EZDRM Configuration
AWS MediaLive and MediaPackage

Table of Contents

Version 1

Prerequisites

Installation of AWS Command Line Interface (CLI) pip install is required prior to configuration. Python 3.6 or higher is required.

For more information on requirements set up, visit this link in a browser: https://docs.aws.amazon.com/cli/latest/userguide/installing.html

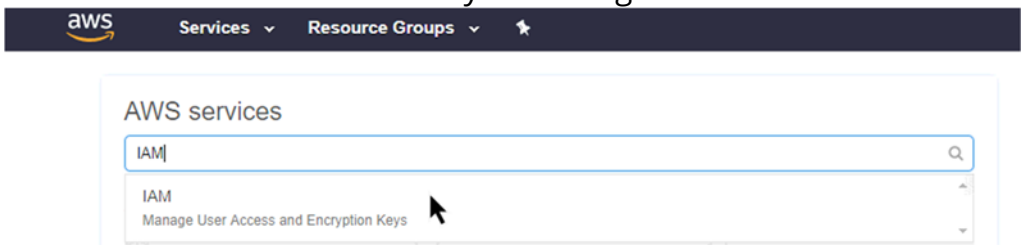
To download Python 3.6: https://www.python.org/downloads/

EZDRM AWS Speke Server Deployment

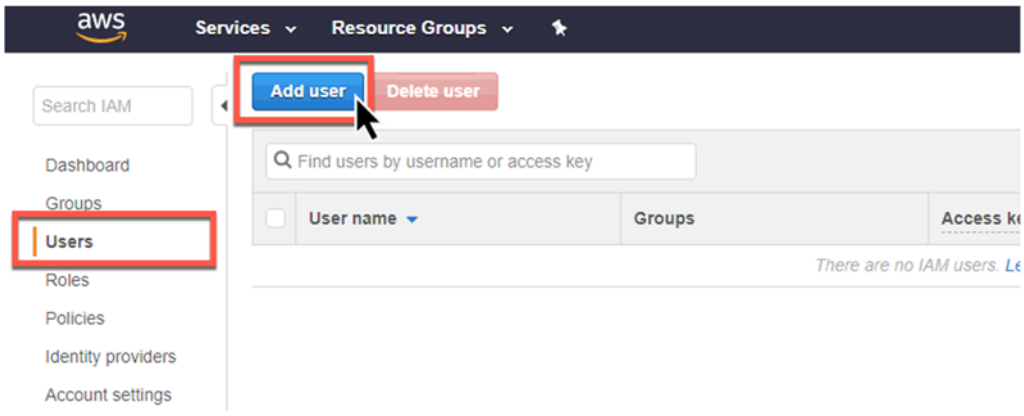
Step 1: Create a New User

To create a new User in AWS complete the following steps:

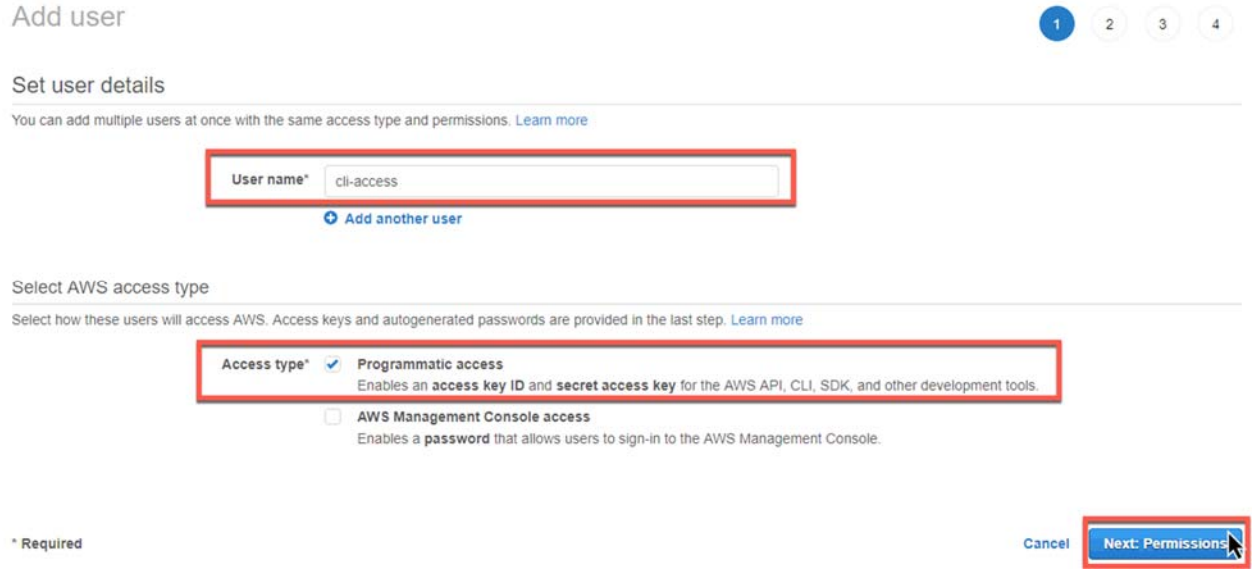
1. Launch the AWS IAM console by searching for IAM.



