

Raw WebSocket API

- Overview
- Command set for stream publishing and playback
 - connect
 - publish
 - unpublish
 - play
 - stop
- Publishing/playback stream metrics
- Stream availability checking

Overview

All WCS SDK: WebSDK, AndroidSDK, iOS SDK use WebSocket transport protocol to communicate with server. To exchange SDP between client and server and establish WebRTC mediasession, the custom command set is implemented over WebSocket.

WebSocket messages exchange can be seen while WebSDK example application works in Chrome browser, [Two Way Streaming](#) for example. To do this open Developer tools by pressing F12 key, go to Network section, choose host name and go to Frames tab

The screenshot shows a web browser window with the URL `https://demo.flashphoner.com/client2/examples/demo/streaming/two_way_streaming/two_way_streaming.html`. The application interface is titled "Two-way Streaming" and features a "Local" video player showing a pinkish-red gradient. Below it is a "PUBLISHING" section with a "Player" video player showing a grey screen. At the bottom, there is a status bar indicating "wss://demo.flashphoner.com:8443" and "ESTABLISHED".

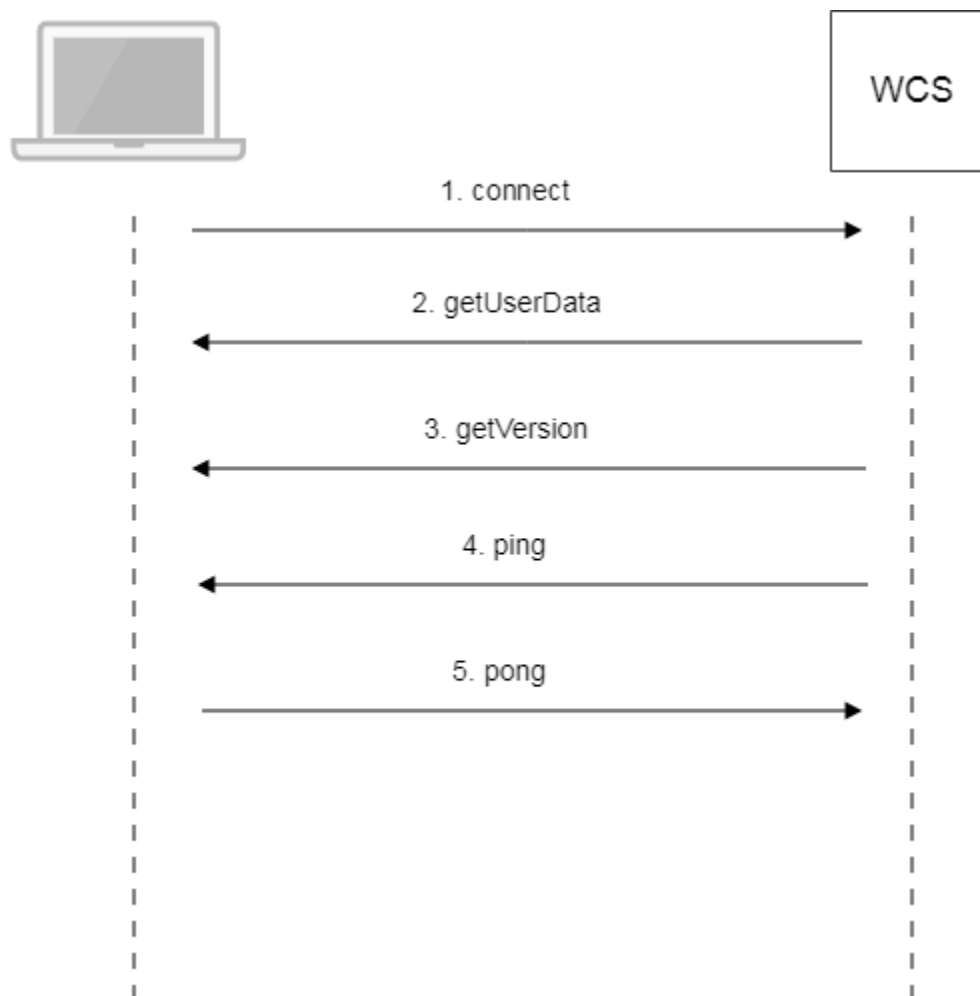
The browser's developer tools are open, showing the Network tab. The filter is set to "All". The selected resource is `demo.flashphoner.com`. The "Frames" sub-tab is active, showing a list of frames. The first frame is selected, and its "Data" tab is open, displaying a JSON message:

```
{
  "message": "connection",
  "data": {
    "appKey": "defaultApp",
    "mediaProviders": {
      "WebRTC": {
        "MSE": {
          "WSPlayer": {
            "clientVersion": "0.5.28",
            "clientOSVer": "0.5.28"
          }
        }
      }
    }
  }
}
```

Command set for stream publishing and playback

connect

When establishing connection with server and creating a session, messages exchange flow as follows:



1. Client sends connection message

```
{
  "message": "connection",
  "data": [
    {
      "appKey": "defaultApp",
      "mediaProviders": ["WebRTC", "MSE", "WSPlayer"],
      "clientVersion": "0.5.28",
      "clientOSVersion": "5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.80 Safari/537.36",
      "clientBrowserVersion": "Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.80 Safari/537.36"
    }
  ]
}
```

Message parameters

Parameter	Description
appKey	Server application key to communicate with backend server while session goes
mediaProviders	List of supported publication or playback technologies
clientVersion	Client SDK version

clientOSVersion	Client OS version
clientBrowserVersion	Client browser version

2. Server sends getUserData message

```
{
  "message": "getUserData",
  "data": [
    {
      "useWsTunnel": false,
      "useWsTunnelPacketization2": false,
      "useBase64BinaryEncoding": false,
      "mediaProviders": ["WebRTC", "MSE", "WSPayer"],
      "authToken": "/5.129.23.83:50066/188.40.244.249:8443",
      "status": "ESTABLISHED",
      "clientVersion": "0.5.28",
      "clientOSVersion": "5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.80 Safari/537.36",
      "clientBrowserVersion": "Mozilla/5.0 (Windows NT 6.1; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/71.0.3578.80 Safari/537.36",
      "forceLogger": false,
      "keepAlive": false
    }
  ]
}
```

This message confirms successful WebSocket connection establishing and data receiving from client

Message parameters

Parameter	Description
useWSTunnel	WS tunnel is used
useWsTunnelPacketization2	WS tunnel packetization is used
useBase64BinaryEncoding	Base64 encoding is used
mediaProviders	List of supported publication or playback technologies
authToken	Authentication token, the same as session Id formed with client and server IP address and port
status	WebSocket session status
clientVersion	Client SDK version
clientOSVersion	Client OS version
clientBrowserVersion	Client browser version
forceLogger	Logger is forced
keepAlive	Keep Alive is used

