云翼 300 系列交换机软件配置指南， 1.5 版

2014 年 2 月 21 日

思科系统公司
www.cisco.com

思科在全球设有 200 多个办事处。
地址、电话号码和传真号码
在思科网站上列出，网址为：
www.cisco.com/go/offices。

文本部件号：OL-28657-01

目 录

前言

本文档介绍如何在您的网络中配置云翼 300 系列交换机。

本指南不介绍如何安装交换机。有关信息，请参阅您的交换机硬件安装指南。

约定

本出版物采用如下约定表示指令和信息：

对于命令描述：

- 命令和关键字以粗体文本表示。
- 由您提供值的参数以*斜体*表示。
- 方括号 ([ ]) 表示可选元素。
- 花括号 ({ }) 将必选项组合在一起，竖线 (|) 用于分隔可替代元素。
- 方括号内的花括号和竖线 ([{ | }]) 表示可选元素中的某个必选项。

对于交互示例：

- 终端会话和系统显示内容以screen字体显示。
- 输入的信息以**bold screen**字体显示。
- *非打印字符（如密码或制表符）放在尖括号 (< >) 中。*

注释、注意事项和警告使用如下约定和符号：

注意 表示*读者需要记录*。注释中包含有用的建议或包含对本手册中所没有的材料的引用。

注意 表示*读者应当小心*。在这种情况下，您的操作可能会导致设备损坏或数据丢失。

警告 **重要安全性说明**

此警告符号表示存在危险。您目前所处情形有可能遭受身体伤害。在操作任何设备之前，请务必意识到触电危险并熟悉标准工作程序，以免发生事故。请根据每个警告结尾处的声明号来查找此设备随附的安全警告的翻译文本。声明 1071

请妥善保存这些说明

相关出版物

- *Cisco Smart Install 配置指南*
- *云翼 300 系列交换机安装指南*
- *云翼 300 系列交换机发行说明*

获取文档和提交服务请求

有关获取文档、提交服务请求和收集其他信息的信息，请参阅以下网页上的*思科产品文档更新*：http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html。

通过 RSS 源的方式订阅*思科产品文档更新*（其中列出了所有新的和修订过的思科技术文档），并通过阅读器应用将内容直接发送至您的桌面。RSS 源是一种免费服务。

云翼 300 系列交换机

云翼 300 系列交换机概述

云翼 300 系列交换机为作为智能安装网络一部分的室内环境提供基于云的服务。该交换机可让室内设备和应用充分利用网络基础架构智能。

云翼 300 系列交换机是云网络中的一个关键组件。

室内客户端交换机

云翼 300 系列交换机可用作教室、酒店客房、医院病房和办公室中的室内客户端交换机。该交换机是提供 PC、交换和路由功能的混合平台。它为以下组件提供各种接口：

- 输入设备（如键盘、鼠标、麦克风和摄像头）
- 输出设备（如显示器、电视、投影仪、扬声器和头戴式耳机）

该交换机还集成了一个无线接入点，允许 802.11b/g/n 客户端通过无线方式连接到网络。

网络聚合器

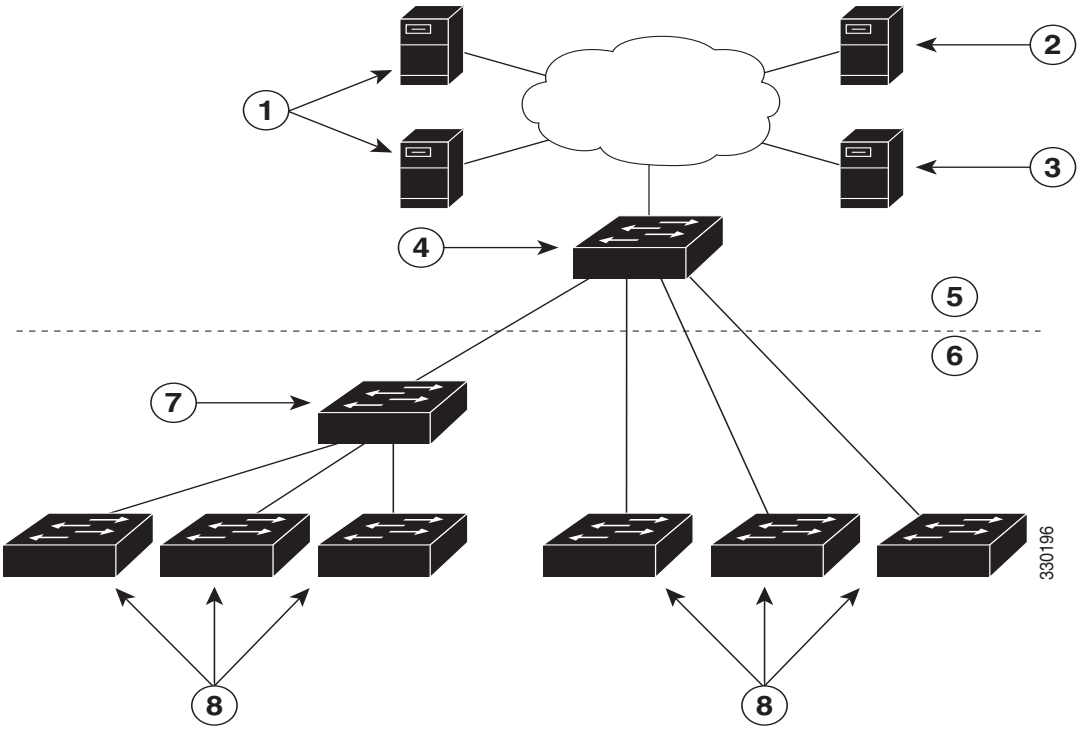
以太网交换机（如 Catalyst 3000 系列交换机）是一个智能安装指挥交换机，可安全地管理云翼 300 交换机。 Medianet 上的智能服务和 Catalyst 交换机中的安全性可提高云服务交付的质量。

云和应用交付服务器

数据中心服务器提供特定于环境的内容、计算能力、存储和托管以及其他云应用，包括用于客户端交换机的第三方应用。

图 1-1 显示了一个典型的智能安装配置，云翼 300 系列交换机在其中用作客户端交换机。

图 1-1　　典型智能安装云翼网络图



1	云和应用交付服务器	5	聚合层
2	DHCP 服务器	6	接入层
3	TFTP 服务器	7	中间交换机
4	指挥交换机	8	客户端交换机

云翼 300 系列交换机的功能和应用

注意　本指南中未介绍这些功能和应用。

云翼 300 系列交换机可提供如下功能和应用：

- 云翼监视
- 云翼视频会议
- 视频流
- 显示 Adobe Flash 文件
- 显示 Windows Office 文件
- 显示 PDF 文件
- MP3 和 AAC 音频支持
- AVI、 WAV 和 MPG4 视频支持以及 H.264/AVC 编码和解码视频支持
- JPG 支持

- WebEx 会议
- 软件升级功能
- 屏幕捕捉功能
- VLC 播放器
- VPN 支持
- SNMP 支持
- VLAN 支持（您最多可以在云翼 300 交换机上配置 6 个主用 VLAN。）

集中管理和配置

云翼 300 系列交换机专用于智能安装网络。智能安装是一种即插即用配置和镜像管理功能。您可以将一台交换机发送到某个位置，然后将其接入网络并接通电源即可使用，无需进行本地配置。

智能安装网络

使用智能安装的网络包括一组网络连接设备（称为客户端），这些设备由一个通用第 3 层交换机或充当指挥交换机的路由器提供服务。

所有云翼 300 系列交换机都可用作智能安装网络中的智能安装客户端交换机。最终用户不需要配置客户端交换机：所有交换机都通过 TFTP 服务器上安装的 GUI 集中配置并通过指挥交换机进行管理。

智能安装指挥交换机

智能安装指挥交换机为客户端交换机的镜像和配置提供单一管理点。在网络中首次安装客户端交换机时，指挥交换机会自动检测新交换机并确定要下载的正确镜像和配置文件。它可以将一个 IP 地址和主机名分配给客户端。如果网络中的独立交换机被具有相同 SKU 的另一交换机（即具有相同产品 ID 的交换机）代替，后者将自动获取与上一交换机相同的配置和镜像。

智能安装指挥交换机支持网络中的以下功能：

- 云翼配置文件的配置管理
- 来自相邻交换机和客户端交换机的思科发现协议 (CDP) 信息整合
- DHCP 监听

指挥交换机还可以支持网络中的以下功能，或者网络中的其他设备可以提供这些功能：

- DHCP 服务器
- 用于存储镜像和配置文件的 TFTP 服务器

有关配置指挥交换机的信息，请参阅第 2-5 页上的“ 配置智能安装指挥交换机” 一节。

DHCP 和 TFTP 服务器

DHCP 是智能安装网络的骨干：智能安装客户端交换机使用 DHCP 获取 IP 地址，而智能安装指挥交换机则监听 DHCP 消息。所有 DHCP 通信都通过指挥交换机，因此它可以监听来自客户端交换机的所有 DHCP 数据包。

指挥交换机可用作 DHCP 和 TFTP 服务器，并且可以存储配置和镜像文件。不过，在大型网络中，指挥交换机可以使用第三方 DHCP 和 TFTP 服务器。客户端交换机从 TFTP 服务器下载镜像和配置文件。

DHCP 服务器为客户端交换机提供 IP 地址，而 DHCP 选项则用于发送信息和文件：

- 客户端交换机的 TFTP 服务器 IP 地址
- 客户端交换机的配置文件名
- 客户端交换机的镜像文件名和位置
- 客户端交换机的主机名
- 网络中其他交换机的指挥交换机 IP 地址

有关配置 DHCP 服务器的信息，请参阅第 2-2 页上的“ 配置 DHCP 服务器” 一节。有关配置 TFTP 服务器的信息，请参阅第 2-7 页上的“ 配置 TFTP 服务器” 一节。

注意 在不使用 DHCP 为客户端分配 IP 地址的网络中，可以在客户端交换机上配置静态 IP 地址。有关详情，请参阅第 2-5 页上的“ 使用静态 IP 地址” 一节。

GUI 和配置文件

您可以使用 GUI 将云翼 300 系列交换机集中配置为智能安装客户端。您需要在 TFTP 服务器上安装 GUI（请参阅第 2-8 页上的“ 在 CentOS/Fedora 服务器上设置 GUI” 一节）。

指挥交换机需要一些信息来管理客户端交换机。您可以使用 GUI 来创建下列信息文件，指挥交换机可以从 TFTP 服务器来检索这些文件：

镜像列表文件

指定需要加载到客户端交换机的镜像：

- 根文件系统镜像 - 指定交换机的关键文件和子目录。根文件系统与根目录位于同一分区。在交换机启动时，所有文件系统都会附加到根文件系统。
- 可启动的 Linux 内核镜像 - 指定交换机上运行的 Linux 操作系统内核。
- 思科应用镜像 - 指定交换机上运行的思科应用。
- 第三方应用镜像 - 指定交换机上运行的第三方应用。
- 字体镜像 - 指定桌面和 GUI 的语言。

您可以将镜像列表文件配置为智能安装指挥交换机配置文件的一部分。

云翼配置文件

指定适用于组中所有客户端交换机的一种常见配置，并指定适用于组中单个客户端交换机的一种单独配置。组包括 SSID、无线安全设置和无线电设置等组件。可以使用 CLI 输入 GUI 中特定于云翼 300 系列交换机的命令，创建云翼配置文件（请参阅第 2-23 页上的“ 管理云翼配置文件” 一节和第 4 章，“ 配置本地 CLI - Clish”）。

智能安装指挥交换机配置文件

指定将哪个镜像列表文件和云翼配置文件加载到客户端交换机组。

应用和升级镜像和配置文件

当交换机启动时，它将连接到指挥交换机。如果交换机检测到任何新镜像或配置文件，它会自动在出厂默认模式下重新启动，然后下载和安装新镜像或配置文件。

下面是支持的镜像和配置升级类型：

- 由用户启动升级 - 适用于网络中连接到指挥交换机的单个客户端交换机。用户可以关闭并开启交换机，也可以按住“ 重置” 按钮 5 秒钟从出厂默认模式启动。在任一情况下，交换机都会连接到指挥交换机并可以检测任何新镜像或配置文件。
- 由用户启动升级 - 适用于网络中连接到指挥交换机的单个客户端交换机。管理员通过使用 GUI 或连接到交换机（例如，通过 Telnet 连接）重启交换机来启动升级。

有关详细信息，请参阅第 2-32 页上的“ 交换机镜像和配置升级” 一节。

注意　不支持按需升级和计划下载。无法使用 **write erase** 和 **reload**、 **vstack download-image**、 **vstack download-config** 或 **archive download-sw** 特权 EXEC 命令从指挥交换机升级交换机。

配置智能安装网络

配置指挥交换机和 DHCP 服务器

指挥交换机用于管理网络中的交换机。对于每组交换机，指挥交换机配置文件会指定镜像列表文件和云翼配置文件。

指挥交换机可管理下列云翼配置文件：

- 启动配置 - 客户端交换机在启动时使用的配置。
- 备份配置 - 存储在指挥交换机中的客户端交换机启动配置的精确副本。
- 种子配置 - 指挥交换机上作为客户端交换机启动配置基础的配置。如果找不到启动和备份配置，指挥交换机将向客户端交换机提供种子配置。

有关管理和创建云翼配置文件的信息，请参阅第 2-23 页上的“ 管理云翼配置文件” 一节。

DHCP 和智能安装

注 如果您的智能安装网络不使用 DHCP，请参阅第 2-5 页上的“ 使用静态 IP 地址” 一节。

注 本节介绍在智能安装网络上配置指挥交换机和 DHCP 服务器的一些基本任务。有关智能安装和智能安装指挥交换机的更多信息，请参阅 *智能安装配置指南版本 12.2(58)SE*。

典型的智能安装网络使用 DHCP 协议和 DHCP 服务器。在 DHCP 网络中，自动在指挥交换机上启用 DHCP 监听。指挥交换机监听 DHCP 向客户端交换机提供和请求的信息，并使用 DHCP 监听插入在智能安装操作中使用的 DHCP 选项。

智能安装网络中的 DHCP 服务器可以通过下列方式之一定位：

- 智能安装指挥交换机可充当网络中的 DHCP 服务器。当 DHCP 提供的信息发送到客户端交换机时，指挥交换机将分配 IP 地址并在提供和确认中作为 DHCP 选项分配配置、镜像和主机名。默认情况下启用 DHCP 监听。
- DHCP 服务器可以是智能安装网络中的其他设备（第三方服务器）。在此情况下，客户端和 DHCP 服务器之间的 DHCP 数据包通过指挥交换机。

注　您可以配置一个加入窗口时段，使指挥交换机只能在该窗口期间修改 DHCP 提供的信息并向客户端发送镜像和配置文件。加入窗口可将智能安装限制在一个指定时段，并作为安全防范措施，控制客户端何时能够接收这些文件。请参阅 *智能安装配置指南版本 12.2(58)SE* 中的“ 使用加入窗口” 一节。

- 第三方服务器和指挥交换机 DHCP 服务器可以在网络中共存。在此情况下，指挥交换机在智能安装网络中仅负责交换机的 DHCP 请求。指挥交换机维护智能安装数据库和池。第三方服务器维护其他 DHCP 数据库功能。

配置 DHCP 服务器

DHCP 服务器可以是指挥交换机、运行 Cisco IOS 的另一思科设备或者第三方服务器。还可以让指挥交换机充当智能安装 DHCP 服务器，让另一设备执行所有其他 DHCP 服务器功能。

无论采用哪种方式，都要使用以下步骤之一将 Cisco 设备设置为 DHCP 服务器。如果选择将第三方设备配置为 DHCP 服务器，请遵循产品文档中有关配置网络地址和 TFTP 服务器的说明。

- 第 2-3 页上的将指挥交换机配置为 DHCP 服务器
- 第 2-4 页上的将另一设备配置为 DHCP 服务器

DHCP 服务器配置指南

- 如果指挥交换机（或运行 Cisco IOS 的另一设备）是 DHCP 服务器，并且网络重新加载，则服务器可能向交换机分配新 IP 地址，该地址之后可能不再可用。如果指挥交换机 IP 地址改变，它将不再是智能安装指挥交换机。为防止发生这种情况，应通过在 DHCP 服务器上输入 **ip dhcp remember** 全局配置命令或 **remember** DHCP 池配置命令来启用 *DHCP 记忆功能*。
- 如果使用外部设备作为 DHCP 服务器，则可以配置 DHCP 服务器为指挥交换机 IP 地址发送选项 125/子选项 16，以避免篡改 DHCP 服务器。
- 第三方 DHCP 服务器需要一个“ IP 地址到 MAC 地址” 的绑定，以确保在重新加载时为交换机提供相同的 IP 地址。

将指挥交换机配置为 DHCP 服务器

可以将指挥交换机配置为 DHCP 服务器，并直接从智能安装指挥交换机创建 DHCP 服务器池。

在特权 EXEC 模式下，在指挥交换机上执行如下步骤将其配置为 DHCP 服务器：

	命令	目的
步骤 1	**config terminal**	进入全局配置模式。
步骤 2	**vstack director** *ip_ address*	通过在设备上输入接口的 IP 地址将该设备配置为智能安装指挥交换机。
步骤 3	**vstack basic**	将该设备启用为智能安装指挥交换机。
步骤 4	**vstack dhcp-localserver** *poolname*	为智能安装 DHCP 服务器地址池创建一个名称，并进入 vstack DHCP 池配置模式。
步骤 5	**address-pool** *network-number mask prefix-length*	指定 DHCP 地址池的子网号和子网掩码。 **注**　前缀长度指定构成地址前缀的位数。前缀是指定客户端掩码的另一种方法。前缀长度前面必须有一个正斜杠 (/)。
步骤 6	**default-router** *ip_address*	为池指定 DHCP 默认路由器的 IP 地址。 **注**　我们建议 DHCP 的默认路由器地址位于 VLAN 1 上。新安装的设备将在 VLAN 1 上搜索 DHCP 和 TFTP。
步骤 7	**file-server** *address*	指定 TFTP 服务器的 IP 地址。 **注**　如果指挥交换机也是 TFTP 服务器，则必须启用它。请参阅第 2-7 页上的“ 配置 TFTP 服务器” 一节。
步骤 8	**exit**	返回全局配置模式。
步骤 9	**ip dhcp remember**	（可选）配置 DHCP 服务器以记住设备的 IP 绑定。如果网络或设备重新加载，DHCP 服务器会向客户端发送 IP 地址。此地址与重新加载前所用的 IP 地址相同。
步骤 10	**end**	返回特权 EXEC 模式。
步骤 11	**copy running-config startup config**	（可选）将输入保存在配置文件中。
步骤 12	**show dhcp server**	通过显示由设备识别的 DHCP 服务器来验证配置。

下例显示了如何将智能安装指挥交换机配置为 DHCP 服务器：

```
Director# configure terminal
Director(config)# vstack director 1.1.1.20
Director(config)# vstack basic
Director(config)# vstack dhcp-localserver pool1
Director(config-vstack-dhcp)# address-pool 1.1.1.0 255.255.255.0
Director(config-vstack-dhcp)# default-router 1.1.1.30
Director(config-vstack-dhcp)# file-server 1.1.1.40
Director(config-vstack-dhcp)# exit
Director(config)# ip dhcp remember
Director(config)# end
```

指挥交换机上默认启用 DHCP 监听。

将另一设备配置为 DHCP 服务器

如果智能安装指挥交换机不是 DHCP 服务器，则可以使用 Cisco IOS DHCP 命令在智能安装网络外部配置服务器池。指挥交换机必须连接到 DHCP 服务器。有关配置其他 DHCP 服务器选项的步骤，请参阅 Cisco.com 上 *Cisco IOS IP 配置指南版本 12.2* 中“ IP 寻址服务”一节中的“ 配置 DHCP”部分或 *Cisco IOS IP 配置指南版本 15.1* 中的“ IP 寻址服务”一节。

从特权 EXEC 模式开始，执行以下步骤：

	命令	目的
步骤 1	**config terminal**	进入全局配置模式。
步骤 2	**ip dhcp pool** *poolname*	为 DHCP 服务器地址池创建一个名称，并进入 DHCP 池配置模式。
步骤 3	**bootfile** *filename*	指定要使用的配置文件的名称。
步骤 4	**network** *network-number mask prefix-length*	指定 DHCP 地址池的子网号和子网掩码。 **注**　前缀长度指定构成地址前缀的位数。前缀是指定客户端掩码的另一种方法。前缀长度前面必须有一个正斜杠 (/)。
步骤 5	**option 150** *address*	指定 TFTP 服务器的 IP 地址。
步骤 6	**remember**	（可选）配置 DHCP 池以记住设备的 IP 绑定。如果网络或设备重新加载， DHCP 服务器会向客户端发送 IP 地址。此地址与重新加载前所用的 IP 地址相同。
步骤 7	**end**	返回特权 EXEC 模式。

下例显示了如何将另一设备配置为 DHCP 服务器：

```
Switch # configure terminal
Switch(config)# ip dhcp pool pool1
Switch(dhcp-config)# network 10.10.10.0 255.255.255.0
Switch(dhcp-config)# bootfile config-boot.text
Switch(dhcp-config)# option 150 10.10.10.1
Switch(dhcp-config)# remember
Switch(config-if)# end
```

当指挥交换机是第 3 层交换机时，默认情况下会启用 DHCP 监听。当 DHCP 服务器与指挥交换机之间存在中继代理时，必须在中继代理上启用 DHCP 监听。

要在 Cisco DHCP 中继设备上启用 DHCP 监听，请输入下列全局配置命令：

ip dhcp snooping

ip dhcp snooping vlan 1

ip dhcp snooping vlan *vlan-id*（用于任何其他配置的智能安装 VLAN）

no ip dhcp snooping information option（如果 DHCP 服务器在运行 Cisco IOS）

还必须在与服务器连接的指挥交换机界面上输入 **ip dhcp snooping trust** 接口配置命令。

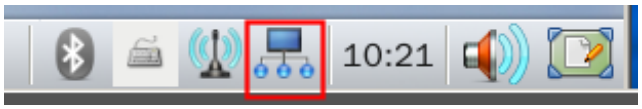
如果指挥交换机与 DHCP 服务器位于不同的 VLAN 上，则必须在与客户端交换机连接的 VLAN 界面上启用 IP 路由，并输入如下命令：

ip helper *address*（DHCP 服务器的 IP 地址）

使用静态 IP 地址

在使用静态 IP 地址的智能安装网络中，需要通过本地桌面 GUI 配置客户端交换机上的 IP 地址。

步骤 1　从本地桌面双击状态栏上的有线网络图标。



注　如果状态栏上没有有线网络图标，请单击**主页**按钮，然后转到“ 设置” >“ 有线网络”。

步骤 2　在“ 有线网络” 窗口中，单击**网络配置**按钮。

步骤 3　在“ 用户验证” 窗口中，输入根用户名和密码。

步骤 4　在“ 网络配置” 窗口中，从“ 网络类型” 下拉列表中选择“ **手动 （静态）**”。

步骤 5　输入 IP 地址 （必填）、掩码 （必填）、网关 （可选）、 DNS 服务器和 IBD 指挥交换机 （可选） IP 地址。

注　如果您不配置网关，请输入以下 Linux 命令，将主机路由添加到 IBD 和网络文件系统 (NFS) 服务器中：

```
# route add -net ip_address netmask subnet_mask gw gateway_ip_address
```

步骤 6　点击**确定**。

配置智能安装指挥交换机

智能安装网络中的指挥交换机必须是运行 Cisco IOS 版本 12.2(58)SE 或更高版本的第 3 层交换机或者是运行 Cisco IOS 版本 15.1(3)T 或更高版本的路由器。

要将某个设备配置为指挥交换机，请在 **vstack director** *ip_ address* 全局配置命令中输入其中一个第 3 层接口的 IP 地址，并通过输入 **vstack basic** 命令将其启用为指挥交换机。

注　如果输入 **no vstack** 全局配置命令，以在设备上禁用智能安装，则在该设备上将不支持 **vstack director** *ip_ address* 和 **vstack basic** 全局配置命令。要在设备上重新启用智能安装，请输入 **vstack** 全局配置命令。

在将设备配置为指挥交换机时，默认情况下系统会自动在 VLAN 1 上启用 DHCP 监听，并且指挥交换机会构建指挥交换机数据库。

数据库列出智能安装网络中的客户端设备，并包括每个交换机的以下信息：

- 产品型号 (PID)
- MAC 地址
- IP 地址

- 主机名
- 网络拓扑，包括与交换机对接的相邻交换机
- 序列号

注 当指挥交换机是交换机时，默认情况下在 VLAN 1 上启用 DHCP 监听。它还在通过输入 **vstack vlan** *vlan-range* 全局配置命令配置的任何其他智能安装管理 VLAN 上启用。我们建议使用 VLAN 1 接口作为指挥交换机 IP 地址，因为新安装的客户端使用 VLAN 1 来广播 DHCP 请求。

在使用 DHCP 来分配 IP 地址的智能安装网络中，只需要配置指挥交换机。客户端交换机不需要任何配置。

一组客户端只能有一个指挥交换机，并且不能配置备份指挥交换机。如果指挥交换机出现故障：

- 必须重构指挥交换机数据库。
- 对不支持智能安装的交换机进行的任何升级都可能失败。
- 累积的下载状态已丢失。
- 在指挥交换机重新启动之前，可能不会发生配置备份。

在下列情况下，指挥交换机可能更改状态并成为客户端交换机：

- 具有指挥交换机 IP 地址的指挥交换机接口被删除。
- 具有指挥交换机 IP 地址的指挥交换机接口被删除。
- 指挥交换机 IP 地址发生更改。

如果指挥交换机成为客户端，系统会禁用 DHCP 监听，且不再使用指挥交换机数据库。

如果指挥交换机 IP 地址由 DHCP 提供，并且在客户端交换机上配置了不同的指挥交换机 IP 地址，则客户端将是指挥交换机的智能安装网络的较长部分。

智能安装依赖 TFTP 服务器来存储镜像和配置文件。TFTP 服务器可以是一个外部设备，也可以使用指挥交换机来充当 TFTP 服务器。如果指挥交换机是 TFTP 服务器，则指挥交换机上的可用闪存文件空间必须能够容纳客户端 Cisco IOS 镜像和配置文件。请参阅第 2-7 页上的“ 配置 TFTP 服务器” 一节。

在使用 DHCP 的智能安装网络中， DHCP 服务器可以是一个外部设备，也可以使用指挥交换机来充当 DHCP 服务器。请参阅第 2-2 页上的“ DHCP 服务器配置指南” 一节。指挥交换机在 VLAN 1 上和配置为智能安装管理 VLAN 的任何其他 VLAN 上监听并让它的所有 DHCP 数据包通过。来自中间交换机或客户端交换机或者来自外部 DHCP 服务器的所有网络 DHCP 数据包必须通过指挥交换机，该指挥交换机必须能够监听来自客户端的所有 DHCP 数据包。

注 DCHP 提供的智能安装选项是选项 125 子选项 5 （镜像列表文件）、选项 125 子选项 16 （指挥交换机 IP 地址）和选项 67 （配置文件）。

指挥交换机通过从网络智能安装交换机收集信息，为网络构建一个拓扑指挥交换机数据库。指挥交换机使用该数据库：

- 将配置文件和镜像分配到客户端。
- 作为参考为网络交换机的按需升级获取产品型号、镜像名称和配置文件。

指挥交换机基于 CDP 更新从相邻交换机和通过支持智能安装的客户端发送到指挥交换机的智能安装消息定期更新指挥交换机数据库。更新包含有关相邻客户端的信息。

配置 TFTP 服务器

智能安装在 TFTP 服务器上存储镜像和配置文件。

如果使用外部设备作为 TFTP 服务器，则镜像列表文件和配置文件存将储在 TFTP 服务器上的如下位置：

文件	TFTP 服务器上的位置
镜像列表文件	/opt/Tftproot/imglist
云翼配置文件	/opt/Tftproot/sb_conf
组关联文件	/opt/Tftproot/

如果使用外部设备作为 TFTP 服务器，则属于镜像列表文件的文件将存储在 TFTP 服务器上的如下位置：

文件	TFTP 服务器上的位置
出厂模式操作系统	/opt/Tftproot/images/FM_OS
操作系统文件（包括根文件系统镜像和可启动 Linux 内核镜像）	/opt/Tftproot/images/OS
思科应用文件	/opt/Tftproot/images/CiscoApp
第三方应用文件	/opt/Tftproot/images/Partner
字体应用	/opt/Tftproot/images/Fonts

指挥交换机可以用作服务器，从而无需使用外部 TFTP 服务设备。如果指挥交换机是 TFTP 服务器，则镜像和配置文件将存储在指挥交换机闪存中。如果指挥交换机没有可用内存空间，则可以将文件存储在第三方服务器上并指向该位置。

如果 TFTP 服务器是第三方设备，并且创建了具有相同名称的另一文件，则禁用服务器选项以更改文件名称。否则，可能创建重复的镜像列表文件。

当指定 **flash**: 作为从中检索文件的位置时，指挥交换机将自动获取所需镜像和配置文件，并充当 TFTP 服务器。

选择指南

选择指挥交换机作为 TFTP 服务器的指南：

- 指挥交换机上的总闪存空间（已用和未用）必须足够大，以便容纳指挥交换机镜像和配置文件以及客户端交换机所需的镜像和配置文件。
- 指挥交换机上必须有足够的可用闪存来容纳客户端 Cisco IOS 镜像和配置文件。 Cisco IOS 镜像文件的大小各不相同，具体取决于产品型号和镜像的大小。
- 每个客户端配置文件的副本都存储在指挥交换机上闪存文件系统的根目录中。每个计划的客户端必须有足够的空间。
- 大多数指挥交换机设备都有足够的闪存来容纳一个客户端 Cisco IOS 镜像和少量客户端配置文件。例如， Catalyst 3750 交换机的最大闪存大小可能有 64 MB，这只能容纳四至五个镜像（具体取决于镜像大小）。
- 如果指挥交换机是交换机，并且智能安装网络包括的客户端交换机有多个产品型号，则应使用外部 TFTP 服务器。

安装和使用 GUI

使用 GUI

您可以在不同的交换机组中为不同的受众配置和部署云翼 300 系列交换机。例如，一所小学可以将一组应用提供给一年级学生，将另一组应用提供给二年级学生。您需要使用 GUI 创建两个交换机组，将一年级学生的交换机与一个交换机组关联，将二年级学生的交换机与另一交换机组关联，然后生成一个不同的交换机客户端配置文件并将其推送到每个交换机组。

使用 GUI 来配置和管理智能安装网络中的云翼 300 系列交换机。您可以点击此处

- 创建交换机组（请参阅第 2-16 页上的“ 创建交换机组” 一节）。
- 将个别交换机添加到 GUI 或将交换机列表导入 GUI（请参阅第 2-17 页上的“ 管理云翼交换机列表” 一节）。
- 根据如下一个或多个组件创建智能安装组-设备关联文件，将交换机添加到交换机组：
  - MAC 地址
  - 产品型号 (PID)
  - 位置

  有关更多信息，请参阅第 2-19 页上的“ 向交换机组添加成员” 一节。
- 创建云翼配置文件（请参阅第 2-23 页上的“ 管理云翼配置文件” 一节）。

在 CentOS/Fedora 服务器上设置 GUI

注　设置 GUI 需要熟悉 Linux Distribution 和 Linux Shell 命令。

注　在 GUI 安装期间，必须保持互联网连接。

在设置 GUI 之前，下载并安装以下软件：

- Internet Explorer 版本 9.0 或 Firefox Mozilla 8.0.1 或更高版本
- CentOs 6.2/Fedora 14、 15 和 16
- Yum Package Manager - 此软件应属于 Fedora 预装软件包的一部分。如果您在安装 Fedora 过程中未安装 Yum Package Manager，可以从 http://yum.baseurl.org/ 下载它。

注　请确保 Yum 存储库可访问。例如，如果您使用 Fedora 14，则会弃用您的系统的默认存储库。

更新 Yum 存储库

例如，如果已弃用您的系统的默认存储库，那么您需要手动更新它们。

注　如果您使用 CentOS，请参阅 http://wiki.centos.org/AdditionalResources/Repositories 了解存储库信息。

在安装 GUI 之前，请使用以下步骤在 Fedora 上更新 Yum 存储库：

步骤 1　更新 /etc/yum.repos.d/fedora.repo

```
[fedora]
name=Fedora $releasever - $basearch
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/releases/$releasever/Everythin
g/$basearch/os/
baseurl=http://archives.fedoraproject.org/pub/archive/fedora/linux/releases/$releasever/Ev
erything/$basearch/os/
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=fedora-$releasever&arch=$basea
rch
enabled=1
metadata_expire=7d
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch

[fedora-debuginfo]
name=Fedora $releasever - $basearch - Debug
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/releases/$releasever/Everythin
g/$basearch/debug/
baseurl=http://archives.fedoraproject.org/pub/archive/fedora/linux/releases/$releasever/Ev
erything/$basearch/debug/
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=fedora-debug-$releasever&arch=
$basearch
enabled=0
metadata_expire=7d
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch

[fedora-source]
name=Fedora $releasever - Source
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/releases/$releasever/Everythin
g/source/SRPMS/
baseurl=http://archives.fedoraproject.org/pub/archive/fedora/linux/releases/$releasever/Ev
erything/$basearch/SRPMS/
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=fedora-source-$releasever&arch
=$basearch
enabled=0
metadata_expire=7d
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch
```

步骤 2　更新 /etc/yum.repos.d/fedora-updates.repo

```
[updates]
name=Fedora $releasever - $basearch - Updates
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/updates/$releasever/$basearch/
baseurl=http://archives.fedoraproject.org/pub/archive/fedora/linux/updates/$releasever/$ba
search/
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=updates-released-f$releasever&
arch=$basearch
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch

[updates-debuginfo]
name=Fedora $releasever - $basearch - Updates - Debug
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/updates/$releasever/$basearch/
debug/
baseurl=http://archives.fedoraproject.org/pub/archive/fedora/linux/updates/$releasever/$ba
search/debug/
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=updates-released-debug-f$relea
sever&arch=$basearch
enabled=0
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch

[updates-source]
name=Fedora $releasever - Updates Source
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/updates/$releasever/SRPMS/
baseurl=http://archives.fedoraproject.org/pub/archive/fedora/linux/updates/$releasever/SRP
MS/
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=updates-released-source-f$rele
asever&arch=$basearch
enabled=0
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch
```