2. Go to the Users menu. Click the **Add user** button.

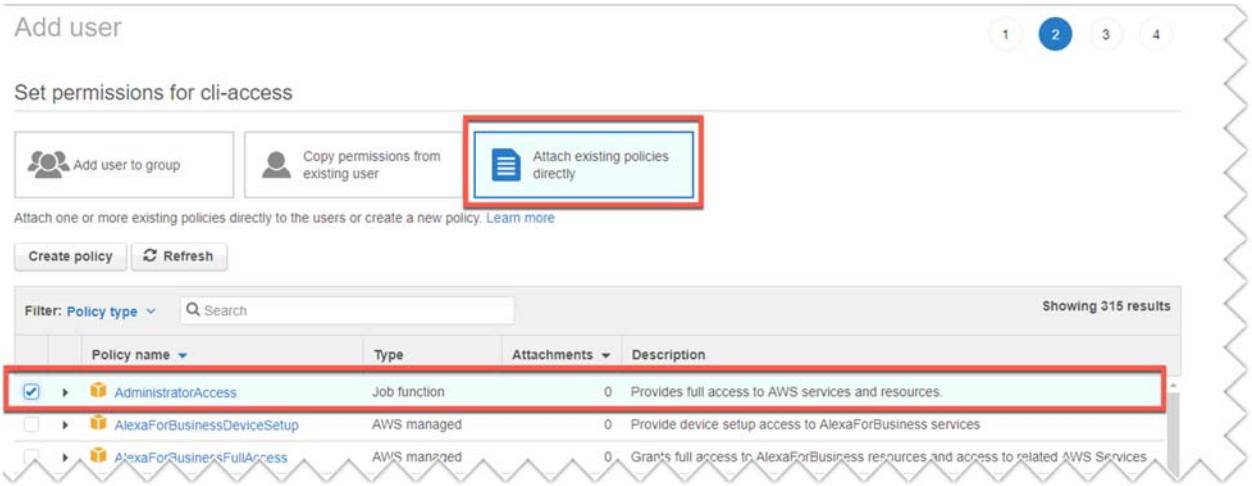


3. Enter a **User name**, we suggest “cli-access”. Then under the “Select Access Type” section, click the checkbox to enable **Programmatic access**. This enables an access key ID and secret access key for the AWS CLI.



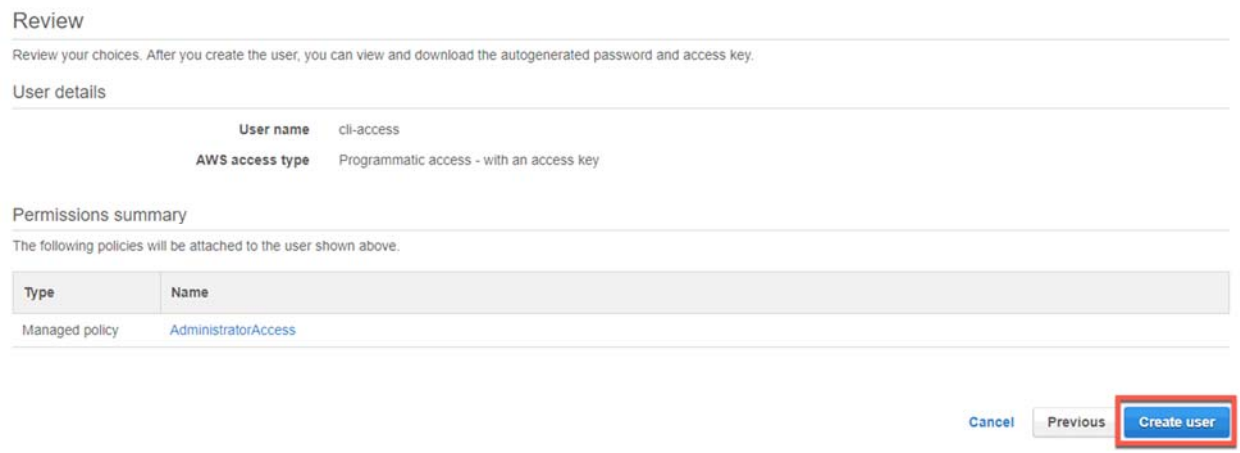
4. Click the **Next: Permissions** button.

5. Select the **Attach existing policies directly** menu and click the checkbox to select **AdministratorAccess**. This provides full access to AWS services and resources.

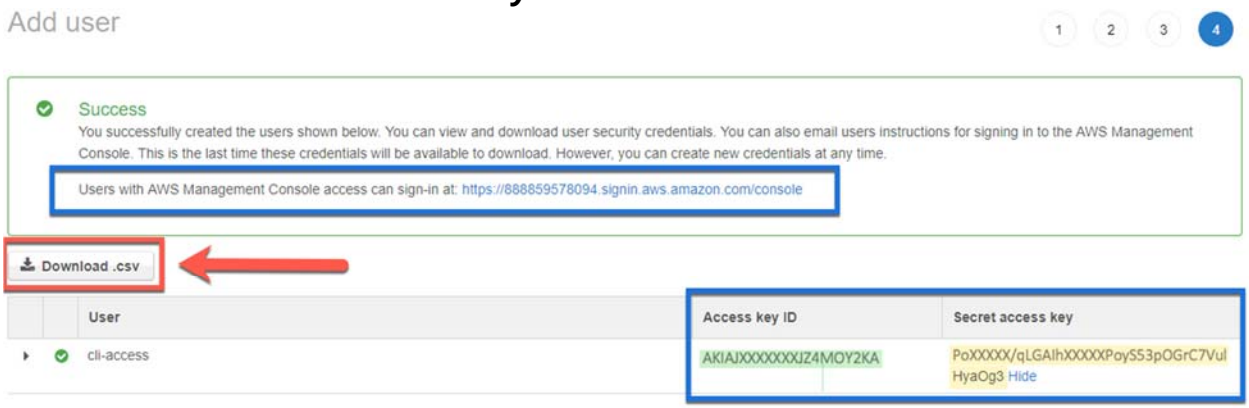


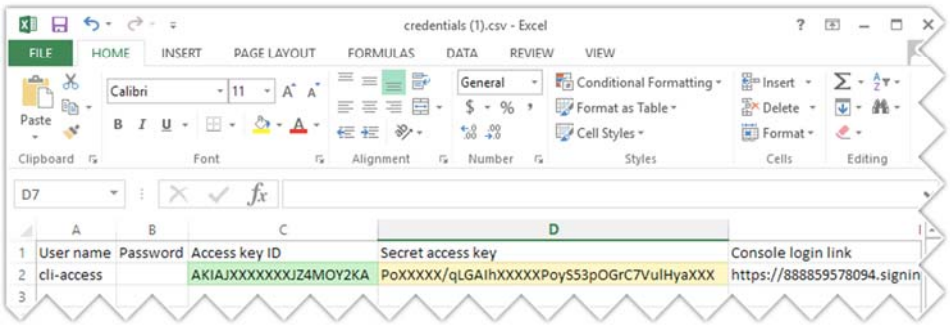
6. Click the **Next: Review** button.