3. Server sends getVersion message

```
{
  "message": "getVersion",
  "data": [
    "5.1.3674-9b5b684ac81a1c7e62d7a38e0c3a2c2433faacf9"
  ]
}
```

Message contains WCS version

4. If there are no other messages received from client, server periodically sends ping

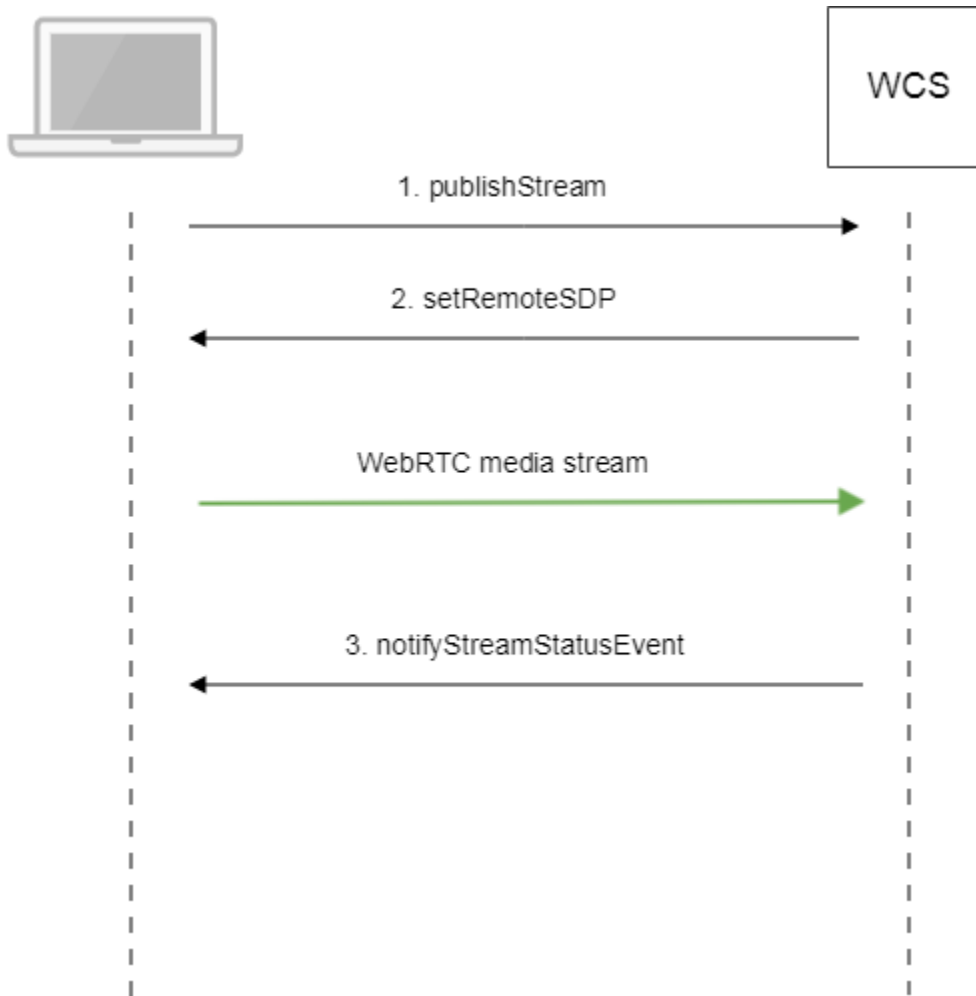
```
{
  "message": "ping",
  "data": []
}
```

on which client must respond withpong

```
{
  "message": "pong",
  "data": [null]
}
```

publish

When publishing a stream, messages exchange flow as follows:



