状态。如果端口处于错误禁用状态，请使用 **shut** 和 **no shut** 接口配置命令重新启用端口。

	使用 **show env power** 和 **show power inline** 特权 EXEC 命令可查看 PoE 状态和功率预算（PoE 可用）。 查看运行的配置以确认未在端口上配置 **power inline never**。 将未通电的以太网设备直接连接到交换机端口。仅使用跳线，不使用已有的配线电缆。输入 **shut** 和 **no shut** 接口配置命令，然后确认以太网链路是否已建立。如果此连接良好，请使用短跳线将用电设备连接到此端口，并确认其通电。如果设备通电，请确认是否所有的中间接线板已正确连接。 在交换机选择一个端口，断开其余端口的以太网电缆。使用短跳线，将用电设备连接到该 PoE 端口。确认用电设备所用功率小于该交换机端口可提供的功率。 使用 **show power inline** 特权 EXEC 命令确认用电设备可以在端口未关闭时接收电量。或者，通过观察用电设备确认其处于通电状态。 如果用电设备在当只有一个用电设备连接到交换机时才能处于通电状态，请在其余端口上输入 **shut** 和 **no shut** 接口配置命令，然后将以太网电缆依次重新连接到交换机 PoE 端口。使用 **show interface status** 和 **show power inline** 特权 EXEC 命令来监管内联电源统计信息和端口状态。 如果仍然没有任何端口能 PoE，则电源中 PoE 部分的保险丝可能会断开。这通常会产生警报。请再次检查日志，了解更早的系统消息报告的警报。
Inspur IP 电话断开连接或重置。 正常工作后，Inspur 电话会间歇性地重新加载或断开与 PoE 的连接。	确认交换机到用电设备的所有电力连接。任何不可靠的连接可能会导致电力中断和用电设备无法稳定运行，例如不稳定的用电设备会断开连接和重新加载。 确认从交换机端口到用电设备的电缆长度不超过 100 米。 请注意在用电设备出现无法连接的情况时，交换机所处位置的电气环境以及用电设备是否发生变化。 请注意在用电设备出现无法连接的情况时，是否有任何错误消息出现。使用 **show log** 特权 EXEC 命令查看错误消息。 确认 IP 电话在重新加载之前不会立即失去对呼叫管理器的访问（这可能是网络问题，而非 PoE 问题）。 使用非 PoE 设备更换用电设备，并确认设备是否正常工作。如果非 PoE 设备有链路问题或高误码率，则问题可能出在交换机端口和用电设备之间的不可靠电缆连接。
非 Inspur 的用电设备不能在 Inspur PoE 交换机上工作。 当非 Inspur 的用电设备连接到	请使用 **show power inline** 命令确认交换机的功率预算（PoE 可用）在用电设备连接之前或之后是否会耗尽。在这类用电设备连接之前，请确认有足够的电源可用。

Inspur PoE 交换机上，要么无法连通电源，要么开机后马上关机。非 PoE 设备正常工作。	使用 **show interface status** 命令确认交换机是否检测到连接的用电设备。 使用 **show log** 命令查看端口上电流情况的系统消息。 请准确地识别症状：是否初时用电设备通电，而后断电？如果是，则问题可能是初始涌入（或流入）电流超过端口的电流门限限制。

软件故障排除的配置示例

示例：Ping 某 IP 主机

此示例说明如何 ping 一个 IP 主机：

```
Device# ping 172.20.52.3
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echoes to 172.20.52.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
Device#
```

表格 208：Ping 输出显示的字符

字符	描述
!	每个感叹号表示接收到一个应答。
.	每个句号表示等待一个应答时，网络服务超时。
U	收到一个目的地不可达的 PDU。
C	收到一个经历过拥塞的数据包。
I	用户中断测试。
?	未知数据包类型。
&	数据包生存期已过。

若要结束此次 ping 会话，请输入退出序列（默认情况下为 **Ctrl- ^ X**）。同时按下并释放 **Ctrl**，**Shift** 和 **6** 键，然后按 **X** 键。

示例：对 IP 主机执行 Traceroute

此示例说明如何对 IP 主机执行 **traceroute**：

```
Device# traceroute ip 192.0.2.10
Type escape sequence to abort.
Tracing the route to 192.0.2.10
1 192.0.2.1 0 msec 0 msec 4 msec
2 192.0.2.203 12 msec 8 msec 0 msec
3 192.0.2.100 4 msec 0 msec 0 msec
4 192.0.2.10 0 msec 4 msec 0 msec
```

结果显示了跳数、路由器的 IP 地址以及发送的三个探针的各自往返时间（以毫秒为单位）。

表格 209：Traceroute 输出显示的字符

字符	描述
*	探针超时。
?	未知数据包类型。
A	管理不可达。通常，此输出说明有访问列表限制了流量。
H	主机不可达。
N	网络不可达。
P	协议不可达。
Q	源抑制。
U	端口不可达。

要结束跟踪进程，请输入退出序列（默认情况下为 **Ctrl-^X**）。同时按下并释放 **Ctrl**，**Shift** 和 **6** 键，然后按 **X** 键。

示例：启用所有系统诊断

注意： 由于调试输出的优先级高于其他网络流量，并且 **debug all** 特权 EXEC 命令比其他 **debug** 命令生成的输出多，所以它可能导致交换机性能严重降低甚至不可用。所以无论在哪种场景下，最好使用更具体的 **debug** 命令。

这个命令会禁用所有系统诊断：

```
Device# debug all
```

no debug all 特权 EXEC 命令会禁用所有诊断输出。使用 **no debug all** 命令便于用户确保将所有启用的 debug 命令停止。

其他参考资料

相关文档

相关主题	文档标题
系统管理命令	系统管理命令参考文献（Inspur 6650 交换机）
独立平台命令参考	配置基础命令参考文献， Inspur INOS（Inspur 6650 交换机）
独立平台配置信息	配置基础配置指南，Inspur INOS（Inspur 6650 交换机）

标准和 RFC

标准/RFC	标题
无	—

技术助手

描述	链接
Inspur 支持网站提供了广泛的在线资源，包括用于故障排除以及解决 Inspur 产品问题和技术问题的文档和工具。	http://wwwicntnetworks.com

要获得有关产品的安全和技术信息，用户可以订阅不同的服务，例如产品警报工具（从现场记录获取），Inspur 技术服务实时通讯和简单讯息聚合订阅（RSS）。 要获取 Inspur 支持网站上的大多数工具，需要有 icntnetworks.com 的用户 ID 和密码。	

软件配置故障排除的历史特性与信息

版本	修订
Inspur INOS 12.2	引入了这一特性。

使用闪存文件系统

关于闪存文件系统的信息

闪存文件系统是一个可以存储文件的单独闪存设备。该设备提供几个常用命令用来管理软件包和配置文件。设备上缺省的闪存文件系统被命名为 flash：。
从活跃设备或者任何堆栈成员的角度看，flash:特指本地闪存设备，其与所查看文件系统所在的设备相同。在设备堆栈中，可以在活跃的设备上查看众多堆栈成员的每一个闪存设备。这些闪存文件系统的名字包含相应的设备成员编号。例如，从活跃设备上查看，flash-3：代表的文件系统与设备堆栈成员 3 上的 flash：相同。使用 **show file systems** 特权 EXEC 命令来查看所有文件系统，包含设备堆栈上的闪存文件系统。
每次只有一个用户可以管理设备堆栈的软件包和配置文件。

显示可用的文件系统

为了查看设备上可用的文件系统，使用 **show file systems** 特权 EXEC 命令，单个设备的具体示例如下：

```
Device# show file systems
File Systems:
        Size(b)          Free(b)          Type       Flags       Prefixes
 *     15998976         5135872         flash         rw         flash:
               -               -        opaque         rw         bs:
```

```
           -             -       opaque       rw       vb:
       52428        520138        nvram       rw       nvram:
           -             -      network       rw       tftp:
           -             -       opaque       rw       null:
           -             -       opaque       rw       system:
           -             -       opaque       ro       xmodem:
           -             -       opaque       ro       ymodem:
```

该示例展示了一个设备栈。此例中，活跃设备是设备栈成员 1；flash-2 代表位于设备栈成员 2 的文件系统；flash-3 代表位于设备栈成员 3 的文件系统，以此类似，直到 flash-9 代表位于设备栈成员 9 的文件系统。该示例也列出了事故信息目录以及插到活跃设备上的 USB 闪存驱动器。

```
Device# show file systems
File Systems:
         Size(b)         Free(b)        Type     Flags     Prefixes
       145898496         5479424        disk       rw      crashinfo:crashinfo-1:
       248512512        85983232        disk       rw      crashinfo-2:stby-crashinfo:
       146014208        17301504        disk       rw      crashinfo-3:
       146014208               0        disk       rw      crashinfo-4:
       146014208         1572864        disk       rw      crashinfo-5:
       248512512        30932992        disk       rw      crashinfo-6:
       146014208         6291456        disk       rw      crashinfo-7:
       146276352        15728640        disk       rw      crashinfo-8:
       146276352        73400320        disk       rw      crashinfo-9:
 *     741621760       481730560        disk       rw      flash:flash-1:
      1622147072      1360527360        disk       rw      flash-2:stby-flash:
       729546752       469762048        disk       rw      flash-3:
       729546752       469762048        disk       rw      flash-4:
       729546752       469762048        disk       rw      flash-5:
      1622147072      1340604416        disk       rw      flash-6:
       729546752       469762048        disk       rw      flash-7:
      1749549056      1487929344        disk       rw      flash-8:
      1749549056      1487929344        disk       rw      flash-9:
               0               0        disk       rw      unix:
               -               -        disk       rw      usbflash0:usbflash0-1:
               -               -        disk       rw      usbflash0-2: stby-usbflash0:
               -               -        disk       rw      usbflash0-3:
               -               -        disk       rw      usbflash0-4:
               -               -        disk       rw      usbflash0-5:
               -               -        disk       rw      usbflash0-6:
               -               -        disk       rw      usbflash0-7:
               -               -        disk       rw      usbflash0-8:
               -               -        disk       rw      usbflash0-9:
```

```
          0             0        disk       ro       webui:
          -             -      opaque       rw       system:
          -             -      opaque       rw       tmpsys:
    2097152       2055643       nvram       rw       stby-nvram:
          -             -       nvram       rw       stby-rcsf:
          -             -      opaque       rw       null:
          -             -      opaque       ro       tar:
          -             -     network       rw       tftp:
    2097152       2055643       nvram       rw       nvram:
          -             -      opaque       wo       syslog:
          -             -     network       rw       rcp:
          -             -     network       rw       http:
          -             -     network       rw       ftp:
          -             -     network       rw       scp:
          -             -     network       rw       https:
          -             -      opaque       ro       cns:
          -             -      opaque       rw       revrcsf:
```

表 202：文件系统字段描述

域	数值
大小（比特）	文件系统的内存比特大小。
可用（比特）	文件系统的可用内存比特大小。
类型	文件系统类型。 **disk**——闪存内存设备、USB 或者事故信息文件文件系统。 **network**——网络设备文件系统，例如 FTP 服务器或者 HTTP 服务器。 **nvram**——非易失随机存取存储设备文件系统（NVRAM）。 **opaque**——本地生成的伪文件系统（例如，该系统）或者下载接口（例如 brimux）文件系统。 **unknown**——未知类型文件系统。
标志位	文件系统权限。 **ro**——只读。 **rw**——读/写。 **wo**——只写。
前缀	文件系统别名。 **cashinfo**——事故信息文件。 **flash**: ——闪存文件系统。 **ftp**: —— FTP 服务器。 **http**: —— HTTP 服务器。 **https**: ——安全 HTTP 服务器。 **nvram**: ——NVRAM。

	null: ——空拷贝，你可以拷贝一个远程文件到空拷贝，然后查看该文件大小。 **rcp:** ——远程拷贝协议（Remote Copy Protocol，RCP）服务器。 **scp**: ——会话控制协议（Session Control Protocol，SCP）服务器。 **system**: ——包括系统内存和当前运行配置。 **tftp**: —— TFTP 网络服务器。 **usbflash0**: —— USB 闪存内存设备。 **xmodem**: ——使用 Xmodem 协议从网络中获取文件。 **ymodem**: ——使用 Ymodem 协议从网络中获取文件。

设置默认文件系统

cd*filesystem:*特权 EXEC 命令用来指定默认文件系统的文件或目录。可以设置文件系统的相关命令省略 *filesystem:*参数。例如，对于所有带有 *filesystem:*参数的特权 EXEC 命令，可以使用 **cd** 命令指定的文件系统。
默认文件系统为 *flash:*。
可以使用 **pwd** 特权 EXEC 命令来查看由 **cd** 命令指定的当前默认文件系统。

显示文件系统上的文件信息

在对文件系统进行操作前，可以查看文件系统上的内容。例如，在把新的配置文件拷贝到至闪存之前，管理员也许希望确认文件系统上是否存在重名文件。类似情况，拷贝闪存配置文件到另一个位置之前，管理员也许想确认该文件名的文件名，以便在其他命令中使用。使用下表所列的特权 EXEC 命令来查看文件系统上的文件信息。

表 203：显示文件信息的命令

命令	描述
dir/all [*filesystem:filename*]	查看文件系统上的文件列表。
show file systems	查看文件系统每个文件的更多信息。
show file information*file-url*	查看指定某个文件的信息。
show file descriptors	查看打开文件的描述符列表。文件描述符为打开文件的内部标志，可以通过这个命令查看是否其他用户打开了某个文件。

例如，使用 **dir** 特权 EXEC 命令查看文件系统上的所有文件。

```
device# dir flash:
Directory of flash:/
7386 -rwx 2097152 Jan 23 2013 14:06:49 +00:00 nvram_config
7378 drwx 4096 Jan 23 2013 09:35:11 +00:00 mnt
```

```
7385 -rw- 221775876 Jan 23 2013 14:15:13 +00:00
cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin
7389 -rwx 556 Jan 21 2013 20:47:30 +00:00 vlan.dat
712413184 bytes total (445063168 bytes free)
device#
```

改变目录与查看工作目录(CLI)

按照以下步骤改变目录并显示工作目录。

总步骤

1. **enable**
2. **dir** *filesystem*
3. **cd** *directory_name*
4. **pwd**
5. **cd**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** Device**> enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**dir** *filesystem* **示例：** Device# dir flash:	查看指定文件系统的所有目录。 对于 *filesystem:*参数，使用 flash：查看系统板闪存设备。 使用 flash-*n* 来读取设备成员栈中的闪存分区，*n* 为设备栈成员的编号，例如，flash-4。
步骤 3	**cd** *directory_name* **示例：** Device# cd new_configs	切换到指定的目录。 该命令展现了如何切换到 *new_configs* 目录。
步骤 4	**pwd** **示例：** Device# pwd	查看工作目录。
步骤 5	**cd** **示例：** Device# cd	切换到默认目录。

创建目录(CLI)

在特权 EXEC 模式中按照以下步骤创建目录。

总步骤

1. **dir***filesystem:*
2. **mkdir** *directory_name*
3. **dir** *filesystem:*

具体步骤

	命令或操作	目的
步骤 1	**dir***filesystem:* **例如：** Device# dir flash:	查看指定文件系统的所有目录。对于 *filesystem:*参数，使用 flash：查看系统板闪存设备。
步骤 2	**mkdir** *directory_name* **例如：** Device# mkdir new_configs	创建新目录。目录名称区分大小写且斜线（/）之间字符数量小于 45 个；目录名称不能包含控制字符、空格、斜线、引用符号、分号或者冒号。
步骤 3	**dir** *filesystem:* **例如：** Device# dir flash:	验证创建的目录。

删除目录

要移除目录以及其中所有文件与子目录，使用 **delete /force /recursive** *filesystem:/file-url* 特权 EXEC 命令。

使用**/recursive** 关键字来删除指定的目录以及其中包含的所有子目录与文件。使用**/force** 关键字抑制每个确认删除目录中文件的提示。在删除过程开始时，用户只被提示一次。

对于 *filesystem*，使用 **flash:**来指定系统板闪存设备。对于 *file-url*，输入要删除的目录名称。目录中的所有文件以及子目录都会移除。

注意： 目录被删除后其内容不能被恢复。

拷贝文件

要把文件从源拷贝到目的位置，请使用 **copy** *source-url destination-url* 特权 EXEC 命令。对于源

以及目的 URL，可以使用 **running-config** 以及 **startup-config** 关键字。比如，**copy running-config startup-config** 命令会把当前的运行配置保存到闪存的 NVRAM 区，以在系统初始化时使用。也可以使用 Xmodem 或 Ymodem 协议，对网络机器上的文件使用特殊的文件系统（**xmodem:**、**ymodem:**）进行拷贝。

网络文件系统 URL 包括 ftp:、rcp:以及 tftp:，语法如下：

- FTP——ftp:[[//username [:password]@location]/directory]/filename
- RCP——rcp:[[//username@location]/directory]/filename
- TFTP——tftp:[[//location]/directory]/filename

本地可写的文件系统包括 flash:。

存在一些不合法的源目组合。具体来说，用户不能拷贝这些组合：

- 从运行配置到运行配置
- 从启动配置到启动配置
- 从一台设备到相同的设备（如 **copy flash: flash:**命令是非法的）

把文件从一个堆栈中的设备拷贝到相同堆栈中的另一台设备

要把文件从一个堆栈中的设备拷贝到相同堆栈中的另一台设备，使用 **flash-X:**标注，其中 **X** 是设备编号。

要查看堆栈中的所有设备，请在特权 EXEC 模式中使用 **show switch** 命令。如以下有 9 个成员的设备堆栈示例所示：

```
Device# show switch
Switch/Stack Mac Address : 0006.f6b9.b580 - Local Mac Address Mac persistency wait
time:
Indefinite
H/W Current
Switch# Role Mac Address Priority Version State
------------------------------------------------------------
*1 Active 0006.f6b9.b580 15 P3B Ready
2 Standby 0006.f6ba.0c80 14 P3B Ready
3 Member 0006.f6ba.3300 7 P3B Ready
4 Member 0006.f6b9.df80 6 P3B Ready
5 Member 0006.f6ba.3880 13 P1A Ready
6 Member 1ce6.c7b6.ef00 4 PP Ready
7 Member 2037.06ce.2580 3 P2A Ready
8 Member 2037.0653.7e00 2 P5A Ready
9 Member 2037.0653.9280 1 P5B Ready
```

要显示特定设备上的所有可用文件系统，请使用 **copy** 命令，如以下有 5 个成员的堆栈示例所示：

```
Device# copy flash: ?
crashinfo-1: Copy to crashinfo-1: file system
crashinfo-2: Copy to crashinfo-2: file system
crashinfo-3: Copy to crashinfo-3: file system
crashinfo-4: Copy to crashinfo-4: file system
crashinfo-5: Copy to crashinfo-5: file system
```

```
crashinfo: Copy to crashinfo: file system
flash-1: Copy to flash-1: file system
flash-2: Copy to flash-2: file system
flash-3: Copy to flash-3: file system
flash-4: Copy to flash-4: file system
flash-5: Copy to flash-5: file system
flash: Copy to flash: file system
ftp: Copy to ftp: file system
http: Copy to http: file system
https: Copy to https: file system
null: Copy to null: file system
nvram: Copy to nvram: file system
rcp: Copy to rcp: file system
revrcsf: Copy to revrcsf: file system
running-config Update (merge with) current system configuration
scp: Copy to scp: file system
startup-config Copy to startup configuration
stby-crashinfo: Copy to stby-crashinfo: file system
stby-flash: Copy to stby-flash: file system
stby-nvram: Copy to stby-nvram: file system
stby-rcsf: Copy to stby-rcsf: file system
stby-usbflash0: Copy to stby-usbflash0: file system
syslog: Copy to syslog: file system
system: Copy to system: file system
tftp: Copy to tftp: file system
tmpsys: Copy to tmpsys: file system
usbflash0-1: Copy to usbflash0-1: file system
usbflash0-2: Copy to usbflash0-2: file system
usbflash0-3: Copy to usbflash0-3: file system
usbflash0-4: Copy to usbflash0-4: file system
usbflash0-5: Copy to usbflash0-5: file system
usbflash0: Copy to usbflash0: file system
Device#
```

此示例展示了如何把保存在设备 2 闪存分区的配置文件拷贝到设备 4 的闪存分区中。示例中假设设备 2 和设备 4 在相同堆栈。

```
Device# copy flash-2:config.txt flash-4:config.txt
```

删除文件

如何不再需要一个闪存设备上的文件，可以将其永久删除。要删除指定闪存设备的文件或目录，请使用特权 EXEC 命令 **delete** [**/force**] [**/recursive**] [*filesystem:*]*/file-url*。

使用**/recursive** 关键字删除一个目录以及其中包含的所有子目录与文件。使用**/force** 关键字来抑制每个提示用户确认删除目录中文件的确认提示。用户仅在此删除过程开始时会被提示一

次。使用**/force** 以及**/recursive** 关键字，可以删除由 **archive download-sw** 命令安装但不再需要的软件镜像。
如果省略了 *filesystem:*选项，设备会使用 **cd** 命令指定的默认设备。对于 *file-url*，用户需指定要删除文件的路径（目录）以及文件名。
尝试删除任何文件时，系统会提示用户确认删除。
注意： 文件被删除后无法被恢复。
此示例展示了如何删除默认闪存设备中的 *myconfig* 文件：

```
Device# delete myconfig
```

创建、显示与提取文件

用户可以创建一个文件并向其中写入多个文件、列出一个文件中的多个文件，并从一个文件中提取出多个文件，如下一节所述。
在特权 EXEC 模式中按照以下步骤创建文件，显示文件内容并提取文件：

总步骤

1. **archive tar /create** *destination-url* **flash:** */file-url*
2. **archive tar /table** *source-url*
3. **archive tar /xtract** *source-url* **flash:***/file-url* [*dir/file...*]
4. **more** [**/ascii** | **/binary** | **/ebcdic**] */file-url*

具体步骤

	命令或操作	目的
步骤 1	**archive tar /create** *destination-url***flash:** */file-url* **示例：** `device# archive tar /create tftp:172.20.10.30/saved. flash:/new-configs`	创建一个文件并向其中添加文件。 对于 *destination-url*，请指定本地或网络文件系统的别名目的 URL，以及要创建的文件名： • 本地闪存文件系统语法：**flash:** • FTP 语法：**ftp**:[[*//username*[:*password*]*@location*]/*directory*]/*-filename* • RCP 语法：**rcp**:[[*//username@location*]/*directory*]/*-filename* • TFTP 语法：**tftp**:[[*//location*]/*directory*]/*-filename* 对于 **flash:***/file-url*，请指定要创建新文件的本地文件系统位置。也可以在源目录中指定可选的要添加到新文件中的文件或目录列表。如果不指定，此级别下的所有文件与目录都会被写到新创建的文件中。

步骤 2	**archive tar /table** *source-url* **示例：** `device# archive tar /tableflash: /new_configs`	显示文件内容。 对于 *source-url*，请指定本地或网络文件系统的别名源 URL。*-filename*.是要显示的文件。支持以下选项： • 本地闪存文件系统语法：**flash:** • FTP 语法：**ftp**:[[//*username*[:*password*]@*location*]/*directory*]/*-filename* • RCP 语法：**rcp**:[[//*username@location*]/*directory*]/*-filename* • TFTP 语法：**tftp**:[[//*location*]/*directory*]/*-filename* 也可以在此文件后指定一个文件列表，限制显示的文件。此时仅会显示这些文件。如果未指定，所有文件及目录都会显示。
步骤 3	**archive tar /xtract** *source-url* **flash:**/*file-url* [*dir/file...*] **示例：** `device# archive tar /xtract tftp:/172.20.10.30/saved.flash:/new-configs`	把文件提取到闪存文件系统的一个目录中。 对于 *source-url*，指定本地文件系统的源 URL。 *-filename* 是要从中提取文件的文件名。支持以下选项： • 本地闪存文件系统语法：**flash:** • FTP 语法：**ftp**:[[//*username*[:*password*]@*location*]/*directory*]/*-filename* • RCP 语法：**rcp**:[[//*username@location*]/*directory*]/*-filename* • TFTP 语法：**tftp**:[[//*location*]/*directory*]/*-filename* 对于 **flash:**/*file-url* [*dir/file...*]，指定文件要被提取到的本地文件系统位置。使用 *dir/file...*选项指定在文件中的要被提取的文件或目录列表。如果未指定，所有文件及目录都会被提取。
步骤 4	**more** [**/ascii** | **/binary** | **/ebcdic**]/*file-url* **示例：** `device# moreflash:/new-configs`	显示任何可读文件的内容，包括远程文件系统中的文件。

其他参考资料

相关文档

相关主题	文档标题
管理 flash:文件系统的命令	*Inspur INOS 配置基础命令参考*

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

标准

标准	标题
不支持新标准或修订的标准，且支持的现有标准未被修改	-

RFC

RFC	标题
不支持新 RFC 或修订的 RFC，且支持的现有 RFC 未被修改	-

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

管理配置文件

管理配置文件的前提

- 用户至少应该基本熟悉 Inspur INOS 环境以及命令行界面。
- 用户设备上至少应该运行着最少配置。可以使用 **setup** 命令创建基本的配置文件。

管理配置文件的限制

- 此文档中描述的许多 Inspur INOS 命令仅在设备的特定配置模式中可用。
- 一些 Inspur INOS 配置命令仅在特定的设备平台上可用，且命令语法在不同平台上可能有所不同。

关于管理配置文件的信息

配置文件中包含用于定义 Inspur 设备功能的 Inspur 软件命令。系统启动（从 startup-config 文件读取）或当用户在 CLI 的配置模式中输入命令时，命令会被 Inspur INOS 软件解析（翻译并执行）。

启动配置文件（startup-config）在系统启动时被用来配置系统软件。运行配置文件（running-config）包含系统的当前配置。这两个配置文件可以不同。比如，如果希望在短期内更改配置，用户可以使用 EXEC 命令 **configure terminal** 更改运行配置，但不使用 EXEC 命令 **copy running-config startup-config** 保存配置。

要更改运行配置，请使用 **configure terminal** 命令，如*修改配置文件（CLI）*一节所述。使用 Inspur INOS 配置模式时，命令通常会立即执行，且会在输入命令后立即保存到运行配置文件，或在退出配置模式时保存。

要更改启动配置文件，用户可以使用 EXEC 命令 **copy running-config startup-config** 把运行配置文件保存到启动配置中，也可以把配置文件从文件服务器拷贝到启动配置中（参见*把配置文件从 TFTP 服务器拷贝到设备（CLI）*）。

配置模式及选择配置源

要在设备上进入配置模式，请在特权 EXEC 提示符后输入 **configure** 命令。Inspur INOS 软件会返回以下提示，请求用户指定使用终端（terminal）、内存（memory）或网络服务器上储存的文件（network）作为配置命令来源：

```
Configuring from terminal, memory, or network [terminal]?
```

通过终端进行配置允许用户在命令行中输入配置命令，如下一节所述。更多信息参见*重新执行启动配置文件中的配置命令（CLI）*。

通过网络进行配置允许用户通过网络加载并执行配置命令。更多信息参见*把配置文件从 TFTP 服务器拷贝到设备（CLI）*。

使用 CLI 更改配置文件

Inspur INOS软件支持每行使用一条配置命令。可以按需输入任意数量的命令。可以向配置文件中添加输入命令的注释描述，注释前使用一个感叹号（!）。因为注释不会保存到

NVRAM或配置文件的活跃拷贝中，在用户使用EXEC命令**show running-config**或**more system:running-config**时，注释不会出现。使用**show startup-config**或**morenvram:startup-config** EXEC命令列出启动配置时，注释也不会被显示。在配置文件加载到设备上时，注释会被去除。

然而，可以查看储存到文件传输协议（File Transfer Protocol，FTP）服务器，远程拷贝协议（RemoteCopy Protocol，RCP）服务器或简单文件传输协议（Trivial File Transfer Protocol，TFTP）服务器上的配置文件注释。使用CLI配置系统软件时，命令会在输入后执行。

配置文件的位置

配置文件会被保存到以下位置：

- 运行配置被保存到 RAM 中。
- 在除了 A 类闪存文件系统平台以外的所有平台上，启动配置会被储存到非易失性随机存取存储器（非易失性随机存取存储器，NVRAM）中。
- 在 A 类闪存文件系统平台上，启动配置会被保存到 CONFIG_FILE 环境变量指定的位置（参见*指定 A 类闪存文件系统的 CONFIG_FILE 环境变量（CLI）*）。CONFIG_FILE 变量默认使用 NVRAM，也可以使用以下文件系统中的文件：
  - **nvram:**（NVRAM）
  - **bootflash:**（内部闪存内存）
  - **usbflash0:**（闪存文件系统）

把配置文件从网络服务器拷贝到设备

可以把 TFTP、RCP 或 FTP 服务器上的配置文件拷贝到设备的运行配置或启动配置中。执行此操作的原因可能有：

- 恢复备份的配置文件。
- 使用另一台设备的配置文件。例如，向网络中添加了新设备，并希望其配置与原始设备类似。通过把文件拷贝到新设备上，管理员可以仅更改相关的部分，而不用重新创建整个配置文件。
- 把相同的配置命令加载到网络的所有设备上，让所有设备有相似的配置。

