

Automatização das sequências de comandos do Cisco IOS XE usando TCL e EEM

Introdução

Este documento descreve um método para criar um único script EEM capaz de executar qualquer sequência de comandos conforme definido por um administrador de rede.

Pré-requisitos

Requisitos

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Conhecimento das operações do Cisco IOS® XE
- Compreensão de scripts do Embedded Event Manager (EEM)
- Familiaridade com os switches Cisco Catalyst
- Conhecimento básico de script TCL

Componentes Utilizados

As informações neste documento são baseadas no Cisco Catalyst 9600X Series Switch (CS-C9600X) executando IOS XE versão 17.18.2.

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 rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

Informações de Apoio

O administrador cria um arquivo de texto simples que contém a sequência de comandos desejada. O script EEM chama esse arquivo em um disparador especificado e executa todos os comandos listados no arquivo de texto.

Configuração e procedimento passo a passo

Etapa 1. Criação de Arquivo de Texto com Sequência de Comandos

1. Crie um arquivo de texto contendo os comandos necessários na sequência desejada.
2. Copie o arquivo para a memória flash do dispositivo.
3. Use scripts FTP, TFTP ou TCL para criar ou transferir o arquivo.

Criando um arquivo de texto usando script TCL

Sintaxe:

```
<#root>
```

```
tclsh  
puts [open "flash:COMMANDS_TO_EXECUTE.txt" w+] {
```

```
Enter Commands to Execute
```

```
}  
exit
```

Exemplo de configuração:

Para criar uma sequência de comandos que feche a interface FiftyGigE1/1/0/3:

```
C9600-Silicon1-SVL#tclsh  
C9600-Silicon1-SVL(tcl)#puts [open "flash:COMMANDS_TO_EXECUTE.txt" w+] {  
configure terminal  
interface fiftyGigE 1/1/0/3  
shutdown  
end  
show run interface fiftyGigE 1/1/0/3  
}  
C9600-Silicon1-SVL(tcl)#exit  
C9600-Silicon1-SVL#
```

Verifique o arquivo na flash:

```
C9600-Silicon1-SVL#dir flash:
Directory of flash:/
```

```
357117 -rw- 101 Jan 25 2026 21:37:09 +05:30 COMMANDS_TO_EXECUTE.txt
```

Conteúdo de COMMANDS_TO_EXECUTE.txt

```
configure terminal
interface fiftyGigE 1/1/0/3
shutdown
end
show run interface fiftyGigE 1/1/0/3
```

Esse arquivo pode ser criado por script TCL ou transferido usando FTP, TFTP ou RCP. O arquivo deve ser editado conforme necessário para incluir qualquer sequência de comandos exigida pelo administrador da rede.

Etapa 2. Executar o Arquivo Usando o Script EEM

Criação de script EEM

Este miniaplicativo EEM executa todos os comandos do arquivo especificado COMMANDS_TO_EXECUTE.

```
event manager applet Cisco_Live authorization bypass
event none maxrun 1000
action 010 cli command "enable"
action 020 cli command "term exec prompt timestamp"
action 030 cli command "term exec prompt expand"
action 040 file open fileout1 flash:OUTPUT.txt a+
action 050 file open filein1 flash:COMMANDS_TO_EXECUTE.txt r
action 060 file read filein1 input
action 070 foreach line $input "\n"
action 080 cli command "$line"
action 090 file write fileout1 "$_cli_result"
action 100 end
action 110 file close fileout1
action 120 file close filein1
```

Explicação do script EEM

++ miniaplicativo do gerenciador de eventos desvio de autorização do Cisco_Live:

Nome do miniaplicativo do gerenciador de eventos: Registra um miniaplicativo EEM com o nome especificado ('Cisco_Live' neste caso).

Desvio de autorização: Palavras-chave opcionais que especificam o applet devem ignorar a autorização AAA.

++ evento none maxrun 1000:

Evento nenhum: Isso especifica que a política EEM é registrada com o detector de eventos 'none', o que significa que a política não é acionada por nenhum evento automático, mas deve ser executada manualmente.

Maxrun 1000: Esse parâmetro opcional define o tempo de execução máximo do applet ou da política como 1000 segundos. Se não for especificado, o tempo de execução máximo padrão será de 20 segundos.

ação ++ 020 cli termo carimbo de data/hora do prompt exec:

Ação 020: Este é o identificador do rótulo ou da sequência para a ação no script EEM.

term exec prompt timestamp é um comando CLI que habilita timestamps no prompt na sessão de terminal atual.

Esse recurso ajuda na solução de problemas e na auditoria, fornecendo uma referência de tempo para cada comando executado.

O timestamp aparece antes da saída dos comandos, facilitando a correlação de eventos e execuções de comandos com horários específicos.

++ ação 030 cli termo exec prompt expandir:

term exec prompt expand ativa o modo de prompt expandido para a sessão de terminal atual.

E configura o terminal para exibir o prompt completo sem truncamento ou abreviação.

Ele ajuda em scripts ou cenários de solução de problemas em que o contexto de prompt completo é necessário para esclarecimento ou análise.

O comando afeta somente a sessão de terminal atual e não persiste após o término da sessão.

++ ação 040 arquivo abrir arquivo1 flash:OUTPUT.txt a+:

Arquivo aberto: Especifica a operação para abrir um arquivo.

Fileout1: Este é o descritor de arquivo ou nome do identificador usado no script para fazer referência ao arquivo aberto.

