

Déplacement de fichiers et d'images entre un commutateur CatOS et un serveur TFTP via SNMP

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Conventions](#)

[Copiez une configuration du serveur TFTP sur le commutateur de Catalyst exécutant CatOS](#)

[Instructions pas à pas](#)

[Vérifiez les résultats](#)

[Dépannez le processus](#)

[Copiez une configuration du commutateur de Catalyst exécutant CatOS sur le serveur TFTP](#)

[Instructions pas à pas](#)

[Vérifiez le processus](#)

[Dépannez le processus](#)

[Copiez une image du logiciel système du serveur TFTP sur le commutateur de Catalyst exécutant CatOS](#)

[Instructions pas à pas](#)

[Vérifiez le processus](#)

[Dépannez le processus](#)

[Copiez une image du logiciel système du commutateur de Catalyst exécutant CatOS sur le serveur TFTP](#)

[Instructions pas à pas](#)

[Vérifiez le processus](#)

[Dépannez le processus](#)

[Exemple de script Unix](#)

[Annexe A — Détails d'objet MIB](#)

[Informations connexes](#)

Introduction

Ce document décrit comment déplacer des fichiers de configuration et des images du logiciel système entre un commutateur avec les systèmes d'exploitation de Catalyst (CatOS) et un serveur de Protocole TFTP (Trivial File Transfer Protocol) sur l'UNIX avec le Protocole SNMP (Simple Network Management Protocol).

Conditions préalables

Conditions requises

Vérifiez que vous pouvez cingler l'adresse IP du serveur TFTP du commutateur de Catalyst :

```
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
```

Ces procédures sont :

- Pas applicable pour des Commutateurs de Catalyst basés sur le logiciel de Cisco IOS®, tel que la gamme du Catalyst 2900/3500XL.
- Pas applicable pour les modules MSFC et MSFC2 de gamme Catalyst 6000 avec le logiciel de Cisco IOS.
- Pas applicable si la chaîne de caractères de la communauté en lecture-écriture SNMP n'est pas configurée ou est connue sur le commutateur. Référez-vous à [comment configurer des chaînes de caractères de la communauté SNMP](#) pour la procédure détaillée sur la façon dont configurer les chaînes de caractères de la communauté SNMP.
- Basé sur la ligne de commande syntaxe d'utilitaires [NET-SNMP](#) (précédemment connu sous le nom d'UCD-SNMP). [Si vous avez le quelque d'autres applications SNMP, telles que la vue de HP ou le NetView ouverte, la syntaxe pourraient être différentes de ces exemples.](#)
- Basé sur [CISCO-STACK-MIB](#), qui est pris en charge par le SYSTÈME D'EXPLOITATION de Catalyst puisque la version de logiciel du module initiale de superviseur. Référez-vous à la page de [sous-produit prise en charge par MIB](#) sur Cisco.com pour vérifier que votre commutateur prend en charge [CISCO-STACK-MIB](#). Ces objets MIB de ce MIB sont utilisés :

Nom d'objet MIB	OID
tftpHost	.1.3.6.1.4.1.9.5.1.5.1
tftpFile	.1.3.6.1.4.1.9.5.1.5.2
tftpModule	.1.3.6.1.4.1.9.5.1.5.3
tftpAction	.1.3.6.1.4.1.9.5.1.5.4
tftpResult	.1.3.6.1.4.1.9.5.1.5.5

Voir l'[annexe A](#) pour plus d'informations sur ces objets MIB avec des définitions.

Composants utilisés

Les informations dans ce document sont basées sur les Commutateurs qui exécutent seulement le logiciel Catalyst OS.

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

Conventions

Dans tous les exemples, ces valeurs sont utilisées pour l'illustration :

- Commutateur du Catalyst 6509 avec CatOS
- **172.16.99.66** = adresse IP du commutateur du Catalyst 6509
- **privé** = chaîne de caractères de la communauté en lecture-écriture SNMP. Utilisez la chaîne en lecture/écriture configurée sur votre commutateur. Vérifiez ceci avec la commande de **show snmp** sur le commutateur CLI.
- **public** = chaîne de caractères de la communauté en lecture seule SNMP. Utilisez la chaîne en lecture seule configurée sur votre commutateur. Vérifiez ceci avec la commande de **show snmp** sur le commutateur CLI.
- **171.68.191.135** = adresse IP du serveur TFTP

C'est la syntaxe pour les commandes de **snmpset** et de **snmpwalk** dans ces exemples :

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

Pour plus d'informations sur les conventions utilisées dans ce document, reportez-vous à [Conventions relatives aux conseils techniques Cisco](#).

