Live Streaming on AWS

Implementation Guide

Live Streaming on AWS: Implementation Guide

Table of Contents

Build highly available live video streaming content using AWS Media Services and Amazon CloudFront

Publication date: *November 2016 (last update (p. 25): November 2021)*

Amazon Web Services (AWS) lets broadcasters and content owners to seamlessly scale infrastructure to broadcast live content to a global audience. The Live Streaming on AWS solution helps you build highly available live video streaming content using AWS Media Services and Amazon CloudFront that is highly resilient and secure to deliver real-time viewing experiences to your customers.

This solution provides the following features:

- Encodes and packages your content for adaptive bitrate streaming across multiple screens via HTTP live streaming (HLS), Dynamic Adaptive Streaming over HTTP (DASH), and Common Media Application Format (CMAF) by automatically configuring AWS Elemental MediaLive and AWS Elemental MediaPackage.
- Provides an elastic, highly available, global content delivery network for live video streaming using Amazon CloudFront.

With this solution, you can run it only during a live event and then after the program ends, delete the solution's stack to ensure you only pay for the infrastructure that you use.

This implementation guide discusses architectural considerations and configuration steps for deploying Live Streaming on AWS in the AWS Cloud. It includes a link to an AWS CloudFormation template that launches and configures the AWS services required to deploy this solution using AWS best practices for security and availability.

The guide is intended for IT infrastructure architects, administrators, and DevOps professionals who have practical experience with video streaming and architecting in the AWS Cloud.

Cost

You are responsible for the cost of the AWS services used while running this solution. As of November 2021, the cost for running this solution in the US East (N. Virginia) with:

- Approximately 1,000 viewers for a one-hour live event using a standard definition (SD)-540p encoding profile is approximately **$2.25** for live encoding and packaging + **$67.24** for 791GB distribution = **$69.49 for the one-hour event**.
- Approximately 10,000 viewers for a one-hour live event using a high definition (HD)-1080p encoding profile is approximately **$12.50** for live encoding and packaging + **$1531.49** for 18,017GB distribution = **$1,543.99 for the one-hour event**.

These cost estimates depend on many factors, which are detailed in the following cost examples.

> **Note**
> The examples provided are likely higher than the actual costs of running this solution. The intent was to provide a guide to the pricing that is easily understood. Where assumptions were needed, we used factors that were straightforward to calculate and also likely be more expensive than the actual cost.

For an additional cost example for streaming a live event, refer to the FAQs about live streaming on AWS blog in the *AWS Media Blog*.

Cost Example 1

Cost example 1 covers a use case of approximately 1,000 viewers viewing a live event for about one hour with a **SD-540p** encoding profile selected in the CloudFormation template. This cost example is based on the following factors:

- Pricing Region: US-East-1, assuming standard pricing (no free-tier or discounts).
- Viewers consume the highest bitrate: Note that bitrate consumption is a mix of all the streams, but the highest bitrate is used in the calculation to show the upper cost range. Additionally, the Quality-Defined Variable Bitrate (QVBR) and variable video complexity can result in an output bandwidth that is 10-50% lower in price than the estimate provided in Table 1.
- 99% cache/hit ratio between the content delivery network (CDN) and AWS Elemental MediaPackage.

> **Note**
> Storage of the test player is not included in this cost estimate.

Table 1 summarizes the total pricing for the live streaming event. Tables 2 through 4 break down the cost for each AWS service.

Table 1: Cost breakdown for 1,000 viewers for a one-hour live event

AWS service	Function	Cost per hour
AWS Elemental MediaLive	Input and outputs for channel	$1.71
AWS Elemental MediaPackage	Ingest of channel	$0.14
	Packaging and origination	$0.40
Amazon CloudFront	Distribution	$67.24

AWS service	Function	Cost per hour
	Total:	**$69.49/hour**

AWS Elemental MediaLive pricing

Table 2 breaks down the AWS Elemental MediaLive pricing which assumes HD AVC input and SD AVC outputs with less than 10 Mbps bitrate and less than 30 frames per second (fps) frame rate.

