

Redirecting a stream to a SIP call using /call/inject_stream

- [Redirecting a stream published on server](#)
- [Injecting stream from file](#)
- [Stopping stream injection](#)

Redirecting a stream published on server

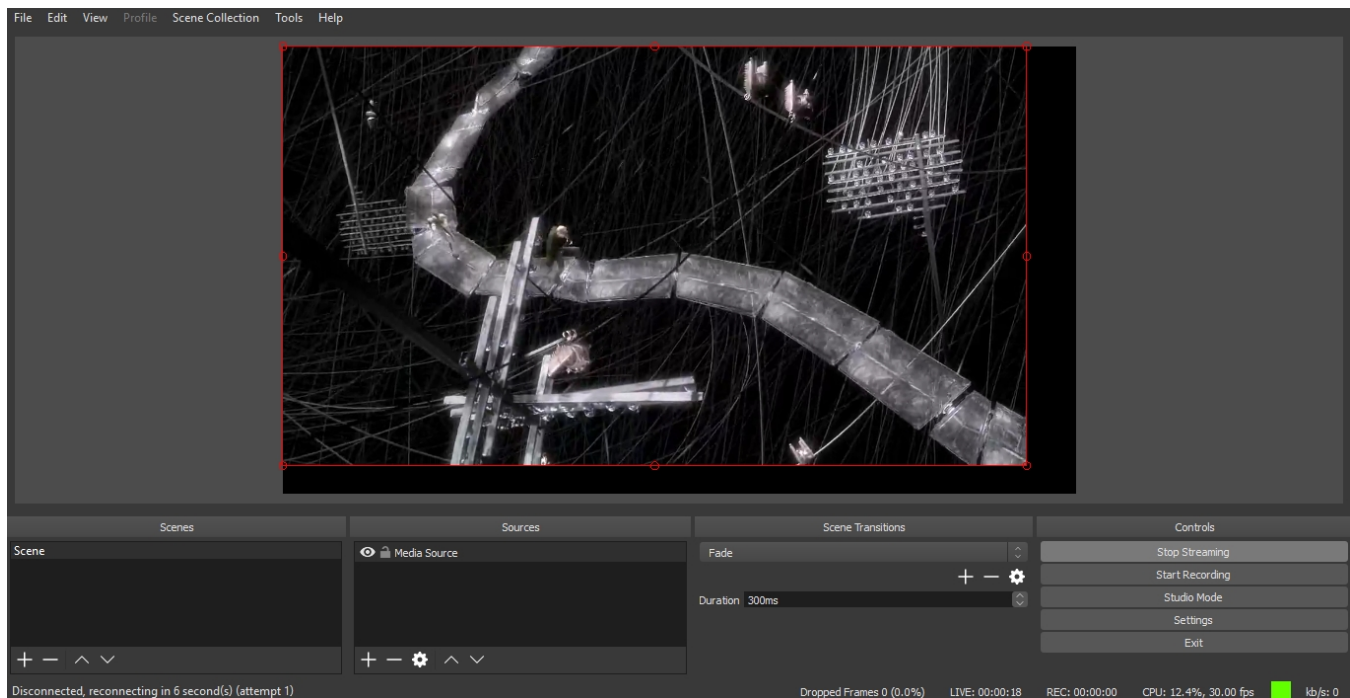
1. For the test we use:

- two SIP accounts;
- a softphone to answer to the call;
- the [REST client](#) of the Chrome browser;
- the program to broadcast an RTMP stream, OBS Studio.

2. Run the RTMP stream broadcast program. Configure and start broadcasting to your WCS server using the following parameters:

URL of the server: <http://demo.flashphoner.com:1935/live/>

Stream key: obsStream



3. Open the REST client. Send the /call/startup query to the WCS server, and in the query parameters specify the following:

- parameters of your SIP account the call is made from;
- the name of your second SIP account the call is made to.

Method: POST Request URL: http://p12.flashphoner.com:9091/rest-api/call/startup

Parameters

Body content type: application/json Editor view: Raw input

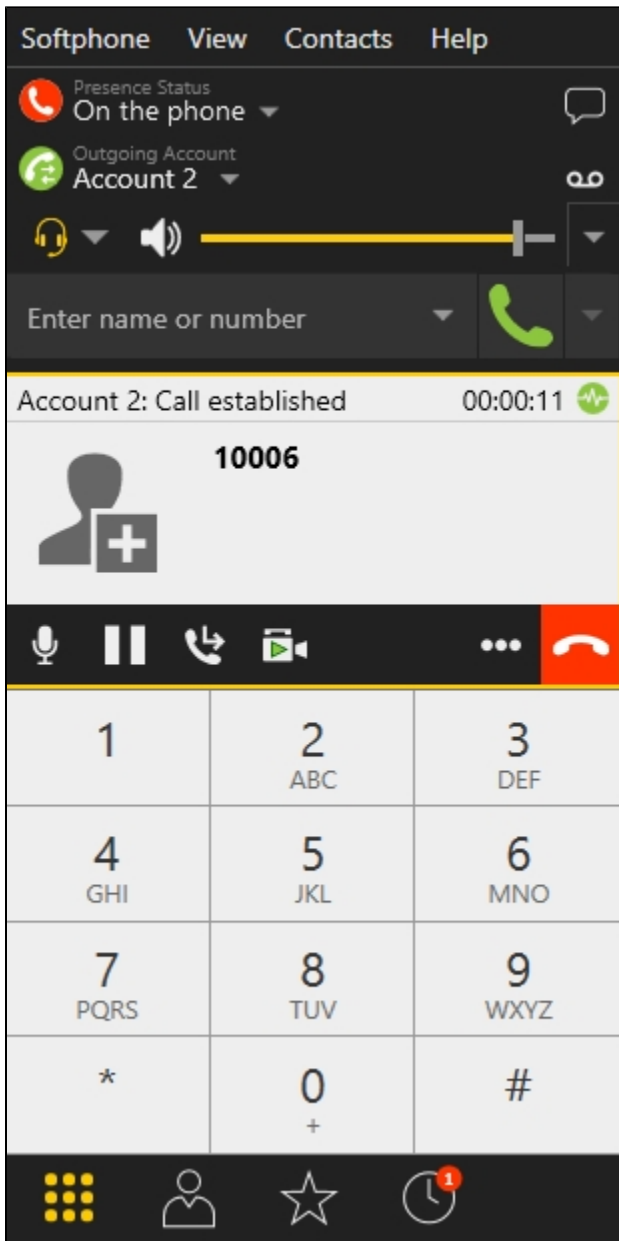
FORMAT JSON MINIFY JSON

```
{
  "callId": "12345678913",
  "callee": "10005",
  "toStream": "call_stream1",
  "hasAudio": "true",
  "hasVideo": "true",
  "sipLogin": "10006",
  "sipAuthenticationName": "10006",
  "sipPassword": "*****",
  "sipDomain": "domain.net",
  "sipOutboundProxy": "domain.net",
  "sipPort": "5060",
  "appKey": "defaultApp",
  "sipRegisterRequired": "true"
}
```