步骤 3　更新 /etc/yum.repos.d/fedora-updates-testing.repo

```
[updates-testing]
name=Fedora $releasever - $basearch - Test Updates
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/updates/testing/$releasever/$b
asearch/
baseurl=http://archives.fedoraproject.org/pub/archive/fedora/linux/updates/testing/$releas
ever/$basearch/
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=updates-testing-f$releasever&a
rch=$basearch
enabled=0
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch

[updates-testing-debuginfo]
name=Fedora $releasever - $basearch - Test Updates Debug
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/updates/testing/$releasever/$b
asearch/debug/
baseurl=http://archives.fedoraproject.org/pub/archive/fedora/linux/updates/testing/$releas
ever/$basearch/debug/
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=updates-testing-debug-f$releas
ever&arch=$basearch
enabled=0
```

```
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch

[updates-testing-source]
name=Fedora $releasever - Test Updates Source
failovermethod=priority
#baseurl=http://download.fedoraproject.org/pub/fedora/linux/updates/testing/$releasever/SR
PMS/
baseurl=http://archives.fedoraproject.org/pub/archive/fedora/linux/updates/testing/$releas
ever/SRPMS/
#mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=updates-testing-source-f$relea
sever&arch=$basearch
enabled=0
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-fedora-$basearch
```

步骤 4　在作为根的终端中执行 **yum makecache** 命令。

安装 GUI、关联的软件组件和镜像

要在 TFTP 服务器上安装 GUI、关联的软件组件和镜像，请运行 SMI_GUI_release_v1.3.tar.gz 发行包或更高版本的发行包中包含的 installUI.sh 这一 Linux Shell 脚本。

要通过运行 Linux Shell 脚本来安装 GUI，请按下列步骤操作：

步骤 1　从官方网站下载云翼 300 系列操作系统的最新版本。该软件包的文件名是 edge300k9-1.3.0.tar

步骤 2　将软件包复制到您要设置 GUI 所在的服务器。

步骤 3　通过输入 **su** Linux 命令切换到超级用户（根用户），然后输入您的根用户密码。

步骤 4　将目录更改至包含发布软件包 edge300k9-1.3.0.tar 的目录。

步骤 5　通过输入 **tar xvf edge300k9-13.0.tar** 提取软件包并获得 SMI_UI_release-1.3.tar.gz。

步骤 6　通过输入 **tar zxvf SMI_UI_release-1.3.tar.gz** 提取 SMI_UI_release-1.3.tar.gz。

步骤 7　通过输入 **cd SMI_GUI** Linux 命令将您的目录更改至 SMI_GUI。

步骤 8　确保系统已连接到互联网。运行 ./installUI.sh。该 GUI 安装在服务器上的 /var/www/html/smartinstall 目录中。

步骤 9　当您看到“是否要立即重新启动系统来完成安装”时，按**回车键**重新启动系统。

步骤 10　打开浏览器（确保启用了 JavaScript）并输入 http://*ip-address*/smartinstall（其中 *ip-address* 是服务器的 IP 地址），确认是否可以打开 GUI。

在运行 installUI.sh 脚本后，会从互联网自动添加 TFTP 和 HTTP 服务器软件包。通过使用以下命令，您可以将 edge300k9-1.3.0.tar 中后缀为 delivery.tar.gz 的镜像移动到 TFTP 服务器：

```
mv os-sunbird-1.3-delivery.tar.gz /opt/Tftproot/image/OS/
mv fm-os-sunbird-1.3-delivery.tar.gz /opt/Tftproot/image/FM_OS
mv fonts-sunbird-1.3-delivery.tar.gz /opt/Tftproot/image/Fonts
mv 3rd-app-sunbird-1.3-delivery.tar.gz /opt/Tftproot/image/Partner
```

注　使用 GUI 创建的指挥交换机配置文件保存在 /opt/Tftproot 目录中。

访问 GUI

可以通过 Microsoft Internet Explorer 或 Mozilla Firefox 访问 GUI。确保在浏览器上启用了 JavaScript。

要访问 GUI，请执行以下步骤：

步骤 1　打开浏览器，并输入 URL **http://***ip-address***/smartinstall**，其中 *ip-address* 是 GUI 服务器的 IP 地址。

步骤 2　输入您的用户名和密码。

默认用户名和密码均为 **cisco**。为安全起见，请更改用户名和密码（请参阅第 2-12 页上的“更改 GUI 登录凭据”一节）。

步骤 3　点击**确定**。系统随即会打开“主页”屏幕。“主页”屏幕提供了有关 GUI 的说明。

步骤 4　（可选）在屏幕右上角，从下拉列表中选择一种语言。

注　如果所选语言为简体中文或繁体中文，GUI 服务器就必须支持中文字符集。

更改 GUI 登录凭据

要更改 GUI 登录凭据，请执行以下步骤：

步骤 1　在菜单上，单击**管理信息**。屏幕上将打开“修改管理信息”屏幕。

“原来的用户名”字段显示您的现有用户名。

步骤 2　在“原来的密码”字段中，输入现有密码。

步骤 3　在“新的用户名”字段中，输入新用户名。

步骤 4　在“新的密码”和“确认新的密码”字段中，输入新的密码。

新的密码应该遵守以下规则：

- 密码应该至少包含以下类别中的三类字符：a-z、A-Z、0-9 和 !@#$%^&*()。
- 密码中的任何字符均不能连续重复三次以上。
- 密码不能为“cisco”以及任何类似形式，包括改变字母大小写，以 1、| 或 ! 代替 i、以 0 代替 o 或以 $ 代替 s。

步骤 5　点击**提交**。

注　如果忘记了密码，则可以通过运行智能安装根目录中的 reset.sh 文件，将用户名和密码重置为 *cisco*。

管理镜像文件服务器（可选）

云翼交换机的镜像和配置文件存储在镜像文件服务器中。默认情况下，镜像文件服务器就是运行 GUI 的服务器，但它也可以在单独的服务器上运行。

云翼镜像（OS、FM_OS、CiscoApp、PARTNER 和 FONTS 镜像）、镜像列表文件、指挥交换机配置文件和云翼配置文件均存储在每个站点的分布式镜像文件服务器中。

要将镜像文件服务器添加到 GUI，请执行以下操作之一：

- 将镜像文件服务器手动添加到 GUI“ 镜像文件服务器列表” 屏幕中。
- 将镜像文件服务器列表导入 GUI“ 镜像文件服务器列表” 屏幕中。
- 在 GUI 中，克隆现有的镜像文件服务器，然后编辑该镜像文件服务器。

创建镜像文件服务器

要运行一个单独的镜像文件服务器，您应该将该服务器添加到 GUI 中。要添加一个单独的镜像文件服务器，请遵循以下步骤：

步骤 1　在菜单上，选择管理 > 管理镜像文件服务器。您随即会看到“ 管理镜像文件服务器” 屏幕。

步骤 2　单击表格上面的“ 添加一个镜像文件服务器”。您随即会看到“ 添加一个镜像文件服务器” 屏幕。

步骤 3　在“ 服务器名” 字段中，输入您想添加的镜像文件服务器的名称。服务器名应该是唯一的，并且不能超过 30 个字符。

步骤 4　在“ IP 地址” 字段中，输入镜像文件服务器的有效 IPv4 或 IPv6 地址。

步骤 5　在“ 用户名” 和“ 密码” 字段中，输入镜像文件服务器的 samba 帐户信息。

步骤 6　点击添加按钮。您随即会看到“ 镜像文件服务器列表” 屏幕，并且镜像文件服务器已添加到“ 镜像文件服务器列表” 表格中。“ 镜像文件服务器列表” 表格还会显示镜像文件服务器的行编号，以及镜像文件服务器的创建日期。

“ 镜像文件服务器列表” 表格的最右列，提供用以管理镜像文件服务器的以下链接：

- **编辑 -** 打开“ 编辑镜像文件服务器” 屏幕。此屏幕包含与“ 添加一个镜像文件服务器” 屏幕相同的字段。您可以使用该屏幕对镜像文件服务器进行除更改服务器名之外的任何更改，服务器名用于识别镜像文件服务器。有关更多信息，请参阅第 2-14 页上的“ 克隆、修改和删除镜像文件服务器” 一节。
- **克隆 -** 在 GUI 中已添加任何现有镜像文件服务器的情况下，以快速模式添加镜像文件服务器。有关更多信息，请参阅第 2-14 页上的“ 克隆、修改和删除镜像文件服务器” 一节。
- **删除 -** 删除镜像文件服务器。
- **成员操作 -** 打开一个屏幕，让您可以将组分配到镜像文件服务器。有关更多信息，请参阅第 2-15 页上的“ 将组分配到镜像文件服务器” 一节。

导入镜像文件服务器列表

您可以将包含镜像文件服务器信息的 Microsoft Excel 电子表格导入 GUI。遵循如下电子表格要求：

- 电子表格可以使用任何名称，但必须使用 .csv 扩展名保存，且大小不能超过 2 MB。
- 电子表格的第一行必须是标题行，且不能包括任何镜像文件服务器信息。从第二行开始可以包含镜像文件服务器信息。
- 标题行必须包括以下标题：镜像文件服务器名、服务器 IP 地址、用户名和密码。

要将电子表格导入 GUI，请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理镜像文件服务器**。您随即会看到“ 管理镜像文件服务器” 屏幕。

步骤 2　在“ 上传电子表格” 字段右侧，单击带黑箭头的图标。

步骤 3　导航到一个电子表格文件，并按照浏览器说明将文件目录和名称填入“ 上传电子表格” 字段。

步骤 4　单击**上传**将信息上传到“ 镜像文件服务器列表” 屏幕上的表格中。

注　如果电子表格中 IP 地址格式不符合要求，或与“ 镜像文件服务器列表” 屏幕上表格中现有 IP 地址重复， GUI 会拒绝该记录，并显示一条错误消息。

克隆、修改和删除镜像文件服务器

要在 GUI 中克隆、修改或删除镜像文件服务器，请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理镜像文件服务器**。您随即会看到“ 管理镜像文件服务器” 屏幕。

“ 管理镜像文件服务器” 表格中的“ 动作” 列中提供用于从 GUI 中编辑、克隆或删除镜像文件服务器的链接。

步骤 2　采取下列操作之一：

- 要编辑镜像文件服务器，请单击“ 动作” 列中相应的**编辑**链接。您随即会看到“ 编辑镜像文件服务器” 屏幕。您可以更改“ IP 地址”、“ 用户名” 和“ 密码” 字段。在完成之后，单击**更新**。
- 要克隆镜像文件服务器行，请在“ 动作” 列中单击相应的**克隆**链接。您随即会看到“ 添加一个镜像文件服务器” 屏幕。您必须修改“ 服务器名” 和“ IP 地址” 字段。作为可选项，您可以修改“ 用户名” 和“ 密码” 字段。在完成之后，单击**添加**。
- 要从 GUI 中删除镜像文件服务器，请在“ 动作” 列中单击相应的**删除**链接。确认删除并重新加载屏幕。

使用搜索功能克隆、修改和删除镜像文件服务器

要使用搜索功能在 GUI 中克隆、修改和删除镜像文件服务器，请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理镜像文件服务器**。您随即会看到“ 管理镜像文件服务器” 屏幕。

步骤 2　单击**搜索镜像文件服务器**。您随即会看到“ 搜索镜像文件服务器” 屏幕。

步骤 3　选中复选框，指明搜索条件的类型，然后在相应字段中输入条件。

例如，选中**服务器名**复选框然后输入 **server1**，即可搜索服务器名中包含 server1 的所有镜像文件服务器。还可以选中 **IP 地址**复选框然后在相应字段中输入 IP 地址，搜索镜像文件服务器。

步骤 4　单击**按以上条件搜索**。搜索结果显示在屏幕底部的表格中。默认情况下，表格中的所有镜像文件服务器均自动处于选定（选中）状态。

步骤 5　采取下列操作之一：

- 要编辑镜像文件服务器，请单击“ 动作” 列中相应的**编辑**链接。您随即会看到“ 编辑镜像文件服务器” 屏幕。您可以更改“ IP 地址”、“ 用户名” 和“ 密码” 字段。在完成之后，单击**更新**。
- 要克隆镜像文件服务器行，请在“ 动作” 列中单击相应的**克隆**链接。您随即会看到“ 添加一个镜像文件服务器” 屏幕。您必须修改“ 服务器名” 和“ IP 地址” 字段。作为可选项，您可以修改“ 用户名” 和“ 密码” 字段。在完成之后，单击**添加**。
- 要从 GUI 中删除镜像文件服务器，请在“ 动作” 列中单击相应的**删除**链接。确认删除并重新加载屏幕。
- 要删除搜索结果中所有选定的镜像文件服务器，请单击**删除选定的镜像文件服务器**。如果不想删除所有镜像文件服务器，请取消选中那些您不想删除的镜像文件服务器的复选框。

将组分配到镜像文件服务器

可以将组分配到镜像文件服务器。每个组只能使用一个镜像文件服务器。您可以更改每个镜像文件服务器的成员（即组），步骤如下：

步骤 1　在菜单上，选择**管理 > 管理镜像文件服务器**。您随即会看到“ 管理镜像文件服务器” 屏幕。

步骤 2　对于您想要分配组的镜像文件服务器，在“ 镜像文件服务器列表” 表格的最右列（“ 动作”）中，单击**成员操作**。您随即会看到“ 分组” 屏幕。

步骤 3　在“ 不使用镜像文件服务器” 字段的组中，按键盘上的 **Crtl** 键并单击组名，选择您想分配到镜像文件服务器的组。

步骤 4　单击左尖括号 (<<) 将组移至使用 <image server name> 字段的组，或单击右尖括号 (>>) 将客户端移回至“ 不使用镜像文件服务器” 字段的组。

步骤 5　单击**提交更改**。屏幕下半部的表格显示已分配到镜像文件服务器的组的详情。

管理交换机组

为便于配置和管理，您可以对智能安装网络中的客户端交换机分组。这些组基于下列交换机组件之一：

- MAC 地址
- 产品型号 (PID)
- 位置

使用 GUI 生成智能安装组-设备关联文件。指挥交换机使用这些文件按组（而非单独）配置交换机。此文件存储在 TFTP 服务器上的 /opt/Tftproot/ 目录中，后缀为“ IBDconf”。尽管可以手动输入 MAC 地址、产品型号和位置，但也可以将包含交换机信息的电子表格导入 GUI。

注　您可以使用 CLI 按 MAC 地址或产品型号对客户端交换机分组（请参阅第 2-20 页上的“ 使用 Cisco IOS CLI 配置智能安装组” 一节）。不过，我们建议使用 GUI 对客户端交换机分组，仅当 GUI 不可用时才使用 CLI。

注　如果您更改的任何组的成员的配置已下载到指挥交换机，页面底部会显示一个“ 更新” 栏。您可以单击**更新**按钮将新成员的信息更新到指挥交换机。

创建交换机组

要创建可以向其中添加交换机的交换机组，请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理组**。您随即会看到“ 管理组” 屏幕。

步骤 2　单击表上方的**添加组**。您随即会看到“ 添加组” 屏幕。

步骤 3　在“ 组名” 字段中，输入一个有意义的名称。

步骤 4　（可选）从“ 镜像文件服务器” 下拉列表中，选择一个镜像文件服务器。

步骤 5　（可选）在“ 描述” 字段中，输入提供有关该组的详细信息的描述。

步骤 6　点击**添加**按钮。您随即会看到“ 组列表” 屏幕，该组被添加到“ 组列表” 表格中。“ 组列表” 表格中还显示了组的行 ID 和组创建的日期。

“ 组列表” 表格最右列提供了用于管理组的以下链接：

- **编辑 -** 打开“ 编辑组” 屏幕。此屏幕包含与“ 添加组” 屏幕相同的字段。您可以在这里更改镜像文件服务器和描述。
- **删除 -** 删除组。
- **成员操作 -** 打开一个屏幕，让您可以将智能安装交换机客户端添加到组，或从组中将其删除。有关更多信息，请参阅第 2-19 页上的“ 向交换机组添加成员” 一节。

管理云翼交换机列表

智能安装指挥交换机可发现交换机客户端并将它们添加到指挥交换机数据库。不过，发现的客户端交换机不显示在 GUI 中。要将客户端交换机添加到 GUI，请执行以下操作：

- 将一个客户端交换机列表导入 GUI“ 云翼列表” 屏幕。
- 将客户端交换机手动添加到 GUI“ 云翼列表” 屏幕。
- 在 GUI 中，克隆现有的客户端交换机，并编辑该客户端交换机。

导入客户端交换机列表

您可以将带有客户端交换机信息的 Microsoft Excel 电子表格或文本文件导入 GUI。遵循如下电子表格要求：

- 电子表格可以使用任何名称，但必须使用 .csv 或 .txt 扩展名保存，且大小不能超过 2 MB。文本文件还必须具有逗号分隔的值。
- 电子表格的第一行必须是标题行，且不能包括任何交换机信息。交换机信息可以从第二行开始。
- 标题行必须包含以下标题：MAC、产品型号、位置。不要包括组信息：组通过 GUI 分配。
- MAC 地址必须由六组十六进制数字构成，每组两个数字，各组用冒号分隔。例如：AA:01:BB:02:CC:03。
- 产品型号必须是字母数字字符，最多可以包含 49 个字符。

注　电子表格不应包含组信息。必须使用 GUI 将交换机分配给组。

要将电子表格导入 GUI，请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理云翼**。您随即会看到“ 管理云翼” 屏幕。

步骤 2　在“ 上传电子表格” 字段右侧，单击带黑箭头的图标。

步骤 3　导航到一个电子表格或文本文件，并按照浏览器说明将文件目录和名称放入“ 上传电子表格” 字段。

步骤 4　单击**上传**将信息上传到“ 云翼列表” 屏幕上的表格中。

注　如果电子表格或文本文件中的 MAC 地址格式不符合要求，或与“ 云翼列表” 屏幕上表格中的现有 MAC 地址重复，GUI 会拒绝该记录，并显示一条错误消息。

注　有关详细信息，请参阅附录 B，“ 导入带客户端交换机信息的电子表格”。

手动添加客户端交换机

要将客户端交换机手动添加到 GUI，请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理云翼**。您随即会看到“ 管理云翼” 屏幕。

步骤 2　单击**添加云翼**选项卡。您随即会看到“ 添加云翼” 屏幕。

步骤 3　输入以下信息：

- “ MAC 地址” 字段：输入 MAC 地址，格式为六组十六进制数字，每组两个数字，各组用冒号分隔。例如：AA:01:BB:02:CC:03。

  注　如果输入的 MAC 地址格式不符合要求或与“ 云翼列表” 屏幕上表格中已有的 MAC 地址重复， GUI 会拒绝让您进入，并显示一条错误消息。

- “ 产品型号” 字段：输入产品型号，产品型号必须是字母数字字符，并且最多可以包含 49 个字符。
- “ 位置” 字段：输入位置，它是一个对您有意义的名称。位置必须是字母数字字符，最多可以包含 49 个字符。
- “ 组名” 字段：从下拉列表中，选择交换机所属的组。如果没有已有组，管理员可以单击下拉列表右侧的“ 创建组” 链接进行创建。

  注　一个交换机只能属于一个组。

步骤 4　点击**添加**保存您的更改，并返回到“ 添加云翼” 页面。您可以继续添加另一个云翼，或点击**后退**返回到“ 云翼列表” 屏幕。

克隆、修改和删除客户端交换机

要在 GUI 中克隆、修改或删除客户端交换机，请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理云翼**。您随即会看到“ 管理云翼” 屏幕。

“ 管理云翼” 表的“ 动作” 列提供了在 GUI 中修改、克隆或删除客户端交换机的链接。

步骤 2　采取下列操作之一：

- 要修改交换机，请单击“ 动作” 列中相应的**修改**链接。您随即会看到“ 云翼列表” 屏幕。此屏幕包含与“ 添加云翼” 屏幕相同的字段。您可以更改 PID 和 LOCATION 字段，并将交换机分配给另一个组。在完成之后，单击**更新**。
- 要克隆交换机行，请在“ 动作” 列中单击相应的**克隆**链接。您随即会看到“ 添加云翼” 屏幕。必须修改 MAC 字段（两个交换机的 MAC 地址不能相同）。作为一个选项，您可以修改产品型号和位置字段，并将交换机分配给另一个组。在完成之后，单击**添加**。
- 要从 GUI 中删除交换机，请在“ 动作” 列中单击相应的**删除**链接。确认删除并重新加载屏幕。

使用搜索功能克隆、修改和删除交换机

要使用搜索功能在 GUI 中克隆、修改和删除客户端交换机，请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理云翼**。您随即会看到“ 管理云翼” 屏幕。

步骤 2　单击**搜索云翼**。您随即会看到“ 云翼列表” 屏幕。

步骤 3　选中一个复选框以指定搜索条件的类型，在相应的字段中输入条件或者单击字段中显示的条件。

例如，选中**位置**复选框可以按位置搜索。还可以选中 MAC 复选框并在相应的字段中输入 1，从而仅搜索 MAC 地址中包括 1 的交换机。

步骤 4　单击**按以上条件搜索**。搜索结果显示在屏幕底部的表格中。默认情况下，自动选择（选中）表格中的所有交换机。

步骤 5　采取下列操作之一：

- 要修改交换机，请单击“ 动作” 列中相应的**修改**链接。您随即会看到“ 云翼列表” 屏幕。此屏幕包含与“ 添加云翼” 屏幕相同的字段。您可以修改 MAC、产品型号和位置字段，并将交换机分配给另一个组。在完成之后，单击**更新**。
- 要克隆交换机行，请在“ 动作” 列中单击相应的**克隆**链接。您随即会看到“ 添加云翼” 屏幕。必须修改 SN 和 MAC 字段（两个交换机的 MAC 地址不能相同）。作为一个选项，您可以修改产品型号和位置字段，并将交换机分配给另一个组。在完成之后，单击**添加**。
- 要从 GUI 中删除交换机，请在“ 动作” 列中单击相应的**删除**。确认删除并重新加载屏幕。
- 要删除搜索结果中选中的所有交换机，请单击**删除选中的云翼**。如果不想删除所有交换机，请清除您不想删除的交换机的复选框。

向交换机组添加成员

可以使用 GUI 向交换机组添加成员或者修改交换机组中的成员。

注　您还可以使用 CLI 基于 MAC 地址或产品型号添加一组自定义交换机（请参阅第 2-20 页上的“ 使用 Cisco IOS CLI 配置智能安装组” 一节）。我们建议使用 GUI 对客户端交换机分组，仅当 GUI 不可用时才使用 CLI。

在“ 分组” 屏幕中向交换机组添加成员

要在 GUI 中向交换机组添加客户端（请参阅第 2-16 页上的“ 管理交换机组” 一节），请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理组**。您随即会看到“ 管理组” 屏幕。

步骤 2　对于要向其中添加客户端的组，在“ 组列表” 表格的最右边一列（动作）中单击**成员操作**。您随即会看到“ 分组” 屏幕。

步骤 3　在“ 未分组的云翼” 字段中，通过按键盘上的 **Crtl** 键并单击客户端名称选择要分配到该组的客户端。

步骤 4　单击左尖括号 (<<) 可将客户端移到该组字段中，单击右尖括号 (>>) 可将客户端移回“ 未分组的云翼” 字段中。

步骤 5　单击**提交更改**。屏幕下半部的表格显示已添加到该组的客户端详细信息。

使用搜索功能将成员分配到交换机组或更改交换机组的成员

要使用搜索功能将成员分配到交换机组或将成员从一个交换机组更改到另一交换机组，请执行以下步骤：

步骤 1　在菜单上，选择**管理 > 管理云翼**。您随即会看到“ 管理云翼” 屏幕。

步骤 2　单击**对云翼分组**按钮。您随即会看到“ 选择分组条件” 屏幕。

步骤 3　选中一个复选框以指定搜索条件的类型。在相应的字段中输入条件，或者单击字段中显示的条件。

例如，选中**位置**复选框可以按位置搜索。还可以选中 MAC 复选框并在相应的字段中输入 1，仅搜索 MAC 地址中包括 1 的交换机。

步骤 4　单击**按以上条件搜索**。搜索结果显示在屏幕底部的表格中。默认情况下，将选择（选中）所有交换机。

步骤 5　在“ 将选中的云翼分组到” 按钮右侧的下拉列表中，为选中的交换机选择交换机组。如果不希望重新分配某些交换机，请取消选中这些交换机的复选框。

步骤 6　单击**将选中的云翼分组到**按钮完成分配。

使用 Cisco IOS CLI 配置智能安装组

您可以使用 CLI 按 MAC 地址或产品型号对客户端交换机分组。我们建议使用 GUI 对客户端交换机分组，仅当 GUI 不可用时才使用 CLI。

注　有关使用 GUI 对客户端交换机分组的信息，请参阅第 2-16 页上的“ 创建交换机组” 一节和第 2-19 页上的“ 向交换机组添加成员” 一节。

注　云翼 300 系列交换机不支持 CLI 生成的分组文件与 GUI 生成的分组文件混用。您*只能*使用 GUI 或者*只能*使用 CLI 生成分组文件。

基于 MAC 地址自定义组

您可以基于 MAC 地址配置自定义组。 MAC 地址匹配优先于其他匹配。与组中的 MAC 地址不匹配的交换机可以获取另一个组的配置和镜像，或者获取默认配置。

从特权 EXEC 模式开始，在指挥交换机中执行以下步骤以基于 MAC 地址配置组：

	命令	目的
步骤 1	**config terminal**	进入全局配置模式。
步骤 2	**vstack group custom** *group_name* **mac**	基于 MAC 地址匹配标识自定义组，并进入该组的智能安装组配置模式。
步骤 3	**match** *mac_address*	输入要添加到自定义组的客户端交换机的 MAC 地址。对要添加的每个 MAC 地址重复此命令。 **注**　要查看智能安装网络中交换机的 MAC 地址，请输入 **show vstack neighbors all** 特权 EXEC 命令。添加到组中的交换机使用相同的镜像和配置文件。
步骤 4	**image** *location image_name*-**imglist.txt**	输入自定义组的位置和镜像列表文件。 • *location* - 如果 TFTP 服务器是指挥交换机并且文件在指挥交换机闪存中，请输入 **flash:**，或者输入 **tftp:** 和镜像的位置。还可以输入 **flash0:**、**flash1:** 或 **usb:**。 **注**　尽管可以在命令行帮助中看到以下选项，但这些选项不受支持：**flash1:**、 **ftp:**、 **http:**、**https:**、 **null:**、 **nvram:**、 **rcp:**、 **scp:**、**system:**、 **tmpsys:**。 • *image_name*-**imglist.txt** 是要下载的镜像列表文件。
步骤 5	**config** *location* **config.text.***config_filename*	输入自定义组的位置和配置文件。 • *location* - 如果 TFTP 服务器是指挥交换机并且文件在指挥交换机闪存中，请输入 **flash:**，或者输入 **tftp:** 和镜像的位置。还可以输入 **flash0:**、**flash1:** 或 **usb:**。 **注**　尽管可以在命令行帮助中看到以下选项，但这些选项不受支持：**flash1:**、 **ftp:**、 **http:**、**https:**、 **null:**、 **nvram:**、 **rcp:**、 **scp:**、**system:**、 **tmpsys:**。 • **config.text.***config_filename* - 输入组的配置文件的文件名。
步骤 6	**end**	返回特权 EXEC 模式。
步骤 7	**copy running-config startup config**	（可选）将输入保存在配置文件中。
步骤 8	**show vstack group custom detail**	验证配置。

注　指挥交换机自动为新组创建一个指挥交换机配置文件，并将其保存在 TFTP 服务器上。

下例创建一个名为 testgroup3 的自定义组，其中包括由 MAC 地址标识的三个交换机，并配置该组使用指定的镜像文件 (global-imglist.txt) 和配置文件 (config.text.classroom)。

```
Director# configure terminal
Director(config)# vstack group custom textgroup3 mac
Director(config-vstack-group)# match mac 0023.34ca.c180
Director(config-vstack-group)# match mac 001a.a1b4.ee00
Director(config-vstack-group)# match mac 00:1B:54:44:C6:00
Director(config-vstack-group)# image tftp://101.122.33.10/global-imglist.txt
Director(config-vstack-group)# config tftp://101.122.33.10/config.text.classroom
Director(config-vstack-group)# exit
Director(config)# end
```

为此组创建的指挥交换机配置文件是 testgroup3-imagelist.txt。

基于产品型号的自定义组

您可以基于产品型号 (PID) 配置自定义组。与组中的 MAC 地址不匹配的交换机可以获取另一个组的配置和镜像，或者获取默认配置。

从特权 EXEC 模式开始，在指挥交换机中执行以下步骤以基于 PID 地址配置组：

	命令	目的
步骤 1	**config terminal**	进入全局配置模式。
步骤 2	**vstack group custom** *group_name* **product-id**	基于产品型号匹配标识自定义组，并进入该组的智能安装组配置模式。
步骤 3	**match** *product-id*	在自定义组中输入客户端交换机的产品型号。
步骤 4	**image** *location image_name*-**imglist.txt**	输入自定义组的位置和镜像列表文件。 • *location* - 如果 TFTP 服务器是指挥交换机并且文件在指挥交换机闪存中，请输入 **flash:**，或者输入 **tftp:** 和镜像的位置。还可以输入 **flash0:**、**flash1:** 或 **usb:**。 **注** 尽管可以在命令行帮助中看到以下选项，但这些选项不受支持：**flash1:**、**ftp:**、**http:**、**https:**、**null:**、**nvram:**、**rcp:**、**scp:**、**system:**、**tmpsys:**。 • *image_name*-**imglist.txt** 是要下载的镜像列表文件。
步骤 5	**config** *location* **config.text.***config_filename*	输入自定义组的位置和配置文件。 • *location* - 如果 TFTP 服务器是指挥交换机并且文件在指挥交换机闪存中，请输入 **flash:**，或者输入 **tftp:** 和镜像的位置。还可以输入 **flash0:**、**flash1:** 或 **usb:**。 **注** 尽管可以在命令行帮助中看到以下选项，但这些选项不受支持：**flash1:**、**ftp:**、**http:**、**https:**、**null:**、**nvram:**、**rcp:**、**scp:**、**system:**、**tmpsys:**。 • **config.text.***config_filename* - 输入组的配置文件的文件名。

	命令	目的
步骤 6	**end**	返回特权 EXEC 模式。
步骤 7	**copy running-config startup config**	（可选）将输入保存在配置文件中。
步骤 8	**show vstack group custom detail**	验证配置。

注　指挥交换机自动为新组创建一个指挥交换机配置文件，并将其保存在 TFTP 服务器上。

下例创建一个名为 testgroup4 的自定义组，其中包括由产品型号标识的交换机，并配置该组使用指定的镜像文件 (global.imglist.txt) 和配置文件 (config.text.classroom)。

```
Director# configure terminal
Director(config)# vstack group custom testgroup4 product-id
Director(config-vstack-group)# match EDGE_300
Director(config-vstack-group)# image tftp://101.122.33.10/global-imglist.txt
Director(config-vstack-group)# config tftp://101.122.33.10/config.text.classroom
Director(config-vstack-group)# exit
Director(config)# end
```

为此组创建的指挥交换机配置文件是 testgroup4-imagelist.txt。

管理云翼配置文件

- 使用 GUI 配置组
- 使用 GUI 配置云翼
- 使用 CLI 模式配置云翼或组
- 使用 CLI 模式修改组或云翼
- 使用自动完成功能输入命令

注　在 GUI 上，客户端交换机称为云翼。

云翼配置文件

云翼配置文件是位于 TFTP 服务器上并由指挥交换机管理的客户端交换机配置文件。云翼配置文件由以下部分组成：

- 一个通用配置，该配置适用于组中的所有客户端交换机，并包括多个 GUI 字段，如配置根密码，将所有交换机设置为默认设置，以及为组中的所有交换机配置接口特征。还可以切换到 CLI 模式来配置组。
- 一个单独配置，该配置适用于单个客户端交换机，并包括仅为单个客户端交换机配置接口特征的 GUI 字段、蓝牙设置、 SSID、无线安全设置等。单独交换机由其 MAC 地址标识。还可以切换到 CLI 模式来配置云翼。

使用 GUI 配置组

要使用 GUI 配置组，请执行以下步骤：

步骤 1　在菜单上，选择**配置 > 配置组**。您随即会看到“ 配置组” 屏幕。

步骤 2　单击该组“ 动作” 列中的“ 配置” 链接。

注　如果您是第一次进行配置，则加载该页时，会将每个字段都设置为默认值。管理员可以点击**恢复默认设置**按钮来恢复默认值。

步骤 3　单击以下选项卡以配置组：

基本设置	
组名	显示组的名称。您可以更改组的名称。
根用户密码	输入组的根用户（管理员）密码。这是必填字段。
学生密码	输入组的默认用户密码。
登陆界面	启用或禁用不输入用户名和密码时对 GUI 的访问权限。
操作系统版本	从下拉列表中选择操作系统镜像。
工厂模式操作系统版本	从下拉列表中选择工厂模式操作系统镜像。
思科软件版本	从下拉列表中选择思科应用镜像。
第三方软件版本	从下拉列表中选择合作伙伴应用镜像。
字体	从下拉列表中选择字体文件。
分辨率	从下拉列表中选择视频分辨率。
蓝牙	启用或禁用蓝牙。
语言	从下拉列表中选择语言。
时区	从下拉列表中选择时区。
NTP 服务器	输入 NTP 服务器的 IP 地址。
云翼数目	显示云翼交换机数目。
WiFi	
SSID	输入 SSID 名称。
广播 SSID	启用或禁用 SSID 名称广播。
无线	启用或禁用无线电。
无线模式	从下拉列表中选择一种模式。 • 802.11b/g - 网络中的设备支持 802.11b 和 802.11g。 • 802.11b - 无线网络中的所有设备仅支持 802.11b。 • 802.11g - 无线网络中的所有设备仅支持 802.11g。 • 802.11n - 无线网络中的所有设备仅支持 802.11n。 • 802.11g/n - 网络中的设备支持 802.11g 和 802.11n。 • 802.11b/g/n - 网络中的设备支持 802.11b、 802.11g 和 802.11n。
通道	选择接入点的通道号（用于设置频率）。