Copiez une configuration du serveur TFTP sur le commutateur de Catalyst exécutant CatOS

Ces étapes vous guident par le processus pour copier un fichier de configuration.

Instructions pas à pas

Procédez comme suit :

1. Créez un nouveau fichier, **commutateur-config**, dans le répertoire de **/tftpboot** de serveur TFTP. Sur l'UNIX, utilisez cette syntaxe : **<filename de toucher >**.

```
touch switch-config
```

2. Changez les autorisations du fichier à **777**. Utilisez cette syntaxe : **<permissions > <filename de chmod >**.

```
chmod 777 switch-config
```

3. Définissez l'adresse IP du serveur TFTP avec l'objet MIB de **tftpHost** :

```
chmod 777 switch-config
```

4. Définissez le nom du fichier TFTP que vous utiliserez pour copier la configuration, avec l'objet **tftpFile** MIB :

```
chmod 777 switch-config
```

5. Sélectionnez le module sur le commutateur de Catalyst où la configuration sera fournie, avec l'objet MIB de **tftpModule**. Sélectionnez le module de superviseur et pas le module MSFC ou MSFC2, autrement il échoue. Vérifiez le numéro de module correct pour la commande de **snmpset** avec une commande de **show module** sur le commutateur CLI. Un résultat typique est :

```

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>--

```

Dans la sortie témoin, le numéro de module de superviseur est 2 et est dans l'emplacement le numéro 2. Utilisation **2** afin de définir l'objet MIB de **tftpModule** :

```

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>--

```

6. Employez l'objet MIB de **tftpAction** afin de définir le fichier de configuration de commutateur qui doit être transféré du serveur TFTP vers le commutateur avec la valeur d'objet MIB de **2 = downloadConfig**. Voyez le MIB objecter des détails dans l'[annexe A](#) :

```

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>--

```

[Vérifiez les résultats](#)

Afin de vérifier les résultats de ces exécutions, exécutez une de ces étapes :

1. Votez le .1.3.6.1.4.1.9.5.1.5) objet MIB de **tftpGrp** (et comparez les résultats à l'[annexe A](#) :

```

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>--

```

2. Votez l'objet MIB de **tftpResult** et comparez la sortie aux détails d'objet MIB dans l'[annexe A](#) :

```

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>--

```

[Dépannez le processus](#)

Si le téléchargement est réussi, la sortie d'objet MIB est égale à 2 (ou à succès). Si vous recevez

une autre sortie, comparez-la à l'[annexe A](#) pour l'objet de **tftpResult** et prenez les mesures appropriées.

[Copiez une configuration du commutateur de Catalyst exécutant CatOS sur le serveur TFTP](#)

Ces étapes vous guident par le processus pour copier un fichier de configuration.

[Instructions pas à pas](#)

Procédez comme suit :

1. Créez un nouveau fichier, commutateur-**config**, dans le répertoire de **/tftpboot** de serveur TFTP. Sur l'UNIX, utilisez cette syntaxe : **<filename de toucher >**.

```
touch switch-config
```

2. Changez les autorisations du fichier à **777** avec cette syntaxe : **<permissions > <filename de chmod >**.

```
chmod 777 switch-config
```

3. Définissez l'adresse IP du serveur TFTP avec l'objet MIB de **tftpHost**. La syntaxe est la suivante :

```
chmod 777 switch-config
```

4. Définissez le nom du fichier TFTP que vous utiliserez pour copier la configuration, avec l'objet **tftpFile** MIB :

```
chmod 777 switch-config
```

5. Sélectionnez le module sur le commutateur de Catalyst où la configuration sera fournie, avec l'objet MIB de **tftpModule**. Sélectionnez le module de superviseur et pas le module MSFC ou MSFC2, autrement il échoue. Vérifiez le numéro de module correct pour la commande de **snmpset** avec une commande de **show module** sur le commutateur CLI. Un résultat typique est :

```
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>--
```

