

通过 SNMP 在 CatcOs 交换机和 TFTP 服务器之间移动文件和镜像

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[规则](#)

[将配置从TFTP服务器复制到运行CatOS的Catalyst交换机](#)

[逐步指导](#)

[检查结果](#)

[排除流程故障](#)

[将配置从运行CatOS的Catalyst交换机复制到TFTP服务器](#)

[逐步指导](#)

[验证流程](#)

[排除流程故障](#)

[将系统软件映像从TFTP服务器复制到运行CatOS的Catalyst交换机](#)

[逐步指导](#)

[验证流程](#)

[排除流程故障](#)

[将系统软件映像从运行CatOS的Catalyst交换机复制到TFTP服务器](#)

[逐步指导](#)

[验证流程](#)

[排除流程故障](#)

[UNIX 脚本示例](#)

[附录A - MIB对象详细信息](#)

[相关信息](#)

简介

本文档介绍如何在具有Catalyst操作系统(CatOS)的交换机和具有简单网络管理协议(SNMP)的UNIX上的简单文件传输协议(TFTP)服务器之间移动配置文件和系统软件映像。

先决条件

要求

验证您能从Catalyst交换机ping TFTP服务器的IP地址：

```
Cat6509> (enable) ping 171.68.191.135
```

```
!!!!!
```

```
----171.68.191.135 PING Statistics----
```

```
5 packets transmitted, 5 packets received, 0% packet loss
```

```
round-trip (ms)  min/avg/max = 2/2/2
```

这些步骤包括：

- 不适用于基于Cisco IOS®软件（如Catalyst 2900/3500XL系列）的Catalyst交换机。
- 不适用于带有Cisco IOS软件的Catalyst 6000系列MSFC和MSFC2模块。
- 如果交换机上未配置或未知SNMP读写社区字符串，则不适用。有关如何配置[SNMP社区字符串](#)的详细过程，请参阅[如何配置SNMP社区字符串](#)。
- 基于NET-SNMP(以前称为UCD-SNMP)实用程序的命令行语法。如果您有其他SNMP应用程序，如HP Open View或NetView，则语法可能与这些示例不同。
- 基于[CISCO-STACK-MIB](#)，自初始管理引擎模块软件版本起，Catalyst OS支持该MIB。请参阅Cisco.com上的“产品支持的MIB”页，以验证您的交换机是否支持[CISCO-STACK-MIB](#)。使用来自此MIB的以下MIB对象：

MIB 对象名称	OID
fftpHost	.1.3.6.1.4.1.9.5.1.5.1
fftpFile	.1.3.6.1.4.1.9.5.1.5.2
fftpModule	.1.3.6.1.4.1.9.5.1.5.3
fftpAction	.1.3.6.1.4.1.9.5.1.5.4
fftpResult	.1.3.6.1.4.1.9.5.1.5.5

有关这些MIB对象及其定义的详细信息，请参阅[附录A](#)。

使用的组件

本文档中的信息基于仅运行Catalyst OS软件的交换机。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您使用的是真实网络，请确保您已经了解所有命令的潜在影响。

规则

在所有示例中，这些值用于说明：

- 带CatOS的Catalyst 6509交换机
- 172.16.99.66 = Catalyst 6509交换机的IP地址
- private = SNMP读写社区字符串。使用在交换机上配置的读写字符串。在交换机CLI上使用show snmp命令来验证这一点。
- public = SNMP只读社区字符串。使用在交换机上配置的只读字符串。在交换机CLI上使用show snmp命令来验证这一点。
- 171.68.191.135 = TFTP服务器的IP地址

以下是以下示例中snmpset和snmpwalk命令的语法：

```
snmpset [options...] <hostname> {<community>} [<objectID> <type> <value> ...] snmpwalk  
[options...] <hostname> {<community>} [<objectID>]
```