Table 2: MediaLive pricing

Input / Output	Cost per hour
HD Input (AVC 10-20Mbps)	$0.294
Output 1 SD (512x288, 400k)	$0.354
Output 2 SD (640x360, 800k)	$0.354
Output 3 SD (768x432, 1200k)	$0.354
Output 4 SD (960x540, 1800k)	$0.354
Total:	**$1.71/hour**

AWS Elemental MediaPackage pricing

MediaPackage charges $0.03 per Gigabyte (GB) per hour for standard Live ingest, based on the aggregate bitrate of all live input streams. This example has four input streams, so the rate of GB streaming per hour is determined by:

1. Adding the bitrate of all streams in either kbps or Mbps:
   - 400k + 800k + 1200k + 1800k = 4200 kbps
   - .4 + .8 + 1.2 + 1.8 = 4.2 Mbps
2. Convert kbps or Mbps to Gigabits (Gbit) per second (ps). 1024 Mb = 1 Gbit:
   - 4.2 Mbps / 1024 = 0.0041015625 Gbits ps
3. Convert Gbits to GB. 1 Gbit = 0.125 (or 1/8) GB.
   - 0.0041015625 Gbits ps * 0.125 = 0.0005126953125 GBps
4. Convert GB per second to GB per hour:
   - 0.0005126953125 GBps * 60s * 60mins = 1.845703125 or 1.8 GB/hour

     ((sum of stream bitrates in Kbps) / 1024 (Mbps conversion)) / 1024 (Gbit conversion) * .125 (bits to Bytes conversion) * 60 seconds * 60 minutes = GB / hour

Table 3: MediaPackage ingest pricing

Input / Output	kbps
Output 1 SD (512x288, 400k)	400
Output 2 SD (640x360, 800k)	800

Input / Output	kbps
Output 3 SD (768x432, 1200k)	1200
Output 4 SD (960x540, 1800k)	1800
Total kbps:	4200
Convert kpbs to GB/hour 4,200 kbps / 8 (bits to bytes) / 1024 (kpbs to Mbps) / 1024 (Mbps to Gbps) x 60 (seconds) x 60 (minutes)	1.80 GB/hour
GB/hour with redundancy	3.60 GB/hour
Cost (GB/hour * $0.03)	**$0.14/hour**

Table 4: MediaPackage packaging and origin pricing

AWS service	Cost per hour
Total MediaPackage with 1% of egress (7.91 GB x $0.05)	$0.40

Note
We assume that MediaPackage serves 1% of the traffic to CloudFront and viewers, and the rest is served by CloudFront. Table 5 provides the 1% of egress for the MediaPackage estimate in Table 4.

Viewer traffic pricing

The cost estimate for viewer traffic assumes that all viewers get the highest bitrate for the one-hour live streaming event.

Table 5: Viewer traffic pricing

AWS service	Function	Total
Amazon CloudFront	Average Mbps per viewer	1.8
	Total MB per sec (1000 x 1.8 / 8)	225
	Total Egress per hour (225 / 1024 x 60 x 60)	791 GB/hour
	Total cost (791 GB * $0.085):	**$67.24/hour**

Cost example 2

Cost example 1 covers a use case of approximately 10,000 viewers viewing a live event for about one hour with an HD-1080p encoding profile selected in the CloudFormation template. This cost example is based on the following factors:

- Pricing Region: US-East-1, assuming standard pricing (no free-tier or discounts).
- Viewers consume the highest bitrate: Note that bitrate consumption is a mix of all the streams, but the highest bitrate is used in the calculation to show the upper cost range. Additionally, the QVBR and variable video complexity can result in an output bandwidth that is 10-50% lower in price than the estimate provided in Table 6.
- 99% cache/hit ratio between the CDN and AWS Elemental MediaPackage.

Note
Storage of the test player is not included in this cost estimate.

Table 6 summarizes the total pricing for the live streaming event. Tables 7 through 10 breaks down the cost for each AWS service.