Dans la sortie témoin, le numéro de module de superviseur est 2 et est dans l'emplacement le numéro 2. Utilisation 2 afin de définir l'objet MIB de **tftpModule** :

```
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>--
```

6. Employez l'objet MIB de **tftpAction** afin de définir que le fichier de configuration de commutateur doit être transféré du serveur TFTP vers le commutateur avec la valeur d'objet MIB de **3 = uploadConfig**. Voyez le MIB objecter des détails dans l'[annexe A](#) :

```

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>--

```

Vérifiez le processus

Afin de vérifier les résultats de ces exécutions, exécutez une de ces étapes :

1. Votez le **.1.3.6.1.4.1.9.5.1.5**) objet MIB de **tftpGrp** (et comparez les résultats à l'[annexe A](#) :

```

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>--

```

2. Votez l'objet MIB de **tftpResult** et comparez la sortie aux détails d'objet MIB dans l'[annexe A](#) :

```

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>--

```

Dépannez le processus

Si le téléchargement est réussi, la sortie d'objet MIB est égale à 2 (ou à succès). Si vous recevez une autre sortie, comparez-la à l'[annexe A](#) pour l'objet de **tftpResult** et prenez les mesures appropriées.

Note: Cette procédure transfère les deux le par défaut et les configurations non paramétrées par défaut du commutateur, comme vu dans la sortie du **show config toute la** commande sur le commutateur CLI dans le mode enable. La commande de **show config** sur le commutateur affiche seulement les configurations non paramétrées par défaut.

Copiez une image du logiciel système du serveur TFTP sur le commutateur de Catalyst exécutant CatOS

Ces étapes vous guident par le processus pour copier une image logicielle.

Instructions pas à pas

Procédez comme suit :

1. Téléchargez et placez le fichier d'image du Supervisor correct dans le répertoire de **/tftpboot**

sur le serveur TFTP. Dans cet exemple, **cat6000-sup.5-4-2a.bin** est utilisé pour l'illustration.

2. Changez les autorisations du fichier à **777** avec cette syntaxe : **<permissions > <filename de chmod >**.

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

3. Définissez l'adresse IP du serveur TFTP qui utilise l'objet MIB de **tftpHost** :

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

4. Définissez le nom du fichier TFTP que vous utiliserez pour copier le fichier d'image :

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

5. Dans cet exemple, le numéro de module de superviseur est 2 et est dans l'emplacement le numéro 2 comme vu dans la sortie de commande de **show module**. Utilisation **2** afin de définir l'objet MIB de **tftpModule** :

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

Ceci signifie que l'image de CatOS actuelle dans le répertoire de **/tftpboot** sur le serveur TFTP est transférée vers l'éclair de module de superviseur comme vu dans la sortie de la commande de **show flash**.

6. Employez l'objet MIB de **tftpAction** afin de définir que le fichier d'image est transféré du serveur TFTP vers le commutateur avec la valeur d'objet MIB de **4 = downloadSw**. Voyez le MIB objecter des détails dans l'[annexe A](#) :

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

Vérifiez le processus

Afin de vérifier les résultats de ces exécutions, exécutez une de ces étapes :

1. Votez le **.1.3.6.1.4.1.9.5.1.5**) objet MIB de **tftpGrp** (et comparez les résultats à l'[annexe A](#) :

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

Note: La dernière entrée prouve que le transfert d'images est dans le processus. Attendez quelques minutes, puis votez l'objet MIB de **tftpResult** de nouveau afin de vérifier qu'il a transféré avec succès. Cette étape peut prendre quelques minutes pour se terminer, qui dépend de la taille du fichier image (octets). Tandis que le processus de transfert d'images est en cours, si vous émettez une commande de **show flash** sur le commutateur, vous verrez :

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

2. Votez l'objet MIB de **tftpResult** et comparez la sortie aux détails d'objet MIB dans l'[annexe A](#) :

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

Dépannez le processus

Si le téléchargement est réussi, la sortie d'objet MIB est égale à 2 (ou à succès). Si vous recevez l'autre sortie, comparez-la à l'[annexe A](#) pour l'objet de **tftpResult** et prenez les mesures appropriées.

Une fois que le transfert d'images se termine avec succès, vérifiez que les correspondances de taille du fichier image (octets) qui affiché dans la sortie de commande de **show flash** au fichier dans le serveur TFTP (cat6000-sup.5-4-2a.bin, dans cet exemple).