有关文档规则的详细信息，请参阅 [Cisco 技术提示规则](#)。

将配置从TFTP服务器复制到运行CatOS的Catalyst交换机

这些步骤将指导您完成复制配置文件的过程。

逐步指导

请完成以下步骤：

1. 在TFTP服务器/tftpboot目录中创建新文件switch-config。在UNIX上，使用以下语法：**按**

<filename>。

```
touch switch-config
```

2. 将文件的权限更改为777。请使用以下语法：**chmod <permissions> <filename>**。

```
chmod 777 switch-config
```

3. 使用tftpHost MIB对象定义TFTP服务器的IP地址：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.1.0 s 171.68.191.135
enterprises.9.5.1.5.1.0 = "171.68.191.135"
```

4. 使用tftpFile MIB对象定义用于复制配置的TFTP文件名：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.2.0 s switch-config
enterprises.9.5.1.5.2.0 = switch-config
```

5. 在Catalyst交换机上选择要传送配置的模块，并带有tftpModule MIB对象。选择Supervisor模块，而不是MSFC或MSFC2模块，否则它会失败。在交换机CLI上使用**show module**命令验证**snmpset**命令的模块编号是否正确。典型输出为：

Mod	Slot	Ports	Module-Type	Model	Sub	Status
2	2	2	1000BaseX Supervisor	WS-X6K-SUP1A-2GE	yes	ok
16	2	1	Multilayer Switch Feature	WS-F6K-MSFC	no	OK
....						

--<snip>--

在输出示例中，Supervisor模块编号为2，位于插槽编号2中。使用2定义tftpModule MIB对象：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.3.0 i 2
enterprises.9.5.1.5.3.0 = 2
```

6. 使用tftpAction MIB对象定义要从TFTP服务器传输到MIB对象值为2 = downloadConfig的交换机的交换机配置文件。请参阅附录A中的MIB对象[详细信息](#)：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.4.0 i 2
enterprises.9.5.1.5.4.0 = 2
```

检查结果

要验证这些操作的结果，请执行以下步骤之一：

1. 轮询tftpGrp(.1.3.6.1.4.1.9.5.1.5)MIB对象，并将结果与附录A[进行比较](#)：

```
% snmpwalk 172.16.99.66 public .1.3.6.1.4.1.9.5.1.5
enterprises.9.5.1.5.1.0 = "171.68.191.135"
!--- IP address of the TFTP server enterprises.9.5.1.5.2.0 = "switch-config" !--- name of
the switch configuration file enterprises.9.5.1.5.3.0 = 2 !--- Module number. In this case,
Supervisor module enterprises.9.5.1.5.4.0 = 2 !--- TFTP action. 2 = Download configuration
from TFTP server to the switch enterprises.9.5.1.5.5.0 = 2 !--- Result of the TFTP action,
2 = Success
```

2. 轮询tftpResult MIB对象，并将输出与附录A中的MIB对象详细[信息比较](#)：

```
% snmpwalk 172.16.99.66 public .1.3.6.1.4.1.9.5.1.5.5
enterprises.9.5.1.5.5.0 = 2
!--- Result of the TFTP action, 2 = Success
```

排除流程故障

如果下载成功，则MIB对象输出等于2（或成功）。如果您收到其他输出，请将其与tftpResult对象的[附录A进行比较，并采取相应的步骤。](#)

将配置从运行CatOS的Catalyst交换机复制到TFTP服务器

这些步骤将指导您完成复制配置文件的过程。

逐步指导

请完成以下步骤：

1. 在TFTP服务器/tftpboot目录中创建新文件switch-config。在UNIX上，使用以下语法：[按<filename>](#)。

```
touch switch-config
```

2. 使用以下语法将文件的权限更改为777：[chmod <permissions> <filename>](#)。

```
chmod 777 switch-config
```

