

Exemplo de configuração do módulo do shell CLI do Cisco IOS

Índice

[Introdução](#)

[Pré-requisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Informações de Apoio](#)

[Configurar](#)

[Tubulações e reorientação](#)

[Applet encaixados](#)

[Condições](#)

[Circuitos](#)

[Operações de série](#)

[Exemplo de configuração](#)

[Verificar](#)

[Troubleshooting](#)

[Informações Relacionadas](#)

Introdução

Este documento descreve o módulo do shell do [®] do Cisco IOS (IOS.sh) CLI.

Pré-requisitos

Requisitos

Cisco recomenda que você tem o conhecimento do Cisco IOS Release 15.1(4)M, do 15.1(2)S, e de umas liberações mais atrasadas. Cisco IOS.sh deve ser configurado e permitido a fim usar as características e as funções de Cisco IOS.sh em seu roteador.

[Componentes Utilizados](#)

As informações neste documento são baseadas nas seguintes versões de hardware e software:

- Cisco Catalyst 6500 Series Switch

- Cisco IOS Release 15.1(1)SY

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se a sua rede estiver ativa, certifique-se de que entende o impacto potencial de qualquer comando.

Informações de Apoio

Cisco IOS.sh é um módulo que permita que você use o script de shell no IOS Cisco CLI. Este módulo ajuda a automatizar ações rotineiras e simplifica o Gerenciamento de dispositivos. Pode ser usado conjuntamente com outras ferramentas de gerenciamento tais como o gerente encaixado Cisco IOS do evento (EEM).

Configurar

Nota: Use a [Command Lookup Tool](#) ([somente clientes registrados](#)) para obter mais informações sobre os comandos usados nesta seção.

A fim permitir o shell CLI no dispositivo, entre:

```
Switch#terminal shell
```

A fim desabilitar o shell CLI, entre:

```
Switch#terminal no shell
```

Este suportes de recurso o variável de caminho, que contém uma lista de diretórios onde o shell pode procurar pela função/applet. Também, o shell CLI apoia estas características:

Tubulações e reorientação

```
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)
```

Applet encaixados

Cisco IOS.sh contém diversos applet incorporados que ajudam a criar pedidos e filtros para as funções do usuário:

```
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)
```

Condições

Estão aqui os comandos para **se....então....operador outro**.

```
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#
```

Estes operadores de comparação são usados a fim trabalhar com valores de número inteiros:

Operadores Descrição

- eq Arg1-eq Arg2. Retifique se Arg1 é igual a Arg2
- ne Arg1-ne Arg2. Retifique se Arg1 não é igual a Arg2
- lt Arg1-lt Arg2. Retifique se Arg1 é menos do que Arg2
- GT Arg1-gt Arg2. Retifique se Arg1 é maior do que Arg2
- ge Arg1-ge Arg2. Retifique se Arg1 é greate do que ou igual a Arg2
- le Arg1-le Arg2. Retifique se Arg1 é inferior ou igual a Arg2

Estas circunstâncias estão disponíveis a fim trabalhar com arquivos:

Operadores Descrição

- a ou - e Retifique se o arquivo existe
- d Retifique se o arquivo existe e é um diretório
- f Retifique se o arquivo existe e é um arquivo regular
- r Retifique se o arquivo existe e é legível
- s Retifique se o arquivo existe e tem um tamanho maior de zero
- w Retifique se o arquivo existe e é executável
- NT Teste se file1 é mais novo do que file2. A data da alteração no arquivo é usada para esta comparação
- ot Teste se file1 é mais velho do que file2

A fim usar diversas condições dentro de um operador de comparação, você pode usar estes operadores lógicos:

```
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#
```

Circuitos

Os laços são os mesmos que no shell de Linux. Dois operadores estão disponíveis.

Estão aqui os comandos para **para o** operador:

```
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#
```

Estiverem aqui os comandos para o operador do **quando**:

```
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#
```

Operações de série

Para comparações de série, **se....então....**o operador **outro** pode ser usado com estas alterações:

Operadores Descrição

==	O == string2 da corda 1, retifica se as cordas são iguais.
!=	String1 != string2, retificam se as cordas não são iguais
<	String1 < string2, retificam se string1 é lexicographically menos do que string2
>	String1 > string2, retificam se string1 é lexicographically maior do que string2

Exemplo de configuração

```
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#
```

Verificar

Use esta seção para confirmar se a sua configuração funciona corretamente.

Inscreva o **comando show terminal** a fim indicar a configuração do terminal atual:

```
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
```

Incorpore as **funções do shell da mostra** comandam a fim indicar todas as funções disponíveis no módulo, que incluem funções definidas pelo utilizador:

```
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
```

```
}
```

Troubleshooting

Atualmente, não existem informações disponíveis específicas sobre Troubleshooting para esta configuração.

Informações Relacionadas

- [Manuais de configuração do Catalyst 6500](#)
- [Suporte Técnico e Documentação - Cisco Systems](#)