Table 6: Cost breakdown for 10,000 viewers for a one-hour live event

AWS service	Function	Cost per hour
AWS Elemental MediaLive	Input and outputs for channel	$3.11
AWS Elemental MediaPackage	Ingest of channel	$0.38
	Packaging and origination	$9.01
Amazon CloudFront	Distribution	$1,531.49
	Total:	**$1,543.99/hour**

AWS Elemental MediaLive pricing

Table 7 breaks down the AWS Elemental MediaLive pricing which assumes HD AVC input and both SD and HD AVC outputs with less than 10 Mbps bit rate and less than 30 fps frame rate.

Table 7: MediaLive pricing

Input / Output	Cost per hour
HD Input (AVC 10-20Mbps)	$0.294
Output 1 SD (512x288, 400k)	$0.354
Output 2 SD (640x360, 800k)	$0.354
Output 3 SD (768x432, 1200k)	$0.354
Output 4 SD (960x540, 1800k)	$0.354
Output 5 HD (1280x720p, 2700k)	$0.702
Output 6 HD (1920x1080p, 4100k)	$0.702
Total:	**$3.11/hour**

AWS Elemental MediaPackage ingest pricing

As detailed in Cost Example 1, MediaPackage pricing is based on GB ingested per second across all streams of live output. Table 8 shows the conversion from kbps to Mbps to Gbits/second to GB/hour.

Table 8: MediaPackage ingest pricing

Input / output	kbps
Output 1 SD (512x288, 400k)	400
Output 2 SD (640x360, 800k)	800
Output 3 SD (768x432, 1200k)	1200
Output 4 SD (960x540, 1800k)	1800
Output 5 (1280x720p_2700k)	2700
Output 6 (1920x1080p_4100k)	4100
Total kbps:	11000
Convert kpbs to GB/hour 11000 kbps / 8 (bits to bytes) / 1024 (kpbs to Mbps) / 1024 (Mbps to Gbps) x 60 (seconds) x 60 (minutes)	4.72
GB/hour with redundancy	9.44
Total (GB/hour * $0.03)	**$0.28/hour**

Table 9: MediaPackage packaging and origin pricing

AWS service	Cost per hour
Total MediaPackage with 1% of egress (180 GB x $0.05)	$9.01

> **Note**
> We assume that MediaPackage serves 1% of the traffic to CloudFront and viewers, and the rest is served by CloudFront. Table 10 provides the 1% of egress for the MediaPackage estimate in Table 9.

Estimating the viewer traffic

The cost estimate for viewer traffic assumes that all viewers get the highest bitrate for the one-hour live streaming event.

Table 10: Viewer traffic pricing

AWS service	Function	Total
Amazon CloudFront	Average Mbps per viewer	4.1

AWS service	Function	Total
	Total MB per sec (10,000 x 4.1 / 8)	5125
	Total Egress per hour (5125 / 1024 x 60 x 60)	18,017 GB/hour
	Total cost (18,017 GB * $0.085):	**$1,531.49/hour**

Note
Pricing is per minute, with a minimum of 10 minutes. Prices are subject to change. For full details, refer to the pricing webpage for each AWS service you will be using in this solution.

Architecture overview

Deploying this solution with the default parameters builds the following environment in the AWS Cloud.