1. Client sendspublishStream message

```

{
  "message": "publishStream",
  "data":
  [
    {
      "mediaSessionId": "90e14ad0-fdb2-11e8-bce2-a3f72476fb49",
      "name": "test",
      "published": true,
      "hasVideo": true,
      "hasAudio": true,
      "status": "PENDING",
      "record": false,
      "mediaProvider": "WebRTC",
      "sdp": "v=0\r\no=- 1712288193115840199 2 IN IP4 127.0.0.1\r\ns=-\r\nnt=0 0\r\na=group:BUNDLE audio
video\r\na=msid-semantic: WMS EZ5Uy6UkPtEoScWkyAcP5rabBY6buERZjQpH\r\nm=audio 9 UDP/TLS/RTP/SAVPF 111 103 104 9
0 8 106 105 13 110 112 113 126\r\nnc=IN IP4 0.0.0.0\r\na=rtpcp:9 IN IP4 0.0.0.0\r\na=ice-ufraq:qg0t\r\na=ice-pwd:
KEfomSLSL4Pxj78N6T3/HAMY\r\na=ice-options:trickle\r\na=fingerprint:sha-256 EE:BC:55:18:45:6C:B4:5C:28:6F:5D:F9:
42:63:1B:A2:99:8D:7A:0B:19:C3:F5:62:12:63:82:B6:F0:2C:36:E2\r\na=setup:actpass\r\na=mid:audio\r\na=extmap:1 urn:
ietf:params:rtp-hdrext:ssrc-audio-level\r\na=sendonly\r\na=rtpcp-mux\r\na=rtpmap:111 opus/48000/2\r\na=rtpcp-fb:
111 transport-cc\r\na=fmtp:111 minptime=10;useinbandfec=1\r\na=rtpmap:103 ISAC/16000\r\na=rtpmap:104 ISAC
/32000\r\na=rtpmap:9 G722/8000\r\na=rtpmap:0 PCMU/8000\r\na=rtpmap:8 PCMA/8000\r\na=rtpmap:106 CN
/32000\r\na=rtpmap:105 CN/16000\r\na=rtpmap:13 CN/8000\r\na=rtpmap:110 telephone-event/48000\r\na=rtpmap:112
telephone-event/32000\r\na=rtpmap:113 telephone-event/16000\r\na=rtpmap:126 telephone-event/8000\r\na=ssrc:
3498121776 cname:Ft3gt16szZ3kAllo\r\na=ssrc:3498121776 msid:EZ5Uy6UkPtEoScWkyAcP5rabBY6buERZjQpH 55944815-d49b-
4ab2-bc19-3ecdca00b4a2\r\na=ssrc:3498121776 mslabel:EZ5Uy6UkPtEoScWkyAcP5rabBY6buERZjQpH\r\na=ssrc:3498121776
label:55944815-d49b-4ab2-bc19-3ecdca00b4a2\r\nm=video 9 UDP/TLS/RTP/SAVPF 96 97 98 99 100 101 102 123 127 122
125 107 108 109 124\r\nnc=IN IP4 0.0.0.0\r\na=rtpcp:9 IN IP4 0.0.0.0\r\na=ice-ufraq:qg0t\r\na=ice-pwd:
KEfomSLSL4Pxj78N6T3/HAMY\r\na=ice-options:trickle\r\na=fingerprint:sha-256 EE:BC:55:18:45:6C:B4:5C:28:6F:5D:F9:
42:63:1B:A2:99:8D:7A:0B:19:C3:F5:62:12:63:82:B6:F0:2C:36:E2\r\na=setup:actpass\r\na=mid:video\r\na=extmap:2 urn:
ietf:params:rtp-hdrext:toffset\r\na=extmap:3 http://www.webrtc.org/experiments/rtp-hdrext/abs-send-
time\r\na=extmap:4 urn:3gpp:video-orientation\r\na=extmap:5 http://www.ietf.org/draft-holmer-rmcat-transport-
wide-cc-extensions-01\r\na=extmap:6 http://www.webrtc.org/experiments/rtp-hdrext/playout-delay\r\na=extmap:7
http://www.webrtc.org/experiments/rtp-hdrext/video-content-type\r\na=extmap:8 http://www.webrtc.org/experiments
/rtp-hdrext/video-timing\r\na=extmap:10 http://tools.ietf.org/html/draft-ietf-avtext-framemarking-
07\r\na=sendonly\r\na=rtpcp-mux\r\na=rtpcp-rsize\r\na=rtpmap:96 VP8/90000\r\na=rtpcp-fb:96 goog-remb\r\na=rtpcp-fb:
96 transport-cc\r\na=rtpcp-fb:96 ccm fir\r\na=rtpcp-fb:96 nack\r\na=rtpcp-fb:96 nack pli\r\na=rtpmap:97 rtx
/90000\r\na=fmtp:97 apt=96\r\na=rtpmap:98 VP9/90000\r\na=rtpcp-fb:98 goog-remb\r\na=rtpcp-fb:98 transport-
cc\r\na=rtpcp-fb:98 ccm fir\r\na=rtpcp-fb:98 nack\r\na=rtpcp-fb:98 nack pli\r\na=fmtp:98 profile-id=0\r\na=rtpmap:
99 rtx/90000\r\na=fmtp:99 apt=98\r\na=rtpmap:100 H264/90000\r\na=rtpcp-fb:100 goog-remb\r\na=rtpcp-fb:100
transport-cc\r\na=rtpcp-fb:100 ccm fir\r\na=rtpcp-fb:100 nack\r\na=rtpcp-fb:100 nack pli\r\na=fmtp:100 level-
asymmetry-allowed=1;packetization-mode=1;profile-level-id=42001f\r\na=rtpmap:101 rtx/90000\r\na=fmtp:101
apt=100\r\na=rtpmap:102 H264/90000\r\na=rtpcp-fb:102 goog-remb\r\na=rtpcp-fb:102 transport-cc\r\na=rtpcp-fb:102
ccm fir\r\na=rtpcp-fb:102 nack\r\na=rtpcp-fb:102 nack pli\r\na=fmtp:102 level-asymmetry-allowed=1;packetization-
mode=0;profile-level-id=42001f\r\na=rtpmap:123 rtx/90000\r\na=fmtp:123 apt=102\r\na=rtpmap:127 H264
/90000\r\na=rtpcp-fb:127 goog-remb\r\na=rtpcp-fb:127 transport-cc\r\na=rtpcp-fb:127 ccm fir\r\na=rtpcp-fb:127
nack\r\na=rtpcp-fb:127 nack pli\r\na=fmtp:127 level-asymmetry-allowed=1;packetization-mode=1;profile-level-
id=42e01f\r\na=rtpmap:122 rtx/90000\r\na=fmtp:122 apt=127\r\na=rtpmap:125 H264/90000\r\na=rtpcp-fb:125 goog-
remb\r\na=rtpcp-fb:125 transport-cc\r\na=rtpcp-fb:125 ccm fir\r\na=rtpcp-fb:125 nack\r\na=rtpcp-fb:125 nack
pli\r\na=fmtp:125 level-asymmetry-allowed=1;packetization-mode=0;profile-level-id=42e01f\r\na=rtpmap:107 rtx
/90000\r\na=fmtp:107 apt=125\r\na=rtpmap:108 red/90000\r\na=rtpmap:109 rtx/90000\r\na=fmtp:109
apt=108\r\na=rtpmap:124 ulpfec/90000\r\na=ssrc-group:FID 3064307282 2143201641\r\na=ssrc:3064307282 cname:
Ft3gt16szZ3kAllo\r\na=ssrc:3064307282 msid:EZ5Uy6UkPtEoScWkyAcP5rabBY6buERZjQpH 77c7bd16-9a65-4f62-9d01-
6ecdd3e58f8a\r\na=ssrc:3064307282 mslabel:EZ5Uy6UkPtEoScWkyAcP5rabBY6buERZjQpH\r\na=ssrc:3064307282 label:
77c7bd16-9a65-4f62-9d01-6ecdd3e58f8a\r\na=ssrc:2143201641 cname:Ft3gt16szZ3kAllo\r\na=ssrc:2143201641 msid:
EZ5Uy6UkPtEoScWkyAcP5rabBY6buERZjQpH 77c7bd16-9a65-4f62-9d01-6ecdd3e58f8a\r\na=ssrc:2143201641 mslabel:
EZ5Uy6UkPtEoScWkyAcP5rabBY6buERZjQpH\r\na=ssrc:2143201641 label:77c7bd16-9a65-4f62-9d01-6ecdd3e58f8a\r\n",
      "bitrate": 0,
      "minBitrate": 0,
      "maxBitrate": 0
    }
  ]
}

```

Message parameters

Parameter	Description
mediaSessionId	WebRTC media session Id

name	Published stream name
hasVideo	Stream has video
hasAudio	Stream has audio
status	Stream status
record	Stream should be recorded on server
mediaProvider	List of supoted publication or playback technologies
sdp	Client SDP
bitrate	Bitrate (will set while publishing if 0)
minBitrate	Minimum bitrate (no limits if 0)
maxBitrate	Maximum bitrate (no limits if 0)

