

在CCM中配置SIP規範化指令碼

目錄

[簡介](#)

[必要條件](#)

[需求](#)

[採用元件](#)

[設定](#)

[在CUCM的中繼上應用指令碼](#)

[在SIP電話上應用指令碼](#)

[制定常見方案的指令碼](#)

[修改標題](#)

[刪除標題](#)

[新增標題](#)

[操作SDP內容](#)

[驗證](#)

[疑難排解](#)

[相關資訊](#)

簡介

本檔案介紹如何在思科呼叫管理器(CCM)中使用會話初始協定(SIP)規範化指令碼，僅提供幾個基本示例。

必要條件

需求

思科建議您瞭解以下主題：

- Cisco Unified Communications Manager(CUCM)版本8.x及更高版本
- SIP通訊協定
- 指令碼知識

採用元件

本檔案中的資訊是根據CCM 11.5版。

本文中的資訊是根據特定實驗室環境內的裝置所建立。文中使用到的所有裝置皆從已清除（預設）的組態來啟動。如果您的網路運作中，請確保您瞭解任何指令可能造成的影響。

設定

在CUCM的中繼上應用指令碼

對於SIP中繼，指令碼需要直接應用於SIP中繼。在SIP配置檔案上而不是中繼上使用指令碼將不起作用。

以下是您需要遵循的步驟：

1. 導覽至Device > Device Setting > SIP Normalization Script Configuration，然後新增指令碼，如下圖所示。

SIP Normalization Script Configuration Related Links

Save Import File

Status: Ready

SIP Normalization Script Info

Name* tes_script

Description

Content*

```
M = {}
function M.outbound_INVITE(msg)
  -- Replacing the Anonymous from From Header
  local from = msg:getHeader("From")
  local newfrom = string.gsub(from, "Anonymous", "1111")
  msg:modifyHeader("From", newfrom)
end
return M
```

Script Execution Error Recovery Action* Message Rollback Only

System Resource Error Recovery Action* Disable Script

Memory Threshold* 50 kilobytes

Lua Instruction Threshold* 1000 instructions

Save Import File

2. 導航至Device > Trunk。

選擇要應用指令碼的Trunk（中繼），並在該指令碼的配置頁面上選擇之前建立的指令碼。選擇 **Enable Trace** 選項，它將在訊號分佈層(SDL)日誌中列印指令碼所做的更改，如下圖所示。儲存配置、應用配置，然後重置中繼，以使更改生效。

Normalization Script

Normalization Script test_script

Enable Trace

	Parameter Name	Parameter Value
1		

Recording Information

在SIP電話上應用指令碼

對於SIP電話，必須使用電話SIP配置檔案上的指令碼。

以下是您需要遵循的步驟。

1. 導航到 **Device > Device Setting > SIP Normalization Script**，然後新增 **New Script**。
2. 導航到 **Device > Device Setting > SIP profile**。選擇要應用指令碼的SIP配置檔案，或通過複製標準SIP配置檔案來建立新指令碼。
3. 在SIP Profile的Configuration頁面上，依次選擇Script和Apply Config並重置配置檔案。
4. 導覽至 **Device > Phone**。選擇要應用指令碼的電話並使用建立的指令碼更改其SIP配置檔案，然後儲存它，應用配置並重置電話以便進行更改。

制定常見方案的指令碼

有關如何開發SIP規範化指令碼的詳細步驟的主要來源如下：[SIP透明度和規範化開發人員指南](#)。

本指南包含可在SIP消息和會話描述協定(SDP)內容及其他高級API中執行操作的不同功能。

下面是幾個基本的示例指令碼：

修改標題

您可以在此處替換傳出SIP INVITE/REINVITE消息的**From**報頭。

```
M = {}
```

```
function M.outbound_INVITE(msg)

    -- Replacing the Anonymous from From Header

    local from = msg.getHeader("From")

    local newfrom = string.gsub(from, "anonymous" , "1111")

    msg.modifyHeader("From", newfrom)

end
```

```
return M
```

代碼

M={}

函式M.outbound_INVITE(msg)

local from =

msg.getHeader("From")

getHeader

string.gsub

修改標題

說明

消息內容的初始化。M在此處獲取SIP消息的所有內容

應用於M (SIP消息) 出站方向INVITE/REINVITE消息的Lua函式。方向始終

有關詳細資訊，請參閱[SIP規範化的開發人員指南的概述部分](#)。

以區域性變數形式存儲From消息的內容

CCM中可用於規範化的函式之一，用於獲取變數中標頭的內容

Lua函式，用於從字串中替換特定內容

同樣，在CCM中有一個可用函式，用於修改報頭內容

刪除標題

此處，從傳入SIP INVITE/REINVITE消息中刪除Cisco-Guide報頭：

```
M = {}
```

```
function M.inbound_INVITE(msg)

    msg:removeHeader("Cisco-Guid")

end

return M
```

新增標題

您可以在此處將INFO新增到Allow標題的內容中。

該內容將新增到逗號後允許的原始內容中。

```
M = {}

function M.outbound_INVITE(msg)

    msg:addHeader("Allow", "INFO")

end

return M
```

操作SDP內容

有可用的API用於修改SIP消息中的SDP內容。為了修改SDP，它必須使用SIP消息對象提供的 **getSdp()** API從Lua SIP消息對象獲取SDP內容主體。然後，指令碼可以使用包含思科API的字串庫來操作SDP。修改時，SDP將使用SIP消息對象提供的 **setSdp(sdp)** API回寫到SIP消息對象。有關這些API的詳細資訊，請參閱[SIP消息API](#)。

```
local sdp = msg:getSdp()

-- modification of the SDP happens at this point

-- Update the SDP associated with the SIP message

msg: setSdp(sdp)
```

附註：該代碼將G.722編解碼器的a=線更改為G722，沒有點。

```
M = {}

function M.inbound_INVITE(msg)
    local sdp = msg:getSdp()

    if sdp
    then
        local g722_line = sdp:getLine("a=", "G.722")
```

```
        if g722_line
            then
                --Replace G.722 with G722. The dot is special and must be escaped using %
when using gsub.
                g722_line = g722_line:gsub("G%.722", "G722")
                sdp = sdp:modifyLine("a=", "G.722", g722_line)
                msg:setSdp(sdp)
            end
        end
    end
end

return M
```

驗證

使用本節內容，確認您的組態是否正常運作。

在應用指令碼時啟用**Trace Option**，以便驗證指令碼是否在SDL日誌中運行。

疑難排解

本節提供的資訊可用於對組態進行疑難排解。

下面是一些檢查指令碼是否未按預期工作：

- 如果指令碼本身沒有執行（在SDL日誌中查詢Before Normalization/After Normalization），則很可能該指令碼沒有正確應用到裝置上，或者存在某種語法錯誤。
- 從開發人員指南驗證CUCM規範化函式，對於Lua，使用任何可用的編譯器對其進行驗證。
- 如果指令碼確實被執行，但並未進行任何更改，則檢查指令碼中使用的方向、消息型別和邏輯。

相關資訊

- https://www.cisco.com/c/en/us/td/docs/voice_ip_comm/cucm/sip_tn/8_5_1/sip_tn/8-trace.html
- <https://supportforums.cisco.com/t5/collaboration-voice-and-video/a-guide-to-sip-normalization-on-cucm-and-lua-scripting/ba-p/3099409>
- [技術支援與文件 - Cisco Systems](#)