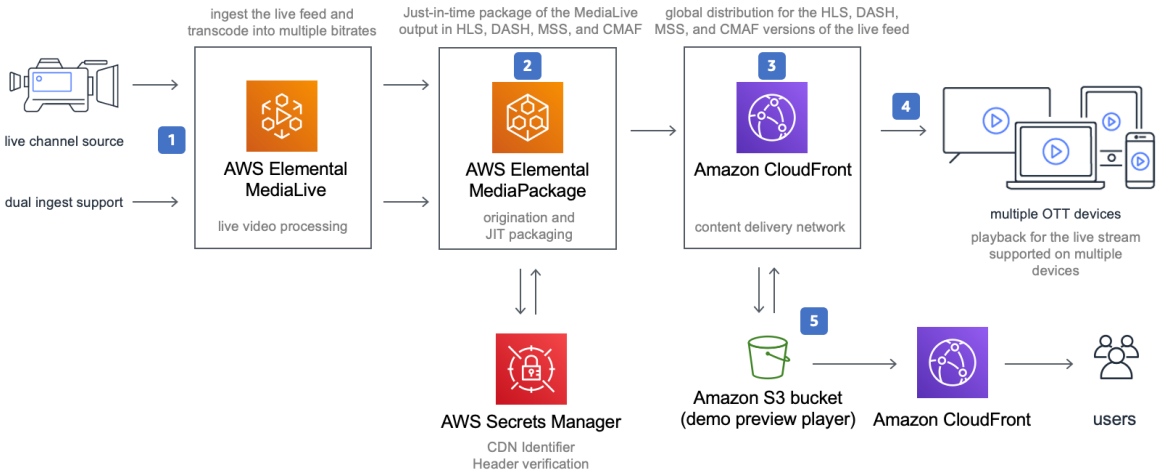

Figure 1: Live Streaming on AWS solution architecture

The solution's AWS CloudFormation template launches the AWS products and services necessary to ingest, transcode, and deliver live streaming video.

1. AWS Elemental MediaLive ingests two input feeds and transcodes your content into two adaptive bitrate (ABR) HTTP Live Streaming (HLS) streams as output.

   Two feeds are ingested for MediaLive redundancy. Each MediaLive transcodes a single ingest feed into ABR outputs. For more information, view the Creating Adaptive Bitrate (ABR) Outputs training.
2. AWS Elemental MediaPackage ingests the MediaLive ABR output and packages the live stream into HTTP Live Streaming (HLS), Dynamic Adaptive Streaming over HTTP (DASH), and Common Media Application Format (CMAF) formats that are delivered to three MediaPackage custom endpoints.

   These three formats were selected because that is what popular streaming devices support. MediaPackage is an origin server that prepares the content for delivery to phones, tablets, smart TVs, and other devices.
3. An Amazon CloudFront distribution is configured to use the MediaPackage custom endpoints as its origin and includes a CDN Identifier custom HTTP header to authenticate requests. MediaPackage only fulfills playback requests that are authorized between MediaPackage and CloudFront using the CDN Identifier. This CDN Identifier is created as part of the AWS CloudFormation deployment and securely stored in AWS Secrets Manager. For details, refer to the CDN Authorization in AWS Elemental MediaPackage in the *AWS Elemental MediaPackage User Guide*.
4. The CloudFront distribution delivers your live stream to viewers with low latency and high transfer speeds.
5. A demo HTML preview player is available to help you test the solution. The player is a static website hosted in an Amazon Simple Storage Service (Amazon S3) bucket. Amazon CloudFront is used to restrict access to the solution's website bucket contents.

You can configure this solution to ingest Real-Time Transport Protocol (RTP), Real-Time Messaging Protocol (RTMP), HTTP Live Streaming (HLS), or AWS Elemental MediaConnect flows. This solution also includes three encoding profiles (p. 10).

Security

When you build systems on AWS infrastructure, security responsibilities are shared between you and AWS. This shared model can reduce your operational burden as AWS operates, manages, and controls the components from the host operating system and virtualization layer down to the physical security of the facilities in which the services operate. For more information about security on AWS, visit AWS Cloud Security.

Amazon CloudFront

This solution deploys a static website hosted in an Amazon S3 bucket. To help reduce latency and improve security, this solution includes an Amazon CloudFront distribution with an origin access identity, which is a special CloudFront user that helps restrict access to the solution's website bucket contents. For more information, refer to Restricting Access to Amazon S3 Content by Using an Origin Access Identity.

Deployment considerations

Encoding profiles

The Live Streaming on AWS solution configures AWS Elemental MediaLive with one of three progressive, 30 frames-per-second encoding profiles. Choose one of the following encoding profiles.

- **HD-1080p profile:** 1920x1080, 1280x720, 960x540, 768x432, 640x360, 512x288
- **HD-720p profile:** 1280x720, 960x540, 768x432, 640x360, 512x288
- **HD-540p profile:** 960x540, 768x432, 640x360, 512x288

Regional deployment

This solution uses AWS Elemental MediaLive, MediaPackage, and MediaConnect, which are currently available in specific AWS Regions only. Therefore, you must launch this solution in an AWS Region where these services are available. For the most current service availability by Region, refer to the AWS Regional Services List. If you use MediaConnect as input, you must deploy this solution in the same Region as your MediaConnect flows.