2. Server sends setRemoteSDP message

```
{
  "message": "setRemoteSDP",
  "data": [
    "90e14ad0-fdb2-11e8-bce2-a3f72476fb49",
    "v=0\r\no=Flashphoner 0 1544579813187 IN IP4 188.40.244.249\r\ns=Flashphoner/1.0\r\nnc=IN IP4 188.40.244.249\r\nnt=0 0\r\nnm=audio 31686 RTP/SAVPF 111 8 9\r\nnc=IN IP4 188.40.244.249\r\nna=mid: audio\r\nna=rtpmap:111 opus/48000/2\r\nna=rtpmap:8 PCMA/8000\r\nna=rtpmap:9 G722/8000\r\nna=ptime:20\r\nna=ice-pwd: 40u9aoieitv0u62tol3pr1g3h4\r\nna=ice-ufrag:90e14ad0-fdb2-11e8-bce2-a3f72476fb49cfmkh1cug233q2\r\nna=fingerprint: SHA-256 33:7A:40:CF:1C:42:03:20:EE:E9:F9:95:24:2A:95:85:D6:25:A9:68:C6:CF:40:09:82:74:B4:C3:F9:97:E6:EE\r\nna=candidate:1 1 udp 2130706431 188.40.244.249 31686 typ host\r\nna=candidate:1 2 udp 2130706431 188.40.244.249 31686 typ host\r\nna=end-of-candidates\r\nna=rtcp-mux\r\nna=rtcp:31686 IN IP4 188.40.244.249\r\nna=recvonly\r\nnm=video 31920 RTP/SAVPF 100 127 102 125 96\r\nnc=IN IP4 188.40.244.249\r\nna=mid: video\r\nna=rtpmap:100 H264/90000\r\nna=fmtp:100 level-asymmetry-allowed=1;packetization-mode=1;profile-level-id=42001f\r\nna=rtpmap:127 H264/90000\r\nna=fmtp:127 level-asymmetry-allowed=1;packetization-mode=1;profile-level-id=42e01f\r\nna=rtpmap:102 H264/90000\r\nna=fmtp:102 level-asymmetry-allowed=1;packetization-mode=0;profile-level-id=42001f\r\nna=rtpmap:125 H264/90000\r\nna=fmtp:125 level-asymmetry-allowed=1;packetization-mode=0;profile-level-id=42e01f\r\nna=rtpmap:96 VP8/90000\r\nna=rtcp-fb:* ccm fir\r\nna=rtcp-fb:* nack\r\nna=rtcp-fb:* nack pli\r\nna=rtcp-fb:* goog-remb\r\nna=ice-pwd:40u9aoieitv0u62tol3pr1g3h4\r\nna=ice-ufrag:90e14ad0-fdb2-11e8-bce2-a3f72476fb49cfmkh1cug233q2\r\nna=fingerprint:SHA-256 33:7A:40:CF:1C:42:03:20:EE:E9:F9:95:24:2A:95:85:D6:25:A9:68:C6:CF:40:09:82:74:B4:C3:F9:97:E6:EE\r\nna=candidate:1 1 udp 2130706431 188.40.244.249 31920 typ host\r\nna=candidate:1 2 udp 2130706431 188.40.244.249 31920 typ host\r\nna=end-of-candidates\r\nna=rtcp-mux\r\nna=rtcp:31920 IN IP4 188.40.244.249\r\nna=recvonly\r\n",
    true
  ]
}
```

Message contains:

- Media session Id
- Server SDP

3. When stream is published, server sends notifyStreamStatusEvent message

```

{
  "message": "notifyStreamStatusEvent",
  "data":
  [
    {
      "mediaSessionId": "90e14ad0-fdb2-11e8-bce2-a3f72476fb49",
      "name": "test",
      "published": true,
      "hasVideo": true,
      "hasAudio": true,
      "status": "PUBLISHING",
      "audioCodec": "opus",
      "videoCodec": "H264",
      "record": false,
      "width": 0,
      "height": 0,
      "bitrate": 0,
      "minBitrate": 0,
      "maxBitrate": 0,
      "quality": 0,
      "createDate": 1544579813186,
      "streamInfo":
      {
        "nodeId": null,
        "appKey": null,
        "sessionId": null,
        "mediaSessionId": "90e14ad0-fdb2-11e8-bce2-a3f72476fb49",
        "name": "test",
        "samplingTime": null,
        "recordTimestamp": null,
        "recordStarted": false
      },
      "mediaProvider": "WebRTC"
    }
  ]
}

```

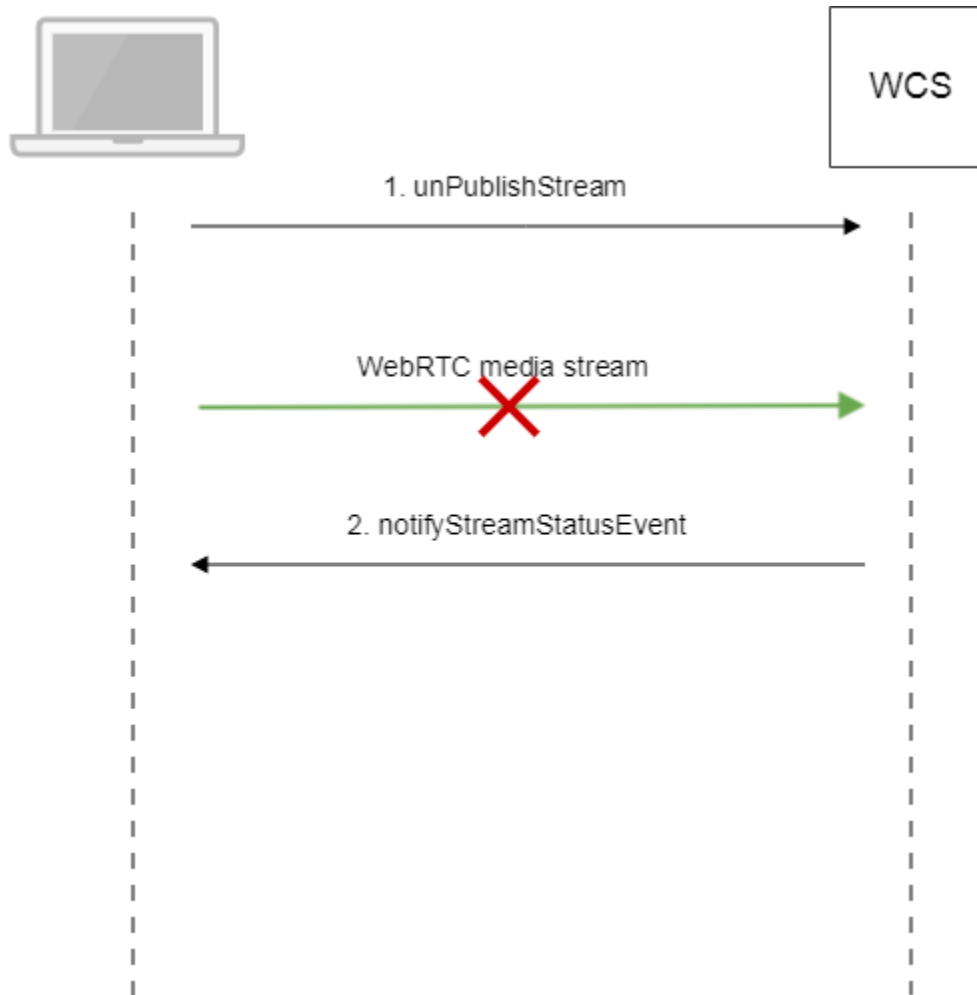
Message parameters

Parameter	Description
mediaSessionId	WebRTC media session Id
name	Published stream name
published	Stream is published
hasVideo	Stream has video
hasAudio	Stream has audio
status	Stream status
audioCodec	Audio codec
videoCodec	Video codec
record	Stream is recorded on server
width	Picture width (set while publishing if 0)
height	Picture height (set while publishing if 0)
bitrate	Bitrate (will set while publishing if 0)
minBitrate	Minimum bitrate (no limits if 0)
maxBitrate	Maximum bitrate (no limits if 0)
quality	Stream quality
createDate	Creation date as integer

streamInfo	Additional stream information
mediaProvider	Publishing technology

unpublish

When stopping a stream publishing, messages exchange flow as follows:



1. Client sends unPublishStream message

```
{
  "message": "unPublishStream",
  "data": [
    {
      "mediaSessionId": "1d91ae20-fdb3-11e8-bddf-9b115426f01b",
      "name": "test",
      "published": true,
      "hasVideo": true,
      "hasAudio": true,
      "status": "PUBLISHING",
      "record": false
    }
  ]
}
```

Message parameters

Parameter	Description
mediaSessionId	WebRTC media session Id
name	Published stream name
published	Stream is published
hasVideo	Stream has video
hasAudio	Stream has audio
status	Stream status
record	Stream is recorded on server

2. Server stops publishing and sends `notifyStreamStatusEvent` message

```
{
  "message": "notifyStreamStatusEvent",
  "data": [
    {
      "mediaSessionId": "1d91ae20-fdb3-11e8-bddf-9b115426f01b",
      "name": "test",
      "published": true,
      "hasVideo": true,
      "hasAudio": true,
      "status": "UNPUBLISHED",
      "audioCodec": "opus",
      "videoCodec": "H264",
      "info": "Stopped by publisher stop",
      "record": false,
      "width": 320,
      "height": 240,
      "bitrate": 0,
      "minBitrate": 0,
      "maxBitrate": 0,
      "quality": 0,
      "createDate": 1544580045898,
      "streamInfo": {
        "nodeId": null,
        "appKey": null,
        "sessionId": null,
        "mediaSessionId": "1d91ae20-fdb3-11e8-bddf-9b115426f01b",
        "name": "test",
        "samplingTime": null,
        "recordTimestamp": null,
        "recordStarted": false
      },
      "mediaProvider": "WebRTC"
    }
  ]
}
```

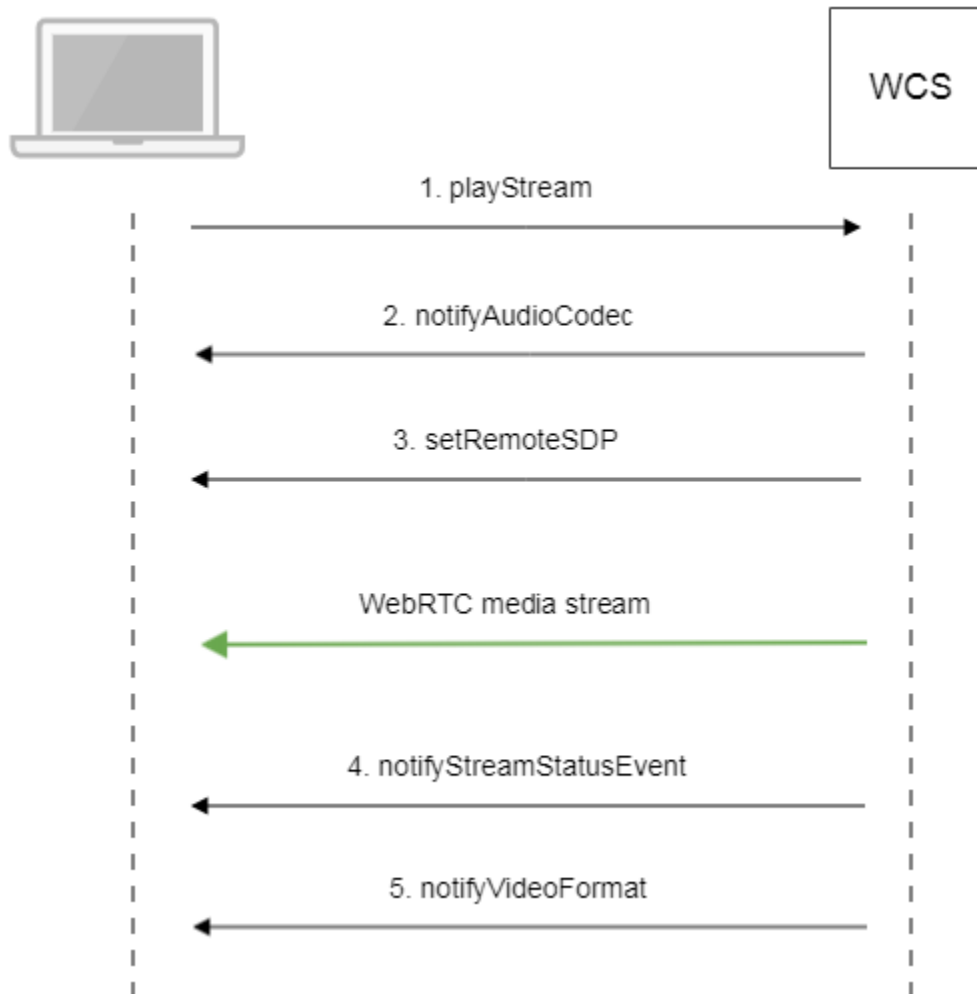
Message parameters

Parameter	Description
mediaSessionId	WebRTC media session Id
name	Published stream name
published	Stream is published
hasVideo	Stream has video
hasAudio	Stream has audio
status	Stream status

audioCodec	Audio codec
videoCodec	Video codec
record	Stream is recorded on server
width	Picture width (set while publishing if 0)
height	Picture height (set while publishing if 0)
bitrate	Bitrate (will set while publishing if 0)
minBitrate	Minimum bitrate (no limits if 0)
maxBitrate	Maximum bitrate (no limits if 0)
quality	Stream quality
createDate	Creation date as integer
streamInfo	Additional stream information
mediaProvider	Publishing technology

play

When playing a stream, messages exchange flow as follows:



1. Client sends `playStream` message

```

{
  "message": "playStream",
  "data": [
    {
      "mediaSessionId": "22588500-fdb3-11e8-bddf-9b115426f01b",
      "name": "test",
      "published": false,
      "hasVideo": true,
      "hasAudio": true,
      "status": "PENDING",
      "record": false,
      "width": 0,
      "height": 0,
      "mediaProvider": "WebRTC",
      "sdp": "v=0\r\no=- 6636414462219466101 2 IN IP4 127.0.0.1\r\ns=-\r\nnt=0 0\r\na=group:BUNDLE audio
video\r\na=msid-semantic: WMS\r\nm=audio 9 UDP/TLS/RTP/SAVPF 111 103 104 9 0 8 106 105 13 110 112 113
126\r\nnc=IN IP4 0.0.0.0\r\na=rtcp:9 IN IP4 0.0.0.0\r\na=ice-ufrag:33IG\r\na=ice-pwd:
R78vJJJetyP2WewpU6QWKGbJA\r\na=ice-options:trickle\r\na=fingerprint:sha-256 72:67:0A:92:8B:C2:CB:13:4E:7F:66:BB:
AF:99:4A:0C:D9:C3:C1:3E:B9:B8:24:EB:2E:64:39:B2:19:76:85:09\r\na=setup:actpass\r\na=mid:audio\r\na=extmap:1 urn:
ietf:params:rtp-hdext:ssrc-audio-level\r\na=recvonly\r\na=rtcp-mux\r\na=rtpmap:111 opus/48000/2\r\na=rtcp-fb:
111 transport-cc\r\na=fmtp:111 minptime=10;useinbandfec=1\r\na=rtpmap:103 ISAC/16000\r\na=rtpmap:104 ISAC
/32000\r\na=rtpmap:9 G722/8000\r\na=rtpmap:0 PCMU/8000\r\na=rtpmap:8 PCMA/8000\r\na=rtpmap:106 CN
/32000\r\na=rtpmap:105 CN/16000\r\na=rtpmap:13 CN/8000\r\na=rtpmap:110 telephone-event/48000\r\na=rtpmap:112
telephone-event/32000\r\na=rtpmap:113 telephone-event/16000\r\na=rtpmap:126 telephone-event/8000\r\nm=video 9
UDP/TLS/RTP/SAVPF 96 97 98 99 100 101 102 123 127 122 125 107 108 109 124\r\nnc=IN IP4 0.0.0.0\r\na=rtcp:9 IN
IP4 0.0.0.0\r\na=ice-ufrag:33IG\r\na=ice-pwd:R78vJJJetyP2WewpU6QWKGbJA\r\na=ice-options:trickle\r\na=fingerprint:
sha-256 72:67:0A:92:8B:C2:CB:13:4E:7F:66:BB:AF:99:4A:0C:D9:C3:C1:3E:B9:B8:24:EB:2E:64:39:B2:19:76:85:
09\r\na=setup:actpass\r\na=mid:video\r\na=extmap:2 urn:ietf:params:rtp-hdext:toffset\r\na=extmap:3 http://www.
webrtc.org/experiments/rtp-hdext/abs-send-time\r\na=extmap:4 urn:3gpp:video-orientation\r\na=extmap:5
http://www.ietf.org/id/draft-holmer-rmcat-transport-wide-cc-extensions-01\r\na=extmap:6 http://www.webrtc.org
/experiments/rtp-hdext/playout-delay\r\na=extmap:7 http://www.webrtc.org/experiments/rtp-hdext/video-content-
type\r\na=extmap:8 http://www.webrtc.org/experiments/rtp-hdext/video-timing\r\na=extmap:10 http://tools.ietf.
org/html/draft-ietf-avtext-framemarking-07\r\na=recvonly\r\na=rtcp-mux\r\na=rtcp-rsize\r\na=rtpmap:96 VP8
/90000\r\na=rtcp-fb:96 goog-remb\r\na=rtcp-fb:96 transport-cc\r\na=rtcp-fb:96 ccm fir\r\na=rtcp-fb:96
nack\r\na=rtcp-fb:96 nack pli\r\na=rtpmap:97 rtx/90000\r\na=fmtp:97 apt=96\r\na=rtpmap:98 VP9/90000\r\na=rtcp-
fb:98 goog-remb\r\na=rtcp-fb:98 transport-cc\r\na=rtcp-fb:98 ccm fir\r\na=rtcp-fb:98 nack\r\na=rtcp-fb:98 nack
pli\r\na=fmtp:98 profile-id=0\r\na=rtpmap:99 rtx/90000\r\na=fmtp:99 apt=98\r\na=rtpmap:100 H264/90000\r\na=rtcp-
fb:100 goog-remb\r\na=rtcp-fb:100 transport-cc\r\na=rtcp-fb:100 ccm fir\r\na=rtcp-fb:100 nack\r\na=rtcp-fb:100
nack pli\r\na=fmtp:100 level-asymmetry-allowed=1;packetization-mode=1;profile-level-id=42001f\r\na=rtpmap:101
rtx/90000\r\na=fmtp:101 apt=100\r\na=rtpmap:102 H264/90000\r\na=rtcp-fb:102 goog-remb\r\na=rtcp-fb:102
transport-cc\r\na=rtcp-fb:102 ccm fir\r\na=rtcp-fb:102 nack\r\na=rtcp-fb:102 nack pli\r\na=fmtp:102 level-
asymmetry-allowed=1;packetization-mode=0;profile-level-id=42001f\r\na=rtpmap:123 rtx/90000\r\na=fmtp:123
apt=102\r\na=rtpmap:127 H264/90000\r\na=rtcp-fb:127 goog-remb\r\na=rtcp-fb:127 transport-cc\r\na=rtcp-fb:127
ccm fir\r\na=rtcp-fb:127 nack\r\na=rtcp-fb:127 nack pli\r\na=fmtp:127 level-asymmetry-allowed=1;packetization-
mode=1;profile-level-id=42e01f\r\na=rtpmap:122 rtx/90000\r\na=fmtp:122 apt=127\r\na=rtpmap:125 H264
/90000\r\na=rtcp-fb:125 goog-remb\r\na=rtcp-fb:125 transport-cc\r\na=rtcp-fb:125 ccm fir\r\na=rtcp-fb:125
nack\r\na=rtcp-fb:125 nack pli\r\na=fmtp:125 level-asymmetry-allowed=1;packetization-mode=0;profile-level-
id=42e01f\r\na=rtpmap:107 rtx/90000\r\na=fmtp:107 apt=125\r\na=rtpmap:108 red/90000\r\na=rtpmap:109 rtx
/90000\r\na=fmtp:109 apt=108\r\na=rtpmap:124 ulpfec/90000\r\n",
      "bitrate": 0,
      "minBitrate": 0,
      "maxBitrate": 0,
      "quality": 0
    }
  ]
}

```

Message parameters

Parameter	Description
mediaSessionId	WebRTC media session Id
name	Published stream name
published	Stream is published (ignored)
hasVideo	Stream has video

hasAudio	Stream has audio
status	Stream status
record	Stream is recorded on server (ignoring)
width	Picture width (play as published if 0)
height	Picture height (play as published if 0)
mediaProvider	Playback technology
sdp	Client SDP
bitrate	Bitrate (will set while playing if 0)
minBitrate	Minimum bitrate (no limits if 0)
maxBitrate	Maximum bitrate (no limits if 0)
quality	Stream quality (play as published if 0)

