

# CUCM 11.5.x TFTP-schaalarchitectuurverbeteringen

## Inhoud

[Inleiding](#)

[Achtergrondinformatie](#)

[Probleem met huidig ontwerp](#)

[Begintijd service](#)

[Overzicht van functies](#)

[Ontwerpwijzigingen](#)

[Prestatieverbeteringen](#)

[Prestatiecijfers](#)

[Log analyse:](#)

[Configuratie-bestandsaanvraag via HTTP in pre 11.5](#)

[Configuratie-bestandsaanvraag via HTTP in 11.5](#)

## Inleiding

Dit document is beschikbaar in informatie over de schaal van het Protocol van de Bestandsoverdracht (TFTP) dat wordt geïmplementeerd als deel van Cisco Unified Communications Manager (CUCM) versie 11.5, de nieuwste optillage naar CUCM. Dit is uitsluitend een technische functie om de TFTP-service met betrekking tot het geheugengebruik te verbeteren en om de configuratie- en statische bestanden te kunnen gebruiken. De bedrijfslogica blijft dezelfde en er is geen impact op andere door TFTP verleende diensten.

## Achtergrondinformatie

Motivering voor deze verbetering

### Probleem met huidig ontwerp

- De logica van de manier waarop TFTP de configuratiebestanden serveert is lang niet veranderd.
- Pre 11.5 bouwt de dienst TFTP de configuratiebestanden en caches alle configuratiebestanden in het geheugen.
- Met meer capaciteit toegevoegd aan CUCM met betrekking tot het aantal telefoons dat wordt ondersteund, is de geheugen voetafdruk van de TFTP-service lineair toegenomen.
- Toekomstige routekaarten hebben de behoefte aan extra capaciteit voor telefoons om in CUCM te kunnen worden geïmplementeerd.
- Daarom wordt het van belang de toename van de afdruk van de geheugen-voetafdruk van de TFTP-dienst aan te pakken.

### Begintijd service

- In een middelgrote tot grote implementaties met 20.000 tot 40.000 telefoons.
- Wanneer een verandering wordt gemaakt die alle telefoons beïnvloedt, bouwt TFTP alle beïnvloede configuratiebestanden en bouwt het geheugen opnieuw op.
- Hierdoor neemt de tijd toe die nodig is om de TFTP-dienst te starten.
- Op het moment dat de telefoons om configuratiebestand vragen wordt er een druk antwoord naar de telefoon gestuurd.

## Overzicht van functies

De nieuwe geïmplementeerde optie pakt de twee hierboven beschreven problemen aan door een cache-loos ontwerp en bouwt het configuratiebestand op aanvraag op. Wanneer een verzoek van telefoon wordt verzonden, bouwt de dienst TFTP het configuratiebestand op de vlieg en dient het in real time aan de telefoon. Het zal het configuratiebestand niet in het geheugen inladen, dat op zijn beurt de service start tijd en de geheugen voetafdruk van de TFTP-service vermindert.

## Ontwerpwijzigingen

Ontwerpwijzigingen vallen onder twee categorieën, namelijk "verbodingsbeheer" en "Bestandsgeneratie van configuratie". In de onderstaande tabel worden de in elke categorie aangebrachte wijzigingen in detail beschreven.

Aansluitingsbeheer		Configuratie van bestanden
HTTP	TFTP	Framework toegevoegd voor bestelling gebouwde en ondertekende configuratiebestanden
De laag van de Netwerkservice is ontworpen om SDL te gebruiken om alle TCP verbindingen aan te kunnen	Geen veranderingen waar de telefoons de configuratiebestanden via UDP vragen	

## Prestatieverbeteringen

Hieronder staan de verbeterde prestaties bij de implementatie van deze nieuwe functie.

- Aanzienlijke vermindering van de geheugenvoetafdruk van de TFTP-service
- Geheugenvoetafdruk is ongeveer 600 MB voor de TFTP-service
- De starttijd van de service is korter omdat de bestanden niet zijn gecachaliseerd
- De tijd van de dienststart is onafhankelijk van het aantal telefoons dat in het systeem wordt ingezet

## Prestatiecijfers

	Aantal telefoons	Tijd om in te nemen Pre 11.5 versie	Tijd overgenomen in versie 11.5
Begintijd service	20000	3 minuten 38 seconden	0 minuten 19 seconden
Bestanden opgeslagen over HTTP	20000	7 minuten 24 seconden	4 minuten 06 seconden
Bestanden opgeslagen via TFTP	20000	5 minuten 36 seconden	4 minuten 11 seconden

**Opmerking:** De bovenstaande nummers bestaan niet slechts uit één testrun, maar zijn het gemiddelde van meerdere testseries.

## Log analyse:

### Gebruikte apparatuur:

CUCM versie 11.5.1.1000-6

Cisco IP Communicator versie 8.6.2

## Configuratie-bestandsaanvraag via HTTP in pre 11.5

Telefonisch aanvragen voor het configuratiebestand

```
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
```

Aangezien alle bestanden na het bouwwerk worden gecached, vindt TFTP het gecacheerde configuratiebestand

```
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
```

Het configuratiebestand is met succes aan de telefoon verzonden

```
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
```

## Configuratie-bestandsaanvraag via HTTP in 11.5

Telefonisch aanvragen voor het configuratiebestand

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

ServeFile-proces verstuurt het signaal 'FileApplication' naar server-DynamicFile

```
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
```

Aangezien cacheless ontwerp wordt geïmplementeerd, ziet u dat TFTP het configuratiebestand

bouwt

```
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]
```

**ServeDynamicFile-proces wordt het signaal 'FileResponse' naar ServeFile verzonden**

```
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
```

**Het gevraagde bestand wordt naar de telefoon verzonden**

```
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
```