How to broadcast video chats to a streaming platform with ultra-low latency using Daily and Amazon IVS

by Sathya Balakrishnan | on 20 JUL 2021 | in Amazon Interactive Video Service, Broadcast, Direct-to-Consumer & Streaming, Media & Entertainment, Media Services | Permalink | Share

Authored by Jessica Mitchell, Engineer, at Daily. The content and opinions in this post are those of the third-party author and AWS is not responsible for the content or accuracy of this post.

At Daily, we build real-time video and audio APIs that let developers embed calls into any site or app, with support for up to 1,000 participants. Recently, there has been a growing demand for a reliable live streaming solution. As more interactions move online, customers who typically rely on WebRTC for real-time conversations now need to host even larger events like webinars, live conferences, fitness classes, concerts, and large meetings on streaming platforms that reach millions.

To meet this growing demand, Daily offers live streaming of Daily calls via RTMP streaming. This format can be consumed by several existing solutions — most notably Amazon Interactive Video Service (Amazon IVS). It also provides ultra-low latency with only a two to five seconds of delay.

Daily prides itself on offering the simplest solution for video calls and live streaming, requiring only a few lines of code to get started. In addition to ease of use and low latency, Daily provides customizable layouts for live streaming or multi-participant calls, and full HD resolution. Our live streaming accommodates up to nine video feeds on screen at a time and a configurable number of simultaneous streams per account.

In this post, we show you how to set up a Daily video call in your app, initiate RTMPS live streaming, and have the stream consumed by the Amazon IVS player.