2. Server sends notifyAudioCodec message

```
{
  "message": "notifyAudioCodec",
  "data": [
    "22588500-fdb3-11e8-bddf-9b115426f01b",
    {
      "name": "opus"
    }
  ]
}
```

Message contains:

- WebRTC media session Id
- Stream audio codec name

3. Server sends setRemoteSDP message

```
{
  "message": "setRemoteSDP",
  "data": [
    "22588500-fdb3-11e8-bddf-9b115426f01b",
    "v=0\r\no=Flashphoner 0 1544580053583 IN IP4 188.40.244.249\r\nns=Flashphoner/1.0\r\nnc=IN IP4 188.40.244.249\r\nnt=0 0\r\nnm=audio 31354 RTP/SAVPF 111 8 9\r\nnc=IN IP4 188.40.244.249\r\nna=mid: audio\r\nna=rtpmap:111 opus/48000/2\r\nna=rtpmap:8 PCMA/8000\r\nna=rtpmap:9 G722/8000\r\nna=ptime:20\r\nna=ice-pwd: 1lbmj56an4qr2airc0adreiede\r\nna=ice-ufrag:22588500-fdb3-11e8-bddf-9b115426f01bv7m1cug2aei5\r\nna=fingerprint: SHA-256 33:7A:40:CF:1C:42:03:20:EE:E9:F9:95:24:2A:95:85:D6:25:A9:68:C6:CF:40:09:82:74:B4:C3:F9:97:E6: EE\r\nna=candidate:1 1 udp 2130706431 188.40.244.249 31354 typ host\r\nna=candidate:1 2 udp 2130706431 188.40.244.249 31354 typ host\r\nna=end-of-candidates\r\nna=rtcp-mux\r\nna=rtcp:31354 IN IP4 188.40.244.249 \r\nna=sendonly\r\nna=ssrc:1734923542 cname:rtp/audio/22588500-fdb3-11e8-bddf-9b115426f01b\r\nnm=video 31248 RTP /SAVPF 100 127 102 125 96\r\nnc=IN IP4 188.40.244.249\r\nna=mid:video\r\nna=rtpmap:100 H264/90000\r\nna=fmtp:100 level-asymmetry-allowed=1;packetization-mode=1;profile-level-id=42001f\r\nna=rtpmap:127 H264/90000\r\nna=fmtp:127 level-asymmetry-allowed=1;packetization-mode=1;profile-level-id=42e01f\r\nna=rtpmap:102 H264/90000\r\nna=fmtp:102 level-asymmetry-allowed=1;packetization-mode=0;profile-level-id=42001f\r\nna=rtpmap:125 H264/90000\r\nna=fmtp:125 level-asymmetry-allowed=1;packetization-mode=0;profile-level-id=42e01f\r\nna=rtpmap:96 VP8/90000\r\nna=rtcp-fb:* ccm fir\r\nna=rtcp-fb:* nack\r\nna=rtcp-fb:* nack pli\r\nna=rtcp-fb:* goog-remb\r\nna=ice-pwd: 1lbmj56an4qr2airc0adreiede\r\nna=ice-ufrag:22588500-fdb3-11e8-bddf-9b115426f01bv7m1cug2aei5\r\nna=fingerprint: SHA-256 33:7A:40:CF:1C:42:03:20:EE:E9:F9:95:24:2A:95:85:D6:25:A9:68:C6:CF:40:09:82:74:B4:C3:F9:97:E6: EE\r\nna=candidate:1 1 udp 2130706431 188.40.244.249 31248 typ host\r\nna=candidate:1 2 udp 2130706431 188.40.244.249 31248 typ host\r\nna=end-of-candidates\r\nna=rtcp-mux\r\nna=rtcp:31248 IN IP4 188.40.244.249 \r\nna=sendonly\r\nna=ssrc:1329233369 cname:rtp/video/22588500-fdb3-11e8-bddf-9b115426f01b\r\n",
    true
  ]
}
```

Message contains:

- Media session Id
- Server SDP

3. When media data transmission is started, server sends `notifyStreamStatusEvent` message

```
{
  "message": "notifyStreamStatusEvent",
  "data": [
    {
      "mediaSessionId": "22588500-fdb3-11e8-bddf-9b115426f01b",
      "name": "test",
      "published": false,
      "hasVideo": true,
      "hasAudio": true,
      "status": "PLAYING",
      "audioCodec": "opus",
      "videoCodec": "H264",
      "record": false,
      "width": 0,
      "height": 0,
      "bitrate": 0,
      "minBitrate": 0,
      "maxBitrate": 0,
      "quality": 0,
      "createDate": 1544580053573,
      "streamInfo": {
        "nodeId": null,
        "appKey": null,
        "sessionId": null,
        "mediaSessionId": "22588500-fdb3-11e8-bddf-9b115426f01b",
        "name": "test",
        "samplingTime": null,
        "recordTimestamp": null,
        "recordStarted": false
      },
      "mediaProvider": "WebRTC"
    }
  ]
}
```

Message parameters

Parameter	Description
mediaSessionId	WebRTC media session Id
name	Stream name to play
published	Stream is published
hasVideo	Stream has video
hasAudio	Stream has audio
status	Stream status
audioCodec	Audio codec
videoCodec	Video codec
record	Stream is recorded on server
width	Picture width (set while playing if 0)
height	Picture height (set while playing if 0)
bitrate	Bitrate (will set while playing if 0)
minBitrate	Minimum bitrate (no limits if 0)
maxBitrate	Maximum bitrate (no limits if 0)

quality	Stream quality
createDate	Creation date as integer
streamInfo	Additional stream information
mediaProvider	Publishing technology

