

CUCM 11.5.x TFTP扩展架构改进

目录

[简介](#)

[背景信息](#)

[当前设计问题](#)

[服务启动时间](#)

[功能概述](#)

[设计更改](#)

[性能改进](#)

[性能数字](#)

[日志分析:](#)

[11.5之前版本中通过HTTP的配置文件请求](#)

[11.5中通过HTTP的配置文件请求](#)

简介

本文档介绍作为思科统一通信管理器(CUCM)版本11.5的一部分实施的简单文件传输协议(TFTP)规模架构功能，该版本是CUCM的最新升级。这纯粹是一项工程功能，目的是改进TFTP服务，使其与内存使用和如何为配置和静态文件提供服务有关。业务逻辑保持不变，对TFTP提供的其他服务没有影响。

背景信息

需要进行此改进并纳入的原因

当前设计问题

- TFTP如何提供配置文件的逻辑已经很久没有改变。
- 在11.5之前，TFTP服务会在内存中构建配置文件并缓存所有配置文件。
- 随着CUCM在支持的电话数量方面增加更多容量，TFTP服务的内存脚印呈线性增加。
- 未来规划图要求增加电话容量，以便在CUCM中实施。
- 因此，解决TFTP服务内存脚印增加的问题变得非常重要。

服务启动时间

- 在配置了2万到4万部电话的大中型部署中。
- 当做出影响所有电话的更改时，TFTP会构建所有受影响的配置文件并重建其缓存。
- 这增加了TFTP服务启动所花的时间。
- 当电话请求配置文件时，会向电话发送忙响应。

功能概述

实施的新功能通过无缓存设计解决了上述两个问题，并按需构建配置文件。从电话发送请求时，TFTP服务会即时构建配置文件并实时向电话提供。它不会缓存内存中的配置文件，从而减少TFTP服务的服务启动时间和内存占用。

设计更改

完成的设计更改分为两类，即“连接管理”和“配置文件生成”。下表详细说明了在每个类别下所做的更改。

HTTP	连接管理	TFTP	配置文件
网络服务层旨在使用SDL处理所有TCP连接		电话通过UDP请求配置文件时，无更改	为按需生成和签名配置

性能改进

以下是实施此新功能所实现的性能改进。

- 显著减少TFTP服务的内存占用
- TFTP服务的内存占用量约为600 MB
- 服务启动时间较短，因为文件未缓存
- 服务开始时间与系统中部署的电话数量无关

性能数字

	电话数量	11.5之前版本所花的时间	11.5版中花费的时间
服务开始时间	20000	3分38秒	0分19秒
通过HTTP提供的文件	20000	7分24秒	4分06秒
通过TFTP提供的文件	20000	5分36秒	4分11秒

注意：上述数字不仅来自一次测试运行，而且是多次测试运行的平均值。

日志分析:

使用的设备：

CUCM 版本 11.5.1.10000-6

Cisco IP Communicator版本8.6.2

11.5之前版本中通过HTTP的配置文件请求

从电话请求配置文件

```
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
```

由于所有文件在构建后都进行缓存，因此TFTP会查找缓存的配置文件

```
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  
配置文件已成功提供给电话
```

```
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
```

11.5中通过HTTP的配置文件请求

从电话请求配置文件

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

ServeFile进程将信号“FileRequest”发送到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
```

由于实施了无缓存设计，您会看到TFTP会构建配置文件

```
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进程将信号“FileResponse”发送到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
```

请求的文件将发送到电话

```
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
```