传输功率	选择接入点无线电传输其无线信号的功率。
通道带宽	选择当接入点以 802.11n 模式运行时的通道带宽。
加密模式	选择加密模式。根据模式，您还需要选择加密类型并输入密钥。
Wifi > 高级	
AP 隔离	配置连接同一 SSID 的客户端的无线分离。
工作模式	配置接入点在 802.11n 模式下工作时的绿灯区或混合模式。
间隔时间	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时数据包之间的时段。
MCS	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时高吞吐量调制和编码方案 (MCS) 的速率。
RDG	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时的反向授权 (RDG)。
APSD 功能	配置接入点的 Wi-Fi 多媒体 (WMM) 省电模式。
WMM 功能	配置接入点的 Wi-Fi 多媒体 (WMM)。
Beacon 间隔	配置接入点的信标间隔。
Bg 保护	配置接入点的 CTS-to-self 保护。
信道分配	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时的通道带宽。
数据帧速率	配置接入点的传输流量指示消息 (DTIM) 间隔。
扩展信道	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时用于扩展通道或辅助通道的控制端频段。
数据包聚合	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时的聚合 MAC 服务数据单元 (A-MSDU) 数据包聚合。
短时隙	配置接入点在 802.11g 模式或 802.11g 混合模式下工作时的短碰撞槽时间。
突发传送	配置接入点的传输脉冲（Tx 脉冲）。
前文传送	配置接入点的前导信号。
IGMP 嗅探	启用或禁用互联网组管理协议 (IGMP) 嗅探。
多播 MSC	配置多播帧的高吞吐量调制和编码方案 (MCS) 的速率。
多播物理层模式	配置多播帧上的 PHY 模式。
以太网端口	
MAC 地址表超时时间	输入动态 MAC 地址在使用或更新之后仍保留在 MAC 地址表中的秒数（从 15 到 3825）。
接口 Gi1/Fe1/Fe2/Fe3/Fe4	单击接口旁边的 + 图标配置接口。
状态	启用或禁用端口。 **注**　Gi1 端口已启用，并且无法禁用。
输出队列策略	从下拉列表中选择在接口上调度的输出通信类型。
暂停	在接口上启用或禁用自动协商流控制。 **注**　此选项在 Gi1 接口上可用。
优先级	选择接口上的传入通信的 QoS 优先级。

速率限制	选择接口上的广播和未知单播通信的速率限制和速率。
速度	选择接口的速度。
双工模式	为接口选择双工模式。

NFS

注　将状态更改为“ 开启” 以输入 NFS 设置。

NFS 服务器	输入网络文件系统 (NFS) 服务器的 IP 地址。
NFS 服务器路径	输入在 NFS 服务器上导出的路径。
云翼路径	输入要在云翼 300 交换机上挂载的路径。
状态	选择“ 开启” 或“ 关闭”。

成员

显示组中云翼交换机的相关信息。

注　可以单击“ 操作” 列中的链接来配置、关闭或重启云翼交换机。

步骤 4　单击**应用更改**按钮。您随即会看到“ 应用设置” 窗口。

步骤 5　输入智能安装指挥交换机 IP 地址、用户名、 Telnet 密码和特权 EXEC 模式密码。如果您在 GUI 服务器上有不止一个接口，就会显示 GUI IP 地址字段，并且您必须选择一个连接到智能安装网络的 IP。

步骤 6　点击**应用**或**应用并重新启动**按钮。

注　当您单击“ 应用” 按钮时，配置文件会下载到指挥交换机和组中使用新配置重新启动的所有云翼交换机。组中未启动的云翼交换机将在启动时进行配置。

注　在应用了第一次的配置后，云翼 300 交换机会将其 IP 地址发送到 GUI。当 GUI 收到云翼 300 交换机的 IP 地址后，它就可以帮助清除 /apps 文件夹。如果您需要在更新镜像前清除旧的应用，这项操作会非常有用。管理员可以通过单击“ 应用设置” 窗口的**清除/apps** 复选框来清除 /apps 文件夹。“ 清除/apps” 操作仅应用于组中处于启动和运行状态的交换机。它不影响关闭或未处于组内的交换机。

使用 GUI 配置云翼

注　您必须首先配置云翼组然后配置云翼，因为云翼配置比云翼组配置的优先级高。仅当您在组配置页中点击**应用**或**应用并重新启动**按钮时，才会生成组-设备关联文件。

要使用 GUI 配置云翼，请执行以下步骤：

步骤 1　在菜单上，选择**配置 > 配置云翼**。您随即会看到“ 配置云翼” 屏幕。

步骤 2　从云翼的“ 动作” 列中单击**配置**链接。您随即会看到“ 云翼配置” 屏幕。

步骤 3　单击以下选项卡之一以配置组：

基本设置	
MAC	显示 MAC 地址。
PID	显示产品型号。
位置	显示位置。
组	显示云翼交换机所属的组。
状态	显示云翼交换机的当前状态（开启、关闭）。
IP	显示云翼交换机的 IP 地址。
根用户密码	显示组的根用户（管理员）密码。
学生密码	显示组的默认用户密码。
操作系统版本	显示操作系统镜像。
工厂模式操作系统版本	显示工厂模式操作系统镜像版本。
思科软件版本	显示思科应用镜像版本。
第三方软件版本	显示第三方软件版本。
字体	显示字体文件。
主机名	输入交换机的主机名。
登陆界面	启用或禁用不输入用户名和密码时对 GUI 的访问权限。
分辨率	从下拉列表中选择视频分辨率。
蓝牙	启用或禁用。
语言	从下拉列表中选择语言。
时区	从下拉列表中选择时区。
NTP 服务器	输入 NTP 服务器的 IP 地址。
WiFi	
SSID	输入 SSID 名称。
广播 SSID	启用或禁用 SSID 名称广播。
无线	启用或禁用无线电。
无线模式	从下拉列表中选择一种模式。 • 802.11b/g - 网络中的设备支持 802.11b 和 802.11g。 • 802.11b - 无线网络中的所有设备仅支持 802.11b。 • 802.11g - 无线网络中的所有设备仅支持 802.11g。 • 802.11n - 无线网络中的所有设备仅支持 802.11n。 • 802.11b/g/n - 网络中的设备支持 802.11b、 802.11g 和 802.11n。
通道	选择接入点的通道号（用于设置频率）。
传输功率	选择接入点无线电传输其无线信号的功率。
通道带宽	选择当接入点以 802.11n 模式运行时的通道带宽。
加密模式	选择加密模式。根据模式，您还需要选择加密类型并输入密钥。

Wifi > 高级	
AP 隔离	配置连接同一 SSID 的客户端的无线分离。
工作模式	配置接入点在 802.11n 模式下工作时的绿灯区或混合模式。
间隔时间	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时数据包之间的时段。
MCS	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时高吞吐量调制和编码方案 (MCS) 的速率。
RDG	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时的反向授权 (RDG)。
APSD 功能	配置接入点的 Wi-Fi 多媒体 (WMM) 省电模式。
WMM 功能	配置接入点的 Wi-Fi 多媒体 (WMM)。
Beacon 间隔	配置接入点的信标间隔。
Bg 保护	配置接入点的 CTS-to-self 保护。
信道分配	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时的通道带宽。
数据帧速率	配置接入点的传输流量指示消息 (DTIM) 间隔。
扩展信道	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时用于扩展通道或辅助通道的控制端频段。
数据包聚合	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时的聚合 MAC 服务数据单元 (A-MSDU) 数据包聚合。
短时隙	配置接入点在 802.11g 模式或 802.11g 混合模式下工作时的短碰撞槽时间。
突发传送	配置接入点的传输脉冲（Tx 脉冲）。
前文传送	配置接入点的前导信号。
IGMP 嗅探	启用或禁用互联网组管理协议 (IGMP) 嗅探。
多播 MSC	配置多播帧的高吞吐量调制和编码方案 (MCS) 的速率。
多播物理层模式	配置多播帧上的 PHY 模式。
以太网端口	
MAC 地址表超时时间	输入动态 MAC 地址在使用或更新之后仍保留在 MAC 地址表中的秒数（从 15 到 3825）。
接口 Gi1/Fe1/Fe2/Fe3/Fe4	单击接口旁边的 + 图标配置接口。
状态	启用或禁用端口。 无法禁用 Gi1 端口。
输出队列策略	从下拉列表中选择在接口上调度的输出通信类型。
暂停	在接口上启用或禁用自动协商流控制。 **注**　此选项在 Gi1 接口上可用。
优先级	选择接口上的传入通信的 QoS 优先级。
速率限制	选择接口上的广播和未知单播通信的速率限制和速率。
速度	选择接口的速度。
双工模式	为接口选择双工模式。

NFS	
注　将状态更改为“ 开启” 以输入 NFS 设置。	
NFS 服务器	输入网络文件系统 (NFS) 服务器的 IP 地址。
NFS 服务器路径	输入在 NFS 服务器上导出的路径。
云翼路径	输入要在云翼上挂载的路径。
状态	选择“ 开启” 或“ 关闭”。

步骤 4　单击**应用更改**按钮。您随即会看到“ 应用设置” 窗口。

步骤 5　点击**应用**或**应用并重新启动**按钮。

注　当您点击“ 应用” 按钮时，会将配置文件保存到 TFTP 服务器。配置将在交换机重启后生效。

使用 CLI 模式配置云翼或组

注　将本部分中的信息与第 4 章 ,“ 配置本地 CLI - Clish”。中介绍的 CLI 命令配合使用。

要使用 CLI 模式配置云翼或组，请执行以下步骤：

步骤 1　执行以下操作之一：

- 在菜单上，选择**配置 > 配置云翼**。您随即会看到“ 配置云翼” 屏幕。
- 在菜单上，选择**配置 > 配置组**。您随即会看到“ 配置组” 屏幕。

步骤 2　从云翼或组的“ 动作” 列中单击**配置**链接。

步骤 3　单击**切换到 CLI 模式**链接。

步骤 4　在“ 镜像选择” 窗口中，选择镜像：

- 操作系统镜像 - 从下拉列表中选择一个操作系统镜像。
- 工厂模式操作系统版本 - 从下拉列表中选择工厂模式操作系统镜像。
- 思科应用镜像 - 从下拉列表中选择一个思科应用镜像。
- 合作伙伴应用镜像 - 从下拉列表中选择一个第三方应用镜像。
- 字体 - 从下拉列表中选择字体文件。
- 指挥交换机 IP 地址 - 输入指挥交换机的 IP 地址 （必填）。
- 用户名或指挥交换机 - 输入您的用户名以访问指挥交换机 （可选）。
- 指挥交换机的 Telnet 密码 - 输入指挥交换机的 Telnet 密码 （可选）。

注　如果输入了指挥交换机的用户名，请为指挥交换机的用户名输入 Telnet 密码。否则，请输入交换机的 Telnet 登录密码。

- 特权 EXEC 模式的密码 - 输入访问特权 EXEC 模式的密码 （可选）。

步骤 5　在“ 配置文件” 字段中，输入 CLI 命令或使用自动完成功能输入 CLI 命令（参见第 2-31 页上的“ 使用自动完成功能输入命令” 一节）。有关 CLI 命令的信息，请参阅第 4 章，“ 配置本地 CLI - Clish”。

步骤 6　单击**解析配置文件并保存**。系统随即会保存文件。屏幕上会显示“ 配置文件已下载到 tftp 服务器” 的消息。如果未能保存文件，系统会显示一条错误消息。

使用 CLI 模式修改组或云翼

要使用 CLI 模式修改云翼或组，请执行以下步骤：

步骤 1　执行以下操作之一：

- 在菜单上，选择**配置 > 配置云翼**。您随即会看到“ 配置云翼” 屏幕。
- 在菜单上，选择**配置 > 配置组**。您随即会看到“ 配置组” 屏幕。

步骤 2　从云翼或组的“ 动作” 列中单击**配置**链接。

步骤 3　单击**切换到 CLI 模式**链接。

步骤 4　在“ 镜像选择” 窗口，进行以下选择：

- 操作系统镜像 - 从下拉列表中选择一个操作系统镜像。
- 工厂模式操作系统版本 - 从下拉列表中选择工厂模式操作系统镜像。
- 思科应用镜像 - 从下拉列表中选择一个思科应用镜像。
- 第三方应用镜像 - 从下拉列表中选择一个第三方应用镜像。
- 指挥交换机 IP 地址 - 输入指挥交换机的 IP 地址（必填）。
- 字体 - 从下拉列表中选择字体文件。
- 用户名或指挥交换机 - 输入您的用户名以访问指挥交换机（可选）。
- 指挥交换机的 Telnet 密码 - 输入指挥交换机的 Telnet 密码（可选）。

注　如果输入了指挥交换机的用户名，请为指挥交换机的用户名输入 Telnet 密码。否则，请输入交换机的 Telnet 登录密码。

- 特权 EXEC 模式的密码 - 输入访问特权 EXEC 模式的密码（可选）。

步骤 5　在“ 配置文件” 字段中，更改 CLI 命令或输入新的 CLI 命令。还可以使用自动完成功能输入新的 CLI 命令（请参阅第 2-31 页上的“ 使用自动完成功能输入命令” 一节）。

步骤 6　在完成之后，执行下列操作之一：

- 使用同一名称保存文件：

  单击**解析配置文件并保存**使用同一名称保存文件。系统随即会保存文件。屏幕上会显示“ 配置文件已下载到 tftp 服务器” 的消息。如果未能保存文件，系统会显示一条错误消息。

使用自动完成功能输入命令

在创建或修改云翼配置文件时，可以使用自动完成功能。该功能通过提供有效的选择可以减少命令语法错误。仅当单击**解析配置文件并保存**或**确定**时才执行语法检查。

要使用自动完成功能，请执行以下步骤：

步骤 1　在智能输入字段中（带一个磅值符 [#]），输入命令的几个首字母。可用命令将显示在智能输入字段下方。

您也可以将光标放置在一个空的智能输入栏，然后按**空格键**。自动完成功能在智能输入字段下方显示您所在的命令模式中的命令。

步骤 2　按 **Tab** 键可自动完成该命令。

您还可以点击智能输入字段下方显示的某个命令，该命令将显示在智能输入字段中。

步骤 3　按 **Enter** 键。该命令将移动到“ 配置文件” 字段中。

注　智能输入字段的提示根据您所在的命令模式相应地更改。例如，当 **configure terminal** 命令移动到“ 配置文件” 字段时，命令模式将更改为：(config)#。

下面是一个如何编辑云翼配置文件的示例：

步骤 1　在“ 配置文件” 字段中，将光标放在想要更改或者添加某个 CLI 命令的位置。

步骤 2　要进行编辑，请执行下列操作之一：

- 手动调整命令而不使用智能输入字段。您可以像在常规的文本框中那样在“ 配置文件” 字段中编辑命令。
- 在智能输入字段中输入某个命令，并按 **Enter** 键添加该命令。光标在“ 配置文件” 字段中的上一位置确定命令插入的位置：
  - 如果将光标放在某个命令行开头，新命令将在该行上方插入。
  - 如果将光标放在某个命令行中间，新命令将在光标位置右侧插入。
  - 如果将光标放在某个命令行末尾，新命令将在该行下方插入。

步骤 3　单击**解析配置文件并保存**以保存更改。系统随即会保存文件。屏幕上会显示“ 配置文件已下载到 tftp 服务器” 的消息。如果未能保存文件，系统会显示一条错误消息。

交换机镜像和配置升级

本节描述升级方法。

注意　从软件版本 1.0 升级到版本 1.1 之前，请从 GUI 中删除“ 工厂模式操作系统版本” 和“ 字体” 选择，然后应用更所作的更改。请参阅第 2-23 页上的“ 管理云翼配置文件” 一节。

注　如果升级遇到任何问题，请参阅第 D-2 页上的“ 故障排除软件升级” 一节。

升级由用户启动

在交换机机房中，用户可以通过下列方法之一启动升级：

- 按“ 重置” 按钮 - 交换机以出厂默认模式启动，连接到指挥交换机，然后下载和安装最新的镜像和配置文件。
- 关闭并打开交换机 - 交换机以正常模式启动，连接到指挥交换机，并检测是否有新的镜像和配置文件。如果有新的镜像和配置文件，交换机将在出厂默认模式下重新启动，并自动下载和安装新的镜像和配置文件。

在任一情况下，交换机都会在安装新镜像和文件之前保存现有镜像和配置文件的副本。如果安装失败，交换机将还原以前的配置。

升级由管理员启动

使用 GUI 可以重启交换机以启动升级。

步骤 1　执行以下操作之一：

- 在菜单上，选择**配置 > 配置云翼**。您随即会看到“ 配置云翼” 屏幕。
- 在菜单上，选择**监控 > 监控云翼**。您随即会看到“ 监控云翼” 屏幕。

步骤 2　在云翼的“ 操作” 列中单击**重启**链接。

注　如果云翼的状态为“ 关闭” ，则“ 操作” 链接不可用。

您可以使用 CLI 连接到交换机（例如通过 Telnet 或 Secure Shell (SSH) 连接），并重新启动交换机以启动升级。

注　不支持按需升级和计划下载。无法使用 **write erase** 和 **reload**、 **vstack download-image**、 **vstack download-config** 或 **archive download-sw** 特权 EXEC 命令从指挥交换机升级交换机。

在智能安装服务器中的 CLI 配置模式

在 GUI 上可以切换到 CLI 模式，以创建云翼配置文件。

GUI 可以生成配置文件。请勿直接编辑文件，除非您是 CLI 配置方面的专家。

有关如何在 GUI 中输入 CLI 以创建云翼配置文件的信息，请参阅第 2-23 页的“ 管理云翼配置文件”一节。

配置指南

CLI 仅使用云翼 300 系列交换机的特定命令。尽管语法与 Cisco IOS CLI 类似，但命令与 Cisco IOS 命令*并不兼容*。

使用 CLI 配置这些交换机设置：

- 基本交换机设置 - 主机名、 MAC 地址、蓝牙设置、密码、网络时间协议 (NTP) 服务器和交换机语言
- 以太网接口设置 - 状态、速度和服务质量 (QoS)
- 无线接口设置 - 状态、无线电、无线模式、通道、无线分离、传输功率、 Wi-Fi 多媒体 (WMM) 和高级无线设置
- SSID 安全设置 - 广播、身份验证和加密

遵循如下配置指南：

- 点击 Web GUI 上的此按钮可进入 CLI 编辑模式。
- 为每个交换机组创建一个云翼配置文件。此文件用于配置组中的*所有*交换机。当组中的某个交换机重新启动时，它将按云翼配置文件中定义的方式进行配置。在重新启动交换机之后，对交换机进行的任何本地更改都将丢失。
- 使用 **configure terminal** 全局命令启动云翼配置文件。使用 **exit** 全局命令结束云翼配置文件。
- 在云翼配置文件中，使用 **system identifier** *mac_address* 系统配置命令启动每个单独的交换机配置。使用 **done** 系统配置命令结束每个单独的交换机配置。

  注　我们建议在单独配置各个交换机之前，使用 **system identifier default** 系统配置命令将组中的所有交换机配置为默认设置。

- 从系统配置模式中，您可以进入以下配置模式：
  - 以太网配置模式

    使用 **interface** 系统配置命令进入此模式。使用 **exit** 全局配置命令返回系统配置模式。
  - WiFi 接口配置模式

    使用 **interface** 系统配置命令进入此模式。我们建议在配置任何无线设置之前，首先使用 **wireless-mode** WiFi 配置命令来设置 802.11 无线模式。使用 **exit** 全局配置命令返回系统配置模式。
  - SSID 配置模式

    使用 **ssid** 系统配置命令进入此模式。使用 **exit** 全局配置命令返回系统配置模式。
- 必须以小写字母输入所有命令。参数可以包括大写字母。
- 如果存在配置冲突，则使用最近的配置。在下例中，不广播 SSID：

```
    ssid NEWAP1
        broadcast ssid on
        broadcast ssid off
    exit
```

云翼配置文件示例

下面是包含两个交换机的云翼配置文件示例：一个交换机的主机名为 switch333， MAC 地址为 1111.1111.1211；另一个交换机的主机名为 switch344， MAC 地址为 1111.1111.1213。

此文件是由智能安装服务器生成的，并调度到不同的云翼交换机，因此，每台交换机仅执行**系统标识默认**配置和它自己的 **MAC 地址系统标识**配置。

```
configure terminal
system identifier default
done
system identifier 1111.1111.1211
   hostname switch333
   mac address-table aging-time 3825
   mac address-table static 1111.1111.1111 vlan 1 interface fe1 default
   interface gi1
       speed 10
   exit
   interface fe3
       speed 10
   exit
   ssid NEWAP1
   exit
done
system identifier 1111.1111.1213
   hostname switch 344
   mac address-table aging-time 3825
   mac address-table static 1111.1111.1111 vlan 1 interface cpu critical
   interface fe3
       priority normal
       output-queue-strategy wrr
       speed 10
   exit
   ssid NEWAP2
       broadcast ssid on
       encryption mode wpapsk type tkip pass-phrase better33safe990-than12sorry_
   exit
   interface bvi1
       wireless-mode 9
       radio on
       channel number 12
       ap-isolation off
       operating-mode greenfield
       channel bandwidth 20/40
       guard-interval 800
       mcs 33
       rdg on
       extension channel upper
       bg-protection on
       beacon-interval 1000
       data-beacon-rate 255
       transmit power 99
       transmit preamble auto
       short-slot on
       packet aggregation on
   exit
done
exit
```

监控云翼交换机

要监控云翼交换机，请执行以下步骤：

步骤 1　执行以下操作之一：

a. 在菜单上，选择**监控 > 监控组**。您随即会看到“ 监控组” 屏幕。

b. 单击该组的“ 操作” 列中的“ **成员**” 链接。您随即会看到成员列表。

或

在菜单上，选择**监控 > 监控云翼**。您随即会看到“ 监控云翼” 屏幕。

步骤 2　单击云翼的“ 操作” 列中的“ **详细信息**” 链接，以显示“ 云翼详细信息” 屏幕。

注　如果云翼的状态为“ 关闭” ，则“ 操作” 链接不可用。

注　为了监控交换机，用户第一次必须应用组配置，并手动重新启动云翼 300 系列交换机。之后，交换机可以获取 GUI 服务器的 IP，并将它们自己的状态报告给 GUI 服务器。

“ 云翼详细信息” 页面会显示以下信息：

系统	
状态	开启或关闭。
主机名	显示所配置的主机名。
CPU 和内存使用情况	单击**显示详细信息**按钮，可显示 CPU 和内存使用情况信息。
磁盘使用情况	显示不同文件系统上的已用和可用磁盘空间容量。
蓝牙状态	开启或关闭。
启动配置	单击**显示详细信息**按钮，可显示启动配置文件。
主机文件	显示主机文件信息。

软件版本	
操作系统版本	显示操作系统镜像。
工厂模式操作系统版本	显示工厂模式操作系统镜像。
思科软件版本	显示思科应用镜像。
第三方软件版本	显示第三方应用镜像。

网络	
IP 模式	静态或 DHCP。
IP 地址	显示交换机 IP 地址。
掩码	显示掩码。
DNS 服务器	通过单击**显示 DNS 文件**按钮显示 DNS 服务器地址。
MAC 地址	显示 MAC 地址。
Bcast	显示子网的广播地址。
网关	显示网关 IP 地址。

WiFi	
状态	开启或关闭。
SSID	显示 SSID。
通道	显示无线通道。
模式	显示接入点的 802.11 无线模式。
加密	显示接入点的身份验证和关联加密。
密钥	显示加密密钥。
接入设备	显示通过 WiFi 连接的设备。

以太网端口	
状态	启用、禁用、已连接。
速度	显示所配置的速度。
双工模式	显示所配置的双工模式。
端口统计	显示端口的接收和发送计数。
QoS	显示所有交换机端口的 QoS 信息
获知的 MAC 地址	显示获知的 MAC 地址列表。

NFS 服务器	
状态	成功或失败。
远程路径	显示 IP 地址和远程路径。
安装点	显示安装点。

配置本地 CLI - Clish

配置指南

您可以在 Clish 中配置云翼 300 系列交换机，Clish 用于本地 CLI 配置。CLI 仅使用云翼 300 系列交换机的特定命令。尽管语法与 Cisco IOS CLI 类似，但命令与 Cisco IOS 命令*并不兼容*。

使用 CLI 配置这些交换机设置：

- 基本交换机设置 - 主机名、 MAC 地址、蓝牙设置、密码、网络时间协议 (NTP) 服务器和交换机语言
- 以太网接口设置 - 状态、速度和服务质量 (QoS)
- 无线接口设置 - 状态、无线电、无线模式、通道、无线分离、传输功率、 Wi-Fi 多媒体 (WMM) 和高级无线设置
- SSID 安全设置 - 广播、身份验证和加密

遵循如下配置指南：

- 在 PC 的命令提示符窗口中输入 **ssh root@*ip-address***，然后在出现欢迎屏幕后输入密码。输入 **clish** 命令进入全局配置模式。
- 使用 **configure terminal** 全局命令启动云翼配置。使用 **exit** 全局命令结束云翼配置文件。
- 在云翼配置中，使用 **system identifier local** 系统配置命令启动每个单独的交换机配置。使用 **done** 系统配置命令结束每个单独的交换机配置。

注 使用 **system identifier local** 命令进行本地 CLI 配置。

- 从系统配置模式中，您可以进入以下配置模式：
  - 以太网配置模式

    使用 **interface** 系统配置命令进入此模式。使用 **exit** 全局配置命令返回系统配置模式。

  - WiFi 接口配置模式

    使用 **interface** 系统配置命令进入此模式。我们建议在配置任何无线设置之前，首先使用 **wireless-mode** WiFi 配置命令来设置 802.11 无线模式。使用 **exit** 全局配置命令返回系统配置模式。

  - SSID 配置模式

    使用 **ssid** 系统配置命令进入此模式。使用 **exit** 全局配置命令返回系统配置模式。

- 必须以小写字母输入所有命令。参数可以包括大写字母。
- 如果存在配置冲突，则使用最近的配置。在下例中，不广播 SSID：

```
ssid NEWAP1
    broadcast ssid on
    broadcast ssid off
exit
```

本地配置和智能安装配置之间的关系

本地配置和智能安装 (SMI) 两者都在云翼 300 系列交换机上有一个配置文件。本地配置和 SMI 两者也都有用于在云翼 300 系列交换机上执行配置文件的脚本，还有一个决定要运行哪个脚本的执行标志。默认情况下，该标志为 SMI。

如果 **show running-configuration** 是在云翼 300 系列交换机上配置的，它会显示运行配置，并显示提供运行配置的源文件。**next-reboot** 命令指定重新启动后下次要运行的配置文件。例如，如果配置了 **next reboot local** 命令，该配置文件将更改为本地配置。

在版本 1.1 和更早版本中，云翼 300 系列交换机会在系统重新启动后检查标志。如果该标志指向本地配置文件，系统会在下次重新启动时将该标志恢复为 SMI，以确保 SMI 正常运行。

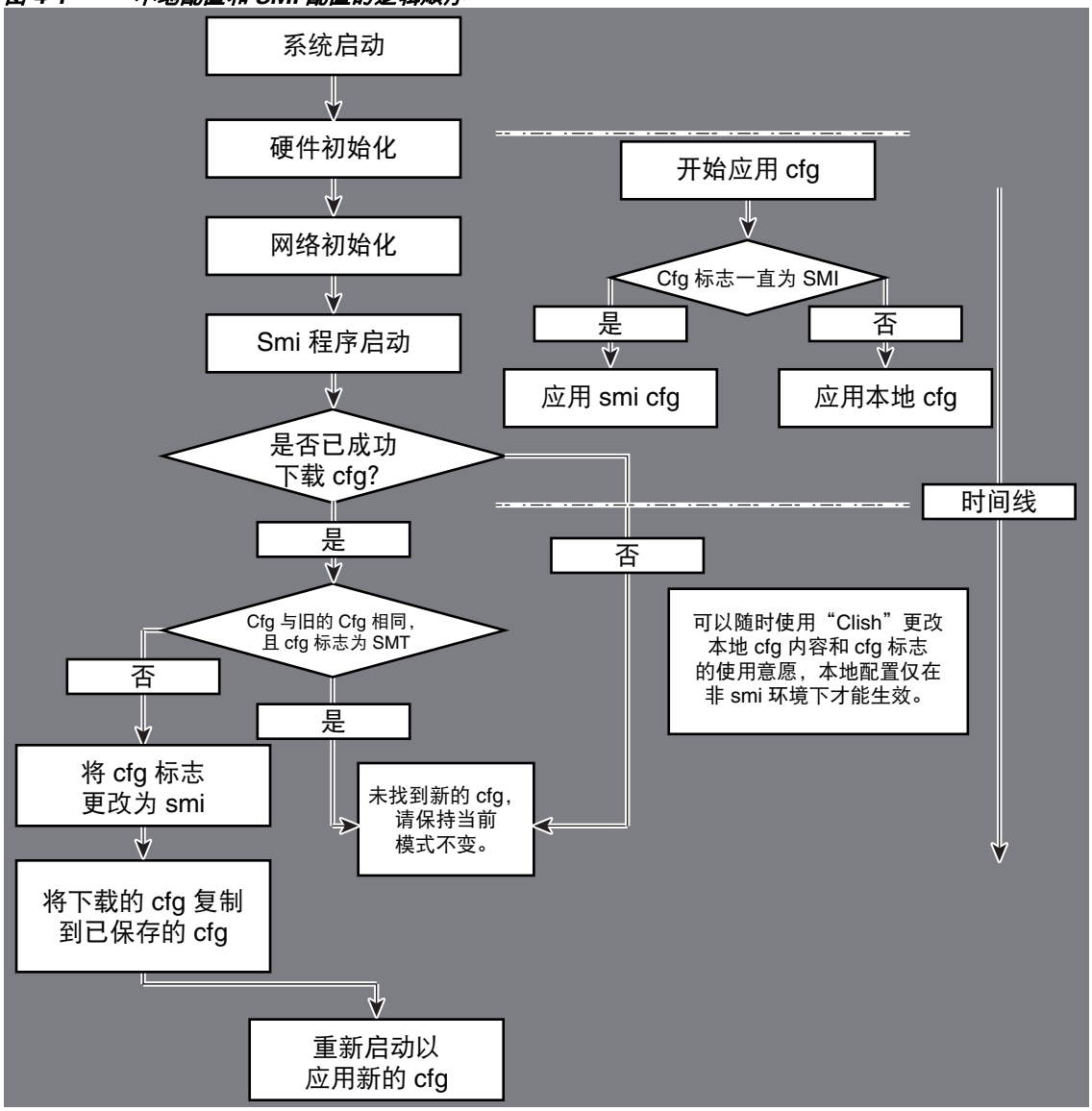
在版本 1.2 及更高版本中，云翼 300 系列交换机会根据网络状态以两种不同的方式处理本地配置：

- 如果云翼 300 系列交换机连接到 SMI 网络且被配置为应用 SMI 配置，系统会始终应用 SMI 配置而不是本地配置。
- 如果云翼 300 系列交换机连接到某个非智能安装环境，它将支持每次重新启动时将本地配置存储在 NAND 闪存中（如果没有为此特定设备设置 SMI 环境），您可以用本章中介绍的方法在其上执行本地配置，然后输入以下两条命令，从而确保下次云翼 300 系列交换机会从本地配置启动配置文件重新启动，否则，所有配置都将存储在 RAM 中，并会在重新启动后丢失。

```
> copy running-config startup-config(local)
> next-reboot local
```

图 4-1 显示本地配置和 SMI 配置的逻辑顺序。

图 4-1　本地配置和 SMI 配置的逻辑顺序



交换机命令参考

注　我们*仅*为不能自我解释的命令提供了语法描述、命令默认模式、使用指南和示例。

使能模式

表 4-1　全局配置命令

命令	功能
configure terminal	启动云翼配置文件，并进入全局配置模式。
copy running-config startup-config	将运行配置保存为启动配置文件。
exit	退出全局配置模式。
export-config	导出配置文件。
import-config	导入配置文件。
next-reboot	选择 next-reboot 模式。
reboot	关机并重新冷启动。
remove	移除本地启动配置。
show	显示运行系统信息。
wifi-mode	在下次重新启动时设置 WiFi 模式。

configure terminal

要启动云翼配置文件并进入全局配置模式，请在全局配置模式下使用 **configure terminal**。

configure terminal

使用指南　每个云翼配置文件必须使用 **configure terminal** 命令启动。

copy running-config startup-config

要将运行配置保存为启动配置文件，请在全局配置模式下使用 **copy running-config startup-config** 命令。

copy running-config startup-config

命令模式　全局配置模式

exit

要退出所处的配置模式，请在任意配置模式中使用 **exit** 命令。

exit

命令模式

全局配置
交换机配置
以太网接口配置
WiFi 接口配置
SSID 配置

使用指南

使用 **exit** 离开某个配置模式并返回到以前的配置模式。
在云翼配置文件结尾，在 **done** 系统配置命令后使用 **exit**。

export-config

要将配置文件导出到 USB 存储或本地目录，请在全局配置模式下使用 **export-config** 命令。

export-config type *type* **to** *destination*

语法说明

type	用于导出配置文件的导出类型： • 全部配置 - 一起复制启动配置、模式文件和 WiFi 客户端网络配置文件。 • Wifi 网络配置 - 一起复制启动配置和 WiFi 客户端网络配置文件。 • 启动配置 - 一起复制模式文件和启动配置本地配置文件。
destination	您要将配置文件导出到的目标。目标可以是 USB 或本地目录。