Flash:OUTPUT.txt: Este é o caminho e o nome do arquivo no sistema de arquivos flash onde o arquivo reside ou será criado.

a+: Este é o modo de acesso a arquivos, o que significa que o arquivo está aberto para leitura e acréscimo. Se o arquivo não existir, ele será criado. Os dados gravados no arquivo serão anexados ao final sem truncar o conteúdo existente.

++ ação 050 arquivo abrir arquivoem1 flash:COMMANDS_TO_EXECUTE.txt r:

Arquivo1: Este é o descritor de arquivo ou nome do identificador usado no script para fazer referência ao arquivo aberto.

Flash:COMMANDS_TO_EXECUTE.txt: Este é o caminho e o nome do arquivo no sistema de arquivos flash onde o arquivo reside.

r: Este é o modo de acesso a arquivos, o que significa que o arquivo está aberto somente para leitura.

++ ação 060 arquivo de leitura em1 entrada:

Leitura do arquivo: Especifica a operação a ser lida de um arquivo.

Arquivo1: Este é o descritor ou identificador de arquivo que foi atribuído quando o arquivo foi aberto (por exemplo, com ação 050 file open file in1. Flash:COMMANDS_TO_EXECUTE.txt r).

Entrada: Este é o nome da variável onde os dados lidos serão armazenados para uso em ações de script subsequentes.

++ ação 070 para cada linha \$input '\n':

Para cada: Este comando especifica uma iteração sobre uma string.

Linha: Esta é a variável de iterador que manterá cada token (linha) durante cada iteração.

\$input: Esta é a variável de string de entrada que contém várias linhas ou entradas.

'\n': Este é o delimitador usado para dividir a string de entrada em tokens, neste caso, o caractere de nova linha.

Cada linha é um token e \n é o divisor nesse caso.

++ ação 080 comando cli \$line:

Comando Cli: Especifica que a ação é executar um comando CLI no dispositivo.

\$linha: A variável que contém a sequência de comando CLI a ser executada. As aspas asseguram que a string inteira seja tratada como um único comando.

++ ação 090 arquivo gravar arquivo1 '\$_cli_result':

Gravação de arquivo: O comando EEM para gravar dados em um arquivo.

Fileout1: O manipulador de arquivos aberto anteriormente por um comando como file open fileout1 flash:OUTPUT.txt a+, que representa o arquivo no qual os dados são gravados.

'\$_cli_result': A variável que contém a saída do comando CLI ou qualquer dado de sequência de caracteres a ser gravado no arquivo. As aspas asseguram que todo o conteúdo seja tratado como uma única sequência de caracteres.

Esse comando é normalmente usado após executar comandos CLI e capturar sua saída em \$_cli_result, permitindo que o script registre ou archive a saída em um arquivo no armazenamento flash do dispositivo.

++ ação 100 terminar:

Fim: A palavra-chave que significa o fim de um bloco condicional, como if, else ou while.

++ ação 120 arquivo fechar arquivo em1:

Fechamento de arquivo: A operação para fechar o descritor de arquivo.

Arquivo1: O identificador do descritor de arquivo a ser fechado, que deve ter sido aberto anteriormente no script.

Execução de script EEM

Para disparar manualmente o script EEM:

```
C9600-Silicon1-SVL#event manager run Cisco_Live
```

Neste exemplo, o script EEM é acionado manualmente. O script pode ter qualquer disparador de acordo com os requisitos de rede.

Resultado da execução do arquivo

A exibição do arquivo de saída mostra os comandos executados e seus resultados:

```
<#root>
```

```
C9600-Silicon1-SVL#more flash:OUTPUT.txt
```

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

```
C9600-Silicon1-SVL(config)#
```

```
C9600-Silicon1-SVL(config-if)#
```

```
C9600-Silicon1-SVL(config-if)#
```

```
C9600-Silicon1-SVL#Load for five secs: 0%/0%; one minute: 1%; five minutes: 1%
```

```
No time source, *22:01:35.406 IST Sun Jan 25 2026
```

```
----- show running-config interface FiftyGigE1/1/0/3 -----
```

```
Building configuration...
```

```
Current configuration : 127 bytes
```

```
!
```

```
interface FiftyGigE1/1/0/3
```

```
description To-Host3-Ten 1/1/3
```

```
switchport access vlan 10
```

```
switchport mode access
```

```
shutdown
```

end

A interface FiftyGigE1/1/0/3 foi desativada com êxito de acordo com a sequência de comandos.

Summary

Este documento demonstrou como combinar o script TCL e o EEM da Cisco para criar um script EEM único e flexível capaz de executar qualquer sequência de comandos armazenados em um arquivo de texto. A abordagem permite que os administradores de rede definam sequências de comandos externamente e acionem sua execução automática ou manualmente, melhorando a automação e a eficiência operacional em switches Cisco Catalyst que executam o Cisco IOS XE.

Sobre esta tradução

A Cisco traduziu este documento com a ajuda de tecnologias de tradução automática e humana para oferecer conteúdo de suporte aos seus usuários no seu próprio idioma, independentemente da localização.

Observe que mesmo a melhor tradução automática não será tão precisa quanto as realizadas por um tradutor profissional.

A Cisco Systems, Inc. não se responsabiliza pela precisão destas traduções e recomenda que o documento original em inglês ([link fornecido](#)) seja sempre consultado.