

# Configurer le script de normalisation SIP dans CCM

## Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Configuration](#)

[Appliquer un script à la liaison dans CUCM](#)

[Appliquer le script sur le téléphone SIP](#)

[Développer un script pour les scénarios courants](#)

[Modifier l'en-tête](#)

[Supprimer l'en-tête](#)

[Ajouter un en-tête](#)

[Manipuler le contenu SDP](#)

[Vérification](#)

[Dépannage](#)

[Informations connexes](#)

## Introduction

Ce document décrit comment utiliser le script de normalisation SIP (Session Initiation Protocol) dans Cisco Call Manager (CCM) avec quelques exemples de base.

## Conditions préalables

### Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Cisco Unified Communications Manager (CUCM) version 8.x et ultérieure
- Protocole SIP
- Savoir de script

### Components Used

Les informations de ce document sont basées sur la version 11.5 de CCM.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Si votre réseau est en ligne, assurez-vous de bien comprendre l'incidence possible des commandes.

# Configuration

## Appliquer un script à la liaison dans CUCM

Dans le cas de la ligne principale SIP, le script doit être appliqué directement sur la ligne principale SIP. L'utilisation du script sur le profil SIP au lieu de Trunk ne fonctionnera pas.

Voici les étapes à suivre :

1. Accédez à **Device > Device Setting > SIP Normalization Script Configuration** et ajoutez un nouveau script comme illustré dans l'image.

**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. Accédez à **Device > Trunk**.

Sélectionnez la liaison sur laquelle vous voulez appliquer le script et, sur la page de configuration de celui-ci, sélectionnez le script créé précédemment. Sélectionnez l'option **Enable Trace** et elle imprime les modifications effectuées par le script dans les journaux SDL (Signal Distribution Layer), comme indiqué dans l'image. Enregistrez la configuration, Appliquez la configuration, puis réinitialisez la liaison afin que les modifications prennent effet.

**Normalization Script**

Normalization Script

Enable Trace

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

**Recording Information**

## Appliquer le script sur le téléphone SIP

Dans le cas des téléphones SIP, vous devez utiliser le script sur le profil SIP des téléphones.

Voici les étapes à suivre.

1. Accédez à **Device > Device Setting > SIP Normalization Script** et ajoutez le nouveau script.
2. Accédez à **Device > Device Setting > SIP profile**. Sélectionnez le profil SIP sur lequel vous voulez appliquer le script ou en créer un nouveau en copiant le profil SIP standard.
3. Sur la page Configuration du profil SIP, sélectionnez le script suivi de Apply Config et réinitialisez le profil.
4. Accédez à **Device > Phone**. Sélectionnez le téléphone sur lequel vous voulez appliquer le script et modifiez son profil SIP avec celui créé, puis enregistrez-le, appliquez la configuration et réinitialisez le téléphone afin d'apporter des modifications.

## Développer un script pour les scénarios courants

Voici la principale source d'informations détaillées sur le développement du script de normalisation SIP : [Guide du développeur pour la transparence et la normalisation SIP](#).

Ce guide contient différentes fonctions disponibles pour effectuer des manipulations dans les messages SIP, le contenu SDP (Session Description Protocol) et d'autres API avancées.

Voici quelques exemples de script de base :

### Modifier l'en-tête

Ici vous remplacez l'anonyme de **From Header** du message SIP INVITE/REINVITE sortant.

```
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={}

fonction

M.outbound\_INVITE(msg)

Explication

Initialisation du contenu du message. M obtient ici tout le contenu du message SIP

Fonction Lua appliquée au message M(SIP Message) dans la direction sortant message INVITE/REINVITE. La direction est toujours décidée en termes de CUCM, qu'elle soit entrante ou sortante. Le type de message peut être de différents types tels que INVITE, 183, 200.

Pour plus de détails, reportez-vous à la [section Overview du Developers Guide](#)

## [for SIP Normalization.](#)

local de = msg :	Stocke le contenu du message <b>De</b> sous forme de variable locale
getHeader(« De »)	
getHeader	Une des fonctions disponibles pour la normalisation dans CCM afin d'obtenir le contenu d'un en-tête dans une variable
chaîne.gsub	Une fonction Lua afin de remplacer un contenu particulier d'une chaîne
modifierEnTête	Encore une fois, une fonction disponible dans CCM afin de modifier le contenu de l'en-tête

## Supprimer l'en-tête

Ici, vous supprimez l'en-tête Cisco-Guide du message SIP INVITE/REINVITE entrant :

```
M = {}

function M.inbound_INVITE(msg)

    msg:removeHeader("Cisco-Guid")

end

return M
```

## Ajouter un en-tête

Dans ce cas, vous ajoutez l'INFO dans le contenu de l'en-tête Allow.

Cette valeur est ajoutée dans le contenu d'origine de Allow après une virgule.

```
M = {}

function M.outbound_INVITE(msg)

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

end

return M
```

## Manipuler le contenu SDP

Des API sont disponibles afin de modifier le contenu SDP à partir d'un message SIP. Pour modifier SDP, il doit obtenir le corps du contenu SDP de l'objet Message SIP Lua avec l'API **getSdp()** fournie par l'objet Message SIP. Le script peut ensuite utiliser la bibliothèque de chaînes qui inclut les API de Cisco afin de manipuler le SDP. Lors de la modification, le SDP est réécrit à l'objet Message SIP avec l'API **setSdp(sdp)** fournie par l'objet Message SIP. Référez-vous aux [API de messages SIP](#) pour plus d'informations sur ces API.

```
local sdp = msg:getSdp()

-- modification of the SDP happens at this point
```

```
-- Update the SDP associated with the SIP message
```

```
msg: etSdp(sdp)
```

**Note:** Le code change a= ligne pour que le codec G.722 soit G722 sans point.

```
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
```

## Vérification

Utilisez cette section pour confirmer que votre configuration fonctionne correctement.

Activez l'**option Trace** lorsque vous appliquez le script afin de vérifier si le script fonctionne ou non à partir des journaux SDL.

## Dépannage

Cette section fournit des informations que vous pouvez utiliser pour dépanner votre configuration.

Voici quelques vérifications si le script ne fonctionne pas comme prévu :

- Si le script n'est pas exécuté par lui-même (Rechercher avant la normalisation/Après la normalisation dans les journaux SDL), il est fort probable qu'il n'a pas été appliqué correctement sur le périphérique ou qu'il n'y a pas eu d'erreur de syntaxe.
- Vérifiez les fonctions de normalisation CUCM à partir du guide des développeurs et pour Lua, utilisez n'importe quel compilateur disponible afin de le vérifier.
- Si le script est exécuté mais qu'il n'apporte aucune modification, vérifiez la direction, le type de message et la logique utilisée dans le script.

## Informations connexes

- [https://www.cisco.com/c/en/us/td/docs/voice\\_ip\\_comm/cucm/sip\\_tn/8\\_5\\_1/sip\\_tn/8-trace.html](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>
- [Support et documentation techniques - Cisco Systems](#)