EXEC 命令 **copy{ftp: | rcp: | tftp:system:running-config}**会把配置文件加载到设备上，就好像在命令行输入命令一样。在添加命令前设备不会擦除现有的运行配置。如果拷贝的配置文件命令替换了现有配置文件中的命令，现有命令会被擦除。比如，如果拷贝的配置文件中有某条命令的 IP 地址与现有配置不同，拷贝配置的 IP 地址会被使用。然而，现有配置中的一些命令可能不会被代替。此时，最终生成的配置文件会混合使用现有配置以及拷贝的配置，且优先使用拷贝的配置。

要把配置文件完全恢复为服务器上保存文件的拷贝，用户需要直接把配置文件拷贝到启动文件中（使用 **copy ftp:| rcp:| tftp:} nvram:startup-config** 命令）并重启设备。

要把服务器上的配置文件拷贝到设备上，请执行下一节所述的任务。

使用哪种协议取决于使用哪种类型的服务器。FTP 以及 RCP 传输机制能提供比 TFTP 更快的性能及更可靠的传输能力。这些性能提升的原因是 FTP 以及 RCP 传输机制是基于 TCP/IP 协议栈构建的，是面向连接的。

把配置文件从设备拷贝到 TFTP 服务器

在一些 TFTP 实现中，用户在拷贝文件之前必须在 TFTP 服务器上创建一个空文件，并授予读

写以及执行权限。更多信息参见使用的 TFTP 文档。

把配置文件从设备拷贝到 RCP 服务器

用户可以把配置文件从设备拷贝到 RCP 服务器上。

UNIX 社区把网络作为资源使用的初次尝试，推动了远程 Shell 协议的设计与实现，其中就包括远程 Shell（remote shell，rsh）以及远程拷贝（remote copy，rcp）功能。rsh 和 rcp 让用户可以在网络上远程执行命令，并在本地与远程主机或服务器的文件系统之间拷贝文件。Inspur 的 rsh 与 rcp 实现可以与标准的实现进行互操作。

rcp 的 **copy** 命令依赖于远程系统上的 rsh 服务器（或守护进程）。要使用 rcp 拷贝文件，用户需创建一个用于文件分发的服务器，这与对 TFTP 服务器的操作相同。用户仅需要对支持远程 Shell（rsh）的服务器有访问权限（多数 UNIX 系统都支持 rsh）。因为要把文件从一个地方拷贝到另一个地方，用户必须对源文件有读权限，对目的文件有写权限。如果目的文件不存在，rcp 或创建该文件。

虽然 Inspur 的 rcp 实现模仿了 UNIX rcp 实现的功能——在网络上的系统之间拷贝文件——但是 Inspurrcp 命令语法与 UNIX rcp 命令语法不同。

Inspur 的 rcp 支持使用 rcp 作为一组 **copy** 命令的传输机制。这些 rcp 的 **copy** 命令风格与 Inspur TFTP **copy** 命令类似，但是能提供更快的性能以及更可靠的数据传输服务。这些性能提升的原因是 RCP 传输机制是基于 TCP/IP 协议栈构建的，是面向连接的。可以使用 rcp 命令在设备和网络服务器之间拷贝系统镜像以及配置文件。

也可以启用 rcp 支持，允许远程系统上的用户在设备上拷贝文件。

要配置Inspur INOS软件允许远程用户在设备上来回拷贝文件，请使用全局配置命令**iprcmd rcp-enable**。

RCP 用户名的要求

RCP 协议要求客户端在每个 RCP 请求中向服务器发送远程用户名。使用 RCP 把配置文件从设备拷贝到服务器时，Inspur INOS 软件会发送按照以下顺序遇到的第一个合法用户名：

1 **copy** EXEC 命令中指定的用户名。
2 全局配置命令 **ip rcmd remote-username** 设置的用户名。
3 与当前 tty（terminal）进程关联的远程用户名。比如，如果用户通过 Telnet 连接设备，并通过命令 **username** 进行了验证，设备软件会把 Telnet 用户名当作远程用户名发送。
4 设置主机名。

要使 RCP 拷贝请求能成功执行，网络服务器上必须为远程用户名定义了一个账户。如果服务器上有对应的目录结构，配置文件或镜像会到拷贝到服务器上与远程用户名关联的目录中，或从目录中拷贝出。例如，如果系统镜像位于服务器上用户的 home 目录下，可以把改用户名指定为远程用户名。

更多信息参见使用的 RCP 服务器文档。

使用命令 **ip rcmd remote-username** 指定所有拷贝使用的用户名（rcmd 是一个 UNIX 程序，在超级用户级别使用，可以使用基于预留端口号的认证机制在远程机器上执行命令，rcmd 表示“remotecommand”）。如果仅希望给特定的拷贝操作指定用户名，请在 **copy** 命令中包含用户名。

如果要向服务器写文件，必须正确配置 RCP 服务器接收来自设备用户的 RCP 写请求。对于 UNIX 系统，必须在 RCP 服务器上的.rhosts 文件中为远程用户添加一个条目。例如，假设设备有以下配置：

```
hostname Device1
```

```
ip rcmd remote-username User0
```

如果设备的 IP 能翻译成 device1.example.com，则 RCP 服务器上 User0 的.rhosts 文件应包含以下配置：

```
Device1.example.com Device1
```

把配置文件从设备拷贝到 FTP 服务器

可以把配置文件从设备拷贝到 FTP 服务器上。

理解 FTP 用户名及密码

FTP 协议要求客户端在每个 FTP 请求中向服务器发送远程用户名及密码。使用 FTP 把配置文件从设备拷贝到服务器时，Inspur INOS 软件会发送按照以下顺序遇到的第一个合法用户名：

1 **copy** EXEC 命令中指定的用户名。
2 全局配置命令 **ip ftp username** 设置的用户名。
3 匿名。

设备会发送按照以下顺序遇到的第一个合法密码：

1 **copy** 命令中指定的密码。
2 命令 **ip ftp password** 设置的密码。
3 设备产生的密码 *username* @*devicename*.*domain*。*username* 变量是与当前会话关联的用户名，*devicename* 是配置的主机名，*domain* 是设备的域名。

用户名与密码必须与 FTP 服务器上的账户关联。如果向服务器写文件，必须正确配置 FTP 服务器接受来自设备用户的 FTP 写请求。如果服务器上有对应的目录结构，配置文件或镜像会到拷贝到服务器上与远程用户名关联的目录中，或从目录中拷贝出。例如，如果系统镜像位于服务器上用户的 home 目录下，可以把改用户名指定为远程用户名。

更多信息参见使用的 FTP 服务器文档。

使用 **ip ftp username** 和 **ip ftp password** 全局配置命令为所有拷贝命令指定用户名和密码。如果仅希望给特定的拷贝操作指定用户名，请在 **copy**EXEC 命令中包含用户名。

通过 VRF 拷贝文件

可以通过 **copy** 命令中指定的 VRF 接口拷贝文件。在 **copy** 命令中指定 VRF 更简单且更高效，因为用户无需请求更改配置就可以直接更改源接口。

以下示例展示了如何使用 **copy** 命令通过 VRF 拷贝文件：

```
Device# copy scp: flash-1: vrf test-vrf
Address or name of remote host [10.1.2.3]?
Source username [ScpUser]?
Source filename [/auto/tftp-server/ScpUser/vrf_test.txt]?
Destination filename [vrf_test.txt]?
Getting the vrf name as test-vrf
Password:
Sending file modes: C0644 10 vrf_test.txt
!
223 bytes copied in 22.740 secs (10 bytes/sec)
```

把配置文件从一台交换机拷贝到另一台交换机

可以把配置从一台交换机拷贝到另一台交换机。这是一个 2 步的过程——把配置文件从交换机拷贝到 TFTP 服务器，然后从 TFTP 拷贝到另一台交换机上。

要从交换机上拷贝当前的配置，运行 **copy startup-config tftp:**命令。执行命令会把配置拷贝到 TFTP 服务器上。
接着，登录到另一台交换机上并运行 **copy tftp: startup-config** 命令。配置此时会被拷贝到另一台交换机上。
在配置拷贝完成后，使用 **write memory** 命令保存配置，然后重启交换机或者运行 **copy startup-config running-config** 命令。
更多信息参见 *Inspur INOS 配置基础命令参考，Inspur INOS（Inspur 3850 交换机）*。

拷贝比 NVRAM 空间大的文件

要维护超出 NVRAM 大小的配置文件，请查看以下内容。

压缩配置文件

全局配置命令 **service compress-config** 会指定在 NVRAM 中压缩存储配置文件。配置文件被压缩后，设备就能正常工作。系统重启时会识别配置文件被压缩，系统会扩展该文件并正常执行操作。EXEC 命令 **more nvram:startup-config** 会在显示配置之前进行扩展。
压缩配置文件之前，请查阅对应的硬件安装及维护说明，验证系统 ROM 是否支持文件压缩。如果不支持，可以安装支持文件压缩的新 ROM。
配置的大小不能超过 NVRAM 大小的三倍。对于 128KB 大小的 NVRAM，最大展开配置文件的大小是 384 KB。
只有 Inspur INOS 10.0 以及之后版本的引导 ROM 支持全局配置命令 **service compress-config**。安装新 ROM 仅需在 ROM 中没有 Inspur INOS 10.0 版时执行一次。如果引导 ROM 没有识别压缩的配置，会显示以下信息：

```
Boot ROMs do not support NVRAM compression Config NOT written to NVRAM
```

在 A 类闪存文件系统中把配置保存到闪存内存

在 A 类闪存文件系统的设备上，可以把启动配置保存在闪存内存中。设置环境变量 CONFIG_FILE 可以把启动配置保存在内部闪存内存或 PCMCIA 插槽上的闪存内存。
更多信息参见*指定 A 类闪存文件系统的 CONFIG_FILE 环境变量（CLI）*。
编辑或更改大型配置文件时要小心。每次输入 EXEC 命令 **copy system:running-config nvram:startup-config** 时闪存空间都会被使用。因为闪存的文件管理（如优化空闲空间）不会自动进行，必须小心注意可用的闪存空间。使用 **squeeze** 命令来回收已使用的空间。建议使用至少 20 MB 的大容量闪存卡。

通过网络加载配置命令

可以把大型配置文件保存在 FTP、RCP 或 TFTP 服务器上，并在系统启动时进行下载。要使用网络服务器保存大型配置，更多信息参见*把配置文件从设备拷贝到 TFTP 服务器（CLI）*以及*配置设备下载配置文件*。

配置设备下载配置文件

可以配置设备在系统启动时加载一个或两个配置文件。配置文件会被加载到内存中并被读入，就好像在命令行中输入的命令一样。此时，设备的配置文件会混用原始启动配置以及一个或两个下载的配置文件。

网络配置文件与主机配置文件

出于历史原因，设备下载的第一个文件被称为网络配置文件。设备下载的第二个配置文件被称为主机配置文件。当网络上的所有设备都使用许多相同的命令时，可以使用这两个配置文

件。网络配置文件包含为所有设备配置的标准命令。主机配置文件包含特定于一台主机的命令。如果加载这两个文件，应该优先使用主机配置文件。网络配置文件以及主机配置文件必须都位于通过 TFTP、RCP 或 FTP 可达的网络服务器上，且必须都可读。

如何管理配置文件信息

显示配置文件信息（CLI）

要显示配置文件的信息，请完成本节配置任务：

总步骤

1. enable

2. show boot

3. more *file-url*

4. show running-config

5. show startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**show boot** **示例：** `Device# show boot`	显示设置的 BOOT 环境变量内容，CONFIG_FILE 环境变量指向的配置文件名，以及 BOOTLDR 环境变量的内容。
步骤 3	**more** *file-url* **示例：** `Device# more 10.1.1.1`	显示指定文件的特性。
步骤 4	**show running-config** **示例：** `Device# show running-config`	显示运行配置文件的内容（等同于 **more system:running-config** 命令）
步骤 5	**show startup-config** **示例：** `Device# show startup-config`	显示启动配置文件的内容（等同于 **more nvram:startup-config** 命令） 在除了 A 类闪存文件系统平台以外的所有平台上，默认的 startup-config 文件通常被保存在 NVRAM 中。 在 A 类闪存文件系统平台上，CONFIG_FILE 环境变量指向默认的 startup-config 文件。 CONFIG_FILE 变量默认值指向 NVRAM。

修改配置文件（CLI）

Inspur INOS软件支持每行使用一条配置命令。可以按需输入任意数量的命令。可以向配置文件中添加输入命令的注释描述，注释前使用一个感叹号（!）。因为注释不会保存到NVRAM或配置文件的活跃拷贝中，在用户使用EXEC命令**show running-config**或**more system:running-config**时，注释不会出现。使用**show startup-config**或**morenvram:startup-config** EXEC命令列出启动配置时，注释也不会被显示。在配置文件加载到设备上时，注释会被去除。

然而，可以查看储存到文件传输协议（File Transfer Protocol，FTP）服务器，远程拷贝协议（RemoteCopy Protocol，RCP）服务器或简单文件传输协议（Trivial File Transfer Protocol，TFTP）服务器上的配置文件注释。使用CLI配置系统软件时，命令会在输入后执行。要使用CLI配置系统软件，请在特权EXEC模式中使用以下命令：

总步骤

1. **enable**
2. **configure terminal**
3. **configuration command**
4. 进行以下操作之一：
   - **end**
   - **^Z**
5. **copy system:running-config nvram:startup-config**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**configuration command** **示例：** `Device(config)# configuration command`	输入必要的配置命令。Inspur INOS 文档集描述了按照不同技术组织的配置命令。
步骤 4	进行以下操作之一： • **end** • **^Z** **示例：** `Device(config)# end`	结束配置会话并退出到 EXEC 模式。 **注释：**同时按下 Ctrl 和 Z 键时，屏幕上会显示^Z。
步骤 5	**copy system:running-confignvram:startup-config** **示例：** `Device# copy system:running-confignvram:startup-config`	把运行配置保存为启动配置文件。也可以使用 **copy running-config startup-config** 命令，但是用户应该要知道此命令时不太准确的。在多数平台上，此命令会把配置保存到 NVRAM 中。在 A 类闪存文件系统平台上，此步骤会把配置保存到

		CONFIG_FILE 环境变量指定的位置（默认的变量 CONFIG_FILE 指定文件应该被保存到 NVRAM）。

示例

以下示例配置了设备提示符名称。由感叹号（!）标注的注释不会执行任何命令。**hostname** 命令被用来把设备名从 device 改为 new_name。按下 Ctrl-Z（^Z）或输入 **end** 命令，用户会退出配置模式。**copy system:running-config nvram:startup-config** 命令会把当前配置保存为启动配置。

```
Device# configure terminal
Device(config)# !The following command provides the switch host name.
Device(config)# hostname new_name
new_name(config)# end
new_name# copy system:running-config nvram:startup-config
```

启动配置使用 NVRAM 时，系统会把当前配置信息按照配置命令文本形式存储，并只记录非默认的配置。系统会对内存文件计算校验和，以防止数据损坏。

注释： 一些特定的命令可能不会被保存到 NVRAM 中。重启机器后需要再次输入这些命令。这些文档在文档中标示出。建议用户保存一个这些命令的列表，以在设备重启后快速重新配置。

把配置文件从设备拷贝到 TFTP 服务器（CLI）

要把配置从设备拷贝到 TFTP 服务器，请完成本节的配置任务：

总步骤

1. enable

2. copy system:running-config tftp: [[[**//***location*]**/***directory*]**/***filename*]

3. copy nvram:startup-config tftp: [[[**//***location*]**/***directory*]**/***filename*]

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**copy system:running-config tftp:** [[[**//***location*]**/***directory*]**/***filename*] **示例：** `Device# copy system:running-config tftp://server1/topdir/file10`	把运行配置文件拷贝到 TFTP 服务器。
步骤 3	**copy nvram:startup-config tftp:** [[[**//***location*]**/***directory*]**/***filename*] **示例：** `Device# copy nvram:startup-config tftp://server1/1stdir/file10`	把启动配置文件拷贝到 TFTP 服务器。

示例

以下示例把配置文件从设备拷贝到 TFTP 服务器：

```
Device# copy system:running-config tftp://172.16.2.155/tokyo-confg
Write file tokyo-confg on host 172.16.2.155? [confirm] Y
Writing tokyo-confg!!! [OK]
```

接下来做什么？

输入 **copy** 命令后，系统可能提示用户输入额外信息或让用户确认操作。显示的提示取决于用户在 **copy** 命令中提供了多少信息以及当前的全局配置命令 **file prompt** 设置。

把配置文件从设备拷贝到 RCP 服务器（CLI）

要把启动配置文件或运行配置文件从设备拷贝到 RCP 服务器，请在特权 EXEC 模式中使用以下命令配置：

总步骤

1. **enable**
2. **configure terminal**
3. **ip rcmd remote-username** *username*
4. **end**
5. 进行以下操作之一：
   - **copy system:running-config rcp:** [[[**//**[*username***@**]*location*]**/***directory*]**/***filename*]
   - **copy nvram:startup-config rcp:** [[[**//**[*username***@**]*location*]**/***directory*]**/***filename*]

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**ip rcmd remote-username** *username* **示例：** `Device(config)# ip rcmd remote-username NetAdmin1`	（可选）更改默认远程用户名。
步骤 4	**end** **示例：** `Device(config)# end`	（可选）退出全局配置模式。
步骤 5	进行以下操作之一： • **copy system:running-config rcp:** [[[**//**[*username***@**]*location*]**/***directory*]**/***filename*] • **copy nvram:startup-config rcp:**	• 指定把设备运行配置文件保存到 RCP 服务器上 或 • 指定把设备启动配置文件

	[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*] **示例：** `Device# copy system:running-config rcp://NetAdmin1@example.com/dir-files/file1`	保存到 RCP 服务器上

示例

在 RCP 服务器上保存运行配置文件

以下示例把名为 runfile2-confg 的运行配置文件保存到 IP 地址为 172.16.101.101 的远程主机的 netadmin1 目录下：

```
Device# copy system:running-config rcp://netadmin1@172.16.101.101/runfile2-confg
Write file runfile2-confg on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Device#
```

在 RCP 服务器上保存启动配置文件

以下示例展示了使用 RCP 把启动配置文件拷贝到服务器上：

```
Device# configure terminal
Device(config)# ip rcmd remote-username netadmin2
Device(config)# end
Device# copy nvram:startup-config rcp:
Remote host[]? 172.16.101.101
Name of configuration file to write [start-confg]?
Write file start-confg on host 172.16.101.101?[confirm]
![OK]
```

接下来做什么？

输入 **copy** 命令后，系统可能提示用户输入额外信息或让用户确认操作。显示的提示取决于用户在 **copy** 命令中提供了多少信息以及当前的全局配置命令 **file prompt** 设置。

把配置文件从设备拷贝到 FTP 服务器（CLI）

要把启动配置文件或运行配置文件从设备拷贝到 FTP 服务器上，请完成以下配置任务：

总步骤

1. **enable**
2. **configure terminal**
3. **ip ftp username** *username*
4. **ip ftp password** *password*
5. **end**
6. 进行以下操作之一：
   - **copy system:running-config ftp:** [[[**//**[*username* [**:***password*]**@**]*location*]**/***directory*]**/***filename*]
   - **copy nvram:startup-config ftp:** [[[**//**[*username* [**:***password*]**@**]*location*]**/***directory*]**/***filename*]

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**ip ftp username** *username* **示例：** `Device(config)# ip ftp username NetAdmin1`	（可选）指定默认远程用户名。
步骤 4	**ip ftp password** *password* **示例：** `Device(config)# ip ftp password adminpassword`	（可选）指定默认密码。
步骤 5	**end** **示例：** `Device(config)# end`	（可选）退出全局配置模式。只有覆盖了默认的远程用户名或密码时（见步骤 2 和步骤 3）才需要进行此步骤。
步骤 6	进行以下操作之一： • **copy system:running-config ftp:** [[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/***filename*] 或 • **copy nvram:startup-config ftp:** [[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/***filename*] **示例：** `Device# copy system:running-config ftp:`	把运行配置或启动配置拷贝到 FTP 服务器上的指定位置。

示例

在 FTP 服务器上保存运行配置文件

以下示例把名为 runfile-confg 的运行配置文件保存到 IP 地址为 172.16.101.101 的远程主机的 netadmin1 目录下：

```
Device# copy system:running-config ftp://netadmin1:mypass@172.16.101.101/runfile-confg
Write file runfile-confg on host 172.16.101.101?[confirm]
Building configuration...[OK]
Connected to 172.16.101.101
Device#
```

在 FTP 服务器上保存启动配置文件

以下示例展示了使用 FTP 把启动配置文件拷贝到服务器上：

```
Device# configure terminal
Device(config)# ip ftp username netadmin2
Device(config)# ip ftp password mypass
Device(config)# end
Device# copy nvram:startup-config ftp:
```

```
Remote host[]? 172.16.101.101
Name of configuration file to write [start-confg]?
Write file start-confg on host 172.16.101.101?[confirm]
![OK]
```

接下来做什么？

输入 **copy** 命令后，系统可能提示用户输入额外信息或让用户确认操作。显示的提示取决于用户在 **copy** 命令中提供了多少信息以及当前的全局配置命令 **file prompt** 设置。

把配置文件从 TFTP 服务器拷贝到设备（CLI）

要把配置文件从 TFTP 服务器拷贝到设备上，请完成以下配置任务：

总步骤

1. **enable**
2. **copy tftp:** [[[**//***location*]**/***directory*]**/***filename*] **system:running-config**
3. **copy tftp:** [[[**//***location*]**/***directory*]**/***filename*] **nvram:startup-config**
4. **copy tftp:** [[[**//***location*]**/***directory*]**/***filename*]**flash-[n]:/directory/startup-config**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**copy tftp:** [[[**//***location*]**/***directory*]**/***filename*] **system:running-config** **示例：** `Device# copy tftp://server1/dir10/datasource system:running-config`	把配置文件从 TFTP 服务器拷贝到运行配置中。
步骤 3	**copy tftp:** [[[**//***location*]**/***directory*]**/***filename*] **nvram:startup-config** **示例：** `Device# copy tftp://server1/dir10/datasource nvram:startup-config`	把配置文件从 TFTP 服务器拷贝到启动配置中。
步骤 4	**copy tftp:** [[[**//***location*]**/***directory*]**/***filename*]**flash-[n]:/directory/startup-config** **示例：** `Device# copy`	把配置文件从 TFTP 服务器拷贝到启动配置中。

	tftp://server1/dir10/datasource flash:startup-config	

示例

以下示例使用 IP 地址 172.16.2.155 上的 tokyo-confg 文件配置系统：

```
Device# copy tftp://172.16.2.155/tokyo-confg system:running-config
Configure using tokyo-confg from 172.16.2.155? [confirm] Y
Booting tokyo-confg from 172.16.2.155:!!! [OK - 874/16000 bytes]
```

接下来做什么？

输入 **copy** 命令后，系统可能提示用户输入额外信息或让用户确认操作。显示的提示取决于用户在 **copy** 命令中提供了多少信息以及当前的全局配置命令 **file prompt** 设置。

把配置文件从 RCP 服务器拷贝到设备（CLI）

要把配置文件从 RCP 服务器拷贝到运行配置或启动配置中，请完成以下配置任务：

总步骤

1. **enable**
2. **configure terminal**
3. **ip rcmd remote-username** *username*
4. **end**
5. 进行以下配置之一：
   - **copy rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*]**system:running-config**
   - **copy rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*]**nvram:startup-config**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	（可选）通过终端进入配置模式。如果覆盖默认远程用户名（见步骤 3），则需要进行此步骤。
步骤 3	**ip rcmd remote-username** *username* **示例：** `Device(config)# ip rcmd remote-username NetAdmin1`	（可选）指定远程用户名。
步骤 4	**end** **示例：** `Device(config)# end`	（可选）退出全局配置模式。如果覆盖了默认远程用户名（见步骤 2），则需要进行此步骤。
步骤 5	进行以下配置之一： • **copy rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/**	把配置文件从 RCP 服务器拷贝到运行配置或启动配置中。

	filename]**system:running-config**
	• **copy**
	rcp:[[[**//**[*username***@**]*location*]**/***directory*]**/**
	filename]**nvram:startup-config**
	示例：
	`Device# copy`
	`rcp://[user1@example.com/dir10/fileone]`
	`nvram:startup-config`	

示例

从 RCP 拷贝 Running-Config

以下示例拷贝了 IP 地址为 172.16.101.101 远程服务器上的 netadmin1 目录中的配置文件 host1-confg，在设备上加载并运行文件中的命令：

```
Device# copy rcp://netadmin1@172.16.101.101/host1-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![OK]
Device#
%SYS-5-CONFIG: Configured from host1-config by rcp from 172.16.101.101
```

从 RCP 拷贝 Startup-Config

以下示例指定使用远程用户名 netadmin1，然后把 IP 地址为 172.16.101.101 远程服务器上的 netadmin1 目录中的配置文件 host2-confg 拷贝到启动配置中。

```
Device# configure terminal
Device(config)# ip rcmd remote-username netadmin1
Device(config)# end
Device# copy rcp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[rtr2-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
Device#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by rcp from
172.16.101.101
```

接下来做什么？

输入 **copy** 命令后，系统可能提示用户输入额外信息或让用户确认操作。显示的提示取决于用户在 **copy** 命令中提供了多少信息以及当前的全局配置命令 **file prompt** 设置。

把配置文件从 FTP 服务器拷贝到设备（CLI）

要把配置文件从 FTP 服务器拷贝到运行配置或启动配置中，请完成以下配置任务：

总步骤

1. enable

2. configure terminal

3. ip ftp username *username*

4. ip ftp password *password*

5. end

6. 进行以下操作之一：

- **copy ftp:** [[[**//**[*username*[**:***password*]**@**]*location*] **/***directory*]**/***filename*]**system:running-config**
- **copy ftp:** [[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/***filename*]**nvram:startup-config**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	（可选）允许进入全局配置模式。如果希望覆盖默认的远程用户名或密码（见步骤 3 和步骤 4），则需要进行此步骤。
步骤 3	**ip ftp username** *username* **示例：** `Device(config)# ip ftp username NetAdmin1`	（可选）指定默认远程用户名。
步骤 4	**ip ftp password** *password* **示例：** `Device(config)# ip ftp password adminpassword`	（可选）指定默认密码。
步骤 5	**end** **示例：** `Device(config)# end`	（可选）退出全局配置模式。如果覆盖默认远程用户名或密码（见步骤 3 和步骤 4），则需要进行此步骤。
步骤 6	进行以下操作之一： • **copy ftp:** [[[**//**[*username*[**:***password*]**@**]*location*] **/***directory*]**/***filename*]**system:running-config** • **copy ftp:** [[[**//**[*username*[**:***password*]**@**]*location*]**/***directory*]**/***filename*]**nvram:startup-config** **示例：** `Device# copy ftp:nvram:startup-config`	使用 FTP 从网络服务器上把配置文件拷贝到运行配置或启动配置中。

示例

从 FTP 拷贝 Running-Config

以下示例拷贝了 IP 地址为 172.16.101.101 远程服务器上的 netadmin1 目录中的配置文件 host1-confg，在设备上加载并运行文件中的命令：

```
Device# copy ftp://netadmin1:mypass@172.16.101.101/host1-confg system:running-config
Configure using host1-confg from 172.16.101.101? [confirm]
Connected to 172.16.101.101
Loading 1112 byte file host1-confg:![OK]
Device#
%SYS-5-CONFIG: Configured from host1-config by ftp from 172.16.101.101
```

从 FTP 拷贝到 Startup-Config

以下示例指定使用远程用户名 netadmin1，然后把 IP 地址为 172.16.101.101 远程服务器上的 netadmin1 目录中的配置文件 host2-confg 拷贝到启动配置中。

```
Device# configure terminal
Device(config)# ip ftp username netadmin1
Device(config)# ip ftp password mypass
Device(config)# end
Device# copy ftp: nvram:startup-config
Address of remote host [255.255.255.255]? 172.16.101.101
Name of configuration file[host1-confg]? host2-confg
Configure using host2-confg from 172.16.101.101?[confirm]
Connected to 172.16.101.101
Loading 1112 byte file host2-confg:![OK]
[OK]
Device#
%SYS-5-CONFIG_NV:Non-volatile store configured from host2-config by ftp from
172.16.101.101
```

接下来做什么？

输入 **copy** 命令后，系统可能提示用户输入额外信息或让用户确认操作。显示的提示取决于用户在 **copy** 命令中提供了多少信息以及当前的全局配置命令 **file prompt** 设置。

维护大于 NVRAM 的配置文件

要维护超过了 NVRAM 大小的配置文件，请执行以下所述的配置任务：

压缩配置文件（CLI）

要压缩配置文件，请完成本节中的配置任务：

总步骤

1. **enable**
2. **configure terminal**
3. **service compress-config**
4. **end**
5. 进行以下操作之一：

- 使用FTP、RCP或TFTP 拷贝新的配置文件。
- **configure terminal**

6. copy system:running-config nvram:startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**service compress-config** **示例：** `Device(config)# service compress-config`	指定要压缩的配置文件。
步骤 4	**end** **示例：** `Device(config)# end`	退出全局配置模式。
步骤 5	进行以下操作之一： • 使用FTP、RCP或TFTP 拷贝新的配置文件。 • **configure terminal** **示例：** `Device# configure terminal`	输入新配置： • 如果尝试加载比 NVRAM 大三倍以上的配置，系统会显示以下错误消息： "[buffer overflow - *file-size* /*buffer-size* bytes]."
步骤 6	**copy system:running-config nvram:startup-config** **示例：** `Device(config)# copy system:running-configngvram:startup-config`	完成对运行配置的修改后，保存新的配置。