AWS CloudFormation template

This solution uses AWS CloudFormation to automate the deployment of this live streaming solution on the AWS Cloud. It includes the following AWS CloudFormation template, which you can download before deployment:

live-streaming-on-aws.template: Use this template to launch the solution and all associated components. The default configuration deploys an AWS Lambda function, an AWS Elemental MediaLive input and channel, an AWS Elemental MediaPackage channel, two Amazon CloudFront distributions, and an Amazon Simple Storage Service (Amazon S3) bucket for the demo HTML preview player, but you can also customize the template based on your specific needs.

Automated deployment

Before you launch the solution, please review the architecture, configuration, network security, and other considerations discussed in this guide. Follow the step-by-step instructions in this section to configure and deploy the solution into your account.

Time to deploy: Approximately 20 minutes

Launch the stack

Important
This solution includes an option to send anonymous operational metrics to AWS. We use this data to better understand how customers use this solution and related services and products. AWS owns the data gathered though this survey. Data collection is subject to the AWS Privacy Policy.
To opt out of this feature, download the template, modify the AWS CloudFormation mapping section, and then use the AWS CloudFormation console to upload your template and deploy the solution. For more information, refer to the Collection of operational metrics (p. 23) section of this guide.

This automated AWS CloudFormation template deploys the Live Streaming on AWS solution in the AWS Cloud.

Note
You are responsible for the cost of the AWS services used while running this solution. Refer to the Cost (p. 2) section for more details. For full details, refer to the pricing webpage for each AWS service you will be using in this solution.

1. Sign in to the AWS Management Console and select the button to launch the `live-streaming-on-aws` AWS CloudFormation template.

   

   You can also download the template as a starting point for your own implementation.
2. The template launches in the US East (N. Virginia) Region by default. To launch this solution in a different AWS Region, use the Region selector in the console navigation bar.

   Note
   This solution uses the AWS Elemental MediaLive, MediaPackage, and MediaConnect services, which are currently available in specific AWS Regions only. Therefore, you must launch this solution in an AWS Region where these services are available. For the most current service availability by Region, refer to the AWS Regional Service List.
3. On the **Create stack** page, verify that the correct template URL is in the **Amazon S3 URL** text box and choose **Next**.
4. On the **Specify stack details** page, assign a name to your Live Streaming on AWS solution stack.
5. Under **Parameters**, review the parameters for the template, and modify them as necessary.

   This solution uses the following default values. Refer to URL_PULL (HLS) input configuration (p. 18), RTMP_PULL input configuration (p. 19), RTMP_PUSH and RTP_PUSH input configurations (p. 20), and MEDIACONNECT input configuration (p. 21) for detailed instruction for setting up each input type.

Parameter	Default	Description
LIVE STREAM SOURCE		
Source Input Type	`URL_PULL`	Specify the input type for AWS Elemental MediaLive: `RTP_PUSH`, `RTMP_PUSH`, `RTMP_PULL`, `URL_PULL`, or `MEDIACONNECT`.
URL_PULL and RTML_PULL CONFIGURATION		
Primary Source URL	`https://d15an60oaeed9r.cloudfront.net/live_stream_v2/sports_reel_with_markers.m3u8`	The primary source URL for the live feed. By default, this parameter contains the primary demo source URL.
Primary Source Username	<Optional input>	If authentication is required to access the source, enter the username.
Primary Source Password	<Optional input>	If authentication is required to access the source, enter the password.
Secondary Source URL	`https://d3h5srgm8b0t83.cloudfront.net/live_stream_v2/sports_reel_with_markers.m3u8`	The secondary (backup) source URL for the live feed. By default, this parameter contains the secondary demo source URL.
Secondary Source Username	<Optional input>	If authentication is required to access the secondary source, enter the username.
Secondary Source Password	<Optional input>	If authentication is required to access the secondary source, enter the password.
RTP_PUSH and RTMP_PUSH CONFIGURATION		
Input CIDR Block	*<Requires input>*	Specify the CIDR block for the MediaLive security group for push input types.
MEDIACONNECT CONFIGURATION		

