

Konfigurieren des SIP-Normungs-Skripts in CCM

Inhalt

[Einführung](#)

[Voraussetzungen](#)

[Anforderungen](#)

[Verwendete Komponenten](#)

[Konfigurieren](#)

[Skript auf Trunk in CUCM anwenden](#)

[Skript auf SIP-Telefon anwenden](#)

[Skript für häufige Szenarien entwickeln](#)

[Header ändern](#)

[Header entfernen](#)

[Header hinzufügen](#)

[Bearbeiten von SDP-Inhalten](#)

[Überprüfen](#)

[Fehlerbehebung](#)

[Zugehörige Informationen](#)

Einführung

In diesem Dokument wird die Verwendung des SIP-Normalisierungs-Skripts (Session Initiation Protocol) im Cisco Call Manager (CCM) mit einigen einfachen Beispielen beschrieben.

Voraussetzungen

Anforderungen

Cisco empfiehlt, über Kenntnisse in folgenden Bereichen zu verfügen:

- Cisco Unified Communications Manager (CUCM) Version 8.x und höher
- SIP-Protokoll
- Skriptwissen

Verwendete Komponenten

Die Informationen in diesem Dokument basieren auf CCM-Version 11.5.

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.

Konfigurieren

Skript auf Trunk in CUCM anwenden

Bei einem SIP-Trunk muss das Skript direkt auf den SIP-Trunk angewendet werden. Die Verwendung von Skripten in SIP-Profilen anstelle von Trunks funktioniert nicht.

Führen Sie die folgenden Schritte aus:

1. Navigieren Sie zu **Gerät > Geräteeinstellung > SIP Normalization Script Configuration**, und fügen Sie ein neues Skript hinzu, wie im Bild gezeigt.

SIP Normalization Script Configuration Related Links

Save Import File

Status: Ready

SIP Normalization Script Info

Name*

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*

System Resource Error Recovery Action*

Memory Threshold* kilobytes

Lua Instruction Threshold* instructions

Save Import File

2. Navigieren Sie zu **Gerät > Trunk**.

Wählen Sie den Trunk aus, auf den das Skript angewendet werden soll, und wählen Sie auf der Konfigurationsseite des Skripts das zuvor erstellte Skript aus. Wählen Sie die Option **Enable Trace (Trace aktivieren)** aus, um die vom Skript vorgenommenen Änderungen in den Protokollen der Signalverteilungsschicht (SDL) wie im Bild gezeigt zu drucken. Speichern Sie die Konfiguration, wenden Sie Konfig. an, und setzen Sie den Trunk zurück, damit die Änderungen wirksam werden.

Normalization Script

Normalization Script

Enable Trace

	Parameter Name	Parameter Value
1	<input type="text"/>	<input type="text"/>

Recording Information

Skript auf SIP-Telefon anwenden

Bei SIP-Telefonen müssen Sie das Skript für das SIP-Profil der Telefone verwenden.

Hier sind die Schritte, die Sie durchführen müssen.

1. Navigieren Sie zu **Gerät > Geräteeinstellung > SIP-Normalisierungsskript**, und fügen Sie das neue Skript hinzu.
2. Navigieren Sie zu **Gerät > Geräteeinstellung > SIP-Profil**. Wählen Sie das SIP-Profil aus, auf das Sie das Skript anwenden möchten, oder erstellen Sie ein neues Profil, indem Sie das Standard-SIP-Profil kopieren.
3. Wählen Sie auf der Konfigurationsseite des SIP-Profiles das Skript aus, gefolgt von Apply Config (Konfiguration anwenden) und Reset (Profil zurücksetzen).
4. Navigieren Sie zu **Gerät > Telefon**. Wählen Sie das Telefon aus, auf das Sie das Skript anwenden möchten, und ändern Sie das SIP-Profil des Telefons mit dem erstellten, gefolgt vom Speichern, wenden Sie die Konfiguration an, und setzen Sie das Telefon zurück, um Änderungen vorzunehmen.

Skript für häufige Szenarien entwickeln

Hier finden Sie die wichtigsten Schritte zur Entwicklung eines SIP-Normalisierungsskripts: [Entwicklerhandbuch für SIP-Transparenz und -Normalisierung](#).

Dieser Leitfaden enthält verschiedene Funktionen, die zur Manipulation von SIP-Nachrichten, SDP-Inhalten (Session Description Protocol) und anderen erweiterten APIs verfügbar sind.

