

Остановка видеопотока со стороны WCS-сервера

- [Описание](#)
- [Keep-Alive видеопотока](#)
 - [REST API](#)
 - [Серверная часть](#)
- [REST-вызовы для остановки потока](#)
 - [REST-методы и статусы ответа](#)
 - [Параметры](#)
 - [Фильтрация потоков по параметрам](#)
 - [Отправка REST-запроса к WCS-серверу](#)
- [Возобновление потока](#)
- [Известные проблемы](#)

Описание

Любой поток, опубликованный или захваченный на WCS, может быть остановлен на стороне сервера. Для остановки публикации или воспроизведения потока со стороны WCS-сервера доступны два способа

1. Возврат кода состояния HTTP 403 FORBIDDEN на keep-alive запрос для потока
2. REST-запрос к WCS-серверу

Keep-Alive видеопотока

Keep-alive сообщения могут быть использованы для завершения потоков со стороны WCS-сервера.

Для этого web-сервер, на котором сконфигурирован REST Hooks, должен вернуть код состояния HTTP 403 FORBIDDEN на keep-alive запрос для потока.

Таким образом могут быть остановлены публикация или воспроизведение потока, или и то, и другое.

Чтобы различить опубликованные и воспроизводимые потоки, скрипт web-сервера должен обрабатывать 'published' параметр [метода StreamKeepAliveEvent](#)

REST API

Сконфигурируйте web-сервер [с REST Hooks](#). Он должен обрабатывать keep-alive запросы от WCS-сервера, используя, например, PHP-скрипт, и определять [restClientConfig](#) для метода [StreamKeepAliveEvent](#)

```
"StreamKeepAliveEvent" : {  
  "clientExclude" : "",  
  "restExclude" : "sdp",  
  "restOnError" : "FAIL",  
  "restPolicy" : "NOTIFY",  
  "restOverwrite" : ""  
}
```

Серверная часть

Чтобы включить отправку keep-alive сообщений для потоков, необходимо:

- 1) Включить настройку keep-alive в [flashphoner.properties](#)

```
keep_alive_streaming_sessions_enabled=true
```

- 2) Определить keep-alive интервал и приложение, которое будет принимать ответы на keep-alive REST-запросы

```
streaming_sessions_keep_alive_interval=10000  
streaming_sessions_keep_alive_app_keys=defaultApp
```

Может быть использовано другое, чем 'defaultApp', приложение.

Используйте [Command Line Interface](#), команду 'show apps', чтобы просмотреть список приложений и их ключи.

3) Добавить к этому приложению REST-метод [StreamKeepAliveEvent](#), используя следующую команду

```
add app-rest-method defaultApp StreamKeepAliveEvent
```

4) Подключить REST API к приложению из командной строки

```
update app defaultApp http://my-web-server.com/MyAPI
```

Здесь:

- [my-web-server.com](#) - это адрес web-сервера,
- MyAPI - это путь, по которому расположен REST API.

REST-вызовы для остановки потока

Для остановки потока используется REST-вызов /stream/terminate

REST-запрос должен быть HTTP/HTTPS POST запросом в таком виде:

- HTTP:<http://streaming.flashphoner.com:8081/rest-api/stream/terminate>
- HTTPS:<https://streaming.flashphoner.com:8444/rest-api/stream/terminate>

Здесь:

- [streaming.flashphoner.com](#) - адрес WCS-сервера
- 8081 - стандартный REST / HTTP порт WCS-сервера
- 8444 - стандартный HTTPS порт
- rest-api - обязательная часть URL
- /stream/terminate - используемый REST-метод

REST-методы и статусы ответа

REST-метод	Пример тела REST-запроса	Пример тела REST-ответа	Статусы ответа
/stream /terminate	<pre>{ "mediaSessionId" : "41c3f621-a847-4639" }</pre>		200 - Команда на завершение потока принята 404 - Поток не найден

Параметры

Имя параметра	Описание	Пример
mediaSessionId	Идентификатор медиа сессии	41c3f621-a847-4639
name	Имя потока	streamName
published	Если true, то поток является опубликованным; если false, то поток воспроизводится	true
status	Текущий статус потока	PUBLISHING

Фильтрация потоков по параметрам

Параметры запроса /stream/terminate рассматриваются как фильтры, будут остановлены все потоки, соответствующие этим фильтрам. Например, можно остановить всех подписчиков для всех опубликованных потоков

```
{"published": false}
```

или всех подписчиков для определенного потока

```
{"name": "streamName", "published": false}
```

Можно остановить публикацию по статусу

```
{"name": "streamName", "status": "PUBLISHING"}
```

Также можно остановить все потоки в определенных медиасессиях

```
{"mediaSessionIds": ["41c3f621-a847-4639", "554916e0-931c-2479"]}
```

или в одной медиасессии

```
{"mediaSessionId": "41c3f621-a847-4639"}
```

Отправка REST-запроса к WCS-серверу

Для отправки REST-запроса к WCS-серверу необходимо использовать REST-клиент, например расширение [Advanced REST Console](#) для браузера Chrome.

Возобновление потока

После остановки потока со стороны сервера публикация и воспроизведение потока могут быть возобновлены.

Например, в случае WebRTC / WebSocket потока, опубликованного с использованием клиента `client2/examples/min/streaming.html` и воспроизводимого с использованием клиента `client2/examples/demo/streaming/player/player.html`:

1) Поток опубликован

WCS Server URL:

Stream name:

Name: streamName Status: PUBLISHING



2) Поток воспроизводится

Player



WCS URL

Stream

3) Воспроизведение потока остановлено со стороны WCS-сервера

WCS URL

Stream

Volume

Full Screen

FAILED

Stopped by publisher stop

Публикация потока будет возобновлена после нажатия кнопки 'publish'.
Воспроизведение потока будет возобновлено нажатия кнопке 'Start'.

Известные проблемы

1) "Can't find mediasession" сообщения в flashphoner.log

Симптомы: В случае остановки со стороны сервера RTMFP-потока, опубликованного с использованием клиента `client2/examples/demo/streaming/flash_client/chat.html`, публикация потока останавливается, однако сессия потока не уничтожается. В логе [flashphoner.log](#) появляются сообщения "Can't find mediasession".

Решение: Для завершения сессии необходимо кликнуть кнопку 'Stop video' на странице клиента.