Parameter	Default	Description
Primary MediaConnect ARN	<Optional input>	The primary source MediaConnect flow for the live feed. You can create the flow in the MediaConnect console. To provide redundancy, create the primary and secondary flows in different Availability Zones.
Secondary MediaConnect ARN	<Optional input>	The secondary source MediaConnect flow for the live feed. You can create the flow in the MediaConnect console. To provide redundancy, create the primary and secondary flows in different Availability Zones.
ENCODING OPTIONS		
Encoding Profile	`720`	Specify the encoding profile to use with MediaLive.
Start MediaLive Channel	`true`	Choose whether to start the MediaLive channel when the solution is created. We recommend starting the channel if you will use the *HTML preview player* (p. 17).

6. Choose **Next**.
7. On the **Configure stack options** page, choose **Next**.
8. On the **Review** page, review and confirm the settings. Check the box acknowledging that the template will create AWS Identity and Access Management (IAM) resources.
9. Choose **Create** to deploy the stack.

   You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should receive a **CREATE_COMPLETE** status in approximately 20 minutes.

   To test the live stream playback, navigate to the AWS CloudFormation stack **Outputs** tab, select the **Demo Console URL** from the **Value** column.

   Note
   In addition to the primary AWS Lambda function, this solution includes the `solution-helper` Lambda function, which runs only during initial configuration or when resources are updated or deleted.

When running this solution, both Lambda functions are displayed in the AWS Lambda console, do not delete the `solution-helper` function because it is necessary to manage associated resources.

Once the live streaming event is complete, terminate the resources created by this solution to help ensure that you don't incur unnecessary AWS charges.

Additional resources

AWS services

- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaConnect
- Amazon CloudFront
- AWS CloudFormation
- AWS Lambda

AWS blogs

- Connecting AWS Elemental Live On-Premises to AWS Media Services in the Cloud
- Connecting OBS Studio to AWS Media Services in the Cloud
- Connecting FFmpeg Using RTP to AWS Media Services in the Cloud
- Connecting FFmpeg Using RTMP to AWS Media Services in the Cloud
- Connecting VLC Media Player Using RTP to AWS Media Services in the Cloud

Tutorials and Workshops

- AWS Live Streaming and Live-to-VOD Workshop
- Build a Secure Packager and Encoder Key Exchange (SPEKE) Reference Server

HTML preview player

A demo HTML preview player is available to help you test the solution. The player is a static website hosted in an Amazon S3 bucket. It is pre-populated with the URLs that point to the newly created customer stream. The HTML/JavaScript application plays back HLS, DASH, MSS, and CMAF streams. In addition, the solution can be configured to ingest a Demo HLS feed hosted on AWS. Customize the HTML in the DemoBucket S3 bucket to suit your needs. For details about putting the HLS video into a webpage, refer to the Apache 2.0 Video.JS open source project.

To check output on the player, start the AWS Elemental MediaLive channel. Even if you selected `false` for the **Start MediaLive Channel** CloudFormation template parameter, you can go to the MediaLive channel to start the player when you are ready to start testing.

URL_PULL (HLS) input configuration

`URL_PULL` provides the option to ingest an HTTP live streaming (HLS) stream over HTTP or HTTPS. The following parameters are required to configure the solution to ingest an HLS stream:

- **Source Input Type:** `URL_PULL`.
- **Primary Source URL:** The HTTP(s) link to the HLS stream manifest file. The default value is a demo stream from AWS.
- **Primary Source Username:** Only required if you have basic authentication setup on your source HLS stream.
- **Primary Source Password:** Only required if you have basic authentication setup on your source HLS stream.
- **Secondary Source URL:** The HTTP(s) link to the HLS stream manifest file. The default value is a demo stream from AWS.
- **Secondary Source Username:** Only required if you have basic authentication setup on your source HLS stream.
- **Secondary Source Password:** Only required if you have basic authentication setup on your source HLS stream.
- **Encoding Profile:** Select the profile that best matches your source resolution.
- **Start MediaLive Channel:** If your device is ready to stream, select **true**. Otherwise, select **false**—you can start the AWS Elemental MediaLive channel through the AWS Management Console when you're ready to stream.