Im Folgenden sind einige einfache Beispielskripts aufgeführt:

Header ändern

Hier ersetzen Sie die anonyme **From** Header der ausgehenden SIP-INVITE/REINVITE-Nachricht.

```
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
```

Code

M={}

Funktion

M.outbound_INVITE(msg) Erläuterung
Initialisierung des Nachrichteninhalts M erhält hier den gesamten Inhalt der SIP Nachricht.
Lua-Funktion, die auf M(SIP-Nachricht) in ausgehende Richtung INVITE/REINVITE-Nachricht angewendet wird. Die Richtung wird immer im Hinblick auf den CUCM festgelegt, ob er ein- oder ausgeht. Der Meldungstyp kann von verschiedenen Typen sein, z. B. INVITE, 183, 200.

Weitere Informationen finden Sie im [Abschnitt Übersicht im Developers Guide for SIP Normalization](#).

local from = msg.getHeader("From")	Speichert den Inhalt der From -Nachricht in lokalem variablenbasiertem Format
getHeader	Eine der Funktionen für die Normalisierung in CCM, um den Inhalt eines Headers in variabler Form abzurufen
string.gsub	Eine Lua-Funktion, um einen bestimmten Inhalt aus einer Zeichenfolge zu ersetzen.
modifizierenHeader	Auch hier ist eine Funktion in CCM verfügbar, um den Inhalt der Kopfzeile zu ändern.

Header entfernen

Hier entfernen Sie den Cisco-Guide-Header aus der eingehenden SIP-INVITE-/REINVITE-Nachricht:

```
M = {}  
  
function M.inbound_INVITE(msg)  
    msg:removeHeader("Cisco-Guid")  
  
end  
  
return M
```

Header hinzufügen

Hier fügen Sie den INFO in den Inhalt des Allow-Headers hinzu.

Dies wird im ursprünglichen Inhalt von Allow nach einem Komma hinzugefügt.

```
M = {}  
  
function M.outbound_INVITE(msg)  
    msg:addHeader("Allow", "INFO")  
  
end  
  
return M
```

Bearbeiten von SDP-Inhalten

Es stehen APIs zur Verfügung, um den SDP-Inhalt einer SIP-Nachricht zu ändern. Um SDP zu ändern, muss der SDP-Content-Text vom Lua SIP Message-Objekt mit der **getSdp()**-API abgerufen werden, die vom SIP Message-Objekt bereitgestellt wird. Das Skript kann dann die Zeichenfolgenbibliothek verwenden, die die APIs von Cisco enthält, um das SDP zu bearbeiten. Bei der Änderung wird das SDP mit der vom SIP Message-Objekt bereitgestellten **setSdp(sdp)**-API auf das SIP Message-Objekt zurückgeschrieben. Weitere Informationen zu diesen APIs finden Sie unter [SIP-Nachrichten-APIs](#).

```

local sdp = msg:getSdp()

-- modification of the SDP happens at this point

-- Update the SDP associated with the SIP message

msg: setSdp(sdp)

```

Hinweis: Der Code ändert a= Zeile für den G.722-Codec in G722 ohne Punkt.

```

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

    return M
end

```

Überprüfen

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

Aktivieren Sie **Trace Option**, wenn Sie das Skript anwenden, um zu überprüfen, ob das Skript in den SDL-Protokollen funktioniert oder nicht.

Fehlerbehebung

Dieser Abschnitt enthält Informationen, die Sie zur Fehlerbehebung bei Ihrer Konfiguration verwenden können.

Hier einige Überprüfungen, ob das Skript nicht wie erwartet funktioniert:

- Wenn das Skript nicht von selbst ausgeführt wird (achten Sie in den SDL-Protokollen auf Vor-Normalisierung/Nach Normalisierung), wurde es wahrscheinlich nicht korrekt auf das Gerät angewendet, oder es ist ein Syntaxfehler aufgetreten.
- Überprüfen Sie die CUCM-Normalisierungsfunktionen aus dem Entwicklerhandbuch, und verwenden Sie für Lua einen beliebigen verfügbaren Compiler, um diese zu überprüfen.

- Wenn das Skript ausgeführt wird, aber keine Änderungen vornimmt, überprüfen Sie die Richtung, den Meldungstyp und die im Skript verwendete Logik.

Zugehörige Informationen

- 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>
- [Technischer Support und Dokumentation - Cisco Systems](#)