# Table of Contents

What Is AWS Elemental MediaPackage? ................................................................. 1
Are You a First-Time User of AWS Elemental MediaPackage? ........................................... 1
Concepts and Terminology ................................................................................................. 1
How AWS Elemental MediaPackage Works ........................................................................ 2
Features of AWS Elemental MediaPackage ........................................................................ 3
Related Services ......................................................................................................................... 4
Accessing AWS Elemental MediaPackage .................................................................................. 4
Regions for AWS Elemental MediaPackage ............................................................................ 5

## Setting Up ......................................................................................................................... 6
Signing Up for AWS .................................................................................................................. 6
Creating an Admin IAM User ................................................................................................... 6
Creating a Non-Admin IAM User ............................................................................................. 7
  - Step 1: Create Policies ........................................................................................................ 7
  - Step 2: Create User Groups ................................................................................................ 8
  - Step 3: Create Users ........................................................................................................... 8
(Optional) Step 4: Get Set Up to Use Encryption .................................................................... 9

## Getting Started ................................................................................................................ 10
Prerequisites ............................................................................................................................ 10
  - Step 1: Access AWS Elemental MediaPackage ................................................................. 10
  - Step 2: Create a Channel .................................................................................................. 10
  - Step 3: Create Endpoints ................................................................................................ 11
(Optional) Step 4: Monitor AWS Elemental MediaPackage Activity .................................... 11
  - Step 5: Clean Up .............................................................................................................. 11

## Working with Channels .................................................................................................. 13
Creating a Channel .................................................................................................................. 13
Viewing Channel Details ......................................................................................................... 13
Editing a Channel ....................................................................................................................... 14
Deleting a Channel .................................................................................................................... 14
Adding an Endpoint to a Channel ............................................................................................ 15

## Working with Endpoints ................................................................................................ 16
Creating an Endpoint ................................................................................................................ 16
  - Creating an HLS Endpoint .................................................................................................. 16
  - Creating a Microsoft Smooth Streaming Endpoint ............................................................. 19
  - Creating a DASH Endpoint ................................................................................................. 22
Viewing All Endpoints Associated with a Channel ................................................................. 25
Viewing a Single Endpoint ...................................................................................................... 25
Editing an Endpoint ................................................................................................................ 25
Deleting an Endpoint ............................................................................................................... 26
Previewing an Endpoint .......................................................................................................... 26

## Time-shifted Viewing ..................................................................................................... 27
Rules for Start and End Parameters ....................................................................................... 27
  - DASH Parameter Rules ...................................................................................................... 28
  - HLS Parameter Rules ......................................................................................................... 28
  - Microsoft Smooth Parameter Rules .................................................................................. 29

## Rendition Groups ........................................................................................................... 30
When to Use Rendition Groups ............................................................................................. 30
When Not to Use Rendition Groups ....................................................................................... 30

## Monitoring ....................................................................................................................... 32
Monitoring with CloudWatch ............................................................................................... 32
  - AWS Elemental MediaPackage CloudWatch Metrics .......................................................... 33
  - AWS Elemental MediaPackage CloudWatch Dimensions .................................................. 36

## Limits ................................................................................................................................. 38
Soft Limits .............................................................................................................................. 38
Hard Limits ............................................................................................................................ 38
What Is AWS Elemental MediaPackage?

AWS Elemental MediaPackage is a just-in-time video packaging and origination service that runs in the AWS Cloud. With AWS Elemental MediaPackage, you can deliver highly secure, scalable, and reliable video streams to a wide variety of playback devices and content distribution networks (CDNs).

AWS Elemental MediaPackage offers a broadcast-grade viewing experience for viewers, while allowing you the flexibility to control and protect your content. Additionally, the built-in resiliency and scalability of AWS Elemental MediaPackage means that you have the right amount of resources at the right time, with no manual intervention required.

Topics
- Are You a First-Time User of AWS Elemental MediaPackage? (p. 1)
- Concepts and Terminology (p. 1)
- How AWS Elemental MediaPackage Works (p. 2)
- Features of AWS Elemental MediaPackage (p. 3)
- Related Services (p. 4)
- Accessing AWS Elemental MediaPackage (p. 4)
- Regions for AWS Elemental MediaPackage (p. 5)

Are You a First-Time User of AWS Elemental MediaPackage?

If you are a first-time user of AWS Elemental MediaPackage, we recommend that you begin by reading the following sections:

- How AWS Elemental MediaPackage Works (p. 2)
- Concepts and Terminology (p. 1)
- Features of AWS Elemental MediaPackage (p. 3)
- Getting Started with AWS Elemental MediaPackage (p. 10)

Concepts and Terminology

AWS Elemental MediaPackage includes the following components:

Channel

A channel represents the entry point for a content stream into AWS Elemental MediaPackage. Upstream encoders such as AWS Elemental MediaLive send content to the channel. When AWS Elemental MediaPackage receives a content stream, it packages the content and outputs the stream from an endpoint that you create on the channel. There is one channel for each incoming set of ABR streams.
Endpoint

An endpoint is part of a channel and represents the packaging aspect of AWS Elemental MediaPackage. When you create an endpoint on a channel, you indicate what streaming format, packaging parameters, and features the output stream will use. Downstream devices request content from the endpoint. A channel can have multiple endpoints.

Just-in-time packaging

AWS Elemental MediaPackage performs just-in-time packaging (JITP). When a playback device requests content, AWS Elemental MediaPackage dynamically customizes the live video streams and creates a manifest in a format that is compatible with the requesting device.

Origination service

AWS Elemental MediaPackage is considered an origination service because it is the point of distribution for media content delivery.

Packager

A packager prepares output streams for access by different types of players. The packager type specifies the streaming format that AWS Elemental MediaPackage delivers from the endpoint (either DASH-ISO, Microsoft Smooth Streaming, or Apple HLS). Additional packager settings include buffer and update durations and manifest tag handling instructions.

A packager is a part of an endpoint. Each endpoint must have one, and only one, packager. To use different packager types for the same content, create multiple endpoints on the channel.

Stream

A stream refers to the content input and output of AWS Elemental MediaPackage. An upstream encoder sends a live stream as an input to AWS Elemental MediaPackage to the channel. When a downstream device requests playback of the content, AWS Elemental MediaPackage dynamically packages the stream (including specifying the packager type, adding encryption, and configuring track outputs) and delivers it to the requesting device as an output of the endpoint. An endpoint can produce multiple streams.

Track

Tracks make up the output content stream. AWS Elemental MediaPackage includes selected video, audio, and subtitles or captions tracks in the output stream. The stream delivers the tracks to the player (either directly or through a CDN), and the player plays back the tracks based on player logic or network conditions (such as available bandwidth).

How AWS Elemental MediaPackage Works

AWS Elemental MediaPackage uses just-in-time format conversion to deliver over-the-top (OTT) video from a single source to a wide variety of playback devices or content delivery networks (CDNs).

Here is the general processing flow:

1. An upstream encoder (such as AWS Elemental MediaLive) sends an HLS live stream over WebDAV to the AWS Elemental MediaPackage channel ingest URL, and includes the channel's access credentials (as supplied in AWS Elemental MediaPackage). AWS scales resources up and down to handle the incoming traffic.

2. A downstream device requests content from AWS Elemental MediaPackage through the endpoint egress URL. A downstream device is either a video player or a content distribution network (CDN). The egress URL is associated with an endpoint for a specific streaming format (either Apple HLS, DASH-ISO, or Microsoft Smooth Streaming).

3. When AWS Elemental MediaPackage receives the playback request from the downstream device, it dynamically packages the stream according to the settings that you specified on the endpoint.
Packaging can include adding encryption and configuring audio, video, and subtitles or captions track outputs.

4. AWS Elemental MediaPackage delivers the output stream over HTTPS to the requesting device. As with ingest, AWS scales resources up and down to handle changes in traffic.

5. AWS Elemental MediaPackage logs activity through Amazon CloudWatch. You can view information like the number of content requests and ingress or egress bytes. For information about viewing AWS Elemental MediaPackage metrics in CloudWatch, see Monitoring AWS Elemental MediaPackage with Amazon CloudWatch (p. 32).

Throughout the ingest and egress processes, AWS Elemental MediaPackage detects and mitigates potential infrastructure failures before they become a problem for viewers.

The following illustration shows the overall process.

---

**Features of AWS Elemental MediaPackage**

AWS Elemental MediaPackage supports the following features:

**Audio**

AWS Elemental MediaPackage supports multi-language audio inputs, as well as the following audio codecs:
- AAC stereo
• Dolby AC3 and E-AC3 (Dolby Digital and Dolby Digital+)

AWS Elemental MediaPackage accepts these codecs from the input source and passes them through to the output stream.

