

API overview

- [API URLs](#)
- [Settings](#)
 - [Ports](#)
 - [Cross-domain requests](#)
 - [Authorization](#)
- [Testing](#)
- [OpenAPI specification](#)

API URLs

Web Call Server uses two URLs for REST API

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

To invoke the REST API, you have to make request using the following format:

- Protocol: HTTP or HTTPS
- Content Type: application/json
- Method: POST
- Body: JSON

Settings

Ports

By default, server accepts REST HTTP requests on 8081 TCP port and REST HTTPS requests on 8444 TCP port. Those values are defined with the following settings in [flashphoner.properties](#) file

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

Cross-domain requests

Cross-domain request are managed with the following settings in [flashphoner.properties](#) file

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

By default, cross-domain requests from any address are allowed, to make port redirection work.

Authorization

REST API requests are handled without user authorization by default. HTTP basic authorization may be enabled with the following parameter in [flashphoner.properties](#) file if necessary

```
disable_rest_auth=false
```

When authorization is enabled, REST API users are managed from [WCS core CLI](#).

For cross-domain requests, custom headers have to be added using the following parameter in [flashphoner.properties](#) file:

```
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
```

Here <origin> is the request source, e.g. <http://test.flashphoner.com>.

Testing

For testing purpose you can use a REST client. I.e. [Advanced REST Console](#) for browser Google Chrome.


On the screenshot below you can see how to get list of streams by REST API using REST method/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. Below the URL bar, there are tabs for HEADERS, BODY, AUTHORIZATION, and VARIABLES. The HEADERS tab is active, showing a single header: Content-Type: application/json. Below the headers, there is a section for the Response, which shows a 200 OK status with a response size of 1008 B and a response time of 355 ms. The response body is a JSON object, which is displayed in a code editor with line numbers. The JSON object contains the following fields: appKey, sessionId, mediaSessionId, name, published, hasVideo, hasAudio, status, sdp, audioCodec, videoCodec, record, recordName, width, height, bitrate, minBitrate, maxBitrate, quality, and history.

```
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\nnc=IN IP4 0.0.0.0\r\nnt=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-b761as0rig0jjvboenn0d0jjo.mp4",
16  "width": 1920,
17  "height": 1080,
18  "bitrate": 0,
19  "minBitrate": 0,
20  "maxBitrate": 0,
21  "quality": 0,
22  "history": false,
```

OpenAPI specification

Since build [5.2.1343](#) REST API can be displayed and tested according to OpenAPI 3 specification. OpenAPI specification file can be downloaded by URL https://demo.flashphoner.com:8444/rest-api/openapi_spec in YAML format, UI is available by URL <https://demo.flashphoner.com:8444/rest-api-spec/>, where `demo.flashphoner.com` is WCS server address

demo.flashphoner.com:8444/rest-api-spec/  /rest-api/openapi_spec Explore

WCS REST API 2.0.0 OAS3

[/rest-api/openapi_spec](#)

This is the REST API v 2.0 for Web Call Server 5.2.x

[Terms of service](#)
[Contact the developer](#)
[Flashphoner License](#)

Servers

Streaming

- POST** /rest-api/stream/find Find streams
- GET** /rest-api/stream/find_all Find all streams
- POST** /rest-api/stream/event/send Send data to all the stream subscribers
- POST** /rest-api/stream/snapshot Make Stream snapshot
- POST** /rest-api/stream/startRecording Start stream recording
- POST** /rest-api/stream/stopRecording Stop stream recording
- POST** /rest-api/stream/terminate Terminate stream

Every REST API query can be explored and tested using Swagger UI

Streaming

POST /rest-api/stream/find Find streams

Find streams by filter

Parameters Try it out

No parameters

Request body required application/json

Example Value | **Schema**

```
{  "custom": {},  "nodeId": "string",  "appKey": "defaultApp",  "sessionId": "/192.168.1.102:13252/192.168.1.101:8443",  "mediaSessionId": "4f112b20-13d0-11e7-b521-59a9cb7eddeb",  "name": "stream1",  "published": true,  "status": "NEW",  "display": {    "metrics"  },  "customMap": {}}
```

Required fields are marked by * character on Schema tab.

Streaming

POST /rest-api/stream/find Find streams

Find streams by filter

Parameters

Try it out

No parameters

Request body required

application/json

Example Value | Schema

```
StreamFilter {
  custom > {...}
  nodeId > {...}
  appKey string
  sessionId string
  mediaSessionId* string
  name* string
  published example: streamI
  status boolean
  Enum:
    > Array [ 11 ]
  display > [...]
  customMap > {...}
}
```