

Améliorations architecturales d'échelle CUCM 11.5.x TFTP

Contenu

[Introduction](#)

[Informations générales](#)

[Problème avec la conception en cours](#)

[Entretenez commencent le temps](#)

[Vue d'ensemble des fonctionnalités](#)

[Modifications de conception](#)

[Améliorations des performances](#)

[Chiffres de représentation](#)

[Analyse de log :](#)

[Demande de fichier de configuration au-dessus de HTTP dans pré 11.5](#)

[Demande de fichier de configuration au-dessus de HTTP en 11.5](#)

Introduction

Ce document dans environ la caractéristique d'architecture d'échelle de Protocole TFTP (Trivial File Transfer Protocol) mise en application comme partie de version 11.5 du gestionnaire de Cisco Unified Communications (CUCM) la plus nouvelle augmentation à CUCM. C'est purement une caractéristique d'ingénierie afin d'améliorer le service TFTP en ce qui concerne l'utilisation de mémoire et comment il sert la configuration et les fichiers statiques. Le logic des affaires demeure le même et il n'y a aucune incidence en ce qui concerne d'autres services fournis par TFTP.

Informations générales

Raisons pour lesquelles cette amélioration a été exigée et incorporée

Problème avec la conception en cours

- La logique de la façon dont le TFTP sert le hasn de fichiers de configuration ? t changé pendant longtemps.
- Pré 11.5, service TFTP établit les fichiers de configuration et cache toute la dans-mémoire de fichiers de configuration.
- Plus de capacité étant ajouté à CUCM en ce qui concerne le nombre de téléphones pris en charge, la copie de pied de mémoire du service TFTP linéairement accrue.
- De futures feuilles de route ont la condition requise de la capacité supplémentaire pour des téléphones afin de pour être mises en application dans CUCM.
- Par conséquent, adressez les increas de la copie de pied de mémoire du service TFTP devient important.

Entretenez commencent le temps

- Dans un support à de grands déploiements avec 20k aux téléphones 40k configurés.
- Quand on apporte une modification qui affecte tous les téléphones, le TFTP établit tous les fichiers de configuration affectés et les reconstruit cache.
- Ceci augmente le moment pris pour que le service TFTP commence.
- Au moment où la demande de téléphones du fichier de configuration une réponse occupée est envoyée au téléphone.

Vue d'ensemble des fonctionnalités

La nouvelle caractéristique mise en application aborde les deux problèmes ci-dessus par une conception sans cache et établit le à la demande de fichier de configuration. Quand une demande est envoyée du téléphone, le service TFTP établit le fichier de configuration à la volée et le sert au téléphone en temps réel. Il ne cachera pas la dans-mémoire de fichier de configuration qu'à leur tour il réduisent l'heure de début de service et l'empreinte de pas de mémoire du service TFTP.

Modifications de conception

Les modifications de conception faites tombent au-dessous de deux catégories à savoir « Gestion de connexion » et « génération de fichier de configuration ». La table ci-dessous détaille les modifications faites sous chaque catégorie.

Gestion de connexion		Génération de fichier de configuration
HTTP	TFTP	
La couche de service réseau sont conçues pour employer SDL afin de manipuler toutes les connexions TCP	Aucune modifications où les téléphones demandent les fichiers de configuration au-dessus de l'UDP	Le cadre a ajouté pour la construction sur demande et signé des fichiers de configuration

Améliorations des performances

Sont ci-dessous les améliorations des performances réalisées avec l'implémentation de cette nouvelle caractéristique.

- Réduction significative d'empreinte de pas de mémoire de service TFTP
- L'empreinte de pas de mémoire est environ 600 Mo pour le service TFTP
- L'heure de début de service est moins puisque les fichiers ne sont pas cachés
- L'heure de début de service est indépendant du nombre de téléphones déployés dans le système

Chiffres de représentation

	Non des téléphones	Le temps rentrent la version pré 11.5	Version 11.5 rentrée p temps
Entretenez l'heure de début	20000	3 minutes 38 secondes	0 minute 19 secondes
Fichiers servis au-dessus du HTTP	20000	7 minutes 24 secondes	4 minutes 06 secondes

Fichiers servis au-dessus du
TFTP

20000

5 minutes 36 secondes

4 minutes 11 secondes

Remarque: Les nombres ci-dessus ne sont pas simplement d'un essai mais sont une moyenne de plusieurs essais.

Analyse de log :

Périphériques utilisés :

Version 11.5.1.10000-6 CUCM

Version 8.6.2 de Cisco IP Communicator

Demande de fichier de configuration au-dessus de HTTP dans pré 11.5

Demande de 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 cachés après construit, le TFTP trouve le fichier de configuration caché

```
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 avec succès 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
```

Demande de fichier de configuration au-dessus de HTTP en 11.5

Demande de 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 de ServeFile envoie le signal « FileRequest » à ServeDynamicFile

```
00000511.010 |21:47:40.683 |AppInfo | ServeFile::wait_FileRequest Sending the
FileRequest signal to ProcessServeDynamicFile process00000511.011 |21:47:40.683 |AppInfo |<--
ServeFile::wait_FileRequest00000512.000 |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 cacheless est mise en application, vous voyez que le TFTP établit 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 de ServeDynamicFile envoie le signal « FileResponse » à ServeFile

```
00000512.091 |21:47:40.686 |AppInfo |<--ServeDynamicFile::wait_FileRequest00000513.000  
|21:47:40.686 |SdlSig |FileResponse |wait  
|ServeFile(1,600,24,1) |ServeDynamicFile(1,600,25,1) |1,600,14,4.3^**^*  
|*TraceFlagOverrode00000513.002 |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_FileResponse00000514.002  
|21:47:40.686 |AppInfo | HTTPConnection::wait_FileResponse Requested  
file FOUND... Sending file Response00000514.003 |21:47:40.686 |AppInfo |<--  
HTTPConnection::wait_FileResponse
```