命令模式

全局配置模式

使用指南

在云翼 300 系列交换机上有三种配置文件类型：

- 启动配置 - 存储在 /etc/startup-config 中的云翼 300 系列交换机的本地配置。
- 模式文件 - 该文件用于标记启动配置是本地还是智能安装以及 WiFi 模式是 AP 还是 client。
- WiFi 客户端网络配置 - 存储在 /etc/wpa_supplicant 中。

您可以将配置文件导出到 USB 存储或本地目录。如果您选择将配置文件导出到 USB 存储，则会自动检测和挂载配置，并将其导出到外部 USB 存储。

import-config

要从 USB 存储或本地目录导入配置文件，请在全局配置模式下使用 **import-config** 命令。

import-config type *type* **from** *source*

语法说明

type	从来源导入配置文件的导入类型： • 全部配置 - 一起复制启动配置、模式文件和 WiFi 客户端网络配置文件。 • Wifi 网络配置 - 一起复制启动配置和 WiFi 客户端网络配置文件。 • 启动配置 - 一起复制模式文件和启动配置本地配置文件。
source	要导入的配置文件的位置。来源可以是 USB 或本地目录。

命令模式

全局配置模式。

使用指南

在云翼 300 系列交换机上有三种配置文件类型：

- 启动配置 - 存储在 /etc/startup-config 中的云翼 300 系列交换机的本地配置。
- 模式文件 - 该文件用于标记启动配置是本地还是智能安装以及 WiFi 模式是 AP 还是 client。
- WiFi 客户端网络配置 - 存储在 /etc/wpa_supplicant 中。

您可以从 USB 存储或本地目录导入配置文件。如果选择从 USB 存储导入配置文件，则会自动检测和挂载配置，并从外部 USB 存储导入该配置。

next-reboot

要选择 next-reboot 模式，请在全局配置模式下使用 **next-reboot** 命令。

next-reboot

命令模式　　**全局配置模式**

reboot

要暂停和执行冷启动，请在全局配置模式下使用 **reboot** 命令。

reboot

命令模式　　**全局配置模式**

remove

要移除本地启动配置，请在全局配置模式下使用 **remove** 命令。

remove

命令模式　　**全局配置模式**

show

要显示运行系统信息，请在全局配置模式下使用 **show** 命令。

show

命令模式　　**全局配置模式**

wifi-mode

要设置云翼 300 系列交换机的 WiFi 模式，请在全局配置模式下使用 **wifi-mode** 命令。

wifi-mode {ap | client}

语法说明

ap	在重新启动后将 WiFi 模式设置为 AP。
client	在重新启动后将 WiFi 模式设置为 client。

使用指南

此命令将在云翼 300 系列交换机重新启动后生效。如果选择 AP 模式，云翼 300 在重新启动后将在 AP 模式下工作，并且仅特定于 AP 模式的命令可见。如果选择 client 模式，云翼 300 在重新启动后将在 client 模式下工作，并且仅特定于 client 模式的命令可见。

wifi-mode client

要将云翼 300 系列交换机的 WiFi 模式设置为 client 模式，请在全局配置模式下使用 **wifi-mode client** 命令。

wifi-mode client

使用指南　此命令将在云翼 300 系列交换机重新启动后生效。

系统配置模式

表 4-2　系统配置命令

命令	功能
bluetooth	启用或禁用交换机上的蓝牙。
data-store	配置系统数据存储位置。
desktop resolution	配置桌面参数。
do	在全局配置模式或其他配置模式或子模式下执行用户 EXEC 或特权 EXEC 命令。
done	结束个别交换机的配置，并返回到全局配置模式。
exit	退出系统配置模式。
hostname	配置交换机的主机名。
hosts	配置交换机的 IP 地址。
interface	进入以太网接口配置模式配置快速以太网接口或千兆以太网接口，或者进入 WiFi 接口配置模式配置无线接口。
ip address	配置接口的 IP 地址。
ip default-gateway	配置默认网关。
ip name-server	配置 DNS 服务器。
language support	配置交换机的语言。
locale	配置交换机的时区。
login-window	启用或禁用登录窗口。
mac address-table aging-time	配置动态 MAC 地址在使用或更新之后仍保留在 MAC 地址表中的时段。
mac address-table static	将静态 MAC 地址添加到一个或多个接口，并设置默认 QoS 模式。
mgrvlan	配置系统使用的内部 VLAN。
no	删除命令的配置或将命令设置为默认值。
ntp server	配置交换机使用的 NTP 服务器的 IP 地址。
password	设置密码。
snmp-server	启用简单网络管理协议 (SNMP) 座席。
snmp-server community	将团体访问字符串配置为允许访问简单网络管理协议 (SNMP)。
snmp-server contact	配置系统联系人 (sysContact) 字符串。
snmp-server group	配置一个新的简单网络管理协议 (SNMP) 组，或者配置一个将 SNMP 用户映射到 SNMP 视图的表。
snmp-server location	配置系统位置字符串。
snmp-server user	将某个新用户配置为一个简单网络管理协议 (SNMP) 组。
snmp-server view	添加或更新视图条目。
snmp-server	设置 SSID 名称，并进入 SSID 配置模式配置交换机接入点的安全设置。
system identifier local	进入系统配置模式配置本地交换机。

表 4-2　系统配置命令 （续）

命令	功能
vlan	在系统中添加 VLAN。
wvlan	配置 WIFI AP 使用的无线 VLAN。

bluetooth

要启用或禁用交换机上的蓝牙，请在系统配置模式下使用 **bluetooth** 命令。

bluetooth {**on** | **off**}

命令默认　蓝牙处于开启状态。

data-store

要设置网络文件系统 (NFS) 服务器位置，请在系统配置模式下使用 **data-store** 命令。

data-store *remote_ip_addr remote_path destination_path*

语法说明

remote_ip_addr	配置 NFS 服务器的 IP 地址。
remote_path	配置目录路径。
destination_path	配置目标目录。

使用指南

请勿将服务器装载到除 **/mnt** 以外的本地系统目录。

示例

```
data-store 10.10.11.201 /var/ftp/upload /mnt
```

desktop resolution

要配置桌面上的分辨率，请在系统配置模式下使用 **desktop resolution** 命令。

desktop resolution {1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | help}

语法说明

1	1280 x 960p85
2	720p
3	1024 x 768p60
4	1080p
5	720p50
6	1080p50
7	1080i
8	1080i50
9	auto-resolution **注**　如果不支持您设置的分辨率，则会自动将该分辨率切换为 auto-resolution 模式。为了启用此新功能，我们建议您在启动系统之前连接 HDMI 监视器。
help	设置桌面分辨率，输入 1 到 9

命令默认

1024x768p60

使用指南

更改桌面分辨率需要重启。

do

要在全局配置模式或其他配置模式或子模式下执行用户 EXEC 或特权 EXEC 命令，请在任意配置模式下使用 **do** 命令。

do ***command***

语法说明

command	要执行的用户 EXEC 或特权 EXEC 命令。

命令默认

用户 EXEC 或特权 EXEC 命令不是在配置模式下执行的。

命令模式

所有配置模式。

使用指南

在配置您的路由设备时使用此命令执行用户 EXEC 或特权 EXEC 命令（如 **show**、**clear** 和 **debug** 命令）。在执行 EXEC 命令后，系统将返回到您使用过的配置模式。

done

要结束个别交换机配置，并返回到全局配置模式，请在系统配置模式下使用 **done** 命令。

done

使用指南

每个单独的交换机配置必须以 **done** 命令结束。

hostname

要配置交换机的主机名，请在系统配置模式下使用 **hostname** 命令。

hostname *name*

语法说明

name	您分配给交换机的名称。

命令默认

默认主机名为 intel_ce_linux。

使用指南

更改主机名需要重启。

hosts

要配置交换机的 IP 地址，请在系统配置模式下使用 **hosts** 命令。

hosts *ip-address*

语法说明

ip-address	标识交换机的 IP 地址。

interface

要进入以太网接口配置模式来配置快速以太网或千兆以太网接口，或者要进入 WiFi 接口配置模式来配置无线接口，请在系统配置模式下使用 **interface** 命令。

interface {fe1 | fe2 | fe3 | fe4 | gi1 | bvi1}

语法说明

fe1	配置快速以太网 1 接口。
fe2	配置快速以太网 2 接口。
fe3	配置快速以太网 3 接口。
fe4	配置快速以太网 4 接口。
gi1	配置千兆以太网接口。
bvi1	配置无线接口。

使用指南

使用 **interface** 命令进入以太网接口配置模式或 WiFi 接口配置模式。

相关命令

使用 **exit** 命令离开以太网接口配置模式或 WiFi 接口配置模式。

第 4-49 页上的表 4-3 列出了以太网接口配置命令。

第 4-58 页上的表 4-4 列出了 WiFi 接口配置命令。

ip address

要设置接口的 IP 地址，请使用 **ip address** 命令。

ip address {*dhcp* | *ip_address}*

语法说明

dhcp	通过 DHCP 协商的 IP 地址。
ip_address	接口的 IP 地址。

命令默认

默认值为 dhcp。

ip default-gateway

要指定默认网关，请使用 **ip default-gateway** 命令。

ip default-gateway *ip_address*

语法说明

ip_address	默认网关的 IP 地址。

ip name-server

要指定 DNS 服务器，请使用 **ip name-server** 命令。

ip name-server *ip_address*

语法说明

ip_address	DNS 服务器的 IP 地址。

language support

要配置交换机语言，请在系统配置模式下使用 **language support** 命令。

language support {**1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9**}

语法说明

1	英语（美国）
2	西班牙语（欧洲）
3	西班牙语（墨西哥）
4	简体中文
5	繁体中文（香港）
6	繁体中文（台湾）
7	葡萄牙文 (葡萄牙)
8	葡萄牙文 (巴西)
9	泰语

命令默认

默认语言为英语（美国）。

使用指南

更改语言需要重启。

locale

要配置时区，请在系统配置模式下使用 **locale** 命令。

locale *value*

语法说明

value	时区
0	GMT0
1	GMT+1
2	GMT+2
3	GMT+3
4	GMT+4
5	GMT+5
6	GMT+6
7	GMT+7
8	GMT+8
9	GMT+9
10	GMT+10
11	GMT+11
12	GMT+12
13	GMT-1
14	GMT-2
15	GMT-3
16	GMT-4
17	GMT-5
18	GMT-6
19	GMT-7
20	GMT-8
21	GMT-9
22	GMT-10
23	GMT-11
24	GMT-12
25	GMT+13
26	GMT+14

命令默认

默认时区为 GMT0。

login-window

要启用或禁用登录窗口，请在系统配置模式下使用 **login-window** 命令。

login-window *enable* | *disable*

语法说明

enable	启用登录窗口。
disable	禁用登录窗口。

命令默认

默认情况下会启用登录窗口。

mac address-table aging-time

要配置动态 MAC 地址在使用或更新之后仍保留在 MAC 地址表中的时段，请在系统配置模式下使用 **mac address-table aging-time** 命令。

mac address-table aging-time *aging-time*

语法说明

aging-time	动态 MAC 地址在 MAC 地址表中不可用之前经过的时段（以秒为单位）。范围是从 15 到 3825 秒。

命令默认

默认时段为 330 秒。

使用指南

如果某个 MAC 地址的数据包在有效时段内没有到达，系统会将其从 MAC 地址表中删除。如果在将 MAC 地址从表中删除之后该地址的数据包到达，系统会将数据包转发到除其到达的接口之外的所有接口。如果再次收到 MAC 地址，系统会将其添加到表中。

配置 0 秒将禁用计时器，并防止 MAC 地址从 MAC 地址表中删除。

mac address-table static

要将静态 MAC 地址添加到一个或多个 VLAN 和接口，并设置默认 QoS 模式，请在系统配置模式下使用 **mac address-table static** 命令。

mac address-table static *mac-address* **vlan *vlan id*** [**interface** *interface id]* [**default** | **critical**]

语法说明

mac_address	按 MAC 地址标识交换机，格式为 xxxx.xxxx.xxxx。
vlan *vlan-id*	为静态 MAC 地址指定 vlan。
interface *interface id*	（可选）识别静态 MAC 地址应用到的一个或多个接口。 下面是 *interface id* 参数的可能值： • fe1 - 快速以太网接口 1 • fe2 - 快速以太网接口 2 • fe3 - 快速以太网接口 3 • fe4 - 快速以太网接口 4 • gi1 - 千兆以太网接口 • cpu - 交换机的 CPU
default	（可选）配置默认 QoS 模式的接口。
critical	（可选）配置关键 QoS 模式的接口。

使用指南

为防止数据泛滥，可以将静态 MAC 地址添加到一个接口。例如，可以为连接的上行链路交换机配置静态 MAC 地址，以防数据包淹没云翼 300 系列交换机。

配置与其他接口相关的接收相对重要的信息的接口的关键 QoS。例如，为确保优质视频质量，您可以配置与监控摄像头连接的接口的关键 QoS。

示例

本示例显示如何将 1111.1111.1111 静态 MAC 地址分配给 vlan 2 fe1 接口并将 QoS 模式设置为默认值：

```
mac address-table static 1111.1111.1111 vlan 2 interface fe1 default
```

mgrvlan

要配置交换机的管理 VLAN，请在系统配置模式下使用 **mgrvlan** 命令。

mgrvlan *vlan-id*

语法说明

vlan-id	您分配给交换机作为管理 VLAN 的 VLAN ID。范围为 1 到 4094。

命令默认

vlan-id 的默认值为 1。

no

要移除命令的配置或将命令设置为默认值，请在系统配置模式下使用 **no** 命令。

no

命令模式

系统配置

SSID 配置

ntp server

要配置交换机使用的 NTP 服务器的 IP 地址，请在系统配置模式下使用 **ntp server** 命令。

ntp server *ip address*

语法说明

ip address	NTP 服务器的 IP 地址。

password

要设置根密码和学生密码，请在全局配置模式下使用 **password**。

password

命令模式　全局配置模式

snmp-server

要启用简单网络管理协议 (SNMP) 座席，请在系统配置模式下使用 **snmp-server** 命令。要禁用服务，使用此命令的 **no** 形式。

snmp-server

no snmp-server

语法说明　此命令没有任何参数或关键字。

命令默认　无。

snmp-server community

要配置团体访问字符串访问简单网络管理协议 (SNMP)，请在系统配置命令中使用 **snmp-server community**。要删除指定的团体字符串，请使用此命令的 **no** 形式。

snmp-server community *string* [**view** *view-name*] [**ro** | **rw**]

no snmp-server community *string*

语法说明

string	作用像密码且允许访问 SNMP 协议的团体字符串。
view	（可选）定义团体可用的对象。
view-name	先前定义的视图的名字。
ro	（可选）指定只读访问权限。授权管理站只能检索 MIB 对象。
rw	（可选）指定读写访问权限。授权管理站能够检索并修改 MIB 对象。

命令默认

默认情况下，SNMP 团体字符串允许对所有对象的只读访问。

snmp-server contact

要配置系统联系人 (sysContact) 字符串，请在系统配置模式下使用 **snmp-server contact**。要删除系统联系人信息，请使用此命令的 **no** 形式。

snmp-server contact *text*

no snmp-server contact

语法说明

text	描述系统联系人信息的字符串。

命令默认

未设置任何系统联系人字符串。

snmp-server group

要配置新的简单网络管理协议 (SNMP) 组或配置将 SNMP 用户映射到 SNMP 视图的表，请在系统配置模式下使用 **snmp-server group**。要移除指定的 SNMP 组，请使用此命令的 **no** 形式。

snmp-server group *groupname* **{v1 | v2c | v3 {auth | noauth | priv}} [read** *readview***] [write** *writeview***]**

no snmp-server group

语法说明

groupname	组的名称。
v1	指定可能的安全模型中最不安全的模型。
v2c	指定可能的安全模型中第二个最不安全的模型。它允许传输通知和 counter64，所允许整数是通常所允许宽度的两倍。
v3	指定可能的安全模型中最安全的模型。
auth	指定数据包的身份验证，而不对其进行加密。
noauth	指定不对数据包进行任何身份验证。
priv	指定数据包的身份验证，且进行加密。
read	指定一个读取视图。
readview	这种视图名称让您仅能查看座席内容。范围：0 到 64 个字符。
write	指定写入视图。
writeview	这种视图名称让您不仅能够输入数据，而且还能配置座席的内容。范围：0 到 64 个字符。

命令默认

无

snmp-server location

要配置 SNMP 服务器系统位置字符串，请在系统配置模式下使用 **snmp-server location**。要删除位置字符串，请使用此命令的 **no** 形式。

snmp-server location *text*

no snmp-server location

语法说明

text	描述系统位置信息的字符串。

默认值

未设置任何系统位置字符串。

snmp-server user

要将新用户配置到简单网络管理协议 (SNMP) 组，请在系统配置模式下使用 **snmp-server user**。要从 SNMP 组要移除用户，请使用命令的 **no** 形式。

snmp-server user *username groupname* **{v1 | v2c | v3} auth {md5 | sha}** *auth-password* **[priv {des | aes}** *password***]**

no snmp-server user

语法说明

username	连接到主机上的座席的用户的名称。
groupname	与用户关联的组的名称。
v1	指定可能的安全模型中最不安全的模型。
v2c	指定可能的安全模型中第二个最不安全的模型。它允许传输通知和 counter64，这是通常允许的宽度的两倍。
v3	指定可能的安全模型中最安全的模型。
auth	启动身份验证级别设置会话。
md5	指定 MD5 身份验证级别。
sha	指定 SHA 身份验证级别。
auth-password	使座席能够从主机接收数据包的字符串。范围：8 到 64 个字符。
priv	（可选）启动隐私身份验证级别设置会话。
des	（可选）使用 DES 算法进行加密。
aes	（可选）使用 AES 算法进行加密。
password	（可选）该字符串让主机能够加密它发送给座席的消息的内容。范围：8 到 64 个字符。

snmp-server view

要添加或更新视图条目，请使用 **snmp-server view** 全局配置命令。要移除指定的简单网络管理协议服务器视图条目，请使用此命令的 **no** 形式。

snmp-server view *view-name oid-tree* **{included | excluded}**

no snmp-server view *view-name*

语法说明

view-name	您更新或创建的视图记录的标签。用于引用记录的名称。
oid-tree	要包含在视图中或从视图中排除的 ASN.1 子树的对象标识符。要标识子树，请指定一个包含数字的文本字符串，如 1.3.6.2.4。
included	将视图指定为包含。
excluded	将视图指定为排除。

命令默认

无任何视图条目。

ssid

要设置 SSID 名称，并进入 SSID 配置模式配置交换机接入点的安全设置，请在系统配置模式下使用 **ssid** 命令。

ssid *ssid*

语法说明

ssid	接入点的 SSID 名称。该名称最多可以包含 32 个字符。

命令默认

默认的 SSID 名称为 CISCO_EDGE。

相关命令

使用 **exit** 命令可离开 SSID 配置模式。

第 4-82 页上的表 4-5 列出了 SSID 配置命令。

system identifier local

要将所有交换机设置为其默认设置，或者进入系统配置模式配置个别交换机，请在全局配置模式下使用 **system identifier local** 命令。

system identifier local

命令模式　全局配置模式

vlan

要在系统中添加 VLAN，请在系统配置模式下使用 **vlan** 命令。

vlan *vlan-id*

语法说明

vlan-id	分配给端口的 VLAN ID。范围：1 到 4094。并发数量应小于 6。

命令默认

所有端口的默认值为访问模式 vlan 1。

wvlan

要配置交换机的无线 VLAN，请在系统配置模式下使用 **wvlan** 命令。

wvlan *vlan-id*

语法说明

vlan-id	分配给交换机的无线 VLAN ID。

命令默认

vlan-id 的默认值为 1。

以太网接口配置模式

表 4-3　以太网接口配置命令

命令	功能
disable	禁用接口。
duplex	配置接口的双工模式。
enable	启用接口。
exit	退出以太网接口配置模式。
output-queue-strategy	配置在接口上调度的输出通信类型。
priority	配置接口上的传入通信的 QoS 优先级。
rate-limit	配置接口上的广播和未知单播通信的速率限制。
speed	配置接口的速度。
switchport mode	配置交换机的交换机端口模式。

disable

要禁用某个接口，请在以太网接口配置模式下使用 **disable** 命令。

disable {fe1 | fe2 | fe3 | fe4 | gi1}

语法说明

fe1	禁用快速以太网 1 接口。
fe2	禁用快速以太网 2 接口。
fe3	禁用快速以太网 3 接口。
fe4	禁用快速以太网 4 接口。
gi1	禁用千兆以太网接口。

默认值

启用所有接口。

相关命令

enable 命令启用接口。

duplex

要配置接口的双工模式，请在以太网配置模式下使用 **duplex** 命令。

duplex {auto | half | full}

语法说明

auto	配置自动感知双工模式。
half	配置半双工模式。
full	配置全双工模式。

默认值

默认值为自动感知双工模式。

enable

要启用接口，请在以太网接口配置模式或 WiFi 接口配置模式下使用 **enable** 命令。

enable {fe1 | fe2 | fe3 | fe4 | bvi1}

语法说明

fe1	启用快速以太网接口 1。
fe2	启用快速以太网接口 2。
fe3	启用快速以太网接口 3。
fe4	启用快速以太网接口 4。
bvi1	启用无线接口 1。

默认值

启用所有接口。

相关命令

disable 命令禁用接口。

output-queue-strategy

要配置接口上输出通信调度的类型，请在以太网配置模式下使用 **output-queue-strategy** 命令。

output-queue-strategy {strict | wrr}

语法说明

strict	基于队列优先级配置通信调度。
wrr	基于加权轮询 (WRR) 配置通信调度。

默认值

默认通信调度为 **wrr**。

priority

要配置接口上传入通信的 QoS 优先级，请在以太网接口配置模式下使用 **priority** 命令。

priority {**high** | **normal**}

语法说明

high	将传入通信配置为高优先级。
normal	将传入通信配置为正常优先级。

默认值

传入通信为正常优先级。

rate-limit

要配置接口上的广播和未知单播通信的速率限制，请在以太网接口配置模式下使用 **rate-limit** 命令。

rate-limit {none | set broadcast | set unknown-unicast | set both} *rate*

语法说明

none	禁用速率限制。
set broadcast	配置广播通信的速率限制。
set unknown-unicast	配置未知单播通信的速率限制。
set both	配置广播通信和未知单播通信的速率限制。
rate	一个介于 1 MB 和 100 MB 之间的值。

默认值

禁用速率限制。

speed

要配置接口的速度，请在以太网配置模式下使用 **speed** 命令。

speed {**auto** | **10** | **100** | **1000**}

语法说明

auto	配置自动感知速度。
10	配置 10 Mb/s 的速度。
100	配置 100 Mb/s 的速度。
1000	配置 1000 Mb/s 的速度和全双工模式。 **注**　仅在 Gi1 接口中支持 1000 Mb/s 的速度。

默认值

默认值为自动感知速度。

switchport mode

要配置交换机的交换机端口模式，请在以太网配置模式下使用 **switchport mode** 命令。

switchport mode trunk | access vlan *vlan-id*

语法说明

trunk	将交换机端口模式设置为具有特定 VLAN 的 trunkmode。 在配置了 **switchport mode trunk** 后，可以在 switchport mode trunk 模式下配置以下三个命令： • **native** *vlan_id*—Sets native VLAN ID. • **add** *vlan_id*—Adds a VLAN ID to the trunk port. • **remove** *vlan_id*—Removes VLAN ID from the trunk port VLAN list.
access	将交换机端口设置为具有特定 VLAN 的 access 模式。
vlan *vlan-id*	指定 VLAN ID。

命令默认

交换机端口的默认模式是 access。

WiFi 接口配置模式

表 4-4　WiFi 接口配置命令

命令	功能
ap-isolation	配置连接同一 SSID 的客户端的无线分离。
apsd	配置接入点的 Wi-Fi 多媒体 (WMM) 省电模式。
beacon-interval	配置接入点的信标间隔。
bg-protection	配置接入点的 CTS-to-self 保护。
channel bandwidth	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时的通道带宽。
channel number	配置接入点的通道号（用于设置频率）。
data-beacon-rate	配置接入点的传输流量指示消息 (DTIM) 间隔。
enable	启用接口。
exit	退出 WiFi 接口配置模式。
extension channel	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时用于扩展通道或辅助通道的控制端频段。
guard-interval	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时数据包之间的时段。
igmp-snoop	启用或禁用互联网组管理协议 (IGMP) 监视。
mcs	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时高吞吐量调制和编码方案 (MCS) 的速率。
multicast-mcs	配置多播帧的高吞吐量调制和编码方案 (MCS) 的速率。
multicast-phy-mode	配置多播帧上的 PHY 模式。
operating-mode	配置接入点在 802.11n 模式下工作时的绿灯区或混合模式。
packet aggregation	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时的聚合 MAC 服务数据单元 (A-MSDU) 数据包聚合。
radio	打开或关闭接入点无线电。
rdg	配置接入点在 802.11n 模式或 802.11n 混合模式下工作时的反向授权 (RDG)。
short-slot	配置接入点在 802.11g 模式或 802.11g 混合模式下工作时的短碰撞槽时间。
transmit burst	配置接入点的传输脉冲（Tx 脉冲）。
transmit preamble	配置接入点的前导信号。
transmit power	配置接入点无线电传输其无线信号的功率。
wireless-mode	配置接入点的 802.11 无线模式。
wmm	配置接入点的 Wi-Fi 多媒体 (WMM)。

ap-isolation

要为连接到同一 SSID 的客户端配置无线分离，请在 WiFi 接口配置模式下使用 **ap-isolation** 命令。

ap-isolation {on | off}

语法说明

on	启用无线分离。这避免了连接到同一 SSID 的无线客户端相互通信。
off	禁用无线分离。连接到同一 SSID 的无线客户端可以相互通信。

相关命令

WiFi 接口配置

apsd

要配置接入点的 Wi-Fi 多媒体 (WMM) 省电模式，请在 WiFi 接口配置模式下使用 **apsd** 命令。

apsd {on | off}

语法说明

on	启用 WMM 省电模式。
off	禁用 WMM 省电模式。

命令默认

禁用 WMM 省电模式。

使用指南

仅当启用 Wi-Fi 多媒体 (WMM) 时才可以配置 **apsd** 命令。

相关命令

使用 **wmm** 命令启用 WMM。

beacon-interval

要配置接入点的信标间隔，请在 WiFi 接口配置模式下使用 **beacon-interval** 命令。

beacon-interval *interval*

语法说明

interval	介于 20 至 1000 毫秒之间的时段。

命令默认

默认时段为 100 毫秒。

使用指南

默认设置应能够满足大多数网络需求。

配置一个长间隔可以：

- 提高接入点吞吐性能。
- 减少客户端的发现时间并降低漫游效率。
- 减少客户端的能耗。

配置一个短间隔可以：

- 将客户端的发现时间减至最短并提高漫游效率
- 降低接入点吞吐性能。
- 增加客户端的能耗。

bg-protection

注　此命令适用于 802.11b/g 混合模式、 802.11n/g 混合模式和 802.11b/g/n 混合模式。

要配置接入点的 CTS-to-self 保护，请在 WiFi 接口配置模式下使用 **bg-protection** 命令。

bg-protection {auto | on | off}

语法说明

auto	配置自动选择 CTS-to-self 保护。
on	启用 CTS-to-self 保护。
off	禁用 CTS-to-self 保护。

命令默认

默认值是自动选择 CTS-to-self 保护。

使用指南

CTS-to-self 保护可最大限度减少混合模式环境中客户端之间的冲突，但会降低吞吐性能。

channel bandwidth

注　此命令适用于 802.11n 模式或 802.11n 混合模式。

要配置接入点在 802.11n 模式下工作时的信道宽度，请在 WiFi 接口配置模式下使用 **channel bandwidth** 命令。

channel bandwidth {**20** | **20/40**}

语法说明

20	配置 20-MHz 通道带宽。
20/40	配置自动选择 20-MHz 或 40-MHz 通道带宽。

命令默认

默认值是自动选择 20-MHz 或 40-MHz 通道带宽。

使用指南

默认设置应能够满足大多数网络需求。

40-MHz 通道可为 802.11n 客户端提供较高的吞吐性能。

802.11b 和 802.11g 客户端只能在 20-MHz 通道下工作。

相关命令

channel bandwidth 命令的设置会影响 **mcs** 命令的选项。

channel number

要配置接入点的通道号（用于设置频率），请在 WiFi 接口配置模式下使用 **channel number** 命令。

channel number {**auto** | *number*}

语法说明

auto	配置自动选择通道号。
number	一个介于 1 到 14 之间的值，或者 0（自动选择）。

命令默认

默认通道号为 6。

使用指南

我们建议使用默认通道号或自动选择通道号，并且仅在网络中发生干扰时才更改通道号。

如果需要更改通道号，请根据您所在的位置使用以下号码：

- 中国和欧洲：1 至 13
- 美国：1 至 11
- 日本：14（仅限 11b）

data-beacon-rate

要配置接入点的传输流量指示消息 (DTIM) 的时间间隔，请在 WiFi 接口配置中使用 **data-beacon-rate** 命令。

data-beacon-rate *rate*

语法说明

rate	介于 1 和 255 毫秒之间的值。

命令默认

默认速率是 1 毫秒。

使用指南

DTIM 间隔是信标间隔的倍数。在更改 DTIM 的时间间隔之前，请考虑网络中客户端的类型：笔记本电脑使用短间隔时性能较好，而移动电话使用长间隔时性能较好。

长间隔可让客户端节省能量，但可能延迟多播和广播通信。

短间隔可以缩短多播和广播通信的传输时间，但可能增加客户端的能耗。

相关命令

beacon-interval 命令的设置会影响 **data-beacon-rate** 命令。

extension channel

注　此命令适用于 802.11n 模式或 802.11n 混合模式。

要配置接入点在 802.11n 模式下工作时用于扩展通道或辅助通道的控制边带，请在 WiFi 接口配置模式下使用 **extension channel** 命令。

extension channel {**upper** | **lower**}

语法说明

upper	配置较高扩展通道。
lower	配置较低扩展通道。

命令默认

配置较低扩展通道。

使用指南

此命令仅在配置 40-MHz 通道带宽时才生效。

当主通道号处于较低范围时（例如，在 1– 4 范围内），请使用较高扩展通道。

当主通道号处于较高范围时（例如，在 10– 13 范围内），请使用较低扩展通道。

当主通道号处于中间范围时（例如，在 5– 9 范围内），请使用较高或较低扩展通道。

相关命令

使用 **channel bandwidth** 命令配置通道带宽。

使用 **channel number** 命令配置主通道号。

guard-interval

注　此命令适用于 802.11n 模式或 802.11n 混合模式。

要配置接入点在 802.11n 模式下工作时数据包之间的时段，请在 WiFi 接口配置模式下使用 **guard-interval** 命令。

guard-interval {**400** | **800**}

语法说明

400	将短保护间隔时间配置为 400 毫微秒。
800	将长保护间隔时间配置为 800 毫微秒。

命令默认

默认值为 400 毫微秒 (ns)。

使用指南

使用 400-ns 间隔可增加 802.11n 客户端的吞吐性能，但可能存在某些数据包错误和多路径干扰的风险。

使用 800-ns 间隔可将数据包错误和多路径干扰减至最低，但会降低 802.11n 客户端吞吐性能。

相关命令

guard-interval 命令的设置影响 **mcs** 命令的选项。

igmp-snoop

要在无线接口上启用或禁用 IGMP 监视，请在 WiFi 接口配置模式下使用 **igmp-snoop** 命令。

igmp-snoop {**on** | **off**}

命令默认　IGMP 监视关闭。

mcs

注　此命令适用于 802.11n 模式或 802.11n 混合模式。

要配置接入点在 802.11n 模式下工作时高吞吐量调制和编码方案 (MCS) 的速率，请在 WiFi 接口配置模式下使用 **mcs** 命令。

mcs *index_number*

语法说明

index_number	介于 0 到 15 之间的值，或者 33（自动选择）。

命令默认

默认值为 33（自动配置速率）。

使用指南

此表根据 MCS、保护间隔时间和通道带宽显示了 MCS 索引号及其潜在数据速率 (Mb/s)。

	保护间隔时间 800 ns		保护间隔时间 400 ns	
索引号	**20-MHz 通道带宽**	**40-MHz 通道带宽**	**20-MHz 通道带宽**	**40-MHz 通道带宽**
0	6.5	13.5	7 2/9	15
1	13	27	14 4/9	30
2	19.5	40.5	21 2/3	45
3	26	54	28 8/9	60
4	39	81	43 1/3	90
5	52	109	57 5/9	120
6	58.5	121.5	65	135
11	52	108	57 7/9	120
12	78	162	86 2/3	180
13	104	216	115 5/9	240
14	117	243	130	270
15	130	270	144 4/9	300
33	配置自动选择 MCS 索引号。			