示例

以下示例展示了把一个 129 KB 的配置文件压缩到 11 KB 的过程。

```
Device# configure terminal
Device(config)# service compress-config
Device(config)# end
Device# copy tftp://172.16.2.15/tokyo-confg system:running-config
Configure using tokyo-confg from 172.16.2.155? [confirm] y
Booting tokyo-confg from 172.16.2.155:!!! [OK - 874/16000 bytes]
Device# copy system:running-config nvram:startup-config
Building configuration...
Compressing configuration from 129648 bytes to 11077 bytes
[OK]
```

在 A 类闪存文件系统中把配置存储到闪存（CLI）

要把启动配置保存到闪存中，请完成本节的配置任务：

总步骤

1. enable

2. copy nvram:startup-config *flash-filesystem:filename*

3. configure terminal

4. boot config flash-filesystem: *filename*

5. end

6. 进行以下操作之一：

- 使用 FTP、RCP 或 TFTP 拷贝新配置文件。如果尝试加载比 NVRAM 大三倍以上的配置，系统会显示错误消息："[buffer overflow - *file-size* /*buffer-size* bytes]."
- **configure terminal**

7. copy system:running-config nvram:startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**copy nvram:startup-config** *flash-filesystem:filename* **示例：** `Device# copy nvram:startup-config usbflash0:switch-config`	把当前启动配置拷贝到新位置。
步骤 3	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 4	**boot config flash-filesystem:** *filename* **示例：** `Device(config)# boot config usbflash0:switch-config`	设置环境变量 CONFIG_FILE，把启动配置文件保存到闪存中。
步骤 5	**end** **示例：** `Device(config)# end`	退出全局配置模式。
步骤 6	进行以下操作之一： • 使用 FTP、RCP 或 TFTP 拷贝新的配置文件。如果尝试加载比 NVRAM 大三倍以上的配置，系统会显示以下错误消息："[buffer overflow - *file-size* /*buffer-size* bytes]."	输入新配置。

	• **configure terminal** **示例：** `Device# configure terminal`	
步骤 7	**copy system:running-config nvram:startup-config** **示例：** `Device(config)# copy system:running-configngvram:startup-config`	完成对运行配置的修改后，保存新的配置。

示例

以下示例把配置文件保存到 usbflash0 中：

```
Device# copy nvram:startup-config usbflash0:switch-config
Device# configure terminal
Device(config)# boot config usbflash0:switch-config
Device(config)# end
Device# copy system:running-config nvram:startup-config
```

通过网络加载配置命令（CLI）

要使用网络服务器存储大型配置，请完成本节中的配置任务：

总步骤

1. enable

2. copy system:running-config {**ftp:** | **rcp:** | **tftp:**}

3. configure terminal

4. boot network {**ftp:**[[[**//**[*username* [**:***password*]**@**]*location*]**/***directory*]**/***filename*] | **rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*] | **tftp:**[[[**//***location*]**/***directory*]**/***filename*]}

5. service config

6. end

7. copy system:running-config nvram:startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**copy system:running-config** {**ftp:**	**rcp:**	**tftp:**} **示例：** `Device# copy system:running-config ftp:`	把运行配置保存到 FTP、RCP 或 TFTP 服务器上。
步骤 3	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 4	**boot network** {**ftp:**[[[**//**[*username* [**:***password*]**@**]*location*]**/***directory*]**/***filename*]	**rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*]	**tftp:**[[[**//***location*]**/***directory*]**/***filename*]}	指定启动时要通过网络服务器加载的启动配置文件。

	示例： `Device(config)# boot network ftp://user1:guessme@example.com/dir10/file1`	
步骤 5	**service config** **示例：** `Device(config)# service config`	设备交换机在系统启动时下载配置文件。
步骤 6	**end** **示例：** `Device(config)# end`	退出全局配置模式。
步骤 7	**copy system:running-config nvram:startup-config** **示例：** `Device(config)# copy system:running-configgnvram:startup-config`	保存配置。

把闪存中的配置文件拷贝为启动配置或运行配置（CLI）

要把配置文件从闪存直接拷贝到 NVRAM 中的启动配置或拷贝为运行配置，请执行步骤 2 中的命令：

总步骤

1. enable

2. 进行以下操作之一：

- **copy** *filesystem*: [*partition-number*:][*filename*] **nvram:startup-config**
- **copy** *filesystem*: [*partition-number*:][*filename*] **system:running-config**

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	进行以下操作之一： • **copy** *filesystem*: [*partition-number*:][*filename*] **nvram:startup-config** • **copy** *filesystem*: [*partition-number*:][*filename*] **system:running-config** **示例：** `Device# copy usbflash0:4:INOS-upgrade-1 nvram:startup-config`	• 把配置文件直接加载到 NVRAM 中 或 • 把配置文件拷贝为运行配置

示例

以下示例把名为 INOS-upgrade-1 的配置文件从闪存 PC 卡 usbflash0 分区 4 中拷贝到设备的启动配置：

```
Device# copy usbflash0:4:INOS-upgrade-1 nvram:startup-config
```

```
Copy 'INOS-upgrade-1' from flash device as 'startup-config' ? [yes/no] yes
[OK]
```

在闪存文件系统之间拷贝配置文件（CLI）

在有多个闪存文件系统的平台上，可以把一个闪存文件系统中的文件拷贝到另一个闪存中，比如从内部闪存拷贝到其他的闪存文件系统。把文件拷贝到不同的闪存文件系统让用户可以创建运行配置的备份，并对其他设备复用配置。要在闪存文件系统之间拷贝配置文件，请在 EXEC 模式中使用以下命令：

总步骤

1. enable

2. show *source-filesystem*:

3. copy *source-filesystem*: [*partition-number*:][*filename*] *dest-filesystem*:[*partition-number*:][*filename*]

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**show** *source-filesystem*: **示例：** `Device# show flash:`	显示闪存的配置及内容，验证文件名。
步骤 3	**copy** *source-filesystem*: [*partition-number*:][*filename*] *dest-filesystem*:[*partition-number*:][*filename*] **示例：** `Device# copy flash: usbflash0:`	在闪存设备之间拷贝配置文件。 源设备与目的设备不能相同。比如，命令 **copy usbflash0: usbflash0:**是不合法的。

示例

以下示例把名为 running-config 的文件从内部闪存的分区 1 拷贝到设备 usbflash0 上的分区 1 中。此例中源分区未指定，所以设备提示确认分区编号：

```
Device# copy flash: usbflash0:
System flash
Partition Size Used Free Bank-Size State Copy Mode
1 4096K 3070K 1025K 4096K Read/Write Direct
2 16384K 1671K 14712K 8192K Read/Write Direct
[Type ?<no> for partition directory; ? for full directory; q to abort]
Which partition? [default = 1]
System flash directory, partition 1:
File Length Name/status
1 3142748 dirt/network/mars-test/c3600-j-mz.latest
2 850 running-config
[3143728 bytes used, 1050576 available, 4194304 total]
```

```
usbflash0 flash directory:
File Length Name/status
1 1711088 dirt/gate/c3600-i-mz
2 850 running-config
[1712068 bytes used, 2482236 available, 4194304 total]
Source file name? running-config
Destination file name [running-config]?
Verifying checksum for 'running-config' (file # 2)... OK
Erase flash device before writing? [confirm]
Flash contains files. Are you sure you want to erase? [confirm]
Copy 'running-config' from flash: device
as 'running-config' into usbflash0: device WITH erase? [yes/no] yes
Erasing device...
eeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeeee ...erased!
[OK - 850/4194304 bytes]
Flash device copy took 00:00:30 [hh:mm:ss]
Verifying checksum... OK (0x16)
```

把配置文件从 FTP 服务器拷贝到闪存设备（CLI）

要把配置文件从 FTP 服务器拷贝到闪存设备中，请完成本节中的配置任务：

总步骤

1. enable
2. configure terminal
3. ip ftp username *username*
4. ip ftp password *password*
5. end
6. copy ftp: [[**//***location*]**/***directory*]**/***bundle_name* **flash:**

具体步骤

	命令或操作	目的
步骤 1	**enable** 示例： `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** 示例： `Device# configure terminal`	（可选）进入全局配置模式。如果覆盖默认的远程用户名或密码（见步骤 3 和步骤 4），则需要进行此步骤。
步骤 3	**ip ftp username** *username* 示例： `Device(config)# ip ftp username Admin01`	（可选）指定远程用户名。
步骤 4	**ip ftp password** *password* 示例： `Device(config)# ip ftp password`	（可选）指定远程密码。

	adminpassword	
步骤 5	**end** **示例：** Device(config)# end	（可选）退出配置模式。如果覆盖默认的远程用户名或密码（见步骤 3 和步骤 4），则需要进行此步骤。
步骤 6	**copy ftp:** [[**//***location*]**/***directory*]**/***bundle_name* **flash:** **示例：** Device>copyftp:/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.binflash:	使用 FTP 把配置文件从网络服务器拷贝到闪存设备中。

接下来做什么？

输入 **copy** 命令后，系统可能提示用户输入额外信息或让用户确认操作。显示的提示取决于用户在 **copy** 命令中提供了多少信息以及当前的全局配置命令 **file prompt** 设置。

把配置文件从 RCP 服务器拷贝到闪存设备（CLI）

要把配置文件从 RCP 服务器拷贝到闪存设备，请完成本节的配置任务：

总步骤

1. enable

2. configure terminal

3. ip rcmd remote-username *username*

4. end

5. copy rcp: [[[**//**[*username***@**]*location*]**/***directory*] **/***bundle_name*] **flash:**

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** Device> enable	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** Device# configure terminal	（可选）进入全局配置模式。如果覆盖默认的远程用户名（见步骤 3），则需要进行此步骤。
步骤 3	**ip rcmd remote-username** *username* **示例：** Device(config)# ip rcmd remote-username Admin01	（可选）指定远程用户名。
步骤 4	**end** **示例：** Device(config)# end	（可选）退出配置模式。如果覆盖默认的远程用户名（见步骤 3），则需要进行此步骤。
步骤 5	**copy rcp:** [[[**//**[*username***@**]*location*]**/***directory*] **/***bundle_name*] **flash:**	使用 RCP 把配置文件从网络服务器拷贝到闪存设备上。请响应或确

	示例： `Device# copyrcp://netadmin@172.16.101.101/bundle1 flash:`	认设备对其他信息的请求提示。显示的提示取决于用户在 **copy** 命令中提供了多少信息以及当前的全局配置命令 **file prompt** 设置。

把配置文件从 TFTP 服务器拷贝到闪存设备（CLI）

要把配置文件从 TFTP 服务器拷贝到闪存设备，请完成本节的配置任务：

总步骤

1. enable

2. copy tftp: [[[**//***location*]**/***directory*]**/**bundle_name **flash:**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**copy tftp:** [[[**//***location*]**/***directory*]**/**bundle_name **flash:** **示例：** `Device#copy tftp:/cat3k_caa-universalk9.SSA.03.12.02.EZP.150-12.02.EZP.150-12.02.EZP.bin flash:`	把配置文件从 TFTP 网络服务器拷贝到闪存设备上。请响应或确认设备对其他信息的请求提示。显示的提示取决于用户在 **copy** 命令中提供了多少信息以及当前的全局配置命令 **file prompt** 设置。

示例

以下示例展示了把名为 switch-config 的配置文件从 TFTP 服务器拷贝到 usbflash0 的闪存卡的过程。拷贝的文件被重命名为 new-config。

```
Device#
copy tftp:switch-config usbflash0:new-config
```

重新执行启动配置文件中的配置命令（CLI）

要重新执行启动配置文件中的命令，请完成本节的配置任务：

总步骤

1. enable

2. configure memory

具体步骤

	命令或操作	目的

步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure memory** **示例：** `Device# configure memory`	重新执行配置文件中的配置命令。

清除启动配置（CLI）

可以清除启动配置中的配置信息。如果重启设备时没有启动配置，设备会进入 Setup 命令系统，用户可以从头开始配置设备。要清除启动配置的内容，请完成本节的配置任务：

总步骤

1. enable

2. erase nvram

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**erase nvram** **示例：** `Device# erase nvram`	清除启动配置的内容。 **注释：**对除了A类闪存文件系统平台以外的所有平台来说，此命令会擦除NVRAM。在A类闪存文件系统平台上，使用EXEC命令**erasestartup-config**时，设备会删除环境变量CONFIG_FILE指向的配置。如果变量指向NVRAM，则设备会擦除NVRAM。如果环境变量CONFIG_FILE指向了一个闪存设备及文件名，设备会删除该配置文件。即设备会把文件标记为“已删除”，但不会擦除文件。此特性允许用户恢复被删除的文件。

删除指定的配置文件（CLI）

要删除指定闪存设备上的指定配置，请完成本节的配置任务：

总步骤

1. enable

2. delete *flash-filesystem*:*filename*

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**delete** *flash-filesystem*:*filename* **示例：** `Device# deleteusbflash0:myconfig`	删除指定闪存设备上的指定配置文件。注释：在 A 类以及 B 类闪存文件系统中，删除闪存的指定文件时系统会把文件标记为已删除。这允许用户在以后使用 **undelete** EXEC 命令恢复已删除的文件。已擦除的文件不能被恢复。要永久擦除配置文件，需使用 **squeeze** EXEC 命令。在 C 类闪存文件系统中，不能恢复已被删除的文件。如果尝试擦除或删除环境变量 CONFIG_FILE 指定的配置文件，系统会提示用户确认删除操作。

指定 A 类闪存文件系统的 CONFIG_FILE 环境变量（CLI）

在 A 类闪存文件系统中，可以配置 Inspur INOS 软件加载由 CONFIG_FILE 环境变量指定的启动配置文件。CONFIG_FILE 变量默认指向 NVRAM。要更改 CONFIG_FILE 环境变，请完成本节的配置任务：

总步骤

1. enable

2. copy [*flash-url* | *ftp-url* | *rcp-url* | *tftp-url* | **system:running-config** | **nvram:startup-config**] *dest-flash-url*

3. configure terminal

4. boot config *dest-flash-url*

5. end

6. copy system:running-config nvram:startup-config

7. show boot

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**copy** [*flash-url* | *ftp-url* | *rcp-url* | *tftp-url* | **system:running-config** | **nvram:startup-config**] *dest-flash-url* **示例：** `Device# copy system:running-config nvram:startup-config`	把配置文件从设备加载重启使用的位置拷贝到闪存文件系统中。
步骤 3	**configure terminal**	进入全局配置模式。

	示例： `Device# configure terminal`	
步骤 4	**boot config** *dest-flash-url* **示例：** `Device(config)# boot config 172.16.1.1`	设置 CONFIG_FILE 环境变量。此步骤会修改运行时的 CONFIG_FILE 变量值。
步骤 5	**end** **示例：** `Device(config)# end`	退出全局配置模式。
步骤 6	**copy system:running-config nvram:startup-config** **示例：** `Device# copy system:running-config nvram:startup-config`	把步骤 3 中执行的配置保存到启动配置中。
步骤 7	**show boot** **示例：** `Device# show boot`	（可选）验证 CONFIG_FILE 环境变量的内容。

示例

以下示例把运行配置文件拷贝到设备中。此配置在随后系统重启时会被用作启动配置：

```
Device# copy system:running-config usbflash0:config2
Device# configure terminal
Device(config)# boot config usbflash0:config2
Device(config)# end
Device# copy system:running-config nvram:startup-config
[ok]
Device# show boot
BOOT variable = usbflash0:rsp-boot-m
CONFIG_FILE variable = nvram:
Current CONFIG_FILE variable = usbflash0:config2
Configuration register is 0x010F
```

接下来做什么？

指定了启动配置文件的位置后，**nvram:startup-config** 命令会等同于启动配置文件的新位置。**more nvram:startup-config** EXEC 命令会显示启动配置文件，无论其位置如何。**erase nvram:startup-config** EXEC 命令会擦除 NVRAM 的内容，并删除 CONFIG_FILE 环境变量指向的文件。

使用 **copy system:running-config nvram:startup-config** 命令保存配置时，设备会把配置文件的完整版本保存到 CONFIG_FILE 环境变量指定的位置，并把一份提取版本保存到 NVRAM。提取的版本不包含访问列表信息。如果 NVRAM 中包含完整的配置文件，设备会提示用户确认使用提取版本覆盖完整版本。如果 NVRAM 中包含提取配置，设备不会提示用户确认操作，而是覆盖 NVRAM 中的现有提取配置文件。

注释： 如果把闪存设备中的一个文件指定为 CONFIG_FILE 环境变量的值，每次使用 **copy system:running-config nvram:startup-config** 命令保存配置文件时，旧的配置文件都会被标记为“已删除”，且新的配置文件会被保存到该设备。最终，闪存会被填满，因为旧的配置文件

仍然占用空间。使用 **squeeze** EXEC 命令永久删除旧配置文件并回收空间。

配置设备下载配置文件

可以指定一个网络配置文件名以及主机配置文件名的有序列表。Inspur INOS 软件会扫描此列表，直到加载了恰当的网络或主机配置文件。

要配置设备在系统启动时下载配置文件，请执行以下所述任务中的至少一个：

- 配置设备下载网络配置文件（CLI）
- 配置设备下载主机配置文件（CLI）

如果设备启动时不能加载配置文件，它会每 10 分钟尝试一次（默认设置），直到主机提供了请求的文件。每次请求失败时，设备会在控制台终端上显示以下消息：

```
Booting host-confg... [timed out]
```

如果启动配置文件存在问题，或配置寄存器被设置为忽略 NVRAM，设备会进入 Setup 命令系统。

配置设备下载网络配置文件（CLI）

要配置 Inspur INOS 软件在启动时从服务器上下载网络配置文件，请完成本节中的配置任务：

总步骤

1. enable

2. configure terminal

3. boot network {**ftp:**[[[**//**[*username* [**:***password*]**@**]*location*]**/***directory*]**/***filename*] | **rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*] | **tftp:**[[[**//***location*]**/***directory*]**/***filename*]}

4. service config

5. end

6. copy system:running-config nvram:startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**boot network** {**ftp:**[[[**//**[*username* [**:***password*]**@**]*location*]**/***directory*]**/***filename*] |**rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*]	**tftp:**[[[**//***location*]**/***directory*]**/***filename*]} **示例：** `Device(config)# boot network tftp:hostfile1`	指定启动时要下载网络配置文件以及要使用的协议（TFTP、RCP 或 FTP）。 • 如果不指定网络配置文件名，Inspur INOS 软件会使用默认的文件名 network-confg。如果省略了地址，设备会使用广播地址。 • 可以指定多个网络配置文件。软件会按照输入顺序尝试加

		载，直到成功加载了一个文件。此过程可以用来保存加载到网络服务器上的配置信息不同的文件。
步骤 4	**service config** **示例：** `Device(config)# service config`	让系统在重启时自动加载网络文件。
步骤 5	**end** **示例：** `Device(config)# end`	退出全局配置模式。
步骤 6	**copy system:running-config nvram:startup-config** **示例：** `Device# copy system:running-config nvram:startup-config`	把运行配置保存到启动配置中。

配置设备下载主机配置文件（CLI）

要配置 Inspur INOS 软件在启动时从服务器上下载主机配置文件，请完成本节中的配置任务：

总步骤

1. enable

2. configure terminal

3. boot host {**ftp:**[[[**//**[*username* [**:***password*]**@**]*location*]**/***directory*]**/***filename*] | **rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*] | **tftp:**[[[**//***location*]**/***directory*]**/***filename*] }

4. service config

5. end

6. copy system:running-config nvram:startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** `Device> enable`	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**boot host** {**ftp:**[[[**//**[*username* [**:***password*]**@**]*location*]**/***directory*]**/***filename*] |**rcp:**[[[**//**[*username***@**]*location*]**/***directory*]**/***filename*] | **tftp:**[[[**//***location*]**/***directory*]**/***filename*] } **示例：** `Device(config)# boot host tftp:hostfile1`	指定启动时要下载主机配置文件以及要使用的协议（TFTP、RCP 或 FTP）。 • 如果不指定网络配置文件名，设备会使用自己的名称来构建主机配置文件名，把名称变为小写，移除所有域信息，并附加“-confg”。如果没有可用的主机名信息，系统会使用默认的主机配置文件名 device-confg。如果省略

		了地址，设备会使用广播地址。 • 可以指定多个主机配置文件。Inspur INOS 软件会按照输入顺序尝试加载，直到成功加载了一个文件。此过程可以用来保存加载到网络服务器上的配置信息不同的文件。
步骤 4	**service config** **示例：** `Device(config)# service config`	让系统在重启时自动加载主机文件。
步骤 5	**end** **示例：** `Device(config)# end`	退出全局配置模式。
步骤 6	**copy system:running-config nvram:startup-config** **示例：** `Device# copy system:running-config nvram:startup-config`	把运行配置保存到启动配置中。

示例

以下示例配置设备下载名为 hostfile1 的主机配置文件以及名为 networkfile1 的网络配置文件。设备会使用 TFTP 以及广播地址来获取文件：

```
Device# configure terminal
Device(config)# boot host tftp:hostfile1
Device(config)# boot network tftp:networkfile1
Device(config)# service config
Device(config)# end
Device# copy system:running-config nvram:startup-config
```

其他参考资料

相关文档

相关主题	文档标题
Inspur INOS 命令	Inspur INOS 主命令列表，所有版本
Inspur INOS 配置命令	*Inspur INOS 配置基础命令参考*

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

标准

标准	标题
不支持新标准或修订的标准，且支持的现有标准未被修改	-

RFC

RFC	标题
不支持新 RFC 或修订的 RFC，且支持的现有 RFC 未被修改	-

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

配置替换与配置回滚

配置替换与配置回滚的前提

用作配置替换与配置回滚特性的输入配置文件格式必须符合以下标准 Inspur 软件配置文件缩进规则：

- 在新行中开始所有命令，除非命令在配置子模式中，否则无缩进。
- 第一级配置子模式中的命令使用一个空格缩进。
- 第二级配置子模式中的命令使用两个空格缩进。
- 后续子模式中的命令按此规则缩进。

这些缩进规则描述了软件是如何为**showrunning-config**或**copy running-config** *destination-url*这样的命令创建配置文件的。Inspur设备上生成的任意配置文件都符合这些规则。
需要有比当前运行配置与保存的替换配置文件大小之和更大的空闲内存空间。

配置替换与配置回滚的限制

如果设备的空闲空间小于两配置文件（当前运行配置与保存的替换配置）的大小之和，配置替换操作不会进行。
特定的Inspur配置命令不能添加到运行配置或从中移除，如网络设备物理组件（如物理接口）附加的命令。如果Ethernet 0接口物理存在于设备上，配置替换操作不能从当前的运行配置中移除**interface ethernet 0**命令。类似的，如果Ethernet 1接口不存在于设备上，**interfaceethernet 1**命令不能被添加到运行配置中。尝试执行这些类型更改的配置替换操作会产生错误消息，表示这些特定的命令行替换失败。
在极少数情况下，如果不重载设备，特定的Inspur配置命令不能从运行配置中移除。尝试移除此类命令的配置替换操作会产生错误消息，表示这些特定的命令行替换失败。

关于配置替换与配置回滚的信息

配置存档

Inspur INOS 配置存档功能旨在提供一种存储、组织及管理 Inspur INOS 配置文件存档的机制，增强 **configure replace** 命令提供的配置回滚能力。在引入此特性之前，用户可以使用命令 **copy running-config** *destination-url* 保存运行配置的副本，本地或远程存储替换文件。然而，这种方式缺少自动化的文件管理能力。配置替换与配置回滚特性有能力自动把运行配置的副本保存为 Inspur INOS 配置存档。这些存档文件作为检查点配置参考，可以被 **configure replace** 命令用来回退到之前的配置状态。
archive config 命令允许用户把 Inspur INOS 配置保存到配置存档中。存档配置会使用标准位置以及文件名前缀保存，并为连续的文件附加上递增版本号（以及可选的时间戳）。此功能提供了一种对保存的 Inspur INOS 配置文件一致性标识的方式。可以指定在存档中保存运行配置的多少个版本。存档中保存了最大数量的文件之后，保存下一个新文件时最旧的文件会被自动删除。**show archive** 命令会显示保存在 Inspur INOS 配置存档中的所有配置文件信息。Inspur INOS 配置存档中保存着配置文件，并可为 **configure replace** 命令使用，存档可位于以下文件系统中：FTP、HTTP、RCP 以及 TFTP。

配置替换

特权 EXEC 命令 **configure replace** 提供了使用保存的 Inspur INOS 配置文件替换当前运行配置的能力。此功能可以用来回退到之前的配置状态，能有效地回滚自之前的配置状态保存以来进行的所有配置更改。
使用命令时，必须指定一个保存的Inspur INOS配置作为当前运行配置的替换配置文件。替换文件必须是Inspur INOS设备生成的完整配置（如**copyrunning-config** *destination-url*命令生成的配置），如果替换文件是外部生成的，其必须符合Inspur INOS设备生成文件的格式。输入**configure replace**命令时，当前运行配置会与指定的替换配置进行比较，并会产生一组diffs（文件差异）。用于比较两个文件的算法与**show archiveconfig differences**命令使用的算法相同。最终的diffs会被Inspur INOS解析器使用，以应用替换配置的状态。此过程中只有diffs会被应用，进而避免了因重新应用当前运行配置中已存在的配置命令而导致的潜在的服务中断可能。此算法能通过多遍处理过程有效地解决对于顺序相关命令（如访问列表）的配置

更改。在正常情况下，三遍以下的处理就能完成配置替换操作；限制最多执行五遍，排除任何循环行为。

Inspur INOS特权EXEC命令**copy** *source-url* **running-config**常被用来把保存的Inspur INOS配置文件拷贝到运行配置中。使用**copy** *source-url* **running-config**命令替代特权EXEC命令**configure replace** *target-url*时，应注意以下几点主要的区别：

- **copy** *source-url* **running-config**命令是合并操作，会保留源文件以及当前运行配置中的所有命令。此命令不会从当前运行配置中移除源文件中没有的命令。相比之下，**configure replace** *target-url*命令会从当前运行配置中移除源文件中没有的命令，并把需要添加的命令添加到当前运行配置中。
- **copy** *source-url* **running-config**命令会应用源文件中的每一条命令，无论该命令是否已经存在于当前的运行配置中。此算法是低效的，且有时会导致服务中断。相比之下，**configure replace** *target-url*只应用需要被应用的命令——当前运行配置中已有命令不会被重新应用。
- 部分的配置文件可以用作 **copy** *source-url* **running-config** 命令的源文件，而 **configure replace** *target-url* 命令的替换文件必须使用完整的 Inspur INOS 配置文件。

配置替换操作引入了锁特性。使用 **configure replace** 命令时，运行配置文件在配置替换操作过程中会被锁住。锁机制避免了其他用户在替换操作进行时更改运行配置，这样的更改可能导致替换操作不成功终止。可以在输入 **configure replace** 命令时使用 **no lock** 关键字来禁用运行配置锁。

运行配置锁在配置替换操作结束时会被自动清除。可以使用**show configurationlock**命令显示当前可能应用到运行配置上的锁。

配置回滚

回滚的概念来源于数据库操作中常用的事务处理模型。在数据库事务中，用户可能对一个数据库表进行一组更改操作。之后用户必须选择是提交更改（永久应用更改）还是回滚更改（丢弃更改并退回到表的之前状态）。在这种情况下，回滚意味着包含更改记录的日志文件被丢弃，且更改不被应用。回滚操作的结果是在更改应用之前回退到之前的状态。

configure replace 命令允许用户回退到之前的配置状态，高效地回滚自之前配置状态保存以来进行的配置更改。Inspur INOS 配置回滚功能不会回滚已应用的一组特定更改，而是使用回退到特定配置状态的概念，基于已保存的 Inspur INOS 配置文件进行回滚操作。此概念类似于数据库中通过保存检查点（一个保存的数据库版本）来保留特定状态的理念。

如果希望使用配置回滚功能，在执行任何配置更改前必须保存 Inspur INOS 的运行配置。之后用户可以输入配置更改，并使用保存的配置文件回滚更改（使用 **configure replace** *target-url* 命令）。此外，如同一些回滚模型一样，因为可以指定使用任何保存的 Inspur INOS 配置文件作为替换配置，用户只能被限制执行固定数量的回滚操作。

配置回滚已确认的更改

配置回滚已确认的更改特性允许在进行配置更改时可以对更改进行确认。如果没有收到确认，配置会返回应用更改前的状态。此特性能防护因配置更改而无意中导致的网络设备以及用户或管理应用的连接性丢失。

配置替换与配置回滚的益处

- 允许用户回退到之前的配置状态，高效地回滚配置更改。

- 允许用户使用在不进行设备重启且无需手动撤销 CLI 对运行配置更改的情况下，使用启动配置文件替换当前运行配置文件，进而检查系统停机时间。
- 允许用户回退到任何保存的 Inspur INOS 配置状态。
- 简化配置更改操作，允许用户给设备应用完整的配置文件，且仅有需要被添加或移除的命令会受影响。
- 使用 **configure replace** 命令替代 **copy** *source-url* **running-config** 命令，能提高效率并避免重新应用已存在命令而产生的服务中断风险。

如何使用配置替换与配置回滚

创建配置存档（CLI）

使用 **configure replace** 命令无需先决配置。可以把 **configure replace** 命令与 Inspur 配置存档的 **archive config** 命令组合使用，这能为配置回滚提供显著的好处。使用 **archive config** 命令之前，必须设置配置存档。执行此任务来进行配置存档特性设置。

总步骤

1. enable
2. configure terminal
3. archive
4. path *url*
5. maximum *number*
6. time-period *minutes*
7. end
8. archive config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**archive** **示例：** `Device(config)# archive`	进入存档配置模式。