Captions

AWS Elemental MediaPackage supports input 608/708 captions and passes them through to the output stream.

DRM

AWS Elemental MediaPackage supports content protection through digital rights management (DRM).

Subtitles

AWS Elemental MediaPackage supports input WebVTT text-based subtitles. AWS Elemental MediaPackage translates the subtitles to the appropriate format based on the packager that is used on the endpoint:
• For HLS: WebVTT is passed through
• For DASH: subtitles are translated to EBU-TT
• For Microsoft Smooth Streaming: subtitles are translated to DFXP

Time-shift Viewing

AWS Elemental MediaPackage allows playback of a stream at a time earlier than the current time. Start-over, catch-up TV, and time delay are all supported. For more information about setting up time-shift capabilities, see *Time-shifted Viewing* (p. 27).

Video

AWS Elemental MediaPackage supports the input H.264 video codec and passes it through to the output stream. H.265/HEVC is also supported with restricted playback devices.

Related Services

• **Amazon CloudFront** is a global content delivery network (CDN) service that securely delivers data and videos to your viewers. Use CloudFront to deliver content with the best possible performance. For more information, see *Amazon CloudFront*.

• **Amazon CloudWatch** is a monitoring service for AWS Cloud resources and the applications that you run on AWS. Use CloudWatch to track metrics such as ingress and egress request counts. For more information, see *Amazon CloudWatch*.

• **AWS Elemental MediaLive** is a live video processing service that encodes high-quality live video streams for broadcast television and multi-screen devices. Use AWS Elemental MediaLive to encode content streams and send them to AWS Elemental MediaPackage for packaging. For more information about how encoders (such as AWS Elemental MediaLive) work with AWS Elemental MediaPackage, see *How AWS Elemental MediaPackage Works* (p. 2).

• **AWS Identity and Access Management (IAM)** is a web service that helps you securely control access to AWS resources for your users. Use IAM to control who can use your AWS resources (authentication) and what resources users can use in which ways (authorization). For more information, see *Setting Up* (p. 6).

Accessing AWS Elemental MediaPackage

You can access AWS Elemental MediaPackage through the console, AWS Command Line Interface (AWS CLI), or AWS Elemental MediaPackage REST API.
Regions for AWS Elemental MediaPackage

AWS Elemental MediaPackage is available in the following regions:

- US West (Oregon)
- US East (N. Virginia)
- EU (Ireland)
- Asia Pacific (Singapore)
- Asia Pacific (Sydney)
Setting Up AWS Elemental MediaPackage

Before you use AWS Elemental MediaPackage for the first time, you must sign up for AWS if you don't already have an AWS account. Next, you must create AWS Identity and Access Management (IAM) users to allow access to AWS Elemental MediaPackage.

Topics
- Signing Up for AWS (p. 6)
- Creating an Admin IAM User (p. 6)
- Creating a Non-Admin IAM User (p. 7)
- (Optional) Get Set Up to Use Encryption (p. 9)

Signing Up for AWS

If you do not have an AWS account, use the following procedure to create one.

To sign up for AWS
2. Follow the online instructions.

Creating an Admin IAM User

An IAM user is an entity that you create in AWS to represent the person or service that uses it to interact with AWS. A user in AWS consists of a name and credentials. You should create an IAM user for yourself and add the IAM user to an Administrators group. This group gives you access to all services and their resources in your account. The next time that you access your AWS account, you should sign in with your IAM user credentials.

To create users with limited permissions, see Creating a Non-Admin IAM User (p. 7).

To create an IAM user for yourself and add the user to an Administrators group
1. Use your AWS account email address and password to sign in to the AWS Management Console as the AWS account root user.
2. In the navigation pane of the console, choose Users, and then choose Add user.
3. For User name, type Administrator.
4. Select the check box next to AWS Management Console access, select Custom password, and then type the new user's password in the text box. You can optionally select Require password reset to force the user to select a new password the next time the user signs in.
5. Choose Next: Permissions.
6. On the Set permissions for user page, choose Add user to group.
7. Choose Create group.
8. In the Create group dialog box, type Administrators.
For Filter, choose Job function.  
In the policy list, select the check box for AdministratorAccess. Then choose Create group.  
Back in the list of groups, select the check box for your new group. Choose Refresh if necessary to see the group in the list.  
Choose Next: Review to see the list of group memberships to be added to the new user. When you are ready to proceed, choose Create user.

You can use this same process to create more groups and users, and to give your users access to your AWS account resources. To learn about using policies to restrict users' permissions to specific AWS resources, go to Access Management and Example Policies.

For information about creating users with limited permissions, see Creating a Non-Admin IAM User (p. 7).

Creating a Non-Admin IAM User

Users in the Administrators group for an account have access to all AWS services and resources in that account. This section describes how to create users with permissions that are limited to AWS Elemental MediaPackage.

Step 1: Create Policies

Create two policies for AWS Elemental MediaPackage: one to provide read/write access, and one to provide read-only access. Perform these steps one time only for each policy.

To create policies for AWS Elemental MediaPackage

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.  
2. Use your Administrator user credentials to sign in to the IAM console.  
3. In the navigation pane of the console, choose Policies, and then choose Create policy.  
4. Choose the JSON tab and paste the following policy:

```json
{
   "Version": "2012-10-17",
   "Statement": {
      "Effect": "Allow",
      "Action": "mediapackage:*",
      "Resource": "*"
   }
}
```

This policy allows all actions on all resources in AWS Elemental MediaPackage.  
5. Choose Review policy.  
6. On the Review policy page, for Name, type MediaPackageAllAccess, and then choose Create policy.  
7. On the Policies page, repeat the steps in this section to create a read-only policy. Use the following policy and call it MediaPackageReadOnlyAccess:

```json
{
   "Version": "2012-10-17",
   "Statement": {
```
**Step 2: Create User Groups**

Create a user group for each of the policies that you created in step 1. This way, when you create additional users you can add the users to a group rather than attaching individual policies to each user.

**To create groups for users who need access to AWS Elemental MediaPackage**

1. In the navigation pane of the IAM console, choose **Groups**, and then choose **Create New Group**.
2. On the **Set Group Name** page, type a name for the group, such as **MediaPackageAdmins**. Choose **Next Step**.
3. On the **Attach Policy** page, for **Filter**, choose **Customer Managed**.
4. In the policy list, choose the **MediaPackageAllAccess** policy that you created.
5. Change **Filter** to **AWS Managed** and type **ReadOnlyAccess**. This policy allows read-only permissions for all AWS services except AWS Elemental MediaPackage. This access is necessary so that AWS Elemental MediaPackage can make calls to Amazon CloudWatch on the user's behalf.
6. In the updated policy list, choose the **ReadOnlyAccess** policy, and then choose **Next Step** to see the summary of the new group.
7. On the **Review** page, verify that the correct policies are added to this group, and then choose **Create Group**.
8. On the **Groups** page, repeat the steps in this section to create a user group that has read-only permissions. In step 4, choose **MediaPackageReadOnlyAccess**.

**Step 3: Create Users**

Create IAM users for the individuals who require access to AWS Elemental MediaPackage, and add each user to the appropriate user group to ensure that they have the right level of permissions. If you already have users created, skip to step 6 to modify the permissions for the users.

**To create users who can access AWS Elemental MediaPackage**

1. In the navigation pane of the IAM console, choose **Users**, and then choose **Add user**.
2. For **User name**, type the name that the user will use to sign in to AWS Elemental MediaPackage.
3. Select the check box next to **AWS Management Console access**, select **Custom password**, and then type the new user's password in the box. You can optionally select **Require password reset** to force the user to create a password the next time the user signs in.
4. Choose **Next: Permissions**.
5. On the **Set permissions for user** page, choose **Add user to group**.
6. In the group list, choose the group with the appropriate attached policy. Remember that permissions levels are as follows:
   - The group with the **MediaPackageAllAccess** policy allows all actions on all resources in AWS Elemental MediaPackage.
   - The group with the **MediaPackageReadOnlyAccess** policy allows read-only rights for all resources in AWS Elemental MediaPackage.
7. Choose **Next: Review** to see the list of group memberships to be added to the new user. When you are ready to proceed, choose **Create user**.

**(Optional) Get Set Up to Use Encryption**

You can protect your content from unauthorized use through encryption. Digital rights management systems provide keys to AWS Elemental MediaPackage for content encryption and licenses to supported players for decryption.