> **Note**
> For a full list of input types and configuration details, refer to the Creating an input topic in the *AWS Elemental MediaLive User Guide*.

RTMP_PULL input configuration

`RTMP_PULL` provides the option to ingest an RTMP stream. The following parameters are required to configure the solution to ingest an RTMP stream:

- **Source Input Type:** `RTMP_PULL`.
- **Primary Source URL:** The RTMP link to the primary source stream, for example rtmp://203.0.113.20:1935/primary.
- **Primary Source Username:** Only required if you have basic authentication setup on your source stream.
- **Primary Source Password:** Only required if you have basic authentication setup on your source stream.
- **Secondary Source URL:** The RTMP link to the primary source stream, for example rtmp://203.0.113.20:1935/secondary.
- **Secondary Source Username:** Only required if you have basic authentication setup on your source stream.
- **Secondary Source Password:** Only required if you have basic authentication setup on your source stream.
- **Encoding Profile:** Select the profile that best matches your source resolution.
- **Start MediaLive Channel:** If your device is ready to stream, select **true**. Otherwise, select **false**—you can start the AWS Elemental MediaLive channel through the AWS Management Console when you're ready to stream.

Note
For a full list of input types and configuration details, refer to the Creating an input topic in the *AWS Elemental MediaLive User Guide*.

RTMP_PUSH and RTP_PUSH input configurations

`RTMP_PUSH` and `RTP_PUSH` provide the option to push a transport stream (TS) to AWS Elemental MediaLive. In both options, the following parameters are required to configure the solution:

- **Source Input Type:** `RTP_PUSH` or `RTMP_PUSH`.
- **Input Security Group CIDR Block:** A valid CIDR block used to create a security group to restrict access to the MediaLive input.
- **Encoding Profile:** Select the profile that best matches your source resolution.
- **Start MediaLive Channel:** If your device is ready to stream, select **true**. Otherwise, select **false**—you can start the AWS Elemental MediaLive channel through the AWS Management Console when you're ready to stream.

> **Note**
> For a full list of input types and configuration details, refer to the Creating an input topic in the *AWS Elemental MediaLive User Guide*.

MEDIACONNECT input configuration

`MEDIACONNECT` provides the option to ingest a stream from AWS Elemental MediaConnect. The following parameters are required to configure the solution to ingest from MediaConnect:

- **Source Input Type:** `MEDIACONNECT`.
- **Primary MediaConnect ARN:** The ARN of the primary source stream, for example: arn:aws:mediaconnect:uswest1:111122223333:flow:1bgf67:primary. This MediaConnect flow must be in a different Availability Zone from the secondary stream.
- **Secondary MediaConnect ARN:** The ARN of the secondary source stream, for example: arn:aws:mediaconnect:uswest1:111122223333:flow:1bgf67:secondary. This MediaConnect flow must be in a different Availability Zone from the primary stream.
- **Encoding Profile:** Select the profile that best matches your source resolution.
- **Start MediaLive Channel:** If your device is ready to stream, select **true**. Otherwise, select **false**—you can start the AWS Elemental MediaLive channel through the AWS Management Console when you're ready to stream.

Note
For a full list of input types and configuration details, refer to the Creating an input topic in the *AWS Elemental MediaLive User Guide*.

Uninstall the solution

You can uninstall the Live Streaming on AWS solution from the AWS Management Console or by using the AWS Command Line Interface. You must manually delete the Amazon Simple Storage Service (Amazon S3) bucket created by this solution. AWS Solutions Implementations do not automatically delete the S3 bucket in case you have stored data to retain.

Using the AWS Management Console

1. Sign in to the AWS CloudFormation console.
2. On the **Stacks** page, select this solution's installation stack.
3. Choose **Delete**.