步骤 4	**path** *url*
	示例：
	`Device(config-archive)#`
	`pathflash:myconfiguration`
		指定 Inspur INOS 配置存档的位置以及文件名前缀。
		注释：如果 **path** 中指定一个目录，目录名必须后跟一个正斜线，如 **path** flash:/directory/。文件名后无需正斜线。

步骤 5	**maximum** *number* **示例：** `Device(config-archive)# maximum14`	（可选）设置要保存在 Inspur INOS 配置存档中的运行配置存档的最大数量。 • *number* 参数是要保存在 Inspur INOS 配置存档中的运行配置存档的最大数量。合法值范围从 1 到 14，默认值是 10。 **注释：**使用此命令前，必须配置 **path** 命令，指定 Inspur INOS 配置存档的位置以及文件名前缀。
步骤 6	**time-period** *minutes* **示例：** `Device(config-archive)#time-period 1440`	（可选）设置在 Inspur INOS 配置存档中自动保存当前运行配置存档文件的时间增量。 • *minutes* 参数以分钟为单位指定在 Inspur INOS 配置存档中自动保存当前运行配置存档文件的频率。 **注释：**使用此命令前，必须配置 **path** 命令，指定 Inspur INOS 配置存档的位置以及文件名前缀。
步骤 7	**end** **示例：** `Device(config-archive)# end`	返回特权 EXEC 模式。
步骤 8	**archive config** **示例：** `Device# archive config`	把当前运行配置文件保存到配置存档中。 **注释：**使用此命令前必须配置 **path** 命令。

执行配置替换或配置回滚操作（CLI）

执行此任务，使用保存的 Inspur INOS 配置文件替换当前的运行配置文件。

注释： 执行此过程之前必须创建配置存档，详细步骤见*创建配置存档（CLI）*。以下过程详述了在当前运行配置存在问题时如何返回存档的配置。

总步骤

1. enable

2. configure replace *target-url* [**nolock**] [**list**] [**force**] [**ignore case**] [**revert trigger** [**error**][**timer** *minutes*]| **time** *minutes*]]

3. configure revert { **now** | **timer** {*minutes* | **idle** *minutes*} }

4. configure confirm

5. exit

具体步骤

	命令或操作	目的

步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式，在提示时输入密码。
步骤 2	**configure replace** *target-url*[**nolock**] [**list**] [**force**] [**ignorecase**] [**revert trigger** [**error**][**timer** *minutes*] | **time** *minutes*]] **示例：** `Device# configure replace flash: startup-config time120`	使用保存的 Inspur INOS 配置文件替换当前的运行配置文件。 • *target - url*参数是要替换当前运行配置的保存的Inspur INOS配置文件的URL（可被Inspur INOS文件系统访问）。这些配置文件可以使用**archiveconfig**命令创建。 • **list**关键字列出Inspur INOS软件解析器在每遍配置替换操作中使用的命令。执行的总解析次数也会被显示。 • **force**关键字会使用指定的保存Inspur INOS配置文件替换当前运行配置文件，且不提示用户进行确认。 • **time** *minutes*关键字及参数指定确认时间（单位为分钟），在此时间内用户必须输入**configure confirm**命令确认替换当前的运行配置文件。如果在指定的时间限制之内没有输入**configure confirm**命令，配置替换操作会被自动撤销（换句话说，当前的运行配置文件会被恢复为输入**configurereplace**命令之前的配置状态）。 • **nolock**关键字会禁用运行配置文件锁。运行配置文件锁会防止配置替换操作期间用户更改运行配置。 • **revert trigger** 关键字会设置在以下事件触发时回退到原始配置： ◦ **error**——在错误时回退到原始配置。 ◦ **timer** *minutes*——如果超过了指定的时间，回退到原始配置。 • **ignore case** 关键字允许配置忽略确认命令的大小写。
步骤 3	**configure revert** { **now** | **timer**{*minutes* |	（可选）在特权 EXEC 模式中使用

	idle *minutes*} } **示例：** `Device# configure revert now`	**configure revert** 命令，取消计时的回退并立即触发回退操作，或重置计时回退参数。 • **now**——立即触发回退。 • **timer**——重置回退计时器设置。 • 使用 **timer** 关键字以及 *minutes* 参数指定新的回退时间。 • 使用 **idle** 关键字以及分钟数来设置回退到保存配置之前允许的最大无活动时间。
步骤 4	**configure confirm** **Example:** `Device# configure confirm`	（可选）确认使用保存的 Inspur INOS 配置文件替换当前的运行配置文件。 **注释：**仅在指定了 **configure replace** 命令的 **time** *seconds* 关键字及参数时使用此命令。
步骤 5	**exit** **示例：** `Device# exit`	返回用户 EXEC 模式。

特性监控及故障排除（CLI）

要对配置替换与配置回滚特性进行监控与故障排除，请执行此任务。

总步骤

1. enable

2. show archive

3. debug archive versioning

4. debug archive config timestamp

5. exit

具体步骤

步骤1 **enable**

使用此命令启用特权 EXEC 模式，在提示时输入密码。

示例：

```
Device>enable
Device#
```

步骤 2 **show archive**

使用此命令显示保存在 Inspur INOS 配置存档中的文件信息。

示例：

```
Device# show archive
There are currently 1 archive configurations saved.
The next archive file will be named flash:myconfiguration-2
Archive #    Name
0
```

```
1            flash:myconfiguration-1 <- Most Recent
2
3
4
5
6
7
8
9
10
11
12
13
14
```

以下是运行配置的几个存档文件被保存之后的 **show archive** 命令示例输出。此例中，存档文件的最大数量被设置为 **3**。

示例：

```
Device# show archive
There are currently 3 archive configurations saved.
The next archive file will be named flash:myconfiguration-8
Archive #    Name
0
1            :Deleted
2            :Deleted
3            :Deleted
4            :Deleted
5            flash:myconfiguration-5
6            flash:myconfiguration-6
7            flash:myconfiguration-7 <- Most Recent
8
9
10
11
12
13
14
```

步骤 3　debug archive versioning

使用此命令启用 Inspur INOS 配置存档活动的调试，帮助进行配置替换与回滚的监控及故障排除。

示例：

```
Device# debug archive versioning
Jan 9 06:46:28.419:backup_running_config
Jan 9 06:46:28.419:Current = 7
Jan 9 06:46:28.443:Writing backup file flash:myconfiguration-7
```

```
Jan 9 06:46:29.547: backup worked
```

步骤 4 **debug archive config timestamp**

使用此命令显示每一步配置替换操作的处理时间以及处理的配置文件的大小的调试信息。

示例：

```
Device# debug archive config timestamp
Device# configure replace flash:myconfiguration force
Timing Debug Statistics for INOS Config Replace operation:
    Time to read file usbflash0:sample_2.cfg = 0 msec (0 sec)
    Number of lines read:55
    Size of file 1054
Starting Pass 1
    Time to read file system:running-config = 0 msec (0 sec)
    Number of lines read:93
    Size of file :2539
    Time taken for positive rollback pass = 320 msec (0 sec)
    Time taken for negative rollback pass = 0 msec (0 sec)
    Time taken for negative incremental diffs pass = 59 msec (0 sec)
    Time taken by PI to apply changes = 0 msec (0 sec)
    Time taken for Pass 1 = 380 msec (0 sec)
Starting Pass 2
    Time to read file system:running-config = 0 msec (0 sec)
    Number of lines read:55
    Size of file 1054
    Time taken for positive rollback pass = 0 msec (0 sec)
    Time taken for negative rollback pass = 0 msec (0 sec)
    Time taken for Pass 2 = 0 msec (0 sec)
Total number of passes:1
Rollback Done
```

步骤 5 **exit**

使用此命令返回用户 EXEC 模式。

示例：

```
Device# exit
Device>
```

配置替换与配置回滚的配置示例

创建配置存档

以下示例展示了如何进行 Inspur INOS 配置存档的初始配置。此例中，指定 flash:myconfiguration 作为配置存档的位置以及文件名前缀，设置保存的最大存档文件数量为 10。

```
configure terminal
!
```

```
archive
 path flash:myconfiguration
 maximum 10
end
```

使用保存的 Inspur INOS 配置文件替换当前的运行配置

以下示例展示了如何使用名为 flash:myconfiguration 的保存 Inspur INOS 配置文件替换当前的运行配置。**configure replace** 命令会交互式地提示用户确认操作。

```
Device# configure replace flash:myconfiguration
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
Total number of passes: 1
Rollback Done
```

在以下示例中，指定的 **list** 关键字会显示在配置替换操作期间要应用的命令行：

```
Device# configure replace flash:myconfiguration list
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
!Pass 1
!List of Commands:
no snmp-server community public ro
snmp-server community mystring ro
end
Total number of passes: 1
Rollback Done
```

回退到启动配置文件

以下示例展示了如何使用 **configure replace** 命令回退到 Inspur INOS 启动配置文件。此例也使用 **force** 关键字覆盖交换式的用户提示：

```
Device# configure replace flash:startup-config force
Total number of passes: 1
Rollback Done
```

执行带有 configure confirm 命令的配置替换操作

以下示例展示使用 **configure replace** 命令以及 **time** *minutes* 关键字。必须在指定的时间内输入 **configure confirm** 命令，确认替换当前的运行配置文件。如果在指定的时间限制内没有输

入 **configure confirm** 命令，配置替换操作会自动撤销（换句话说，当前的运行配置文件会被恢复为输入 **configurereplace** 命令之前的配置状态）。

```
Device# configure replace flash:startup-config time 120
This will apply all necessary additions and deletions
to replace the current running configuration with the
contents of the specified configuration file, which is
assumed to be a complete configuration, not a partial
configuration. Enter Y if you are sure you want to proceed. ? [no]: Y
Total number of passes: 1
Rollback Done
Device# configure confirm
```

以下示例展示了使用 **configure revert** 命令以及 **timer** 关键字。要取消计时回滚并立即触发回滚，或者重置计时回滚的参数，必须输入 **configure revert** 命令。

```
Device# configure revert timer 100
```

执行配置回滚操作

以下示例展示了如何更改当前的运行配置，并在此后回滚更改。作为配置回滚操作的一部分，必须在更改文件之前保存当前的运行配置。此例中，**archive config** 命令被用来保存当前的运行配置。**configure replace** 命令生成的输出表示完成回滚仅执行了一遍操作。

注释： 使用 **archive config** 命令之前，必须配置 **path** 命令指定 Inspur INOS 配置存档的位置以及文件名前缀。

先把当前运行配置保存到配置存档中：

```
archive config
```

更改运行配置文件后，假设希望回滚这些更改并返回到进行更改前的配置。**show archive** 命令用来验证用作替换文件的配置版本。**configure replace** 命令用来回退替换配置文件：

```
Device# show archive
There are currently 1 archive configurations saved.
The next archive file will be named flash:myconfiguration-2
Archive #    Name
  0
  1          flash:myconfiguration-1 <- Most Recent
  2
  3
  4
  5
  6
  7
  8
  9
  10
Device# configure replace flash:myconfiguration-1
Total number of passes: 1
Rollback Done
```

其他参考资料

相关文档

相关主题	文档标题
配置锁	*独占配置更改访问以及访问会锁*
管理配置文件的命令	*Inspur INOS 配置基础命令参考*
关于管理配置文件的信息	*管理配置文件*

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

标准

标准	标题
不支持新标准或修订的标准，且支持的现有标准未被修改	-

RFC

RFC	标题
不支持新 RFC 或修订的 RFC，且支持的现有 RFC 未被修改	-

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

VLAN

配置 VTP

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

配置 VTP 的前提条件

用户在创建 VLAN 之前，必须确定是否在网络中使用 VTP（VLAN Trunking Protocol，VTP）。使用 VTP，用户可以集中在一个或多个设备上进行配置变更，这些变更的配置可以被自动通告给网络中的其他设备。不使用 VTP，用户无法向其他设备发送关于 VLAN 的信息。

VTP 的工作环境要求更新要在一个单独的设备上进行，更新的内容通过 VTP 发送到域中的其他设备上。如果有很多对 VLAN 数据库的更新同时发生在一个域中的多台设备上，这会导致 VLAN 数据库不一致。

设备总共支持 4094 个 VLAN。但是，配置的特性数量会影响设备硬件的使用。如果 VTP 通告设备有一个新 VLAN，且此时设备已经在使用最大可用硬件资源，那么设备会发送一条消息指出没有足够的硬件资源可用，然后关闭 VLAN。**show vlan** 用户 EXEC 命令的输出显示 VLAN 处于挂起状态。

由于 VTP 通告通过中继端口发送和接收，所以用户必须确保在设备或设备堆栈上至少配置一个中继端口，并且此中继端口必须与另一个设备的中继端口相连通。否则，设备无法接收任何 VTP 通告。

配置 VTP 的限制条件

配置 VTP 的限制条件如下所示：

- 用户不能在设备堆栈中混用 Inspur 3850 和 Inspur 6650 交换机。

注意： 在将 VTP 客户端添加到 VTP 域之前，请先确认其 VTP 配置修订号低于 VTP 域中其他设备的配置修订号。VTP 域中的设备会使用 VTP 配置修订号最高设备的 VLAN 配置。如果用户添加的设备修订号高于 VTP 域中的修订号，那么该设备可以清除 VTP 服务器和 VTP 域中所有的 VLAN 信息。

关于 VTP 的信息

VTP

VTP 是一种二层消息传递协议，在全网的基础上管理 VLAN 的添加、删除和重命名，从而维护 VLAN 配置的一致性。VTP 能减少由于错误配置或配置不一致导致的一系列问题，例如重复的 VLAN 名称，错误的 VLAN 类型规范和安全违规。

VTP 的功能支持跨堆栈工作，并且堆栈中的所有设备都维护从活跃设备继承的相同 VLAN 和 VTP 配置。当设备通过 VTP 消息得知有新的 VLAN 出现，或当用户配置了新的 VLAN，该新 VLAN 的信息将传播到堆栈中的其他所有交换机上。

当一台设备加入堆栈或者多个堆栈进行合并的时候，新设备会从活跃设备获取到 VTP 信息。

VTP 域

VTP 域（也成为 VLAN 管理域），由一个或一个以上共享 VTP 域名的具有相同管理责任的、相互连接的设备或设备堆栈组成。一个设备只能在一个 VTP 域中工作。用户可以在域中做全局 VLAN 配置的更改。

默认情况下，设备会处于 VTP 非管理域（no-management-domain）状态，直到交换机通过中继链路（携带很多 VLAN 流量的链路）接收到关于域的通告，或者用户在交换机上配置了域名。只有当管理域的域名被确定或者被习得，用户才能在 VTP 服务器上创建或修改 VLAN，且更改 VLAN 的消息才能在网络上传播。

如果设备通过中继链路收到了 VTP 通告，该设备会继承管理域的域名和 VTP 配置的修订号。然后设备会忽略来自其他域名或修订号更小的通告。

当用户在 VTP 服务器上对 VLAN 配置进行了更改，这些变更会通告到 VTP 域中的所有设备。VTP 通告会通过所有的 IEEE 中继连接进行传播，包括 IEEE802.1Q。VTP 能跨越多种 LAN 类型进行 VLAN 的动态映射，每种 VLAN 都有唯一的名称和与之关联的内部索引。这种映射为管理员减少了大量的设备管理任务。

如果交换机配置为透明模式，用户可以创建或修改 VLAN，但所做的修改只会影响单个交换机，不会被发送到域中的其他设备上。但是，设备在这种模式下所做的配置更改会存到设备正在运行的配置中，并可保存到设备的启动配置文件中。

VTP 模式

表格 211：VTP 模式

VTP 模式	描述
VTP 服务器	在 VTP 服务器模式下，用户可以创建、修改和删除 VLAN，还可以为整个 VTP 域确定其他配置参数（例如 VTP 版本）。VTP 服务器向位于同一个 VTP 域中的其他设备通告 VLAN 配置，并且通过在中继链路上接收通告消息来实现与其他设备进行 VLAN 配置的同步。 VTP 服务器是默认模式。 在 VTP 服务器模式下，VLAN 的配置存储与 NVRAM。如果设备在将配置写入 NVRAM 时检测到故障，VTP 模式会自动从服务器模式更改为客户端模式。如果发生这种情况，除非 NVRAM 运行，否则设备无法返回 VTP 服务器模式
VTP 客户端	VTP 客户端与服务器模式的工作方式相似，通过中继传送、接收 VTP 更新，但是用户不能在 VTP 客户端上创建、修改和删除 VLAN。VLAN 在域中其他处于服务器模式的设备上进行配置。 在 VTP 客户端模式的 VTP 版本 1 和版本 2 中，NVRAM 并不保存 VLAN 的配置信息。在 VTP 版本 3 中，在客户端模式下 NVRAM 会保存 VLAN 的配置信息
VTP 透明	处于透明模式的设备不会加入 VTP 域中。一个 VTP 透明设备不会通告自己的 VLAN 配置，也不会根据接收到的通告信息同步自己的 VLAN 配置。但是在 VTP 版本 2 或版本 3 中，透明设备会从中继接口转发从其他设备接收到的 VTP 通告。用户可以在处于 VTP 透明模式的设备上创建、修改和删除 VLAN 配置。 当设备处于 VTP 透明模式，VTP 和 VLAN 配置存储于 NVRAM 中，不会被通告到其他设备上。在这种模式下，VTP 模式和域名存储于设备正在运行的配置中，并且用户可以使用 **copy running-config startup-config** 特权 EXEC 命令将这些信息存储到设备的启动配置文件中。 在一个设备堆栈中，对于堆栈中的所有设备，正在运行的配置和已经保存的配置没有区别
VTP 关闭	VTP 关闭模式与 VTP 透明模式的交换机工作方式相同，但它不会通过中继转发 VTP 通告

VTP 通告

在 VTP 域中的每个设备都会通过中继端口向保留的多播地址发送周期性的全局配置通告。相邻设备会接收通告并更新他们的 VTP 和 VLAN 配置。

VTP 通告发送以下全域信息：

- VTP 域名
- VTP 配置修订号
- 更新者身份和更新时间戳
- VLAN 配置的 MD5 摘要散列码，包括每个 VLAN 的最大传输单元（Maximum Transmission Unit，MTU）大小

- 帧格式

VTP 通告为每个配置好的 VLAN 发送以下 VLAN 信息：

- VLAN ID（包括 IEEE802.1Q）
- VLAN 名字
- VLAN 类型
- VLAN 状态
- 针对 VLAN 类型的附加 VLAN 配置信息

在 VTP 版本 3 中，VTP 通告也包括主服务器 ID、一个实例编号和一个开始索引。

VTP 版本 2

如果用户在网络中使用 VTP，必须决定使用 VTP 的哪个版本。VTP 默认运行版本 1。

与版本 1 相比，VTP 版本 2 提供了下列额外功能：

- 支持令牌环——VTP 版本 2 支持令牌环网桥中转功能（Token Ring Bridge Relay Function，TrBRF）和令牌环集中器中继功能（Token Ring Concentrator Relay Function，TrCRF）VLAN；
- 支持无法识别的类型长度值（Type-Length-Value，TLV）——VTP 服务器或客户端将配置变更消息传播到其他中继，即使对无法解析的 TLV 也是如此。当设备在 VTP 服务器模式下运行时，无法识别的 TLV 会保存在 NVRAM 中；
- 依赖于版本的透明模式——在 VTP 版本 1 中，VTP 透明设备会检查 VTP 消息的域名和版本，并仅在版本和域名都匹配时才能转发消息。尽管 VTP 版本 2 只支持一个域，但 VTP 版本 2 的透明设备仅在域名匹配时就能转发消息；
- 一致性检查——在 VTP 版本 2 中，只有当用户通过 CLI 或 SNMP 输入新信息的时候，才执行一致性检查（例如 VLAN 名和值）。对于从 VTP 消息和 NVRAM 中获取的信息不进行一致性检查。如果接收到的 VTP 消息的 MD5 摘要散列码是正确的，说明信息可靠。

VTP 版本 3

与版本 1 和版本 2 相比，VTP 版本 3 提供了下列额外功能：

- 认证加强——用户可以将认证配置为 **hidden** 或 **secret**。当处于 **hidden** 模式下，密码字符串中的密钥保存在 VLAN 数据库文件中，但在配置信息中不以纯文本显示。相反，与密码关联的密钥以十六进制格式保存在运行的配置中。如果用户在域中输入接管命令，则需要重新输入密码。当用户输入 **secret** 关键字时，可以直接配置密码的密钥；
- 支持扩展的 VLAN（VLAN 1006 到 4094）数据库传播——VTP 版本 1 和版本 2 只能传播 VLAN1 到 1005；

注释： VTP 修剪仍然只适用于 VLAN 1 到 1005，VLAN 1002 到 1005 仍然保留，不能修改。

- 支持域中的任意数据库——除了传播 VTP 信息，版本 3 可以传播多生成树（Multiple Spanning Tree，MST）协议数据库信息。为每个使用 VTP 的应用程序分别运行 VTP 协议的实例；
- VTP 主服务器和 VTP 辅助服务器——VTP 主服务器负责更新数据库信息，并发送满足系统中所有设备的更新。VTP 辅助服务器只能将从主服务器接收到的 VTP 配置更新备份到其 NVRAM 中。

  默认情况下，所有的设备都是辅助服务器。用户可以输入 **vtp primary** 特权 EXEC 命令指定一个主服务器。当管理员在域中发出接管消息时，主服务器的状态仅用于数据库更新。

VTP 域可以在没有主服务器的环境下工作。如果设备重新加载或设备的域参数被更改，即使在设备上配置了密码，主服务器的状态也会丢失；

- 基于每个中继（每端口）VTP 打开或关闭的选项——用户可以通过输入**[no] vtp** 接口配置命令来启用或禁用每个端口的 VTP。当用户在中继端口上禁用 VTP 时，将禁用该端口的所有 VTP 实例。用户不能将 MST 数据库的 VTP 设为*关闭*的同时将 VLAN 数据库设为*开启*。

  当用户将全局的 VTP 模式设为关闭时，会应用于系统中的所有中继端口。但是，用户可以指定某 VTP 是 on 或 off。例如，用户可以为了 VLAN 数据库将设备配置成 VTP 服务器，再将 MST 数据库的 VTP *关闭*。

VTP 修剪

VTP 修剪通过将泛洪数据流限制到流量到达目的设备必须经过的中继链路上，从而提高网络可用带宽。如果没有 VTP 修剪，设备会通过某 VTP 域中所有中继链路来泛洪广播、多播和未知单播，即便接收设备可能会丢弃接收到的帧。默认情况下 VTP 修剪被禁用。

VTP 修剪会在可修剪列表中的中继端口上阻止不必要的泛洪数据流流向 VLAN。只有在可修剪列表中的 VLAN 可以被修剪。默认情况下，VLAN2 到 1001 是可修剪设备的中继端口。如果 VLAN 被配置为可修剪，泛洪将继续进行。所有 VTP 版本都支持 VTP 修剪。

VTP 修剪在交换网络中被禁用。设备 A 上的端口 1 和设备 D 上的端口 2 分配给红色 VLAN。如果广播从连接到设备 A 的主机发送出来，设备 A 会洪泛广播，并且网络中的每个设备都会接收广播，即使设备 C，E 和 F 与红色 VLAN 没有直连端口。

图 137：未启用 VTP 修剪的流量泛洪

Switch x	交换机 x
Port x	端口 x
Red VLAN	红色 VLAN

VTP 修剪在交换网络中被启用。来自设备 A 的广播流量不会被转发到设备 C，E 和 F，因为红色 VLAN 的流量已在所示链路（设备 B 上的端口 5 和设备 D 上的端口 4）上被修剪。

图 138：启用 VTP 修剪的优化泛洪流量

Switch x	交换机 x
Port x	端口 x
Red VLAN	红色 VLAN
Flooded traffic is pruned	泛洪流量被修剪

对于 VTP 版本 1 和 2，当用户在 VTP 服务器上启用修剪时，整个 VTP 域将启用修剪。在 VTP 版本 3 中，用户必须在域中的每个设备上手动启用修剪。VLAN 的可修剪或不可修剪只影响在中继上的（不是在 VTP 域中的所有设备上）那些 VLAN 的可修剪性。

VTP 修剪在启用后几秒钟内生效。 VTP 修剪不会修剪那些不可修剪的 VLAN 流量。 VLAN 1 和 VLAN 1002 到 1005 是不可修剪的;来自这些 VLAN 的流量不能修剪。扩展范围的 VLAN（高于 1005 的 VLAN ID）也是不可修剪的。

VTP 和设备堆栈

设备堆栈中所有成员使用相同的 VTP 配置。当设备堆栈处于 VTP 服务器、客户端或透明模

式下，堆栈中的所有设备都携带相同 VTP 配置。

- 当设备加入堆栈后，它会从活跃交换机继承 VTP 和 VLAN 性质。
- 所有 VTP 更新都在堆栈中传送。
- 当堆栈中的设备更改 VTP 模式时，堆栈中的其他设备也会更改 VTP 模式，并且设备的 VLAN 数据库会保持一致。

VTP 版本 3 在独立设备或堆栈上具有相同功能，除非该设备堆栈是 VTP 数据库的主服务器。在这种情况下，活跃交换机的 MAC 地址将用作主服务器 ID。如果活跃设备正在重新加载或已关机，则选择一台新的活跃交换机。

- 如果不配置永久 MAC 地址，当选择新的活跃设备时，它将使用当前堆栈的 MAC 地址发送接管消息。

注释： 默认情况下，永久 MAC 地址已启用。

VTP 配置指南

VTP 配置需求

当用户需要配置 VTP 的时候，用户必须配置中继端口，以便设备可以从域中的其他设备发送和接收 VTP 通告。

VTP 设置

VTP 信息保存在 VTP VLAN 数据库中。当 VTP 模式为透明时，VTP 域名和模式也保存在设备正在运行的配置文件中，通过输入 **copy running-config startup-config** 特权 EXEC 命令，可以将其保存到设备启动配置文件中。如果要将 VTP 模式保存为透明，即使设备重置，用户也必须使用此命令。

当用户将 VTP 信息保存到设备启动配置文件中并重启设备时，设备配置选择如下：

- 如果启动配置中的 VTP 模式为透明，且 VLAN 数据库和 VLAN 数据库中的 VTP 域名与启动配置文件中的 VTP 域名匹配，则忽略（清除）VLAN 数据库，并使用启动配置文件中的 VTP 和 VLAN 配置。VLAN 数据库版本号在 VLAN 数据库中保持不变。
- 如果启动配置中的 VTP 模式或域名与 VLAN 数据库不匹配，那么 VLAN ID 1 到 1005 的域名、VTP 模式和配置将使用 VLAN 数据库中的信息。

配置 VTP 的域名

在初次配置 VTP 的时候，用户必须分配域名。用户需要为 VTP 域中的所有设备配置相同的域名。处于 VTP 透明模式的设备不与其他设备交换 VTP 信息，用户不需要为它们配置 VTP 域名。

注释： 如果 NVRAM 和 DRAM 有足够存储空间，则 VTP 域中的所有设备都应处于 VTP 服务器模式。

注意： 如果所有的设备都以 VTP 客户端模式运行，请不要配置 VTP 域。如果用户配置了 VTP 域，那么该域的 VLAN 配置将无法修改。请确保 VTP 域中至少有一台设备被配置为 VTP 服务器模式。

VTP 域的密码

用户可以为 VTP 域配置密码，但并不是必须配置。如果用户配置了域密码，那么所有设备都将使用相同的密码，并且用户需要在管理域中的每个设备上配置该密码。没有密码或密码错误的设备将拒绝 VTP 通告。

如果为域配置 VTP 密码，在使用正确的密码配置之前，未使用 VTP 配置引导的设备将不会接受 VTP 通告。配置之后，设备会接受通告中使用相同密码和域名的 VTP 通告。

如果用户要向具有 VTP 功能的网络添加新设备，只有在该设备上配置了适正确的密码之后，

新设备才会习得域名。

注意： 配置 VTP 域密码时，如果不为域中的每个设备分配管理域密码的话，管理域将无法正常工作。

VTP 版本

决定使用哪个 VTP 版本时，请遵循以下指导：

- VTP 域中的所有设备必须使用相同的域名，但它们不需要使用相同的版本。
- 如果一台 VTP 版本 2 可用（默认情况下版本 2 未启用）的设备禁用了版本 2，则该设备可以与运行 VTP 版本 1 的设备在同一个域中工作。
- 如果一台运行 VTP 版本 1 但也能运行 VTP 版本 2 的设备接收到了 VTP 版本 3 的通告，它将自动转换到版本 2。
- 如果运行 VTP 版本 3 的设备与运行 VTP 版本 1 的设备相连，则 VTP 版本 1 的设备将转换到 VTP 版本 2，VTP 版本 3 设备发送缩减版的 VTP 数据包，以便 VTP 版本 2 设备更新其数据库。
- 如果一台运行 VTP 版本 3 的设备有扩展 VLAN，则不能转换到版本 1 或版本 2。
- 除非同一个 VTP 域中的所有设备都能使用 VTP 版本 2，否则不要启用设备上的 VTP 版本 2。当用户在一台设备上启用了版本 2，域中所有能使用 VTP 版本 2 的设备都将启用版本 2。如果此时域中某台设备只有版本 1，该设备将不会与启用版本 2 的设备交换 VTP 信息。
- Inspur 建议将使用版本 1 和版本 2 的设备放置在网络的边缘，因为处于边缘的设备不需要转发 VTP 版本 3 的通告。
- 如果用户的网络环境支持令牌环网桥中转功能和令牌环集中器中继功能，用户必须启用 VTP 版本 2 或版本 3，以便令牌环 VLAN 交换能正常运行。要运行令牌环和令牌环网，请禁用 VTP 版本 2。
- VTP 版本 1 和版本 2 不为扩展的 VLAN（VLAN1006 到 4094）传播配置信息。用户必须手动配置这些设备。VTP 版本 3 支持扩展的 VLAN 数据库的传播。
- 当 VTP 版本 3 设备的中继端口接收到了来自 VTP 版本 2 设备的消息后，该设备会在那个中继上以 VTP 版本 2 的格式发送缩减版的 VLAN 数据库。VTP 版本 3 设备不会在中继上发送 VTP 版本 2 格式的数据包，除非它先在中继上收到了 VTP 版本 2 数据包。
- 当 VTP 版本 3 设备在中继端口上检测到了 VTP 版本 2 设备时，该设备将继续发送 VTP 版本 3 数据包，同时也发送 VTP 版本 2 数据包，使得两种类型的邻接设备在同一中继上共存。
- 一台 VTP 版本 3 设备不会接受来自 VTP 版本 2 或 VTP 版本 1 设备的配置信息。
- 两个 VTP 版本 3 区域只能通过 VTP 版本 1 或版本 2 区域以透明模式通信。
- 只能使用 VTP 版本 1 的设备不能与 VTP 版本 3 设备进行互操作。
- VTP 版本 1 和 VTP 版本 2 不能为扩展的 VLAN（VLAN1006 到 4094）传递配置信息。用户必须在每台设备上手动配置 VLAN。