If you want to encrypt content, you must have a DRM solution provider. You can get set up by working with one of our AWS DRM solution providers. For information, see [http://docs.aws.amazon.com/speke/latest/documentation/customer-onboarding.html](http://docs.aws.amazon.com/speke/latest/documentation/customer-onboarding.html).
Getting Started with AWS Elemental MediaPackage

This Getting Started tutorial shows you how to use the AWS Elemental MediaPackage console to create a channel and endpoints for streaming live videos.

Topics
- Prerequisites (p. 10)
- Step 1: Access AWS Elemental MediaPackage (p. 10)
- Step 2: Create a Channel (p. 10)
- Step 3: Create Endpoints (p. 11)
- (Optional) Step 4: Monitor AWS Elemental MediaPackage Activity (p. 11)
- Step 5: Clean Up (p. 11)

Prerequisites

Before you can use AWS Elemental MediaPackage, you need an AWS account and the appropriate permissions to access, view, and edit AWS Elemental MediaPackage components. Complete the steps in Setting Up (p. 6), and then return to this tutorial.

Step 1: Access AWS Elemental MediaPackage

Using your IAM credentials, sign in to the AWS Elemental MediaPackage console:

https://<region>.console.aws.amazon.com/mediapackage/home

Step 2: Create a Channel

The channel is the first component in AWS Elemental MediaPackage. It represents the input to AWS Elemental MediaPackage for incoming content from an encoder such as AWS Elemental MediaLive.

To create a channel

1. On the AWS Elemental MediaPackage Channels page, choose Create channel.
2. For ID, type a name that describes the channel, such as channelHLS1. The ID is the primary identifier for the channel, and must be unique for your account in the region. Supported characters are letters, numbers, underscore (_), and dash (-). You cannot use spaces in the ID.
3. Keep the defaults for the remaining fields, and then choose Create channel.

AWS Elemental MediaPackage displays the new channel's details page.
4. On the channel's details page, note the values for Input URL, Username, and Password. AWS Elemental MediaPackage securely generates these values when it creates the channel. You can't change the values.
Provide the information from these fields to the person in charge of the upstream encoder. In the stream configuration in the encoder, this person must type the destination as the input URL, and the WebDAV credentials as the channel's user name and password. The upstream encoder must push WebDAV over HTTPS to AWS Elemental MediaPackage, and include these credentials.

**Step 3: Create Endpoints**

The endpoint is attached to a channel, and represents the output of the content. You can associate multiple endpoints to a single channel. Each endpoint gives players and downstream CDNs (such as Amazon CloudFront) access to the content for playback.

**To create an endpoint**

1. On the Channels page, choose the channel that the endpoint will be associated with.
2. On the details page for the channel, choose either Add and edit endpoint or Add endpoints if there are no existing endpoints.
3. For ID, type a name that describes the endpoint, such as HLSendpoint1. The ID is the primary identifier for the endpoint, and must be unique for your account in the region. Supported characters are letters, numbers, underscore (_), and dash (-). You cannot use spaces in the ID.
4. Keep the defaults for the remaining fields, and then choose Save endpoints.

AWS Elemental MediaPackage displays the channel's details page, including the endpoint that you just created.
5. On the channel's details page, note the value in the URL field for the endpoint. Provide this information to the person in charge of the downstream device (CDN or player). In the downstream device, this person must type the request destination as the endpoint's URL.

(Optional) **Step 4: Monitor AWS Elemental MediaPackage Activity**

Use Amazon CloudWatch to track AWS Elemental MediaPackage activity, such as the counts of ingest and egress bytes, response times, and request counts.

**To view metrics using the CloudWatch console**

Metrics are grouped first by the service namespace, and then by the various dimension combinations within each namespace.

2. In the navigation pane, choose Metrics.
3. Under All metrics, choose the AWS/MediaPackage namespace.
4. Select the metric dimension to view the metrics (for example, choose channel to view metrics per channel). For a list of AWS Elemental MediaPackage metrics, see AWS Elemental MediaPackage CloudWatch Metrics (p. 33).

**Step 5: Clean Up**

To avoid extraneous charges, be sure to delete all unnecessary channels and endpoints. You must delete all endpoints on a channel before the channel can be deleted.
To delete an endpoint

1. On the **Channels page**, choose the channel that the endpoint is associated with.
2. On the channel details page, choose the name of the endpoint to be deleted.
3. On the endpoint details page, choose **Delete endpoint**.
4. On the **Delete Endpoints** page, choose **Save all**.

To delete a channel

1. On the **Channels** page, choose the channel using one the following methods:
   - Choose the channel name
   - Select the check box next to the channel name
2. Choose **Delete selected** or **Delete channel**.
3. In the confirmation dialog box, choose **Delete**.

AWS Elemental MediaPackage removes the channel and all associated endpoints.
Creating a Channel

Create a channel to start receiving content streams. Later, you add an endpoint to the channel. This endpoint is the access point for content playback requests.

You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to create a channel.

For information about creating a channel through the AWS CLI or AWS Elemental MediaPackage API, see the AWS Elemental MediaPackage API Reference.

To create a channel (console)

2. On the Channels page, choose Create channel.
3. For ID, type a name that describes the channel. The ID is the primary identifier for the channel, and must be unique for your account in the region.
4. (Optional) For Description, type any descriptive text that helps you to identify the channel.
5. For Input type, choose Apple HLS.
6. Choose Create channel.

AWS Elemental MediaPackage displays the new channel's details page.

The channel is active and can start receiving content as soon as it's created. AWS Elemental MediaPackage scales resources up and down to allow the right amount of capacity for your traffic.
You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to view the channels that are associated with your AWS account.

For information about viewing channel details through the AWS CLI or AWS Elemental MediaPackage API, see the AWS Elemental MediaPackage API Reference.

To view channels (console)

2. If the Channels page doesn't appear, on the AWS Elemental MediaPackage home page, choose Skip and go to console.

All existing channels are displayed on the console.

3. (Optional) To adjust your viewing preferences (such as page size and properties that are displayed), choose Preferences.
4. To view more information about a specific channel, choose the name of the channel that you want to view.

AWS Elemental MediaPackage displays important information such as the values for Input URL and the WebDAV Username and Password. Provide this information for the upstream encoder stream destination settings.

Editing a Channel

Edit a channel's description for easier identification later. You can edit only the description on a channel.

You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to edit a channel's settings.

For information about editing a channel through the AWS CLI or AWS Elemental MediaPackage API, see the AWS Elemental MediaPackage API Reference.

To edit a channel (console)

2. If the Channels page doesn't appear, on the AWS Elemental MediaPackage home page, choose Skip and go to console.
3. On the Channels page, choose the name of the channel that you want to edit.
4. On the channel's details page, choose Edit channel.
5. Revise the description.
6. Choose Save changes.

Deleting a Channel

Delete a channel to stop AWS Elemental MediaPackage from receiving further content. You must delete the channel's endpoints (as described in Deleting an Endpoint (p. 26)) before you can delete the channel.

You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to delete a channel.

For information about deleting a channel using the AWS CLI or AWS Elemental MediaPackage API, see the AWS Elemental MediaPackage API Reference.
To delete a channel (console)

2. If the Channels page doesn't appear, on the AWS Elemental MediaPackage home page, choose Skip and go to console.
3. On the Channels page, choose the channel using one the following methods:
   - Choose the channel name
   - Select the check box next to the channel name
4. Choose Delete selected or Delete channel.
5. In the confirmation dialog box, choose Delete to proceed with the deletion.

Adding an Endpoint to a Channel

Add an endpoint to a channel to allow downstream video players and content distribution networks (CDNs) to start requesting content playback.

You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to add an endpoint to a channel.

For instructions on adding endpoints to a channel from the AWS Elemental MediaPackage console, see Working with Endpoints (p. 16).
Working with Endpoints in AWS Elemental MediaPackage

An endpoint defines a single delivery point of a channel. The endpoint holds all the information that is needed for AWS Elemental MediaPackage to integrate with a player or content distribution network (CDN) such as Amazon CloudFront. Configure the endpoint to output content in one of the available stream formats:

- Apple HLS – packages content to Apple HTTP Live Streaming (HLS)
- Microsoft Smooth – packages content for Microsoft Smooth Streaming players
- DASH-ISO – packages content for the DASH-ISO ABR streaming protocol

Additionally, the endpoint holds information about digital rights management (DRM) and encryption integration, stream bit rate presentation order, and more.

When you create an endpoint, it provides a public URL that is fixed for the lifetime of the endpoint, regardless of any failures or upgrades that might happen over time. This URL is how the player or CDN accesses the stream from the endpoint.