我们建议使用自动选择 MCS 索引号。仅当网络中客户端接收的信号强度指示 (RSSI) 可以支持选择的 MCS 索引号时，才将 MCS 索引更改为固定的号。

相关命令

channel bandwidth 命令的设置影响 **mcs** 命令的选项。

guard-interval 命令的设置影响 **mcs** 命令的选项。

multicast-mcs

注　此命令适用于 802.11n 模式或 802.11n 混合模式。

要配置接入点在 802.11n 模式下工作时多播帧的高吞吐量调制和编码方案 (MCS) 的速率，请在 WiFi 接口配置模式下使用 **multicast-mcs** 命令。

multicast-mcs *index_number*

语法说明

index_number	0 和 15 之间的值。

命令默认

默认值为 2。

使用指南

此表根据 MCS、保护间隔时间和通道带宽显示了 MCS 索引号及其潜在数据速率 (Mb/s)。

	保护间隔时间 800 ns		保护间隔时间 400 ns	
索引号	20-MHz 通道带宽	40-MHz 通道带宽	20-MHz 通道带宽	40-MHz 通道带宽
0	6.5	13.5	7 2/9	15
1	13	27	14 4/9	30
2	19.5	40.5	21 2/3	45
3	26	54	28 8/9	60
4	39	81	43 1/3	90
5	52	109	57 5/9	120
6	58.5	121.5	65	135
7	65	135	72 2/9	152.5
8	13	27	14 4/9	30
9	26	54	28 8/9	60
10	39	81	43 1/3	90
11	52	108	57 7/9	120
12	78	162	86 2/3	180
13	104	216	115 5/9	240
14	117	243	130	270
15	130	270	144 4/9	300

multicast-phy-mode

要配置接入点在 802.11n 模式下工作时多播帧上的 PHY 模式，请在 WiFi 接口配置模式下使用 **multicast-phy-mode** 命令。

multicast-phy-mode {**0** | **1** | **2** | **3**}

语法说明

0	指定已禁用模式。
1	指定 CCK (802.11b)。
2	指定 OFDM (802.11g)。
3	指定 HTMIX (802.11b/g/n)。

命令默认

默认值为 2。

operating-mode

注　此命令适用于 802.11n 模式。

要配置接入点在 802.11n 模式下工作时的绿地或混合模式，请在 WiFi 接口配置模式下使用 **operating-mode** 命令。

operating-mode {**greenfield** | **mixed**}

语法说明

greenfield	配置绿地模式，这可以提高 802.11n 吞吐性能，但会阻止覆盖区域的 802.11b 和 802.11g 客户端识别 802.11n 通信。
mixed	配置混合模式，这可以让覆盖区域的 802.11b 和 802.11g 客户端识别 802.11n 通信。

命令默认

默认值为混合模式。

使用指南

如果覆盖区域仅有 802.11n 客户端，请使用绿地模式。如果 802.11b、 802.11g 和 802.11n 客户端在同一覆盖区域内共存时使用绿地模式，可能会发生数据包冲突。

当 802.11b、 802.11g 和 802.11n 客户端在同一覆盖区域共存时，请使用混合模式。

packet aggregation

注　此命令适用于 802.11n 模式或 802.11n 混合模式。

要配置接入点在 802.11n 模式下工作时的聚合 MAC 服务数据单元 (A-MSDU) 数据包聚合，请在 WiFi 接口配置模式下使用 **packet aggregation** 命令。

packet aggregation {on | off}

语法说明

on	启用数据包聚合。
off	禁用数据包聚合。

命令默认

数据包聚合处于关闭状态。

使用指南

如果网络通信主要包含数据，则启用数据包聚合。

如果网络通信主要包含语音、视频或其他多媒体通信，则禁用数据包聚合。

radio

要打开或关闭接入点无线电，请在 WiFi 接口配置模式下使用 **radio** 命令。

radio {on | off}

语法说明

on	启用无线电。
off	禁用无线电。

命令默认

无线电被禁用。

使用指南

如果不打算使用接入点，请关闭无线电。如果要使用 AP 功能，请确保打开无线电。

rdg

注　此命令适用于 802.11n 模式或 802.11n 混合模式。

要配置接入点在 802.11n 模式下工作时的反向授权 (RDG)，请在 WiFi 接口配置模式下使用 **rdg** 命令。

rdg {on | off}

语法说明

on	启用 RDG。
off	禁用 RDG。

命令默认

禁用 RDG。

使用指南

在启用 RDG 时，保留通道传输机会的发射器允许接收器按保留的方向发送数据包。在禁用 RDG 时，数据包在通道传输机会保留期间只能按一个方向传输。

启用 RDG 可提高 802.11n 通信的吞吐性能。

short-slot

注　此命令适用于 802.11g 模式或 802.11g 混合模式。

要配置接入点在 802.11g 模式或 802.11g 混合模式下工作时的短碰撞槽时间，请在 WiFi 接口配置模式下使用 **short-slot** 命令。

short-slot {**on** | **off**}

语法说明

on	启用短碰撞槽时间。
off	禁用短碰撞槽时间。

命令默认

启用短碰撞槽时间。

使用指南

启用短碰撞槽时间可以提高 802.11g 客户端的吞吐性能。

如果网络中大多数为 802.11b 客户端，请禁用短碰撞槽时间。

transmit burst

要配置接入点的传输脉冲（Tx 脉冲），请在 WiFi 接口配置模式下使用 **transmit burst** 命令。

transmit burst {**on** | **off**}

语法说明

on	启用 Tx 脉冲。
off	禁用 Tx 脉冲。

命令默认

启用 Tx 脉冲。

使用指南

启用 Tx 脉冲可以提高吞吐性能。

如果发现网络中存在无线干扰，请禁用 Tx 脉冲。

transmit preamble

要配置接入点的前导信号，请在 WiFi 接口配置模式下使用 **transmit preamble** 命令。

transmit preamble {**long** | **short** | **auto**}

语法说明

long	配置长前导。
short	配置短前导。
auto	配置自动选择前导信号。

命令默认

默认值是长前导。

使用指南

使用长前导设置可以与使用 1 和 2 Mb/s 的传统 802.11 系统兼容。

配置短前导设置可以提高吞吐性能。

transmit power

要配置接入点无线电传输其无线信号的功率，请在 WiFi 接口配置模式下使用 **transmit power** 命令。

transmit power *percentage*

语法说明

percentage	1 和 100 之间的值。

命令默认

默认值为 100%。

使用指南

要远距离传输无线信号，请使用 100% 设置。

要近距离传输无线信号，例如当所有客户端都在一个小房间时，请降低该百分比。

wireless-mode

要配置接入点的 802.11 无线模式，请在 WiFi 接口配置模式下使用 **wireless-mode** 命令。

wireless-mode {**0** | **1** | **4** | **6** | **7** | **9**}

语法说明

0	配置 802.11b/g 混合模式。
1	配置 802.11b 模式。
4	配置 802.11g 模式。
6	配置 802.11n 模式。
7	配置 802.11n/g 混合模式。
9	配置 802.11b/g/n 混合模式。

命令默认

默认值为 802.11b/g/n 混合模式。

使用指南

802.11b/g 混合模式 - 如果网络中的设备支持 802.11b 和 802.11g，请选择此模式。
802.11b 模式 - 如果无线网络中的所有设备仅支持 802.11b，请选择此模式。
802.11g 模式 - 如果无线网络中的所有设备仅支持 802.11g，请选择此模式。
802.11n 模式 - 如果无线网络中的所有设备仅支持 802.11n，请选择此模式。
802.11b/g/n 混合模式 - 如果网络中的设备支持 802.11b、 802.11g 和 802.11n，请选择此模式。

wmm

要配置接入点的 Wi-Fi 多媒体 (WMM)，请在 WiFi 接口配置模式下使用 **wmm** 命令。

wmm {**on** | **off**}

语法说明

on	启用 WMM。
off	禁用 WMM。

命令默认

禁用 WMM。

使用指南

WMM 为无线通信提供 QoS。如果存在许多混合媒体通信（语音、视频、数据），请启用 WMM。

相关命令

使用 **apsd** 命令可配置 WMM 省电模式。

SSID 配置模式

要进入 SSID 模式，请执行以下步骤：

```
configure terminal
system identifier local
ssid test
```

表 4-5　SSID 配置命令

命令	功能
broadcast ssid	启用或禁用 SSID 名称广播。
encryption mode (open, shared, or WEP configuration)	配置接入点的开放式共享 Wi-Fi 保护接入 (WPA)、WPA1WPA2、 WPA2、 WPA2PSK、 WPAPSK、WPAPSKWPA2PSK 身份验证和关联加密。
encryption mode （ WPA 配置 ）	
exit	退出 SSID 配置模式。
no	删除命令的配置或将命令设置为默认值。
radius-server	配置 RADIUS 服务器的名称。

注　SSID 的配置将在退出 SSID 配置模式后生效。

broadcast ssid

要启用或禁用 SSID 名称的广播，请在 SSID 配置模式下使用 **broadcast ssid** 命令。

broadcast ssid {**on** | **off**}

语法说明

on	启用 SSID 名称的广播。
off	禁用 SSID 名称的广播。

命令默认

广播 SSID。

使用指南

禁用 SSID 广播以增强安全性。只有知道 SSID 的无线客户端可以连接到接入点。

启用 SSID 广播可提供更广泛的可用性和更方便的接入。

encryption mode (open, shared, or WEP configuration)

要配置接入点的开放、共享或有线等效加密 (WEP) 身份验证和关联加密，请在 SSID 配置模式下使用 **encryption mode** 命令。

encryption mode {**open** | **shared**} **type** {**none** | **wep** {**key** {**1** | **2** | **3** | **4**} {**hex** *number* | **ascii** *phrase*}}}

语法说明

open	配置无需身份验证的开放接入。
shared	配置带有共享密钥的身份验证。
none	配置无加密。
wep	配置 WEP 加密。
key 1 **key 2** **key 3** **key 4**	配置 WEP 加密的密钥号。 (只能使用四个密钥之一)
hex *number*	配置带有十六进制密钥的身份验证或带有十六进制密钥的身份验证和加密： • 在选择 **none** 关键字时，请配置带有十六进制密钥的身份验证。 • 在选择 **wep** 关键字时，请配置带有十六进制密钥的身份验证和加密。 对于 *number*，请输入 10 或 26 个十六进制数字。
ascii *phrase*	配置带有密码的身份验证或带有密码的身份验证和加密： • 在选择 **none** 关键字时，请配置带有密码的身份验证。 • 在选择 **wep** 关键字时，请配置带有密码的身份验证和加密。 对于 *phrase*，输入 5 或 13 个字母数字字符。支持短划线 (-) 和下划线 (_) 字符。

命令默认

默认值为开放接入和无加密。

使用指南

对于无加密的共享接入，WEP 十六进制数或密码仅用于身份验证。

对于使用 WEP 加密的共享接入，WEP 十六进制数或密码同时用于身份验证和加密。

示例

下例显示如何使用 key 3 和密码 3uifsfis-_0r5 配置共享身份验证和 WEP 加密：

```
encryption mode shared type wep key 3 ascii 3uifsfis-_0r5
```

encryption mode （ WPA 配置 ）

要配置接入点的 Wi-Fi 保护接入 (WPA) 身份验证和关联加密，请在 SSID 配置模式下使用 **encryption mode** 命令。

encryption mode {**wpapsk** | **wpa2psk** | **wpapskwpa2psk**} **type** {**tkip** | **aes** | **tkipaes**} **pass-phrase** *phrase*

语法说明

wpapsk	配置带有预共享密钥 (PSK) 身份验证的 WPA。
wpa2psk	配置带有 PSK 身份验证的 WPA2。
wpapskwpa2psk	配置带有 PSK 身份验证的合并 WPA 和 WPA2。
tkip	配置临时密钥完整性协议 (TKIP) 加密。
aes	配置高级加密标准 (AES) 的加密。
tkipaes	配置合并 TKIP 和 AES 加密。
pass-phrase *phrase*	配置密码。对于 *phrase*，输入至少 8 个至多 63 个字母数字字符。支持短划线 (-) 和下划线 (_) 字符。

命令默认

默认值为开放接入和无加密。

示例

下例显示如何使用密码 safE478_Ty33Yep- 配置带有合并 TKIP 和 AES 加密的合并 WPA 和 WPA2 身份验证：

```
encryption mode wpapskwpa2psk type tkipaes pass-phrase safE478_Ty33Yep-
```

加密模式 (802.1x)

要配置接入点的 Wi-Fi 保护接入 (WPA) 身份验证和关联加密，请在 SSID 配置模式下使用 **encryption mode** 命令。

注　加密模式 (802.1x) 应该与 RADIUS 服务器一起使用。

encryption mode {**wpa** | **wpa2** | **wpa1wpa2**} **type** {**tkip** | **aes** | **tkipaes**}

语法说明

wpa	配置带有 802.1x 身份验证的 WPA。
wpa2	配置带有 802.1x 身份验证的 WPA2。
wpa1wpa2	配置带有 802.1x 身份验证的合并 WPA 和 WPA2。
tkip	配置临时密钥完整性协议 (TKIP) 加密。
aes	配置高级加密标准 (AES) 的加密。
tkipaes	配置合并 TKIP 和 AES 加密。

命令默认

默认模式是 wpa2psk 接入、 tkipaes 加密，密码为 Cisco123。

示例

下例显示如何使用 802.1x 身份验证方法配置带有合并 TKIP 和 AES 加密的合并 WPA 和 WPA2 身份验证：

```
encryption mode wpa1wpa2 type tkipaes
```

radius-server

要配置 radius-server 的相关信息，请在 SSID 配置模式下使用 **radius-server**。

radius-server host *hostname* [**auth-port** ***port_number***] [**key** *secret*]

语法说明

hostname	Radius 服务器的 IP 地址。
auth-port	指定 Radius 服务器的身份验证端口号。
port_number	Radius 服务器的身份验证端口号。
key	指定 Radius 服务器上的身份验证服务的密码。
secret	Radius 服务器上的身份验证服务的密码。

命令默认

port_number 的默认值是 1812。

secret 的默认值为空。

示例

下例显示如何配置 Radius 服务器的相关信息：

```
radius-server host 192.168.1.1 auth-port 1812 key pass1234
```

Show 命令

您可以在全局配置模式下使用以下 **show** 命令显示云翼 300 系列交换机上的配置：

- **show 3rd-party-software-version**：显示第三方软件版本。
- **show bluetooth**：显示蓝牙状态。
- **show channel**：显示 AP 无线信道设置。
- **show cisco-software-version**：显示思科软件版本。
- **show cpu**：显示 CPU。
- **show desktop-resolution**：显示桌面分辨率信息。
- **show dhcp**：显示 DHCP 信息。
- **show disk**：显示空间使用。
- **show dns**：显示 DNS 信息。
- **show factory-mode-os-version**：显示工厂模式操作系统版本。
- **show hdmi-display-info**：显示当前已连接的 HDMI 接收器信息。
- **show hostname**：显示主机名。
- **show interfaces**：显示接口状态和配置。
- **show ip**：显示 IP 信息。
- **show mac**：显示 MAC 表信息。
- **show memory**：显示内存使用情况。
- **show nfs**：显示 NFS 安装状态。
- **show os-version**：显示正常模式操作系统版本。
- **show port-statistics**：显示交换机端口统计信息。
- **show port-status**：显示交换机端口状态。
- **show qos**：显示当前的 QoS 配置。
- **show running-config**：显示当前操作配置。
- **show snmp**：显示 SNMP 通信的状态。
- **show snmp group**：显示路由器上的组的名称、安全模型、不同视图的状态以及每个组的存储类型。
- **show snmp user**：显示组用户名表中每个简单网络管理协议 (SNMP) 用户名的信息。
- **show snmp view**：显示系列名称、存储类型、简单网络管理协议 (SNMP) 配置和相关 MIB 的状态。
- **show ssid**：显示 AP 无线 ssid 设置。
- **show startup-config**：显示启动配置的内容。
- **show USB**：显示 USB 设备信息。
- **show vlan**：显示 VLAN 配置。
- **show vstack config**：显示智能安装 VLAN 配置。
- **show wifi-client-status**：显示 WiFi 客户端状态（仅用于 WiFi 客户端模式）。
- **show wireless-clients**：显示关联的 AP 无线 wireless-clients。

- **show wireless-clients-number**：显示关联的无线客户端数量。
- **show wireless-mode**：显示 AP 无线 wireless-mode 设置。

配置 Web GUI

基于 Web 的 GUI 用于配置云翼 300 系列交换机并在本地或远程监控状态。 Web GUI 配置的实施不会在智能安装配置和本地配置之间引发任何冲突。

在智能安装 (SMI) 环境中， Web GUI 仅监控配置状态。配置是从 SMI 的 running-config 文件中检索到的。在非 SMI 环境中，您可以通过 Web GUI 配置和监控云翼 300。然后 Web GUI 会为本地 Clish 生成一个 Clish 配置文件以执行。本地 Clish 记录已完成的配置并向 Web GUI 提供反馈。

要使用基于 Web 的 GUI 配置云翼 300 系列交换机，请执行以下步骤：

登录

您可以在网页 **https://**[*云翼 300 的 IP 地址*] 上访问基于 Web 的 GUI，并通过输入**根**密码在本地或远程登录到 Web 门户。

图 5-1　　“ 登录”页

选择**记住密码**选项，以便下次访问网站时，可以直接进入相关页面。

注　如果您通过 Internet Explorer 访问 Web GUI 并启用记住密码功能，请确保云翼 300 系列交换机上的日期和时间是正确的。存储的 Cookie 用于将来的身份验证。如果云翼 300 的日期和时间不正确，身份验证将失败，且记住密码功能不会生效。

欢迎

在登录到 Web GUI 后，您将看到“ 欢迎” 页面。

图 5-2　　欢迎页面

“ 欢迎” 页面显示云翼 300 系列交换机的简介及其特性和功能。

基本配置

在“ 基本信息” 选项卡中，您可以配置云翼 300 系列交换机的基本信息，导入和导出配置文件，以及配置 IP 地址。“ 基本信息” 选项卡包括以下三个部分：

- 基本信息
- 导入和导出配置文件
- IP 配置

基本信息

在“ 基本信息” 部分中，您可以配置主机名、登陆 GUI、分辨率、无线模式、蓝牙、语言、本地化、 NTP 服务器和日志大小。

图 5-3　　基本信息配置

当云翼 300 系列交换机处于 WiFi AP 模式时，点击**切换到客户端模式**按钮会切换到 WiFi 客户端模式。如果处于 WiFi 客户端模式下，请点击**切换到 AP 模式**按钮会切换到 AP 模式。

在语言下拉列表中的语言选项与 **language support** Clish 命令的下列值相关：

- 1=en_US.utf8, English (US)
- 3=es_MX.utf8, Spanish (Mexico)
- 4=zh_CN.utf8, Simplified Chinese
- 6=zh_TW.utf8, Traditional Chinese (TW)
- 7=pt_PT.utf, Portuguese (PT)
- 9=th_TH.utf8, Thai

在应用了对分辨率、主机名或语言的修改之后，或者在导入某个配置文件之后，都需要重新启动交换机。在您已更改过的项目旁边将出现红色标示。

图 5-4　　基本信息更改的重新启动标示

当您点击“ 基本信息” 选项卡右下方的**应用**按钮后，会显示一个提示窗口，可以在该窗口中选择是否要立即重新启动交换机。

对于其他字段，点击**应用**按钮会直接应用对交换机的更改。将显示“ 已成功应用更改” 提示消息以确认更改。

注　点击**重置**会将“ 基本信息” 选项卡上的所有字段重新设置为工厂默认值。

导入和导出配置文件

在云翼 300 系列交换机上有三种配置文件类型：

- 启动配置 - 存储在 /etc/startup-config 中的云翼 300 系列交换机的本地配置。
- 模式文件 - 该文件用于标记启动配置是本地还是智能安装以及 WiFi 模式是 AP 还是 client。
- WiFi 客户端网络配置 - 存储在 /etc/wpa_supplicant 中。

您可以在“ 基本信息” 选项卡中为云翼 300 系列交换机导入或导出配置文件。

图 5-5　　导入和导出

导入配置文件

您可以从 USB 存储或本地目录导入配置文件。如果选择从 USB 存储导入配置文件，则会自动检测和挂载配置，并从外部 USB 存储导入该配置。

注　已导入的启动配置或全部配置需要重新启动才能生效。如果云翼 300 系列交换机处于 WiFi 客户端模式下，导入的 WiFi 客户端网络将立即生效。

要导入配置文件，请执行以下步骤：

步骤 1　选择以下导入类型之一：

- 全部配置 - 一起复制全部三个配置文件。
- Wifi 网络配置 - 复制启动配置和 WiFi 客户端网络配置。
- 启动配置 - 一起复制模式文件和启动配置本地配置。

步骤 2　在“ 导入路径” 字段中输入配置文件的路径。

步骤 3　点击**导入**以导入配置文件。

注　如果在“ 导入路径” 为空时点击**导入**按钮，会显示一条警告消息。

导出配置文件

您可以将配置文件导出到 USB 存储或本地目录。如果您选择将配置文件导出到 USB 存储，则会自动检测和挂载配置，并将其导出到外部 USB 存储。

要导出配置文件，请执行以下步骤：

步骤 1　选择以下导出类型之一：

- 全部配置 - 一起复制全部三个配置文件。
- Wifi 网络配置 - 复制启动配置和 WiFi 客户端网络配置。
- 启动配置 - 一起复制模式文件和启动配置本地配置。

步骤 2　在**导出路径**字段中输入您要将配置文件导出到的目标。

步骤 3　点击**导出**按钮导出配置文件。

注　如果在**导出路径**字段为空时点击**导出**按钮，会显示一条警告消息。

IP 配置

您可以在“ 基本信息” 选项卡的“ IP 配置” 部分中选择 DHCP 或静态模式。默认情况下，云翼 300 系列交换机会使用 DHCP 来获取 IP 地址。

图 5-6　　IP 配置

注　如果选择 DHCP 作为 IPv4 连接类型，IPv4 地址、IPv4 掩码和 IPv4 默认网关字段会被禁用并显示为灰色。仅可以为 DHCP 模式配置 DNS 服务器。

配置静态 IP 地址

要配置静态 IP 地址，请执行以下步骤：

步骤 1　从“ IPv4 连接类别” 下拉列表中，选择**静态**。

步骤 2　输入 IPv4 地址、 IPv4 掩码和 IPv4 默认网关。

注　如果输入的值无效，系统会显示一条警告消息。

步骤 3　输入 DNS 服务器信息。

步骤 4　点击**应用**以应用更改。

注　点击“ 重置” 会将“ 基本信息” 选项卡中的所有字段重新设置为工厂默认值。

注　如果您通过使用 Web GUI 更改云翼 300 系列交换机的 IP 地址，您需要在浏览器的地址栏中输入新的 IP 地址。否则，配置和监控功能将无法使用，因为初始 IP 地址不再存在。

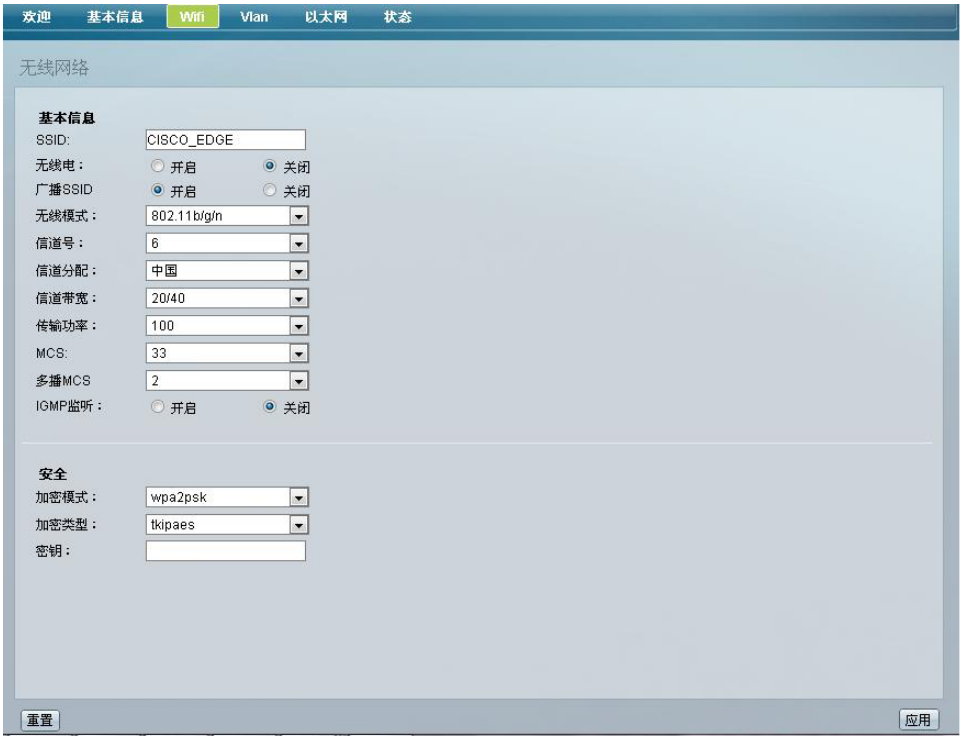
WiFi AP 配置

您可以在“ WiFi”选项卡中配置 SSID 名称、无线电、广播 SSID、无线模式、信道号、信道分配、信道带宽、传输功率、 MCS、多播 MCS、 IGMP 监听和安全。

注　当云翼 300 系列交换机处于 AP 模式时，仅可以看到 WiFi AP 配置。

图 5-7 显示“ WiFi”选项卡。

图 5-7　WiFi AP 配置



VLAN 配置

您可以点击 VLAN 列右侧的齿轮图标来添加或删除 VLAN。您还可以在“ 模式” 字段中选择 Gi1、fe1、fe2、fe3 和 fe4 端口类型。当端口类型为 trunk 时，您可以表右侧的选择列表中选择一个本征 VLAN。

图 5-8 显示“ Vlan” 选项卡。

图 5-8　　Vlan 配置

建议您保持选中**在 gi1 上使用未打标签的 cpu vlan**，否则系统可能因错误配置造成网络中断。

注　当云翼 300 系列交换机处于 WiFi 客户端模式时，是看不到 CPU 和 WiFi 接口的。

以太网配置

您可以在“ 以太网”选项卡中配置状态（用于下行链路）、输出队列策略、暂停（用于下行链路）、优先级、速率限制、速度、双工、启用和禁用。

图 5-9　以太网配置

监控状态

您可以在“ 状态”选项卡中监控系统、版本、网络、 WiFi、以太网端口、其他设备、日志、系统状态和以太网/WiFi/蓝牙/USB 状态。（请参阅图 5-10、图 5-11 和图 5-12。）

图 5-10　监控状态 - 系统、版本信息和网络

图 5-11　监控状态 - WiFi、以太网端口和其他设备

图 5-12　监控状态 - 日志

配置 HTTP API

通过使用 HTTP API，可以在本地或以远程方式在云翼 300 系列交换机上运行应用来管理交换机。交换机的管理包括配置交换机、监控状态以及安装和升级软件。

从版本 1.5 开始就支持对云翼 300 系列交换机进行 HTTP API 配置。本章介绍每个 HTTP API，包括请求、回复、参数限制和错误代码。

注 Curl 用作请求 API 的一个示例工具。 10.140.44.134 用作云翼 300 系列交换机的 IP 地址的一个示例，而 cisco123! 用作 一个管理员密码示例。

要通过使用 HTTP API 配置云翼 300 系列交换机，请参阅以下各节：

系统 API

使用此节中的命令可配置系统 API。

注　Curl 用作请求 API 的一个示例工具。 10.140.44.134 用作云翼 300 系列交换机的 IP 地址的一个示例，而 cisco123! 用作 一个管理员密码示例。

设置主机名

示例：将主机名设置为 cisco

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"hostname" : "cisco"}' https://10.140.44.134/api/1.0/sys/hostname
```

回复

```
{"success":"true","updatedAt":"2012-11-06 17:39:47"}
```

参数限制

主机名的长度应该为 1 到 64，并且有效参数集为 {a-zA-Z0-9-_}，否则系统会报告 004 错误。

获取主机名

示例：获取主机名

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/hostname
```

回复

```
{"hostname":"cisco","success":"true","getAt":"2012-11-06 17:44:37"}
```

设置日志大小

示例：将日志大小设置为 20 MB

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"size":"20"}'https://10.140.44.134/api/1.0/sys/log/size
```

回复

```
{"success":"true","updatedAt":"2012-11-06 17:39:47"}
```

参数限制

日志大小参数应为整数（范围为 1 至 100），否则系统会报告 004 错误。

获取日志大小

示例：获取日志大小

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/log/size
```

回复

```
{"size":"20","success":"true","getAt":"2011-04-21 04:33:55"}
```

删除日志

示例：删除所有日志

请求

```
curl -k -X DELETE -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/log
```

回复

```
{"success":"true","updatedAt":"2012-12-27 08:24:08"}
```

设置帐户

示例：将管理员帐户密码从 cisco123! 更改 为 cisco123

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"account"
: "cisco123"}' https://10.140.44.134/api/1.0/sys/account
```

回复

```
{"success":"true","updatedAt":"2012-11-06 17:48:06"}
```

参数限制

密码不能为空，并且有效参数集为 {a-zA-Z0-9~!@#$%^&*+=-_}，否则系统会报告 004 错误。密码应遵循 BusyBox Linux 密码要求，否则系统会报告 005 错误。

获取帐户

如果违反，系统会报告 003 错误。

设置 LoginGui

示例：将 LoginGui 设置为 enable

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"loginGui" : "enable"}' https://10.140.44.134/api/1.0/sys/loginGui
```

回复

```
{"success":"true","updatedAt":"2012-11-06 17:39:47"}
```

参数限制

LoginGui 的有效字符串：enable 和 disable。否则系统会报告 004 错误。

获取 LoginGui

示例：获取登陆界面

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/loginGui
```

回复

```
{"loginGui":"enable","success":"true","getAt":"2012-11-06 19:17:41"}
```

设置分辨率

示例：将分辨率设置为 1080p

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"resolution" : "4"}' https://10.140.44.134/api/1.0/sys/resolution
```

回复

```
{"success":"true","updatedAt":"2012-11-06 17:39:47"}
```

参数限制

有效参数集为数字 1-9。否则系统会报告 004 错误。

获取分辨率

示例：获取分辨率

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/resolution
```

回复

```
{"resolution":"9","success":"true","getAt":"2012-12-14 08:55:27"}
```

获取 Hdmi 信息

示例：获取 hdmi 信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/hdmi
```

回复

```
{"hdmi":" - Manufacturer ID: 0x294d\n - Product code : 0x9135\n - Sink name : HDMI TV\n -
Sink size (WxH) : 80 x 45\nCurrent working mode:\n 1920x1080p@59.94\nSupported modes:\n -
720x480p60 \n - 1024x768p60 \n - 1920x1080p60 \n - 1280x720p60 \n - 1920x1080p50 \n -
1280x720p50 \n - 1280x960p85 \n - 720x480p59.94 \n - 1024x768p59.94 \n - 1920x1080p59.94
\n - 1280x720p59.94 \n\n","success":"true","getAt":"2012-12-14 08:41:33"}
```

设置蓝牙

示例：将蓝牙设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"bluetooth" : "on"}' https://10.140.44.134/api/1.0/sys/bluetooth
```

回复

```
{"success":"true","updatedAt":"2012-11-06 17:39:47"}
```

参数限制

有效参数为开启/关闭。否则系统会报告 004 错误。

获取蓝牙

示例：获取蓝牙

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/bluetooth
```

回复

```
{"bluetooth":"on","success":"true","getAt":"2012-11-06 19:43:08"}
```

设置语言

示例：将语言设置为 1

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"language" : "1"}' https://10.140.44.134/api/1.0/sys/language
```

回复

```
{"success":"true","updatedAt":"2012-11-06 19:49:21"}
```

参数限制

有效参数集为数字 1-9。否则系统会报告 004 错误。

获取语言

示例：获取语言

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/language
```

回复

```
{"language":"1","success":"true","getAt":"2012-11-06 19:49:24"}
```

设置本地化

示例：将本地化设置为 8

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"locale"
: "8"}' https://10.140.44.134/api/1.0/sys/locale
```

回复

```
{"success":"true","updatedAt":"2012-11-06 19:49:21"}
```

参数限制

有效参数集为数字 0-26。否则系统会报告 004 错误。

获取本地化

示例：获取本地化

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/locale
```

回复

```
{"locale":"9","success":"true","getAt":"2012-11-06 20:13:31"}
```

设置 ntpServer

示例：将 NTP 服务器设置为 202.120.2.101

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"ntpServer" : "ntp.sjtu.edu.cn"}' https://10.140.44.134/api/1.0/sys/ntpServer
```

回复

```
{"success":"true","updatedAt":"2012-11-06 19:49:21"}
```

参数限制

参数应该是一个有效的 IPv4 地址或有效的域名。

获取 ntpServer

示例：获取 ntPServer

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/ntpServer
```

回复

```
{"ntpServer":"ntp.sjtu.edu.cn","success":"true","getAt":"2012-12-27 08:04:34"}
```

设置时间

示例：设置时间

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"time" :
"2012-11-06 17:20:20"}' https://10.140.44.134/api/1.0/sys/time
```

回复

```
{"success":"true","updatedAt":"2012-11-06 19:49:21"}
```

参数限制

参数应采用 YYYY-MM-DD HH:MM:SS 格式。否则系统会报告 004 错误。

获取时间

示例：获取时间

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/time
```

回复

```
{"time":"2012-11-07 15:22:15","success":"true","getAt":"2012-11-07 06:22:15"}
```

设置 CPU

不适用。如果做出该请求，则系统会报告 003 错误。

获取 CPU

示例：获取 CPU

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/cpu
```

回复

```
{"cpu":"CPU:   9% usr   0% sys   0% nic  90% idle   0% io   0% irq   0%
sirq","success":"true","getAt":"2012-11-07 06:35:06"}
```

设置内存

不适用。如果做出该请求，则系统会报告 003 错误。

获取内存

示例：获取内存

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/memory
```

回复

```
{"memory":"Mem: 378112K used, 1310780K free, 0K shrd, 57188K buff, 155920K
cached","success":"true","getAt":"2012-11-07 08:08:07"}
```

设置进程

不适用。如果做出该请求，则系统会报告 003 错误。

获取进程

示例：获取进程

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/proc
```

回复

```
{"proc":"UID        PID  PPID  C STIME TTY          TIME CMD\nroot         1     0  0
10:57 ?00:00:01 init      \nroot         2     0  0 10:57 ?        00:00:00
[kthreadd]\nroot 3     2  0 10:57 ?        00:00:00 [migration\/0]\nroot         4     2
```

