

Deliver WebRTC streams to viewers using the Wowza Video REST API

Updated on 05/25/2022 5:44 pm PDT

WebRTC lets you capture video from your browser without installing plug-ins or downloading third-party software. With Wowza Video, you can deliver WebRTC streams to viewers or other video services directly from the transcoder. Egress costs for direct playback are typically greater than the costs associated with delivering streams to viewers through a CDN. For more information about cost, audience size, and other workflow considerations, see [WebRTC workflows in Wowza Video](#).

You'll need to choose which workflow you'll use, **live stream** or **transcoder**, before you begin this task. See [Decide between a live stream or transcoder workflow](#) for more information about these workflows.

Before you start

You should be familiar with the following concepts:

- **API authentication methods.** We use JSON web tokens for API authentication. See [Authentication](#) for more information.
- **Environment variables.** We use environment variables for the API version and your JWT in the cURL API request examples in this topic to make it easier for you to copy, paste, and run commands in your Terminal or Command Prompt window. If you don't set environment variables for these values, you'll need to manually enter the correct values in the code samples throughout this tutorial. See [Tools for testing the API](#) for instructions.

Live stream workflow

[Transcoder workflow](#)

1. Create a live stream

Create a live stream that receives a WebRTC source by sending a POST request to the `/live_streams` endpoint.

You can use the following sample request, making sure to:

- Set `encoder` to **other_webrtc**.
- Set `aspect_ratio_height` to 1280. Recommended value for WebRTC streams.
- Set `aspect_ratio_width` to 720. Recommended value for WebRTC streams.
- Set `broadcast_location` to the region that's closest to your video source.
- Set `delivery_method` to **push**. Wowza Video doesn't support **pull** or **cdn** connections for WebRTC.
- Change any values unique to your broadcast, using the API reference documentation as a resource. See the **Endpoint Reference** button below.

Sample request

[Endpoint Reference](#)

```

curl -X POST \
-H "Content-Type: application/json" \
-H "Authorization: Bearer ${WV_JWT}" \
-d '{
  "live_stream": {
    "aspect_ratio_height": 720,
    "aspect_ratio_width": 1280,
    "billing_mode": "pay_as_you_go",
    "broadcast_location": "us_west_california",
    "delivery_method": "push",
    "encoder": "other_webrtc",
    "name": "MyWebRTCStream",
    "transcoder_type": "transcoded"
  }
}' "${WV_HOST}/api/${WV_VERSION}/live_streams"

```

Sample response

The response includes:

- An *id* for the live stream you'll use in the next step.
- *source_connection_information* you'll use in the next step to configure the WebRTC publish page.
 - *sdp_url*, *application_name*, and *stream_name*
- An array of *direct_playback_urls*. You'll use the URL with the name *webrtc*.

```

{
  "live_stream": {
    "id": "1234abcd",
    "name": "MyWebRTCStream",
    ...
    "encoder": "other_webrtc",
    ...
    "source_connection_information": {
      "sdp_url": "wss://[subdomain].entrypoint.video.wowza.com/webrtc-session.json",
      "application_name": "app-30z15349",
      "stream_name": "32a5814b",
    },
    ...
    "direct_playback_urls": {
      ...
      "webrtc": [
        {
          "name": "source",
          "url": "wss://[subdomain].entrypoint.video.wowza.com/webrtc-session.json",
          "application_name": "app-1880020Q",
          "stream_name": "3f8f1889"
        },
        {
          "name": "webrtc",
          "url": "wss://[subdomain].entrypoint.video.wowza.com/webrtc-session.json",
          "application_name": "app-1880020Q",
          "stream_name": "3f8f1889_stream1",
          "output_id": "wpdlffwf"
        }
      ]
    },
    ...
  }
}

```

(Optional) Decrease latency

The default buffer size for a live stream is 4 seconds. To decrease the latency of the stream and get closer to real-time, you can change the buffer to 0. You'll make this change on the transcoder created for the live stream.

1. Send a request to GET `/transcoders` to get the transcoder ID for your live stream. See the [API reference](#) for more information about this endpoint.
2. Send a request to PATCH `/transcoders/[ID]`, where `ID` is the transcoder ID returned in the previous step, and set the `buffer_size` to 0. See the [API reference](#) for more information about this endpoint.

