

# Поймите Речевой Расширяемый язык разметки гипертекста (XML) (VXML) / Кэш HTTP клиентского голосового портала (CVP) для Медиа-файлов

## Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Общие сведения](#)

[Приглашение шлюза кэширующиеся факторы](#)

[Как определить, кэшируется ли шлюз должным образом](#)

[Вычислите FreshTime](#)

[Удаление устаревших кэшированных записей](#)

[Команда Audio-Prompt Load](#)

[Время даты](#)

## Введение

Этот документ описывает Речевой Расширяемый язык разметки гипертекста (XML) (VXML) / кэш Протокола HTTP клиентского голосового портала (CVP) для медиа-файлов.

## Предварительные условия

### Требования

Компания Cisco рекомендует предварительно ознакомиться со следующими предметами:

- Шлюз VXML
- CVP

### Используемые компоненты

Настоящий документ не имеет жесткой привязки к каким-либо конкретным версиям программного обеспечения и оборудования.

Сведения, представленные в этом документе, были получены от устройств, работающих в специальной лабораторной среде. Все устройства, описанные в этом документе, были запущены с чистой (стандартной) конфигурацией. В рабочей сети необходимо изучить потенциальное воздействие всех команд до их использования.

## Общие сведения

В Кэше Клиента HTTP существует два типа кэша, вовлеченного в хранение медиа-файлов: кэш Медиаплеера IVR и кэш Клиента HTTP.

Параметр кэша сервера перезаписывает параметры настройки клиента HTTP, эти параметры передаются от сервера или через заголовки сообщения http, или через vxml сценарии приложений.

## Приглашение шлюза кэширующиеся факторы

Шаг 1. Когда аудио приглашения сохранены на медиасerverе HTTP, надлежащее приглашение шлюза, кэширующиеся методы необходимы для оптимизации и производительности шлюза и потребления пропускной способности сети. Снижения производительности шлюза приблизительно на 35-40%, если кэширование отключено полностью.

Настраивать кэширование на множестве шлюзов следование шлюза:

- .. ivr вызывают память 15000
- .. файл 500 http client cache memory
- .. пул http client cache memory 15000

**Примечание:** Файл `http client cache memory` представляет самый большой файл приглашения размера (в Кбайтах), который может кэшироваться. В целом приглашения клиента, больше, чем 500К (приблизительно минута в длине), должны быть разбиты в меньшие, более управляемые части для упрощения загрузки и кэширования. Например, музыка очереди могла быть повторяющейся петлей 30-секундного приглашения. Обратите внимание также, что, потому что приглашения переданы потоком, приглашение не будет кэшироваться, пока не будет игратья целое приглашение. Поэтому рекомендуется сделать приглашения управляемым размером.

Шаг 2. Синхронизируйте дату и время между шлюзом и медиасerverом HTTP.

**Примечание:** Синхронизация не должна быть точной, но по крайней мере в течение минуты или два. Времена, которые "not synchronized", могут заставить приглашения никогда не обновлять, или они обновят с каждым вызовом, оба из которых являются нежелательным поведением.

Шаг 3. На медиасerverе, набор истечение содержания (например, 15 минут).

**Примечание:** В IIS это сделано под вкладкой HTTP Header. Приглашение шлюза будет обновлено после на этот раз период. Выбранный период должен отразить, как чаще г повторно записывают приглашения и сколько времени вы готовы ждать для имения новой быстрой загрузки после модификации.

Программы> Средства администрирования> Диспетчер IIS

Перейдите к файлу .wav, который вы хотите модифицировать.

Щелкните правой кнопкой> Свойства> Заголовки HTTP

Включите истечение Содержания.

## Как определить, кэшируется ли шлюз должным образом

Чтобы определить, настроили ли вы должным образом кэширование шлюза, сделайте придерживающееся:

IIS входит в систему записей медиасервера каждый раз запросы клиента приглашение. Если кэширование установлено правильно, эти запросы появляются приблизительно каждые X минуты (X, то, что было определено как интервал обновления в Шаг 3 выше) для какого-то конкретного приглашения. Журнал расположен в: C : \WINNT\system32\LogFiles\W3SVC1\ex\*

или

Сделайте **show http client cache** на шлюзе. **Новый Столбец времени** должен равняться набору периода времени обновления на медиасервере HTTP. Например, если бы период обновления был установлен в 15 минут, то это должно сказать 900 секунд. Столбец **Age** показывает, сколько секунд прошло, так как приглашение было в последний раз обновлено. В целом этот номер является меньше, чем **Новое Время**. Однако, если никакой вызов недавно никогда не обратился к приглашению, этот номер может быть больше, чем новое время. Приглашения только обновлены, когда инициировано вызовом, и быстрое **Новое Время** истекло. Если **Новое Время** очень максимальное значение, единственный способ удалить приглашение из кэша (кроме команд hidden) состоит в том, чтобы повторно загрузить шлюз.