```
0 10:57 ?00:00:03 [ksoftirqd\/0]\nroot         5     2  0 10:57 ?        00:00:00
[events\/0]\nroot 6     2  0 10:57 ?        00:00:00 [khelper]\nroot         9     2  0
10:57 ?        00:00:00 [kstop\/0]\nroot       134     2  0 10:57 ?        00:00:00
[kblockd\/0]\nroot       136     2 0 10:57 ?        00:00:00 [kacpid]\nroot       137
2  0 10:57 ?        00:00:00 [kacpi_notify]\nroot       216     2  0 10:57 ?
00:00:00 [ata\/0]\nroot       217     2 0 10:57 ?        00:00:00 [ata_aux]\nroot
218     2  0 10:57 ?        00:00:00 [ksuspend_usbd]\nroot       224     2  0 10:57 ?
00:00:00 [khubd]\nroot      227     2 0 10:57 ?       00:00:00 [kseriod]\nroot      229
2  0 10:57 ?        00:00:00 [kgameportd]\nroot       232     2  0 10:57 ?        00:00:00
[kmmcd]\nroot       283     2  0 10:57 ?        00:00:00 [pdflush]\nroot       284     2
0 10:57 ?        00:00:00 [pdflush]\nroot       285     2  0 10:57 ?        00:00:00
[kswapd0]\nroot       326     2  0 10:57 ?        00:00:00 [aio\/0]\nroot       337     2
0 10:57 ?        00:00:00 [nfsiod]\nroot 342     2  0 10:57 ?        00:00:00
[cifsoplockd]\nroot       343     2  0 10:57 ?00:00:00 [cifsdnotifyd]\nroot       527
2  0 10:57 ?        00:00:00 [scsi_eh_0]\nroot 529     2  0 10:57 ?        00:00:00
[scsi_eh_1]\nroot       533     2  0 10:57 ?00:00:00 [mtdblockd]\nroot       579     2  0
10:57 ?        00:00:00 [kpsmoused]\nroot 588     2  0 10:57 ?        00:00:00
[hid_compat]\nroot       612     2  1 10:57 ?00:06:35 [Glob_Spectra]\nroot       618     2
0 10:57 ?        00:02:18 [nandflush]\nroot 628     2  0 10:57 ?        00:00:00
[krfcommd]\nroot       631     2  0 10:57 ?        00:00:00 [rpciod\/0]\nroot       664
2  0 10:57 ?        00:00:00 [scsi_eh_2]\nroot       665     2 0 10:57 ?        00:00:00
[usb-storage]\nroot       672     2  0 10:57 ?       00:00:06 [kjournald]\nroot       935
1  0 10:57 ?        00:00:00 udevd -d\nroot      1750     2  0 10:57 ?        00:00:00
[SEC int]\nroot      1854     2  0 10:57 ?        00:00:35 [Clock_ISR]\nroot      1869
1  0 10:57 ?        00:00:00 \/bin\/cdp eth0 start\nroot 2142     1  0 10:57 ?
00:03:14 \/bin\/gdl_server blender_config 1\nroot      2211     2 0 10:57 ?
00:00:00 [VidDec_hal_pars]\nroot      2212     2  0 10:57 ?        00:00:00
[VidDec_hal_deco]\nroot      2218     2  0 10:57 ?        00:00:00 [VidPProc_ISR]\nroot
2219     2  0 10:57 ?        00:00:00 [VidPProc_IO]\nroot      2223     2  0 10:57
?00:00:24 [VidRend_IO]\nroot      2224     2  0 10:57 ?        00:00:00 [VidRend_IO]\nroot
2232     2  0 10:57 ?        00:01:18 [Audio_Rend_ISR]\nroot      2233     2  0 10:57
?00:00:27 [Audio_Timing]\nroot      2234     2  0 10:57 ?        00:00:08
[Audio_Pipe_Mgr]\nroot 2235     2  0 10:57 ?        00:00:00 [Audio_DSP0_ISR]\nroot
2236     2  0 10:57 ?00:00:00 [Audio_DSP1_ISR]\nroot      2426     1  0 10:58 ?
00:00:00 \/bin\/konfd\nroot 2899     1  0 10:58 ?        00:00:00 php-fpm: master process
(\/usr\/local\/cisco\/php\/etc\/php-fpm.conf)\nroot      2900  2899  0 10:58 ?
00:00:02 php-fpm: pool root \nroot      2901  2899  0 10:58 ?        00:00:02 php-fpm:
pool root \nroot      2907     1  0 10:58 ?        00:00:00 nginx: master process
\/usr\/local\/cisco\/nginx\/sbin\/nginx\nroot      2908  2907  0 10:58 ?        00:00:14
nginx: worker process \nroot      2917     1  0 10:58 ?        00:00:00
\/usr\/local\/cisco\/\/sbin\/cupsd -C \/usr\/local\/cisco\/\/etc\/cups\/cupsd.conf\nroot
2920     1  0 10:58 ?        00:00:00 \/usr\/local\/cisco\/sbin\/xinetd\nroot      2930
1  0 10:58 ?        00:00:19 \/usr\/local\/cisco\/sbin\/snmpd\nroot      2934     1  0
10:58 ?        00:00:00 \/sbin\/sshd\nroot 2941     1  0 10:58 ?        00:00:16
\/usr\/local\/cisco\/sbin\/cron\nroot      2943     1  0 10:58 ?        00:00:00
\/usr\/local\/cisco\/bin\/mosaic_server 0 5050 10000\nroot      2944 1  0 10:58 ?
00:00:00 \/usr\/local\/cisco\/bin\/mosaic_server 1 5052 12000\nroot 2949    1  0 10:58 ?
00:00:00 audio_setup_outputs\nroot     2958    2  0 10:58 ?00:00:00 [Audio_Input]\nroot
2959  2949  0 10:58 ?        00:00:00 [sh] <defunct>\nroot 2964     2  0 10:58 ?
00:00:01 [kjournald]\nroot      2966     1  0 10:58 ?00:00:01
\/usr\/local\/cisco\/sbin\/rsyslogd -c5\n1000      2977     1  0 10:58 ?00:00:08
dbus-daemon --system\n1001      2979     1  0 10:58 ?        00:00:00 hald
--daemon=yes\nroot      2980  2979  0 10:58 ?        00:00:00 hald-runner\nroot      2994
2980  0 10:58 ?        00:00:00 hald-addon-storage: polling \/dev\/ciscoapps (every 2
sec)\nroot 2997     1  0 10:58 ?        00:00:01 bluetoothd\nroot      3004     1  0 10:58
?00:00:00 Agent_3g\nroot      3018     1  0 10:58 ?        00:00:00 wan_detector\nroot
3046 1  0 10:58 ?        00:00:05 slim\nroot      3089  3046  0 10:58 tty2     00:02:06
\/usr\/bin\/X -auth \/var\/run\/slim.auth\nroot      3134     1  0 10:58 ?        00:00:00
dhclient br0\nroot      3135     1  0 10:58 ?        00:00:19 \/bin\/sh
\/scripts\/status_check.sh\nroot 3209     1  0 10:59 ?       00:00:00 \/sbin\/smi\nroot
3243     2  0 10:59 ?00:00:00 [kjournald]\nstudent   3380  3046  0 10:59 ?        00:00:00
\/bin\/sh \/etc\/xinitrc xfce4\nstudent   3392  3380  0 10:59 ?        00:00:00 \/bin\/sh
\/scripts\/startxfce4\nstudent 3004  3392  0 10:59 ?       00:00:01 xfce4-session\nroot
3491     1  0 10:59 ?00:00:00 init      \nstudent   3502     1  0 10:59 ?        00:00:00
```

dbus-launch --autolaunch 4b8ead68809b704d85084ca50000005c --binary-syntax
--close-stderr\nstudent   3504   1 0 10:59 ?00:00:00
\/usr\/local\/cisco\/bin\/dbus-daemon --fork --print-pid 5 --print-address 7
--session\nstudent   3506   1 0 10:59 ?   00:00:00
\/usr\/local\/cisco\/lib\/xfce4\/xfconf\/xfconfd\nstudent   3554   1 0 10:59 ?
00:00:21 xfwm4\nstudent   3555 1 0 10:59 ?   00:00:00 xfsettingsd\nstudent   3565
1 0 10:59 ?   00:00:35 xfce4-panel\nstudent   3576   1 0 10:59 ?   00:00:00
Thunar --daemon\nstudent   3578 1 0 10:59 ?   00:00:03 xfdesktop\nstudent   3597
3004 0 10:59 ?   00:00:00 3G_Dongle\nstudent   3598 3004 0 10:59 ?   00:00:13
BlueToothUI\nstudent   3602 3565 0 10:59 ?   00:00:24
\/usr\/local\/cisco\/lib\/xfce4\/panel\/wrapper
\/usr\/local\/cisco\/lib\/xfce4\/panel\/plugins\/libsystray.so 6 16777251 systray
Notification Area Area where notification icons appear \nstudent 3621 3004 0 10:59 ?
00:00:03 wifi_status hide\nstudent   3632 3004 0 10:59 ?   00:00:12 wired_status
hide > \/dev\/null\nstudent 3635   1 0 10:59 ?   00:00:00
xfce4-settings-helper\nstudent   3679   1 0 10:59 ?00:00:00
\/usr\/local\/cisco\/libexec\/gvfsd\nstudent   3692   1 0 10:59 ?   00:00:00
\/usr\/local\/cisco\/libexec\/gconfd-2\nstudent   3709   1 0 10:59 ?   00:00:00
\/usr\/local\/cisco\/libexec\/gvfs-hal-volume-monitor\nstudent   3717   1 0 10:59
?00:00:00 \/usr\/local\/cisco\/libexec\/\/gvfs-fuse-daemon
\/apps\/localconfig\/student\/.gvfs\nstudent   3730   1 0 10:59 ?   00:00:00
\/usr\/local\/cisco\/libexec\/gvfsd-trash --spawner :1.11
\/org\/gtk\/gvfs\/exec_spaw\/0\nstudent   0044   1 0 11:00 ?   00:00:00
\/usr\/local\/cisco\/lib\/scim-1.0\/scim-launcher -d -c simple -e all -f socket
--no-stay\nstudent   4068   1 0 11:00 ?   00:00:00
\/usr\/local\/cisco\/lib\/scim-1.0\/scim-helper-manager\nstudent 4069   1 0 11:00 ?
00:00:00 \/usr\/local\/cisco\/lib\/scim-1.0\/scim-panel-gtk --display :0.0 -c socket -d
--no-stay\nstudent   4071   1 0 11:00 ?00:00:00
\/usr\/local\/cisco\/lib\/scim-1.0\/scim-launcher -d -c socket -e socket -f x11\nroot 6319
2980 0 16:40 ?   00:00:00 hald-addon-input: Listening on \/dev\/input\/event0
\/dev\/input\/event1 \/dev\/input\/event2\nroot   6510   2 0 16:41 ?   00:00:00
[Audio_Recovery]\nroot   6574   2 0 16:41 ?   00:00:00 [scsi_eh_4]\nroot
6575 2 0 16:41 ?   00:00:00 [usb-storage]\nroot   6680 2980 0 16:41 ?
00:00:00 hald-addon-storage: polling \/dev\/sdb (every 2 sec)\nroot   7088   2 0
16:41 ?00:00:00 [kjournald]\nstudent   7938   1 0 11:04 ?   00:00:08
\/usr\/local\/cisco\/bin\/Terminal\nstudent   7997 7938 0 11:04 ?   00:00:00
gnome-pty-helper\nstudent   7998 7938 0 11:04 pts\/0   00:00:00 bash\nroot   10418
7998 0 11:40 pts\/0   00:00:00 -bash\nroot   15412 3135 0 17:23 ?   00:00:00
sleep 1\nroot   15432 2901 0 17:23 ?00:00:00 sh -c cd
'\/usr\/local\/cisco\/nginx\/html\/api\/1.0\/sys' ; \/usr\/local\/cisco\/bin\/ps -ef
2>&1\nroot   15433 15432 0 17:23 ?   00:00:00 \/usr\/local\/cisco\/bin\/ps
-ef\n","success":"true","getAt":"2012-11-07 08:23:50"}

设置存储

不适用。如果做出该请求，则系统会报告 003 错误。

获取存储

示例：获取存储

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/storage
```

回复

{"storage":"Filesystem   Size   Used Available Use% Mounted
on\n\/dev\/root 1.6G   1.2G   343.1M 78% \/\ntmpfs   512.0M   428.0K
511.6M 0% \/tmp\ntmpfs   4.0K   0   4.0K   0% \/media\ntmpfs

20.0M   228.0K   19.8M   1% \/var\n\/dev\/ciscoapps   1.8G   527.4M   1.2G
30% \/apps\n\/dev\/Glob_Spectraa1   96.6M   77.9M   13.8M 85%
\/tmp\/smi_spectraa1\ntmpfs   32.0M   17.8M   14.2M 56%
\/tmp\/firefox_cached\n64.104.163.32:\/var\/www\/html\/api 25.6G   6.2G   18.1G 26%
\/usr\/local\/cisco\/nginx\/html\/api\n\/dev\/sdb1   7.3G   308.1M
6.7G   4% \/media\/sdb1\n","success":"true","getAt":"2012-11-07 08:35:55"}

设置模式

不适用。如果做出该请求，则系统会报告 003 错误。

获取模式

示例：获取模式

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/model
```

回复

{"model":"CS-E300-AP-K9","success":"true","getAt":"2012-11-07 08:59:00"}

设置 IP

示例：将 IP 地址设置为 64.104.163.55，掩码设置为 255.255.255.128

请求

```
curl -m 5 -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"type" : "static","ipv4":"64.104.163.55","netmask":"255.255.255.128"}'
https://64.104.163.47/api/1.0/sys/ip
```

回复

{"success":"true","updatedAt":"2012-11-06 19:49:21"}

参数限制

参数类型为 static 或 DHCP。如果在类型字段中指定 static，则还必须指定 ipv4 和掩码。IPv4 必须是一个有效 IPv4 地址。掩码必须是以下字符串之一：

255.0.0.0

255.128.0.0

255.192.0.0

255.224.0.0

255.240.0.0

255.248.0.0

255.252.0.0

255.254.0.0

255.255.0.0

255.255.128.0

255.255.192.0
255.255.224.0
255.255.240.0
255.255.248.0
255.255.252.0
255.255.254.0
255.255.255.0
255.255.255.128
255.255.255.192
255.255.255.224
255.255.255.240
255.255.255.248
255.255.255.252
255.255.255.254
255.255.255.255

注　因为在执行之后 IP 地址已改变，所以必须指定“ -m 5”以确保永远不能附加该命令。

获取 IP 地址

示例：获取 IP 地址

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/ip
```

回复

```
{"ip":"10.140.44.134","success":"true","getAt":"2012-11-08 08:48:53"}
```

设置网关

示例：设置网关

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"gateway":"64.104.163.1"}' https://10.140.44.134/api/1.0/sys/gateway
```

回复

```
{"success":"true","updatedAt":"2011-04-21 05:46:07"}
```

参数限制

网关应该是一个有效 IP 地址。

获取网关

示例：获取网关

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/gateway
```

回复

```
{"gateway":"64.104.163.1","success":"true","getAt":"2012-11-08 08:49:59"}
```

设置 DNS

示例：将 DNS 设置为 8.8.8.8

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"dns":"8.8.8.8"}' https://10.140.44.134/api/1.0/sys/dns
```

回复

```
{"success":"true","updatedAt":"2012-12-14 07:50:00"}
```

参数限制

DNS 应该是一个有效 IP 地址。

获取 DNS

示例：获取 DNS

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/dns
```

回复

```
{"dns":"64.104.123.144 171.70.168.183 ","success":"true","getAt":"2012-11-08 08:50:37"}
```

设置无线模式

示例：设置无线模式

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wifiMode":"client"}' https://10.140.44.134/api/1.0/sys/wifiMode
```

回复

```
{"success":"true","updatedAt":"2012-12-14 08:11:16"}
```

参数限制

无线模式应为 AP 或 client。

获取无线模式

示例：获取无线模式

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys/wifiMode
```

回复

```
{"wifiMode":"ap","success":"true","getAt":"2012-12-14 08:09:59"}
```

设置 Chrome 浏览器的一个代理

示例：设置 Chrome 浏览器的一个代理

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"proxy" :
{"host":"10.10.10.10","scheme":"http","port":"10","username":"cisco","password":"cisco"}}'
https://10.140.44.134/api/1.0/sys/proxy
```

回复

```
{"success":"true","updatedAt":"2012-12-14 08:11:16"}
```

参数限制

主机：IP 地址/主机名。如果未指定，则代理设置会被设置为无。

方案：http/https

端口：应该是在 0 到 65535 之间的一个整数。

用户名和密码：可选。指定代理的帐户信息。

注　您必须同时指定用户名和密码。如果仅指定了用户名，将同时删除用户名和密码。

获取 Chrome 浏览器的代理

示例：获取 Chrome 浏览器的代理

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.28.47/api/1.0/sys/proxy
```

回复

```
{"proxy":"http://cisco:cisco@10.10.10.10:10","success":"true","getAt":"2011-04-21
04:04:03"}
```

设置系统信息

示例：将主机名设置为 ce300， NTP 服务器设置为 202.120.2.101，日志大小设置为 30 M。

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"hostname":"ce300", "ntpServer" : "202.120.2.101",?log_size?:?30?}'
https://10.140.44.134/api/1.0/sys
```

回复

```
{"success":"true","updatedAt":"2012-11-06 19:49:21"}
```

参数限制

参数应该是一个有效 IPv4 地址。

获取系统信息

示例：获取所有系统信息

请求：

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/sys
```

回复

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.134/api/1.0/sys{"hostname":"intel_ce_linux", "size":"30",
"ip":"10.140.44.134","gateway":"10.140.28.1","dns":"64.104.123.144
171.70.168.183","language":"1","model":"CS-E300-AP-K9","locale":"8","time":"2012-12-14
16:13:03","ntpServer":"10.81.254.202","loginGui":"disable","resolution":"9","wifiMode":"cl
ient","bluetooth":"on","memory":"Mem: 380776K used, 1308116K free, 0K shrd, 27764K buff,
170748K cached","storage":"Filesystem                Size      Used Available Use% Mounted
on\n\/dev\/root                  1.6G      1.2G    300.9M  81% \/\ntmpfs
512.0M 172.0K    511.8M   0% \/tmp\ntmpfs                      4.0K         0      4.0K
0% \/media\ntmpfs                     20.0M     88.0K     19.9M   0%
\/var\n\/dev\/ciscoapps 1.8G    539.6M      1.2G  30% \/apps\n\/dev\/Glob_Spectraa1
96.6M     84.4M      7.2M  92% \/tmp\/smi_spectraa1\ntmpfs                     32.0M
17.8M   14.2M 56% \/tmp\/firefox_cached\n10.140.28.35:\/var\/www\/html\/api
25.6G      7.2G 17.1G  30% \/usr\/local\/cisco\/nginx\/html\/api\n","cpu":"CPU:   0% usr
72% sys   0% nic  27% idle   0% io   0% irq   0% sirq","proc":"UID        PID  PPID  C
STIME TTY          TIME CMD\nroot         1     0  0 15:35 ?        00:00:01 init
\nroot        2     0  0 15:35 ?00:00:00 [kthreadd]\nroot        3     2  0 15:35 ?
00:00:00 [migration\/0]\nroot 4     2  0 15:35 ?       00:00:00 [ksoftirqd\/0]\nroot
5     2  0 15:35 ?00:00:00 [events\/0]\nroot         6     2  0 15:35 ?        00:00:00
[khelper]\nroot         9 2  0 15:35 ?        00:00:00 [kstop\/0]\nroot       134     2  0
15:35 ?        00:00:00 [kblockd\/0]\nroot       136     2  0 15:35 ?        00:00:00
[kacpid]\nroot       137     2  0 15:35 ?       00:00:00 [kacpi_notify]\nroot       216
2  0 15:35 ?        00:00:00 [ata\/0]\nroot       217     2  0 15:35 ?        00:00:00
[ata_aux]\nroot       218     2  0 15:35 ?00:00:00 [ksuspend_usbd]\nroot       224     2
0 15:35 ?        00:00:00 [khubd]\nroot 227     2  0 15:35 ?        00:00:00
[kseriod]\nroot       229     2  0 15:35 ?00:00:00 [kgameportd]\nroot       232     2  0
15:35 ?        00:00:00 [kmmcd]\nroot       283 2  0 15:35 ?        00:00:00
[pdflush]\nroot       284     2  0 15:35 ?        00:00:00 [pdflush]\nroot       285     2
0 15:35 ?        00:00:00 [kswapd0]\nroot       326     2  0 15:35 ?        00:00:00
[aio\/0]\nroot       337     2  0 15:35 ?        00:00:00 [nfsiod]\nroot 342     2  0
15:35 ?        00:00:00 [cifsoplockd]\nroot       343     2  0 15:35 ?00:00:00
[cifsdnotifyd]\nroot       527     2  0 15:35 ?        00:00:00 [scsi_eh_0]\nroot 529
2  0 15:35 ?        00:00:00 [scsi_eh_1]\nroot       533     2  0 15:35 ?00:00:00
```

```
[mtdblockd]\nroot       579     2  0 15:35 ?        00:00:00 [kpsmoused]\nroot 588     2
0 15:35 ?        00:00:00 [hid_compat]\nroot       612     2  6 15:35 ?00:02:22
[Glob_Spectra]\nroot       618     2  0 15:35 ?        00:00:11 [nandflush]\nroot 628
2  0 15:35 ?        00:00:00 [krfcommd]\nroot       631     2  0 15:35 ?        00:00:00
[rpciod\/0]\nroot       688     2  0 15:35 ?        00:00:00 [scsi_eh_2]\nroot       689
2 0 15:35 ?        00:00:00 [usb-storage]\nroot       704     2  0 15:35 ?        00:00:00
[kjournald]\nroot       917     1  0 15:35 ?        00:00:00 udevd -d\nroot      2102
2  0 15:35 ?        00:00:00 [SEC int]\nroot      2134     1  0 15:35 ?        00:00:00
\/bin\/cdp eth0 start\nroot     2138     2  0 15:35 ?        00:00:04 [Clock_ISR]\nroot
2166     1  0 15:35 ?        00:00:21 \/bin\/gdl_server blender_config 1\nroot     2220
2  0 15:35 ?00:00:00 [VidDec_hal_pars]\nroot      2221     2  0 15:35 ?        00:00:00
[VidDec_hal_deco]\nroot      2227     2  0 15:35 ?        00:00:00 [VidPProc_ISR]\nroot
2228     2  0 15:35 ?        00:00:00 [VidPProc_IO]\nroot      2232     2  0 15:35
?00:00:02 [VidRend_IO]\nroot      2233     2  0 15:35 ?        00:00:00 [VidRend_ISR]\nroot
2237     2  0 15:35 ?        00:00:08 [Audio_Rend_ISR]\nroot      2238     2  0 15:35
?00:00:03 [Audio_Timing]\nroot      2239     2  0 15:35 ?        00:00:00
[Audio_Pipe_Mgr]\nroot 2240     2  0 15:35 ?        00:00:00 [Audio_DSP0_ISR]\nroot
2241     2  0 15:35 ?00:00:00 [Audio_DSP1_ISR]\nroot      2446     1  0 15:36 ?
00:00:00 \/usr\/local\/cisco\/sbin\/snmpd -p \/var\/run\/snmpd.pid\nroot      2448     1
0 15:36 ?        00:00:00 \/bin\/konfd\nroot      2583     1  0 15:36 ?        00:00:00
php-fpm: master process (\/usr\/local\/cisco\/php\/etc\/php-fpm.conf)
\nroot      2584  2583  0 15:36 ?00:00:00 php-fpm: pool root \nroot      2585  2583  0
15:36 ?        00:00:00 php-fpm: pool root \nroot      2592     1  0 15:36 ?
00:00:00 nginx: master process \/usr\/local\/cisco\/nginx\/sbin\/nginx\nroot      2602
1  0 15:36 ?00:00:00 \/usr\/local\/cisco\/\/sbin\/cupsd -C
\/usr\/local\/cisco\/\/etc\/cups\/cupsd.conf\nroot      2606     1  0 15:36 ?
00:00:00 \/usr\/local\/cisco\/sbin\/xinetd\nroot      2611     1  0 15:36 ?
00:00:00 \/sbin\/sshd\nroot      2620     1  0 15:36 ?00:00:11
\/usr\/local\/cisco\/sbin\/cron\nroot      2623     1  0 15:36 ?        00:00:00
\/usr\/local\/cisco\/bin\/mosaic_server 0 5050 10000\nroot      2624     1  0 15:36
?00:00:00 \/usr\/local\/cisco\/bin\/mosaic_server 1 5052 12000\nroot      2632     1  0
15:36 ?00:00:00 audio_setup_outputs\nroot      2641     2  0 15:36 ?        00:00:00
[Audio_Input]\nroot      2643  2632  0 15:36 ?        00:00:00 [sh] <defunct>\nroot
2645 2  0 15:36 ?        00:00:00 [Audio_Recovery]\nroot      2647     2  0 15:36 ?
00:00:00 [kjournald]\nroot      2649     1  0 15:36 ?        00:00:00
\/usr\/local\/cisco\/sbin\/rsyslogd -c5\n1000      2660     1  0 15:36 ?        00:00:00
dbus-daemon --system\n1001      2662     1 0 15:36 ?        00:00:00 hald
--daemon=yes\nroot      2663  2662  0 15:36 ?        00:00:00 hald-runner\nroot      2668
2663  0 15:36 ?        00:00:00 hald-addon-input: Listening on \/dev\/input\/event2
\/dev\/input\/event1 \/dev\/input\/event0\nroot      2677  2663  0 15:36 ?00:00:00
hald-addon-storage: polling \/dev\/ciscoapps (every 2 sec)\nroot    2680    1 0 15:36 ?
00:00:00 bluetoothd\nroot      2687     1  0 15:36 ?        00:00:00 Agent_3g\nroot
2715     1  0 15:36 ?        00:00:00 wan_detector\nroot      2743     1  0 15:36
?00:00:04 slim\nroot     2763  2743  1 15:36 tty2     00:00:22 \/usr\/bin\/X\nroot 2830
1  0 15:36 ?        00:00:00 dhclient br0\nroot      2831     1  0 15:36 ?00:00:01
\/bin\/sh \/scripts\/status_check.sh\nstudent   2886  2743  0 15:36 ?        00:00:00
\/bin\/sh \/etc\/xinitrc xfce4\nstudent   2891  2886  0 15:36 ?        00:00:00 \/bin\/sh
\/scripts\/startxfce4\nstudent   2916  2891  0 15:36 ?        00:00:01 xfce4-session\nroot
2944     1  0 15:36 ?        00:00:00 \/sbin\/smi\nroot      2958     1  0 15:36 ?00:00:00
\/bin\/heartbeat\nroot      2995     2  0 15:36 ?        00:00:00 [kjournald]\nstudent
3100     1  0 15:37 ?        00:00:00 dbus-launch --autolaunch
a822b50ba91705398f791a9200000055 --binary-syntax --close-stderr\nstudent   3120     1  0
15:37 ?        00:00:00 \/usr\/local\/cisco\/bin\/dbus-daemon --fork --print-pid 5
--print-address 7 --session\nstudent   3128     1 0 15:37 ?        00:00:00
\/usr\/local\/cisco\/lib\/xfce4\/xfconf\/xfconfd\nroot      3211 1  0 15:37 ?
00:00:00 init      \nstudent   3225     1  0 15:37 ?        00:00:00 xfsettingsd\nstudent
3244     1  0 15:37 ?        00:00:00 \/usr\/local\/cisco\/libexec\/gvfsd\nstudent  3255
1  0 15:37 ?        00:00:00 \/usr\/local\/cisco\/libexec\/\/gvfs-fuse-daemon
\/apps\/localconfig\/student\/.gvfs\nstudent   3259     1  0 15:37 ?        00:00:02
xfwm4\nstudent   3269     1  0 15:37 ?        00:00:02 xfce4-panel\nstudent   3278     1
0 15:37 ?        00:00:00 Thunar --daemon\nstudent   3287     1  0 15:37 ?        00:00:02
xfdesktop\nstudent   3309  2916  0 15:37 ?        00:00:00 3G_Dongle\nstudent   3311  2916
0 15:37 ?        00:00:01 BlueToothUI\nstudent   3329  2916  0 15:37 ?        00:00:01
wifi_status hide\nstudent   3337  2916  0 15:37 ?        00:00:03 wired_status hide >
```

\/dev\/null\nstudent 3338   1 0 15:37 ?   00:00:00 xfce4-settings-helper\nstudent
3363 3269 0 15:37 ?00:00:01 \/usr\/local\/cisco\/lib\/xfce4\/panel\/wrapper
\/usr\/local\/cisco\/lib\/xfce4\/panel\/plugins\/libsystray.so 6 16777251 systray
Notification Area Area where notification icons appear \nstudent   3401   1 0 15:37 ?
00:00:00 \/usr\/local\/cisco\/libexec\/gvfs-hal-volume-monitor\nstudent   3411   1 0
15:37 ?   00:00:00 \/usr\/local\/cisco\/libexec\/gvfsd-trash --spawner :1.4
\/org\/gtk\/gvfs\/exec_spaw\/0\nstudent 3443   1 0 15:37 ?   00:00:00
\/usr\/local\/cisco\/libexec\/gconfd-2\nstudent   3942 1 0 15:38 ?   00:00:00
\/usr\/local\/cisco\/lib\/scim-1.0\/scim-launcher -d -c simple -e all -f socket
--no-stay\nstudent   4079   1 0 15:38 ?   00:00:00
\/usr\/local\/cisco\/lib\/scim-1.0\/scim-helper-manager\nstudent 4080   1 0 15:38 ?
00:00:00 \/usr\/local\/cisco\/lib\/scim-1.0\/scim-panel-gtk --display :0.0 -c socket -d
--no-stay\nstudent 4082   1 0 15:38 ?   00:00:00
\/usr\/local\/cisco\/lib\/scim-1.0\/scim-launcher -d -c socket -e socket -f x11\nstudent
4290   1 0 15:38 ?   00:00:03 \/usr\/local\/cisco\/bin\/Terminal\nstudent   4349
4290 0 15:38 ?   00:00:00 gnome-pty-helper\nstudent 4350 4290 0 15:38 pts\/0
00:00:00 bash\nroot   4428 4350 0 15:38 pts\/0   00:00:00 -bash\nroot   5492
2592 0 15:39 ?   00:00:01 nginx: worker process   \nroot 6203 2831 0
16:13 ?   00:00:00 sleep 1\nroot   6225 2585 0 16:13 ?   00:00:00 sh -c cd
'\/usr\/local\/cisco\/nginx\/html\/api\/1.0\/sys' ;
LC_ALL=zh_CN.utf-8;\/usr\/local\/cisco\/bin\/ps -ef 2>&1\nroot   6226 6225 0 16:13 ?
00:00:00 \/usr\/local\/cisco\/bin\/ps -ef\nroot   15865   2 0 15:50 ?   00:00:00
[RtmpTimerTask]\nroot   15866 2 0 15:50 ?   00:00:02 [RtmpMlmeTask]\nroot
15867   2 0 15:50 ?   00:00:00 [RtmpCmdQTask]\nstudent 23415 4290 0 15:58
pts\/1   00:00:00 bash\nroot   23433 23415 0 15:58 pts\/1   00:00:00 -bash\nroot
30190 23433 0 16:05 pts\/1   00:00:00 clish\n","success":"true","getAt":"2012-12-14
08:13:07"}

以太网 API

使用此节中的命令可配置以太网 API。

注　Curl 用作请求 API 的一个示例工具。 10.140.44.134 用作云翼 300 系列交换机的 IP 地址的一个示例，而 cisco123! 用作 一个管理员密码示例。

设置 Gi1 状态

不适用。如果做出该请求，则系统会报告 003 错误。

获取 Gi1 状态

示例：获取 gi1 状态

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/gi1/status
```

回复

```
{"status":"enable","success":"true","getAt":"2012-11-08 08:47:44"}
```

设置 Gi1 MAC

不适用。如果做出该请求，则系统会报告 003 错误。

获取 Gi1 MAC

示例：获取 gi1 mac

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/gi1/mac
```

回复

```
{"mac":"1C:AA:07:97:A3:C0","success":"true","getAt":"2012-11-08 08:51:19"}
```

设置 Gi1 输出队列策略

示例：将 gi1 输出队列策略设置为 wrr

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"oqs" :
"wrr"}' https://10.140.44.134/api/1.0/eth/gi1/oqs
```

回复

```
{"success":"true","updatedAt":"2012-11-08 08:54:26"}
```

参数限制

wrr 和 strict 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Gi1 输出队列策略

示例：获取 gi1 输出队列策略

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/gi1/oqs
```

回复

```
{"oqs":"wrr","success":"true","getAt":"2012-11-08 09:07:08"}
```

设置 Gi1 暂停

示例：将 gi1 暂停设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"pause" :
"on"}' https://10.140.44.134/api/1.0/eth/gi1/pause
```

回复

```
{"success":"true","updatedAt":"2012-11-08 08:54:26"}
```

参数限制

on 和 off 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Gi1 暂停

示例：获取 gi1 暂停

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/gi1/pause
```

回复

```
{"pause":"on","success":"true","getAt":"2012-11-08 09:29:55"}
```

设置 Gi1 优先级

示例：将 gi1 暂停设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"priority" : "normal"}' https://10.140.44.134/api/1.0/eth/gi1/priority
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

normal 和 high 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Gi1 优先级

示例：获取 gi1 优先级

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/gi1/priority
```

回复

```
{"priority":"normal","success":"true","getAt":"2012-11-08 09:39:52"}
```

设置 Gi1 速率限制

示例：将 gi1 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100"}' https://10.140.44.134/api/1.0/eth/gi1/rateLim
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Gi1 速率限制

示例：获取 gi1 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/gi1/rateLim
```

回复

```
{"rateLim":"set unknown-unicast 100","success":"true","getAt":"2012-11-09 03:21:16"}
```

设置 Gi1 速度

示例：将 gi1 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"speed" :
"auto"}' https://10.140.44.134/api/1.0/eth/gi1/speed
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 10、 100 和 1000 是有效参数。

获取 Gi1 速度

示例：获取 gi1 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/gi1/speed
```