3. 使用tftpHost MIB对象定义TFTP服务器的IP地址。语法为：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.1.0 s 171.68.191.135
enterprises.9.5.1.5.1.0 = "171.68.191.135"
```

4. 使用tftpFile MIB对象定义用于复制配置的TFTP文件名：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.2.0 s switch-config
enterprises.9.5.1.5.2.0 = switch-config
```

5. 在Catalyst交换机上选择要传送配置的模块，并带有tftpModule MIB对象。选择Supervisor模块，而不是MSFC或MSFC2模块，否则它会失败。在交换机CLI上使用show module命令验证snmpset命令的模块编号是否正确。典型输出为：

Mod	Slot	Ports	Module-Type	Model	Sub	Status
2	2	2	1000BaseX Supervisor	WS-X6K-SUP1A-2GE	yes	ok
16	2	1	Multilayer Switch Feature	WS-F6K-MSFC	no	OK
....						

--<snip>--

在输出示例中，Supervisor模块编号为2，位于插槽编号2中。使用2可定义tftpModule MIB对象：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.3.0 i 2
enterprises.9.5.1.5.3.0 = 2
```

6. 使用tftpAction MIB对象以定义交换机配置文件将从TFTP服务器传输到MIB对象值为3 = uploadConfig的交换机。请参阅附录A中的MIB对象[详细信息](#)：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.4.0 i 3
enterprises.9.5.1.5.4.0 = 3
```

验证流程

要验证这些操作的结果，请执行以下步骤之一：

1. 轮询tftpGrp(.1.3.6.1.4.1.9.5.1.5)MIB对象，并将结果与附录A[进行比较](#):

```
% snmpwalk 172.16.99.66 public .1.3.6.1.4.1.9.5.1.5
enterprises.9.5.1.5.1.0 = "171.68.191.135"
!--- IP address of the TFTP server enterprises.9.5.1.5.2.0 = "switch-config" !--- name of
the switch configuration file enterprises.9.5.1.5.3.0 = 2 !--- Module number. In this case,
Supervisor module enterprises.9.5.1.5.4.0 = 1 !--- TFTP action enterprises.9.5.1.5.5.0 = 2
!--- Result of the TFTP action, 2 = Success
```

2. 轮询tftpResult MIB对象，并将输出与附录A中的MIB对象详细[信息比较](#):

```
% snmpwalk 172.16.99.66 public .1.3.6.1.4.1.9.5.1.5.5
enterprises.9.5.1.5.5.0 = 2
!--- Result of the TFTP action, 2 = Success
```

排除流程故障

如果下载成功，则MIB对象输出等于2（或成功）。如果您收到其他输出，请将其与tftpResult对象的[附录A进行比较，并采取相应的步骤](#)。

注意： 此过程从交换机传输默认和非默认配置，如启用模式下交换机CLI上show config all命令的输出所示。交换机上的show config命令仅显示非默认配置。

将系统软件映像从TFTP服务器复制到运行CatOS的Catalyst交换机

这些步骤将指导您完成复制软件映像的过程。

逐步指导

请完成以下步骤：

1. 下载正确的Supervisor映像文件并将其放置到TFTP服务器的/tftpboot目录中。在本示例中，使用cat6000-sup.5-4-2a.bin进行图示。
2. 使用以下语法将文件的权限更改为777:chmod <permissions> <filename>。

```
chmod 777 cat6000-sup.5-4-2a.bin
```

3. 定义使用tftpHost MIB对象的TFTP服务器的IP地址：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.1.0 s 171.68.191.135
enterprises.9.5.1.5.1.0 = "171.68.191.135"
```

4. 定义用于复制映像文件的TFTP文件名：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.2.0 s cat6000-sup.5-4-2a.bin
enterprises.9.5.1.5.2.0 = "cat6000-sup.5-4-2a.bin"
```