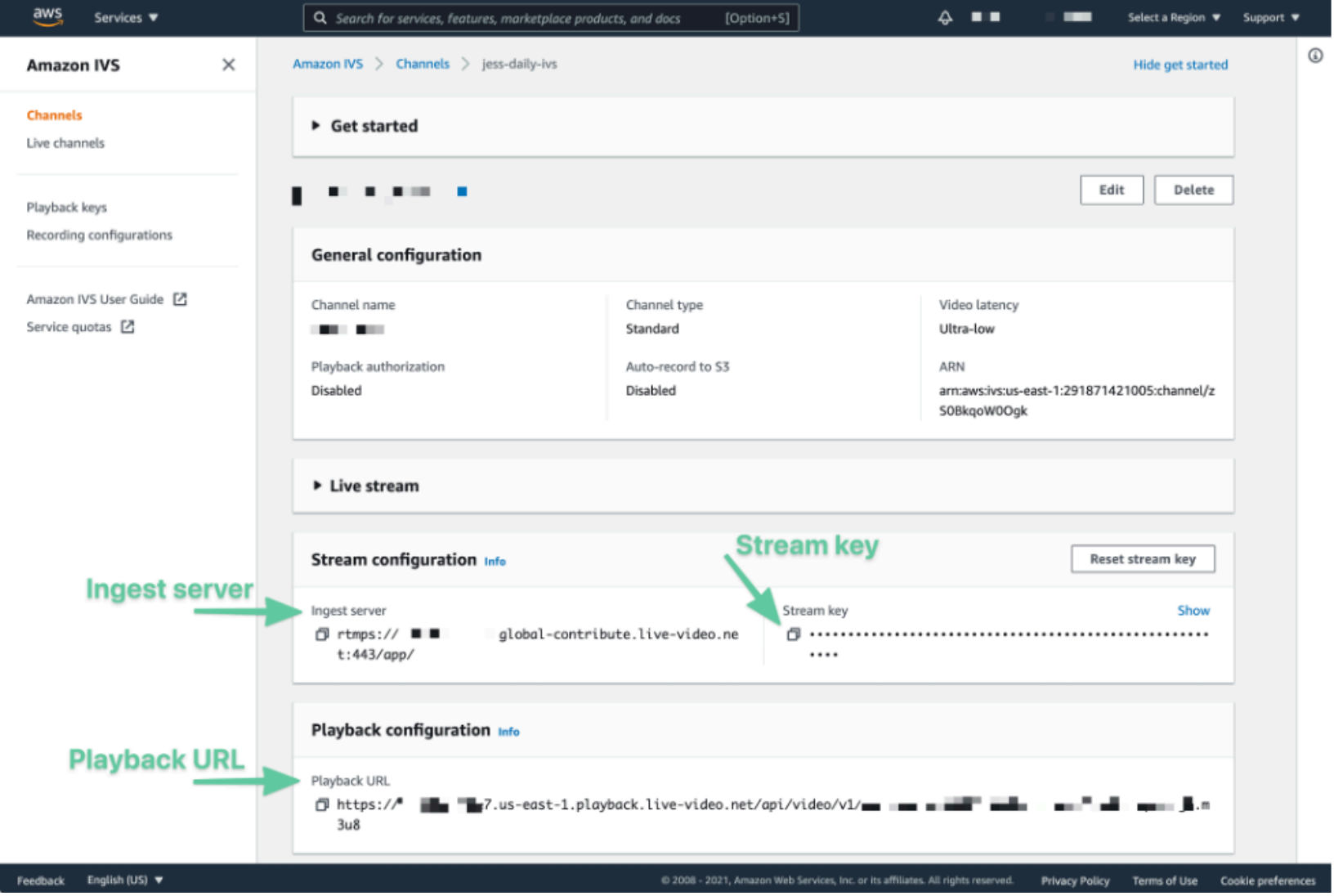
Requirements

To get started, there are a few requirements you will need to set up:

1. A Daily paid account (Launch or Scale) to access the live streaming feature.
2. A Daily room, which can be created via the Daily dashboard or the REST API.
3. A meeting token for the Daily room, with the property `is_owner` set to `true`.
4. An AWS account to use Amazon IVS.
5. An Amazon IVS streaming channel. A channel can be created through the Amazon IVS console or through the Amazon IVS API directly.

Once you have a streaming channel created, keep the following pieces of information provided by Amazon IVS:

1. The ingest server
2. The stream key
3. The playback URL



Amazon IVS's console for a new streaming channel

Create a Daily video call

Adding a Daily video call to your app is as simple as adding a few lines of code. To demonstrate this, let's look at Daily's prebuilt demo app, which uses our prebuilt video UI. (Developers also can build their own custom video interfaces with our front-end libraries.)