如何配置 VTP

配置 VTP 模式（CLI）

用户可以将 VTP 模式配置为以下之一：

- VTP 服务器模式——在 VTP 服务器模式下，用户可以修改 VLAN 配置并将修改的配置信

息通过网络传播。

- VTP 客户端模式——在 VTP 客户端模式下，用户不能修改 VLAN 配置。客户端设备接收来自 VTP 域中的 VTP 服务器的 VTP 更新，根据接收到的更新信息修改配置。
- VTP 透明模式——在 VTP 透明模式下，设备上的 VTP 被禁用。设备不发送 VTP 更新，如果收到来自其他设备的 VTP 更新也不修改配置。但是，一台运行 VTP 版本 2 的 VTP 透明设备会将中继链路上收到的 VTP 通告转发出去。
- VTP 关闭模式——VTP 关闭模式与 VTP 透明模式除了不在中继上转发 VTP 通告这点，其余运行效果一样。

当用户配置域名的时候，域名不能被移除；用户只能将设备分配到其他域中。

总步骤

1. **enable**
2. **configure terminal**
3. **vtp domain** *domain-name*
4. **vtp mode{client| server| transparent| off}{vlan| mst| unknown}**
5. **vtp password** *password*
6. **end**
7. **show vtp status**
8. **copy running-config startup-config**

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式。如果出现提示，请输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**vtp domain** *domain-name* **示例：** `Device(config)#` **vtp domain eng_group**	配置 VTP 管理域名。名称可以是 1 到 32 个字符。具有相同管理责任的所有以 VTP 服务器或客户端模式运行的设备必须配置相同的域名。 对于服务器模式以外的模式，此命令是可选项。VTP 服务器模式需要一个域名。如果设备具有到 VTP 域的中继连接，则设备从域中的 VTP 服务器获取域名。 用户应该在配置其他 VTP 参数之前配置 VTP 域。
步骤 4	**vtp mode {client | server | transparent |off} {vlan | mst | unknown}** **示例：** `Device(config)#` **vtp mode server**	为设备配置 VTP 模式（客户端、服务器、透明或关闭） • **vlan**——VLAN 数据库为默认配置。 • **mst**——多生成树数据库 • **unknown**——未知类型数据库
步骤 5	**vtp password** *password* **示例：**	（可选）设置 VTP 域的密码。密码可以是 8 到 64 个字符。如果配置了 VTP 密

	Device(config)# **vtp password mypassword**	码，若用户没有为域中的每个设备分配相同的密码，VTP 域无法正常工作。
步骤 6	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式。
步骤 7	**show vtp status** **示例：** Device# **show vtp status**	请在显示的 *VTP 操作模式*和 *VTP 域名*字段中确认配置的条目。
步骤 8	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）在启动配置文件中保存配置信息。 只有 VTP 模式和域名可以保存在设备正在运行的配置中，并可被复制到启动配置文件中。

配置 VTP 版本 3 的密码（CLI）

用户可以在设备上配置 VTP 版本 3 的密码。

总步骤

1. enable

2. configure terminal

3. vtp password *password***[hidden | secret]**

4. end

5. show vtp password

6. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** Device>**enable**	启用特权 EXEC 模式。如果出现提示，请输入密码。
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式。
步骤 3	**vtp domain** *password***[hidden| secret]** **示例：** Device(config)# **vtp password mypasswordhidden**	(可选)为 VTP 域设置密码。密码可以是 8 到 64 个字符。 • （可选）**hidden**——保存根据 nvram:valn.dat 文件中的密码串生成的秘钥。如果用户通过配置 VTP 主服务器配置接管，系统将提示重新输入密码。 • （可选）**secret**——直接配置密码。密码必须包含 32 个十六进制字符。
步骤 4	**end** **示例：**	返回特权 EXEC 模式。

	Device(config)# **end**	
步骤 5	**show vtp password** **示例：** Device# **show vtp password**	请确认输入。输出如下： VTP 密码: 89914640C8D90868B6A0D8103847A733
步骤 6	**copy running-config startup-config** **示例：** Device# **copy running-configstartup-config**	（可选）在配置文件中保存配置的条目。

配置 VTP 版本 3 的主服务器（CLI）

当用户将一台 VTP 服务器配置为主服务器时，接管操作开始。

总步骤

1. vtp primary [vlan | mst] [force]

具体步骤

	命令或操作	目的
步骤 1	**vtp primary [vlan | mst] [force]** **示例：** Device# **vtp primary vlanforce**	将设备的操作状态从辅助服务器（默认）更改为主服务器，并将配置发布到域。如果设备密码配置为 **hidden**，系统将提示用户重新输入密码。 • （可选）**vlan**——选择 VLAN 数据库为接管特性。这是默认选项。 • （可选）**mst**——选择多生成树数据库作为接管特性。 • （可选）**force**——覆盖任何有冲突的服务器配置。如果用户不输入 **force**，系统将在接管前提示确认。

启用 VTP 版本（CLI）

VTP 版本 2 和版本 3 在默认情况下被禁用。

- 当用户在一台设备上启用了版本 2，域中所有能使用 VTP 版本 2 的设备都将启用版本 2。若要启动 VTP 版本 3，用户必须在每台设备上手动配置。
- 在使用 VTP 版本 1 和版本 2 的情况下，用户只能在 VTP 服务器或透明模式下的设备上配置版本。如果设备运行的是 VTP 版本 3，则当设备处于客户端模式时，若没有扩展 VLAN，并且未配置隐藏密码，可以将版本更改为版本 2。

注意： 在同一 VTP 域中的 VTP 版本 1 和 VTP 版本 2 设备不可进行互操作。除非 VTP 域中的每个设备都支持版本 2，否则不要启用 VTP 版本 2。

- 在 TrCRF 和 TrBRF 令牌环网络环境下，用户必须启用 VTP 版本 2 或版本 3，以便令牌环 VLAN 交换能正常运行。要运行令牌环和令牌环网，请禁用 VTP 版本 2。

注意： 在 VTP 版本 3 中，主服务器和辅助服务器都可以在域中实例上运行。

总步骤

1. enable

2. configure terminal

3. vtp version {1 | 2 | 3}

4. end

5. show vtp status

6. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式。如果出现提示，请输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**vtp version {1 | 2 | 3}** **示例：** `Device(config)#` **vtp version 2**	在设备上启用 VTP 版本。默认为 VTP 版本 1。
步骤 4	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**show vtp status** **示例：** `Device#` **show vtp status**	请确认配置的 VTP 版本已经启用。
步骤 6	**copy running-config startup-config** **示例：** `Device#` **copy running-configstartup-config**	（可选）在配置文件中保存配置的条目。

启用 VTP 修剪（CLI）

在开始前

VTP 修剪不是为在 VTP 透明模式下工作而设计的。如果网络中的一个或多个设备处于 VTP 透明模式，则应执行以下操作之一：

- 关闭整个网络的 VTP 修剪。
- 通过使设备中继上的所有 VLAN 到其上行的 VTP 透明设备都不可修剪，从而关闭 VTP 修剪。

在接口上配置 VTP 修剪，请使用 **switchport trunk pruning vlan** 接口配置命令。VTP 修剪在接口中继时运行。用户可以设置 VLAN 的可修剪性，是否为 VTP 域启用 VTP 修剪，是否存在任何给定的 VLAN，以及接口是否正在中继。

总步骤

1. enable

2. configure terminal

3. vtp pruning

4. end

5. show vtp status

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式。如果出现提示，请输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**vtp pruning** **示例：** `Device(config)#` **vtp pruning**	在 VTP 管理域启用修剪。 默认情况下，修剪被禁用。用户仅需要为一台 VTP 服务器模式下的设备启用修剪。
步骤 4	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 5	**show vtp status** **示例：** `Device#` **show vtp status**	请在显示的 *VTP 修剪模式*字段中验证配置的条目。

基于端口配置 VTP（CLI）

在 VTP 版本 3 下，用户可以在每个端口上启用或禁用 VTP。用户只能在处于中继模式的端口上启用 VTP。入向和出向 VTP 流量被阻止，不会被转发。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. vtp

5. end

6. show running-config interface *interface-id*

7. show vtp status

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	启用特权 EXEC 模式。如果出现提示，请输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。

步骤 3	**interface** *interface-id* **示例：** `Device(config)#` **interface gigabitethernet1/0/1**	标识接口，并进入接口配置模式。
步骤 4	**vtp** **示例：** `Device(config)#` **vtp**	在指定端口启用 VTP。
步骤 5	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 6	**show running-config interface** *interface-id* **示例：** `Device#` **show running-config interfacegigabitethernet1/0/1**	确认对端口的修改。
步骤 7	**show vtp status** **示例：** `Device#` **show vtp status**	确认配置。

向 VTP 域添加 VTP 客户端（CLI）

在将设备添加到 VTP 域*之前*，请按照以下步骤确认并重置设备上的 VTP 配置修订号。

在开始前

在将 VTP 客户端添加到 VTP 域之前，请确认其 VTP 配置修订号*低于* VTP 域中其他设备的配置修订号。VTP 域中的设备会使用 VTP 配置修订号最高设备的 VLAN 配置。在 VTP 版本 1 和版本 2 中，如果用户添加的设备修订号高于 VTP 域中的修订号，那么该设备可以清除 VTP 服务器和 VTP 域中所有的 VLAN 信息。在 VTP 版本 3 中，VLAN 信息不会被擦除。

用户可以在不影响 VTP 域中的其他设备的情况下，使用 **vtp mode transparent** 全局配置命令禁用设备上的 VTP，然后更改其 VLAN 信息。

总步骤

1. enable
2. show vtp status
3. configure terminal
4. vtp domain domain-name
5. end
6. show vtp status
7. configure terminal
8. vtp domain domain-name
9. end
10. show vtp status

具体步骤

	命令或操作	目的
步骤 1	**enable**	启用特权 EXEC 模式。如果出现提示，请

	示例： `Device>`**enable**	输入密码。
步骤 2	**show vtp status** **示例：** `Device#` **show vtp status**	检查 VTP 配置修订号。 如果修订号为 0，向 VTP 域添加设备。 如果修订号比 0 大，按下面步骤操作： • 记录下域名。 • 记录下配置修订号。 • 继续下一步骤来重设设备配置修订号。
步骤 3	**configure terminal** **示例：** `Device#`**configure terminal**	进入全局配置模式。
步骤 4	**vtp domain** *domain-name* **示例：** `Device(config)#` **vtp domain domain123**	将步骤一中显示的原始域名修改为新域名。
步骤 5	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。设备上的 VLAN 信息被更新，配置修订号被重置为 0。
步骤 6	**show vtp status** **示例：** `Device#` **show vtp status**	确认配置修订号已被重置为 0。
步骤 7	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 8	**vtp domain** *domain-name* **示例：** `Device(config)#` **vtp domain domain012**	在设备上输入原始域名。
步骤 9	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。设备上的 VLAN 信息被更新。
步骤 10	**show vtp status** **示例：** `Device#` **show vtp status**	（可选）请确认现在的域名与步骤一时的域名相同，以及配置修订号为 0。

监控 VTP

这部分描述了用于显示和监控 VTP 配置的命令。

用户通过查看 VTP 配置信息来监视 VTP：域名、当前 VTP 版本和 VLAN 个数。用户还可以查看设备发送和接收的通告的统计信息。

表格 212：VTP 监控命令

命令	目的

show vtp counters	显示已发送和已接收 VTP 消息的数量。
show vtp devices[conflict]	显示域中所有有关 VTP 版本 3 设备的信息。冲突是指与主服务器冲突的 VTP 版本 3 设备。在设备处于透明或关闭模式时，**show vtp devices** 命令不显示设备的信息。
show vtp interface[*interface-id***]**	显示所有接口或特定接口的 VTP 状态和配置。
show vtp password	显示 VTP 密码。密码显示的形式依赖于用户是否输入关键字 **hidden** 以及设备上是否启用加密技术。
show vtp status	显示 VTP 设备配置信息。

VTP 配置示例

示例：将交换机配置为主服务器

以下示例表明在配置隐藏或秘密密码时，如何将设备配置为 VLAN 数据库的主服务器（默认）：

```
Device# vtp primary vlan
Enter VTP password: mypassword
This switch is becoming Primary server for vlan feature in the VTP domain
VTP Database Conf Switch ID Primary Server Revision System Name
------------ ---- -------------- -------------- -------- --------------------
VLANDB Yes 00d0.00b8.1400=00d0.00b8.1400 1 stp7
Do you want to continue (y/n) [n]? y
```

接下来做什么？

VTP 配置完成后，用户可以配置以下事项：

- VLAN
- VLAN 群组
- VLAN 中继
- 语音 VLAN

其他参考文档

相关文档

相关主题	文档标题
有关本章中使用命令的完整语法和使用信息。	*VLAN 命令参考（Inspur6650 交换机）* *二/三层命令参考（Inspur6650 交换机）*
附加配置命令和步骤	*LAN 交换配置指南，Inspur INOS（Inspur6650 交换机）* *二/三层配置指南（Inspur6650 交换机）*

错误信息解释

描述	链接

为了便于用户研究和解决此版本中的系统错误消息，可以使用错误消息解码器工具。	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
RFC 1573	MIB-II 的接口组的演变
RFC 1757	远程网络监控管理
RFC 2021	使用 SMIv2 的传输控制协议的 SNMPv2 管理信息库

技术助手

描述	链接
Inspur 支持网站提供了广泛的在线资源，包括用于故障排除以及解决 Inspur 产品问题和技术问题的文档和工具。 要获得有关产品的安全和技术信息，用户可以订阅不同的服务，例如产品警报工具（从现场记录获取），Inspur 技术服务实时通讯和简单讯息聚合订阅（RSS）。 要获取 Inspur 支持网站上的大多数工具，需要有 icntnetworks.com 的用户 ID 和密码。	http://www.icntnetworks.com

VTP 的历史特性和信息

版本	修订
Inspur INOS 12.2	此特性被引入。

配置 VLAN

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问

http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

VLAN 的前提

以下是配置 VLAN 的前提和注意事项：

- 在创建 VLAN 之前，用户必须决定是否使用 VLAN 中继协议（VLAN Trunking Protocol，VTP）来维护网络的全局 VLAN 配置。
- 如果用户要在设备上配置多个 VLAN 并启用路由功能，可以将交换机数据库管理（Switch Database Management，SDM）功能设置为 VLAN 模板，该模板可配置系统资源以便支持最大单播 MAC 地址数。
- 运行 LAN Base 特性集的设备仅支持 SVI 上的静态路由。
- 设备中应有一个 VLAN，以便能够将其添加到 VLAN 组。

VLAN 的限制

以下是 VLAN 的限制：

- 设备支持每 VLAN 生成树加（per-VLAN spanning-tree plus，PVST+）或快速 PVST+最多包含 128 个生成树实例。每个 VLAN 允许有一个生成树实例。
- 设备支持 IEEE 802.1Q 中继方法，用于通过以太网端口发送 VLAN 流量。
- 不支持配置接口 VLAN 路由器的 MAC 地址。接口 VLAN 已有默认分配的 MAC 地址。
- 设备不支持私有 VLAN。
- 用户不能在交换机堆栈中混用 Inspur3850 和 Inspur6650 交换机。

关于 VLAN 的信息

逻辑网络

VLAN 是根据功能、项目团队或应用程序逻辑划分的交换网络，无需考虑用户的物理位置。VLAN 具有与物理 LAN 相同的属性，但可将物理上位于不同 LAN 段的终端设备分成一组。任何设备端口都可以属于 VLAN，并且单播、广播和组播数据包只能转发和泛洪到 VLAN 中的终端设备上。每个 VLAN 被认为是一个逻辑网络，并且目的地不属于 VLAN 的数据包必须通过路由器或支持回退桥接的设备转发。在设备堆栈中，VLAN 可以由不同堆栈的端口形成。由于 VLAN 被认为是一个单独的逻辑网络，所以它包含自己的网桥管理信息库（Management Information Base，MIB）信息，并且可以支持实现自己的生成树。

图 139：VLAN 作为逻辑定义网络

Cisco router	思科路由器
Gigabit Ethernet	千兆以太网
Engineering VLAN	工程部 VLAN
Marketing VLAN	市场部 VLAN
Accounting VLAN	会计部 VLAN

Floor x	层 x

VLAN 通常与 IP 子网关联。例如，特定 IP 子网中的所有终端设备属于同一 VLAN。设备上的接口 VLAN 成员资格是基于逐个接口手动分配的。使用此方法将设备接口分配给 VLAN 时，称为基于接口或静态的 VLAN 成员资格。

VLAN 之间的流量必须被路由。

设备可以通过使用设备虚拟接口（Switch Virtual Interface，SVI）在 VLAN 之间路由流量。必须显式配置 SVI 并为其分配一个 IP 地址，以便 SVI 在 VLAN 之间路由流量。

VLAN 支持

设备支持 VTP 客户端，服务器和透明模式下的 VLAN。VLAN 由 1 到 4094 之间的数字标识。VLAN 1 是默认 VLAN，是在系统初始化期间创建的。VLAN ID 1002 到 1005 被保留用于令牌环和 FDDI VLAN。除了 1002 到 1005 之外的所有 VLAN 都可用于用户配置。

有 3 个 VTP 版本：VTP 版本 1，版本 2 和版本 3。所有的 VTP 版本都支持正常的和扩展范围的 VLAN，但只有在 VTP 版本 3 中设备才支持传播扩展范围的 VLAN 配置信息。当在 VTP 版本 1 和 2 中创建扩展范围的 VLAN 时，不会传播其配置信息。即使设备上的本地 VTP 数据库记录不更新，但扩展范围的 VLAN 配置信息也会创建并存储在设备正在运行的配置文件中。用户可以在设备上配置多达 4094 个 VLAN。

VLAN 端口成员资格模式

用户通过分配成员资格模式来配置端口属于某 VLAN，该成员资格模式指定端口承载的流量类型以及它可以被多少个 VLAN 使用。

当一个端口属于某 VLAN 时，该设备将基于每个 VLAN 了解和管理与该端口关联的地址。

表格 213：端口成员资格模式和特性

成员资格模式	VLAN 成员资格特性	VTP 特性
静态接入	一个静态接入端口可以属于一个 VLAN，并且可以手动分配给该 VLAN。	VTP 配置不是必需的。如果用户不希望 VTP 全局传播信息，请将 VTP 模式设置为透明。要参与 VTP，设备或设备堆栈上必须至少有一个中继端口连接到第二个设备或设备堆栈的中继端口上。
中继（IEEE802.1Q）： • IEEE802.1Q——行业标准的中继封装。	中继端口是所有 VLAN 的默认成员，包括扩展范围的 VLAN，但是成员资格可受限于允许传输的 VLAN 列表的配置，用户还可以修改可修剪列表，阻止在中继端口上把流量泛洪到列表中的 VLAN。	建议使用 VTP，但不是必需的。VTP 通过在全网的基础上管理 VLAN 的添加、删除和重命名，从而维护 VLAN 配置的一致性。VTP 与其他设备通过中继链路交换 VLAN 配置信息。
语音 VLAN	语音 VLAN 端口是连接到 Inspur IP 电话的接入端口，	VTP 配置不是必需的；VTP 对语音 VLAN 没有影响。

	对于连接到电话的设备，使用一个 VLAN 用于传输语音流量，另一个 VLAN 用于传输数据流量。	

VLAN 配置文件

VLAN ID 1 到 1005 的配置写在 vlan.dat 文件（VLAN 数据库），用户可以通过输入 **show vlan** 特权 EXEC 命令查看配置信息。vlan.dat 文件存储在闪存中。如果 VTP 模式为透明，文件也将保存到设备正在运行的配置文件中。

在设备堆栈中，整个堆栈使用相同的 vlan.dat 文件和运行配置。在某些设备上，vlan.dat 文件存储在活跃设备的闪存中。

用户可以使用接口配置模式定义端口成员资格模式，以及从 VLAN 添加和删除端口。这些命令的结果将写入设备正在运行的配置文件中，用户可以通过输入 **show running-config** 特权 EXEC 命令查看该文件。

当用户将 VLAN 和 VTP 信息（包括扩展范围的 VLAN 配置信息）保存到设备启动配置文件中并重启设备时，设备的配置选择如下：

- 如果启动配置中的 VTP 处于透明模式，且 VLAN 数据库和 VLAN 数据库中的 VTP 域名与启动配置文件中的 VTP 域名匹配，则忽略（清除）VLAN 数据库，并使用启动配置文件中的 VTP 和 VLAN 配置。VLAN 数据库版本号在 VLAN 数据库中保持不变。
- 如果启动配置中的 VTP 模式或域名与 VLAN 数据库不匹配，那么 VLAN ID 1 到 1005 的域名、VTP 模式和配置将使用 VLAN 数据库中的信息。
- 在 VTP 版本 1 和 2 中，如果 VTP 是服务器模式，VLAN ID 1 至 1005 的域名和 VLAN 配置将使用 VLAN 数据库信息。VTP 版本 3 还支持 VLAN 1006 到 4094。
- 从 15.0（02）SE6 镜像开始，在 VTP 透明和关闭模式下，即使 VLAN 不应用于接口，它们仍从 startup-config 创建。

正常范围 VLAN 配置指南

正常范围的 VLAN 指 ID 从 1 到 1005 的 VLAN。

在网络中创建和修改正常范围 VLAN 时，请参考以下指南：

- 正常范围的 VLAN 标识编号介于 1 和 1001 之间。VLAN 编号 1002 到 1005 为令牌环和 FDDI VLAN 保留。
- VLAN ID 1 到 1005 的 VLAN 配置一般都保存到 VLAN 数据库中。如果 VTP 模式为透明，VTP 何 VLAN 配置也将保存到设备正在运行的配置文件中。
- 如果设备处于 VTP 服务器或 VTP 透明模式，用户可以添加、修改或移除 VLAN 数据库中 VLAN 2 到 1001 的配置信息（VLAN ID 为 1 和 1002 到 1005 之间的配置信息是自动创建的，无法删除）。
- 在 VTP 透明模式下创建的扩展范围 VLAN 不会保存于 VLAN 数据库中，也不会传播。VTP 版本 3 支持在 VTP 服务器模式下的扩展范围 VLAN（VLAN 1006 到 4094）的数据库传播。
- 在用户能创建 VLAN 之前，设备必须处于 VTP 服务器模式或 VTP 透明模式。如果设备是一个 VTP 服务器，用户必须定义 VTP 域，否则 VTP 将无法运行。

- 设备不支持令牌环或 FDDI。设备不转发 FDDI，FDDI-Net，TrCRF 或 TrBRF 流量，但它通过 VTP 传播 VLAN 配置。
- 设备支持 128 个生成树实例。如果设备具有比支持的生成树实例更多的活跃 VLAN，那么可以在 128 个 VLAN 上启用生成树，并在其余 VLAN 上禁用生成树。如果用户已经在设备上使用了所有可用的生成树实例，那么在 VTP 域中的任何位置添加另一个 VLAN 会在该设备上创建一个未运行生成树的 VLAN。如果用户在该设备的中继端口上具有默认允许列表（允许所有 VLAN），则所有中继端口上都具有 VLAN。根据网络的拓扑情况，新 VLAN 中可能会创建出一个不会断开的环路，特别是在有几个相邻设备都已用完生成树实例的情况下。用户可以通过在已耗尽其对生成树实例的分配的设备的中继端口上设置允许列表来防止这种情况发生。

  如果设备上的 VLAN 数量超过支持的生成树实例数，我们建议用户在设备上配置 IEEE 802.1s Multiple STP（MSTP），便于将多个 VLAN 映射到单个生成树实例上。
- 当堆栈中的设备习得一个新 VLAN，或删除、修改现有 VLAN（通过网络端口上的 VTP 或通过 CLI）时，VLAN 信息将通告给所有堆栈成员。
- 当一台设备加入堆栈或堆栈进行合并时，新设备上的 VTP 信息（vlan.dat 文件）将与活跃设备保持一致。

扩展范围 VLAN 的配置指南

扩展范围的 VLAN 指 VLAN ID 编号从 1006 到 4094.
在创建扩展范围的 VLAN 时，请参考以下指南：

- 扩展范围的 VLAN ID 不保存在 VLAN 数据库中，VTP 无法识别，除非设备使用 VTP 版本 3。
- 用户不能将扩展范围的 VLAN 包含进可修剪范围。
- 对于 VTP 版本 1 或 2，用户可以在全局配置模式下将 VTP 模式设置为透明。用户应该将此配置保存到启动配置，以便设备以 VTP 透明模式启动。否则，一旦设备重置，扩展范围 VLAN 的配置将丢失。如果在 VTP 版本 3 中创建扩展范围的 VLAN，它们将无法转换到 VTP 版本 1 或 2。
- 在设备堆栈中，整个堆栈使用相同的运行配置和保存的配置，并且在堆栈中共享扩展范围 VLAN 的信息。

如何配置 VLAN

如何配置正常范围 VLAN

在 VLAN 数据库中创建一个新的正常范围 VLAN 或修改一个现有 VLAN 时，可以设置以下参数：

- VLAN ID
- VLAN 名
- VLAN 类型
  - 以太网
  - 光纤分布式数据接口（Fiber Distributed Data Interface，FDDI）

- FDDI 网络实体名称（Network Entity Title，NET）
- 令牌环网桥中转功能或令牌环集中器中继功能
- 令牌环
- 令牌环网

- VLAN 状态（活跃或暂停）
- VLAN 的最大传输单元（Maximum Transmission Unit，MTU）
- 安全性关联标识符（Security Association Identifier，SAID）
- TrBRF VLAN 的网桥标识符
- FDDI 和 TrCRF VLAN 的环号
- TrCRF VLAN 的父 VLAN 号
- TrCRF VLAN 的生成树协议（Spanning Tree Protocol，STP）类型
- 从一个 VLAN 类型转换到另一个 VLAN 类型时使用的 VLAN 号

如果用户尝试手动删除 vlan.dat 文件，可能会导致 VLAN 数据库不一致。如果要修改 VLAN 配置，请按照本节中的步骤操作。

创建或修改一个以太网 VLAN（CLI）

在开始前

对于 VTP 版本 1 和 2，如果设备处于 VTP 透明模式，可以分配大于 1006 的 VLAN ID，但它们将不会被添加到 VLAN 数据库。

设备仅支持以太网接口。由于不支持本地 FDDI 和令牌环 VLAN，因此用户仅为其他设备配置用于 VTP 全局通告的 FDDI 和令牌环特定介质的特性。

虽然设备不支持令牌环连接，但是具有令牌环连接的远程设备可以通过支持令牌环连接的设备进行管理。运行 VTP 版本 2 的设备会通告关于这些令牌环 VLAN 的信息：

- 令牌环 TrBRF VLAN
- 令牌环 TrCRF VLAN

总步骤

1. configure terminal

2. vlan *vlan-id*

3. name *vlan-name*

4. media { ethernet | fd-net | fddi | tokenring | trn-net }

5. remote-span

6. end

7. show vlan {name*vlan-name***| id** *vlan-id***}**

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 2	**vlan** *vlan-id* **示例：** `Device(config)# vlan 20`	输入 VLAN ID 和 VLAN 配置模式。输入一个新的 VLAN ID 来创建 VLAN，或输入现有 VLAN ID 来修改该 VLAN。 **注释：** 这个命令可用的 VLAN ID 范围从 1 到 4094。
步骤 3	**name***vlan-name* **示例：**	（可选）为 VLAN 输入一个名称。如果没有为 VLAN 输入名称，默认名是将带有前