5. 在本示例中，Supervisor模块编号为2，并且位于插槽编号2中，如show module命令输出所示。使用2以定义tftpModule MIB对象：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.3.0 i 2
enterprises.9.5.1.5.3.0 = 2
```

这意味着TFTP服务器上/tftpboot目录中的CatOS映像将传输到Supervisor模块闪存，如show flash命令的输出所示。

6. 使用tftpAction MIB对象以定义将映像文件从TFTP服务器传输到MIB对象值为4 = downloadSw的交换机。请参阅附录A中的MIB对象[详细信息](#):

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.4.0 i 4
enterprises.9.5.1.5.4.0 = 4
```

验证流程

要验证此操作的结果，请执行以下步骤之一：

1. 轮询tftpGrp(.1.3.6.1.4.1.9.5.1.5)MIB对象，并将结果与附录A[进行比较](#)：

```
% snmpwalk 172.16.99.66 public .1.3.6.1.4.1.9.5.1.5
enterprises.9.5.1.5.1.0 = "171.68.191.135"
!--- IP address of the TFTP server enterprises.9.5.1.5.2.0 = "cat6000-sup.5-4-2a.bin" !---
name of the switch image file enterprises.9.5.1.5.3.0 = 0 enterprises.9.5.1.5.4.0 = 4 !---
TFTP action, 4 = downloadSw enterprises.9.5.1.5.5.0 = 1 !--- Result of the TFTP action, 1 =
In Process
```

注：最后一个条目显示图像传输正在进行。等待几分钟，然后再次轮询tftpResult MIB对象，以验证其已成功传输。此步骤可能需要几分钟才能完成，具体取决于映像文件大小（字节）。当映像传输过程正在进行时，如果在交换机上发出show flash命令，您将看到：

```
Cat6509> (enable) show flash
TFTP session in progress. Try again later.
```

2. 轮询tftpResult MIB对象，并将输出与附录A中的MIB对象详细[信息比较](#)：

```
% snmpwalk 172.16.99.66 public .1.3.6.1.4.1.9.5.1.5.5
enterprises.9.5.1.5.5.0 = 2
!--- Result of the TFTP action, 2 = Success
```

排除流程故障

如果下载成功，则MIB对象输出等于2（或成功）。如果您收到其他输出，请将其与tftpResult对象的[附录A进行比较，并采取相应的步骤。](#)

成功完成映像传输后，验证映像文件大小（字节）是否与TFTP服务器(本例中为cat6000-sup.5-4-2a.bin)中的文件(show flash命令输出中显示的文件)匹配。

将系统软件映像从运行CatOS的Catalyst交换机复制到TFTP服务器

这些步骤将指导您完成复制软件映像的过程。

逐步指导

请完成以下步骤：

1. 在TFTP服务器的/tftpboot目录中创建一个新文件image.bin。在UNIX上，使用以下语法：按<filename>。使用.bin作为文件扩展名。

```
touch image.bin
```

2. 使用以下语法将文件的权限更改为777:chmod <permissions> <filename>。

```
chmod 777 image.bin
```

3. 使用tftpHost MIB对象定义TFTP服务器的IP地址：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.1.0 s 171.68.191.135
enterprises.9.5.1.5.1.0 = "171.68.191.135"
```

4. 定义将用于复制映像文件的TFTP文件名和tftpFile MIB对象：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.2.0 s image.bin
enterprises.9.5.1.5.2.0 = "image.bin"
```

5. 在本示例中，Supervisor模块编号为2，并且位于插槽编号2中，如show module命令输出所示

。使用2定义tftpModule MIB对象：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.3.0 i 2
enterprises.9.5.1.5.3.0 = 2
```

这意味着在闪存中的Supervisor模块上运行的CatOS映像会传输到TFTP服务器，如show flash命令的输出所示。

6. 使用tftpAction MIB对象以定义将映像文件从TFTP服务器传输到MIB对象值为5 = uploadSw的交换机。请参阅附录A中的MIB对象[详细信息](#)：