In `index.html`, there is a script tag to include `daily-js`, Daily's front-end JavaScript library. There is also a `<div>` element that can be selected in the `index.js` file via its ID. This is where the video iframe is inserted by `daily-js`.

In `index.js`, there are three main steps to start a live stream. These steps require the Daily and Amazon IVS prerequisite data created earlier, including the RTMPS endpoint as shown by the following.

```
async function setup() {
  callFrame = await window.DailyIframe.createFrame(
    document.getElementById('callframe'),
    {
      iframeStyle: {
        // ... add custom styles
      },
    }
  );
  // ...
}

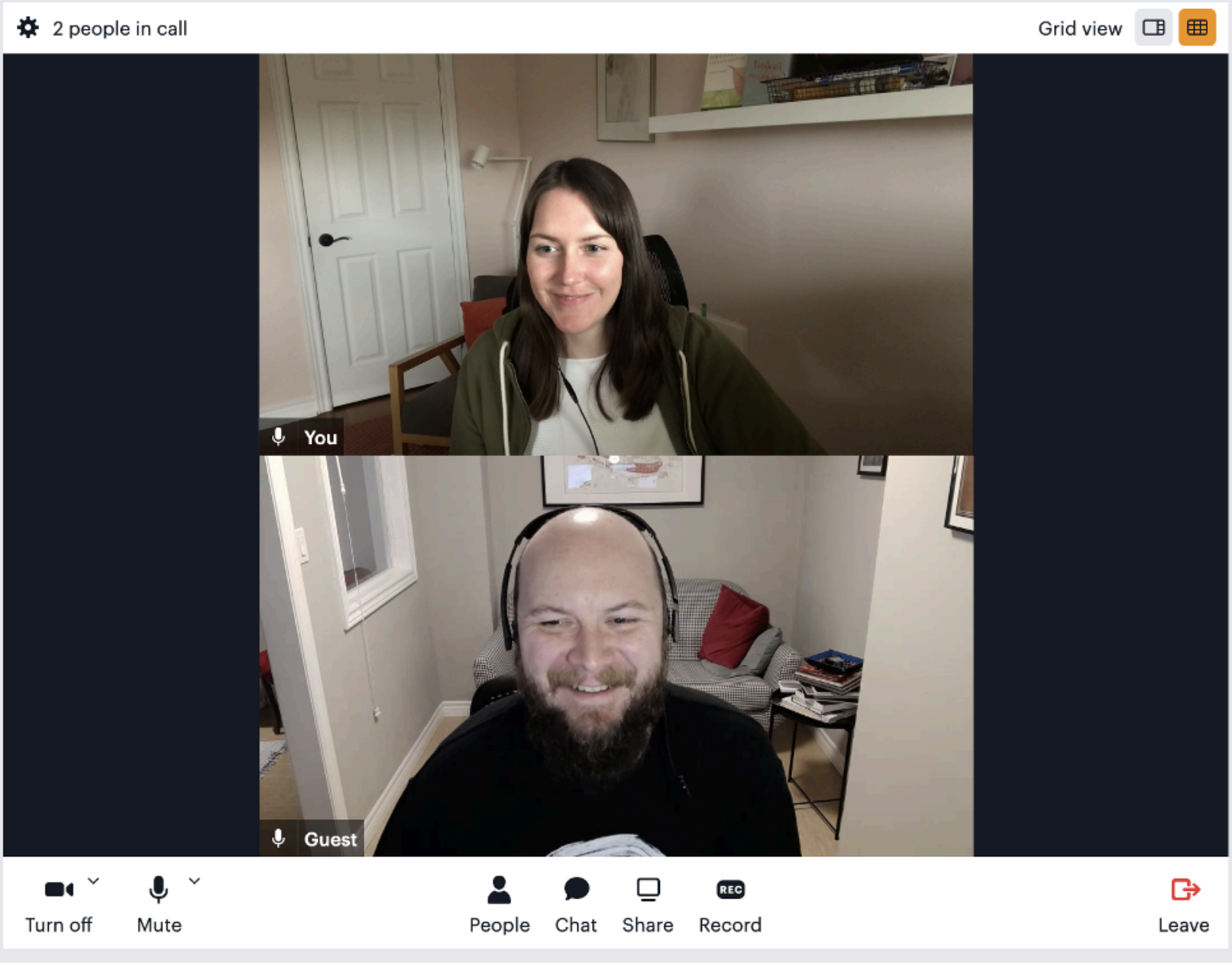
async function joinCall() {
  const roomURL = 'DAILY_ROOM_URL';
  const token = 'MEETING_OWNER_TOKEN';
  await callFrame.join({
    url: roomURL,
    token,
  });
}

function startLiveStreaming() {
  callFrame.startLiveStreaming({ rtmpUrl: 'INSERT_RTMP_URL' });
  // rtmpUrl = rtmps://RTMP_ENDPOINT/STREAM_KEY
}
```