7. Review the new User settings on the Review page and click the **Create user** button.



8. The last step is the Success screen. Here you will have access to the **Console Login Link**, the **Access key ID** and the **Secret access key. Download the .csv** file with the key information and save this file where it can be accessed for future reference.
Important Note: *This is the ONLY TIME you will have access to the* ***Secret access key****. It is important to download the .csv with this information or copy and paste the keys into a saved document. Once you leave this screen you will no longer have access to the* ***Secret access key****.*





Step 2: Create AWS CLI Access

To create the AWS CLI access, open Command Prompt.

1. Type the command **aws configure** and hit enter.

2. You will be prompted to enter your **AWS Access Key ID** and hit enter.

3. You will be prompted to enter your **Secret Access Key** and hit enter.

4. You will be prompted to enter your **Default region name** (Example: **us-east-1**) and hit enter. You can find the region that is closest to you here: https://docs.aws.amazon.com/general/latest/gr/rande.html

5. You will be prompted to enter the Default output format and hit enter. You will enter the format **json**.

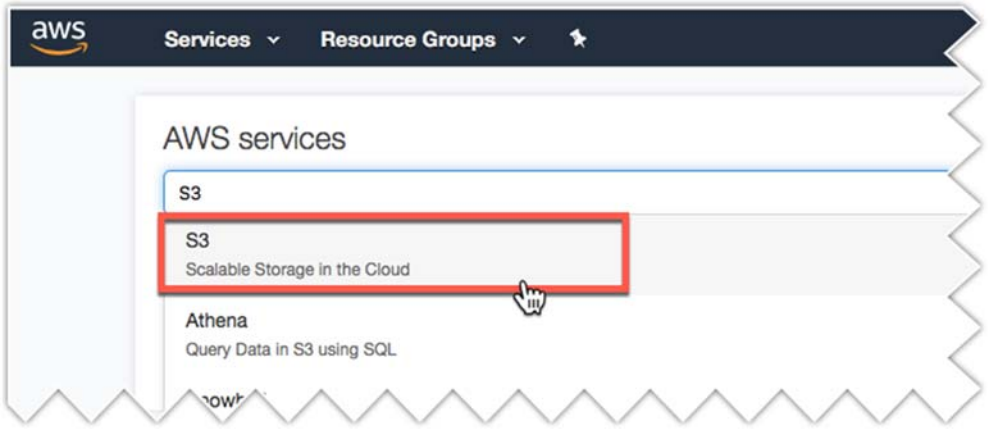
Sample Command Prompt:

```
C:\Users\User>aws configure
AWS Access Key ID [****************Y2KA]: your-access-key-id-here
AWS Secret Access Key [****************aOg3]:your-secret-access-key-id-here
Default region name [us-east-1]: your-region-here
Default output format [json]: json
```

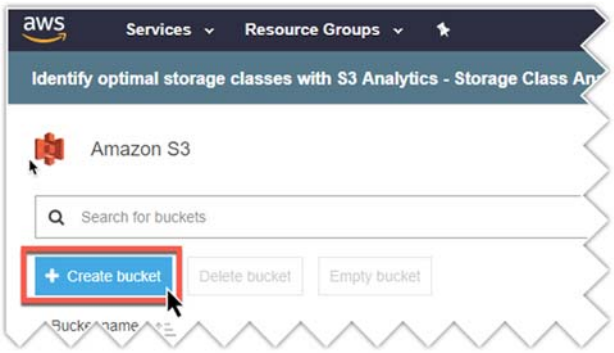
Optional: you could enter the command "aws sts get-session-token" to confirm that your installation of Python and AWS CLI is working properly.

Step 3: Create an S3 Bucket

1. From the AWS Console, search for **S3 Scalable Storage in the Cloud** and open.



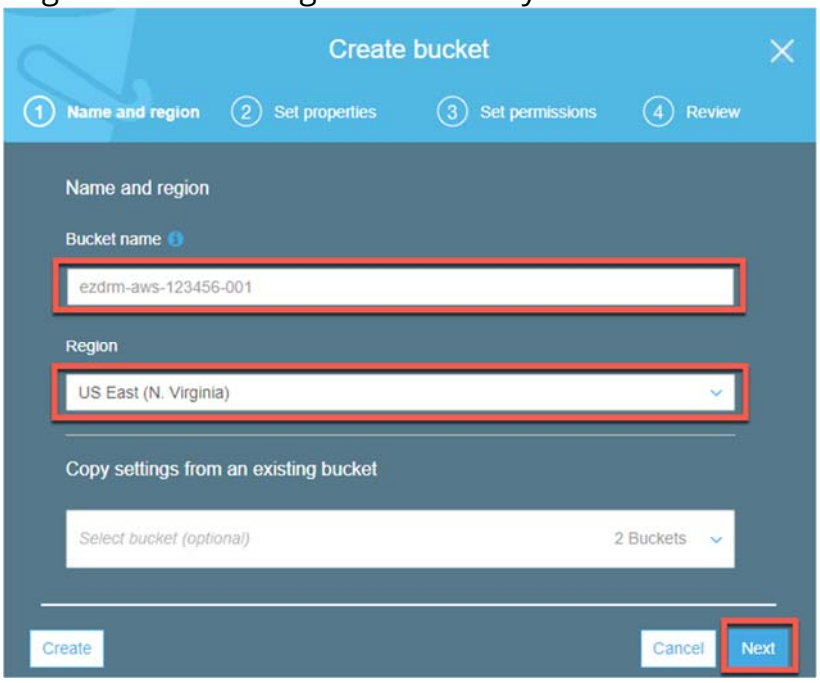
2. Click the **Create Bucket** button.