Намного легче только добавить заголовок как реальный Заголовок HTTP через IIS.

Это может быть сделано через IIS 6 или 7.

<http://weblogs.asp.net/joelvarty/archive/2009/03/23/force-ie7-compatibility-mode-in-ie8-with-iis-settings.aspx>

## Вычислите FreshTime

Существует несколько переменных, которые могут влиять на FreshTime файла, такого как: заголовки сообщения http от сервера и значение cache refresh, настроенное через CLI, и т.д. Таким образом, как мы знаем, какое значение файл использует для его FreshTime? FreshTime файла определен в следующих приоритетах:

1) Если один из заголовков сообщения http содержит придерживающееся, когда файл загружен от http server:

Контроль кэша: max-age = <оценивает в секундах>

Затем вышеупомянутое <значение в секундах> будет использоваться в качестве FreshTime для этого файла.

2) Если (1) не присутствует, но следующие два заголовка включены в сообщение http:

Истекает: <время даты окончания действия>

Дата: <Время текущей даты>

Затем различие <время даты окончания действия> - <Время текущей даты> будет использоваться в качестве FreshTime для этого файла.

3) Спецификация HTTP/1.1, RFC 2616 (Транспортный протокол HyperText), рекомендует, чтобы или заголовков сообщения http, как описано в (1) или (2) выше присутствовали. Если сервер будет не в состоянии передать оба (1) или (2) в его ответе HTTP, то мы возьмем 10% различия между Датой и Модифицируемый в последний раз от заголовков сообщения:

Модифицируемый в последний раз: <модифицируемое в последний раз время даты>

Дата: <Время текущей даты>

Таким образом, FreshTime для этого файла вычислен как:

$$\text{FreshTime} = 10\% \times ((\text{Модифицируемый в последний раз}) - (\text{Дата}))$$

4) Наконец, когда CLI config cache refresh играет роль, это. CLI позволяет пользователю назначать эвристическое значение FreshTime на файлы как предварительное значение в случае, если это ни один из вышеупомянутых (1) - (3) заголовки сообщения присутствует.

```
c5400-02(config)#http client cache refresh ?
```

```
<1-864000> Time value in seconds
```

Значение обновления по умолчанию составляет 86400 секунд (24 часа).

**Примечание:** Когда любой из заголовков сообщения (1) - (3) присутствует, настроенный http client cache refresh не имеет никакого эффекта на файлы.

**Примечание:** Этот CLI, если в действительности, не обратной силы. Т.е. недавно настроенное значение обновления только применяется к новым входящим файлам. Это не имеет никакого эффекта на записи уже в кэше.

## Удаление устаревших кэшированных записей

**Примечание:** Маршрутизатор никогда не обновляет устаревших файлов самостоятельно автоматически.



самостоятельно, потому что это - легитимный владелец той записи. Если устаревшая запись имеет подсчет ссылок 1 на нем, и это не имеет никакого родителя или потомка, связанного с ним, означая, что файл не посреди загрузки обновления, клиент http перезванивает для уведомления Медиаплеера для выпуска этой устаревшей записи.

## Команда Audio-Prompt Load

Иногда, это может быть выбираемо или необходимо вручную загрузить аудио файл в маршрутизатор. К настоящему времени нам уже говорят, что маршрутизатор автоматически не переходит к http server для обновления устаревших кэшированных записей. Те записи обновлены только, когда они необходимы. Ручная загрузка может преодолеть эту проблему.

Другой сценарий ручная загрузка может быть полезной, должен предварительно загрузить большую звуковую подсказку в непотоковом режиме. Это может быть сделано, прежде чем первый вызов получен так, чтобы абонент не испытывал задержки быстрой загрузки.

Для ручной загрузки определенного аудио файл введите следующую команду CLI:

```
<url> audio-prompt load
```

Вышеупомянутый <url> - то, где аудио файл находится на сервере. Конечно, кэш клиента http, как ожидают, будет должным образом настроен, чтобы сохранить этот файл в кэше.

**Примечание:** Если <url> является активным приглашением, т.е. в настоящее время в воспроизведении, этот CLI не вступает в силу.

## Время даты

Кроме того, удостоверьтесь, что синхронизируется дата и время между шлюзом и медиасервером HTTP. Это - необходимость.

**% Warning:** Не используйте **ясный кэш клиента http** в GW VXML. Если эта команда вызвана на очень загруженном/активном GW VXML, она, как известно, вызывает проблемы, повреждение памяти и сбои. В основном использование **ясного кэша клиента IP http** всему не рекомендуют. То, что это делает, является им, обновляет все записи от кэша, и что происходит, он, создает и удаляет узлы из связанного списка кэша, который вызывает некоторые проблемы. Команда в процессе того, чтобы быть удаленным из IOS. Рекомендуемая команда является **set http client cache stale**, что делает эта команда он, просто обновляет недавно измененную часть кэша.