

# В плеере по RTMP

- Описание
  - RTMP-кодеки
  - Схема работы
- Краткое руководство по тестированию
  - Трансляция видеопотока на сервер и воспроизведение его по RTMP в программном плеере
- Последовательность выполнения операций (Call flow)
- Обработка параметров, указанных в URL потока
  - Передача параметров соединения как параметров потока
- Управление порядком дорожек в RTMP потоке
- Подавление звука при воспроизведении RTMP
- Известные проблемы

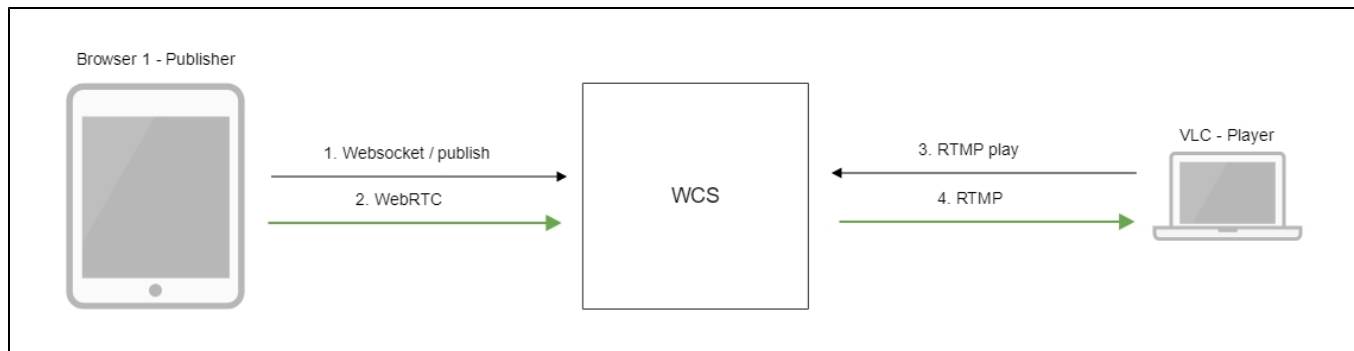
## Описание

Поток, опубликованный на WCS-сервере, можно воспроизвести по RTMP в стороннем плеере. В этом случае WCS сам выступает в качестве RTMP-источника.

## RTMP-кодеки

- Видео: H.264
- Аудио: AAC, G.711, Speex

## Схема работы



1. Браузер устанавливает соединение с сервером по Websocket
2. Браузер захватывает камеру и микрофон и отправляет на сервер WebRTC-поток
3. Проигрыватель VLC устанавливает соединение с сервером по RTMP
4. Проигрыватель VLC получает поток с сервера и воспроизводит его