```
% snmpset 172.16.99.66 private .1.3.6.1.4.1.9.5.1.5.4.0 i 5
enterprises.9.5.1.5.4.0 = 5
```

验证流程

要验证这些操作的结果，请执行以下步骤之一：

1. 轮询tftpGrp(.1.3.6.1.4.1.9.5.1.5)MIB对象，并将结果与附录A[进行比较](#)：

```
% snmpwalk 172.16.99.66 public .1.3.6.1.4.1.9.5.1.5
enterprises.9.5.1.5.1.0 = "171.68.191.135"
!--- IP address of the TFTP server enterprises.9.5.1.5.2.0 = "image.bin" !--- name of the
switch image file enterprises.9.5.1.5.3.0 = 2 !--- Module number. In this case, Supervisor
module enterprises.9.5.1.5.4.0 = 5 !--- TFTP action, 5 = uploadSw enterprises.9.5.1.5.5.0 =
1 !--- Result of the TFTP action, 1 = In Process
```

注：最后一个条目显示图像传输正在进行。等待几分钟，然后再次轮询tftpResult MIB对象，以验证其已成功传输。此步骤可能需要几分钟才能完成，具体取决于映像文件大小（字节）。

2. 轮询tftpResult MIB对象，并将输出与附录A中的MIB对象详细[信息比较](#)：

```
% snmpwalk 172.16.99.66 public .1.3.6.1.4.1.9.5.1.5.5
enterprises.9.5.1.5.5.0 = 2 --> Result of the TFTP action, 2 = Success
```

排除流程故障

如果下载成功，则MIB对象输出等于2（或成功）。如果您收到其他输出，请将其与tftpResult对象的[附录A进行比较，并采取相应的步骤](#)。

成功完成映像传输后，验证TFTP服务器(本例中为image.bin)中文件的show flash命令输出中显示的映像文件大小（字节）**是否匹配**。

注意：如果闪存中有多个映像(show flash)，则只有启动Supervisor模块的映像才会通过此过程传输到TFTP服务器。使用show boot命令查看BOOT变量=，该变量显示Supervisor模块使用闪存中的哪个映像进行启动。有关详细[信息，请参阅升级软件映像和使用Catalyst交换机上的配置文件](#)。

UNIX 脚本示例

注意：这些脚本仅作为示例提供，Cisco Systems不以任何方式支持。

在交换机上自动执行配置文件和Cisco IOS迁移的脚本

```
#!/bin/sh
# Script to automate config file & IOS migration of switches
# supporting STACK-MIB including 5000, 5500, 1400, 2900, 1200
if [ ! -f SW ] ;
then
echo
```



```

echo "File SW does not exist!!!"
echo
echo "Syntax is 'switch.sh'"
echo "where each line in file SW lists:"
echo "Switchname Filename Serverip Module# Moduleaction Community"
echo
echo "Switchname must resolve"
echo "Filename must exist in server tftpboot directory 777"
echo "Serverip is the ip of the server for the file"
echo "Module# is usually '1'"
echo "Module action is as per STACK-MIB: "
echo "- 2 - config file - server > switch"
echo "- 3 - config file - switch > server"
echo "- 4 - software image - server > switch"
echo "- 5 - software image - switch > server"
echo "Community is *write* community"
echo
exit
fi
cat SW |
while read SW
do
    SWNAME=\Qecho $SW | cut -d' ' -f 1\Q
    FILE=\Qecho $SW | cut -d' ' -f 2\Q
    SERVER=\Qecho $SW | cut -d' ' -f 3\Q
    MODULE=\Qecho $SW | cut -d' ' -f 4\Q
    ACTION=\Qecho $SW | cut -d' ' -f 5\Q
    CMTY=\Qecho $SW | cut -d' ' -f 6\Q
    echo
    echo $SWNAME
    echo $FILE
    echo $SERVER
    echo $MODULE
    echo $ACTION
    echo $CMTY
    echo
    # '-t #' can be modified to adjust timeout
    snmpset -t 100 -c $CMTY $SWNAME .1.3.6.1.4.1.9.5.1.5.1.0 octetstring $SERVER
    sleep 5
    snmpset -t 100 -c $CMTY $SWNAME .1.3.6.1.4.1.9.5.1.5.2.0 octetstring $FILE
    sleep 5
    snmpset -t 100 -c $CMTY $SWNAME .1.3.6.1.4.1.9.5.1.5.3.0 integer $MODULE
    sleep 5
    snmpset -t 100 -c $CMTY $SWNAME .1.3.6.1.4.1.9.5.1.5.4.0 integer $ACTION
sleep 60
echo
echo Check Progress...
echo
echo
echo "Switch $SWNAME: \\c"; snmpget -t 100 -c $CMTY $SWNAME .1.3.6.1.4.1.9.5.1.5.5.0 | cut
-d": " -f 3
done

```

交换机期望脚本在交换机上执行特定命令