Using AWS Command Line Interface

Determine whether the AWS Command Line Interface (AWS CLI) is available in your environment. For installation instructions, refer to What Is the AWS Command Line Interface in the *AWS CLI User Guide*. After confirming that the AWS CLI is available, run the following command.

```
$ aws cloudformation delete-stack --stack-name <installation-stack-name>
```

Delete the Amazon S3 buckets

This solution is configured to retain the solution-created Amazon S3 buckets if you decide to delete the AWS CloudFormation stack to prevent accidental data loss. After uninstalling the solution, you can manually delete the S3 buckets if you do not need to retain the data. Follow these steps to delete the Amazon S3 buckets.

1. Sign in to the Amazon S3 console.
2. Choose **Buckets** from the left navigation pane.
3. Locate the *<stack-name>* S3 buckets.
4. Select the S3 bucket and choose **Delete**.

To delete the S3 buckets using AWS CLI, run the following command:

```
$ aws s3 rb s3://<bucket-name> --force
```

Collection of operational metrics

This solution includes an option to send operational metrics to AWS. We use this data to better understand how customers use this solution and related products and services. When enabled, the following information is collected and sent to AWS when the AWS CloudFormation template is launched:

- **Solution ID:** The AWS solution identifier
- **Unique ID (UUID):** Randomly generated, unique identifier for each live streaming solution deployment
- **Timestamp:** Data-collection timestamp
- **Launch Data:** The AWS Region where the stack was launched
- **Source Input Type:** The input type you selected at launch
- **Source Input Codec:** The codec you selected at launch
- **Source Input Resolution:** The resolution you selected at launch

AWS owns the data gathered through this survey. Data collection is subject to the AWS Privacy Policy. To opt out of this feature, complete the following steps before launching the AWS CloudFormation template.

1. Download the AWS CloudFormation template to your local hard drive.
2. Open the AWS CloudFormation template with a text editor.
3. Modify the AWS CloudFormation template mapping section from:

```
AnonymousData:
    SendAnonymousData:
      Data: Yes
```

   to:

```
AnonymousData:
    SendAnonymousData:
      Data: No
```

4. Sign in to the AWS CloudFormation console.
5. Select **Create stack**.
6. On the **Create stack** page, **Specify template** section, select **Upload a template file**.
7. Under **Upload a template file**, choose **Choose file** and select the edited template from your local drive.
8. Choose **Next** and follow the steps in Launch the stack (p. 12) in the Automated Deployment section of this guide.

Source Code

This solution includes two source code options, NodeJS and Python. Use the **Source Code** AWS CloudFormation parameter to choose the source code you want. You can visit our GitHub repository to download the templates and scripts for this solution, and to share your customizations with others.

Document Revisions

Date	Change	
November 2016	Initial release	
March 2017	Added an AWS CloudFormation parameter for the Elemental Live server version.	
April 2017	Changed all Elemental references to AWS Elemental to reflect the new name.	
September 2018	Added information about AWS Elemental MediaLive and MediaPackage functionality, and encoding profiles.	
December 2018	Added information about the Amazon CloudFront distribution for the static website hosted in the Amazon S3 bucket.	
March 2019	Added information about MediaConnect input and CMAF format functionality.	
December 2019	Updated Cost information; updated the Lambda run-times to Node 12.x and Python 3.8.	
July 2020	Updated encoding settings and appendices with input configuration details. For a detailed description of the changes from version 2.3.0 to version 2.4.0, refer to the CHANGELOG.md file in the GitHub repository.	
June 2021	Documentation updates only: Expanded the cost estimates to provide more granular information, clarified the collection of operational metrics information, and updated guide organization to improve readability.	
August 2021	Release version 3.0.0: Updated with new features and bug fixes. For more information, refer to the CHANGELOG.md file in the GitHub repository.	
September 2021	Documentation updates only: Provided additional details about the demo preview player.	
November 2021	Release version 3.1.0: bug fixes. For more information, refer to the CHANGELOG.md file in the GitHub repository.	

Contributors

- Tom Nightingale
- Tom Gilman
- Joan Morgan
- Eddie Goynes
- Kiran Patel
- Aijun Peng

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents AWS current product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. AWS responsibilities and liabilities to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Live Streaming on AWS is licensed under the terms of the of the Apache License Version 2.0 available at The Apache Software Foundation.