回复

```
{"speed":"auto","success":"true","getAt":"2012-11-14 07:03:00"}
```

设置 Gi1 双工

示例：将 gi1 双工设置为 auto

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"duplex"
: "auto"}' https://10.140.44.134/api/1.0/eth/gi1/duplex
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 full 和 half 是有效参数。

获取 Gi1 双工

示例：获取 gi1 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/gi1/duplex
```

回复

```
{"duplex":"auto","success":"true","getAt":"2012-11-15 01:18:14"}
```

设置 Gi1 信息

示例：将 gi1 速率限制设置为 unknown-unicast，暂停设置为关闭

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100", "pause":"off"}' https://10.140.44.134/api/1.0/eth/gi1
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Gi1 信息

示例：获取 gi1 信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/gi1
```

回复

```
{"status":"enable","dns":"64.104.123.144
171.70.168.183","mac":"1C:AA:07:97:A3:C0","oqs":"strict","pause":"on","priority":"normal",
"rateLim":"set unknown-unicast 100","speed":"100","success":"true","getAt":"2012-11-09
08:53:52"}
```

设置 Fe1 状态

示例：将 fe1 状态设置为 disable

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"status"
: "disable"}' https://10.140.44.134/api/1.0/eth/fe1/status
```

回复

```
{"success":"true","updatedAt":"2012-11-08 08:54:26"}
```

参数限制

enable 和 disable 是状态的有效字符串。否则系统会报告 004 错误。

获取 Fe1 状态

示例：获取 fe1 状态

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe1/status
```

回复

```
{"status":"disable","success":"true","getAt":"2012-11-21 06:33:40"}
```

设置 Fe1 输出队列策略

示例：将 gi1 输出队列策略设置为 wrr

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"oqs" :
"wrr"}' https://10.140.44.134/api/1.0/eth/fe1/oqs
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:35:35"}
```

参数限制

wrr 和 strict 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe1 输出队列策略

示例：获取 fe1 输出队列策略

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe1/oqs
```

回复

```
{"oqs":"wrr","success":"true","getAt":"2012-11-08 09:07:08"}
```

设置 Fe1 优先级

示例：将 fe1 暂停设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"priority" : "normal"}' https://10.140.44.134/api/1.0/eth/fe1/priority
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:42:18"}
```

参数限制

normal 和 high 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe1 优先级

示例：获取 gi1 优先级

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe1/priority
```

回复

```
{"priority":"normal","success":"true","getAt":"2012-11-08 09:39:52"}
```

设置 Fe1 速率限制

示例：将 fe1 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100"}' https://10.140.44.134/api/1.0/eth/fe1/rateLim
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:43:28"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe1 速率限制

示例：获取 fe1 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe1/rateLim
```

回复

```
{"rateLim":"set unknown-unicast 100","success":"true","getAt":"2012-11-09 03:21:16"}
```

设置 Fe1 速度

示例：将 fe1 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"speed" :
"auto"}' https://10.140.44.134/api/1.0/eth/fe1/speed
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 10 和 100 是有效参数。

获取 Fe1 速度

示例：获取 fe1 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe1/speed
```

回复

```
{"speed":"auto","success":"true","getAt":"2012-11-14 07:03:00"}
```

设置 Fe1 双工

示例：将 fe1 双工设置为 auto

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"duplex"
: "auto"}' https://10.140.44.134/api/1.0/eth/fe1/duplex
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 full 和 half 是有效参数。

获取 Fe1 双工

示例：获取 fe1 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe1/duplex
```

回复

```
{"duplex":"auto","success":"true","getAt":"2012-11-15 01:18:14"}
```

设置 Fe1 信息

示例：将 fe1 速率限制设置为 unknown-unicast，暂停设置为关闭

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100","duplex":"auto"}' https://10.140.44.134/api/1.0/eth/fe1
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe1 信息

示例：获取 fe1 信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe1
```

回复

```
{"status":"disable","oqs":"wrr","priority":"normal","rateLim":"set unknown-unicast
100","speed":"auto","duplex":"auto","success":"true","getAt":"2012-11-21 06:47:52"}
```

设置 Fe2 状态

示例：将 fe2 状态设置为 disable

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"status"
: "disable"}' https://10.140.44.134/api/1.0/eth/fe2/status
```

回复

```
{"success":"true","updatedAt":"2012-11-08 08:54:26"}
```

参数限制

enable 和 disable 是状态的有效字符串。否则系统会报告 004 错误。

获取 Fe2 状态

示例：获取 fe2 状态

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe2/status
```

回复

```
{"status":"disable","success":"true","getAt":"2012-11-21 06:33:40"}
```

设置 Fe2 输出队列策略

示例：将 gi1 输出队列策略设置为 wrr

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"oqs" :
"wrr"}' https://10.140.44.134/api/1.0/eth/fe2/oqs
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:35:35"}
```

参数限制

wrr 和 strict 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe2 输出队列策略

示例：获取 fe2 输出队列策略

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe2/oqs
```

回复

```
{"oqs":"wrr","success":"true","getAt":"2012-11-08 09:07:08"}
```

设置 Fe2 优先级

示例：将 fe2 暂停设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"priority" : "normal"}' https://10.140.44.134/api/1.0/eth/fe2/priority
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:42:18"}
```

参数限制

normal 和 high 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe2 优先级

示例：获取 gi1 优先级

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe2/priority
```

回复

```
{"priority":"normal","success":"true","getAt":"2012-11-08 09:39:52"}
```

设置 Fe2 速率限制

示例：将 fe2 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100"}' https://10.140.44.134/api/1.0/eth/fe2/rateLim
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:43:28"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe2 速率限制

示例：获取 fe2 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe2/rateLim
```

回复

```
{"rateLim":"set unknown-unicast 100","success":"true","getAt":"2012-11-09 03:21:16"}
```

设置 Fe2 速度

示例：将 fe2 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"speed" :
"auto"}' https://10.140.44.134/api/1.0/eth/fe2/speed
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 10 和 100 是有效参数。

获取 Fe2 速度

示例：获取 fe2 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe2/speed
```

回复

```
{"speed":"auto","success":"true","getAt":"2012-11-14 07:03:00"}
```

设置 Fe2 双工

示例：将 fe2 双工设置为 auto

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"duplex"
: "auto"}' https://10.140.44.134/api/1.0/eth/fe2/duplex
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 full 和 half 是有效参数。

获取 Fe2 双工

示例：获取 fe2 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe2/duplex
```

回复

```
{"duplex":"auto","success":"true","getAt":"2012-11-15 01:18:14"}
```

设置 Fe2 信息

示例：将 fe2 速率限制设置为 unknown-unicast，暂停设置为关闭

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100","duplex":"auto"}' https://10.140.44.134/api/1.0/eth/fe2
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe2 信息

示例：获取 fe2 信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe2
```

回复

```
{"status":"disable","oqs":"wrr","priority":"normal","rateLim":"set unknown-unicast
100","speed":"auto","duplex":"auto","success":"true","getAt":"2012-11-21 06:47:52"}
```

设置 Fe3 状态

示例：将 fe3 状态设置为 disable

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"status"
: "disable"}' https://10.140.44.134/api/1.0/eth/fe3/status
```

回复

```
{"success":"true","updatedAt":"2012-11-08 08:54:26"}
```

参数限制

enable 和 disable 是状态的有效字符串。否则系统会报告 004 错误。

获取 fe3 状态

示例：获取 fe3 状态

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/status
```

回复

```
{"status":"disable","success":"true","getAt":"2012-11-21 06:33:40"}
```

设置 Fe3 输出队列策略

示例：将 gi1 输出队列策略设置为 wrr

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"oqs" :
"wrr"}' https://10.140.44.134/api/1.0/eth/fe3/oqs
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:35:35"}
```

参数限制

wrr 和 strict 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe3 输出队列策略

示例：获取 fe3 输出队列策略

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/oqs
```

回复

```
{"oqs":"wrr","success":"true","getAt":"2012-11-08 09:07:08"}
```

设置 Fe3 优先级

示例：将 fe3 暂停设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"priority" : "normal"}' https://10.140.44.134/api/1.0/eth/fe3/priority
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:42:18"}
```

参数限制

normal 和 high 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 fe3 优先级

示例：获取 gi1 优先级

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/priority
```

回复

```
{"priority":"normal","success":"true","getAt":"2012-11-08 09:39:52"}
```

设置 Fe3 速率限制

示例：将 fe3 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100"}' https://10.140.44.134/api/1.0/eth/fe3/rateLim
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:43:28"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe3 速率限制

示例：获取 fe3 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/rateLim
```

回复

```
{"rateLim":"set unknown-unicast 100","success":"true","getAt":"2012-11-09 03:21:16"}
```

设置 Fe3 速度

示例：将 fe3 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"speed" :
"auto"}' https://10.140.44.134/api/1.0/eth/fe3/speed
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 10 和 100 是有效参数。

获取 Fe3 速度

示例：获取 fe3 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/speed
```

回复

```
{"speed":"auto","success":"true","getAt":"2012-11-14 07:03:00"}
```

设置 Fe3 双工

示例：将 fe3 双工设置为 auto

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"duplex"
: "auto"}' https://10.140.44.134/api/1.0/eth/fe3/duplex
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 full 和 half 是有效参数。

获取 Fe3 双工

示例：获取 fe3 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/duplex
```

回复

```
{"duplex":"auto","success":"true","getAt":"2012-11-15 01:18:14"}
```

设置 Fe3 信息

示例：将 fe3 速率限制设置为 unknown-unicast，暂停设置为关闭

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100","duplex":"auto"}' https://10.140.44.134/api/1.0/eth/fe3
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe3 信息

示例：获取 fe3 信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3
```

回复

```
{"status":"disable","oqs":"wrr","priority":"normal","rateLim":"set unknown-unicast
100","speed":"auto","duplex":"auto","success":"true","getAt":"2012-11-21 06:47:52"}
```

设置 Fe3 状态

示例：将 fe3 状态设置为 disable

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"status"
: "disable"}' https://10.140.44.134/api/1.0/eth/fe3/status
```

回复

```
{"success":"true","updatedAt":"2012-11-08 08:54:26"}
```

参数限制

enable 和 disable 是状态的有效字符串。否则系统会报告 004 错误。

获取 Fe3 状态

示例：获取 fe3 状态

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/status
```

回复

```
{"status":"disable","success":"true","getAt":"2012-11-21 06:33:40"}
```

设置 Fe3 输出队列策略

示例：将 gi1 输出队列策略设置为 wrr

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"oqs" :
"wrr"}' https://10.140.44.134/api/1.0/eth/fe3/oqs
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:35:35"}
```

参数限制

wrr 和 strict 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe3 输出队列策略

示例：获取 fe3 输出队列策略

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/oqs
```

回复

```
{"oqs":"wrr","success":"true","getAt":"2012-11-08 09:07:08"}
```

设置 Fe3 优先级

示例：将 fe3 暂停设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"priority" : "normal"}' https://10.140.44.134/api/1.0/eth/fe3/priority
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:42:18"}
```

参数限制

normal 和 high 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe3 优先级

示例：获取 gi1 优先级

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/priority
```

回复

```
{"priority":"normal","success":"true","getAt":"2012-11-08 09:39:52"}
```

设置 Fe3 速率限制

示例：将 fe3 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100"}' https://10.140.44.134/api/1.0/eth/fe3/rateLim
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:43:28"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe3 速率限制

示例：获取 fe3 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/rateLim
```

回复

```
{"rateLim":"set unknown-unicast 100","success":"true","getAt":"2012-11-09 03:21:16"}
```

设置 Fe3 速度

示例：将 fe3 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"speed" :
"auto"}' https://10.140.44.134/api/1.0/eth/fe3/speed
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 10、 100 和 1000 是有效参数。

获取 Fe3 速度

示例：获取 fe3 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/speed
```

回复

```
{"speed":"auto","success":"true","getAt":"2012-11-14 07:03:00"}
```

设置 Fe3 双工

示例：将 fe3 双工设置为 auto

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"duplex"
: "auto"}' https://10.140.44.134/api/1.0/eth/fe3/duplex
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 full 和 half 是有效参数。

获取 Fe3 双工

示例：获取 fe3 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3/duplex
```

回复

```
{"duplex":"auto","success":"true","getAt":"2012-11-15 01:18:14"}
```

设置 Fe3 信息

示例：将 fe3 速率限制设置为 unknown-unicast，暂停设置为关闭

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100","duplex":"auto"}' https://10.140.44.134/api/1.0/eth/fe3
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe3 信息

示例：获取 fe3 信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe3
```

回复

```
{"status":"disable","oqs":"wrr","priority":"normal","rateLim":"set unknown-unicast
100","speed":"auto","duplex":"auto","success":"true","getAt":"2012-11-21 06:47:52"}
```

设置 Fe4 状态

示例：将 fe4 状态设置为 disable

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"status"
: "disable"}' https://10.140.44.134/api/1.0/eth/fe4/status
```

回复

```
{"success":"true","updatedAt":"2012-11-08 08:54:26"}
```

参数限制

enable 和 disable 是状态的有效字符串。否则系统会报告 004 错误。

获取 Fe4 状态

示例：获取 fe4 状态

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/status
```

回复

```
{"status":"disable","success":"true","getAt":"2012-11-21 06:33:40"}
```

设置 Fe4 输出队列策略

示例：将 gi1 输出队列策略设置为 wrr

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"oqs" :
"wrr"}' https://10.140.44.134/api/1.0/eth/fe4/oqs
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:35:35"}
```

参数限制

wrr 和 strict 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe4 输出队列策略

示例：获取 fe4 输出队列策略

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/oqs
```

回复

```
{"oqs":"wrr","success":"true","getAt":"2012-11-08 09:07:08"}
```

设置 Fe4 优先级

示例：将 fe4 暂停设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"priority" : "normal"}' https://10.140.44.134/api/1.0/eth/fe4/priority
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:42:18"}
```

参数限制

normal 和 high 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe4 优先级

示例：获取 fe4 优先级

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/priority
```

回复

```
{"priority":"normal","success":"true","getAt":"2012-11-08 09:39:52"}
```

设置 Fe4 速率限制

示例：将 fe4 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100"}' https://10.140.44.134/api/1.0/eth/fe4/rateLim
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:43:28"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe4 速率限制

示例：获取 fe4 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/rateLim
```

回复

```
{"rateLim":"set unknown-unicast 100","success":"true","getAt":"2012-11-09 03:21:16"}
```

设置 Fe4 速度

示例：将 fe4 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"speed" :
"auto"}' https://10.140.44.134/api/1.0/eth/fe4/speed
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 10 和 100 是有效参数。

获取 Fe4 速度

示例：获取 fe4 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/speed
```

回复

```
{"speed":"auto","success":"true","getAt":"2012-11-14 07:03:00"}
```

设置 Fe4 双工

示例：将 fe4 双工设置为 auto

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"duplex"
: "auto"}' https://10.140.44.134/api/1.0/eth/fe4/duplex
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 full 和 half 是有效参数。

获取 Fe4 双工

示例：获取 fe4 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/duplex
```

回复

```
{"duplex":"auto","success":"true","getAt":"2012-11-15 01:18:14"}
```

设置 Fe4 信息

示例：将 fe4 速率限制设置为 unknown-unicast，暂停设置为关闭

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100","duplex":"auto"}' https://10.140.44.134/api/1.0/eth/fe4
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe4 信息

示例：获取 fe4 信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4
```

回复

```
{"status":"disable","oqs":"wrr","priority":"normal","rateLim":"set unknown-unicast
100","speed":"auto","duplex":"auto","success":"true","getAt":"2012-11-21 06:47:52"}
```

设置 Fe4 状态

示例：将 fe4 状态设置为 disable

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"status"
: "disable"}' https://10.140.44.134/api/1.0/eth/fe4/status
```

回复

```
{"success":"true","updatedAt":"2012-11-08 08:54:26"}
```

参数限制

enable 和 disable 是状态的有效字符串。否则系统会报告 004 错误。

获取 Fe4 状态

示例：获取 fe4 状态

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/status
```

回复

```
{"status":"disable","success":"true","getAt":"2012-11-21 06:33:40"}
```

设置 Fe4 输出队列策略

示例：将 gi1 输出队列策略设置为 wrr

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"oqs" :
"wrr"}' https://10.140.44.134/api/1.0/eth/fe4/oqs
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:35:35"}
```

参数限制

wrr 和 strict 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe4 输出队列策略

示例：获取 fe4 输出队列策略

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/oqs
```

回复

```
{"oqs":"wrr","success":"true","getAt":"2012-11-08 09:07:08"}
```

设置 Fe4 优先级

示例：将 fe4 暂停设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"priority" : "normal"}' https://10.140.44.134/api/1.0/eth/fe4/priority
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:42:18"}
```

参数限制

normal 和 high 是 oqs 的有效字符串。否则系统会报告 004 错误。

获取 Fe4 优先级

示例：获取 gi1 优先级

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/priority
```

回复

```
{"priority":"normal","success":"true","getAt":"2012-11-08 09:39:52"}
```

设置 Fe4 速率限制

示例：将 fe4 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100"}' https://10.140.44.134/api/1.0/eth/fe4/rateLim
```

回复

```
{"success":"true","updatedAt":"2012-11-21 06:43:28"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe4 速率限制

示例：获取 fe4 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/rateLim
```

回复

```
{"rateLim":"set unknown-unicast 100","success":"true","getAt":"2012-11-09 03:21:16"}
```

设置 Fe4 速度

示例：将 fe4 速率限制设置为 unknown-unicast

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"speed" :
"auto"}' https://10.140.44.134/api/1.0/eth/fe4/speed
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 10、 100 和 1000 是有效参数。

获取 Fe4 速度

示例：获取 fe4 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/speed
```

回复

```
{"speed":"auto","success":"true","getAt":"2012-11-14 07:03:00"}
```

设置 Fe4 双工

示例：将 fe4 双工设置为 auto

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"duplex"
: "auto"}' https://10.140.44.134/api/1.0/eth/fe4/duplex
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 auto、 full 和 half 是有效参数。

获取 Fe4 双工

示例：获取 fe4 速率限制

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4/duplex
```

回复

```
{"duplex":"auto","success":"true","getAt":"2012-11-15 01:18:14"}
```

设置 Fe4 信息

示例：将 fe4 速率限制设置为 unknown-unicast，暂停设置为关闭

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"rateLim"
: "set unknown-unicast 100","duplex":"auto"}' https://10.140.44.134/api/1.0/eth/fe4
```

回复

```
{"success":"true","updatedAt":"2012-11-08 09:37:31"}
```

参数限制

仅 none 和 set broadcast/unknown-unicast/both [1-100] 是有效参数。

获取 Fe4 信息

示例：获取 fe4 信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/eth/fe4
```

回复

```
{"status":"disable","oqs":"wrr","priority":"normal","rateLim":"set unknown-unicast
100","speed":"auto","duplex":"auto","success":"true","getAt":"2012-11-21 06:47:52"}
```

发出一个命令

注　Curl 用作请求 API 的一个示例工具。 10.140.44.134 用作云翼 300 系列交换机的 IP 地址的一个示例，而 cisco123! 用作 一个管理员密码示例。

重新启动云翼 300

示例：重新启动云翼 300

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/cmd/reboot
```

回复

```
{"reboot":"true","success":"true","getAt":"2012-11-12 05:10:43"}
```

镜像版本信息

注　Curl 用作请求 API 的一个示例工具。 10.140.44.134 用作云翼 300 系列交换机的 IP 地址的一个示例，而 cisco123! 用作 一个管理员密码示例。

获取操作系统版本信息

示例：获取操作系统版本信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/image/os
```

回复

```
{"os":"1.3.9.1","success":"true","getAt":"2012-11-12 05:51:56"}
```

获取第 3 个应用版本信息

示例：获取第 3 个应用版本信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/image/3rdapp
```

回复

```
{"3rdapp":"1.3.9.1","success":"true","getAt":"2012-11-12 05:56:39"}
```

一次性获取操作系统和第 3 个应用版本

示例：一次性获取操作系统和第 3 个应用版本信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/image
```

回复

```
{"os":"1.3.9.1","3rdapp":"1.3.9.1","success":"true","getAt":"2012-11-12 08:18:58"}
```

AP 信息

注　Curl 用作请求 API 的一个示例工具。 10.140.44.134 用作云翼 300 系列交换机的 IP 地址的一个示例，而 cisco123! 用作 一个管理员密码示例。

设置 AP SSID

示例：将 SSID 设置为 cisco

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"ssid" :
"cisco"}' https://10.140.44.134/api/1.0/wifi/ap/ssid
```

回复

```
{"success":"true","updatedAt":"2012-11-12 06:25:47"}
```

参数限制

AP SSID 的长度应该为 1 到 32 个字符，并且有效参数集为 {a-zA-Z0-9-_}，否则系统会报告 004 错误。

获取 AP SSID

示例：获取 SSID

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/wifi/ap/ssid
```

回复

```
{"ssid":"abc","success":"true","getAt":"2012-11-12 06:22:54"}
```

设置 AP 无线电

示例：将 AP 无线电设置为关闭

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"radio" :
"off"}' https://10.140.44.134/api/1.0/wifi/ap/radio
```

回复

```
{"success":"true","updatedAt":"2012-11-12 06:34:20"}
```

参数限制

仅开启和关闭是有效参数，否则系统会报告 004 错误。

获取无线电状态

示例：获取无线电状态

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/wifi/ap/radio
```

回复

```
{"radio":"on","success":"true","getAt":"2012-11-12 06:33:20"}
```

设置无线模式

示例：将无线模式设置为 9

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wirelessMode" : "7"}' https://10.140.44.134/api/1.0/wifi/ap/wirelessMode
```

回复

```
{"success":"true","updatedAt":"2012-11-12 07:25:00"}
```

参数限制

仅 0、 1、 4、 6、 7 和 9 是有效参数，否则系统会报告 004 错误。

获取无线网模式

示例：获取无线网模式

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/wifi/ap/wirelessMode
```

回复

```
{"wirelessMode":"9","success":"true","getAt":"2012-11-12 07:23:22"}
```

设置信道号

示例：将信道号设置为 9

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"channelNumber" : "9"}' https://10.140.44.134/api/1.0/wifi/ap/channelNumber
```

回复

```
{"success":"true","updatedAt":"2012-11-12 07:32:04"}
```

参数限制

仅 0 到 14 的整数是有效参数，否则系统会报告 004 错误。

获取信道号

示例：获取信道号

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.134/api/1.0/wifi/ap/channelNumber
```

回复

```
{"channelNumber":"6","success":"true","getAt":"2012-11-12 07:29:41"}
```

设置信道分配

示例：将信道分配设置为中国

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"channelAllocation" : "4"}' https://10.140.44.134/api/1.0/wifi/ap/channelAllocation
```

回复

```
{"success":"true","updatedAt":"2012-11-12 07:32:04"}
```

参数限制

仅 1 到 4 的整数是有效参数，否则系统会报告 004 错误。

获取信道分配

示例：获取信道分配

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.134/api/1.0/wifi/ap/channelAllocation
```

回复

```
{"channelNumber":"6","success":"true","getAt":"2012-11-12 07:29:41"}
```

设置信道带宽

示例：将信道带宽设置为 20

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"channelBandwidth" : "20"}' https://10.140.44.134/api/1.0/wifi/ap/channelBandwidth
```

回复

```
{"success":"true","updatedAt":"2012-11-12 07:32:04"}
```

参数限制

仅 20 和 20/40 是有效参数，否则系统会报告 004 错误。

获取信道带宽

示例：获取信道带宽

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.134/api/1.0/wifi/ap/channelBandwidth
```

回复

```
{"channelNumber":"6","success":"true","getAt":"2012-11-12 07:29:41"}
```

设置传输功率

示例：将传输功率设置为 50

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"transmitPower" : "50"}' https://10.140.44.134/api/1.0/wifi/ap/transmitPower
```

回复

```
{"success":"true","updatedAt":"2012-11-12 07:49:35"}
```

参数限制

仅 1 到 100 的整数是有效参数，否则系统会报告 004 错误。

获取传输功率

示例：获取传输功率

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.134/api/1.0/wifi/ap/transmitPower
```

回复

```
{"transmitPower":"100","success":"true","getAt":"2012-11-12 07:48:15"}
```

设置 MCS

示例：将 mcs 设置为 15

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"mcs" :
"15"}' https://10.140.44.134/api/1.0/wifi/ap/mcs
```

回复

```
{"success":"true","updatedAt":"2012-11-12 07:49:35"}
```

参数限制

仅 0 到 15 的整数和 33 是有效参数，否则系统会报告 004 错误。

获取 MCS

示例：获取 mcs

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/wifi/ap/mcs
```

回复

```
{"mcs":"33","success":"true","getAt":"2012-11-12 07:57:08"}
```

设置 IGMP 监听

示例：将 IGMP 监听设置为开启

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"igmpSnoop" : "on"}' https://10.140.44.134/api/1.0/wifi/ap/igmpSnoop
```

回复

```
{"success":"true","updatedAt":"2012-11-12 08:08:05"}
```

参数限制

仅开启和关闭是有效参数，否则系统会报告 004 错误。

获取 IGMP 监听

示例：获取 igmp 监听

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/wifi/ap/igmpSnoop
```

回复

```
{"igmpSnoop":"off","success":"true","getAt":"2012-11-12 08:06:44"}
```

设置加密

示例 1：将加密模式设置为 open，类型设置为 none

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"mode" :
"open","type":"none"}' https://10.140.44.134/api/1.0/wifi/ap/encryption
```

回复

```
{"success":"true","updatedAt":"2012-11-12 08:08:05"}
```

示例 2：将加密模式设置为 open，类型设置为 wep，密钥号设置为 1，密钥类型设置为 ASCII，密钥值设置为 cisco

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"mode" :
"open","type":"wep","keyNum":"1","keyType":"ascii","key":"cisco"}'
https://10.140.44.134/api/1.0/wifi/ap/encryption
```

回复

```
{"success":"true","updatedAt":"2012-11-12 08:08:05"}
```

参数限制

以下是云翼 300 上的命令：

```
encryption mode open type none
encryption mode open type wep key [1-4] [ascii|hex] [key]
```

当模式为 open 时，只有 none 和 wep 是有效类型。

使用 none 类型时，不能指定任何其他参数。

使用 wep 类型时，必须指定密钥号、密钥类型和密钥。

密钥号：1-4

密钥类型：ASCII 或 hex

当密钥类型是 ASCII 时，{a-zA-Z0-9-_} 是有效的字符集，且长度必须为 5 或 13。

当密钥类型是 hex 时，{a-f0-9} 是有效的字符集，且长度必须为 10 或 26。

示例 3：将加密模式设置为 shared，类型设置为 wep，密钥号设置为 1，密钥类型设置为 ASCII，密钥值设置为 cisco

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"mode" :
"shared","type":"wep","keyNum":"1","keyType":"ascii","key":"cisco"}'
https://10.140.44.134/api/1.0/wifi/ap/encryption
```

回复

```
{"success":"true","updatedAt":"2012-11-12 08:08:05"}
```

参数限制

以下是云翼 300 上的命令：

```
encryption mode shared type wep key  [1-4] [ascii|hex] [key]
```

使用 shared 模式时，必须指定其他 4 个参数：

密钥类型：必须为 wep

密钥号：1-4

密钥类型：ASCII 或 hex

当密钥类型是 ASCII 时，{a-zA-Z0-9-_} 是有效的字符集，且长度必须为 5 或 13。

当密钥类型是 hex 时，{a-f0-9} 是有效的字符集，且长度必须为 10 或 26。

示例 4：将加密模式设置为 wpa，密钥类型设置为 aes

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"mode":"wpa", "type":"aes"}' https://10.140.44.134/api/1.0/wifi/ap/encryption
ªÿ³¥{"success":"true","updatedAt":"2012-11-12 08:08:05"}
```

参数限制

以下是云翼 300 上的命令：

```
encryption mode wpa type [tkip|aes|tkipaes]
```

使用 wpa 模式时，必须指定以下几种类型：

类型：必须为 tkip、 aes 或 tkipaes

示例 5：将加密模式设置为 wpa，密钥类型设置为 aes

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"mode":"wpa", "type":"aes"}' https://10.140.44.134/api/1.0/wifi/ap/encryption
```

回复

```
{"success":"true","updatedAt":"2012-11-12 08:08:05"}
```

参数限制

以下是云翼 300 上的命令：

```
encryption mode [wpa|wpa2|wpa1wpa2] type [tkip|aes|tkipaes]
```

使用 wpa、 wpa2 或 wpa1wpa2 模式时，必须指定以下几种类型：

类型：必须为 tkip、 aes 或 tkipaes

示例 6：将加密模式设置为 wpapsk，密钥类型设置为 tkipaes

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"mode":"wpapsk", "type":"tkipaes", "passPhrase":"cisco12345"}'
https://10.140.44.134/api/1.0/wifi/ap/encryption
```

回复

```
{"success":"true","updatedAt":"2012-11-12 08:08:05"}
```

参数限制

以下是云翼 300 上的命令：

```
encryption mode [wpapsk|wpa2psk|wpapskwpa2psk] type [tkip|aes|tkipaes] pass-phrase [key]
```

使用 wpapsk、 wpa2psk 或 wpapskwpa2psk 模式时，必须指定以下几种类型：

类型：必须为 tkip、 aes 或 tkipaes，

口令：口令的有效字符集为 {0-9a-zA-Z_-}，长度在 8 到 63 之间。

设置 Radius 服务器

示例 1：将 RADIUS 服务器主机设置为 1.1.1.1，身份验证端口设置为 444，密钥设置为 cisco123

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"host":"1.1.1.1", "auth-port":"444", "key":"cisco123"}'
https://10.140.44.134/api/1.0/wifi/ap/radius
```

回复

```
{"success":"true","updatedAt":"2012-11-12 08:08:05"}
```

参数限制

主机：必需参数。您必须输入一个有效的 IPv4 地址。

身份验证端口：可选参数。您必须输入一个在 0 到 65535 之间的数字。

密钥：可选参数。有效字符集为 {0-9 a-z A-Z _-`~!@#$%^*()+:,<>./[]{}&}

获取 Radius 服务器

示例：获取 Radius 服务器

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/wifi/ap/radius
```

回复

```
{"host_1":"1.1.1.1","auth-port_1":"444","key_1":"cisco123","host_2":"2.2.2.2","key_2":"2.2
.2.2","host_3":"3.3.3.3","auth-port_3":"1234","key_3":"cisco","success":"true","getAt":"20
12-11-14 06:35:30"}
```

设置 AP 信息

示例：将 AP SSID 设置为 cisco， RADIUS 服务器设置为 "{"host":"1.1.1.1", "auth-port":"555", "key":"cisco123"}"

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"ssid":"cisco","radius":{"host":"1.1.1.1", "auth-port":"555", "key":"cisco123"}}'
https://10.140.44.134/api/1.0/wifi/ap/
```

回复

```
{"success":"true","updatedAt":"2012-11-19 02:46:54"}
```

获取 AP 信息

示例：获取 AP 信息

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.134/api/1.0/wifi/ap
```

回复

```
{"channelAllocation":"3","igmpSnoop":"off","radio":"off","wirelessMode":"7","channelBandwi
dth":"20","mcs":"15","radius":{"host_1":"1.1.1.1","auth-port_1":"555","key_1":"cisco123"},
"channelNumber":"9","multicastMcs":"15","ssid":"cisco","encryption":{"mode":"wpa2psk","key
Type":"tkipaes","key":"Cisco123"},"transmitPower":"50","success":"true","getAt":"2012-11-1
9 02:56:25"}
```

WiFi 客户端信息

注　Curl 用作请求 API 的一个示例工具。 10.140.44.134 用作云翼 300 系列交换机的 IP 地址的一个示例，而 cisco123! 用作 一个管理员密码示例。

注　假设 WiFi 客户端 IP 地址为 10.140.44.148。

获取网络的 ID

示例：获取新的网络 ID

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/new_network_id
```

回复

```
{"new_network_id":"3","success":"true","getAt":"2011-04-21 04:26:46"}
```

获取网络的 SSID

示例：获取网络 0 的 SSID

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.148/api/1.0/wifi/client/0/ssid
```

回复

```
{"ssid":"blizzard","success":"true","getAt":"2011-04-21 04:09:14"}
```

参数限制

参数的长度应该小于 33 个字符，否则系统会报告 004 错误。

设置网络的 SSID

示例：将网络 0 的 SSID 设置为云翼 300

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d '{"ssid" :
"blizzard_2"}' https://10.140.44.148/api/1.0/wifi/client/0/ssid
```

回复

```
{"ssid":"blizzard","success":"true","getAt":"2011-04-21 05:07:50"}
```

获取网络的 SSID 扫描状态

示例：检查网络 0 的 SSID 扫描状态

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/0/scan_ssid
```

回复

```
{"scan_ssid":"0","success":"true","getAt":"2011-04-21 04:20:36"}
```

设置网络的 SSID 扫描

示例：设置网络 1 的 SSID 扫描

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"scan_ssid":"1"}' https://10.140.44.148/api/1.0/wifi/client/0/scan_ssid
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

参数限制

仅 0 和 1 是有效参数，否则系统会报告 004 错误。

获取网络的密钥管理类型

示例：获取网络 0 的密钥管理类型

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/0/key_mgmt
```

回复

```
{"key_mgmt":"WPA-EAP","success":"true","getAt":"2011-04-21 04:22:47"}
```

设置网络的密钥管理类型

示例：将网络的密钥管理类型设置为 WPA-EAP

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"key_mgmt":"WPA-EAP"}' https://10.140.44.148/api/1.0/wifi/client/0/key_mgmt
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

参数限制

仅 WPA-PSK、 WPA-EAP 和 None 是有效参数，否则系统会报告 004 错误。

获取网络的 Pairwise 类型

示例：获取网络的 Pairwise 类型

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/0/pairwise
```

回复

```
{"pairwise":"CCMP","success":"true","getAt":"2011-04-21 04:27:17"}
```

设置网络的 Pairwise 类型

示例：将网络的 Pairwise 类型设置为 CCMP

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"pairwise":"CCMP"}' https://10.140.44.148/api/1.0/wifi/client/0/pairwise
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

参数限制

仅 CCMP 和 TKIP 是有效参数，否则系统会报告 004 错误。

获取网络的组

示例：获取网络 0 的组

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.148/api/1.0/wifi/client/0/group
```