```

#!/usr/nms/bin/expect
# Above line points to your expect interpreter
# Add '-d' option to expect line above to enable debugging
# Tested on Cat5000 with regular login; no error-checking
# except for number arguments, but will timeout on failure.
# Tacacs+ lines left in for future releases
set argc [llength $argv]
if { $argc < 4 } {

```



```

    puts "Syntax is:"
    puts "(For system with no Tac+)"
    puts "switch.exp destination \"command\" vtypassword enapassword"
    exit 0 }
set destination [lindex $argv 0]
puts -nonewline "Where we're going:  "
puts $destination
set command [lindex $argv 1]
puts -nonewline "What we're doing:  "
puts $command
set vtypassword [lindex $argv 2]
puts -nonewline "What our password is (vty):  "
puts $vtypassword
set enapassword [lindex $argv 3]
puts -nonewline "What our password is (enable):  "
puts $enapassword
# username only for Tac+
set username [lindex $argv 4]
puts -nonewline "What our username is if Tac+:  "
puts $username
#
set timeout 10
spawn telnet $destination
expect {
    "Enter password:" {
        send "$vtypassword\r" }
    "Username:" {
        send "$username\r"
        exec sleep 1
        expect "Password:"
        send "$vtypassword\r"
    }
}
# Look for non-enable router 'prompt>'
expect -re "(^.*)(\r\n\[^\ ]+> \$)"
# Get into enable mode
send "en\r"
expect {
    "password: " {
        send "$enapassword\r" }
    "Username:" {
        send "$username\r"
        exec sleep 1
        expect "Password:"
        send "$enapassword\r" }
}
# Look for enable router 'prompt#'
expect -re "(^.*)(\r\n\[^\ ]+(enable) \$)"
# Send the command
send "$command\r"
expect {
    -re "(^.*)(\r\n\[^\ ]+(enable) \$)" {
        append buffer $expect_out(1,string)
    }
    -re "(^.*)(\r\n\ --More-- \$)" {
        append buffer $expect_out(1,string)
        send " "
    }
    -re "(^.*)(\r\n\ --More-- \$)" {
        append buffer $expect_out(1,string)
        send " "
    }
}
# Done with command - disable prior to exit

```

```
send "disable\r"
expect -re "(^.*)(\r\n\[^\ \]+ \\\$)"
exec sleep 1
send "logout"
```

Perl脚本通过SNMP显示与“show cam dynamic”相同的输出