	Device(config-vlan)# **name test20**	导零的 *vlan-id* 值附加到单词 VLAN 后。例如，VLAN0004 是 VLAN 4 的默认 VLAN 名称。
步骤 4	**media{ethernet| fd-net| fddi| tokenring| trn-net}** **示例：** Device(config-vlan)# **media ethernet**	配置 VLAN 介质类型。可选命令有： • ethernet——将 VLAN 介质类型设为以太网。 • fd-net——将 VLAN 介质类型设为 FDDI 网。 • fddi——将 VLAN 介质类型设为 FDDI。 • tokenring——将 VLAN 介质类型设为令牌环。 • trn-net——将 VLAN 介质类型设为令牌环网。
步骤 5	**remote-span** **示例：** Device(config-vlan)# **remote-span**	（可选）为远程 SPAN 会话将 VLAN 配置为 RSPAN VLAN。有关远程 SPAN 的详细信息，请参阅 *Inspur 6650 网络管理配置指南*。
步骤 6	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式。
步骤 7	**show vlan{ name** *vlan-name***| id** *vlan-id***}** **示例：** Device# **show vlan name test20 id 20**	验证配置的条目。

删除 VLAN（CLI）

当用户从处于 VTP 服务器模式的设备中删除 VLAN 时，VTP 域中所有设备的 VLAN 数据库都会删除该 VLAN。若从处于 VTP 透明模式的设备中删除 VLAN 时，仅在特定设备或设备堆栈上删除该 VLAN。

用户不能删除不同介质类型的默认 VLAN：以太网 VLAN 1 和 FDDI 或令牌环 VLAN 1002 至 1005。

注意： 用户删除 VLAN 时，分配给该 VLAN 的端口都将变为非活跃状态。除非用户将它们分配给新的 VLAN，否则将一直保持与 VLAN 的关联（因此处于非活跃状态）。

总步骤

1. enable

2. configure terminal

3. no vlan *vlan-id*

4. end

5. show vlan brief

6. copy running-config startup-config

详细步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：**	进入特权 EXEC 模式。在提示时输入密码。

	Device>**enable**	
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式。
步骤 3	**no vlan** *vlan-id* **示例：** Device(config)# **no vlan 4**	通过输入 VLAN ID 来删除 VLAN。
步骤 4	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式。
步骤 5	**show vlan brief** **示例：** Device# **show vlan brief**	验证删除的 VLAN。
步骤 6	**copy running-config startup-config** **示例：** Device# **copy running-config startup-config**	（可选）在配置文件中保存用户配置条目。

为 VLAN 分配静态接入端口（CLI）

用户可以为一个通过禁用 VTP（处于 VTP 透明模式）阻止 VLAN 配置信息全局传播的 VLAN 分配一个静态接入端口。

如果用户想把集群成员设备上的端口分配给 VLAN，请先使用 **rcommand** 特权 EXEC 命令登录到集群成员交换机上。

如果用户将接口分配给不存在的 VLAN 时，新 VLAN 会被创建。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport mode access

5. switchport access vlan *vlan-id*

6. end

7. show running-config interface *interface-id*

8. show interfaces*interface-id***switchport**

9. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例：** Device>**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例：** Device(config)# **interface**	输入要添加到 VLAN 的接口。

	gigabitethernet2/0/1	
步骤 4	**switchport mode access** **示例：** `Device(config-if)# switchport mode access`	为端口（二层接入端口）定义 VLAN 成员资格模式。
步骤 5	**switchport access vlan** *vlan-id* **示例：** `Device(config-if)#` **switchport access vlan 2**	为 VLAN 分配端口。有效的 VLAN ID 是从 1 到 4094。
步骤 6	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 7	**show running-config interface** *interface-id* **示例：** `Device#` **show running-config interfacegigabitethernet2/0/1**	验证接口的 VLAN 成员资格模式。
步骤 8	**show interfaces** *interface-id* **switchport** **示例：** `Device#` **show interfaces gigabitethernet2/0/1switchport**	在显示的 *Administrative Mode* 和 *Access Mode VLAN* 字段中验证用户配置条目。
步骤 9	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）在配置文件中保存用户配置条目。

如何配置扩展 VLAN

扩展范围的 VLAN 使得服务提供商能为更多客户提供他们的基础设施。只要 **switchport** 命令能用 VLAN ID，该命令就能用扩展范围的 VLAN ID。

对于 VTP 版本 1 或 2，扩展范围 VLAN 的配置不在 VLAN 数据库中存储，但由于 VTP 模式为透明，所以配置信息存储于设备正在运行的配置文件中，用户还可以将配置保存到启动配置文件中。在 VTP 版本 3 中创建的扩展范围的 VLAN 将存储在 VLAN 数据库中。

用户只能更改扩展范围 VLAN 上的 MTU 大小和远程 SPAN 配置状态; 所有其他特性必须保持默认状态。

创建扩展范围的 VLAN（CLI）

总步骤

1. enable

2. configure terminal

3. vlan *vlan-id*

4. remote-span

5. exit

6. end

7. show vlan id *vlan-id*

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**vlan** *vlan-id* **示例：** `Device(config)#` **vlan 2000** `Device(config-vlan)#`	输入扩展范围的 VLAN ID 并输入 VLAN 配置模式。范围从 1006 到 4049。
步骤 4	**remote-span** **示例：** `Device(config-if)# remote-span`	（可选）将 VLAN 配置为 RSPAN VLAN。
步骤 5	**exit** **示例：** `Device(config-vlan)#` **exit** `Device(config)#`	返回配置模式。
步骤 6	**end** **示例：** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 7	**show vlan id** *vlan-id* **示例：** `Device#` **show vlan id 2000**	验证 VLAN 已被创建。
步骤 8	**copy running-config startup-config** **示例：** `Device#` **copy running-config startup-config**	（可选）在配置文件中保存用户配置条目。

监控 VLAN

表格 214：特权 EXEC show 命令

命令	目的
show interfaces [vlan *vlan-id***]**	显示所有接口或设备上配置的指定 VLAN 的特性。
show vlan [access-map *name* **| brief | dot1q { tag native } | filter [access-map | vlan] | group [group-name** *name* **] | id** *vlan-id* **| ifindex | mtu | name** *name* **| remote-span |**	显示所有 VLAN 或设备上指定 VLAN 的参数。可选命令如下： • **access-map**——显示 VLAN 接入映射。 • **brief**——简短显示 VTP VLAN 状态。

summary]	• **dol1q**——显示 dot1q 参数。 • **filter**——显示 VLAN 过滤信息。 • **group**——显示 VLAN 组的名称和已连接的可用 VLAN。 • **id**——通过标识编号来显示 VTP VLAN 的状态。 • **ifindex**——显示 SNMP 的 ifIndex 信息。 • **mtu**——显示 VLAN MTU 信息。 • **name**——通过指定的名称显示 VTP VLAN 信息。 • **remote-span**——显示远程 SPAN VLAN。 • **summary**——显示 VLAN 信息的总结。

接下来做什么？

在配置 VLAN 之后，用户可以配置以下内容：

- VLAN 组
- VLAN 中继协议（VLAN Trunking Protocol，VTP）
- VLAN 中继
- 语音 VLAN

其他参考资料

相关文档

相关主题	文档标题
有关本章中使用命令的完整语法和使用信息。	*VLAN 命令参考（Inspur6650 交换机）* *二/三层命令参考（Inspur6650 交换机）*
VLAN 接入映射	*安全配置指南（Inspur6650 交换机）* *安全命令参考资源（Inspur6650 交换机）*
VLAN 和移动代理	*移动配置指南，Inspur INOS（Inspur6650 交换机）*
Inspur 灵活网络流量	*Inspur 灵活网络流量配置，Inspur INOS（Inspur 6650 交换机）*
IGMP 侦听	*IP 多播路由命令参考资源（Inspur 6650 交换机）* *IP 多播路由配置指南（Inspur 6650 交换机）*
IPv6	*IPv6 配置指南（Inspur 6650 交换机）* *IPv6 命令参考资源（Inspur 6650 交换机）*
SPAN	*网络管理命令参考资源（Inspur 6650 交换机）* *网络管理配置指南（Inspur 6650 交换机）*
独立平台的配置信息	*基于身份的网络服务配置指南，Inspur INOS（Inspur 6650 交换机）*

错误信息解释

描述	链接

为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题
RFC 1573	MIB-II 的接口组的演变
RFC 1757	远程网络监控管理
RFC 2021	使用 SMIv2 的传输控制协议的 SNMPv2 管理信息库

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

VTP 的历史特性和信息

版本	修订
Inspur INOS 12.2	支持 VLAN GUI

配置 VLAN 组

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

VLAN 组的前提

- 设备中应有一个 VLAN，以便能够将其添加到 VLAN 组。
- 为了使 VLAN 组正常工作，除了全局启用 DHCP 侦听功能外，用户还必须保证在所有的 VLAN 中都启用了 DHCP 侦听功能。

VLAN 组的限制

能够映射到 VLAN 组的 VLAN 数量不受 Inspur INOS 软件版本的限制。但是，如果 VLAN 组中的 VLAN 数量超过了建议值 32，则不希望设备有移动行为，并且在 VLAN 组中，某些 VLAN 的二层组播会被中断。因此，管理员的责任是在 VLAN 组中配置可行数量的 VLAN。在将 VLAN 添加到已映射到 WLAN 且已有 32 个 VLAN 的 VLAN 组时，会提示警告。但是当新的 VLAN 组映射到多于 32 个 VLAN 的 WLAN 时，会提示错误。
对于 VLAN 组的预期行为，组中映射的 VLAN 必须在设备中。不支持静态 IP 客户端行为。

如何配置 VLAN 组

创建 VLAN 组（CLI）

总结步骤

1. configure terminal

2. vlan group *WORD* **vlan-list** *vlan-ID*

3. end

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 2	**vlan group** *WORD* **vlan-list** *vlan-ID* **示例：** `Device(config)#vlan groupvlangrp1 vlan-list 91-95`	创建给定组名（vlangrp1）的 VLAN 组，并添加命令中列出的所有 VLAN。VLAN 列表的取值范围为 1 到 4096，建议组中的 VLAN 数量为 32。
步骤 3	**end** **示例：** `Device(config)# end`	退出全局配置模式并返回特权 EXEC 模式。或者，按 **CTRL-Z** 退出全局模式。

移除 VLAN 组（CLI）

总步骤

1. configure terminal

2. vlan group *WORD* **vlan-list** *vlan-ID*

3. no vlan group *WORD* **vlan-list** *vlan-ID*

4. end

具体步骤

步骤 1	**configure terminal** **示例：** `Device#`**configure terminal** 进入全局命令模式。
步骤 2	**vlan group** *WORD* **vlan-list** *vlan-ID* **示例：** `Device(config)#`**vlan group vlangrp1 vlan-list 91-95** 创建给定组名（vlangrp1）的 VLAN 组，并添加命令中列出的所有 VLAN。VLAN 列表的取值范围为 1 到 4096，建议组中的 VLAN 数量为 32。
步骤 3	**no vlan group WORD vlan-list vlan-ID** **示例：** `Device(config)#`**no vlan group vlangrp1 vlan-list 91-95** 移除给定组名的 VLAN 组。
步骤 4	**示例：** `Device(config)#` **end** 退出全局配置模式并返回特权 EXEC 模式。或者，按 **CTRL-Z** 退出全局模式。

查看 VLAN 组中的 VLAN

命令	描述
show vlan group	显示 VLAN 组名列表和可用 VLAN 列表。
show vlan group group-name **<group_name>**	显示指定 VLAN 组的具体信息。

接下来做什么？

在配置 VLAN 组后，用户可以配置以下内容：

- VLAN
- VLAN 中继协议（VLAN Trunking Protocol，VTP）
- VLAN 中继
- 语音 VLAN

其他参考资料

相关文档

相关主题	文档标题

有关本章中使用命令的完整语法和使用信息。	*VLAN 命令参考（Inspur6650 交换机）* *二/三层命令参考（Inspur6650 交换机）*
VLAN 接入映射	*安全配置指南（Inspur6650 交换机）* *安全命令参考资源（Inspur6650 交换机）*
VLAN 和移动代理	*移动配置指南，Inspur INOS（Inspur6650 交换机）*
Inspur 灵活网络流量	*Inspur 灵活网络流量配置，Inspur INOS（Inspur 6650 交换机）*
IGMP 侦听	*IP 多播路由命令参考资源（Inspur 6650 交换机）* *IP 多播路由配置指南（Inspur 6650 交换机）*
IPv6	*IPv6 配置指南（Inspur 6650 交换机）* *IPv6 命令参考资源（Inspur 6650 交换机）*
SPAN	*网络管理命令参考资源（Inspur 6650 交换机）* *网络管理配置指南（Inspur 6650 交换机）*
独立平台的配置信息	*基于身份的网络服务配置指南，Inspur INOS（Inspur 6650 交换机）*

错误信息解释

描述	**链接**
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

标准和 RFC

标准/RFC	**标题**
RFC 1573	MIB-II 的接口组的演变
RFC 1757	远程网络监控管理
RFC 2021	使用 SMIv2 的传输控制协议的 SNMPv2 管理信息库

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

VTP 的历史特性和信息

版本	修订
Inspur INOS 12.2	支持 VLAN GUI

配置 VLAN 中继

查询特征信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用浪潮特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问浪潮特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

VLAN 中继的前提

IEEE 802.1Q trunks 在网络的中继策略中增加了如下限制：

- 在通过 IEEE 802.1Qtrunk 连接的 Inspur 设备网络中,每个 VLAN 维护一个生成树实例。非 Inspur 设备的网络中所有 VLAN 维护一个生成树实例。
  - 当通过 IEEE 802.1Q trunk 将 Inspur 设备连接到非 Inspur 设备时，Inspur 设备会将中继的 VLAN 的生成树实例与非 Inspur IEEE 802.1Q 设备的生成树实例合并。但是，每个 VLAN 的生成树信息由 Inspur 设备维护，这些设备由一组非 Inspur IEEE 802.1Q 设备隔开。分离 Inspur 设备的非 InspurIEEE 802.1Q 的云被视为设备之间的单条 trunk 链路。
- 确保 IEEE 802.1Q trunk 的本征 VLAN 在中继链路的两端是相同的。如果中继的一端的本征 VLAN 与另一端的本征 VLAN 不同，则可能会导致生成树环路。
- 在 IEEE 802.1Qtrunk 的本征 VLAN 上禁用生成树，而不禁用网络中每个 VLAN 上的生成树可能会导致生成树环路。建议在 IEEE 802.1Qtrunk 的本征 VLAN 上启用生成树，或在网络中的每个 VLAN 上禁用生成树。在禁用生成树之前，请确保网络无环路。

VLAN 中继的限制条件

以下是 VLAN 的限制条件：

- 中继端口不能是安全端口。
- 中继端口可以分组在 EtherChanne 端口组中，但组中的所有中继必须具有相同的配置。当首次创建组时，所有端口都遵循添加到组中的第一个端口的参数。如果更改其中一个参数的配置，设备会将更改的设置传播到组中的所有端口：
  - 允许 VLAN 列表。
  - 每个 VLAN 的 STP 端口优先级。
  - STP 端口快速设置。
  - 中继状态：
    如果端口组中的一个端口不再是中继，所有端口都不再是中继。
- 建议在每个 VLAN 生成树（PVST）模式下配置不超过 24 个中继端口，并且在多生成树（MST）模式下配置不超过 40 个中继端口。
- 如果用户尝试在中继端口上启用 IEEE 802.1x，则会显示错误消息，并且不启用 IEEE 802.1x。如果用户尝试将启用 IEEE 802.1x 的端口模式更改为中继，则端口模式不会随其他端口改变。
- 动态模式下的接口可以与其邻接接口协商成为中继端口。如果用户尝试在动态端口上启用 IEEE 802.1x，则会显示错误消息，并且不启用 IEEE 802.1x。如果用户尝试将启用 IEEE 802.1x 的端口模式更改为动态，则端口模式不会更改。
- 隧道端口不支持动态中继协议（Dynamic Trunking Protocol，DTP）。
- 设备不支持三层中继；不能配置子接口或在第 3 层接口使用 **encapsulation** 关键字。设备支持第 2 层中继和第 3 层 VLAN 接口，为他们提供等效功能。
- 不能在交换机堆栈中混合使用 Inspur 3850 和 Inspur 6650 交换机。

关于 VLAN 中继的信息

中继概述

中继是一个或多个以太网设备接口与另一个网络设备（如路由器或设备）之间的点对点链路。以太网中继通过单个链路承载多个 VLAN 的流量，用户可以在整个网络上扩展 VLAN。
以下中继封装在所有以太网接口上可用：

- IEEE 802.1Q-中继封装的行业标准。

中继模式

以太网中继接口支持不同的中继模式。可以将接口设置为中继或非中继，或与邻居接口协商中继。自动协商中继，接口必须在同一个 VTP 域中。
中继协商由动态中继协议（DTP）管理，DTP 是点对点协议（Point-to-Point Protocol，PPP）。但是，一些网络互连设备可能不正确地转发 DTP 帧，可能会导致配置错误。

二层接口模式

表 215：二层接口模式

模式	目的
switchport mode access	将接口（接入端口）置于永久非中继模式，并协商将链路转换为非中继链路。无论相邻接口是否是中继接口，接口都将成为非中继接口。
switchport mode dynamic auto	使接口能够将链路转换为中继链路。如果相邻接口设置为 **trunk** 或 **desirable** 模式，则接口变为中继接口。所有以太网接口的默认交换机端口模式是 **dynamic auto** 模式。
switchport mode dynamic desirable	使接口主动尝试将链路转化为中继链路。如果邻接口被设置为 **trunk** 或 **desirable**，或 **auto** 模式，则接口变为中继接口。
switchport mode trunk	使接口进入永久中继模式，并协商将相邻链路转化为中继链路。即使邻接接口不是中继接口，此接口也变为中继接口。
switchportnonegotiate	防止接口生成 DTP 帧。只有当接口 **switchport** 模式为 **access** 或 **trunk** 时，才能使用此命令。必须手动配置邻居接口作为中继接口来建立中继链路。

中继允许 VLAN

缺省情况下，中继端口向所有 VLAN 发送和接收流量。每个中继上允许所有 VLAN ID（1 到 4094）。可以从允许列表中删除 VLAN，阻止来自这些 VLAN 的流量通过中继。

为了降低生成树环路或风暴的风险，可以通过从允许的列表中删除 VLAN 1 来禁用任何单个 VLAN 中继端口上的 VLAN 1。当从 Trunk 端口删除 VLAN 1 时，接口继续发送和接收 VLAN1 中的管理流量，例如，Inspur Discovery Protocol (CDP)，端口聚合协议（PAgP），链路聚合控制协议（Link Aggregation Control Protocol），DTP 以及 VTP。

如果禁止 VLAN 1 的中继端口转换为非中继端口，它会被添加到接入 VLAN。如果接入 VLAN 设置为 1，则端口将添加到 VLAN 1，而不考虑 **switchport trunkallowed** 设置。对于已在端口上禁用的任何 VLAN 也是如此。

如果 VLAN 已启用，VTP 知道此 VLAN，并且 VLAN 位于该端口的允许列表中，则中继端口可以成为该 VLAN 的成员。当 VTP 检测到新启用的 VLAN 并且该 VLAN 在中继端口的允许列表中，中继端口自动成为启用的 VLAN 的成员。当 VTP 检测到新的 VLAN，并且该 VLAN 不在中继端口的允许列表中，中继端口不成为新的 VLAN 的成员。

中继端口的负载均衡

负载均衡会分配设备之间并行中继提供的带宽。为了避免环路，STP 通常仅允许设备之间并行链路中的一个发送流量。使用负载均衡，可以根据流量所属的 VLAN 在链路之间分配流量。可以通过使用 STP 端口优先级或 STP 路径开销在中继端口上配置负载均衡。对于使用 STP 端口优先级的负载均衡，两个负载均衡链路必须连接到同一设备。对于使用 STP 路径成本的负载均衡，每个负载均衡链路可以连接到同一设备或两个不同的设备。

使用 STP 优先级进行网络负载均衡

当同一设备上的两个端口形成环路时，设备使用 STP 端口优先级来决定哪个端口启用，哪个端口阻塞。可以在并行中继端口上设置优先级，以便该端口承载给定 VLAN 的所有流量。对于同一 VLAN，具有较高优先级（较低值）的中继端口转发该 VLAN 流量。对于同一 VLAN，具有较低优先级（较高值）中继端口对于该 VLAN 仍保持阻塞状态。同一个中继端口发送或接收 VLAN 的所有流量。

使用 STP 路径开销进行网络负载均衡

您可以在中继上设置不同的路径开销，把路径开销与不同的 VLAN 关联，为不同 VLAN 阻塞不同端口，以此来配置并行中继以分配 VLAN 流量。VLAN 保持流量分离，并在链路丢失的情况下保持冗余。

特性交互

中继通过以下方式和其他功能交互：

- 中继端口不能是安全端口。
- 中继端口可以分组在 EtherChanne 端口组中，但组中的所有中继必须具有相同的配置。当首次创建组时，所有端口的设置都遵循被添加到组中的第一个端口的参数。如果更改了其中一个参数的配置，设备会将输入的设置传播到组中的所有端口：
  - 允许 VLAN 列表
  - 每个 VLAN 的 STP 端口优先级。
  - STP 端口快速设置。
  - 中继状态：
    如果端口组中的一个端口不再是 trunk，所有端口都不再是 trunk。
- 建议在每个 VLAN 生成树（PVST）模式下配置不超过 24 个 trunk 端口，并且在多生成树（MST）模式下配置不超过 40 个 trunk 端口。
- 如果尝试在中继端口上启用 IEEE 802.1x，则会显示错误消息，并且不启用 IEEE 802.1x。如果尝试将启用 IEEE 802.1x 的端口的模式更改为中继，则端口模式不会更改。
- 动态模式下的端口可以与其邻居协商成为中继端口。如果尝试在动态端口上启用 IEEE 802.1x，则会显示错误消息，并且不启用 IEEE 802.1x。如果尝试将启用 IEEE 802.1x 的端口的模式更改为动态，则端口模式不会更改。

如何配置 VLAN 中继

为了避免中继配置错误，配置连接到设备且不支持 DTP 的接口不转发 DTP 帧，即关闭 DTP。

- 如果不打算将链路作为中继，在接口配置命令 **switchport mode access** 禁用中继。

- 要让不支持 DTP 的设备进行中继，使用接口配置命令 **switchport mode trunk** 和 **switchportnonegotiate** 来配置接口成为中继，但不产生 DTP 帧。

将以太网接口配置为中继端口

配置中继端口（CLI）

因为中继端口发送和接收 VTP 通告，要使用 VTP，必须确保在设备上至少配置了一个中继端口，并且此中继端口连接到另一个设备的中继端口。否则，该设备接收不到任何 VTP 通告。

在开始前

缺省情况下，接口处于二层模式。二层接口的默认模式是 **switchport modedynamic auto**。如果相邻接口支持中继并且配置为允许中继，则链路是二层中继链路；如果接口处于三层模式，则在输入 **switchport** 接口配置命令时，它将成为二层中继。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport mode {dynamic {auto | desirable} | trunk}

5. switchport access vlan*vlan-id*

6. switchport trunk native vlan*vlan-id*

7. end

8. show interfaces *interface-id***switchport**

9. show interfaces *interface-id* **trunk**

10. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>enable`	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet1/0/2`	将指定接口配置为中继，并进入接口配置模式。
步骤 4	**switchport mode {dynamic {auto | desirable} |** **trunk}** **示例：** `Device(config-if)#switchport mode dynamicdesirable`	将接口配置为二层中继（仅当接口为二层接入端口或隧道端口，或指定中继模式时才需要使用）。 • **dynamic auto**——如果相邻接口是 trunk 模式或 desirable 模式，则接口变为中继链路。此模式为默认模式。 • **dynamic desirable**——如果相邻

		接口为 trunk，desirable 或 auto 模式，则将接口设置为中继链路。 • **trunk——**将接口设置为永久中继模式，即使相邻接口不是中继接口，也协商要将链路转换为中继链路。
步骤 5	**switchport access vlan***vlan-id* **示例：** `Device(config-if)# switchport access vlan 200`	（可选）指定默认 VLAN，如果接口停止中继，则使用该 VLAN。
步骤 6	**switchport trunk native vlan***vlan-id* **示例:** `Device(config-if)#switchport trunk native vlan 200`	指定 IEEE 802.1Q 中继的本征 VLAN。
步骤 7	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式。
步骤 8	**show interfaces** *interface-id***switchport** **示例：** `Device# show interfaces gigabitethernet1/0/2switchport`	在 *AdministrativeMode* 和 *Administrative Trunking Encapsulation* 字段中显示接口的交换机端口配置。
步骤 9	**show interfaces***interface-id***trunk** **示例：** `Device# show interfaces gigabitethernet1/0/2trunk`	显示接口的中继配置
步骤 10	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把配置保存在配置文件中。

在中继上定义允许的 VLAN（CLI）

VLAN 1 是所有 Inspur 设备中所有中继端口上的默认 VLAN，以前要求在每个中继链路上始终启用 VLAN 1。可以使用 VLAN 1 最小化特性来禁用任何单个 VLAN 中继链路上的 VLAN 1，以便在 VLAN 1 上不发送或接收用户流量（包括生成树通告）。

总步骤：

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport mode trunk

5. switchport trunk allowed vlan{ *word* **| add | all | except | none | remove}** *vlan-list*

6. end

7. show interfaces *interface-id* **switchport**

8. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>enable`	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet1/0/2`	将指定接口配置为中继，并进入接口配置模式。
步骤 4	**switchport mode trunk** **示例：** `Device(config-if)#switchport mode trunk`	将接口配置为 VLAN 中继端口。
步骤 5	**switchport trunk allowed vlan {** *word* **| add | all |except | none | remove}** *vlan-list* **Example:** `Device(config-if)#switchport trunk allowedvlan remove 2`	（可选）配置中继上允许的 VLAN 列表。 *vlan-list* 参数是从 1 到 4094 的单个 VLAN 号，或由两个 VLAN 号描述的 VLAN 范围，较低的一个在前，用连字符分隔。不要在逗号分隔的 VLAN 参数之间或连字符指定的范围内输入任何空格。 默认情况下允许所有 VLAN
步骤 6	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式。
步骤 7	**show interfaces** *interface-id***switchport** **示例：** `Device# show interfaces gigabitethernet1/0/2switchport`	在显示的 *Trunking VLANs Enabled* 字段中验证条目。
步骤 8	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把配置保存在配置文件中。

更改可修剪列表（CLI）

可修剪列表仅适用于中继端口。每个中继端口都有自己的资格列表。须启用 VTP 修剪才能使此过程生效。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport trunk pruning vlan {add | except | none | remove}*vlan-list [,vlan [,vlan [,,,]]*

5. end

6. show interfaces *interface-id* **switchport**

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>enable`	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet1/0/2`	将指定接口配置为中继，并进入接口配置模式。
步骤 4	**switchport trunk pruning vlan {add	except | none	remove}***vlan-list [,vlan [,vlan [,,,]]*	配置允许从中继中裁剪的 VLAN 列表。有关使用 **add，except，none** 和 **remove** 关键字的说明，请参阅此版本的命令参考。 使用逗号分隔不连续的 VLAN ID，不含空格；用连字符指定 ID 范围。有效 ID 为 2 到 1001。 扩展范围 VLAN（VLAN ID 从 1006 到 4094）不能 修剪。 不可修剪的 VLAN 会接收泛洪流量。 允许修剪 VLAN 的默认列表包含 VLAN 2 到 1001。
步骤 5	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式。
步骤 6	**show interfaces** *interface-id***switchport** **示例：** `Device# show interfaces gigabitethernet1/0/2switchport`	在显示的 *Trunking VLANs Enabled* 字段中验证条目。
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把配置保存在配置文件中。

为未标记流量配置本征 VLAN（CLI）

配置了 IEEE 802.1Q 标记的中继端口可以接收已标记和未标记的流量。默认情况下，设备在为端口配置的本征 VLAN 中转发未标记的流量。默认情况下，本征 VLAN 为 VLAN 1。

本征 VLAN 可以分配使用任何 VLAN ID。

如果数据包具有与输出端口本征 VLAN ID 相同的 VLAN ID，则数据包以无标记的方式发送；否则，设备发送带有标记的数据包。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport trunk native vlan*vlan-id*

5. end

6. show interfaces *interface-id***switchport**

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>enable`	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet1/0/2`	指定配置为 IEEE 802.1Q 中继的接口，并进入接口配置模式。
步骤 4	**switchport trunk native vlan***vlan-id* **Example:** `Device(config-if)#switchport trunk native vlan12`	配置在中继端口上发送和接收未标记流量的 VLAN。 对于 *vlan-id*，取值范围为 1 到 4094。
步骤 5	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式。
步骤 6	**show interfaces** *interface-id***switchport** **示例：** `Device# show interfaces gigabitethernet1/0/2switchport`	在显示的 *Trunking VLANs Enabled* 字段中验证条目。
步骤 7	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）把配置保存在配置文件中。

配置中继端口的负载均衡

使用 STP 端口优先级配置负载均衡（CLI）

如果设备是设备堆栈的成员，则必须使用 **spanning-tree[vlan***vlan-id***] cost***cost* interface 配置命令，而不能使用 **spanning-tree[vlan***vlan-id***] port-priority** *priority* 接口配置命令来选择一个接口