Copiez une image du logiciel système du commutateur de Catalyst exécutant CatOS sur le serveur TFTP

Ces étapes vous guident par le processus pour copier une image logicielle.

Instructions pas à pas

Procédez comme suit :

1. Créez un nouveau fichier **image.bin** dans le répertoire de /tftpboot du serveur TFTP. Sur l'UNIX, utilisez cette syntaxe : **touch < nom du fichier >**. Utilisation **.bin**as l'extension de fichier.

```
touch image.bin
```

2. Changez les autorisations du fichier à **777** avec la syntaxe : **<permissions > <filename de chmod >**.

```
chmod 777 image.bin
```

3. Définissez l'adresse IP du serveur TFTP utilisant l'objet MIB de **tftpHost** :

```
chmod 777 image.bin
```

4. Définissez le nom du fichier TFTP que vous utiliserez pour copier le fichier d'image avec l'objet **tftpFile** MIB :

```
chmod 777 image.bin
```

5. Dans cet exemple, le numéro de module de superviseur est 2 et est dans l'emplacement le numéro 2 comme vu dans la sortie de commande de **show module**. Utilisation 2 afin de définir l'objet MIB de **tftpModule** :

```
chmod 777 image.bin
```


Ceci signifie que l'image de CatOS qui fonctionne sur le module de superviseur dans l'éclair est transférée vers le serveur TFTP comme vu dans la sortie de la commande de **show flash**.

6. Employez l'objet MIB de **tftpAction** afin de définir que le fichier d'image est transféré du serveur TFTP vers le commutateur avec la valeur d'objet MIB de **5 = uploadSw**. Voyez le MIB objecter des détails dans l'[annexe A](#) :

```
chmod 777 image.bin
```

Vérifiez le processus

Afin de vérifier les résultats de ces exécutions, exécutez une de ces étapes :

1. Votez le **.1.3.6.1.4.1.9.5.1.5**) objet MIB de **tftpGrp** (et comparez les résultats à l'[annexe A](#) :

```
chmod 777 image.bin
```

Note: La dernière entrée prouve que le transfert d'images est dans le processus. L'attente quelques minutes et votent alors l'objet MIB de **tftpResult** de nouveau afin de vérifier qu'elle a transféré avec succès. Cette étape peut prendre quelques minutes pour se terminer, qui dépend de la taille du fichier image (octets).

2. Votez l'objet MIB de **tftpResult** et comparez la sortie aux détails d'objet MIB dans l'[annexe A](#) :

```
chmod 777 image.bin
```

Dépannez le processus

Si le téléchargement est réussi, la sortie d'objet MIB est égale à 2 (ou à succès). Si vous recevez l'autre sortie, comparez-la à l'[annexe A](#) pour l'objet de **tftpResult** et prenez les mesures appropriées.

Une fois que le transfert d'images se termine avec succès, vérifiez les correspondances de taille du fichier image (octets) qui sont affichées dans la sortie de commande de **show flash** au fichier dans le serveur TFTP (**image.bin**, dans cet exemple).

Note: Si vous avez de plusieurs images dans l'éclair (**show flash**), seulement l'image dont le module de superviseur a été initialisée, est transféré vers le serveur TFTP avec cette procédure. Utilisez la commande de **show boot** de voir la variable de DÉMARRAGE =, qui affiche quelle image de l'éclair est utilisée par le module de superviseur pour initialiser. Référez-vous aux [mises à niveau d'images logicielles et à fonctionner avec des fichiers de configuration sur le](#) pour en savoir plus de [Commutateurs de Catalyst](#).

Exemple de script Unix

Note: Ces scripts sont fournis comme exemples seulement et ne sont pas pris en charge de quelque façon par Cisco Systems.