```
#!/usr/local/bin/perl
open(TABLE, "bridge-table.csv") || die "Cant' open file: $!\n";
while (<TABLE>) {
($vlan, $unicast_mac, $mod_ports) = split (/,/, $_);
write;
}
exit;

format STDOUT =
set cam permanent @<<<<<<<<<<<<<<<<<<<<<< @<<< @<
$unicast_mac, $mod_ports, $vlan
```

附录A - MIB对象详细信息

对象	tftpHost
OID	.1.3.6.1.4.1.9.5.1.5.1
类型	显示字符串
权限	读写
语法	二进制八位数字符串(0.64)
状态	当前
MIB	CISCO-STACK-MIB
描述	用于TFTP传输或存储设备传输的源/目标主机的名称。如果名称用于TFTP传输，则可以是IP地址或主机名。如果存储设备传输的名称为deviceName: (例如 slot0:、 slot1 :)
树中的OID	::= { iso(1)org(3)dod(6)internet(1)private(4)enterprises(1)cisco(9)workgroup(5)ciscoStackMIB(1)tftpGrp(5)1 }
对象	tftpFile
OID	.1.3.6.1.4.1.9.5.1.5.2
类型	显示字符串
权限	读写
语法	二进制八位数字符串(0.64)
状态	当前
MIB	CISCO-STACK-MIB

描述	用于TFTP传输或存储设备传输的文件名。
树中的OID	::= { iso(1)org(3)dod(6)internet(1)private(4)enterprises(1)cisco(9)workgroup(5)ciscoStackMIB(1)tftpGrp(5)2 }
对象	tftpModule
OID	.1.3.6.1.4.1.9.5.1.5.3
类型	整数
权限	读写
状态	当前
范围	0 - 16
MIB	CISC O-STACK-MIB
描述	传输模块的哪个代码/配置。
树中的OID	::= { ISO(1)org(3)DOD(6)Internet(1)私有(4)企业(1)cisco(9)workgroup(5)ciscoStackMIB(1)tftpGrp(5)3 }
对象	tftpAction
OID	.1.3.6.1.4.1.9.5.1.5.4
类型	整数
权限	读写
状态	当前
价值	<ol style="list-style-type: none"> 1. other (其他) 2. downloadConfig 3. uploadConfig 4. downloadSw 5. uploadSw 6. downloadFw 7. uploadFw
MIB	CISCO-STACK-MIB
描述	<p>如果将此对象设置为可接受值之一，它会使用tftpHost、tftpFile和tftpModule中提供的信息启动请求的操作。</p> <p>downloadConfig(2):从主机/文件上传接收配置</p> <p>uploadConfig(3):将配置发送到主机/文件</p> <p>downloadSw(4):从主机/文件上传软件</p> <p>uploadSw(5):接收软件映像：将软件映像发送到主机/文件</p> <p>downloadFw(6):从主机/文件上传接收固件映像</p> <p>uploadFw(7):向主机/文件发送固件映像</p> <p>如果将此对象设置为任何其他值，则会出现错误。</p>
树中	::= { ISO(1)org(3)DOD(6)Internet(1)私有(4)企业(1)cisco(9)workgroup(5)ciscoStackMIB(1)tftpGrp(5)4 }

的 O I D	
对象	tftpResult
OID	.1.3.6.1.4.1.9.5.1.5.5
类型	整数
权限	只读
状态	当前
价值	<ol style="list-style-type: none"> 1. 正在进行 2. 成功 3. noResponse 4. tooManyRetries 5. noBuffers 6. noProcesses 7. badChecksum 8. badLength 9. badFlash 10. serverError 11. userCancelled 12. 错误代码 13. fileNotFound 14. invalidTftpHost 15. invalidTftpModule 16. accessViolation 17. unknownStatus 18. invalidStorageDevice 19. insufficientSpaceOnStorageDevice 20. insufficientDramSize 21. 不兼容映像
MIB	CISCO-STACK-MIB
描述	包含最后一个TFTP操作请求的结果
树中的 的 OID	::= { ISO(1)org(3)DOD(6)Internet(1)私有(4)企业 (1)cisco(9)workgroup(5)ciscoStackMIB(1)tftpGrp(5))5 }

相关信息

- [通过 SNMP 在路由器和 TFTP 服务器之间移动文件和镜像](#)
- [思科MIB下载](#)
- [技术支持和文档 - Cisco Systems](#)