进入转发状态。将较低的开销分配给希望首先选择的接口，给希望最后选择的接口分配较高的开销。

这些步骤描述如何使用 STP 端口优先级配置具有负载均衡的网络。

总步骤

1. enable

2. configure terminal

3. vtp domain *domain-name*

4. vtp mode server

5. end

6. show vtp status

7. show vlan

8. configure terminal

9. interface *interface-id*

10. switchport mode trunk

11. end

12. show interfaces *interface-id***switchport**

13. 在设备 A 或设备堆中的第二个端口重复上述步骤。

14.在设备 B 上重复上述步骤，配置连接到设备 A 上的中继端口。

15. show vlan

16. configure terminal

17. interface *interface-id*

18. spanning-tree vlan*vlan-range* **port-priority** *priority-value*

19. exit

20. interface *interface-id*

21. spanning-tree vlan*vlan-range* **port-priority** *priority-value*

22. end

23. show running-config

24. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>enable`	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入设备 A 的全局配置模式。
步骤 3	**vtp domain** *domain-name* **示例：** `Device(config)#vtp domain workdomain`	配置 VTP 管理域。域名长度为 1 至 32 个字符。
步骤 4	**vtp mode server** **示例：** `Device(config)#vtp mode server`	将设备 A 配置为 VTP 服务器。
步骤 5	**end**	返回特权 EXEC 模式。

	示例： Device(config)# **end**	
步骤 6	**show vtp status** **示例：** Device# **show vtp status**	验证设备 A 和设备 B 上的 VTP 配置。在显示中，检查 *VTP Operating Mode* 和 *VTP Domain Name* 字段。
步骤 7	**show vlan** **示例：** Device# **show vlan**	验证设备 A 的数据库上是否存在 VLAN。
步骤 8	**configure terminal** **示例：** Device# **configure terminal**	进入全局配置模式。
步骤 9	**interface** *interface-id* **示例：** Device(config)# **interface gigabitethernet1/0/1**	定义要配置为中继的接口，并进入接口配置模式。
步骤 10	**switchport mode trunk** **示例：** Device(config-if)#**switchport mode trunk**	将端口配置为中继端口
步骤 11	**end** **示例：** Device(config)# **end**	返回特权 EXEC 模式。
步骤 12	**show interfaces** *interface-id***switchport** **示例：** Device# **show interfaces gigabitethernet1/0/1 switchport**	验证 VLAN 的配置
步骤 13	在设备A或设备堆中的第二个端口重复上述步骤。	
步骤 14	在设备B上重复上述步骤，配置连接到设备A上的中继端口。	
步骤 15	**show vlan** **示例：** Device# **show vlan**	当中继链路启用时，VTP 会把 VTP 和 VLAN 信息传递给设备 B。该命令验证设备 B 是否已经学到了 VLAN 配置。
步骤 16	**configure terminal** **示例：** Device# **configure terminal**	进入设备 A 的全局配置模式
步骤 17	**interface** *interface-id* **示例：** Device(config)# **interface gigabitethernet1/0/1**	设置接口 STP 端口优先级，进入接口配置模式。
步骤 18	**spanning-tree vlan***vlan-range* **port-priority**	为指定的 VLAN 范围分配端口优先级。

	priority-value **示例：** `Device(config-if)# spanning-tree vlan 8-10 port-priority 16`	输入端口优先级值从 0 到 240。端口优先级值按 16 递增。
步骤 19	**exit** **示例：** `Device(config-if)# exit`	返回全局配置模式。
步骤 20	**interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet1/0/2`	设置接口 STP 端口优先级，进入接口配置模式。
步骤 21	**spanning-tree vlan***vlan-range* **port-priority** *priority-value* **示例：** `Device(config-if)# spanning-tree vlan 3-6 port-priority 16`	为指定的 VLAN 范围分配端口优先级。 输入从 0 到 240 的端口优先级值。端口优先级值按 16 递增。
步骤 22	**end** **示例：** `Device(config-if)# end`	返回特权 EXEC 模式。
步骤 23	**show running-config** **示例：** `Device# show running-config`	验证输入
步骤 24	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）将输入保存到配置文件中。

使用 STP 路径开销配置网络负载均衡（CLI）

这些步骤描述如何使用 STP 路径开销配置具有负载均衡的网络。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport mode trunk

5. exit

6. 在设备 A 或设备 A 堆的第二个接口上重复步骤 2 到 4。

7. end

8. show running-config

9. show vlan

10. configure terminal

11. interface *interface-id*

12. spanning-tree vlan*vlan-range* **cost** *cost-value*

13. end

14.重复步骤 9-13，配置设备 A 上另外的中继接口，并设置 VLAN8、9、10 的生成树路径开销为 30。

15. exit

16. show running-config

17. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例：** `Device>enable`	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet1/0/1`	将接口配置为中继，进入接口配置模式。
步骤 4	**switchport mode trunk** **示例：** `Device(config-if)# switchport mode trunk`	将端口配置为中继端口。
步骤 5	**exit** **示例：** `Device(config-if)# exit`	返回全局配置模式
步骤 6	在设备A或设备 A堆的第二个接口上重复步骤2到4。	
步骤 7	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式
步骤 8	**show running-config** **示例：** `Device# show running-config`	验证输入。在显示中确认接口已配置为中继端口。
步骤 9	**show vlan** **示例：** `Device# show vlan`	当中继链路启用时，设备 A 从其他设备接收 VTP 信息。此命令验证设备 A 已获知 VLAN 配置。
步骤 10	**configure terminal** **示例：** `Device# configure terminal`	进入全局配置模式。
步骤 11	**interface** *interface-id* **示例：** `Device(config)# interface gigabitethernet1/0/1`	定义要设置 STP 开销的接口，并进入接口配置模式。
步骤 12	**spanning-tree vlan***vlan-range* **cost** *cost-*	设置 VLAN2-4 的生成树路径开销为

	value **Example:** `Device(config-if)# spanning-tree vlan 2-4 cost 30`	30。
步骤 13	**end** **示例：** `Device(config)# end`	返回特权 EXEC 模式。
步骤 14	重复步骤9-13，配置设备A上另外的中继接口，并设置VLAN8、9、10的生成树路径开销为30。	
步骤 15	**exit** **示例：** `Device(config)# exit`	返回特权 EXEC 模式
步骤 16	**show running-config** **示例：** `Device# show running-config`	验证输入。在显示中验证两个中继接口的路径开销设置是否正确。
步骤 17	**copy running-config startup-config** **示例：** `Device# copy running-config startup-config`	（可选）将输入保存到配置文件中。

接下来做什么？

配置 VLAN 中继后，可以接着配置如下内容：

- VLAN
- VLAN 组
- 语音 VLAN

其他参考资料

相关

相关主题	文档标题
有关本章中使用的命令的完整语法和使用信息。	*VLAN Command Reference (Inspur 6650 Switches)* *ayer 2/3 Command Reference (Inspur 6650 Switches)*
生成树（STP）	*Network ManagementCommand Reference (Inspur6650 Switches)* *NetworkManagementConfigurationGuide(Inspur6650Switchs)*

错误信息解释

描述	链接
为帮助管理员搜索并解决该版本中的系统错误信息，管理员可使用错误信息解释工具。	http://www.icntnetworks.com

标准和 RFC

标准/RFC	标题

RFC 1573	*Evolution of the Interfaces Group of MIB-II*
RFC 1757	*Remote Network Monitoring Management*
RFC 2021	*SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2*

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com

VLAN 的特征历史与信息

版本	修改

配置语音 VLAN

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。

用户可以使用 Inspur 特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问 Inspur 特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

语音 VLAN 的前提

以下是语音 VLAN 的前提：

- 仅在设备接入端口上支持语音 VLAN 配置；中继端口不支持语音 VLAN 配置。

注释： 中继端口可以承载任意数量的语音 VLAN，类似于常规 VLAN。中继端口不支持语音 VLAN 的配置。

- 在启用语音 VLAN 之前，请输入 **trust device inspur-phone** 接口配置命令在设备上启用 QoS。如果使用自动 QoS 功能，则会自动配置这些设置。
- 您必须在连接到 Inspur IP 电话的设备端口上启用 CDP 才能将配置发送到电话（CDP 默认全局在所有设备接口上启用）。

语音 VLAN 的限制条件

无法在语音 VLAN 中配置静态安全 MAC 地址。

关于语音 VLAN 的信息

语音 VLAN

语音 VLAN 特性使接入端口能够承载来自 IP 电话的 IP 语音流量。当设备连接到 Inspur7960 IP 电话时，电话会发送具有三层 IP 优先级和二层服务等级（class of service，CoS）值的语音流量，默认情况下均设置为 5。因为如果数据不均匀地发送，IP 电话呼叫的话音质量可能变差，所以设备支持基于 IEEE 802.1p CoS 的服务质量（quality of service，QoS）。QoS 使用分类和调度，通过可预测的方式从设备发送网络流量。

Inspur IP 电话语音流量

管理员可以配置一个连接了 Inspur IP 电话的接入端口，将一个 VLAN 用于语音流量，另一个 VLAN 用于来自连接到该电话的设备的数据流量。管理员可配置设备上的接入端口发送 Inspur 发现协议（Inspur Discovery Protocol ，CDP）数据包，指示连接的电话通过以下任何方式向设备发送语音流量：

- 在标记有二层 CoS 优先级值的语音 VLAN 中
- 在标记有二层 CoS 优先级值的接入 VLAN 中
- 在接入 VLAN 中，无标记（没有二层 CoS 优先级值）

注释：在所有配置中，语音流量都携带三层 IP 优先级值（对于语音流量，默认值为 5；对于语音控制流量，默认值为 3）。

Inspur IP 电话数据流量

设备还可以处理带标记的数据流量（IEEE 802.1Q 或 IEEE 802.1p 帧类型的流量），该数据流量来自连接到 Inspur IP 电话上的接入端口的设备。管理员可配置设备上的二层接入端口发送 CDP 数据包，指示连接的电话按照下列模式之一配置电话接入端口：

- 在可信模式下，通过 Inspur IP 电话上的接入端口接收的所有流量流经电话时都不会改变。
- 在不可信模式下，通过 Inspur IP 电话上的接入端口接收的 IEEE 802.1Q 或 IEEE 802.1p 帧中的所有流量都会接收配置的二层 CoS 值。默认的二层 CoS 值为 0。不可信模式是默认模式。

注释： 不管电话上接入端口的可信状态如何，来自连接到 Inspur IP 电话的设备的未标记流量流经电话时不会改变。

语音 VLAN 配置指南

- 因为 Inspur 7960 IP 电话还支持与 PC 或其他设备相连，所以连接到 Inspur IP 电话的设备端口可以携带混合流量。管理员可以配置端口，决定 Inspur IP 电话如何传输语音流量和数据流量。
- 语音 VLAN 应该在设备上存在并处于活动状态，以便 IP 电话在语音 VLAN 上正常通信。使用 **show vlan** 特权 EXEC 命令查看是否存在语音 VLAN（在显示输出中列出）。如果未列出 VLAN，请创建语音 VLAN。
- 如果 Inspur 预标准和 IEEE 802.3af 兼容的供电设备没有交流电源供电，以太网供电（Power over Ethernet，PoE）设备能够自动向这些设备提供电源。
- 当配置了语音 VLAN 时，Port Fast 特性会自动启用。当禁用语音 VLAN 时，Port Fast 特性不会自动禁用。
- 如果 Inspur IP 电话和连接到电话的设备在同一个 VLAN 中，它们必须在同一个 IP 子网中。这些条件表明它们在同一 VLAN 中：
  - 它们都使用 IEEE 802.1p 或无标记数据帧。
  - Inspur IP 电话使用 IEEE 802.1p 帧，设备使用无标记帧。
  - Inspur IP 电话使用无标记帧，设备使用 IEEE 802.1p 帧。
  - Inspur IP 电话使用 IEEE 802.1Q 帧，语音 VLAN 与接入 VLAN 相同。
- 因为在同一个子网的流量不会进行路由，所以当 Inspur IP 电话和连接到电话的设备在同一个 VLAN 和子网但使用不同的帧类型时，它们不能通信（路由将消除帧类型的差异）。
- 语音 VLAN 端口也会是以下端口类型：
  - 动态接入端口。
  - IEEE 802.1x 认证端口。

    注释： 如果在配置了语音 VLAN 并且连接了 Inspur IP 电话的接入端口上启用 IEEE 802.1x，则该电话会与设备的失去连接达 30 秒。

  - 受保护端口。
  - SPAN 或 RSPAN 会话的源或目标端口。
  - 安全端口。

注释： 在配置了语音 VLAN 的接口上启用端口安全时，必须将端口上允许的最大安全地址数设置为接入 VLAN 上允许的最大安全地址数加 2。当端口连接到 Inspur IP 电话时，电话最多需要两个 MAC 地址。电话地址会在语音 VLAN 上学习到，也可能在接入 VLAN 上学习到。将 PC 连接到电话需要额外的 MAC 地址。

如何配置语音 VLAN

配置 Inspur IP 电话语音流量（CLI）

管理员可以配置一个连接到 Inspur IP 电话的端口将 CDP 包发送到电话，以配置电话发送语音流量的方式。电话可以在 IEEE 802.1Q 帧中携带具有二层 CoS 值的指定语音 VLAN 的语音流量。它可以使用 IEEE 802.1p 优先级标记为语音流量提供更高的优先级，并通过本征（接入）VLAN 转发所有语音流量。Inspur IP 电话还可以发送无标记的语音流量或使用自己的配置在接入 VLAN 中发送语音流量。在所有配置中，语音流量都携带三层 IP 优先级值（默认值为 5）。

总步骤

1. configure terminal

2. interface *interface-id*

3. trust device inspur-phone

4. switchport voice vlan {*vlan-id* | **dot1p** | **none** | **untagged**}

5. end

6. 使用以下命令之一：

- **show interfaces** *interface-id* **switchport**
- **show running-config interface** *interface-id*

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 2	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/1**	指定连接到电话的接口，并进入接口配置模式。
步骤 3	**trust device inspur-phone** **示例:** `Device(config-if)#` **trust-device inspurphone**	配置接口信任 Inspur IP 电话的入向流量包。
步骤 4	**switchport voice vlan** {*vlan-id* | **dot1p** | **none** | **untagged**}	配置语音 VLAN。 • *vlan-id*——配置电话通过指定的 VLAN 转发所有语音流

	示例: `Device(config-if)#` **switchport voice vlan dot1p**	量。默认情况下，Inspur IP 电话以 IEEE 802.1Q 优先级 5 转发语音流量。有效的 VLAN ID 为 1 到 4094。 • **dot1p——**配置设备接受标记为 VLAN ID 0（本地 VLAN）的语音和数据 IEEE 802.1p 优先级帧。默认情况下，设备会丢弃所有标记为 VLAN 0 的语音和数据流量。如果配置为 802.1p，Inspur IP 电话将使用 IEEE 802.1p 优先级 5 转发流量。 • **none——**允许电话使用自己的配置发送无标记的语音流量。 • **untagged——**配置电话以发送无标记语音流量。
步骤 5	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。
步骤 6	使用以下命令之一： • **show interfaces** *interface-id* **switchport** • **show running-config interface** *interface-id* **示例:** `Device#` **show interfaces gigabitethernet1/0/1 switchport** 或 `Device#` **show running-config interface gigabitethernet1/0/1**	验证语音 VLAN 条目或 QoS 和语音 VLAN 条目。
步骤 7	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将条目保存在设备启动配置文件中。

配置入向数据帧的优先级（CLI）

管理员可以将 PC 或其他数据设备连接到 Inspur IP 电话端口。要处理带标记的数据流量（在 IEEE 802.1Q 或 IEEE 802.1p 帧中），可以设备配置发送 CDP 数据包，以指示电话如何发送连接到 Inspur IP 电话接入端口的设备的数据包。PC 可以生成带有指定 CoS 值的数据包。可以配置电话不更改（信任）或覆盖（不信任）从连接的设备到达电话端口的帧的优先级。

按照以下步骤设置从 Inspur IP 电话上的非语音端口接收的数据流量的优先级：

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport priority extend {**cos** *value* | **trust**}

5. end

6. show interfaces *interface-id* **switchport**

7. copy running-config startup-config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/1**	指定连接到 Inspur IP 电话的接口，并进入接口配置模式。
步骤 4	**switchport priority extend** {**cos** *value*	**trust**} **示例:** `Device(config-if)#` **switchport priority extend trust**	设置从 Inspur IP 电话接入端口接收的数据流量优先级： • **cos** *value*——配置电话以覆盖从 PC 或具有指定 CoS 值的连接设备接收的优先级。该值为 0 到 7 之间的数字，7 为最高优先级。默认优先级为 **cos**0。 • **trust**——配置电话接入端口以信任从 PC 或连接的设备接收的优先级。
步骤 5	**end** **示例:** `Device(config-if)#` **end**	返回特权 EXEC 模式。
步骤 6	**show interfaces** *interface-id* **switchport** **示例:** `Device#` **show interfaces gigabitethernet1/0/1 switchport**	验证条目。
步骤 7	**copy running-config startup-config**	（可选）将条目保存在设备启动配

	示例: `Device#` **copy running-config startup-config**	置文件中。

监控语音 VLAN

要显示接口的语音 VLAN 配置，请使用 **show interfaces** *interface-id* **switchport** 特权 EXEC 命令。

接下来做什么？

配置了语音 VLAN 后可以做以下配置：

- VLAN
- VLAN 组（VLANgroups）
- VLAN 中继（VLAN Trunking）
- VTP

其他参考资料

相关文档

相关主题	文档题目
有关本章中使用的命令的完整语法和使用信息。	*VLAN 命令参考（Inspur 6650 交换机）* *二/三层命令参考（Inspur 6650 交换机）*
其他配置命令及过程。	*Inspur INOS 的 LAN 交换配置指南（Inspur 6650 交换机）* *二/三层配置指南（Inspur 6650 交换机）*
平台无关的配置信息。	*Inspur INOS 的基于身份的网络服务配置指南（Inspur 6650 交换机）*

错误消息解码器

描述	链接
为了帮助您在本版本中研究和解决系统错误消息，请使用错误消息解码器工具（Error Message Decoder tool）。	http://www.icntnetworks.com

标准和 RFC

标准/RFC	题目
RFC 1573	MIB-II 接口组的演进
RFC 1757	远程网络监控管理
RFC 2021	使用 SMIv2 传输控制协议的 SNMPv2 管理信息库

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。	http://www.icntnetworks.com

为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	

语音 VLAN 的特性历史与信息

版本	修改
Inspur INOS 12.2	引入了此功能。

配置私有 VLAN

查询特性信息

用户的软件版本有可能无法支持这部分文档所提到的全部特性。要想查询最新的警示信息和特性信息，可以查看错误搜索工具（Bug Search Tool），也可以查看自己使用的平台及软件版本的版本信息。用户如需查找本文档中所提到的特性，并且了解都有哪些系统版本支持这个特性，可以查看文档最后的特性信息表。
用户可以使用 Inspur 特性导航（Inspur Feature Navigator）来查询各个平台及不同 Inspur 软件版本所支持的信息。要访问 Inspur 特性导航（Inspur Feature Navigator），可以访问 http://www.icntnetworks.com。用户不需要在 icntnetworks.com 注册账户就可以使用这个导航系统。

私有 VLAN 的前提

VTP 1、2 和 3 的透明模式支持使用私有 VLAN。VTP 3 的服务器模式也支持私有 VLAN。
在设备上配置私有 VLAN 时，请始终使用默认的交换数据库管理（Switch Database Management，SDM）模板来平衡单播路由和二层条目之间的系统资源。如果配置了另一个 SDM 模板，请使用 **sdm prefer default** 全局命令设置默认模板。

私有 VLAN 的限制

注释：在某些情况下，虽然配置被接受且没有错误消息，但命令没有效果。

- 不要在具有私有 VLAN 的设备上配置回退桥接。
- 不要将远程 SPAN（RSPAN）VLAN 配置为私有 VLAN 的主 VLAN 或辅助 VLAN。
- 不要在配置了这些其他特性的接口上配置私有 VLAN 的端口：
  - 动态接入端口 VLAN 成员
  - 动态中继协议（Dynamic Trunking Protocol，DTP）
  - IPv6 安全组（Security Group，SG）
  - 端口聚合协议（Port Aggregation Protocol，PAgP）
  - 链路聚合控制协议（Link Aggregation Control Protocol，LACP）
  - 组播 VLAN 注册（Multicast VLAN Registration，MVR）
  - 语音 VLAN
  - Web 缓存通信协议（Web Cache Communication Protocol，WCCP）
- 您可以在私有 VLAN 的端口上配置基于 IEEE 802.1x 端口的身份验证，但不要在私有 VLAN 端口上配置使用端口安全、语音 VLAN 或基于用户 ACL 的 802.1x。
- 私有 VLAN 主机或混杂端口不能是 SPAN 目标端口。如果把 SPAN 目标端口配置为私有 VLAN 的端口，则该端口将变为非活跃状态。
- 如果在主 VLAN 中的混杂端口上配置静态 MAC 地址，则无需向所有关联的辅助 VLAN 添加相同的静态地址。类似地，如果在辅助 VLAN 中的主机端口上配置静态 MAC 地址，则无需向关联的主 VLAN 添加相同的静态 MAC 地址。此外，从私有 VLAN 的端口删除静态 MAC 地址时，不必从私有 VLAN 删除所有配置的 MAC 地址实例。

  注释：在私有 VLAN 的辅助 VLAN 中学习的动态 MAC 地址将复制到主 VLAN。所有 MAC 条目都是在辅助 VLAN 上学习的，即使从主 VLAN 进入的流量也一样。如果在主 VLAN 中动态学习了一个 MAC 地址，它将不会复制到相关的辅助 VLAN 中。
- 只能为主 VLAN 配置三层 VLAN 接口（SVI）。

关于私有 VLAN 的信息

私有 VLAN 域

私有 VLAN 特性解决了服务提供商在使用 VLAN 时面临的两个问题：

- 运行 IP Base 或 IP Services 镜像时，设备最多支持 4094 个活跃 VLAN。如果服务提供商为每个客户分配一个 VLAN，这将限制服务提供商可以支持的客户数。
- 要启用 IP 路由，每个 VLAN 都会被分配一个子网地址空间或一个地址块，这可能会浪费未使用的 IP 地址，并造成 IP 地址管理问题。

使用私有 VLAN 解决了可扩展性问题并为服务提供商提供 IP 地址管理优势，还为客户提供二层安全性。私有 VLAN 将常规 VLAN 域划分为子域。一个子域由一对 VLAN 表示：*主（primary）* VLAN 和 *辅助（secondary）* VLAN。私有 VLAN 可以有多个 VLAN 对，每个子域有一对。私有 VLAN 中的所有 VLAN 对共享相同的主 VLAN。辅助 VLAN ID 区分一个子域与另一个子域。

图 140：私有 VLAN 域

Private VLAN domain	私有 VLAN 域

Primary VLAN	主 VLAN
Subdomain	子域
Secondary community VLAN	辅助团体 VLAN
Secondary isolated VLAN	辅助隔离 VLAN

辅助 VLAN

辅助 VLAN 包含两种类型：

- 隔离 VLAN——隔离 VLAN 内的端口不能在二层级互相通信。
- 团体 VLAN——团体 VLAN 内的端口可以相互通信，但不能与其他团体中的端口在二层级进行通信。

私有 VLAN 端口

私有 VLAN 在同一私有 VLAN 内的端口之间提供二层隔离。私有 VLAN 端口是以下访问端口类型之一：

- 混杂——混杂端口属于主 VLAN，并且可以和所有接口通信，包括属于与主 VLAN 关联的辅助 VLAN 的团体和隔离的主机端口。
- 隔离——隔离端口是属于隔离辅助 VLAN 的主机端口。它与同一私有 VLAN 中的其他端口具有完全的二层隔离，但混杂端口除外。私有 VLAN 阻止所有流量到隔离端口，除了来自混杂端口的流量。从隔离端口接收的流量仅转发到混杂端口。
- 团体——团体端口是属于团体辅助 VLAN 的主机端口。团体端口与同一个团体 VLAN 中的其他端口以及混杂端口通信。这些接口在二层与其他团体中的所有其他接口和其私有 VLAN 内的隔离端口隔离。

注释：中继端口承载来自常规 VLAN 以及主 VLAN、隔离 VALN 和团体 VLAN 的流量。

主 VLAN 和辅助 VLAN 有以下特征：

- 主 VLAN——一个私有 VLAN 仅有一个主 VALN。私有 VLAN 中的每个端口都属于主 VLAN。主 VLAN 承载从混杂端口到（隔离和团体）主机端口和其他混杂端口的单向流量。
- 隔离 VLAN——私有 VLAN 只有一个隔离 VLAN。隔离 VLAN 是辅助 VLAN，承载从主机到混杂端口和网关的上行单向流量。
- 团体 VLAN——团体 VLAN 是辅助 VLAN，承载从团体端口到混杂端口网关以及同一团体中其他主机端口的上行流量。可以在私有 VLAN 中配置多个团体 VLAN。

混杂端口只能服务于一个主 VLAN、一个隔离 VLAN 和多个团体 VLAN。三层网关通常通过混杂端口连接到设备。使用混杂端口，您可以连接各种设备作为到私有 VLAN 的接入点。例如，您可以使用混杂端口从管理工作站监视或备份所有私有 VLAN 服务器。

网络中的私有 VLAN

在交换环境中，可以为单独终端站或公共终端站组的分配单个私有 VLAN 和相关的 IP 子网。要和私有 VLAN 外部通信，终端站需要只与默认网关通信。

您可以使用私有 VLAN 通过以下方式控制对终端站的访问：

- 将连接到终端站的所选接口配置为隔离端口，以防止在二层进行任何通信。例如，如果

终端站是服务器，则此配置会阻止服务器之间的二层通信。

- 将连接到默认网关和选定终端站（例如备份服务器）的接口配置为混杂端口，以允许所有终端站访问默认网关。

您可以通过将主 VLAN、隔离 VLAN 和团体 VLAN 中继到其他支持私有 VLAN 的设备，以在多个设备上扩展私有 VLAN。为了维护私有 VLAN 配置的安全性，并避免把配置的私有 VLAN 用作其他用途，请在所有中间设备（包括没有私有 VLAN 端口的设备）上配置私有 VLAN。

私有 VLAN 的 IP 编址方案

为每个客户分配一个单独的 VLAN 会创建一个低效的 IP 编址方案：

- 为客户 VLAN 分配一块地址可能会导致 IP 地址闲置。
- 如果 VLAN 中的设备数量增加，则已分配的地址数量可能不足以容纳它们。

通过使用私有 VLAN 减少了这些问题，其中私有 VLAN 中的所有成员共享分配给主 VLAN 的公共地址空间。主机连接到辅助 VLAN，DHCP 服务器从分配给主 VLAN 的地址块中给他们分配 IP 地址。在同一主 VLAN 中，后续 IP 地址可以分配给不同辅助 VLAN 中的客户设备。添加新设备时，DHCP 服务器会从大型子网地址池中为其分配下一个可用地址。

多设备上的私有 VLAN

与常规 VLAN 一样，私有 VLAN 可以跨越多个设备。中继端口将主 VLAN 和辅助 VLAN 传送到相邻设备。中继端口把私有 VLAN 当作任何其他的 VLAN 一样对待。

在多个设备上的私有 VLAN 的一个特性是来自设备 A 隔离端口的流量不能到达设备 B 上的隔离端口。

图 141：多台交换机上的私有 VLAN

Trunk ports	中继端口
Switch A	交换机 A
Switch B	交换机 B
Carries VLAN 100,201,and 202 traffic	承载 VLAN 100、201 和 202 的流量
VLAN 100	VLAN 100
VLAN 201	VLAN 201
VLAN 202	VLAN 202
VLAN 100 = Primary VLAN	VLAN 100 = 主 VLAN
VLAN 201 = Secondary isolated VLAN	VLAN 201 = 辅助隔离 VLAN
VLAN 202 = Secondary community VLAN	VLAN 202 = 辅助团体 VLAN

VTP 1、2 和 3 的透明模式支持私有 VLAN。VTP 3 的服务器模式也支持私有 VLAN。如果使用 VTP 3 设置服务器客户端，则在服务器上配置的私有 VLAN 应反映在客户端上。

私有 VLAN 与其他特性的相互作用

私有 VLAN 与单播、广播和组播流量

在常规 VLAN 中，同一 VLAN 内的设备可以在二层互相通信，但连接到不同 VLAN 内接口的设备必须在三层进行通信。在私有 VLAN 中，混杂端口是主 VLAN 的成员，而主机端口则属

于辅助 VLAN。由于辅助 VLAN 与主 VLAN 关联，因此这些 VLAN 的成员可以在二层上相互通信。

在常规 VLAN 中，广播被转发到该 VLAN 中的所有端口。私有 VLAN 广播的转发取决于发该送广播的端口：

- 隔离端口仅向混杂端口或中继端口发送广播。
- 团体端口向所有混杂端口、中继端口和同一个团体 VLAN 中的端口发送广播。
- 混杂端口向私有 VLAN 中的所有端口（其他混杂端口、中继端口、隔离端口和团体端口）发送广播。

组播流量在私有 VLAN 的边界和单个团体 VLAN 内路由或桥接。

组播流量不在同一隔离 VLAN 中的端口之间或不同辅助 VLAN 中的端口之间转发。

私有 VLAN 组播转发支持如下功能：

- 发送方可以在 VLAN 域外，接收方可以在 VLAN 域内。
- 发送方可以在 VLAN 域内，接收方可以在 VLAN 域外。
- 发送方和接收方可以都在同一团体 VLAN 中。

私有 VLAN 及 SVI

在三层设备中，一个设备虚拟接口（SVI）代表 VLAN 的三层接口。三层设备只能通过主 VLAN 而非辅助 VLAN 与私有 VLAN 通信。只应给主 VLAN 配置三层 VLAN 接口（SVI）。不能为辅助 VLAN 配置三层 VLAN 接口。当 VLAN 配置为辅助 VLAN 时，辅助 VLAN 的 SVI 是无效的。