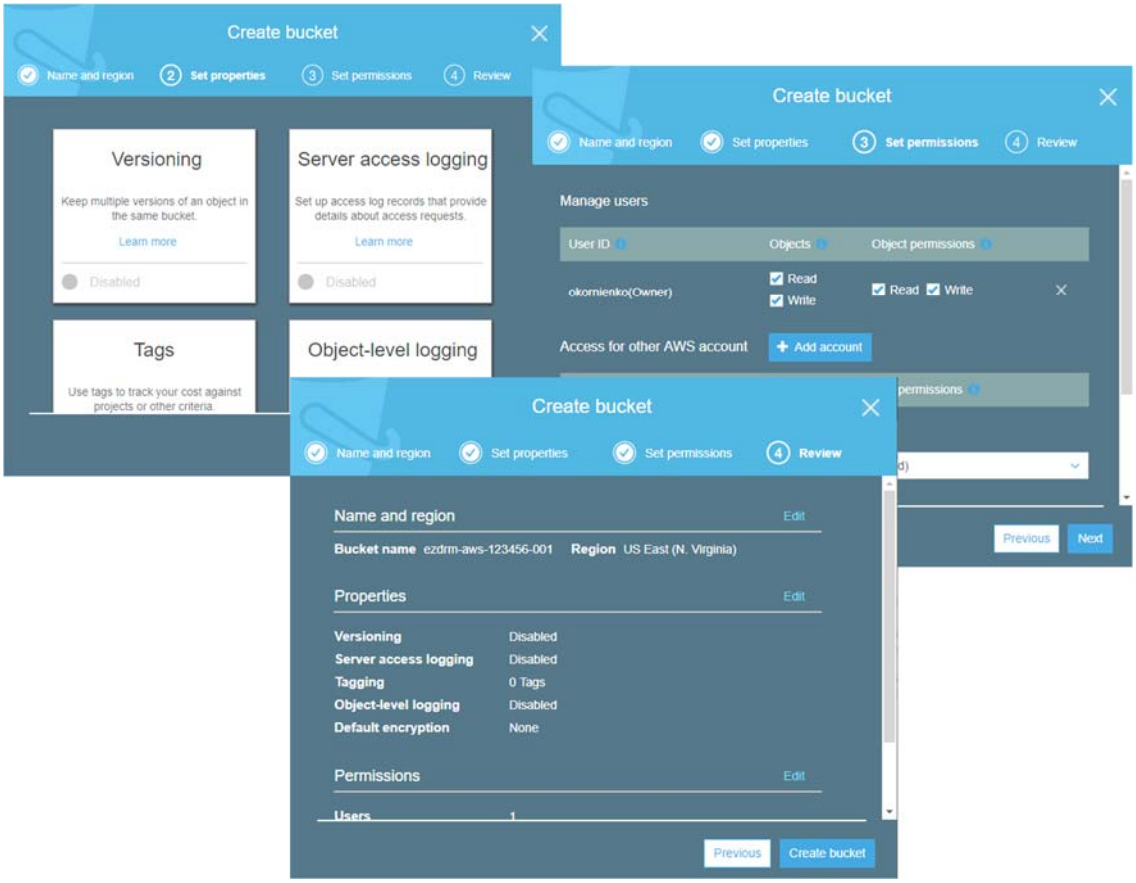
3. Enter the following parameters and click **Next**:

   - Bucket name: Bucket name can be any format you prefer. We recommend that you use a naming convention that is unique and reusable.
     Note: The bucket name must be unique across all existing bucket names in Amazon S3.

   - Region: enter the region closest to you.



4. Click the **Next** button through the next three screens keeping all the default settings, then click the **Create Bucket** button.



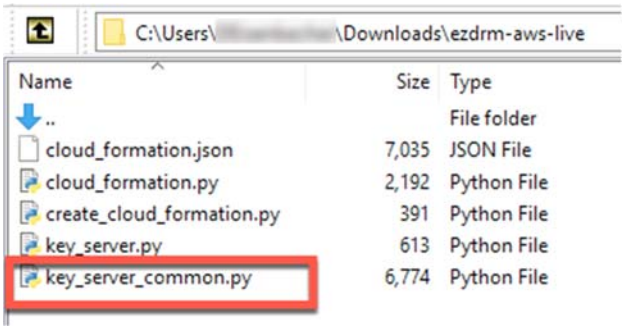
Step 4: Edit the Key Server files

To edit the Key server files:

1. Download the EZDRM AWS zip file through a browser using this link: http://www.ezdrm.com/downloads/ezdrm-aws-live.zip

2. Extract the **ezdrm-aws-live.zip** file and open the **ezdrm-aws-live** folder.

3. Right-click to edit the **key_server_common.py** file.



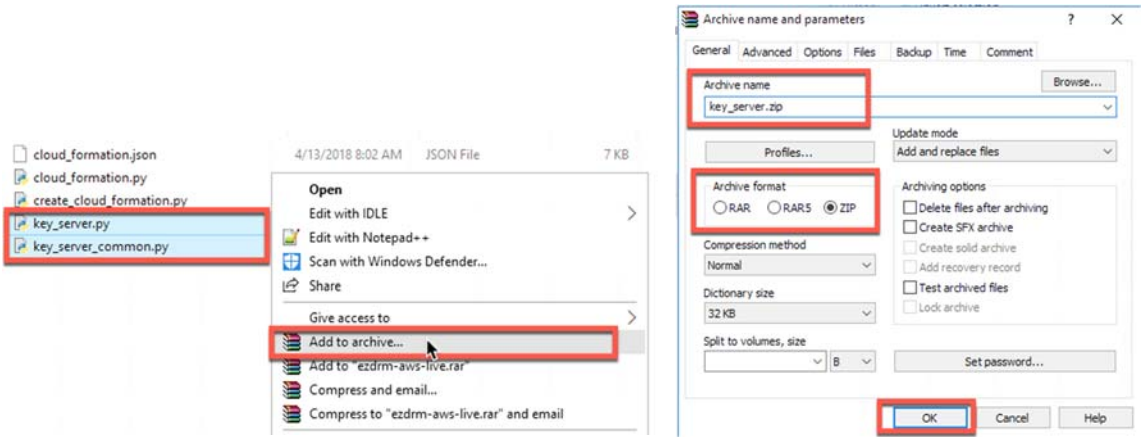
4. Edit Line 78 with your EZDRM username and password and save the file. The parameters are as follows:

Parameter	Description
u	EZDRM username
p	EZDRM password

```
xt = PLAYREADY_CONTENT_KEY

xt = base64.b64encode(get_digest(KEY_STRING, content_id, kid)).decode('utf-8')
 system_ids.get(HLS_SAMPLE_AES_SYSTEM_ID, False) == kid:
t('explicitIV', base64.b64encode(get_digest(KEY_STRING, content_id, kid)).decode('utf-8'))
lopen('http://cpix.ezdrm.com/awslive.aspx?m=' + mst + '&k=' + kid + '&u=USERNAME&p=PASSWORD&c=' + content_id + '&e
ad()
ace("cpix", "urn:dashif:org:cpix")
ace("pskc", "urn:ietf:params:xml:ns:keyprov:pskc")
```

5. After editing the **key_server_common.py** file, combine it with **key_server.py** in a zip file called **key_server.zip**.



Step 5: Create Speke Server

To create the Speke Server, open Command Prompt.

1. Navigate to the extract directory for ezdrm-aws-live.zip in command prompt.

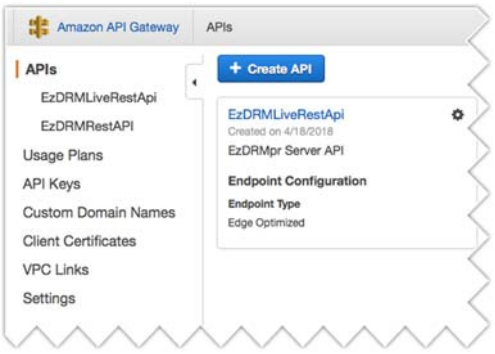
   For example:
   c:\Users\User\Downloads\ezdrm-aws-live\ezdrm-aws-live

2. Run the following python command **python create_cloud_formation.py <<insert S3 bucket name from Step 3 here>>**.

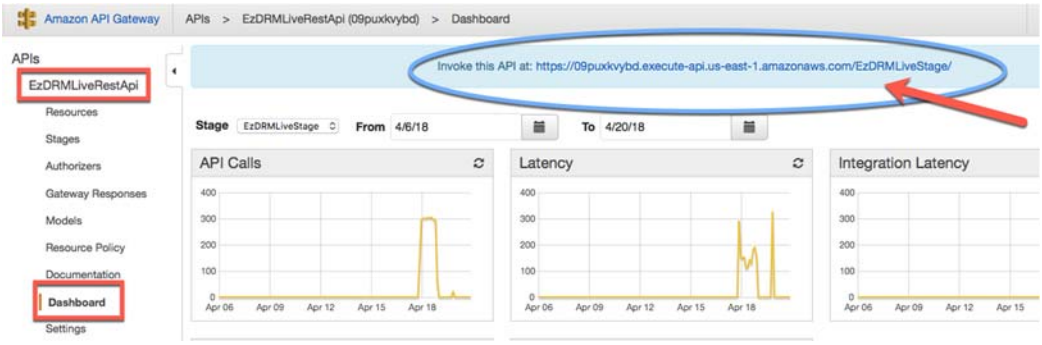
   For example:
   c:\Users\User\Downloads\ezdrm-aws-live\ezdrm-aws-live> python create_cloud_formation.py ezdrm-aws-123456-001

   NOTE: *The boto3 library needs to be installed for this to work. To install, run the command:* ***pip install --upgrade --user boto3****.*
   c:\Users\User\Downloads\ezdrm-aws-live\ezdrm-aws-live> pip install --upgrade --user boto3

   Once complete, you view see the server **EzDRMLiveRestApi** under the **Amazon API Gateway**:



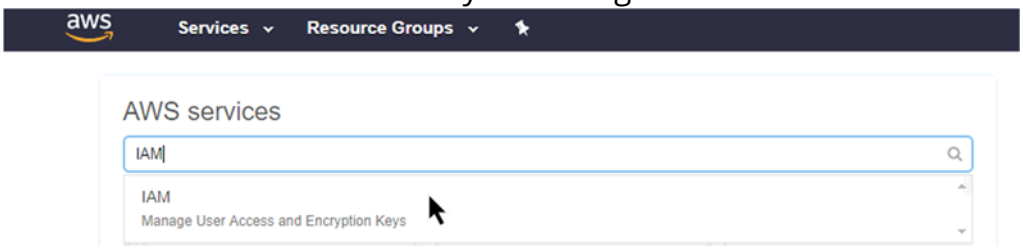
6. Click on the **EzDRMLiveRestApi** link and select the **Dashboard** menu. You will copy the API URL at the top of the **Dashboard** page labeled “Invoke this API”. Paste this URL in a notepad for editing in a future step.



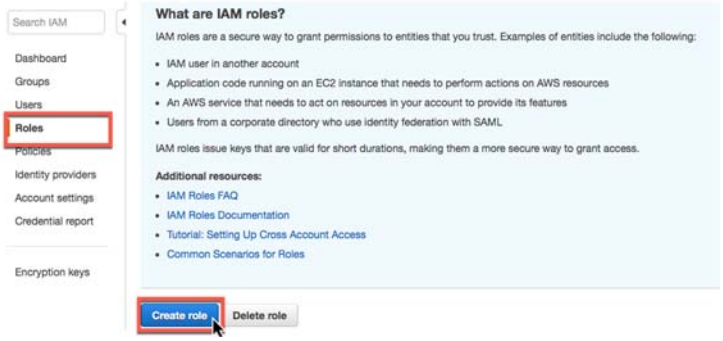
Step 6: Create Role – MediaPackage

To create a the MediaPackage Role in AWS complete the following steps:

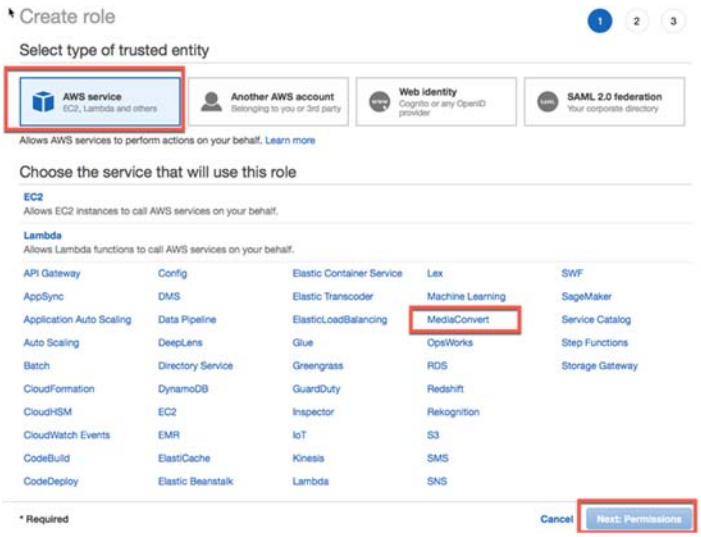
1. Launch the AWS IAM console by searching for IAM.



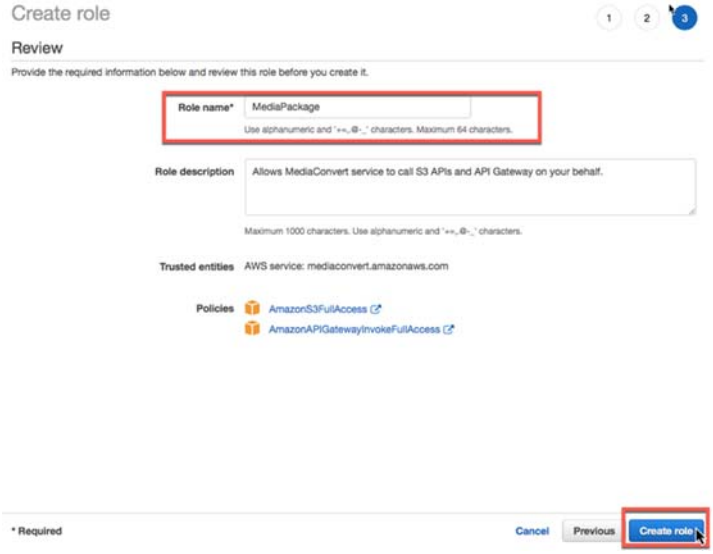
2. Go to the Roles menu. Click the **Add role** button.



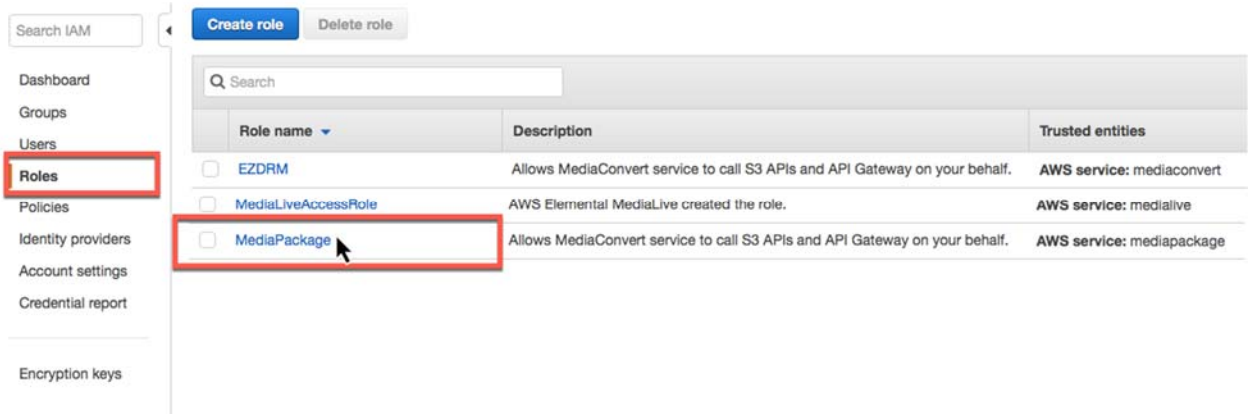
3. Under AWS service select the **MediaConvert** role (there isn't currently a role for MediaPackage) and click the **Next: Permissions** button.