Script pour automatiser le transfert de fichier de configuration et de Cisco IOS sur des

Commutateurs

```
#!/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
```

Le commutateur s'attend à ce que le script exécute une commande particulière sur le commutateur

```

#!/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"
```

Script Perl pour afficher par l'intermédiaire du SNMP la même sortie en tant que « show cam dynamique »

```
#!/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
```

Annexe A — Détails d'objet MIB

Objet	tftpHost
OID	.1.3.6.1.4.1.9.5.1.5.1
Type	DisplayString
Autorisation	lecture/écriture
Syntaxe	OCTET STRING (0..64)
État	Courant
MIB	CISCO-STACK-MIB
Description	Nom de source/de destination host pour le transfert de transfert TFTP ou de périphérique de stockage. Si le nom est pour le transfert TFTP, ce peut être l'adresse IP ou le nom d'hôte. Si le nom pour le transfert de périphérique de stockage, il est dans le format du devicename : (par exemple slot0 : , slot1 :)
OID dans l'arborescence	:: = {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) cisco(9) workgroup(5) ciscoStackMIB(1) tftpGrp(5) 1}
Objet	tftpFile

OID	.1.3.6.1.4.1.9.5.1.5.2
Type	DisplayString
Autorisation	lecture/écriture
Syntaxe	OCTET STRING (0..64)
État	Courant
MIB	CISCO-STACK-MIB
Description	Nom de fichier pour le transfert TFTP ou pour le transfert de périphérique de stockage.
OID dans l'arborescence	:: = {iso(1) org(3) dod(6) internet(1) private(4) enterprises(1) cisco(9) workgroup(5) ciscoStackMIB(1) tftpGrp(5) 2}

Objet	tftpModule
OID	.1.3.6.1.4.1.9.5.1.5.3
Type	Entier
Autorisation	lecture/écriture
État	Courant
Plage	0 - 16
MIB	CISCO-STACK-MIB
Description	Quels code/configuration du module est transférée.
OID dans l'arborescence	:: = {ISO(1) org(3) DOD(6) Internet(1) private(4) enterprises(1) cisco(9) workgroup(5) ciscoStackMIB(1) tftpGrp(5) 3}

Objet	tftpAction
OID	.1.3.6.1.4.1.9.5.1.5.4
Type	Entier
Autorisation	lecture/écriture
État	Courant
Valeurs	<ol style="list-style-type: none"> 1. autre 2. downloadConfig 3. uploadConfig 4. downloadSw 5. uploadSw 6. downloadFw 7. uploadFw
MIB	CISCO-STACK-MIB
Description	Si vous placez cet objet à une des valeurs acceptables, elle initie l'action demandée avec les informations fournies dans le tftpHost, tftpFile, tftpModule. downloadConfig(2) : recevez la configuration à partir de l'hôte/du fichier uploadConfig(3) : envoyez héberger de configuration pour/fichier downloadSw(4) :

	recevez l'image logicielle à partir de l'hôte/du fichier uploadSw(5) : envoyez héberger d'image logicielle pour/fichier downloadFw(6) : recevez l'image de microprogramme à partir de l'hôte/du fichier uploadFw(7) : envoyez héberger d'image de microprogramme pour/fichier si vous placez cet objet à n'importe quelle autre valeur, vous obtiennent une erreur.
OID dans l'arborescence	:: = {ISO(1) org(3) DOD(6) Internet(1) private(4) entreprises(1) cisco(9) workgroup(5) ciscoStackMIB(1) tftpGrp(5) 4}
Objet	tftpResult
OID	.1.3.6.1.4.1.9.5.1.5.5
Type	Entier
Autorisation	en lecture seule
État	Courant
Valeurs	<ul style="list-style-type: none"> 1. inProgress 2. succès 3. noResponse 4. tooManyRetries 5. noBuffers 6. noProcesses 7. badChecksum 8. badLength 9. badFlash 10. serverError 11. userCanceled 12. wrongCode 13. fileNotFound 14. invalidTftpHost 15. invalidTftpModule 16. accessViolation 17. unknownStatus 18. invalidStorageDevice 19. insufficientSpaceOnStorageDevice 20. insufficientDramSize 21. incompatibleImage
MIB	CISCO-STACK-MIB
Description	Contient le résultat de la dernière demande d'intervention TFTP
OID dans l'arborescence	:: = {ISO(1) org(3) DOD(6) Internet(1) private(4) entreprises(1) cisco(9) workgroup(5) ciscoStackMIB(1) tftpGrp(5) 5}

Informations connexes

- [Déplacement de fichiers et d'images entre un routeur et un serveur TFTP via SNMP](#)
- [Téléchargement de MIB de Cisco](#)
- [Support et documentation techniques - Cisco Systems](#)