TCLおよびEEMを使用したCisco IOS XEコマンドシーケンスの自動化

内容

はじめに

前提条件

- 要 件
- 使用するコンポーネント

バックグラウンド情報

設定および手順

- ステップ 1：コマンドシーケンスを使用したテキストファイルの作成
  - TCLスクリプトを使用したテキストファイルの作成
- ステップ 2EEMスクリプトを使用したファイルの実行
  - EEMスクリプトの作成
  - EEMスクリプトの説明
  - EEMスクリプトの実行
  - ファイル実行の結果

要約

はじめに

このドキュメントでは、ネットワーク管理者が定義した任意のコマンドシーケンスを実行できる、一意で単一のEEMスクリプトを作成する方法について説明します。

前提条件

要 件

次の項目に関する知識があることが推奨されます。

- Cisco IOS® XEの動作に関する知識
- Embedded Event Manager(EEM)スクリプトの理解
- Cisco Catalystスイッチの知識
- TCLスクリプトの基礎知識

使用するコンポーネント

このドキュメントの情報は、IOS XE 17.18.2リリースを実行するCisco Catalyst 9600Xシリーズスイッチ(CS-C9600X)に基づくものです。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。本稼働中のネットワークでは、各コマンドによって起こる可能性がある影響を十分確認してください。

バックグラウンド情報

管理者は、目的のコマンドシーケンスを含むプレーンテキストファイルを作成します。次に、EEMスクリプトは指定されたトリガーでこのファイルを呼び出し、テキストファイルに記載されているすべてのコマンドを実行します。

設定と詳細手順

ステップ 1：コマンドシーケンスを使用したテキストファイルの作成

1. 必要なコマンドを含むテキストファイルを目的の順序で作成します。
2. ファイルをデバイスのフラッシュメモリにコピーします。
3. FTP、TFTP、またはTCLスクリプトを使用して、ファイルを作成または転送します。

TCLスクリプトを使用したテキストファイルの作成

構文：

<#root>

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


Enter Commands to Execute



}
exit
```

設定例：

インターフェイスFiftyGigE1/1/0/3をシャットダウンする一連のコマンドを作成するには、次のコマンドを実行します。

```
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#
```

フラッシュ内のファイルを確認します。

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

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

COMMANDS_TO_EXECUTE.txtの内容

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

このファイルは、TCLスクリプトで作成するか、FTP、TFTP、またはRCPを使用して転送できます。ネットワーク管理者が必要とするコマンドシーケンスを含めるには、必要に応じてファイルを編集する必要があります。

ステップ 2EEMスクリプトを使用したファイルの実行

EEMスクリプトの作成

このEEMアプレットは、指定されたファイルCOMMANDS_TO_EXECUTEからすべてのコマンドを実行します。