Topics
- Creating an Endpoint (p. 16)
- Viewing All Endpoints Associated with a Channel (p. 25)
- Viewing a Single Endpoint (p. 25)
- Editing an Endpoint (p. 25)
- Deleting an Endpoint (p. 26)
- Previewing an Endpoint (p. 26)

Creating an Endpoint

Create an endpoint on a channel to define how AWS Elemental MediaPackage prepares content for delivery. Content can’t be served from a channel until it has an endpoint.

You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to create an endpoint.

For information about creating an endpoint through the AWS CLI or AWS Elemental MediaPackage API, see the AWS Elemental MediaPackage API Reference.

Creating an HLS Endpoint

Create an endpoint that formats content for devices that support Apple HLS.

To create an Apple HLS endpoint (console)

1. Access the channel that the endpoint will be associated with, as described in Viewing Channel Details (p. 13).
2. On the details page for the channel, choose either Add and edit endpoint or Add endpoints if there are no existing endpoints.
3. Complete the fields as described in the following topics:
   - New Endpoint Fields (p. 17)
   - Packager Settings Fields (p. 17)
   - Encryption Fields (p. 18)
   - Access Control Fields (p. 19)
   - Streams to Include Fields (p. 19)

4. Choose **Save endpoints**.

   The new endpoint is displayed in the endpoints section of the channel's details page.

   The endpoint is active and can deliver content as soon as requests are sent to its URL endpoints. AWS Elemental MediaPackage scales resources up and down to allow the right amount of capacity for your traffic.

### New Endpoint Fields

1. For **ID**, type a name that describes the endpoint. The ID is the primary identifier for the endpoint, and must be unique for your account in the region.

2. (Optional) For **Description**, type any descriptive text that helps you to identify the endpoint later.

3. For **Manifest name**, type a short string that will be appended to the end of the endpoint URL. The manifest name helps to create a unique path to this endpoint.

4. (Optional) To allow time-shifted playback (start-over and catch-up TV), select **Startover window** and type the maximum amount of time that viewers can seek back on content. For more information about implementing start-over and catch-up TV, see Time-shifted Viewing (p. 27).

5. (Optional) To delay when content is available to players, type the duration (in seconds) for the delay in **Time delay**. The minimum time is five seconds. The maximum time is 86,400 seconds (24 hours).

   Use time delay to redefine the live point and make content available at a time that equals "now" minus the delay specified. With a 60-second time delay, content that AWS Elemental MediaPackage receives at 12:20 isn't available until 12:21. Requests for playback at 12:20 are served with content from 12:19. Likewise, if you're serving content across time zones, you can set a time delay equal to the time zone difference to make content available at, for example, 8:00 local time.

   When you use time delay in conjunction with a startover window, the time delay duration must be less than the startover window duration.

### Packager Settings Fields

1. For **Type**, choose **Apple HLS**.

2. (Optional) For **Segment duration**, type the duration (in seconds) of each segment. If the value that you type here is different from the input segment size, AWS Elemental MediaPackage rounds segments to the nearest multiple of the input segment duration.

3. (Optional) For **Playlist window duration**, type the total duration (in seconds) of the manifest.

4. (Optional) To group all audio tracks into a single HLS rendition group, select **Use audio rendition group**. For more information about rendition groups, see Rendition Groups Reference in AWS Elemental MediaPackage (p. 30).

5. (Optional) In stream sets with a single video track, to include an additional I-frame only stream along with the other tracks in the manifest, select **Include IFrame only stream**. AWS Elemental MediaPackage inserts EXT-I-FRAMES-ONLY tags in the manifest, and then compiles and includes an I-frames only playlist in the stream. This playlist enables player functionality like fast forward and rewind.
6. (Optional) To include EXT-X-PROGRAM-DATE-TIME tags in the output manifest, select **Program date/time interval**, and then type the interval for AWS Elemental MediaPackage to insert the tags in the manifest.

The EXT-X-PROGRAM-DATE-TIME tag synchronizes the stream to the wall clock, enabling functionality like viewer seek in the playback timeline and time display on the player.

7. (Optional) In **Ad markers**, choose how ad markers are included in the packaged content.

Choose from the following:
- **None** – Omit all SCTE-35 ad markers from the output.
- **SCTE-35 enhanced** – Generate ad markers and blackout tags based on the SCTE-35 input messages from the input source.
- **Passthrough** – Copy the SCTE-35 ad markers directly from the input HLS input manifest to the output manifest.

**Encryption Fields**

Protect your content from unauthorized use through encryption. Digital rights management (DRM) systems provide keys to AWS Elemental MediaPackage for content encryption, and licenses to supported players for decryption.

**Note**

To encrypt content, you must have a DRM solution provider. Get set up by working with one of our AWS DRM solution providers. For information, see [http://docs.aws.amazon.com/speke/latest/documentation/customer-on-boarding.html](http://docs.aws.amazon.com/speke/latest/documentation/customer-on-boarding.html).

1. To serve content without copyright protection, keep **No encryption** selected.
2. To serve content with copyright protection, choose **Encrypt content** and complete the additional fields as follows:
   a. **Resource ID** – Identifier that you define for the content, which is sent to the key server to identify the current endpoint. How unique you make this depends on how fine-grained you want access controls to be. The service does not allow you to use the same ID for two simultaneous encryption processes.

   The following example shows a resource ID:
   
   ```
   MovieNight20171126093045
   ```

   b. **System IDs** – Unique identifiers for your streaming protocol and DRM system. Provide up to two IDs for DASH and exactly one for the other streaming protocols. If you provide more than one system ID, enter them on separate lines, and do not separate them with commas or any other punctuation. For a list of common system IDs, see [DASH-IF System IDs](https://dashif.iana.org/systemids.html). If you do not know your IDs, ask your DRM solution provider.

   c. **URL** – The URL from the API Gateway proxy that you set up to talk to your key server.

   The following example shows a URL:
   
   ```
   https://1wm2dxlf33.execute-api.us-west-2.amazonaws.com/SpekeSample/copyProtection
   ```

   d. **Role ARN** – The Amazon Resource Name (ARN) of the IAM role that provides you access to send your requests through API Gateway. Get this from your DRM solution provider.

   The following example shows a role ARN:
   
   ```
   arn:aws:iam::012345678901:role/SpekeAccess
   ```
Creating a Microsoft Smooth Streaming Endpoint

Create an endpoint that formats content for devices that support Microsoft Smooth Streaming.

e. **Encryption method** – Choose **Sample-AES** for Apple HLS Fairplay or **AES-128** for Apple HLS AES-128.

f. **(Optional) Constant initialization vector** – A 128-bit, 16-byte hex value represented by a 32-character string that is used with the key for encrypting content.

g. **(Optional) Key rotation interval** – The frequency, in seconds, of key changes for live workflows, in which content is streamed real time. The service retrieves content keys before the live content begins streaming, and then retrieves them as needed over the lifetime of the workflow. By default, key rotation is set to 60 seconds, which is equivalent to setting it to 60. To disable key rotation, set this interval to 0 (zero).

The following example setting causes the service to rotate keys every thirty minutes:

| 1800 |

h. **(Optional) Repeat EXT-X-KEY** – Boolean that indicates whether to repeat the key before every segment of the manifest. By default, the key is written just once, after the header and before the segments. If you choose **Repeat EXT-X-KEY**, the manifest is written as header, key, segment, key, segment, key, and so on, with every segment preceded by the key. Choose this according to the needs of the player. Choosing this might result in an increase in client requests to the DRM server.

### Access Control Fields

1. To serve content to all requesting IP address, choose **Allow all incoming clients**.
2. To limit the IP addresses that this endpoint serves, choose **Restrict by IP address**.
3. In **Whitelist**, type the IP addresses that this endpoint serves content to.

### Streams to Include Fields

1. **(Optional) For Stream order**, choose from the following:
   - **Original** to sort the output streams in the same order that the incoming source uses.
   - **Ascending** to sort the output streams starting with the lowest bit rate and ending with the highest.
   - **Descending** to sort the output streams starting with the highest bit rate and ending with the lowest.

2. To make all incoming streams available for playback from this endpoint, select **Include all incoming streams**.
3. To limit which incoming streams are available for playback from this endpoint, select **Filter incoming streams** and enter filter criteria:
   - **(Optional) For Min video bitrate**, type the minimum bit rate threshold that video tracks must meet to be available for playback from this endpoint.
   - **(Optional) For Max video bitrate**, type the maximum bit rate that video tracks can have to be available for playback from this endpoint.