4. Enter the **Role name** and click the **Create role** button.

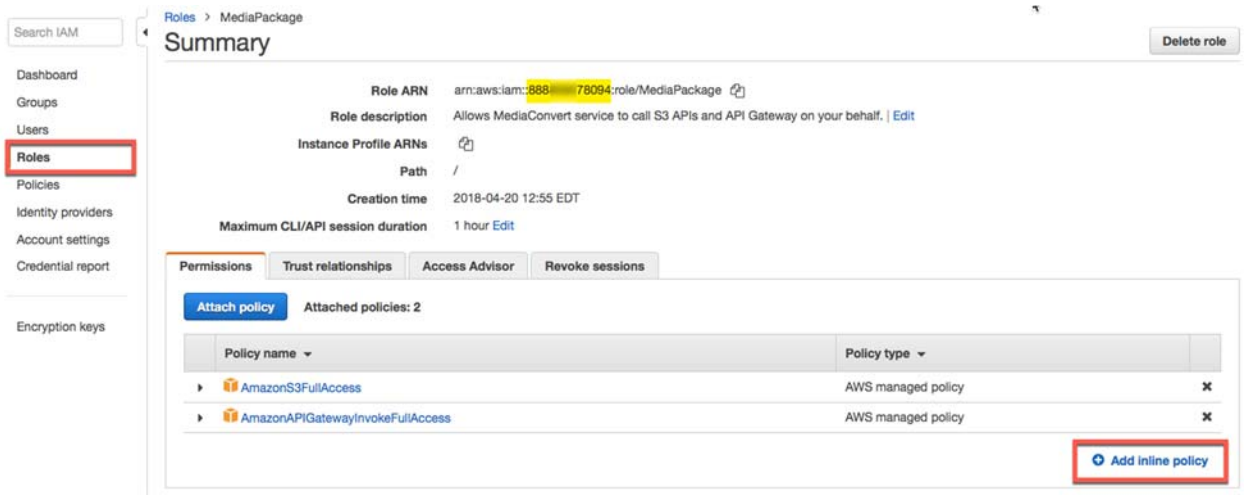


5. Now that the MediaPackage role is created, click on the link to open the role details.

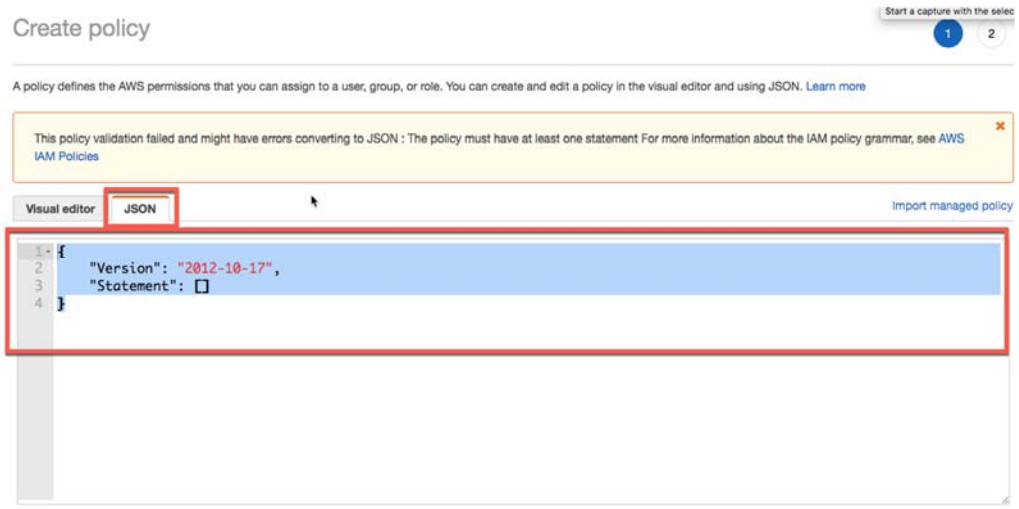


6. Because a role doesn't exist for MediaPackage, you will need to add an inline policy and change the settings of these role. This gives permission to execute the copy protection.

   First, note your **AWS Account ID** as part of the **Role ARN** value (you can also find this value under the My Account menu under Account Settings). Click the link to **Add inline policy**.