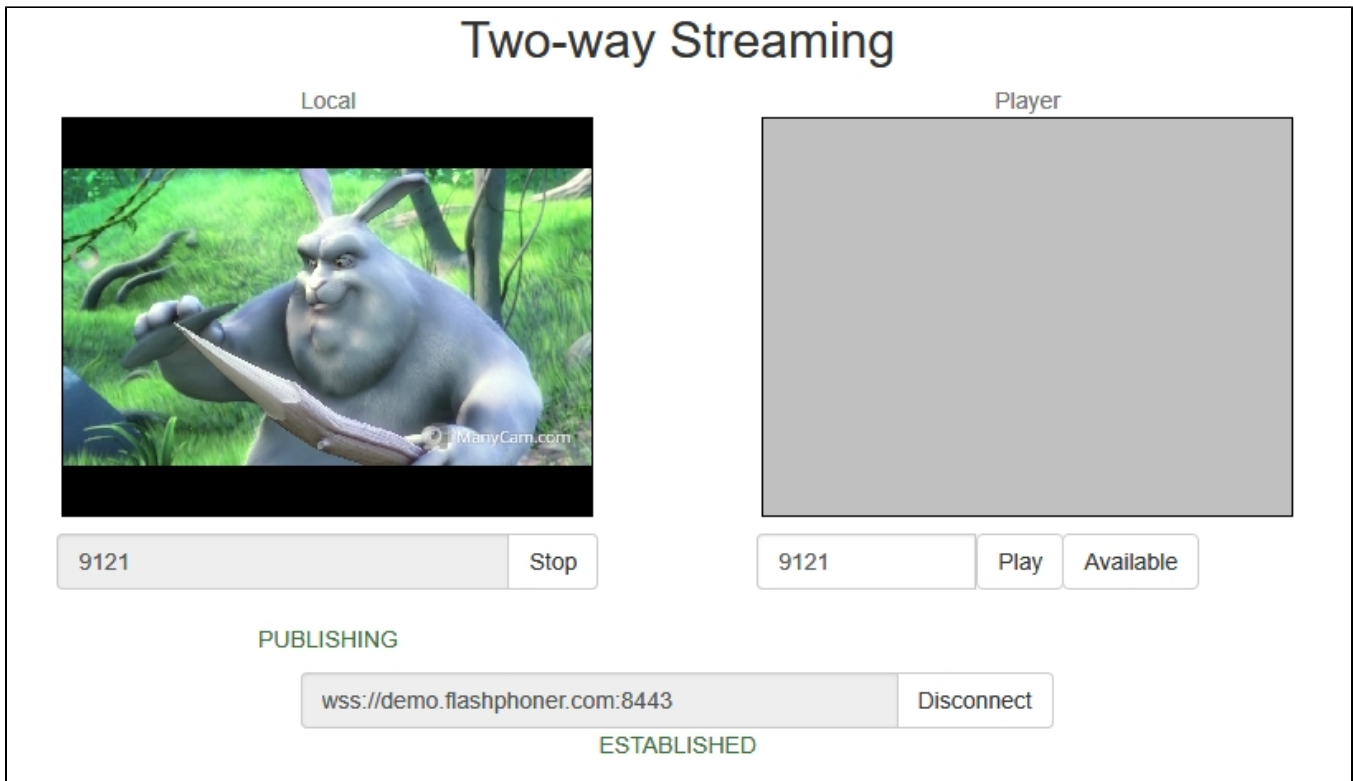
## Краткое руководство по тестированию

### Трансляция видеопотока на сервер и воспроизведение его по RTMP в программном плеере

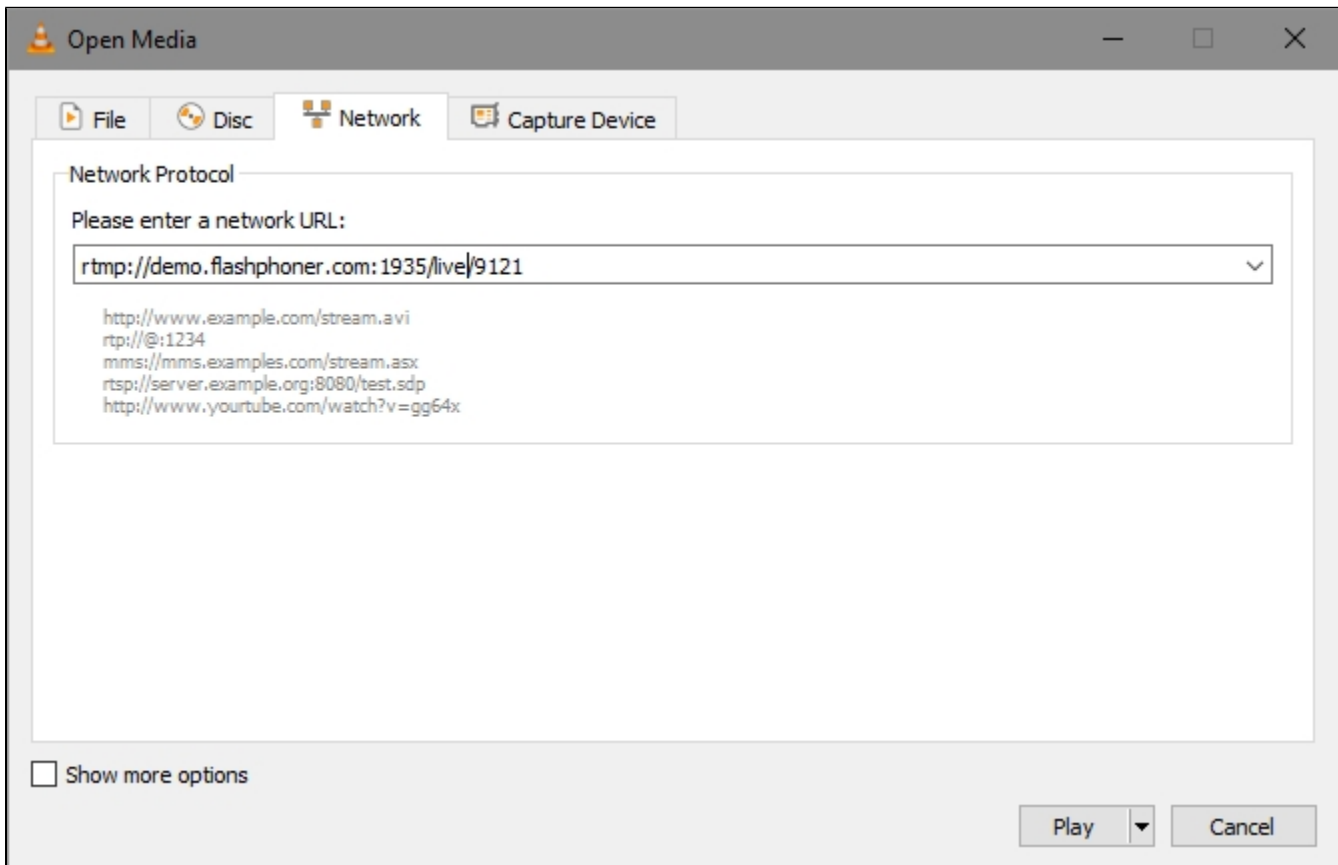
1. Для теста используем:

- демо-сервер [demo.flashphoner.com](http://demo.flashphoner.com);
- веб-приложение [Two Way Streaming](#) для публикации потока;
- проигрыватель VLC для воспроизведения потока.

2. Откройте веб-приложение Two Way Streaming. Нажмите Connect, затем Publish. Скопируйте идентификатор потока:

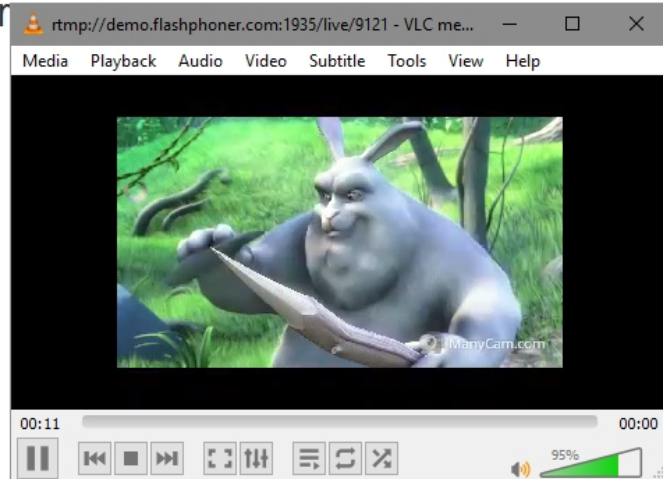
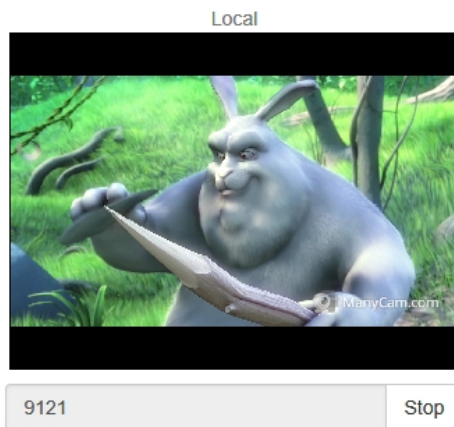


3. Запустите VLC, выберите пункт меню "Медиа - Открыть URL". Введите URL WCS-сервера с указанием идентификатора потока, в данном примере  
rtmp://demo.flashphoner.com:1935/live/9121:



4. Нажмите кнопку "Воспроизвести". Проигрыватель начнет воспроизведение потока:

## Two-way Streaming



PUBLISHING

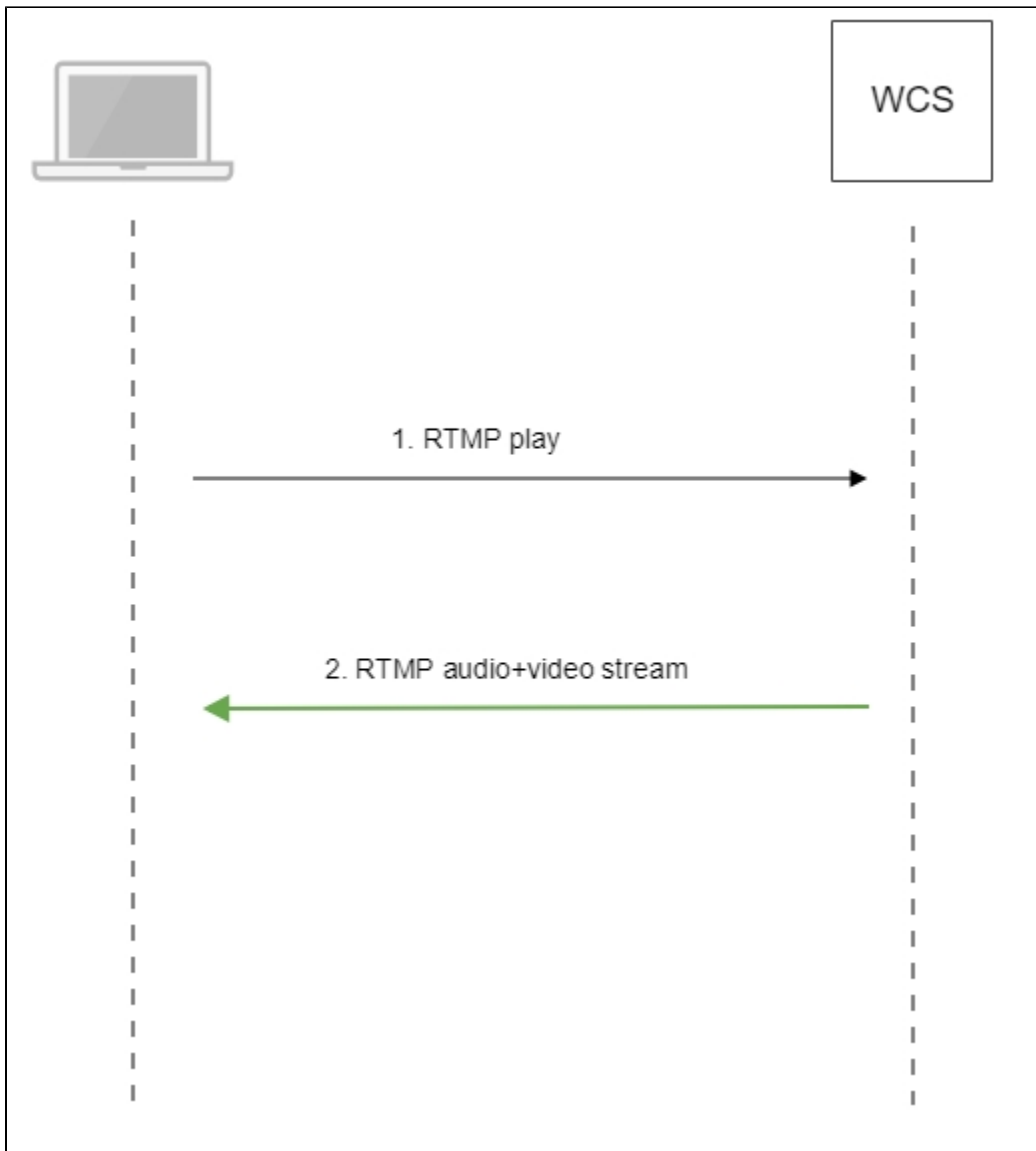
wss://demo.flashphoner.com:8443

Disconnect

ESTABLISHED

## Последовательность выполнения операций (Call flow)

Ниже описана последовательность вызовов при воспроизведении потока по RTMP в программном плеере.



1. Программный проигрыватель устанавливает соединение с WCS-сервером по RTMP

2. Программный проигрыватель получает от WCS медиапоток.

## Обработка параметров, указанных в URL потока

При публикации или воспроизведении RTMP-потока на WCS, в URL потока могут быть указаны параметры RTMP-соединения и параметры потока:

```
rtmp://host:1935/live?connectParam1=val1&connectParam2=val2/streamName?streamParam1=val1&streamParam2=val2
```

Здесь

- host - WCS-сервер;
- connectParam1, connectParam2 - параметры RTMP-соединения;
- streamName - имя потока на сервере;
- streamParam1, streamParam2 - параметры потока.