200 OK 28.50 ms

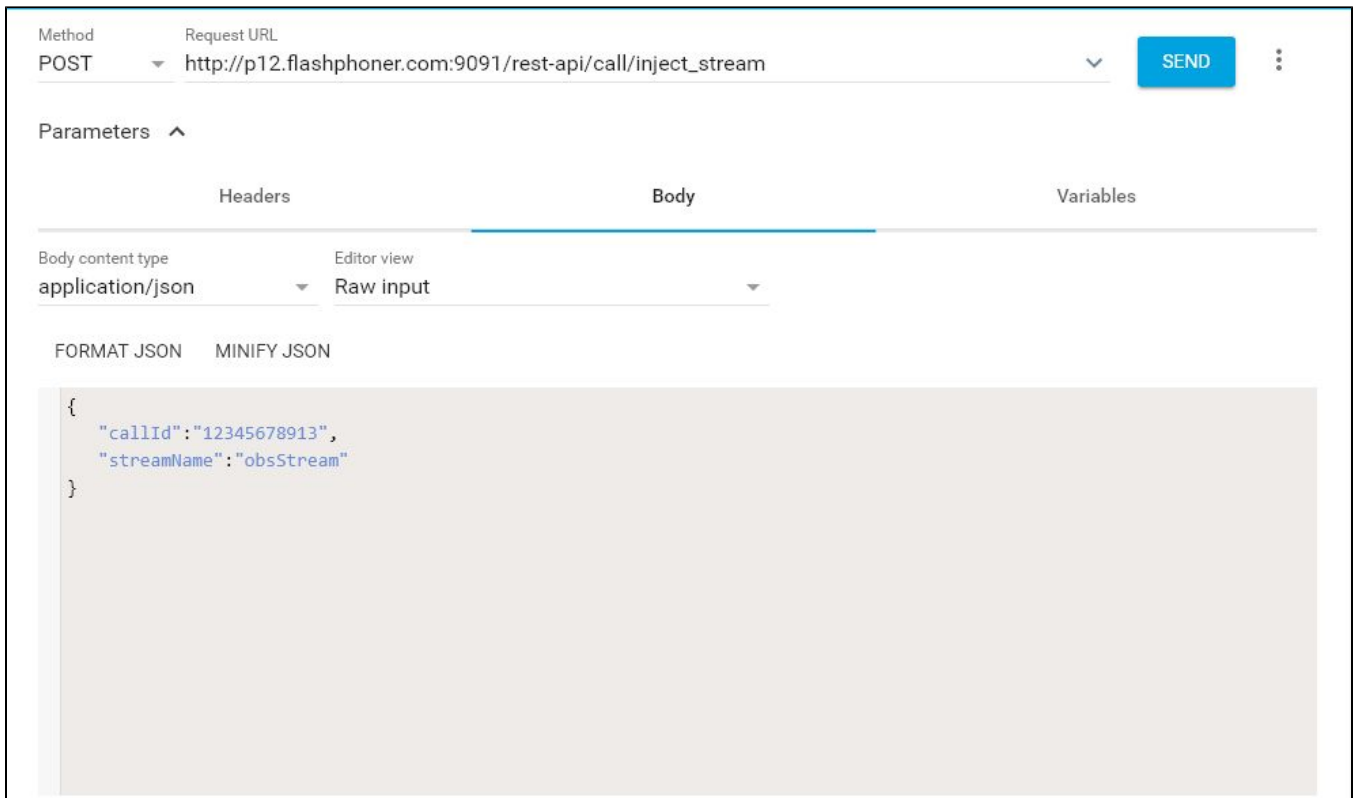
DETAILS

4. Receive the call in the softphone:



5. From the REST client send the `/call/inject_stream` query to the WCS server and specify the following parameters:

- the identifier of the call;
- the name of the broadcast stream (obsStream).



Make sure the audio track of the broadcast can be heard in the softphone.

6. To terminate the call, click the corresponding button in the softphone.

Injecting stream from file

VOD stream from file can be injected to SIP call. If there is video in file, and hasVideo parameter set for the call, video also will be injected. To do this, file name should be set as stream name:

```
{
  "callId": "123456711",
  "streamName": "vod://file.mp4"
}
```

Stopping stream injection

To stop stream injection to SIP call, send the followin REST query

```
/call/inject_stream/terminate
{
  "callId": "123456711"
}
```