The Daily Prebuilt demo functions required to start live streaming

1. Initialize the Daily iframe with the `createFrame()` Daily method.
2. Once the Daily call has been initialized, join the Daily room created earlier with the `join()` method. This method should be passed an object with the Daily room URL and meeting owner token set, also created earlier.
3. After the room has been joined, start live streaming the Daily call by choosing the demo's "Start live streaming" button. This triggers the `startLiveStreaming()` Daily method, which accepts an object with the RTMP URL set, as well as additional configurable options. In the case of Amazon IVS, this is an RTMPS URL.The RTMPS URL uses two pieces of information provided by the Amazon IVS console (or REST API) from the new channel created earlier: the ingest server and the stream key.

As soon as `startLiveStreaming` successfully resolves, your Daily call is officially live streaming.



Daily's embedded iframe with two call participants

Viewing your live stream with Amazon IVS

Now that you have your Daily video streaming, playing the live stream is as simple as passing the playback URL to the Amazon IVS player, using the player SDK. To test the Daily live stream with the player, use the Amazon IVS basic web sample from the code repository list, or follow the player instructions for web, Android, or iOS.

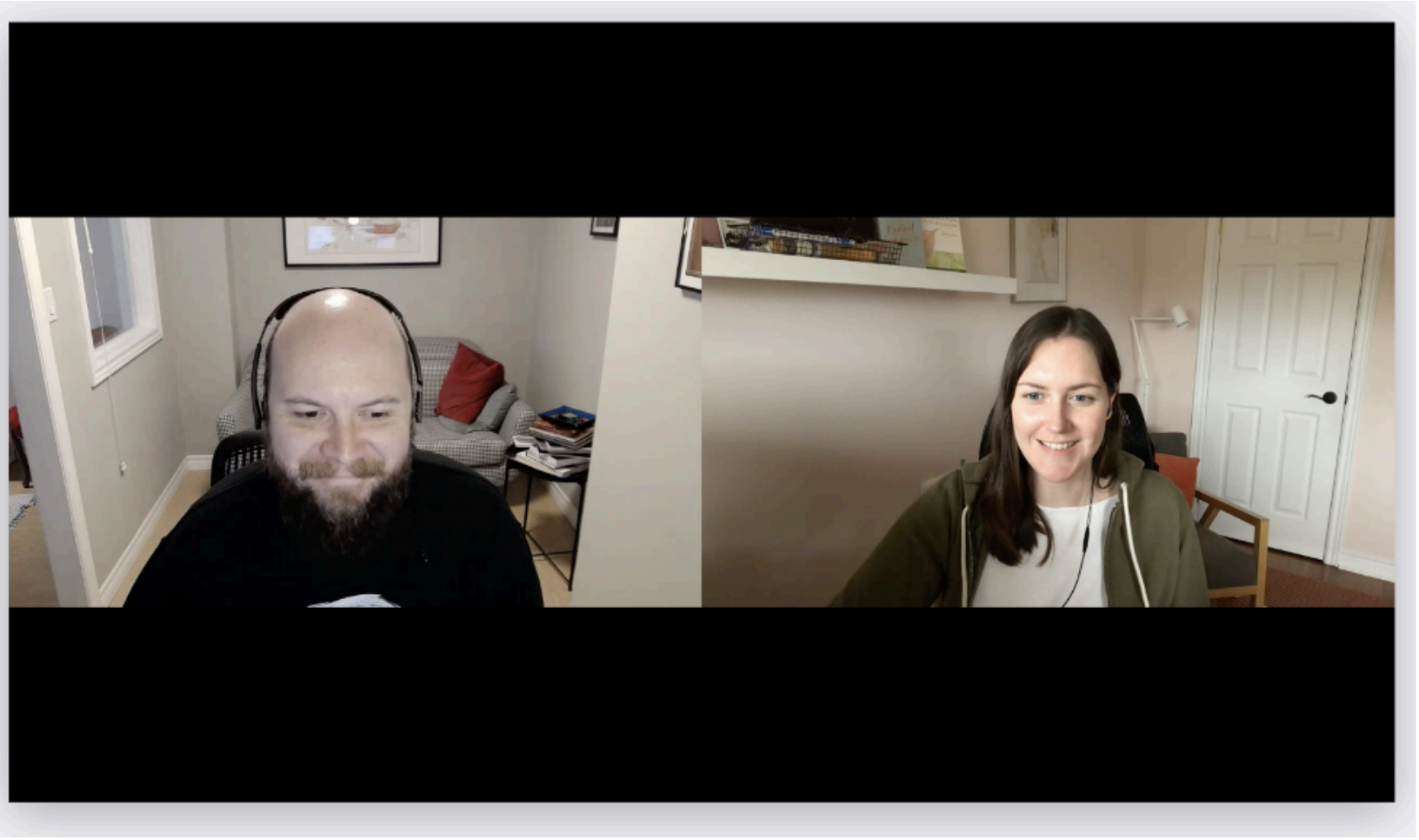
Once you have the player available, use the playback URL supplied by Amazon IVS in the player setup. This playback URL is one of the pieces of information mentioned earlier while creating a new channel.

```
// Replace this with your own Amazon IVS Playback URL
const playbackUrl = 'PLAYBACK_URL';

if (IVSPlayer.isPlayerSupported) {
  const player = IVSPlayer.create();
  player.attachHTMLVideoElement(document.getElementById
  ('video-player'));
  player.load(playbackUrl);
  player.play();
}
```

Example code from the Amazon IVS basic web sample repo

To confirm that the live stream is available, load the player in the browser and you can see your live stream with the ultra-low latency of only 2–5 seconds.



A Daily live stream viewed through the Amazon IVS player

This setup can be further customized by using Daily's various layouts, such as grid, active speaker, screen share, or single participant views.

Once your live stream content has ended, you can stop live streaming your Daily call by simply calling `callFrame.stopLivestreaming()` in your Daily call code.

About Daily

At Daily, our goal is to build video and audio APIs that inspire and support developers. We believe that live video is changing how we all live and work — and it should be much simpler, and faster, for developers and product teams to build creatively with video.

Using Daily with Amazon IVS provides reliable live streaming with ultra-low latency, with only a few lines of code. With Daily and Amazon IVS both being quick and easy to use, we've solved the hard parts of live streaming video for developers so they can focus on building better products faster.

TAGS: Amazon Interactive Video Service, Amazon IVS, AWS Startups, event management, virtual event



Sathya Balakrishnan

Sathya Balakrishnan is a Sr. Startup Solutions Architect based in San Francisco. He works with startups in different verticals to build innovative solutions on AWS. He is passionate about distributed systems, DevOps and emerging technologies like Containers and Serverless. When not working, he likes to read books and spend time with family.