```
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
```

EEMスクリプトの説明

++ event manager applet Cisco_Live authorization bypass:
Event manager applet applet-name：指定した名前（この場合は「Cisco_Live」）でEEMアプレットを登録します。
許可バイパス：アプレットを指定するオプションキーワードは、AAA許可をバイパスする必要があります。

++ event none maxrun 1000:
Event none：これは、EEMポリシーが「none」イベントディテクタに登録されることを指定します。つまり、ポリシーは自動イベントによってトリガーされず、手動で実行されます。
Maxrun 1000：このオプションパラメータは、アプレットまたはポリシーの最大ランタイムを1000秒に設定します。指定しない場合、デフォルトの最大ランタイムは20秒です。

++ action 020 cliコマンドterm exec prompt timestamp:
Action 020：これは、EEMスクリプト内のアクションのラベルまたはシーケンス識別子です。
term exec prompt timestampは、現在のターミナルセッションのプロンプトでタイムスタンプを有効にするCLIコマンドです。
この機能は、実行される各コマンドの時間基準を提供することで、トラブルシューティングと監査に役立ちます。
タイムスタンプはコマンドの出力の前に表示されるため、イベントとコマンド実行を特定の時刻に関連付けることがより簡単になります。

++ action 030 cliコマンドterm exec prompt expand:
term exec prompt expandは、現在のターミナルセッションの拡張プロンプトモードを有効にします。
また、端末が切り詰めや省略形なしで完全なプロンプトを表示するように設定します。
スクリプトの記述を明確にしたり、解析を行うために完全なプロンプトコンテキストが必要な場

合に、スクリプトやトラブルシューティングを行う際に役立ちます。
このコマンドは現在のターミナルセッションにのみ影響し、セッション終了後は保持されません。

++ action 040 file open fileout1 flash:OUTPUT.txt a+:
ファイルを開く：ファイルを開く操作を指定します。
Fileout1：これは、開かれたファイルを参照するためにスクリプト内で使用されるファイル記述子またはハンドル名です。
Flash:OUTPUT.txt：これは、ファイルが存在する、または作成されるフラッシュファイルシステム内のパスおよびファイル名です。
a+：これはファイルアクセスモードであり、ファイルが読み取りおよび追加用に開かれることを意味します。ファイルが存在しない場合は、ファイルが作成されます。ファイルに書き込まれたデータは、既存のコンテンツを切り捨てずに末尾に追加されます。

++ action 050 file open filein1 flash:COMMANDS_TO_EXECUTE.txt r:
Filein1：これは、開かれたファイルを参照するためにスクリプト内で使用されるファイル記述子またはハンドル名です。
Flash:COMMANDS_TO_EXECUTE.txt：これは、ファイルが存在するフラッシュファイルシステム内のパスとファイル名です。
r：これはファイルアクセスモードです。つまり、ファイルは読み取り専用で開かれます。

++ action 060 file read filein1 input:
ファイル読み込み：ファイルから読み込む操作を指定します。
Filein1：これは、ファイルを開いたときに割り当てられたファイル記述子またはハンドルです(たとえば、action 050 file open filein1。Flash:COMMANDS_TO_EXECUTE.txt r)
入力：これは、後続のスクリプトアクションで使用するために読み取りデータが格納される変数名です。

++ action 070 foreach line $input '\n':
Foreach：このコマンドは、文字列の反復を指定します。
Line：これは、反復処理のたびに各トークン（行）を保持するイテレータ変数です。
$input：これは、複数の行またはエントリを含む入力文字列変数です。
'\n'：これは、入力文字列をトークンに分割するために使用されるデリミタです。この場合は改行文字です。
この場合、各行はトークンで、\nはスプリッタです。

++ action 080 cliコマンド$line:
Cliコマンド：デバイスでCLIコマンドを実行することを指定します。

$line：実行するCLIコマンド文字列を含む変数。引用符によって、文字列全体が1つのコマンドとして扱われます。

++ action 090 file write fileout1 '$_cli_result':
ファイル書き込み：データをファイルに書き込むEEMコマンド。
Fileout1：以前にfile open fileout1 flash:OUTPUT.txt a+などのコマンドで開かれたファイルハンドラ。データの書き込み先のファイルを表します。
'$_cli_result': CLIコマンド出力を保持する変数、またはファイルに書き込まれる文字列データ。引用符によって、コンテンツ全体が1つの文字列として扱われます。
通常、このコマンドは、CLIコマンドを実行し、その出力を$_cli_resultにキャプチャした後に使用します。これにより、スクリプトは出力をデバイスのフラッシュストレージ上のファイルに記録またはアーカイブできます。

++ action 100 end:
End: if、else、whileなどの条件付きブロックの終了を示すキーワード。

++アクション120 file close filein1:
File close：ファイル記述子を閉じる操作。
Filein1：閉じるファイル記述子の識別子。スクリプト内で以前に開かれている必要があります。

EEMスクリプトの実行

EEMスクリプトを手動でトリガーするには、次の手順を実行します。

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

この例では、EEMスクリプトは手動でトリガーされます。スクリプトには、ネットワーク要件に従って任意のトリガーを指定できます。

ファイル実行の結果

出力ファイルを表示すると、実行されたコマンドとその結果が表示されます。

<#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
```

インターフェイスFiftyGigE1/1/0/3は、コマンドシーケンスに従って正常にシャットダウンされます。

要約

このドキュメントでは、テキストファイルに保存された任意のコマンドシーケンスを実行できる柔軟な単一のEEMスクリプトを作成するために、TCLスクリプトとシスコのEEMを組み合わせる方法について説明しました。このアプローチにより、ネットワーク管理者はコマンドシーケンスを外部で定義し、その実行を自動または手動でトリガーできるため、Cisco IOS XEを実行するCisco Catalystスイッチの自動化と運用効率が向上します。

翻訳について
シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。