Améliorations architecturales CUCM 11.5.x TFTP

Contenu

Introduction

Ce document traite de la fonctionnalité d'architecture à l'échelle TFTP (Trivial File Transfer Protocol) mise en oeuvre dans le cadre de la version 11.5 de Cisco Unified Communication Manager (CUCM), la plus récente mise à niveau vers CUCM. Il s'agit d'une fonction purement technique qui vise à améliorer le service TFTP en ce qui concerne l'utilisation de la mémoire et la manière dont il gère la configuration et les fichiers statiques. La logique opérationnelle demeure la même et il n'y a aucune incidence sur les autres services fournis par TFTP.

Informations générales

Raisons pour lesquelles cette amélioration a été nécessaire et incorporée

Problème avec la conception actuelle

- La logique de fonctionnement du protocole TFTP pour les fichiers de configuration n'a pas été modifiée depuis longtemps.
- Avant la version 11.5, le service TFTP crée les fichiers de configuration et met en cache tous les fichiers de configuration en mémoire.
- Avec une plus grande capacité ajoutée à CUCM par rapport au nombre de téléphones pris en charge, l'empreinte mémoire du pied du service TFTP a augmenté de façon linéaire.
- Les futures feuilles de route devront être dotées d'une capacité supplémentaire pour les téléphones afin d'être mises en oeuvre dans CUCM.
- Par conséquent, il devient important d'examiner l'augmentation de l'empreinte mémoire du pied du service TFTP.

Heure de démarrage du service

- Dans des déploiements de taille moyenne à grande avec 20 000 à 40 000 téléphones configurés.
- Lorsqu'une modification affecte tous les téléphones, le protocole TFTP génère tous les fichiers de configuration concernés et reconstruit le cache.
- Cela augmente le temps nécessaire au démarrage du service TFTP.
- Au moment où les téléphones demandent un fichier de configuration, une réponse Occupé est envoyée au téléphone.

Présentation des fonctionnalités

La nouvelle fonctionnalité mise en oeuvre répond aux deux problèmes ci-dessus par une conception sans cache et crée le fichier de configuration à la demande. Lorsqu'une demande est envoyée à partir du téléphone, le service TFTP crée le fichier de configuration à la volée et le sert au téléphone en temps réel. Il ne met pas en cache le fichier de configuration en mémoire, ce qui à son tour réduit l'heure de début du service et l'empreinte mémoire du service TFTP.

Modifications de conception

Les modifications apportées à la conception se répartissent en deux catégories : Gestion des connexions et Génération de fichiers de configuration. Le tableau ci-dessous détaille les changements effectués dans chaque catégorie.

Gestion des connexions		Génération de fichiers configuration
HTTP	TFTP	Cadre ajouté pour les fic de configuration signés génération à la deman
La couche de service réseau est conçue pour utiliser SDL afin de gérer toutes les connexions TCP	Aucune modification n'est apportée lorsque les téléphones demandent des fichiers de configuration via UDP	

Amélioration des performances

Vous trouverez ci-dessous les améliorations apportées aux performances lors de la mise en oeuvre de cette nouvelle fonctionnalité.

- Réduction significative de l'empreinte mémoire du service TFTP
- L'empreinte mémoire est d'environ 600 Mo pour le service TFTP
- L'heure de début du service est moindre, car les fichiers ne sont pas mis en cache
- L'heure de début du service est indépendante du nombre de téléphones déployés dans le système

Chiffres de performances

	Nombre de téléphones	Temps nécessaire à la version antérieure à 11.5	Durée de la vers 11.5
Heure de début du service	20000	3 minutes 38 secondes	0 minute 19 sec
Fichiers servis sur	20000	7 minutes 24 secondes	4 minutes 0

HTTP Fichiers servis sur TFTP	20000	5 minutes 36 secondes	secondes 4 minutes 1 secondes

Note: Les chiffres ci-dessus ne proviennent pas seulement d'une série de tests, mais représentent une moyenne de plusieurs séries de tests.