WCS-сервер передает указанные параметры бэкенд-серверу в [REST hook](#), в поле `custom`, например:

Параметры соединения

```
URL:http://localhost:8081/apps/EchoApp/connect
OBJECT:
{
  "nodeId" : "Qb3rAjf3lzoy6PEl1WZkUhrG1DsTykgj@192.168.1.1",
  "appKey" : "flashStreamingApp",
  "sessionId" : "/127.0.0.1:5643/192.168.1.1:1935",
  "useWsTunnel" : false,
  "useWsTunnelPacketization2" : false,
  "useBase64BinaryEncoding" : false,
  "keepAlive" : false,
  "custom" : {
    "connectParam1" : "val1",
    "connectParam2" : "val2"
  },
  "login" : "rQq83sodiCPY0pJXCxGO"
}
```

#### Параметры публикации

```
URL:http://localhost:8081/apps/EchoApp/publishStream
OBJECT:
{
  "nodeId" : "Qb3rAjf3lzoy6PEl1WZkUhrG1DsTykgj@192.168.1.1",
  "appKey" : "flashStreamingApp",
  "sessionId" : "/127.0.0.1:5643/192.168.1.1:1935",
  "mediaSessionId" : "627990f9-8fe5-4e92-bb2a-863cc4eb43de",
  "name" : "stream1",
  "published" : true,
  "hasVideo" : false,
  "hasAudio" : true,
  "status" : "NEW",
  "record" : true,
  "width" : 0,
  "height" : 0,
  "bitrate" : 0,
  "minBitrate" : 0,
  "maxBitrate" : 0,
  "quality" : 0,
  "mediaProvider" : "Flash",
  "custom" : {
    "streamParam1" : "val1",
    "streamParam2" : "val2"
  }
}
```

#### Параметры воспроизведения

```
URL:http://localhost:8081/apps/EchoApp/playStream
OBJECT:
{
  "nodeId" : "Qb3rAjf3lzoy6PEl1WZkUhrG1DsTykgj@192.168.1.1",
  "appKey" : "flashStreamingApp",
  "sessionId" : "/127.0.0.1:5643/192.168.1.1:1935",
  "mediaSessionId" : "stream1/127.0.0.1:5643/192.168.1.1:1935",
  "name" : "stream1",
  "published" : false,
  "hasVideo" : true,
  "hasAudio" : true,
  "status" : "NEW",
  "record" : false,
  "width" : 0,
  "height" : 0,
  "bitrate" : 0,
  "minBitrate" : 0,
  "maxBitrate" : 0,
  "quality" : 0,
  "mediaProvider" : "Flash",
  "custom" : {
    "streamParam1" : "val1",
    "streamParam2" : "val2"
  }
}
```

Эту возможность можно использовать, например, для авторизации клиента на бэкенд-сервере при публикации или воспроизведения RTMP-потока на WCS.

## Передача параметров соединения как параметров потока

В некоторых случаях возникает необходимость передать параметры соединения, например, параметры авторизации, как параметры потока, например

```
rtmp://test.flashphoner.com:1935/live/test?auth=key
```

Эта возможность включается настройкой

```
rtmp_use_stream_params_as_connection=true
```

В этом случае вышеприведенный пример RTMP URL будет аналогичен следующему

```
rtmp://test.flashphoner.com:1935/live?auth=key/test
```

## Управление порядком дорожек в RTMP потоке

Большинство плееров на различных платформах предполагают, что первой в RTMP потоке идет видеодорожка. Для того, чтобы гарантировать такой порядок и отсылать в первую очередь видеоданные, необходимо установить следующий параметр в файле [flashphoner.properties](#):

```
rtmp_send_video_first=true
```

Необходимо отметить, что, если эта настройка активна, при публикации потока, содержащего только аудиодорожку, такой поток не будет воспроизводиться по RTMP, поскольку аудиоданные не будут высылаться клиенту.

## Подавление звука при воспроизведении RTMP

При воспроизведении потока, опубликованного на сервере, как RTMP, звук может быть отключен. Для этого в RTMP URL необходимо передать следующий параметр:

```
rtmp://yourserver:1935/live?suppress_sound=true/streamName
```

При этом звуковая дорожка в проигрываемом потоке заменяется тишиной.

## Известные проблемы

1. При проигрывании FullHD, 2K, 4K потоков с большими размерами кадров, пакеты могут не помещаться в буфер сокета на отправку, из-за чего в некоторых плеерах могут наблюдаться артефакты

Симптомы: при проигрывании потока на хорошем канале эпизодически появляются артефакты

Решение: включить буферизацию RTMP пакетов при отправке настройкой

```
rtmp.server_buffer_enabled=true
```