The minimum and maximum values take into account only the video bit rates. If the video bit rate is **below the minimum** specified rate, it is **not** included in the output, regardless of the sum of the bit rates for other tracks. Likewise, if the video bit rate is **below the maximum** specified rate, it is **included** in the output, regardless of the sum of the bit rates for other tracks.
To create a Microsoft Smooth endpoint (console)

1. Access the channel that the endpoint will be associated with, as described in Viewing Channel Details (p. 13).
2. On the details page for the channel, choose either Add and edit endpoint or Add endpoints if there are no existing endpoints.
3. Complete the fields as described in the following topics:
   - New Endpoint Fields (p. 20)
   - Packager Settings Fields (p. 20)
   - Encryption Fields (p. 21)
   - Access Control Fields (p. 21)
   - Streams to Include Fields (p. 21)
4. Choose Save endpoints.
   
The new endpoint is displayed in the endpoints section of the channel's details page.

New Endpoint Fields

1. For ID, type a name that describes the endpoint. The ID is the primary identifier for the endpoint, and must be unique for your account in the region.
2. (Optional) For Description, type any descriptive text that helps you to identify the endpoint later.
3. For Manifest name, type a short string that will be appended to the end of the endpoint URL. The manifest name helps to create a unique path to this endpoint.
4. (Optional) To create a window of the live stream that's available for on-demand viewing, select Startover window and type the size of the window (in seconds). Viewers can start-over or catch-up on content that falls within the window. For more information about implementing start-over and catch-up TV, see Time-shifted Viewing (p. 27).
5. (Optional) To delay when content is available to players, type the duration (in seconds) for the delay in Time delay. The minimum time is five seconds. The maximum time is 86,400 seconds (24 hours).

   Use time delay to redefine the live point and make content available at a time that equals "now" minus the delay specified. With a 60-second time delay, content that AWS Elemental MediaPackage receives at 12:20 isn't available until 12:21. Requests for playback at 12:20 will be served with content from 12:19. Likewise, if you're serving content across time zones, you can set a time delay equal to the time zone difference to make content available at, for example, 8:00 local time.

   When you use time delay in conjunction with a startover window, the time delay duration must be less than the startover window duration.

Packager Settings Fields

1. For Type, choose Microsoft Smooth.
2. (Optional) For Segment duration, type the duration (in seconds) of each segment. If the value that you type here is different from the input segment size, AWS Elemental MediaPackage rounds segments to the nearest multiple of the input segment duration.
3. (Optional) For Manifest window duration, type the total duration (in seconds) of the manifest.
Encryption Fields

Protect your content from unauthorized use through encryption. Digital rights management (DRM) systems provide keys to AWS Elemental MediaPackage for content encryption, and licenses to supported players for decryption.

**Note**
To encrypt content, you must have a DRM solution provider. Get set up by working with one of our AWS DRM solution providers. For information, see [http://docs.aws.amazon.com/speke/latest/documentation/customer-onboarding.html](http://docs.aws.amazon.com/speke/latest/documentation/customer-onboarding.html).

1. To serve content without copyright protection, keep **No encryption** selected.
2. To serve content with copyright protection, choose **Encrypt content** and complete the additional fields as follows:
   a. **Resource ID** – Identifier that you define for the content, which is sent to the key server to identify the current endpoint. How unique you make this depends on how fine-grained you want access controls to be. The service does not allow you to use the same ID for two simultaneous encryption processes.

   The following example shows a resource ID:
   
   MovieNight20171126093045
   
   b. **System IDs** – Unique identifiers for your streaming protocol and DRM system. Provide up to two IDs for DASH and exactly one for the other streaming protocols. If you provide more than one system ID, enter them on separate lines, and do not separate them with commas or any other punctuation. For a list of common system IDs, see [DASH-IF System IDs](http://docs.aws.amazon.com/speke/latest/documentation/customer-onboarding.html). If you do not know your IDs, ask your DRM solution provider.

   c. **URL** – The URL from the API Gateway proxy that you set up to talk to your key server.

   The following example shows a URL:
   
   https://1wm2dx1f33.execute-api.us-west-2.amazonaws.com/SpekeSample/copyProtection
   
   d. **Role ARN** – The Amazon Resource Name (ARN) of the IAM role that provides you access to send your requests through API Gateway. Get this from your DRM solution provider.

   The following example shows a role ARN:
   
   arn:aws:iam::012345678901:role/SpekeAccess

Access Control Fields

1. To serve content to all requesting IP address, choose **Allow all incoming clients**.
2. To limit the IP addresses that this endpoint serves, choose **Restrict by IP address**.
3. In **Whitelist**, type the IP addresses that this endpoint serves content to.

Streams to Include Fields

1. (Optional) For **Stream order**, choose from the following:
   - **Original** to sort the output streams in the same order that the incoming source uses.
   - **Ascending** to sort the output streams starting with the lowest bit rate and ending with the highest.
Creating a DASH Endpoint

Create an endpoint that formats content for devices that support MPEG-DASH.

To create an MPEG-DASH endpoint (console)

1. Access the channel that the endpoint will be associated with, as described in Viewing Channel Details (p. 13).
2. On the details page for the channel, choose either Add and edit endpoint or Add endpoints if there are no existing endpoints.
3. Complete the fields as described in the following topics:
   - New Endpoint Fields (p. 22)
   - Packager Settings Fields (p. 23)
   - Encryption Fields (p. 23)
   - Access Control Fields (p. 24)
   - Streams to Include Fields (p. 24)
4. Choose Save endpoints.

The new endpoint is displayed in the endpoints section of the channel's details page.

New Endpoint Fields

1. For ID, type a name that describes the endpoint. The ID is the primary identifier for the endpoint, and must be unique for your account in the region.
2. (Optional) For Description, type any descriptive text that helps you to identify the endpoint later.
3. For Manifest name, type a short string that will be appended to the end of the endpoint URL. The manifest name helps to create a unique path to this endpoint.
4. (Optional) To allow time-shifted playback (start-over and catch-up TV), select Startover window and type the maximum amount of time that viewers can seek back on content. For more information about implementing start-over and catch-up TV, see Time-shifted Viewing (p. 27).
5. (Optional) To delay when content is available to players, type the duration (in seconds) for the delay in Time delay. The minimum time is five seconds. The maximum time is 86,400 seconds (24 hours).
Use time delay to redefine the live point and make content available at a time that equals "now" minus the delay specified. With a 60-second time delay, content that AWS Elemental MediaPackage receives at 12:20 isn't available until 12:21. Requests for playback at 12:20 will be served with content from 12:19. Likewise, if you're serving content across time zones, you can set a time delay equal to the time zone difference to make content available at, for example, 8:00 local time.

When you use time delay in conjunction with a startover window, the time delay duration must be less than the startover window duration.

Packager Settings Fields

1. For **Type**, choose **DASH-ISO**.
2. (Optional) For **Segment duration**, type the duration (in seconds) of each segment. If the value that you type here is different from the input segment size, AWS Elemental MediaPackage rounds segments to the nearest multiple of the input segment duration.
3. (Optional) For **Manifest window duration**, type the total duration (in seconds) of the manifest.
4. (Optional) In **Profile**, specify a DASH profile, like HbbTV.

Choose from the following:
- **None** – the output doesn't use a DASH profile
- **Hbbtv 1.5** – the output is HbbTV-compliant

5. (Optional) For **Min update period**, type the minimum amount of time (in seconds) that the player should wait before requesting manifest updates. A lower value means that manifests are updated more frequently, but a lower value also contributes to request and response network traffic.

6. (Optional) For **Min buffer time**, type the minimum amount of time (in seconds) that a player must keep in the buffer. If network conditions interrupt playback, the player will have additional buffered content before playback fails, allowing for recovery time before the viewer's experience is affected.

7. (Optional) For **Suggested presentation delay**, enter the amount of time (in seconds) that the player should be from the end of the manifest. This sets the content start point back x seconds from the end of the manifest (the point where content is live). For example, with a 35-second presentation delay, requests at 5:30 receive content from 5:29:25. When used with time delay, AWS Elemental MediaPackage adds the suggested presentation delay to the time delay duration.

Encryption Fields

Protect your content from unauthorized use through encryption. Digital rights management (DRM) systems provide keys to AWS Elemental MediaPackage for content encryption, and licenses to supported players for decryption.

**Note**
To encrypt content, you must have a DRM solution provider. Get set up by working with one of our AWS DRM solution providers. For information, see [http://docs.aws.amazon.com/speke/latest/documentation/customer-onboarding.html](http://docs.aws.amazon.com/speke/latest/documentation/customer-onboarding.html).

