

Описание REST API

- [API адреса](#)
- [Настройки](#)
 - [Порты](#)
 - [Междоменные запросы](#)
 - [Авторизация](#)
- [Тестирование](#)

API адреса

Web Call Server использует два URL-адреса для REST API

HTTP	HTTPS
http://host:8081/rest-api	https://host:8444/rest-api

Для того чтобы вызвать REST-метод, сформируйте запрос следующего формата:

- Протокол: HTTP или HTTPS
- Content Type: application/json
- Метод: POST
- Тело запроса: JSON

Настройки

Порты

По умолчанию, REST HTTP запросы принимаются на порту 8081, REST HTTPS - на порту 8444. Эти значения определяются следующими настройками в файле [flashphoner.properties](#)

```
http.port=8081
https.port=8444
```

Междоменные запросы

Междоменные запросы управляются следующими настройками в файле [flashphoner.properties](#)

```
rest_access_control_allow_origin=*
rest_access_control_allow_headers=content-type,x-requested-with
rest_access_control_allow_methods=POST
```

По умолчанию разрешены междоменные запросы с любого адреса, чтобы обеспечить перенаправление портов.

Авторизация

По умолчанию, REST API запросы обрабатываются без авторизации. При необходимости, может быть включена Basic авторизация при помощи следующего параметра в файле [flashphoner.properties](#)

```
disable_rest_auth=false
```

Пользователи REST API при включении авторизации управляются из [командной строки ядра WCS](#).

Для междоменных запросов необходимо добавить заголовки при помощи следующей настройки в файле [flashphoner.properties](#):

```
rest_access_control_headers=Access-Control-Allow-Origin: <origin>;Access-Control-Allow-Credentials: true;Access-Control-Allow-Methods: GET, POST, PUT, DELETE, OPTIONS;Access-Control-Allow-Headers: Origin, X-Requested-With, Content-Type, Accept, Authorization
```

Здесь <origin>- источник запроса, например <http://test.flashphoner.com>.

Тестирование

Для тестирования можно использовать REST-клиент. Например [Advanced REST Console](#) для браузера Google Chrome.

На скриншоте приведен пример обращения к серверу с получением списка текущих потоков REST методом `stream/find_all`

The screenshot displays the Advanced REST Client interface. At the top, the Method is set to **POST** and the URL is `http://demo.flashphoner.com:8081/rest-api/stream/find_all`. A yellow **SEND** button is visible on the right. Below the URL bar, there are tabs for **HEADERS**, **BODY**, **AUTHORIZATION**, and **VARIABLES**. The **HEADERS** tab is active, showing a single header: **Content-Type** with the value `application/json`. Below the headers, there is a section for **Response** showing a **200 OK** status. The response body is a JSON object with the following structure:

```
1 {
2   "appKey": "flashStreamingApp",
3   "sessionId": "/36.71.175.77:63690/188.40.244.249:1935",
4   "mediaSessionId": "02d63bad-0a05-4ea6-b3be-c616cd44529d",
5   "name": "123",
6   "published": true,
7   "hasVideo": true,
8   "hasAudio": true,
9   "status": "PUBLISHING",
10  "sdp": "v=0\r\no=- 1988962254 1988962254 IN IP4 0.0.0.0\r\ns=- IN IP4 0.0.0.0\r\na=rtcp:0 0\r\na=sdplang:en\r\nm=video 0 RTP/AVP 119
11      127\r\na=rtpmap:119 H264/90000\r\na=fmtp:119 profile-level-id=42e01f;packetization-mode=1\r\na=rtpmap:127 FLV/90000\r\na
    =sendonly\r\nm=audio 0 RTP/AVP 97 8 0\r\na=rtpmap:97 SPEEX/16000\r\na=rtpmap:8 PCMA/8000\r\na=rtpmap:0 PCMU/8000\r\na
    =sendonly\r\n",
12  "audioCodec": "mpeg4-generic",
13  "videoCodec": "H264",
14  "record": true,
15  "recordName": "stream-02d63bad-0a05-4ea6-b3be-c616cd44529d-b761as0rig0jvboenn0d0jjo.mp4",
16  "width": 1920,
17  "height": 1080,
18  "bitrate": 0,
19  "minBitrate": 0,
20  "maxBitrate": 0,
21  "quality": 0,
22  "history": false,
```