4. Server sends `notifyVideoFormat` message

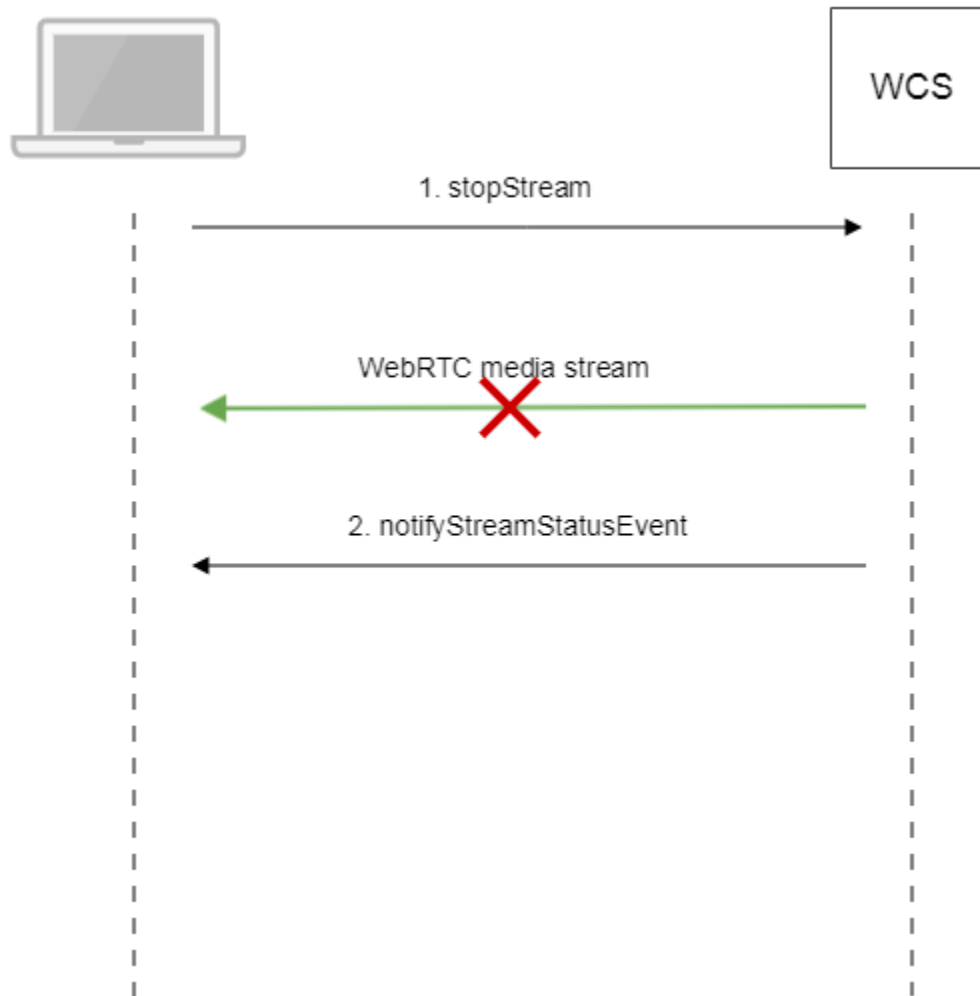
```
{
  "message": "notifyVideoFormat",
  "data": [
    {
      "streamerVideoWidth": 320,
      "streamerVideoHeight": 240,
      "mediaSessionId": "22588500-fdb3-11e8-bddf-9b115426f01b",
      "status": "RESIZE"
    }
  ]
}
```

Message parameters

Parameter	Description
streamerVideoWidth	Streamer picture width
streamerVideoHeight	Streamer picture height
mediaSessionId	WebRTC media session Id
status	Stream video status

stop

When stopping a stream playback, messages exchange flow as follows:



1. Client sends stopStream message

```
{
  "message": "stopStream",
  "data": [
    {
      "mediaSessionId": "22588500-fdb3-11e8-bddf-9b115426f01b",
      "name": "test",
      "published": false,
      "hasVideo": true,
      "hasAudio": true,
      "status": "PLAYING",
      "record": false
    }
  ]
}
```

Message parameters

Parameters	Description
mediaSessionId	WebRTC media session Id
name	Published stream name
published	Stream is published
hasVideo	Stream has video

hasAudio	Stream has audio
status	Stream status
record	Stream is recorded on server

2. Server stops playback and sends notifyStreamStatusEvent message

```
{
  "message": "notifyStreamStatusEvent",
  "data": [
    {
      "mediaSessionId": "22588500-fdb3-11e8-bddf-9b115426f01b",
      "name": "test",
      "published": false,
      "hasVideo": true,
      "hasAudio": true,
      "status": "STOPPED",
      "audioCodec": "opus",
      "videoCodec": "H264",
      "info": "Stopped by user",
      "record": false,
      "width": 320,
      "height": 240,
      "bitrate": 0,
      "minBitrate": 0,
      "maxBitrate": 0,
      "quality": 0,
      "createDate": 1544580053573,
      "streamInfo": {
        "nodeId": null,
        "appKey": null,
        "sessionId": null,
        "mediaSessionId": "22588500-fdb3-11e8-bddf-9b115426f01b",
        "name": "test",
        "samplingTime": null,
        "recordTimestamp": null,
        "recordStarted": false
      },
      "mediaProvider": "WebRTC"
    }
  ]
}
```

Messages parameters

Parameter	Description
mediaSessionId	WebRTC media session Id
name	Stream name to play
published	Stream is published
hasVideo	Stream has video
hasAudio	Stream has audio
status	Stream status
audioCodec	Audio codec
videoCodec	Video codec
info	Stream status text description
record	Stream is recorded on server
width	Picture width (set while playing if 0)

height	Picture height (set while playing if 0)
bitrate	Bitrate (will set while playing if 0)
minBitrate	Minimum bitrate (no limits if 0)
maxBitrate	Maximum bitrate (no limits if 0)
quality	Stream quality
createDate	Creation date as integer
streamInfo	Additional stream information
mediaProvider	Publishing technology

Publishing/playback stream metrics

Since build [5.2.576](#), main publishing/playback stream metrics were added to `messagesInboundVideoRate`, `outboundVideoRate`, which are intended to [control publishing/playback channel quality](#):

```
{
  "message": "inboundVideoRate",
  "data": [
    {
      "mediaSessionId": "a12145d0-7eee-11ea-9833-9da59e0cc100",
      "videoRate": 704856,
      "fps": 31,
      "nack": 0,
      "pli": 0,
      "lostPackets": 0,
      "width": 640,
      "height": 360
    }
  ]
}
{
  "message": "outboundVideoRate",
  "data": [
    {
      "mediaSessionId": "0040ed40-7eef-11ea-9833-9da59e0cc100",
      "videoRate": 685256,
      "fps": 30,
      "nack": 0,
      "pli": 13,
      "lostPackets": 0,
      "width": 640,
      "height": 360
    }
  ]
}
```

Messages parameters

Parameter	Description
mediaSessionId	WebRTC media session Id
videoRate	Video bitrate, bps
fps	FPS
nack	NACK packets count
pli	PLI queries count
lostPackets	Lost packets count
width	Picture width

height	Picture height
--------	----------------

Stream availability checking

To check if a stream is available on the server, client should sendavailableStream message

```
[
  {
    "mediaSessionId": "2f693140-7d54-11eb-91fe-1fd0fb28f180",
    "name": "test"
  }
]
```

Message parameters

Parameter	Description
mediaSessionId	WebRTC media session Id
name	Stream name

The server will respond withavailableStream message

```
[
  {
    "status": "false",
    "info": "Restricted access",
    "id": "2f693140-7d54-11eb-91fe-1fd0fb28f180"
  }
]
```

Message parameters

Parameter	Description
status	true if stream is available false if stream is not available
info	The unavailability reason phrase
id	WebRTC media session Id

The unavailability reason phrase is supported since build[5.2.911](#)