1. To serve content without copyright protection, keep **No encryption** selected.
2. To serve content with copyright protection, choose **Encrypt content** and complete the additional fields as follows:
   a. **Resource ID** – Identifier that you define for the content, which is sent to the key server to identify the current endpoint. How unique you make this depends on how fine-grained you want access controls to be. The service does not allow you to use the same ID for two simultaneous encryption processes.

   The following example shows a resource ID:
b. **System IDs** – Unique identifiers for your streaming protocol and DRM system. Provide up to two IDs for DASH and exactly one for the other streaming protocols. If you provide more than one system ID, enter them on separate lines, and do not separate them with commas or any other punctuation. For a list of common system IDs, see [DASH-IF System IDs](#). If you do not know your IDs, ask your DRM solution provider.

c. **URL** – The URL from the API Gateway proxy that you set up to talk to your key server.

The following example shows a URL:

```
https://1wm2dx1f33.execute-api.us-west-2.amazonaws.com/SpekeSample/copyProtection
```

d. **Role ARN** – The Amazon Resource Name (ARN) of the IAM role that provides you access to send your requests through API Gateway. Get this from your DRM solution provider.

The following example shows a role ARN:

```
arn:aws:iam::012345678901:role/SpekeAccess
```

e. **(Optional) Key rotation interval** – The frequency, in seconds, of key changes for live workflows, in which content is streamed real time. The service retrieves content keys before the live content begins streaming, and then retrieves them as needed over the lifetime of the workflow. By default, key rotation is set to 60 seconds, which is equivalent to setting it to 60. To disable key rotation, set this interval to 0 (zero).

The following example setting causes the service to rotate keys every thirty minutes:

```
1800
```

### Access Control Fields

1. To serve content to all requesting IP address, choose **Allow all incoming clients**.
2. To limit the IP addresses that this endpoint serves, choose **Restrict by IP address**.
3. In **Whitelist**, type the IP addresses that this endpoint serves content to.

### Streams to Include Fields

1. **(Optional) For Stream order**, choose from the following:
   - **Original** to sort the output streams in the same order that the incoming source uses.
   - **Ascending** to sort the output streams starting with the lowest bit rate and ending with the highest.
   - **Descending** to sort the output streams starting with the highest bit rate and ending with the lowest.
2. To make all incoming streams available for playback from this endpoint, select **Include all incoming streams**.
3. To limit which incoming streams are available for playback from this endpoint, select **Filter incoming streams** and enter filter criteria:
   - **(Optional) For Min video bitrate**, type the minimum bit rate threshold that video tracks must meet to be available for playback from this endpoint.
   - **(Optional) For Max video bitrate**, type the maximum bit rate that video tracks can have to be available for playback from this endpoint.
The minimum and maximum values take into account only the video bit rates. If the video bit rate is below the minimum specified rate, it is not included in the output, regardless of the sum of the bit rates for other tracks. Likewise, if the video bit rate is below the maximum specified rate, it is included in the output, regardless of the sum of the bit rates for other tracks.

Viewing All Endpoints Associated with a Channel

View all endpoints that are associated with a specific channel to ensure that the content is available in all necessary stream formats.

You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to view the endpoints that are associated with a channel.

For information about viewing endpoints through the AWS CLI or AWS Elemental MediaPackage API, see the AWS Elemental MediaPackage API Reference.

To view a channel's endpoints (console)

1. Access the channel that the endpoint is associated to, as described in Viewing Channel Details (p. 13).
   
   AWS Elemental MediaPackage displays all existing endpoints as a table or as individual cards.

2. (Optional) To adjust your viewing preferences (such as page size and properties that are displayed), choose Preferences.

Viewing a Single Endpoint

View the details about a specific endpoint to obtain its playback URL and to view the packaging settings that it is currently using.

You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to view the details of an endpoint.

For information about viewing endpoint details through the AWS CLI or AWS Elemental MediaPackage API, see the AWS Elemental MediaPackage API Reference.

To view a single endpoint's details (console)

1. Access the channel that the endpoint is associated with, as described in Viewing Channel Details (p. 13).

2. On the channel's details page, choose the endpoint name to view details such as package information and playback preview. For downstream device requests, you must provide the endpoint URL in the Endpoint URL field.

Editing an Endpoint

Edit the packaging preferences on an endpoint to optimize the viewing experience. You can’t change the packager type after you save an endpoint. To serve content with a different packager, create a different endpoint.

You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to change an endpoint's settings.
For information about editing an endpoint through the AWS CLI or AWS Elemental MediaPackage API, see the AWS Elemental MediaPackage API Reference.

**To edit an endpoint (console)**

1. Access the channel that the endpoint is associated with, as described in Viewing Channel Details (p. 13).
2. On the channel's details page, do one of the following:
   - Choose **Add and edit endpoints** and on the **Edit** page, choose the endpoint to edit.
   - Choose the name of the endpoint to edit, and then choose **Edit endpoint**.
3. Edit the endpoint options that you want to change.
   
   For information about endpoint attributes, see Creating an Endpoint (p. 16).
4. Choose **Save all**.

### Deleting an Endpoint

Endpoints can serve content until they are deleted. Delete the endpoint if it should no longer respond to playback requests. You must delete all endpoints from a channel before you can delete the channel.

**Warning**

If you delete an endpoint, the playback URL stops working.

You can use the AWS Elemental MediaPackage console, the AWS CLI, or the AWS Elemental MediaPackage API to delete an endpoint.

For information about deleting an endpoint through the AWS CLI or AWS Elemental MediaPackage API, see the AWS Elemental MediaPackage API Reference.

**To delete an endpoint (console)**

1. Access the channel that the endpoint is associated with, as described in Viewing Channel Details (p. 13).
2. On the channel's details page, choose the endpoint name.
3. On the endpoint details page, choose **Delete endpoint**.
4. On the **Delete Endpoints** page, choose **Save all**.

### Previewing an Endpoint

Preview an endpoint's playback to ensure that AWS Elemental MediaPackage is receiving the content stream and can package it. The preview is helpful for avoiding playback failures after the endpoint is published, as well as for troubleshooting later if there are any playback issues.

You can use the AWS Elemental MediaPackage console to preview playback from the endpoint.

**To preview an endpoint's playback (console)**

1. Access the channel that the endpoint is associated with, as described in Viewing Channel Details (p. 13).
2. On the channel's details page, locate the endpoint name.
3. On the endpoint details page, do one of the following:
   - Choose **Play** to play content with the embedded player.
   - Choose **QR code** to view and scan the QR code for playback on a compatible device.
Time-shifted Viewing Reference in AWS Elemental MediaPackage

Time-shifted viewing means that viewers can start watching a live stream at a time earlier than "now," allowing them to join from the beginning a show that's already in progress or to watch a show that's already completed. AWS Elemental MediaPackage allows a content retention window of up to 72 hours for time-shifted viewing. Time-shifted functionality is controlled by the AWS Elemental MediaPackage endpoint and by the start and end parameters provided in the content request URL.

To enable time-shifted viewing

1. Enable time-shifted viewing by typing a value for Startover time on the AWS Elemental MediaPackage endpoint object. You can do this through either the AWS Elemental MediaPackage console or the REST API.

When requests with start and end parameters are sent to this endpoint, AWS Elemental MediaPackage generates a manifest within the window that is indicated in the request. If no start and end parameters are used, the service generates a standard manifest.

Note
You might notice that the manifest lags behind real time when you initially create a startover window on an endpoint. This is because AWS Elemental MediaPackage starts filling the manifest from the start of the window, and works up to "now." So if you have a 24-hour startover window, AWS Elemental MediaPackage fills the manifest starting 24 hours ago and working up to "now."

2. Ensure that content requests contain start and end parameters as needed. AWS Elemental MediaPackage accepts requests for up to six hours of content.

For packager-specific rules about how you can notate the parameters, see Rules for Start and End Parameters (p. 27).

The start and end parameters determine the time boundaries of the manifest. Expected behaviors are as follows:

- If both start and end parameters are used in the URL, the resulting manifest has a fixed start and end point that correspond to the specified start and end parameters.
- If a start parameter is specified but not an end, the resulting manifest has a fixed start point that corresponds to the specified start parameter, and the end of the manifest grows as the live content progresses.
- If an end parameter is specified but no start, the resulting manifest starts "now" and has a fixed endpoint that corresponds to the specified end parameter.
- If no parameters are specified, a standard manifest is generated starting "now" with no endpoint.

Rules for Start and End Parameters

Start and end parameters denote the beginning and end of a time-shifted manifest. The playback device can append parameters to the end of a manifest request or include the parameters within the request.

In all cases, the date and time must be notated in one of the following formats:
• ISO 8601 dates, such as 2017-08-18T21:18:54+00:00
• POSIX (or Epoch) time, such as 1503091134

The following topics describe the location rules by packager type.