- 如果尝试把含有活跃 SVI 的 VLAN 配置为辅助 VLAN，在禁用 SVI 之前不允许进行此配置。
- 如果尝试在配置为辅助 VLAN 的 VLAN 上创建 SVI，并且辅助 VLAN 已在三层映射，则不会创建 SVI，且会返回错误。如果 SVI 未映射到第 3 层，则能创建 SVI，但它会自动关闭。

当主 VLAN 与辅助 VLAN 关联并映射到辅助 VLAN 时，主 VLAN 上的任何配置都会传送到辅助 VLAN 的 SVI。例如，如果将一个 IP 子网分配给主 VLAN 的 SVI，则此子网是私有 VLAN 的整个 IP 子网地址。

私有 VLAN 和设备堆栈

私有 VLAN 可以在设备堆栈内运行，私有 VLAN 的端口可以位于不同的堆栈成员上。但是，以下对堆栈的更改可能会影响私有 VLAN 的操作：

- 如果堆栈只包含一个私有 VLAN 的混杂端口，并且包含该端口的堆栈成员已从堆栈中删除，该私有 VLAN 中的主机端口会在私有 VLAN 之外失去的连通性。
- 如果一个包含该堆栈中唯一私有 VLAN 混杂端口的堆栈 master 出现故障或离开该堆栈，并且选举出了新的堆栈 master，则在旧堆栈 master 上具有混杂端口的私有 VLAN 中的主机端口会失去私有 VLAN 之外的连通性。
- 如果两个堆栈合并，优胜堆栈上的私有 VLAN 不受影响，但失败设备上的私有 VLAN 配置会在设备重新启动时会丢失。

具有动态 MAC 地址的私有 VLAN

在辅助 VLAN 中学习的 MAC 地址会被复制到主 VLAN，反之则不然。这节省了硬件二层 CAM 空间。主 VLAN 始终被用来进行两个方向上的转发查找。

如果需要，在私有 VLAN 的主 VLAN 中学习的动态 MAC 地址将复制到辅助 VLAN 中。例如，如果在辅助 VLAN 上动态接收一个 MAC 地址，则该 MAC 地址将被当作主 VLAN 的一部分。在隔离 VLAN 的情况下，同一个 MAC 的阻塞条目会在 MAC 地址表中添加给辅助 VLAN。因此，在辅助域中的主机端口上学习的 MAC 将作为阻塞类型条目安装。即使流量从主 VLAN 进入，所有 MAC 条目都在辅助 VLAN 上学习。

然而，如果 MAC 地址是在主 VLAN 中动态学习的，该 MAC 地址将不会被复制到相关联的辅

助 VLAN 中。

具有静态 MAC 地址的私有 VLAN

与传统模型相比，用户无需复制私有 VLAN 主机的静态 MAC 地址 CLI。

示例：

- 在传统模型中，如果用户配置了静态 MAC 地址，则也需要在关联 VLAN 中添加相同的静态 MAC 地址。例如，用户在 VLAN 101 的 1/0/1 端口上配置了 MAC 地址 A（其中 VLAN 101 是辅助 VLAN，VLAN 100 是主 VLAN），则用户必须进行如下配置：

  mac-address static A vlan 101 interface G1/0/1

  mac-address static A vlan 100 interface G1/0/1
- 在这个设备中，用户无需将 MAC 地址复制到相相关联的 VLAN 中。对于上面的示例，用户只需要进行如下配置：

  mac-address static A vlan 101 interface G1/0/1

私有 VLAN 与 VACL / QOS 的相互作用

当与其他平台中的“单向”VLAN 相比，私有 VLAN 在此设备的情况下是双向的。

在二层转发查找后，正确的出向 VLAN 映射产生，所有基于出向 VLAN 的特性处理都在出向 VLAN 环境中进行。

当二层中的数据帧帧在私有 VLAN 内转发时，VLAN 映射会被应用在在入向侧和出向侧。当数据帧从私有 VLAN 内部被路由到外部端口时，在入向侧应用私有 VLAN 的映射。类似地，当帧从外部端口被路由到私有 VLAN 时，在出向侧应用私有 VLAN。这适用于桥接和被路由流量。

桥接：

- 对于从辅助 VLAN 到主 VLAN 的上行流量，辅助 VLAN 的 MAP 应用在入向侧，主 VLAN 的 MAP 应用在出向侧。
- 对于从主 VLAN 到辅助 VLAN 的下行流量，主 VLAN 的 MAP 应用在入向，辅助 VLAN 的 MAP 应用在出方向。

路由：

如果有两个私有 VLAN 域——PV1（sec1，prim1）和 PV2（sec2，prim2）。当帧从 PV1 路由到 PV2：

- 在入端口应用 sec1 的 MAP 和 prim1 的 L3 ACL。
- 在出端口应用 sec2 的 MAP 和 prim2 的 L3 ACL。

对于从独立主机端口到混杂端口的上行或下行的数据包，在入方向应用隔离 VLAN 的 VACL，在出方向应用主 VLAN 的 VACL。这允许用户在同一主 VLAN 域中为不同的辅助 VLAN 配置不同的 VACL。

注释： 不需要使用双向团体 VLAN，因为此设备上的私有 VLAN 始终是双向的。

私有 VLAN 以及 HA 支持

PVLAN 会和高可用性（High Availability，HA）特性无缝协作。切换之前 master 上存在的私有 VLAN 在切换后应该相同（新 master 在 INOS 和 FED 上具有与旧 master 类似的 PVLAN 配置）。

私有 VLAN 配置指南

私有 VLAN 的默认配置

无私有 VLAN 配置。

辅助 VLAN 及主 VLAN 的配置

请按照以下指南配置私有 VLAN：

- VTP 1、2 和 3 的透明模式支持私有 VLAN。如果设备运行的是版本 1 或 2 的 VTP，则必须将 VTP 设置为透明模式。配置私有 VLAN 后，不应把 VTP 模式更改为客户端或服务器。VTP 版本 3 在所有模式下都支持私有 VLAN。
- 使用 VTP 版本 1 或 2 时，在配置私有 VLAN 后，请使用 **copy running-configstartup config** 特权 EXEC 命令在设备启动配置文件中保存 VTP 透明模式配置和私有 VLAN 配置。否则，如果设备重置，它默认会成为 VTP 服务器模式，不支持私有 VLAN 配置。 VTP 版本 3 支持私有 VLAN。
- 版本 1 和 2 的 VTP 不传播私有 VLAN 的配置。必须在每个要设置私有 VLAN 端口的设备上配置私有 VLAN，除非设备运行可传播私有 VLAN 的 VTP 版本 3。
- 不能将 VLAN 1 或 VLAN 1002 至 1005 配置为主 VLAN 或辅助 VLAN。扩展 VLAN（VLAN 的 ID 为 1006 到 4094）可以属于私有 VLAN。
- 主 VLAN 可以有一个隔离的 VLAN 和与其关联的多个团体 VLAN。隔离或团体 VLAN 只能有一个与其关联的主 VLAN。
- 虽然私有 VLAN 包含多个 VLAN，但整个私有 VLAN 只运行一个生成树协议（Spanning Tree Protocol，STP）实例。当辅助 VLAN 与主 VLAN 关联时，主 VLAN 的 STP 参数会传播到辅助 VLAN。
- 从 TFTP 服务器复制 PVLAN 配置并将其应用于运行配置时，将不会建立 PVLAN 关联。需要检查并确保主 VLAN 与所有辅助 VLAN 相关联。
  也可以用 **configure replace flash:config_file force**，而不使用 **copy flash:config_filerunning-config**.。
- 您可以在私有 VLAN 上启用 DHCP 侦听。当在主 VLAN 上启用 DHCP 侦听时，它会传播侦听信息到辅助 VLAN。如果在辅助 VLAN 上配置 DHCP 侦听，且已配置了主 VLAN，则配置不会生效。
- 在私有 VLAN 端口上启用 IP 源地址防护时，必须在主 VLAN 上启用 DHCP 侦听功能。
- 建议裁剪在设备的中继上不承载流量的私有 VLAN。
- 可以对主 VLAN、隔离 VLAN 和团体 VLAN 应用不同的服务质量（quality of service，QoS）配置。
- 注意粘性 ARP 的以下事项：
  - 粘性 ARP 条目是在 SVI 和三层接口上学习的。这些条目不会过期。
  - **ip sticky-arp** 全局配置命令仅在属于私有 VLAN 的 SVI 上支持。
  - **ip sticky-arp** 接口配置命令仅支持：
    - 三层接口
    - 属于常规 VLAN 的 SVI
    - 属于私有 VLAN 的 SVI

    有关使用 **ip sticky-arp** 全局配置和 **ip sticky-arpinterface** 接口配置命令的更多信息，请参阅此版本的命令参考。
- 您可以在主 VLAN 和辅助 VLAN 上配置 VLAN 映射。然而，建议在私有 VLAN 的主 VLAN 和辅助 VLAN 上配置相同的 VLAN 映射。
- PVLAN 是双向的。它们可以应用在入向和出向。
  当二层中的数据帧在私有 VLAN 内转发时，VLAN 映射会应用在在入向和出向端。当数据帧从私有 VLAN 内部路由到外部端口时，私有 VLAN 映射会应用在在入向侧。类似地，当数据帧从外部端口路由到私有 VLAN 时，私有 VLAN 映射会应用在在出向侧。

桥接：

- 对于从辅助 VLAN 到主 VLAN 的上行流量，在入向侧应用辅助 VLAN 的 MAP，在出向侧应用主 VLAN 的 MAP。
- 对于从主 VLAN 到辅助 VLAN 的下行流量，在入方向应用主 VLAN 的 MAP，在出方向应用辅助 VLAN 的 MAP。

路由：

如果有两个私有 VLAN 域——PV1（sec1，prim1）和 PV2（sec2，prim2）。当数据帧从 PV1 路由到 PV2：

- 在入端口应用 sec1 的 MAP 和 prim1 的 L3 ACL。
- 在出端口应用 sec1 的 MAP 和 prim2 的 L3 ACL。
- 对于从独立主机端口到混杂端口的上行或下行的分组，在入方向应用隔离 VLAN 的 VACL，在出方向应用主 VLAN 的 VACL。这允许用户在同一主 VLAN 域中为不同的辅助 VLAN 配置不同的 VACL。

要过滤私有 VLAN 的特定 IP 流量，应该将 VLAN 映射同时应用于主 VLAN 和辅助 VLAN。

- 可以仅在主 VLAN 的 SVI 上应用路由器 ACL。该 ACL 应用于主 VALN 和辅助 VLAN 的三层流量。
- 虽然私有 VLAN 在二层提供主机隔离，但主机可以在三层相互通信。
- 私有 VLAN 支持交换端口分析器（Switched Port Analyzer，PAN）的如下特性：
  - 可以将私有 VLAN 端口配置为 SPAN 源端口。
  - 可以在主 VLAN、隔离 VLAN 和团体 VLAN 上使用基于 VLAN 的 SPAN（VSPAN），或者仅使用一个 VLAN 上的 SPAN 来分别监视出向或入向的流量。

私有 VLAN 端口配置

请按照以下指南配置私有 VLAN 端口：

- 仅使用私有 VLAN 配置命令将端口分配给主 VLAN、隔离 VLAN 或团体 VLAN。当某一 VLAN 是私有 VLAN 配置的一部分时，分配给配置为主 VLAN、隔离 VLAN 或团体 VLAN 的二层接入端口是非活动状态。二层中继端口保持在 STP 转发状态。
- 不要将属于 PAgP 或 LACP EtherChannel 的端口配置为私有 VLAN 端口。虽然端口是私有 VLAN 配置的一部分，但它的任何 EtherChannel 配置都是非活动状态的。
- 在隔离和团体主机端口上启用 Port Fast 和 BPDU 防护，以防止由于配置错误导致的 STP 环路，并加速 STP 收敛。当启用时，STP 将 BPDU 防护功能应用于所有配置 Port Fast 的二层 LAN 端口。不要在混杂端口上启用 Port Fast 和 BPDU 防护。
- 如果删除在私有 VLAN 配置中使用的 VLAN，则与该 VLAN 关联的私有 VLAN 端口将变为非活动状态。
- 如果设备是中继连接的，并且主 VLAN 和辅助 VLAN 没有从中继删除，则私有 VLAN 端口可以在不同的网络设备上。

如何配置私有 VLAN

配置私有 VLAN

配置私有 VLAN，请执行以下步骤：

注释： VTP 1、2 和 3 的透明模式支持私有 VLAN。VTP 3 的服务器模式也支持私有 VLAN。

总步骤

1. 将 VTP 设置为 **transparent** 模式

2. 创建主 VLAN 和辅助 VLAN，并将它们关联起来。
3. 将接口配置为隔离或团体主机端口，并将 VLAN 的全体成员分配给主机端口。
4. 将接口配置为混杂端口，并将混杂端口映射到主、辅助 VLAN 对。
5. 如果使用 VLAN 间路由，则配置主 SVI，并将辅助 VLAN 映射到主 VLAN。
6. 验证私有 VLAN 配置。

具体步骤

步骤 1	将 VTP 设置为 **transparent** 模式 **注释：**对于 VTP3，您可以将模式设置为服务器或透明模式。
步骤 2	创建主 VLAN 和辅助 VLAN，并将他们相连。 请见在私有 VLAN 中配置和连接 VLAN **注释：**如果尚未创建 VLAN，则私有 VLAN 配置过程将创建该 VLAN。
步骤 3	将接口配置为隔离或团体主机端口，并将 VLAN 的全体成员分配给主机端口。 请见配置二层接口作为私有 VLAN 主机接口
步骤 4	将接口配置为混杂端口，并将混杂端口映射到主、辅助 VLAN 对。 请见配置二层接口作为私有 VLAN 混杂接口
步骤 5	如果使用 VLAN 间路由，则配置主 SVI，并将辅助 VLAN 映射到主 VLAN。 请见将辅助 VLAN 映射到主 VLAN 的三层 VLAN 接口
步骤 6	验证私有 VLAN 配置。

在私有 VLAN 中配置和关联 VLAN

在退出 VLAN 配置模式之前，**private-vlan** 命令不会生效。

在私有 VLAN 中配置和关联 VLAN，请执行以下步骤：

总步骤

1. enable

2. configure terminal

3. vtp mode transparent

4. vlan *vlan-id*

5. private-vlan primary

6. exit

7. vlan *vlan-id*

8. private-vlan isolated

9. exit

10. vlan *vlan-id*

11. private-vlan community

12. exit

13. vlan *vlan-id*

14. private-vlan community

15. exit

16. vlan *vlan-id*

17. private-vlan association [**add** | **remove**] *secondary_vlan_list*

18. end

19. show vlan private-vlan [**type**] or **show interfaces status**

20. copy running-config startup config

具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**vtp mode transparent** **示例:** `Device(config)#` **vtp mode transport**	将 VTP 设置为透明模式（禁用 VTP）。 **注释：**对于 VTP 3，可设置为服务器或透明模式。
步骤 4	**vlan** *vlan-id* **示例:** `Device(config)#` **vlan 20**	进入 VLAN 配置模式，并指定或创建一个 VLAN 作为主 VLAN。该 VLAN 的 ID 范围为 2 到 1001 以及 1006 到 4094。
步骤 5	**private-vlan primary** **示例:** `Device(config-vlan)#` **private-vlan primary**	将该 VLAN 指定为主 VLAN。
步骤 6	**exit** **示例:** `Device(config-vlan)#` **exit**	返回全局配置模式。
步骤 7	**vlan** *vlan-id* **示例:** `Device(config)#` **vlan 501**	（可选）进入 VLAN 配置模式，并指定或创建一个 VLAN 作为隔离 VLAN。该 VLAN 的 ID 范围为 2 到 1001 以及 1006 到 4094。
步骤 8	**private-vlan isolated** **示例:** `Device(config-vlan)#` **private-vlan isolated**	将该 VLAN 指定为隔离 VLAN。
步骤 9	**exit** **示例:** `Device(config-vlan)#` **exit**	返回全局配置模式。
步骤 10	**vlan** *vlan-id* **示例:**	（可选）进入 VLAN 配置模式，并指定或创建一个 VLAN 作为团体 VLAN。该 VLAN 的 ID 范围为 2 到 1001 以及 1006 到 4094。

	Device(config)# **vlan 502**	
步骤 11	**private-vlan community** **示例:** Device(config-vlan)# **private-vlan community**	将该 VLAN 指定为团体 VLAN。
步骤 12	**exit** **示例:** Device(config-vlan)# **exit**	返回全局配置模式。
步骤 13	**vlan** *vlan-id* **示例:** Device(config)# **vlan 503**	（可选）进入 VLAN 配置模式，并指定或创建一个 VLAN 作为团体 VLAN。该 VLAN 的 ID 范围为 2 到 1001 以及 1006 到 4094。
步骤 14	**private-vlan community** **示例:** Device(config-vlan)# **private-vlan community**	将该 VLAN 指定为团体 VLAN。
步骤 15	**exit** **示例:** Device(config-vlan)# **exit**	返回全局配置模式。
步骤 16	**vlan** *vlan-id* **示例:** Device(config)# **vlan 20**	进入步骤 4 中指定的主 VLAN 的 VLAN 配置模式。
步骤 17	**private-vlan association** [**add** | **remove**] *secondary_vlan_list* **示例:** Device(config-vlan)# **private-vlan association 501-503**	将辅助 VLAN 与主 VLAN 关联。参数可以是单个私有 VLAN ID 或用连字符连接的私有 VLAN 的 ID 范围。 • *secondary_vlan_list* 参数不能包含空格。它可以包含多个逗号分隔的项目。每个项目可以是单个私有 VLAN 的 ID 或用连字符连接的私有 VLAN 的 ID 范围。 • *secondary_vlan_list* 参数可以包含多个团体 VLAN 的 ID，但只能包含一个隔离的 VLAN 的 ID。 • 输入 *secondary_vlan_list* 或使用带有 *secondary_vlan_list* 的 **add** 关键字将辅助 VLAN 与主 VLAN 关联。 • 使用 **remove** 关键字及 *secondary_vlan_list* 可以清除辅助 VLAN 和主 VLAN 之间的关联。

		• 在退出 VLAN 配置模式之前，命令不会生效。
步骤 18	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 19	**show vlan private-vlan** [**type**] or **show interfaces status** **示例:** `Device#` **show vlan private-vlan**	验证配置。
步骤 20	**copy running-config startup config** **示例:** `Device#` **copy running-config startup-config**	将配置的条目保存在设备启动配置文件中。

将二层接口配置为私有 VLAN 主机端口

用户可以按照以下步骤将二层接口配置为私有 VLAN 主机端口，并将其与主 VLAN 和辅助 VLAN 关联：

注释：隔离 VLAN 和团体 VLAN 都是辅助 VLAN。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport mode private-vlan host

5. switchport private-vlan host-association *primary_vlan_id secondary_vlan_id*

6. end

7. show interfaces [*interface-id*] **switchport**

8. copy running-config startup-config

具体步骤

	命令或操作	**目的**
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** *interface-id*	进入配置二层接口的接口配置模式。

	示例: `Device(config)` **interface gigabitethernet1/0/22**	
步骤 4	**switchport mode private-vlan host** **示例:** `Device(config-if)#` **switchport mode private-vlanhost**	将二层端口配置为私有 VLAN 主机端口。
步骤 5	**switchport private-vlan host-association** *primary_vlan_idsecondary_vlan_id* **示例:** `Device(config-if)#` **switchport private-vlan host-association 20 501**	将二层端口与私有 VLAN 关联。 **注释：**这是将 PVLAN 与第 2 层接口相关联所必需的步骤。
步骤 6	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 7	**show interfaces** [*interface-id*] **switchport** **示例:** `Device#` **show interfaces gigabitethernet1/0/22 switchport**	验证配置。
步骤 8	**copy running-config startup-config** **示例:** `Device#` **copy running-config startup-config**	（可选）将配置的条目保存在设备启动配置文件中。

将二层接口配置为私有 VLAN 混杂端口

用户可以按照以下步骤将二层接口配置为私有 VLAN 混杂端口，并将其与主 VLAN 和辅助 VLAN 进行映射：

注释：隔离 VLAN 和团体 VLAN 都是辅助 VLAN。

总步骤

1. enable

2. configure terminal

3. interface *interface-id*

4. switchport mode private-vlan promiscuous

5. switchport private-vlan mapping *primary_vlan_id* {add | remove} *secondary_vlan_list*

6. end

7. show interfaces [*interface-id*] switchport

8. copy running-config startup config

具体步骤

	命令或操作	**目的**

步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface** *interface-id* **示例:** `Device(config)#` **interface gigabitethernet1/0/2**	进入配置二层接口的接口配置模式。
步骤 4	**switchport mode private-vlan promiscuous** **示例:** `Device(config-if)#` **switchport mode private-vlan promiscuous**	将二层端口配置为私有 VLAN 混杂端口。
步骤 5	**switchportprivate-vlanmapping** *primary_vlan_id* {**add** | **remove**} *secondary_vlan_list* **示例:** `Device(config-if)#` **switchport private-vlan mapping 20 add 501-503**	将私有 VLAN 混杂端口映射到主 VLAN 和所选的辅助 VLAN。 • *secondary_vlan_list* 参数不能包含空格。它可以包含多个逗号分隔的项目。每个项目可以是单个私有 VLAN 的 ID 或用连字符连接的私有 VLAN 的 ID 范围。 • 输入 *secondary_vlan_list* 或使用带有 *secondary_vlan_list* 的 **add** 关键字将辅助 VLAN 映射到私有 VLAN 混杂端口。 • 使用 **remove** 关键字及 *secondary_vlan_list* 可以清除辅助 VLAN 和私有 VLAN 混杂端口之间的映射。
步骤 6	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 7	**show interfaces** [*interface-id*] **switchport** **示例:** `Device#` **show interfaces gigabitethernet1/0/2**	验证配置。

	switchport	
步骤 8	**copy running-config startup config** **示例:** `Device#` **copy running-config startup-config**	将配置的条目保存在设备启动配置文件中。

将辅助 VLAN 映射到主 VLAN 三层接口

如果私有 VLAN 会用于 VLAN 间路由，则为主 VLAN 配置 SVI，并将辅助 VLAN 映射到 SVI。
注释：隔离 VLAN 和团体 VLAN 都是辅助 VLAN。
用户可以按照以下步骤将辅助 VLAN 映射到主 VLAN 的 SVI，以允许对私有 VLAN 流量进行三层交换：
总步骤
1. enable
2. configure terminal
3. interface vlan *primary_vlan_id*
4. private-vlan mapping [add | remove] *secondary_vlan_list*
5. end
6. show interface private-vlan mapping
7. copy running-config startup config
具体步骤

	命令或操作	目的
步骤 1	**enable** **示例:** `Device>`**enable**	进入特权 EXEC 模式。在提示时输入密码。
步骤 2	**configure terminal** **示例:** `Device#` **configure terminal**	进入全局配置模式。
步骤 3	**interface vlan** *primary_vlan_id* **示例:** `Device(config)#` **interface vlan 20**	进入主 VLAN 的接口配置模式，并将 VLAN 配置为 SVI。VLAN 的 ID 范围为 2 到 1001 以及 1006 到 4094。
步骤 4	**private-vlan mapping** [**add** | **remove**] *secondary_vlan_list* **示例:** `Device(config-if)#` **private-vlan mapping 501-503**	将辅助 VLAN 映射到主 VLAN 的三层 VLAN 接口，以允许三层交换私有 VLAN 入口流量。 **注释：private-vlan mapping** 接口配置命令只影响三层交换的私有 VLAN 流量。 • *secondary_vlan_list* 参数不能包含空格。它可以包含多个逗号分隔的项目。每个项目可以是单个私

		有 VLAN 的 ID 或用连字符连接的私有 VLAN 的 ID 范围。 • 输入 *secondary_vlan_list* 或使用带有 *secondary_vlan_list* 的 **add** 关键字将辅助 VLAN 映射到主 VLAN。 • 使用 **remove** 关键字及 *secondary_vlan_list* 可以清除辅助 VLAN 和主 VLAN 之间的映射。
步骤 5	**end** **示例:** `Device(config)#` **end**	返回特权 EXEC 模式。
步骤 6	**show interface private-vlanmapping** **示例:** `Device#` **show interfaces private-vlan mapping**	验证配置。
步骤 7	**copy running-config startup config** **示例:** `Device#` **copy running-config startup-config**	将配置的条目保存在设备启动配置文件中。

监控私有 VLAN

下表展示了用于监控私有 VLAN 的命令。

表 216：私有 VLAN 的监控命令

命令	目的
show interfaces status	显示接口的状态，包括它们所属的 VLAN。
show vlan private-vlan [**type**]	显示设备或设备堆栈的私有 VLAN 信息。
show interface switchport	显示接口上的私有 VLAN 配置。
show interface private-vlanmapping	显示私有 VLAN 映射 VLAN SVI 的有关信息。
show platform vlan pvlan	显示 FED 侧的 PVLAN 信息。
show platform vlan pvlan hardware	显示 FAD 侧的 PVLAN 所拥有的所有硬件资源。

私有 VLAN 的配置示例

示例：在私有 VLAN 中配置和关联 VLAN

此示例显示如何将 VLAN 20 配置为主 VLAN，将 VLAN 501 配置为隔离 VLAN，将 VLAN 502 和 503 配置为团体 VLAN，以将它们关联到私有 VLAN，并验证配置：

```
Device# configure terminal
Device(config)# vlan 20
Device(config-vlan)# private-vlan primary
Device(config-vlan)# exit
Device(config)# vlan 501
Device(config-vlan)# private-vlan isolated
Device(config-vlan)# exit
Device(config)# vlan 502
Device(config-vlan)# private-vlan community
Device(config-vlan)# exit
Device(config)# vlan 503
Device(config-vlan)# private-vlan community
Device(config-vlan)# exit
Device(config)# vlan 20
Device(config-vlan)# private-vlan association 501-503
Device(config-vlan)# end
Device# show vlan private-vlan
Primary Secondary Type
--------- -------------- -----------------
20 501 isolated
20 502 community
20 503 community
```

示例：将接口配置为主机端口

此示例展示如何将接口配置为私有 VLAN 主机端口，将其与私有 VLAN 对关联，并验证配置：

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/22
Device(config-if)# switchport mode private-vlan host
Device(config-if)# switchport private-vlan host-association 20 501
Device(config-if)# end
Device# show interfaces gigabitethernet1/0/22 switchport
Name: Gi1/0/22
Switchport: Enabled
Administrative Mode: private-vlan host
Operational Mode: private-vlan host
Administrative Trunking Encapsulation: negotiate
Operational Trunking Encapsulation: native
Negotiation of Trunking: Off
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
Voice VLAN: none
Administrative private-vlan host-association: 20 501
```

```
Administrative private-vlan mapping: none
Administrative private-vlan trunk native VLAN: none
Administrative private-vlan trunk Native VLAN tagging: enabled
Administrative private-vlan trunk encapsulation: dot1q
Administrative private-vlan trunk normal VLANs: none
Administrative private-vlan trunk private VLANs: none
Operational private-vlan:
20 501
<output truncated>
```

示例：将接口配置为私有 VLAN 混杂端口

此示例展示如何将接口配置为私有 VLAN 混杂端口，并将其映射到私有 VLAN。该接口是主 VLAN 20 的成员，且其映射了辅助 VLAN 501 至 503。

```
Device# configure terminal
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# switchport mode private-vlan promiscuous
Device(config-if)# switchport private-vlan mapping 20 add 501-503
Device(config-if)# end
```

使用 **show vlan private-vlan** 或 **show interface status** 特权 EXEC 命令显示设备上的主 VLAN、辅助 VLAN 以及私有 VLAN 端口。

示例：将辅助 VLAN 映射到主 VLAN 接口

此示例展示如何将 VLAN 501 和 502 的接口映射到主 VLAN 10，以允许进行从私有 VLAN 501 到 502 的辅助 VLAN 入向流量的路由：

```
Device# configure terminal
Device(config)# interface vlan 20
Device(config-if)# private-vlan mapping 501-503
Device(config-if)# end
Device# show interfaces private-vlan mapping
Interface Secondary VLAN Type
--------- -------------- -----------------
vlan20 501 isolated
vlan20 502 community
vlan20 503 community
```

示例：监控私有 VLAN

以下示例展示了 **show vlan private-vlan** 命令的输出：

```
Device# show vlan private-vlan
Primary Secondary Type Ports
```

```
------- --------- ----------------- ----------------------------------------
20 501 isolated Gi1/0/22, Gi1/0/2
20 502 community Gi1/0/2
20 503 community Gi1/0/2
```

接下来做什么？

可进行以下配置：

- VTP
- VLANs
- VLAN 中继
- VLAN 成员策略服务器（VLAN Membership Policy Server，VMPS）
- 语音 VLAN

其他参考资料

相关文档

相关主题	文档题目
CLI 命令	LAN 交换命令参考，InspurINOS 版本

标准和 RFC

标准/RFC	题目
RFC 1573	
RFC 1757	
RFC 2021	

技术助手

描述	链接
Inspur 支持网站为排错和解决 Inspur 产品的技术问题提供了大量的在线资源，包括文档及工具。 为了接收产品的安全及技术信息，管理员可以订阅多种服务，如产品报警工具（通过现场通知访问），Inspur 技术服务简讯以及 RSS 源。 访问 Inspur 支持网站的大部分工具都需要提供 icntnetworks.com 的用户 ID 及密码。	http://www.icntnetworks.com