

Konfigurationsbeispiel für das Cisco IOS Shell-CLI-Modul

Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Hintergrundinformationen](#)

[Konfigurieren](#)

[Leitungen und Umleitung](#)

[Integrierte Applets](#)

[Bedingungen](#)

[Schleifen](#)

[String-Operationen](#)

[Konfigurationsbeispiel](#)

[Überprüfen](#)

[Fehlerbehebung](#)

[Zugehörige Informationen](#)

Einführung

Dieses Dokument beschreibt das Cisco IOS[®] Shell (IOS.sh) CLI-Modul.

Voraussetzungen

Anforderungen

Cisco empfiehlt, dass Sie mit Cisco IOS Release 15.1(4)M, 15.1(2)S und späteren Versionen vertraut sind. Cisco IOS.sh muss konfiguriert und aktiviert sein, damit die Funktionen und Funktionen von Cisco IOS.sh auf Ihrem Router verwendet werden können.

Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf den folgenden Hardware- und Softwareversionen:

- Cisco Catalyst Switches der Serie 6500
- Cisco IOS Version 15.1(1)SY

Die Informationen in diesem Dokument wurden von den Geräten in einer bestimmten Laborumgebung erstellt. Alle in diesem Dokument verwendeten Geräte haben mit einer leeren (Standard-)Konfiguration begonnen. Wenn Ihr Netzwerk in Betrieb ist, stellen Sie sicher, dass Sie die potenziellen Auswirkungen eines Befehls verstehen.

Hintergrundinformationen

Cisco IOS.sh ist ein Modul, mit dem Sie Shell-Scripting in der Cisco IOS-CLI verwenden können. Dieses Modul unterstützt die Automatisierung von Routineaufgaben und vereinfacht das Gerätemanagement. Sie kann in Verbindung mit anderen Verwaltungstools wie Cisco IOS Embedded Event Manager (EEM) verwendet werden.

Konfigurieren

Hinweis: Verwenden Sie das [Command Lookup Tool](#) (nur [registrierte](#) Kunden), um weitere Informationen zu den in diesem Abschnitt verwendeten Befehlen zu erhalten.

Um die Shell-CLI auf dem Gerät zu aktivieren, geben Sie Folgendes ein:

```
Switch#terminal shell
```

Um die Shell-CLI zu deaktivieren, geben Sie Folgendes ein:

```
Switch#terminal no shell
```

Dieses Feature unterstützt die PATH-Variable, die eine Liste von Verzeichnissen enthält, in denen die Shell nach Funktionen/Applets suchen kann. Die Shell-CLI unterstützt außerdem folgende Funktionen:

Leitungen und Umleitung

```
Switch#show version | grep 15.1
```

```
Cisco IOS Software, s2t54 Software (s2t54-ADVENTERPRISEK9-M),  
Version 15.1(1)SY, RELEASE SOFTWARE (fc2)
```

```
Switch#show version | grep 15 > bootdisk:version.txt
```

```
Switch#dir bootdisk:
```

```
Directory of bootdisk:/
```

```
14 -rw- 280 Nov 15 2012 23:25:32 +00:00 version.txt
```

```
1024557056 bytes total (577273856 bytes free)
```

```
Switch#cat bootdisk:version.txt
```

```
Cisco IOS Software, s2t54 Software (s2t54-ADVENTERPRISEK9-M),  
Version 15.1(1)SY, RELEASE SOFTWARE (fc2)
```

Integrierte Applets

Cisco IOS.sh enthält mehrere integrierte Applets, die bei der Erstellung von Anwendungen und Filtern für Benutzerfunktionen helfen:

[[evaluate a logical test expression
cat	output data from a pipe or file to the terminal
cut	edit piped output
echo	echo arguments to the terminal
false	return false in while or if expressions, and set the result
fetch	return values from the configuration database
grep	search for regular expressions in piped output or files
head	print the first lines in the input
interface	print interfaces that match the argument
let	evaluate a numeric expression, and set the result
man	print information for built-ins
more	page piped output to the terminal
nl	number the lines in the input
null	ignore the input
printf	output formatted data to the terminal
read	read input into variables
set_oper	set operational values
sleep	pause execution of the terminal
sort	sort the input
tail	print the tail of the input
true	return true in while or if expressions, and set the result
uname	print system information
wc	count lines, words, and characters

Bedingungen

Hier sind die Befehle für die **if....dann...else** Operator.

```
Switch#x=5
Switch#if [[ $x != 5 ]];
then.else.fi>then
then.else.fi>echo false;
then.else.fi>else
else..fi>echo true;
else..fi>fi;
true
Switch#
```

Diese Vergleichsoperatoren werden für die Arbeit mit ganzzahligen Werten verwendet:

Operatoren Beschreibung

-äq	Arg1-eq Arg2. True, wenn Arg1 gleich Arg2 ist
-n	Arg1-ne Arg2. True, wenn Arg1 nicht gleich Arg2 ist
ln	Arg1-lt Arg2. True, wenn Arg1 kleiner als Arg2 ist
0	Arg1-gt Arg2. True, wenn Arg1 größer als Arg2 ist
ge	Arg1-ge Arg2. True, wenn Arg1 größer oder gleich Arg2 ist
-le	Arg1-le Arg2. True, wenn Arg1 kleiner oder gleich Arg2 ist

Diese Bedingungen sind für die Arbeit mit Dateien verfügbar:

Operatoren Beschreibung

-a oder -e	True, wenn die Datei vorhanden ist
------------	------------------------------------

-d	True, wenn eine Datei vorhanden ist, und wenn es sich um ein Verzeichnis handelt
f	True, wenn eine Datei vorhanden ist und eine reguläre Datei ist
r	True, wenn eine Datei vorhanden ist und lesbar ist
E	True, wenn eine Datei vorhanden ist und eine Größe von mehr als 0 hat
-w	True, wenn die Datei vorhanden ist und ausführbar ist
-nicht	Testen Sie, ob file1 neuer als file2 ist. Für diesen Vergleich wird das Änderungsdatum in der verwendet.
-To	Testen Sie, ob Datei1 älter als Datei2 ist.

Um mehrere Bedingungen in einem Vergleichsoperator zu verwenden, können Sie die folgenden logischen Operatoren verwenden:

```
&&    Logical AND
||    Logical OR
```

Schleifen

Schleifen sind die gleichen wie in der Linux-Shell. Es stehen zwei Operatoren zur Verfügung.

Nachfolgend sind die Befehle für den **for**-Operator aufgeführt:

```
Switch#for x in 11
do..done>do
do..done>ping 192.0.2.$x
do..done>done
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.0.2.10, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
Switch#
```

Nachfolgend sind die Befehle für den **while**-Operator aufgeführt:

```
Switch#for i in 1 2 3
do..done>do
do..done>while [ $i -lt 3 ]
do..done>do
do..done>echo $i
do..done>done
do..done>i=i+1
do..done>done
Switch#
```

String-Operationen

Bei Zeichenfolgenvergleichen wird das **if...dann...else**-Operator kann mit den folgenden Änderungen verwendet werden:

Operatoren Beschreibung

==	String 1 == string2, True, wenn die Zeichenfolgen gleich sind.
!=	String1 != string2, True, wenn die Zeichenfolgen nicht gleich sind
<	String1 < string2, True, wenn string1 lexikografisch kleiner als string2 ist
>	String1 > string2, True, wenn string1 lexikografisch größer als string2 ist

Konfigurationsbeispiel

```
Switch#function add_desc() {  
{..} >show running-config interface Gi$1  
{..} >configuration terminal  
{..} >interface Gi$1  
{..} >description $2  
{..} >end  
{..} >show running-config interface Gi$1  
{..} >}  
Switch#add_desc 5/1 SHELL_TEST  
Building configuration...
```

```
Current configuration : 74 bytes  
!  
interface GigabitEthernet5/1  
ip address 192.0.2.10 255.255.255.0  
end
```

Enter configuration commands, one per line. End with CNTL/Z.
Building configuration...

```
Current configuration : 98 bytes  
!  
interface GigabitEthernet5/1  
description SHELL_TEST  
ip address 192.0.2.10 255.255.255.0  
end
```

Switch#

Überprüfen

In diesem Abschnitt überprüfen Sie, ob Ihre Konfiguration ordnungsgemäß funktioniert.

Geben Sie den Befehl **show terminal** (Terminal anzeigen) ein, um die aktuelle Terminalkonfiguration anzuzeigen:

```
Switch#show terminal  
Line 2, Location: "", Type: "XTERM-COLOR"  
Length: 44 lines, Width: 155 columns  
Baud rate (TX/RX) is 9600/9600  
Status: PSI Enabled, Ready, Active, No Exit Banner, Ctrl-c Enabled  
.....  
Preferred transport is lat.
```

```
Shell: enabled  
Shell trace: off
```

Geben Sie den Befehl **show shell functions** (**show shell-Funktionen**) ein, um alle im Modul verfügbaren Funktionen anzuzeigen, einschließlich benutzerdefinierter Funktionen:

```
Switch#show shell functions  
#User defined functions:  
  
Function namespace: DEFAULT  
function add_desc()
```

```
{
show running-config interface Gi$1
configure terminal
interface Gi$1
description $2
end
show running-config interface Gi$1
}
```

Fehlerbehebung

Für diese Konfiguration sind derzeit keine spezifischen Informationen zur Fehlerbehebung verfügbar.

Zugehörige Informationen

- [Catalyst 6500 Konfigurationsanleitungen](#)
- [Technischer Support und Dokumentation - Cisco Systems](#)