

CUCM 11.5.x архитектурные улучшения масштаба TFTP

Содержание

[Введение](#)

[Общие сведения](#)

[Проблема с текущим Дизайном](#)

[Сервисное время запуска](#)

[Обзор функций](#)

[Изменения структуры](#)

[Повышение производительности](#)

[Графики производительности](#)

[Анализ журнала:](#)

[Запрос файла конфигурации по HTTP в пред 11.5](#)

[Запрос файла конфигурации по HTTP в 11.5](#)

Введение

Этот документ в приближенно опции архитектуры масштаба Протокола TFTP, реализованной как часть версии 11.5 Cisco Unified Communication Manager (CUCM) новейший подъем к CUCM. Это - просто техническая функция для улучшения Сервиса TFTP относительно использования памяти и как это служит конфигурации и статическим файлам. Бизнес-логика остается тем же и нет никакого влияния относительно других сервисов, предоставленных TFTP.

Общие сведения

Причины, почему это улучшение требовалось и включалось

Проблема с текущим Дизайном

- Логика того, как TFTP служит файлам конфигурации, не изменялась в течение длительного времени.
- Пред 11.5, Сервис TFTP создает файлы конфигурации и кэши все файлы конфигурации в оперативной памяти.
- С большей емкостью, добавленной к CUCM относительно количества поддерживаемых телефонов, линейно увеличился след памяти Сервиса TFTP.
- Будущие планы развития имеют требование дополнительной емкости для телефонов, чтобы быть внедренными в CUCM.
- Следовательно, обратитесь к increas следа памяти Сервиса TFTP, становится важным.

Сервисное время запуска

- В настроены средние и крупные развертывания с 20к к телефонам 40к.
- Когда изменение внесено, который влияет на все телефоны, TFTP создает все файлы конфигурации, на которые влияют, и восстанавливает его кэш.
- Это увеличивает время, потраченное для Сервиса TFTP для начала.
- В то время, когда запрос телефонов о файле конфигурации занятый ответ отправлен к телефону.

Обзор функций

Новая характеристика внедрила адреса, вышеупомянутые две проблемы кэшем меньше разрабатывают и создают файл конфигурации по требованию. Когда запрос отправлен с телефона, Сервис TFTP создает файл конфигурации на лету и служит ему телефону в режиме реального времени. Это не будет кэшировать файл конфигурации в оперативной памяти, который в свою очередь это уменьшает сервисное время начала и объем потребляемой памяти Сервиса TFTP.

Изменения структуры

Сделанные изменения структуры подпадают под две категории а именно, 'Управление соединением' и 'Создание файла конфигурации'. Ниже таблицы детализирует изменения, сделанные под каждой категорией.

Управление соединением HTTP	Tftp	Создание файла конфигурации Платформа добавила для требованию сборки и подписанных файлов конфигурации
Уровень сетевого обслуживания разработан для использования SDL для обработки всех TCP - подключений	Никакие изменения, где телефоны запрашивают файлы конфигурации по UDP	

Повышение производительности

Ниже повышения производительности achieved с реализацией этой новой характеристики.

- Существенное снижение в объеме потребляемой памяти Сервиса TFTP
- Объем потребляемой памяти составляет приблизительно 600 МБ для Сервиса TFTP
- Сервисное время начала меньше, так как не кэшируются файлы
- Сервисное время начала независимо от количества телефонов, развернутых в системе

Графики производительности

	Нет. из телефонов	Время принимает Пред 11.5 версий	Время, Потраченное в 11 версиях
Сервисное время начала	20000	3 минуты 38 секунд	0 минут 19 секунд
Файлы, подаваемые по HTTP	20000	7 минут 24 секунды	4 минуты 06 секунд
Файлы, подаваемые по TFTP	20000	5 минут 36 секунд	4 минуты 11 секунд

Примечание: Вышеупомянутые номера не только от одного тестового прогона, но и среднее число нескольких тестовых прогонов.

Анализ журнала:

Используемые устройства:

Версия 11.5.1.10000-6 CUCM

Версия 8.6.2 IP-коммуникатора Cisco

Запрос файла конфигурации по HTTP в пред 11.5

Запрос с телефона для файла конфигурации

```
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
```

Так как все файлы кэшируются, после того, как создано, TFTP находит кэшируемый файл конфигурации

```
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
```

Файл конфигурации успешно подается телефону

```
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
```

Запрос файла конфигурации по HTTP в 11.5

Запрос с телефона для файла конфигурации

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

Процесс ServeFile передает сигнальный 'FileRequest' к 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
```

С тех пор cacheless дизайн внедрен, вы видите, что TFTP создает файл конфигурации

```
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]
```

Процесс ServeDynamicFile передает сигнальный 'FileResponse' к 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
```

Запрошенный файл передается телефону

```
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
```