Topics
  • DASH Parameter Rules (p. 28)
  • HLS Parameter Rules (p. 28)
  • Microsoft Smooth Parameter Rules (p. 29)

DASH Parameter Rules

Start and end parameters in the URL request for DASH content can use standard parameter notation, or can be included as path elements in the URL.

• Query parameter notation – start and end parameters are included at the end of the request URL

  Example

  https://cf98fa7b2ee4450e.mediapackage.us-east-1.amazonaws.com/out/
  v1/997cbb27697d4863bb65488133bbf26f/sports.mpd?
  start=1513717228&end=1513720828

• Path elements – start and end parameters are included in the path of the request URL

  Example

  https://cf98fa7b2ee4450e.mediapackage.us-east-1.amazonaws.com/
  out/start/2017-12-19T13:00:28-08:00/end/2017-12-19T14:00:28-08:00/
  v1/997cbb27697d4863bb65488133bbf26f/sports.mpd

HLS Parameter Rules

Start and end parameters in the URL request for HLS content can use standard parameter notation, or can be included as path elements in the URL.

• Query parameter notation – start and end parameters are included at the end of the request URL

  Example

  https://cf98fa7b2ee4450e.mediapackage.us-east-1.amazonaws.com/out/
  v1/064134724fd74667ba294657a674ae72/comedy.m3u8?
  start=2017-12-19T13:00:28-08:00&end=2017-12-19T14:00:28-08:00

• Path elements – start and end parameters are included in the path of the request URL

  Example

  https://cf98fa7b2ee4450e.mediapackage.us-east-1.amazonaws.com/out/
  start/2017-12-19T13:00:28-08:00/end/2017-12-19T14:00:28-08:00/
  v1/064134724fd74667ba294657a674ae72/comedy.m3u8
Microsoft Smooth Parameter Rules

Start and end parameters in the URL request for Microsoft Smooth Streaming content can be included as path elements in the URL.

- Path elements – start and end parameters are included in the path of the request URL

**Example**

https://cf98fa7b2ee4450e.mediapackage.us-east-1.amazonaws.com/out/start/1513717228/end/1513720828/v1/1f76b3b4f94c44a485c0e4e560afe50e/drama.ism/Manifest
Rendition Groups Reference in AWS Elemental MediaPackage

Rendition groups are used in HLS outputs. A rendition group collects all subtitle or audio tracks and makes them available for all video renditions in the stream. When you enable rendition groups, AWS Elemental MediaPackage pulls together all audio variants (such as different languages or codecs) and groups them for use with any video rendition. AWS Elemental MediaPackage automatically puts subtitles into a rendition group.

The following sections further describe when you can use rendition groups.

Note
DASH and Microsoft Smooth do not use rendition groups. This is because all audio, video, and subtitle or caption tracks are presented to the player, and the player determines which are used during playback.

Topics
- When to Use Rendition Groups (p. 30)
- When Not to Use Rendition Groups (p. 30)

When to Use Rendition Groups

Rendition groups are used only in HLS outputs. Rendition groups are most beneficial when you have multiple languages or multiple audio codecs in your streams. Rendition groups should be used in the following use cases:

- One or more video tracks with multiple audio languages or codecs

  When rendition groups are enabled, AWS Elemental MediaPackage pulls all audio renditions together for shared use between the video tracks. In this way, you don't have to duplicate all the audio options across all the video tracks.

- Multiple audio-only tracks and multiple subtitle tracks

  When both the audio tracks and subtitle tracks are in rendition groups, all the audio options can be combined with any subtitle track.

- One audio-only track and multiple subtitle tracks

  AWS Elemental MediaPackage automatically pulls subtitle tracks into a rendition group so that the audio track can be used with any subtitle. Because there is only one audio and the subtitles are already grouped, you don't need to tell AWS Elemental MediaPackage to use rendition groups in this case.

When Not to Use Rendition Groups

Rendition groups can't or shouldn't be used in the following use cases:

- Multiple video tracks in the stream, but only one language or codec is used for the audio. If the same audio is used with multiple audio tracks, and rendition groups are also used, then your rendition group will have duplicates of the same audio track (one for each video).
Keep the audio and video muxed in the stream, and do not use a rendition group.

- DASH or Microsoft Smooth Streaming outputs. These protocols do not support rendition groups. Instead, the output stream includes all tracks, and the player determines which to play based on rules from the player side or from the manifest (such as language or bit rate selection).

To limit the tracks available to a player, use the stream selection options from the AWS Elemental MediaPackage console or the REST API.
Monitoring AWS Elemental MediaPackage

Monitoring is an important part of maintaining the reliability, availability, and performance of AWS Elemental MediaPackage and your other AWS solutions. AWS provides the following monitoring tools to watch AWS Elemental MediaPackage, report when something is wrong, and take automatic actions when appropriate:

- **Amazon CloudWatch** monitors your AWS resources and the applications that you run on AWS in real time. You can collect and track metrics, create customized dashboards, and set alarms that notify you or take actions when a specified metric reaches a threshold that you specify. For example, you can have CloudWatch track CPU usage or other metrics of your Amazon EC2 instances and automatically launch new instances when needed. For more information, see the Amazon CloudWatch User Guide.

**Topics**
- Monitoring AWS Elemental MediaPackage with Amazon CloudWatch (p. 32)

Monitoring AWS Elemental MediaPackage with Amazon CloudWatch

You can monitor AWS Elemental MediaPackage using CloudWatch, which collects raw data and processes it into readable, near real-time metrics. These statistics are kept for 15 months, so that you can access historical information and gain a better perspective on how your web application or service is performing. You can also set alarms that watch for certain thresholds, and send notifications or take actions when those thresholds are met. For more information, see the Amazon CloudWatch User Guide.

**To view metrics using the AWS Elemental MediaPackage console**
AWS Elemental MediaPackage displays metrics throughout the console.

2. Navigate to the appropriate page to view metrics:
   - For metrics on all channels and endpoints in the region, go to the **Channels** page.
   - For metrics on a specific channel and all of its endpoints, go to the channel's details page.
   - For metrics on a specific endpoint and its channel, go to the endpoint's details page.
3. (Optional) To refine the metrics view, choose **Open in CloudWatch**.

**To view metrics using the CloudWatch console**
Metrics are grouped first by the service namespace, and then by the various dimension combinations within each namespace.

1. Sign in to the AWS Management Console and open the CloudWatch console at https://console.aws.amazon.com/cloudwatch/.
2. In the navigation pane, choose **Metrics**.
3. Under **All metrics**, choose the **AWS/MediaPackage** namespace.
4. Select the metric dimension to view the metrics (for example, choose \texttt{channel} to view metrics per channel).

\textbf{To view metrics using the AWS CLI}

At a command prompt, use the following command:

\begin{verbatim}
aws cloudwatch list-metrics --namespace "AWS/MediaPackage"
\end{verbatim}

\section*{AWS Elemental MediaPackage CloudWatch Metrics}

The AWS/MediaPackage namespace includes the following metrics. AWS Elemental MediaPackage publishes metrics to CloudWatch every minute, if not sooner.

\begin{table}[h]
\centering
\begin{tabular}{|l|p{0.7\textwidth}|}
\hline
\textbf{Metric} & \textbf{Description} \\
\hline
\textbf{EgressBytes} & Number of bytes that AWS Elemental MediaPackage has successfully output for each request. If AWS Elemental MediaPackage has not received any requests for egress in the specified interval, then no data is given. Units: Bytes  \\
& Valid statistics:  \\
& \hspace{1cm} \textbf{Average} – average bytes (\texttt{Sum}/\texttt{SampleCount}) that AWS Elemental MediaPackage has output over the configured interval.  \\
& \hspace{1cm} \textbf{Maximum} – largest individual output request (in bytes) made to AWS Elemental MediaPackage.  \\
& \hspace{1cm} \textbf{Minimum} – smallest individual output request (in bytes) made to AWS Elemental MediaPackage.  \\
& \hspace{1cm} \texttt{SampleCount} – number of requests used in the statistical calculation.  \\
& \hspace{1cm} \texttt{Sum} – total number of bytes that AWS Elemental MediaPackage has output over the configured interval.  \\
& Valid dimensions:  \\
& \hspace{1cm} \texttt{Channel}  \\
& \hspace{1cm} \texttt{Combination Channel and OriginEndpoint}  \\
& \hspace{1cm} \texttt{No dimension}  \\
\hline
\textbf{EgressRequestCount} & Number of content requests that AWS Elemental MediaPackage has received. If AWS Elemental MediaPackage has not received any requests for egress in the specified interval, then no data is given. Units: Count \\
\hline
\end{tabular}
\end{table}
### AWS Elemental MediaPackage CloudWatch Metrics