回复

```
{"group":"CCMP TKIP WEP104 WEP40","success":"true","getAt":"2011-04-21 04:29:11"}
```

设置网络的组

示例：将网络的组设置为 CCMP TKIP WEP104 WEP40

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"group":"CCMP"}' https://10.140.44.148/api/1.0/wifi/client/0/group
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

参数限制

仅 CCMP、 TKIP、 WEP104 和 WEP40 是有效参数，否则系统会报告 004 错误。

获取网络的 PSK

示例：获取网络 0 的 psk

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.148/api/1.0/wifi/client/0/psk
```

回复

```
{"psk":"*","success":"true","getAt":"2011-04-21 04:33:16"}
```

注　如果未设置 PSK，返回值就是 FAIL。如果设置了 PSK，返回值就是 *。

设置网络的 PSK

示例：将网络的 psk 设置为 cisco

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"psk":"cisco"}' https://10.140.44.148/api/1.0/wifi/client/0/psk
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

参数限制

PSK 参数必须为包含 0-9、 a-z、 A-Z _- 的一个字符串，且长度应为 8-63。

获取网络的 wep_key0

示例：获取网络 0 的 wep_key0

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/0/wep_key0
```

回复

```
{"wep_key0":"*","success":"true","getAt":"2011-04-21 04:37:46"}
```

注　wep_key0 采用用于保护客户专用信息的加密文本。

设置网络的 wep_key0

示例 1：将网络 0 的 wep_key0 设置为 cisco

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wep_key0":"cisco"}' https://10.140.44.148/api/1.0/wifi/client/0/wep_key0
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

示例 2：将网络 0 的 wep_key0 设置为 01234567890

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wep_key0":"0x123456789a"}' https://10.140.44.148/api/1.0/wifi/client/0/wep_key0
```

回复

```
{"success":"true","updatedAt":"2011-04-21 04:11:51"}
```

参数限制

参数长度应为 5 或 13（以 ASCII 表示）或者 10 或 26（以十六进制数字表示）。如果参数以十六进制数字表示，其前缀必须为 0x（前缀长度不计）。否则系统会报告 004 错误。

获取网络的 wep_key1

示例：获取网络 0 的 wep_key1

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/0/wep_key1
```

回复

```
{"wep_key1":"*","success":"true","getAt":"2011-04-21 04:37:46"}
```

注　wep_key1 采用用于保护客户专用信息的加密文本。

设置网络的 wep_key1

示例 1：将网络 0 的 wep_key1 设置为 cisco

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wep_key1":"cisco"}' https://10.140.44.148/api/1.0/wifi/client/0/wep_key1
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

示例 2：将网络 0 的 wep_key1 设置为 01234567890

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wep_key1":"0x123456789a"}' https://10.140.44.148/api/1.0/wifi/client/0/wep_key1
```

回复

```
{"success":"true","updatedAt":"2011-04-21 04:11:51"}
```

参数限制

参数长度应为 5 或 13（以 ASCII 表示）或者 10 或 26（以十六进制数字表示）。如果参数以十六进制数字表示，其前缀必须为 0x（前缀长度不计）。否则系统会报告 004 错误。

获取网络的 wep_key2

示例：获取网络 0 的 wep_key2

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/0/wep_key2
```

回复

```
{"wep_key2":"*","success":"true","getAt":"2011-04-21 04:37:46"}
```

注　wep_key2 采用用于保护客户专用信息的加密文本。

设置网络的 wep_key2

示例 1：将网络 0 的 wep_key2 设置为 cisco

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wep_key2":"cisco"}' https://10.140.44.148/api/1.0/wifi/client/0/wep_key2
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

示例 2：将网络 0 的 wep_key2 设置为 01234567890

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wep_key2":"0x123456789a"}' https://10.140.44.148/api/1.0/wifi/client/0/wep_key2
```

回复

```
{"success":"true","updatedAt":"2011-04-21 04:11:51"}
```

参数限制

参数长度应为 5 或 13（以 ASCII 表示）或者 10 或 26（以十六进制数字表示）。如果参数以十六进制数字表示，其前缀必须为 0x（前缀长度不计）。否则系统会报告 004 错误。

获取网络的 wep_key3

示例：获取网络 0 的 wep_key3

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/0/wep_key3
```

回复

```
{"wep_key3":"*","success":"true","getAt":"2011-04-21 04:37:46"}
```

注　wep_key3 采用用于保护客户专用信息的加密文本。

设置网络的 wep_key3

示例 1：将网络 0 的 wep_key3 设置为 cisco

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wep_key3":"cisco"}' https://10.140.44.148/api/1.0/wifi/client/0/wep_key3
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

示例 2：将网络 0 的 wep_key3 设置为 01234567890

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"wep_key3":"0x123456789a"}' https://10.140.44.148/api/1.0/wifi/client/0/wep_key3
```

回复

```
{"success":"true","updatedAt":"2011-04-21 04:11:51"}
```

参数限制

参数长度应为 5 或 13（以 ASCII 表示）或者 10 或 26（以十六进制数字表示）。如果参数以十六进制数字表示，其前缀必须为 0x（前缀长度不计）。否则系统会报告 004 错误。

获取网络的 EAP 类型

示例：获取网络 0 的 EAP 类型

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.148/api/1.0/wifi/client/0/eap
```

回复

```
{"eap":"PEAP","success":"true","getAt":"2011-04-21 04:38:13"}
```

设置网络的 EAP 类型

示例：将网络 0 的 EAP 类型设置为 PEAP

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"eap":"PEAP"}' https://10.140.44.148/api/1.0/wifi/client/0/eap
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

参数限制

仅 MSCHAPV2、TLS、PEAP、TTLS、FAST 和 LEAP 是有效参数，否则系统会报告 004 错误。

获取网络的 EAP 身份字符串

示例：获取网络 0 的 EAP 身份字符串

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/0/identity
```

回复

```
{"identity":"ce300","success":"true","getAt":"2011-04-21 04:43:04"}
```

设置网络的 EAP 身份字符串

示例：将网络 0 的 EAP 身份字符串设置为 ce300

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"identity":"ce300"}' https://10.140.44.148/api/1.0/wifi/client/0/identity
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

获取网络的密码

示例：获取网络 0 的 EAP 密码

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/0/password
```

回复

```
{"password":"*","success":"true","getAt":"2011-04-21 04:49:46"}
```

注　密码已加密，显示为 *。

设置网络的密码

示例：将网络 0 的密码设置为 ce300

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"password":"ce300"}' https://10.140.44.148/api/1.0/wifi/client/0/password
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

设置网络的状态

示例：将网络 0 的状态设置为 enable

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"status":"enable"}' https://10.140.44.148/api/1.0/wifi/client/0/stauts
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

参数限制

仅 enable 和 disable 是有效参数，否则系统会报告 004 错误。

删除网络

示例：删除网络 0

请求

```
curl -k -X DELETE -m 10 -H 'password: cisco123!' -H 'Content-Type: application/json'
https://10.140.44.148/api/1.0/wifi/client/0
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:13:49"}
```

注　参数 – m 设置允许整个操作花费的最长时间（以秒计），此示例中该时间为 10 秒。如果在 ce300 上只有一个网络，删除操作会将其从网络断开。因此，如果未设置 – m，则不会处理该请求。

保存网络配置

示例：保存网络配置

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.148/api/1.0/wifi/client/saving
```

回复

```
{"saving":"OK","success":"true","getAt":"2011-04-21 04:20:33"}
```

显示连接状态

示例：显示连接状态

请求

```
curl -k -X GET -H 'password: cisco123!' https://10.140.44.148/api/1.0/wifi/client/
```

回复

```
{"conn_status":"bssid=a4:56:30:5d:e1:d0\nssid=blizzard\nid=0\nmode=station\npairwise_ciphe
r=CCMP\ngroup_cipher=CCMP\nkey_mgmt=WPA2\/IEEE
802.1X\/EAP\nwpa_state=COMPLETED\nip_address=10.140.44.148\naddress=1c:aa:07:97:a3:c8\nSup
plicant PAE state=AUTHENTICATED\nsuppPortStatus=Authorized\nEAP
state=SUCCESS\nselectedMethod=25 (EAP-PEAP)\nEAP TLS cipher=AES256-SHA\nEAP-PEAPv1 Phase2
method=GTC\n","success":"true","getAt":"2011-04-21 04:56:29"}
```

重新加载保存的配置

示例：重新加载保存的配置

请求

```
curl -k -X GET -H 'password: cisco123!'
https://10.140.44.148/api/1.0/wifi/client/reconfiguration
```

回复

```
{"reconfiguration":"OK","success":"true","getAt":"2011-04-21 04:58:15"}
```

导出配置文件

示例：将 wifi-network-only 配置文件导出到 /tmp 下

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"type":"wifi-network-only","destination":"/tmp"}'
https://10.140.44.148/api/1.0/configuration/export
```

回复

```
{"success":"true","updatedAt":"2011-04-21 05:12:11"}
```

参数限制

类型字段可以是 wifi 网络配置、全部配置和启动配置。目标字段可以是包含 0-9、A-Z 和 a-z 在内的任意字符串。

导入配置文件

示例：从 /tmp 导入 wifi-network-only 配置文件

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"type":"wifi-network-only","source":"/abc"}'
https://10.140.44.148/api/1.0/configuration/import
```

回复

```
{"success":"true","updatedAt":"2011-04-21 06:49:02"}
```

参数限制

类型字段可以是 wifi 网络配置、全部配置和启动配置。源字段可以是包含 0-9、A-Z 和 a-z 在内的任意字符串。

RS232 配置

注　Curl 用作请求 API 的一个示例工具。 10.140.44.134 用作云翼 300 系列交换机的 IP 地址的一个示例，而 cisco123! 用作 一个管理员密码示例。

配置 RS232

示例：配置 RS-232，设备名称为 /dev/ttyS0，波特率为 9600，数据速率为 8，停止位为 1，奇偶校验为无，十六进制命令为 123456

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"device":"/dev/ttyS0","data_rate":"9600","data_bits":"8","stop_bits":"1","parity_bits":"
n","command":"123456"}' https://64.104.163.36/api/1.0/rs232/configuration
```

回复

```
{"success":"true","updatedAt":"2011-04-21 07:11:28"}
```

参数限制

设备参数应该为以 /dev/ 开头的字符串。

data_rate 参数应该为集 {50, 75, 110, 134, 150, 200, 300, 600, 1200, 1800, 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400, 460800, 500000, 576000} 中的一个字符串。

data_bits 参数应该为 7 或 8。

stop_bits 参数应该为 1 或 2。

parity_bits 参数应该为 n、 o 和 e。

升级

注　Curl 用作请求 API 的一个示例工具。 10.140.44.134 用作云翼 300 系列交换机的 IP 地址的一个示例，而 cisco123! 用作 一个管理员密码示例。

升级镜像

示例：将 edge300-1.5.tar 的镜像与应用和配置一起升级。

请求

```
curl -k -X PUT -H 'password: cisco123!' -H 'Content-Type: application/json' -d
'{"version":"edge300-1.5.tar","with_app":"1","with_cfg":"1"}'
https://64.104.163.36/api/1.0/upgrade/images
```

回复

```
{"success":"true","updatedAt":"2011-04-21 07:11:28"}
```

参数限制

版本：软件包版本，它应该与 edge300-[VERSION].tar 匹配。

with_app：0、 1。 0 表示没有需要进行升级的应用，否则，它应该为 1。

with_cfg：0、 1。 0 表示没有需要进行升级的配置，否则，它应该为 1。

前提条件

执行升级操作时拥有超级用户权限。

如果需要一个配置，必须将软件包和配置保存在同一个 U 盘上。

获取升级日志

示例：将升级日志文件保存到当前本地目录

请求

```
curl -O -k -X GET -H 'password: cisco123!' https://64.104.163.34/api/1.0/upgrade/log
```

回复

无。

错误代码

表 6-1　错误代码和说明

错误代码	说明
001	密码无效
002	内容类型错误
003	此资源不支持所使用的 HTTP 方法
004	非法参数
005	执行命令时出错
006	无效的资源
101	Json：超出最大栈深度
102	Json：下溢或模式不匹配
103	Json：发现异常控制字符
104	Json：语法错误，JSON 格式不正确
105	Json：UTF-8 字符格式不正确，可能编码错误
106	Json：未知错误

从 SMI 服务器安装第三方应用

云翼 300 系列交换机支持从智能安装 (SMI) 服务器安装第三方应用。

思科所发布的原始云翼 300 系列交换机软件包包含一个默认第三方应用软件包，其中包括 Open Office 和 Chrome 浏览器。您可以修改第三方应用内容并重新打包它，以便您可以从 SMI 服务器将合作伙伴应用安装到云翼 300。

第三方软件镜像要求

下面是第三方应用镜像在云翼 300 系列交换机上运行的要求：

- 镜像必须是 *delivery.tar.gz 文件格式的单一软件包。
- 该镜像必须包含一个位于单独标头目录中的标头文件。标头文件的名称必须描述该镜像。
- 标头文件的名称还必须是镜像文件的名称。例如，如果第三方应用的标头文件是 3rd-app-edge300-0.2.5.0-delivery.header，则第三方应用镜像文件的名称必须是 3rd-app-edge300-0.2.5.0-delivery.tar.gz。

  下图显示了镜像软件包在 TFTP 服务器上解压缩并放在 /opt/Tftproot/image 目录后的目录结构。粗体文本部分必须匹配：

```
/opt/Tftproot
|---Image
|    |---OS
|    |    |-- os-edge300-0.2.5.0-delivery.tar.gz
|    |    |-- header/os-edge300-0.2.5.0-delivery.header
|    |    |-- root-edge300-0.2.5.0.tar.gz
|    |    |-- bzImage-21official-beta0.1
|    |---CiscoApp
|    |    |-- cisco-app-edge300-0.2.5.0-delivery.tar.gz
|    |    |-- header/cisco-app-edge300-0.2.5.0-delivery.header
|    |    |-- cisco-app-edge300-0.2.5.0.tar.gz
|    |---Partner
|    |    |-- 3rd-app-edge300-0.2.5.0-delivery.tar.gz
|    |    |-- header/3rd-app-edge300-0.2.5.0-delivery.header
|    |    |-- 3rd-app-edge300-0.2.5.0.tar.gz
```

- 标头文件必须指定如下字段，并且 IMAGE_TYPE、 CPU_TYPE 和 VIDEO_OUT 字段必须包含等号 (=) 之后显示的信息：

```
IMAGE_TYPE=3RD_APP
IMAGE_SIZE=
VERSION=
DDR=
SLC=
MLS=
CPU_CORE=
CPU_TYPE=CE4150
USB=
DOWN_PORTS=
UP_PORTS=
WIRELESS_AP=
BT=
ZIGBEE=
VIDEO_OUT=HDMI
```

下面是一个标头文件的示例：

```
IMAGE_TYPE=3RD_APP
IMAGE_VERSION=0.2.5.0
IMAGE_SIZE=1000K
DDR=1G
SLC=1G
MLC=1G
CPU_CORE=1
CPU_TYPE=CE4150
USB=2
DOWN_PORTS=4
UP_PORTS=1
WIRELESS_AP=0
BLUETOOTH=1
ZIGBEE=0
VIDEO_OUT=HDMI
IMAGE_NAME=3rd-app-edge300-0.2.5.0-delivery.tar.gz
```

安装第三方应用软件包

要生成 SMI 第三方应用软件包，请按下列步骤进行操作：

步骤 1　使用以下内容在 ${RELEASE_DEST} 下为第三方应用创建一个标头文件，文件名为 **3rd.header**：

注　${RELEASE_DEST} 可以是 Linux 计算机上您有写入权限的任何位置。

```
IMAGE_TYPE=3RD_APP
IMAGE_VERSION=RELEASEVERSION
IMAGE_SIZE=RELEASESIZEK
DDR=1G
SLC=1G
MLC=1G
CPU_CORE=1
CPU_TYPE=CE4150
USB=2
DOWN_PORTS=4
```

```
UP_PORTS=1
WIRELESS_AP=0
BLUETOOTH=1
ZIGBEE=0
VIDEO_OUT=HDMI
IMAGE_NAME=image/Partner/3rd-app-sunbird-RELEASEVERSION.tar.gz
```

步骤 2　通过思科提供的 SDK，使用命令 **tar -zcvf 3rd-app.tar.gz** 在 ${RELEASE_DEST} 下创建一个 3rd-app.tar.gz 文件。此文件包含要安装的应用。

您还需要两个脚本：pre_install.sh 和 startup.sh，这两个脚本是具有执行权限的可选文件。

a. 当 SMI 下载第三方应用时，它首先会执行 pre_install.sh，通过执行此文件，在该脚本中制定的任务（如配置应用环境和删除旧文件）可以在软件包提取之前完成。

b. SMI 将软件包提取到目标路径。

c. SMI 执行 post_install.sh 来提取该软件包。您添加了可以在此脚本中清除临时文件或创建桌面图标的命令。

对于每个第三方应用软件包，您都需要一个 pre_install.sh 脚本，并且对于其内的每个应用文件夹，您都需要一个 startup.sh 脚本。第三方应用的软件包结构表如下所示：

```
- 3rd-app.tar.gz
   - pre_install.sh
   - {YOUR_APP_FOLDER}
      - startup.sh
      - ......
   -......
```

步骤 3　使用以下脚本生成一个 3rd-app-sunbird-${RELEASE_VERSION}.tar.gz 文件。

```
cd ${RELEASE_DEST}
RELEASE_VERSION=1.5.1 #<-- please change the version as you wish, 1.5.1 Here is an
example.
APP_SIZE=`du -k 3rd-app.tar.gz | awk '{print $1}'`
mv 3rd-app.tar.gz 3rd-app-sunbird-${RELEASE_VERSION}.tar.gz
#Generate 3rd-app md5 checksum
mkdir header
openssl dgst -md5 3rd-app-sunbird-${RELEASE_VERSION}.tar.gz > header/md5.txt
#Generate a header file according to template, version and size info.
APP_RELEASE_HEADER=3rd-app-sunbird-${RELEASE_VERSION}-delivery.header
cp 3rd.header header/3rd-app-sunbird-${RELEASE_VERSION}-delivery.header
cd header
sed -i "s@RELEASEVERSION@${RELEASE_VERSION}@g" ${APP_RELEASE_HEADER}
sed -i "s@RELEASESIZE@${APP_SIZE}@g" ${APP_RELEASE_HEADER}
cd ../
#Generate needed 3rd-app-sunbird-${RELEASE_VERSION}.tar.gz file
tar czvf 3rd-app-sunbird-${RELEASE_VERSION}-delivery.tar.gz
3rd-app-sunbird-${RELEASE_VERSION}.tar.gz header/${APP_RELEASE_HEADER} header/md5.txt
rm -rf header 3rd-app-sunbird-${RELEASE_VERSION}.tar.gz
```

步骤 4　将 3rd-app-sunbird-${RELEASE_VERSION}.tar.gz 文件置于 SMI 服务器的 tftproot 目录中进行安装。有关安装的详细步骤，请参阅第 2 章，“ 配置智能安装网络”。目标目录是 TFTP 服务器的 /opt/Tftproot/image/Partner。

导入带客户端交换机信息的电子表格

第 2-17 页上的“ 导入客户端交换机列表” 一节解释了如何将带有客户端交换机信息的文件导入 GUI。本附录提供了一个提供有更详细步骤的示例。

要将电子表格导入 GUI，请执行以下步骤：

步骤 1 确保电子表格的第一行是标题行，不包括任何交换机信息。交换机信息可以从第二行开始。

步骤 2 将电子表格保存为 CSV 格式。

注 如果出现确认弹出窗口，请单击**确定**或**是**。

步骤 3 在“ 管理云翼” 屏幕中，单击 图标并选择保存的电子表格。

步骤 4 单击**上传**导入电子表格。表中会显示已导入的客户端交换机。

注 如果 MAC 地址 (MAC) 不是唯一的，屏幕上会显示一条警告消息。

注 MAC 地址必须由六组十六进制数字构成，每组两个数字，用冒号分隔。如果电子表格中的 MAC 地址格式不正确，屏幕上会显示一条警告消息。

为智能安装 GUI 设置镜像文件服务器

您可以为智能安装设置您自己的镜像文件服务器，而不使用 GUI 服务器。

注 使用 GUI 服务器作为本地镜像文件服务器（即 GUI 服务器与镜像文件服务器在同一台机器上运行），或者使用一个分布式镜像文件服务器。您*不能*将本地服务器和分布式服务器同时用作镜像文件服务器。

您可以在 Windows 或 Redhat Linux（例如 CentOS/Fedora）上设置镜像文件服务器。智能安装目前不能支持在 Ubuntu 上运行的镜像文件服务器。智能安装 GUI 支持以下两种类型的部署方案：

- 在 Window 2008 上设置镜像文件服务器
- 在 CentOS 6 上设置镜像文件服务器

在 Window 2008 上设置镜像文件服务器

要在 Windows 2008 上配置镜像文件服务器，请执行以下步骤：

步骤 1 在您希望的位置创建一个名为 Tftproot 的文件夹（例如，C:\Tftproot）。

步骤 2 在 Tftproot 文件夹下创建以下子文件夹结构：

```
/Tftproot
|---image
|    |---CiscoApp
|    |---FM_OS
|    |---Fonts
|    |---OS
|    |---Partner
|---imglist
|---sb_conf
```

步骤 3 使用以下步骤共享 Tftproot 文件夹：

a. 右击 Tftproot 文件夹并从“菜单”中选择**属性**。

b. 单击**共享**选项卡，然后单击**共享...** 按钮。您会看到“文件共享”对话框。

c. 点击**共享**按钮。您会看到显示“您的文件夹已共享”的屏幕。

注 您还可以与管理员组中的其他用户共享文件夹。用户密码*不能*包含逗号 (,)。

步骤 4　下载 TFTP 软件，例如， Tftpd32。

步骤 5　在 TFTP 软件中，将当前目录设置为 Tftproot 文件夹的路径（例如， C:\Tftproot）。

步骤 6　将镜像文件服务器添加到智能安装 GUI 中，用户名为“ administrator”，并设置密码。有关将镜像文件服务器添加到 GUI 的更多信息，请参阅第 2-13 页上的“ 创建镜像文件服务器” 一节。

在 CentOS 6 上设置镜像文件服务器

要在 CentOS 6 上配置镜像文件服务器，请执行以下步骤：

步骤 1　在终端输入以下命令创建 Tftproot 文件夹及其子文件夹：

```
mkdir -p /opt/Tftproot/sb_conf
mkdir -p /opt/Tftproot/imglist
mkdir -p /opt/Tftproot/image/CiscoApp
mkdir -p /opt/Tftproot/image/OS
mkdir -p /opt/Tftproot/image/FM_OS
mkdir -p /opt/Tftproot/image/Partner
mkdir -p /opt/Tftproot/image/Fonts
chown apache:apache /opt/Tftproot/*
chmod 777 /opt/Tftproot/ -R
```

步骤 2　输入以下命令安装 TFTP 软件：

```
yum -y install xinetd tftp tftp-server
/sbin/service xinetd start
sed -i "s/\(disable[\t]*= *\).*/\1no/" /etc/xinetd.d/tftp
sed -i "s/\(server_args[\t]*= *\).*/\1-s \/opt\/Tftproot -c/" /etc/xinetd.d/tftp
sed -i '$ a\/sbin/service xinetd start' /etc/rc.d/rc.local
sed -i "s/\(SELINUX=\).*/\1disabled/" /etc/selinux/config
sed -i '$ a\\/sbin\/chkconfig --level 2345 iptables off' /etc/rc.d/rc.local
/etc/init.d/iptables stop
```

步骤 3　输入以下命令设置 samba 帐户，按照您的喜好设置用户名（例如， smbusr）。密码*不能*包含逗号 (,)：

```
useradd smbusr
smbpasswd -a smbusr
enter the password:[Enter your password]
```

步骤 4　使用 vi/vim 或 nano 修改 /etc/samba/smb.conf，如下所示：

```
[Tftproot]
    path = /opt/Tftproot
    valid users = smbusr
    read only = No
    guest ok = Yes
    force create mode = 777
```

步骤 5　输入以下命令重启 samba 服务器：

```
service smb restart
```

注　默认情况下， samba 服务不会自动重启。有关配置重启后自动启动 samba 服务的更多信息，请参阅第 C-3 页上的“ 配置开机后自动启动 Samba 服务” 一节。

步骤 6　将镜像文件服务器添加到智能安装 GUI 中，用户名为“ smbusr”，并设置密码。有关将镜像文件服务器添加到 GUI 的更多信息，请参阅第 2-13 页上的“ 创建镜像文件服务器” 一节。

配置开机后自动启动 Samba 服务

要配置在运行级别 3 和运行级别 5 上启动后自动启动 samba 服务，请执行下列步骤：

步骤 1　输入以下命令，列出所有运行级别上自动启动的 samba (smb) 服务：

```
# chkconfig -list smb
smb 0:off 1:off 2:off 3:off 4:off 5:off 6:off
```

步骤 2　输入以下命令，使 samba 服务器在运行级别 3 和级别 5 上启动时自动启动：

```
# chkconfig -level 35 smb on
```

步骤 3　通过输入以下命令验证配置更改：

```
# chkconfig -list smb
smb 0:off 1:off 2:off 3:on 4:off 5:on 6:off
```

故障排除

一般故障排除

如果云翼 300 系列交换机在智能安装网络中存在问题（例如，升级失败），请按住交换机的“ 重置” 按钮。交换机将以出厂默认模式启动，然后连接至指挥交换机，并下载和安装最新镜像。

如果还是有问题，请按照如下故障排除指南进行操作：

步骤 1　连接到云翼 300 系列交换机（参见*云翼 300 系列交换机安装指南*）：

a. 使用 **ping** [*options*] **host** Linux 命令 ping 指挥交换机来验证连接。

b. 使用云翼 300 系列交换机上的 **ls** [*options*] [*names*] Linux 命令确保：

- 脚本目录 /scripts/smistart.sh 中存在 smistart.sh 脚本。
- tmp 目录 /tmp/smi.lease 中存在 smi.lease 文件。
- 某个目录中已有 dhclient-enter-hooks 脚本。

c. 如果有名称为 dhclient-enter-hooks 的脚本，但 tmp 目录中没有 smi.lease 文件，请确认：

- DHCP 客户端正在运行中，即已定义了 **dhclient** Linux 命令。
- DHCP 服务器正在运行。
- 交换机可以获取 DHCP 服务器的 IP 地址。

d. 如果交换机无法获取 DHCP 服务器的 IP 地址，请使用 **ifconfig** [*interface*] **ifconfig** [*interface address_family parameters addresses*] Linux 命令来定义 DHCP 服务器的 IP 地址。

步骤 2　在智能安装指挥交换机上：

- 确保交换机未丢失其指挥交换机配置。
- 确保在指挥交换机上配置了镜像列表文件和交换机配置文件。
- 输入 **show ip dhcp snooping binding** [*ip-address*] [*mac-address*] 用户 EXEC 命令 £¨ 以显示交换机的 DHCP 监听绑定数据库和配置信息。

步骤 3　在 TFTP 服务器上，确保：

- TFTP 服务器上有指挥交换机上配置的镜像列表文件。
- TFTP 服务器上有镜像列表文件中定义的镜像。
- TFTP 服务器上有指挥交换机配置文件。
- 在升级中必须替换旧镜像的某个新镜像其版本号与旧镜像不同，并且镜像列表文件中定义了该新镜像。

- 正确的硬件参数（包括关键字和值）定义在新影像列表文件中。该新影像在升级中必须替换某个旧镜像。

步骤 4　在交换机上，使用 **vi** [*options*] [*files*]、**cat** [*options*] [*files*] 或 **more** [*options*] [*files*] Linux 命令从 tmp 目录中检索系统日志 (smi_log) 文件。将文件发送给技术支持。

注　云翼 300 系列交换机使用 Linux 的系统日志功能将日志记录系统中的所有必要信息记录到一个内部 USB 磁盘中，并在其大小超过阈值大小时将日志上载到智能安装服务器。

故障排除软件升级

软件下载之后，交换机将重新启动以升级至该软件。如果软件下载失败，交换机将不会重新启动，系统日志文件中将保存一条错误消息。如果交换机连接了监视器，错误消息还会显示在监视器上。

如果软件下载成功，但下载的镜像或配置文件有瑕疵，请将组中的交换机与可以正常使用的镜像和配置文件重新关联。通知最终用户通过重新启动交换机或按“ 重置” 按钮再次升级交换机。

如果软件升级失败，例如，由于断电或者网络连接断开，交换机将保持出厂默认模式，并在系统日志文件中保存一条错误消息。如果交换机连接了监视器，错误消息还会显示在监视器上。要从失败的软件升级中恢复，最终用户需要重新启动交换机或按“ 重置” 按钮。

使用 USB 端口手动升级软件

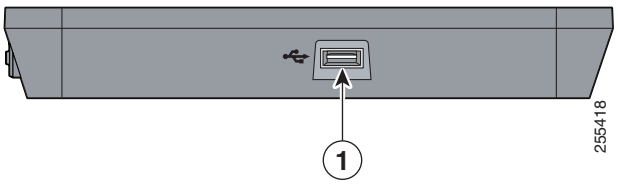
注意　从软件版本 1.0 升级到版本 1.1 之前，请从 GUI 中删除“ 工厂模式操作系统版本” 和“ 字体” 选择，然后应用更所作的更改。请参阅第 2-23 页上的“ 管理云翼配置文件” 一节。

注意　如果设备最初安装的版本为 1.5 或更高版本，请勿通过 USB 手动升级方式降至 1.4 或以下版本。否则，设备可能遭到严重损坏。

如果云翼 300 无法连接到智能安装指挥交换机，可以使用云翼 USB 智能安装工具通过 USB 闪盘升级或恢复系统固件。

使用云翼 300 侧面的 USB 端口执行 USB 智能安装升级。

图 D-1　云翼 300 系列交换机



1	USB 端口

格式化 USB 智能安装闪盘

步骤 1　将具有至少 1 GB 存储容量的 USB 闪盘格式化为 ext3 文件系统：

mkfs.ext3 /dev/sdb1

步骤 2　装载 USB 闪盘并将 smi-usb 镜像解压到该闪盘中：

sudo tar -zpxvf smi-usb-sunbird-1.1.0-delivery.tar.gz -C /media/sdb1

在云翼操作系统版本 1.1.0 或更高版本上使用 USB 智能安装

步骤 1　将所有 USB 闪存设备从云翼 300 交换机断开。将以太网电缆从千兆以太网（上行链路）端口断开。

步骤 2　启动云翼 300 交换机并进入用户桌面。

步骤 3　将 USB 智能安装闪盘插入侧面 USB 端口。

步骤 4　双击桌面上的“ 智能安装”图标。

步骤 5　在弹出的窗口中输入根用户密码，然后单击**确定**。

注　如果您不知道密码，请询问系统管理员。

主窗口显示了目前正在云翼 300 交换机上运行的固件版本，以及要从 USB 闪盘升级的固件镜像版本。

步骤 6　执行以下操作之一：

- 选择“ 常规升级”以升级系统。
- 选择“ 强制升级”将系统恢复到 USB 闪盘所提供的版本。

步骤 7　在“ 警告”窗口中单击**确定**。

注　如果不单击“ 确定”，系统在 10 秒内即会重启。

在升级过程中，电源 LED 将闪烁绿色。 20 至 40 分钟后，系统会使用新安装的固件正常重启。

注　琥珀色电源 LED 指示升级失败。

步骤 8　拔出 USB 智能安装闪盘。将以太网电缆插入千兆以太网（上行链路）端口。

强制在工厂模式下升级软件

要强制在工厂模式下升级软件，请执行以下步骤：

步骤 1　如果在云翼 300 系列交换机上有任何 USB 闪存驱动器，请将其全部断开。

步骤 2　将 USB 闪存驱动器插入云翼 300 系列交换机的 USB 端口侧。

步骤 3　请确保您插入的 USB 端口是侧面板（而不是有许多端口的前面板）上的唯一端口。

步骤 4　按“重置”按钮 4 秒钟可进入工厂模式。

步骤 5　请等待系统从 USB 闪存驱动器重新启动以进行升级。

步骤 6　在升级过程中，电源 LED 将为绿色并会闪烁。等待大约 10 分钟后，系统会使用新安装的固件重新启动。如果电源 LED 的颜色变为黄色，表明升级失败。

在云翼操作系统版本 1.0.0 上使用 USB 智能安装

步骤 1　将所有 USB 闪存设备从云翼 300 交换机断开。将以太网电缆从千兆以太网（上行链路）端口断开。

步骤 2　启动云翼 300 交换机并进入用户桌面。

步骤 3　将 USB 闪盘插入侧面 USB 端口。

步骤 4　当 USB 闪盘图标出现在桌面上时，双击该图标以查看 USB 闪盘的内容。

步骤 5　查找“智能安装”图标并双击。

步骤 6　在弹出的窗口中输入根用户密码，然后单击**确定**。

注　如果您不知道密码，请询问系统管理员。

主窗口显示云翼 300 交换机上正在运行的固件版本和要从 USB 闪盘升级的固件镜像版本。

步骤 7　执行以下操作之一：

- 选择“常规升级”以升级系统。
- 选择“强制升级”将系统恢复到 USB 闪盘所提供的版本。

步骤 8　在“警告”窗口中单击**确定**。

注　如果不单击“确定”，系统在 10 秒内即会重启。

在升级过程中，电源 LED 将闪烁绿色。20 至 40 分钟后，系统会使用新安装的固件正常重启。

注　琥珀色电源 LED 指示升级失败。

步骤 9　拔出 USB 智能安装闪盘。将以太网电缆插入千兆以太网（上行链路）端口。

注　如果“ 智能安装”窗口显示“ PIC 版本过低”，则表明您的云翼 300 硬件版本过低，无法支持 USB 智能安装工具。