Analyse du journal :

Périphériques utilisés :

CUCM version 11.5.1.10000-6

Cisco IP Communicator version 8.6.2

Requête de fichier de configuration sur HTTP avant 11.5

Demande du téléphone pour le fichier de configuration

```
00593088.000 |21:58:11.698 |AppInfo  |   TID[da900b70] HTTPEngine::getRequest(),
[0xa0d6c90~7~10.65.64.132~54462] INFO:: socket(12), ReqTimeout[60],
Request[GET /SEP000C29ED3D88.cnf.xml HTTP/1.1
```

Puisque tous les fichiers sont mis en cache après la génération, TFTP recherche le fichier de configuration mis en cache

```
00593097.000 |21:58:11.698 |AppInfo
|CReqContext::FindAndServe(1)[0xa0d6c90~7~10.65.64.132~54462]
,[(SEP000C29ED3D88.cnf.xml),(6779),(0xf388c2a8)] found in config cache
```

Le fichier de configuration est correctement servi au téléphone

```
00593102.000 |21:58:11.698 |AppInfo  |
HTTPEngine::sendResponse[0xa0d6c90~7~10.65.64.132~54462]
FileName[SEP000C29ED3D88.cnf.xml], Version[HTTP/1.1], Size[6779] 00593103.000 |21:58:11.698
|AppInfo | HTTPEngine::sendResponse[0xa0d6c90~7~10.65.64.132~54462]
INFO:: [85][HTTP/1.1 200 OK
```

Requête de fichier de configuration sur HTTP dans 11.5

Demande du téléphone pour le fichier de configuration

```
00000510.003 |21:47:40.683 |AppInfo  |   HTTPConnection::wait_SdlDataInd Printing the
HTTPRequest :
msgBuffer size [148] --: GET /SEP000C29ED3D88.cnf.xml HTTP/1.1
```

Le processus ServeFile envoie le signal 'FileRequest' à ServeDynamicFile

```
00000511.010 |21:47:40.683 |AppInfo  |   ServeFile::wait_FileRequest Sending the
FileRequest signal to ProcessServeDynamicFile process

00000511.011 |21:47:40.683 |AppInfo  |<--ServeFile::wait_FileRequest
```

```
00000512.000 |21:47:40.683 |SdlSig   |FileRequest                              |wait
|ServeDynamicFile(1,600,25,1)     |ServeFile(1,600,24,1)           |1,600,14,4.3^*^*
|*TraceFlagOverrode
```

Puisque la conception sans cache est implémentée, vous voyez que TFTP crée le fichier de configuration

```
00000512.027 |21:47:40.684 |AppInfo  |TFTPList::GetSupportsFMT(), Pkid[9e9cb809-df9f-4bce-8a41-
37cd5f7e4d21] Name[SEP000C29ED3D88] Class[1] Product[30041] Model[30016] Protocol[0],
DevProfile[0] SUPPORTs[2], Value[2]

00000512.028 |21:47:40.684 |AppInfo  |<--TFTPList::SelectByDeviceID[0,0]

00000512.029 |21:47:40.684 |AppInfo  |   ServeDynamicFile::wait_FileRequest
Build Config file for Device [SEP000C29ED3D88]
```

Le processus ServeDynamicFile envoie le signal 'FileResponse' à ServeFile

```
00000512.091 |21:47:40.686 |AppInfo  |<--ServeDynamicFile::wait_FileRequest
00000513.000 |21:47:40.686 |SdlSig   |FileResponse                             |wait
|ServeFile(1,600,24,1)           |ServeDynamicFile(1,600,25,1)     |1,600,14,4.3^*^*
|*TraceFlagOverrode

00000513.002 |21:47:40.686 |AppInfo  |   ServeFile::wait_FileResponse File
Response signal received by ServeFile process
```

Le fichier demandé est envoyé au téléphone

```
00000514.001 |21:47:40.686 |AppInfo  |-->HTTPConnection::wait_FileResponse
00000514.002 |21:47:40.686 |AppInfo  |   HTTPConnection::wait_FileResponse Requested
file FOUND... Sending file Response
00000514.003 |21:47:40.686 |AppInfo  |<--HTTPConnection::wait_FileResponse
```