(Optional) Change viewer limit

The default viewer limit is 10 for direct playback URLs, but most transcoders can support up to 300 viewers for WebRTC playback. You'll make this change on the transcoder created for the live stream.

1. Send a request to GET `/transcoders` to get the transcoder ID for your live stream. See the [API reference](#) for more information about this endpoint.
2. Send a request to PATCH `/transcoders/[ID]`, where `ID` is the transcoder ID returned in the previous step, and set the `play_maximum_connections` to a value between 10 and 300. See the [API reference](#) for more information about this endpoint.

2. Start the live stream

Before you can connect to the stream through the WebRTC publish page in the next step, you need to start the live stream. You'll need the `[live_stream_id]` returned in step 1.

```
curl -X PUT \
-H "Authorization: Bearer ${WV_JWT}" \
"${WV_HOST}/api/${WV_VERSION}/live_streams/[live_stream_id]/start"
```

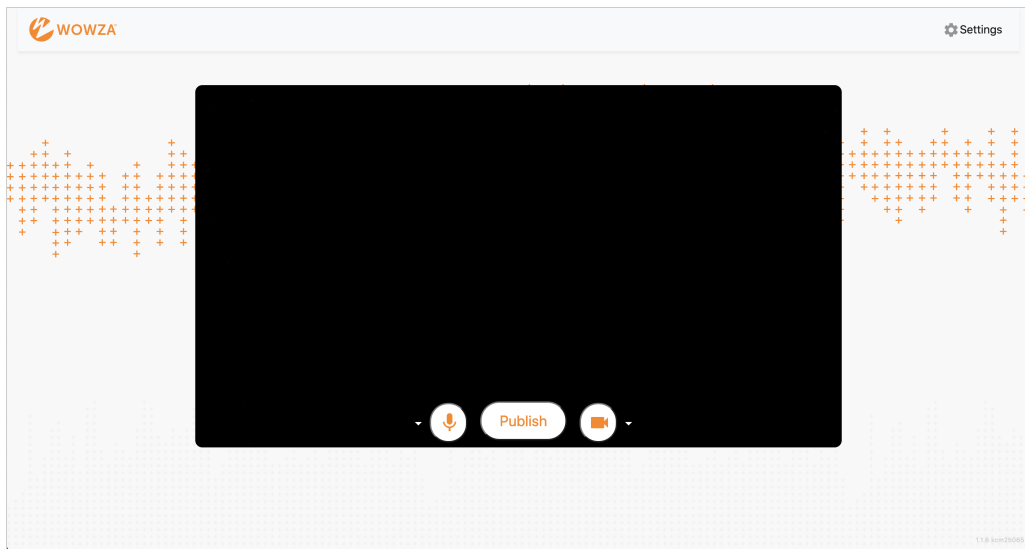
Endpoint Reference

3. Publish the WebRTC live stream

With the live stream started, you can now configure the WebRTC publish page as the source of the stream.

1. Go to wowza.com/webrtc/publish and grant access to your camera and microphone when prompted.

Note: The WebRTC hosted publish page is supported on the latest versions of Chrome and Safari, as well as Edge version 79 and later.



2. Click the **Settings** button on the upper-right corner of the page.
3. Specify the following settings:
 - **SDP URL** – The `sdp_url` from the `WebRTC source_connection_information` object.
 - **Application Name** – The `application_name` from the `WebRTC source_connection_information` object.
 - **Stream Name** – The `stream_name` from the `WebRTC source_connection_information` object.
 - **Audio Bitrate** – You can leave **Audio Bitrate** set to the default value, **64**.
 - **Video Bitrate** – You can leave **Video Bitrate** set to the default value, **3500**.
4. Close the **Settings** dialog box to apply your changes.
5. To change which microphone to use, click the arrow next to the microphone icon.
6. To change the video source, click the arrow next to the video camera icon. To share your screen, select **Screen Share**.

Notes:

- The video source can't be changed after the broadcast is started.
- Screen share functionality isn't supported on mobile devices or Safari.