<table>
<thead>
<tr>
<th>Metric</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sum</td>
<td>Total number of egress requests that AWS Elemental MediaPackage has received.</td>
</tr>
<tr>
<td><strong>Valid dimensions:</strong></td>
<td>Channel, Combination Channel and OriginEndpoint, StatusCodeRange, Combination Channel and StatusCodeRange, Combination Channel, OriginEndpoint, and StatusCodeRange, No dimension</td>
</tr>
<tr>
<td>EgressResponseTime</td>
<td>The time that it took AWS Elemental MediaPackage to process each egress request. If AWS Elemental MediaPackage has not received any requests for egress in the specified interval, then no data is given. Units: Milliseconds</td>
</tr>
<tr>
<td><strong>Valid statistics:</strong></td>
<td>Average, Maximum, Minimum, SampleCount, Sum</td>
</tr>
<tr>
<td><strong>Valid dimensions:</strong></td>
<td>Channel, Combination Channel and OriginEndpoint</td>
</tr>
<tr>
<td>Metric</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>IngressBytes</td>
<td>Number of bytes that AWS Elemental MediaPackage has ingested for each request. If AWS Elemental MediaPackage has not received any requests for egress in the specified interval, then no data is given. Units: Bytes</td>
</tr>
</tbody>
</table>

Valid statistics:
- Average – average bytes (Sum/SampleCount) that AWS Elemental MediaPackage has ingested over the configured interval.
- Maximum – largest individual ingest request (in bytes) made to AWS Elemental MediaPackage.
- Minimum – smallest individual ingest request (in bytes) made to AWS Elemental MediaPackage.
- SampleCount – number of requests used in the statistical calculation.
- Sum – total number of bytes that AWS Elemental MediaPackage has ingested over the configured interval.

Valid dimensions:
- Channel
- No dimension
### AWS Elemental MediaPackage User Guide

### AWS Elemental MediaPackage CloudWatch Dimensions

**Metric** | **Description**
--- | ---
 IngressResponseTime | The time that it took AWS Elemental MediaPackage to process each ingest request. If AWS Elemental MediaPackage has not received any requests for egress in the specified interval, then no data is given. Units: Milliseconds

Valid statistics:
- **Average** – average amount of time (\(\frac{\text{Sum}}{\text{SampleCount}}\)) that it took AWS Elemental MediaPackage to process ingest requests over the configured interval.
- **Maximum** – longest amount of time (in milliseconds) that it took AWS Elemental MediaPackage to process an ingest request and provide a response.
- **Minimum** – shortest amount of time (in milliseconds) that it took AWS Elemental MediaPackage to process an ingest request and provide a response.
- **SampleCount** – number of requests used in the statistical calculation.
- **Sum** – total amount of time that it took AWS Elemental MediaPackage to process ingest requests over the configured interval.

Valid dimensions:
- Channel
- No dimension

### AWS Elemental MediaPackage CloudWatch Dimensions

You can filter the AWS/MediaPackage data using the following dimensions.

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No Dimension</td>
<td>Metrics are aggregated and shown for all channels, endpoints, or status codes.</td>
</tr>
<tr>
<td>Channel</td>
<td>Metrics are shown only for the specified channel. Value: The auto-generated GUID of the channel. Can be used alone or with other dimensions:</td>
</tr>
<tr>
<td></td>
<td>• Alone to show metrics for only the specified channel.</td>
</tr>
<tr>
<td><strong>Dimension</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| OriginEndpoint   | Metrics are shown for the specified channel and endpoint combination.  
                  | Value: The auto-generated GUID of the endpoint.  
                  | Must be used in conjunction with the channel dimension. |
| StatusCodeRange  | Metrics are shown for the specified status code range.  
                  | Value: 2xx, 3xx, 4xx, or 5xx.  
                  | Can be used alone or with other dimensions:  
                  |   • Alone to show all egress requests for the specified status range.  
                  |   • With the channel dimension to show egress requests for all endpoints that are associated with the specified channel, with the specified status code range.  
                  |   • With the channel and originEndpoint dimensions to show egress requests with a specific status code range on the specified endpoint that is associated with the specified channel. |
Limits in AWS Elemental MediaPackage

The following sections provide information about the limits in AWS Elemental MediaPackage. For information about requesting an increase to soft limits, see AWS Service Limits. Hard limits cannot be changed.

**Soft Limits**

The following table describes limits in AWS Elemental MediaPackage that can be increased. For information about changing limits, see AWS Service Limits.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Default Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum Channels</td>
<td>10</td>
</tr>
<tr>
<td>Maximum Endpoints per Channel</td>
<td>10</td>
</tr>
</tbody>
</table>

**Hard Limits**

The following table describes limits within AWS Elemental MediaPackage that can't be increased.

<table>
<thead>
<tr>
<th>Resource or Operation</th>
<th>Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ingest Stream Limits</td>
<td>20 streams per channel, and 10 tracks per stream</td>
</tr>
<tr>
<td>Maximum Content Retention</td>
<td>72 hours</td>
</tr>
<tr>
<td>Maximum Live Manifest Length</td>
<td>5 minutes</td>
</tr>
<tr>
<td>Maximum Time-shifted Manifest Length</td>
<td>6 hours</td>
</tr>
<tr>
<td>Request Rates per Channel</td>
<td>• Ingest: 50 requests per second</td>
</tr>
<tr>
<td></td>
<td>• Egress: 200 requests per second</td>
</tr>
<tr>
<td>REST API Requests</td>
<td>• Steady state: 5 requests per second</td>
</tr>
<tr>
<td></td>
<td>• Bursting: 50 requests per second</td>
</tr>
</tbody>
</table>
AWS Elemental MediaPackage

Resources

The following table lists related resources that you'll find useful as you work with AWS Elemental MediaPackage.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Classes and Workshops</td>
<td>Links to role-based and specialty courses as well as self-paced labs to help sharpen your AWS skills and gain practical experience.</td>
</tr>
<tr>
<td>AWS Developer Tools</td>
<td>Links to developer tools, SDKs, IDE tool kits, and command line tools for developing and managing AWS applications.</td>
</tr>
<tr>
<td>AWS Whitepapers</td>
<td>Links to a comprehensive list of technical AWS whitepapers, covering topics such as architecture, security, and economics and authored by AWS Solutions Architects or other technical experts.</td>
</tr>
<tr>
<td>AWS Support Center</td>
<td>The hub for creating and managing your AWS Support cases. Also includes links to other helpful resources, such as forums, technical FAQs, service health status, and AWS Trusted Advisor.</td>
</tr>
<tr>
<td>AWS Support</td>
<td>The primary web page for information about AWS Support, a one-on-one, fast-response support channel to help you build and run applications in the cloud.</td>
</tr>
<tr>
<td>Contact Us</td>
<td>A central contact point for inquiries concerning AWS billing, account, events, abuse, and other issues.</td>
</tr>
<tr>
<td>AWS Site Terms</td>
<td>Detailed information about our copyright and trademark; your account, license, and site access; and other topics.</td>
</tr>
</tbody>
</table>
Document History for User Guide

The following table describes the documentation for this release of AWS Elemental MediaPackage.

- **API version:** 1.0
- **Latest documentation update:** December 20, 2017

<table>
<thead>
<tr>
<th>Change</th>
<th>Description</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Corrected links and added whitelisting.</td>
<td>Corrected links to the AWS Elemental MediaPackage console and AWS Elemental MediaPackage API Reference. In Working with Endpoints, added reference to access control fields.</td>
<td>December 1, 2017</td>
</tr>
<tr>
<td>Added IAM policy information specific to AWS Elemental MediaPackage.</td>
<td>In Setting Up AWS Elemental MediaPackage (p. 6), added instructions for creating non-admin roles with limited permissions.</td>
<td>December 13, 2017</td>
</tr>
<tr>
<td>Added hard limit information.</td>
<td>In Limits in AWS Elemental MediaPackage (p. 38), added information about limits that can’t be changed (hard limits).</td>
<td>December 20, 2017</td>
</tr>
</tbody>
</table>

**Note**

- The AWS Media Services are not designed or intended for use with applications or in situations requiring fail-safe performance, such as life safety operations, navigation or communication systems, air traffic control, or life support machines in which the unavailability, interruption or failure of the services could lead to death, personal injury, property damage or environmental damage.
AWS Glossary

For the latest AWS terminology, see the AWS Glossary in the AWS General Reference.