7. Click **Publish**.

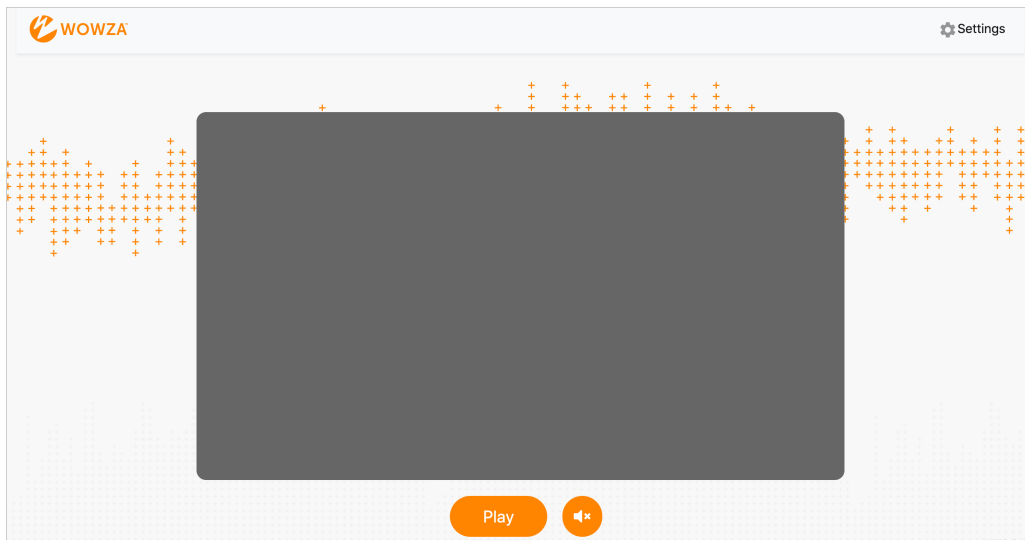
The stream now has source video that you can view on the WebRTC playback page.

4. Play the WebRTC live stream

You can test the playback using our WebRTC playback page. See the [Distributable viewer URL](#) section to get a URL you can send to your viewers.

1. In a new browser tab, go to wowza.com/webrtc/play.

Note: The WebRTC hosted playback page is supported on the latest versions of Chrome and Safari, as well as Edge version 79 and later.



2. Click the **Settings** button on the upper-right corner of the page.
3. Specify the following settings:
 - **SDP URL** – The *url* from the *webrtc* rendition of the WebRTC *direct_playback_urls* object.
 - **Application Name** – The *application_name* from the *webrtc* rendition of the WebRTC *direct_playback_urls* object.
 - **Stream Name** – The *stream_name* from the *webrtc* rendition of the WebRTC *direct_playback_urls* object.

Make sure you don't use the *stream_name* from *source_connection_information*. That object should not be distributed to viewers.

4. Close the **Settings** dialog box to apply your changes.
5. Click **Play**.

Distributable viewer URL

You can get a URL to distribute to your viewers so they can watch the WebRTC stream on a preconfigured, hosted page in one of two ways:

- Log into Wowza Video and use the **WebRTC Playback Page** URL on the **Overview** page for the live stream.
- Using the information in step 3, build the URL with the following format:

```
https://wowza.com/webrtc/play?playApplicationName=[application_name]&playSdpURL=[sdp_url]&playStreamName=[stream_name]&title=[title]
```

Where *sdp_url* is escaped as in this example:

```
wss%3A%2F%2F[subdomain].entrypoint.video.wowza.com%2Fwebrtc-session.json
```

5. Stop the live stream

To avoid incurring additional billing charges, remember to stop the live stream after the stream broadcast is

over.

```
curl -X PUT \  
-H "Authorization: Bearer ${WV_JWT}" \  
"${WV_HOST}/api/${WV_VERSION}/live_streams/[live_stream_id]/stop"
```

Endpoint Reference

More live stream resources

- [Wowza Video REST API reference documentation](#)
- [WebRTC workflows in Wowza Video](#)
- [Deliver WebRTC streams to viewers through a CDN using the Wowza Video REST API](#)