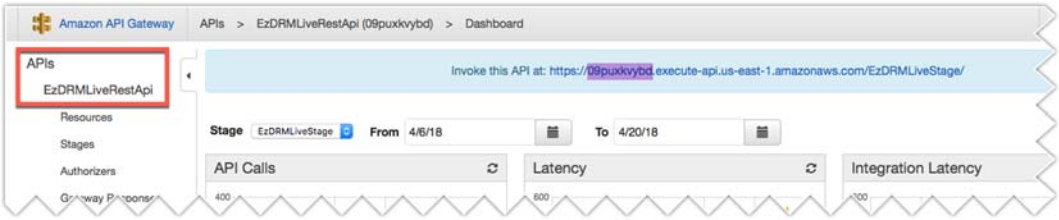


7. Next select the **JSON** tab and replace with the following code:



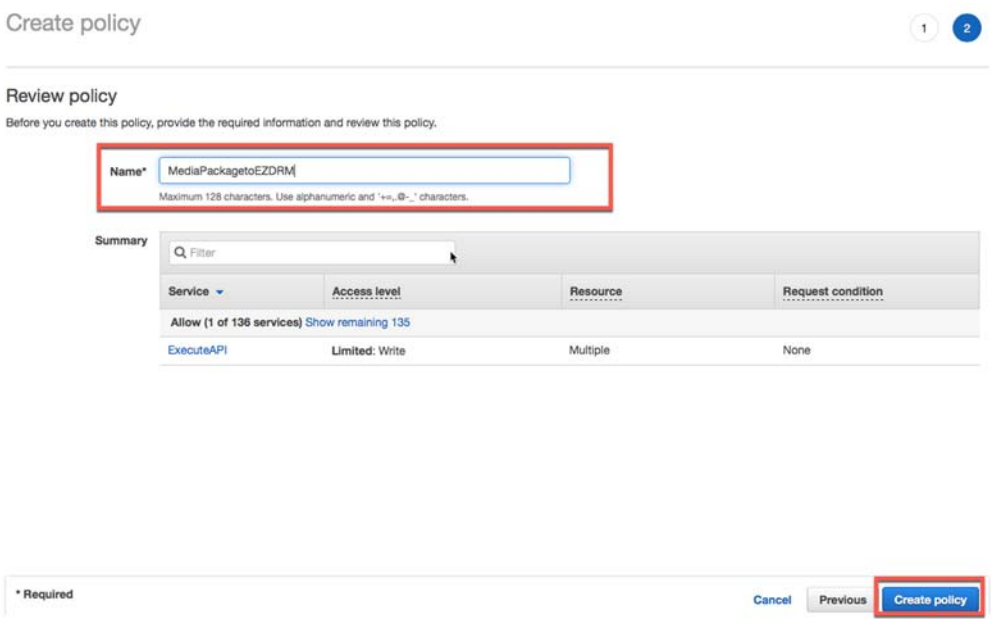

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "execute-api:Invoke"
      ],
      "Resource": [
        "arn:aws:execute-api:us-east-1:888XXXX78094:09puxkvybd/*/GET/client/*/*",
        "arn:aws:execute-api:us-east-1:888XXXX78094:09puxkvybd/*/POST/copyProtection"
      ]
    }
  ]
}
```

The yellow highlighted value is your **AWS Account ID**, the purple highlighted value is from the **EzDRMLiveRestApi Invoke this API** URL created in Step 5 (this value would change if you redeploy the Speke server).



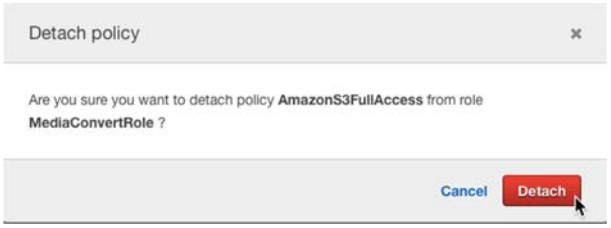
8. Once you've entered the correct code in the JSON tab, click the **Review policy** button.

9. On the Review policy page, fill in the policy **Name** and click **Create policy**.

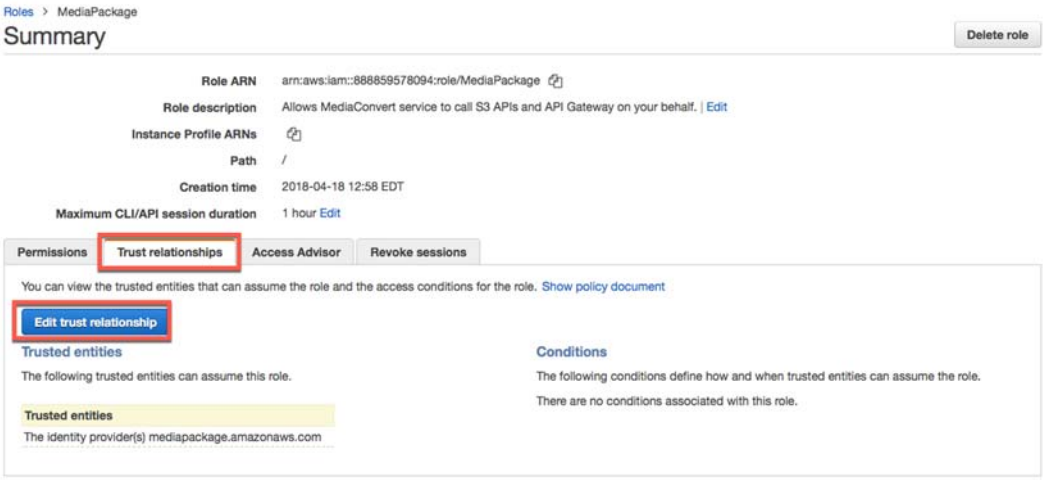


10. Now you will detach the two default policies from the role by clicking the "x" next to **AmazonS3FullAccess** and **AmazonAPIGatewayInvokeFullAccess**.

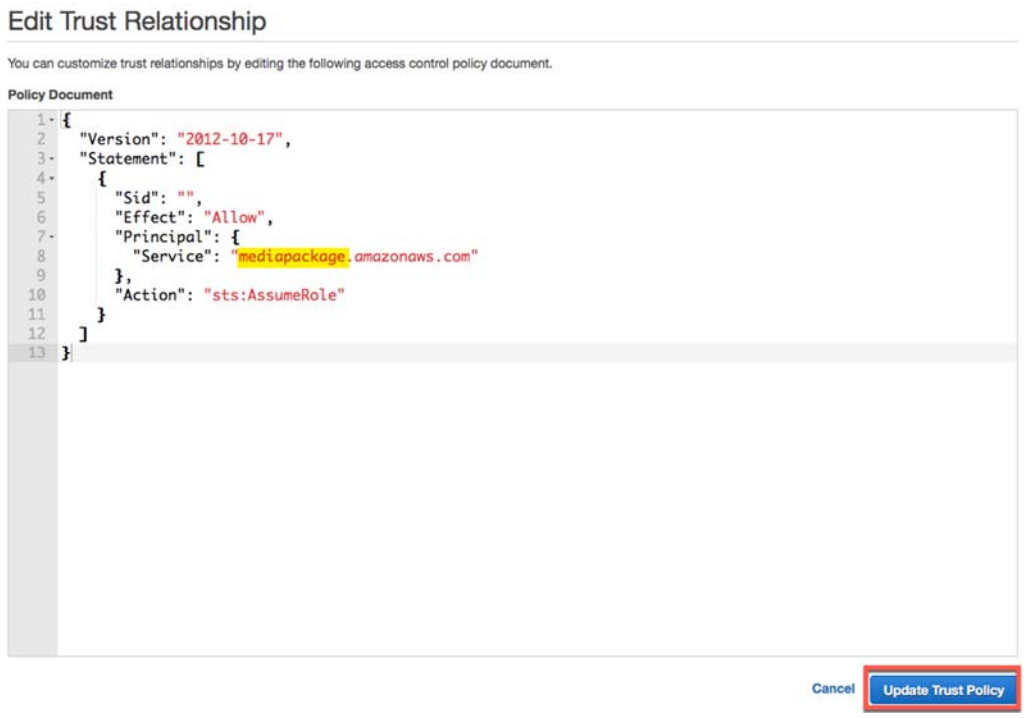
Click **Detach** on the Detach policy confirmation screen for both.



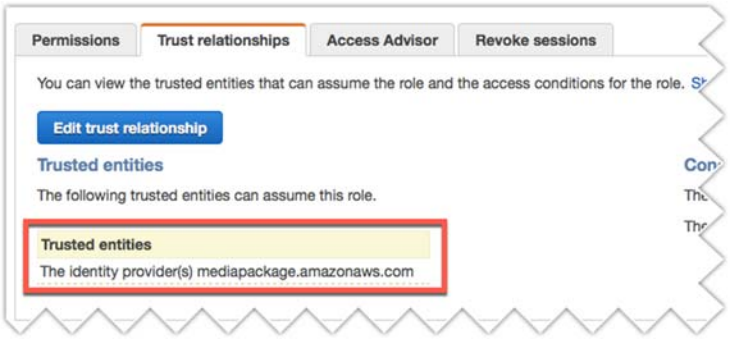
11. Then click on the **Trust relationships** tab and click the **Edit trust relationship** button.



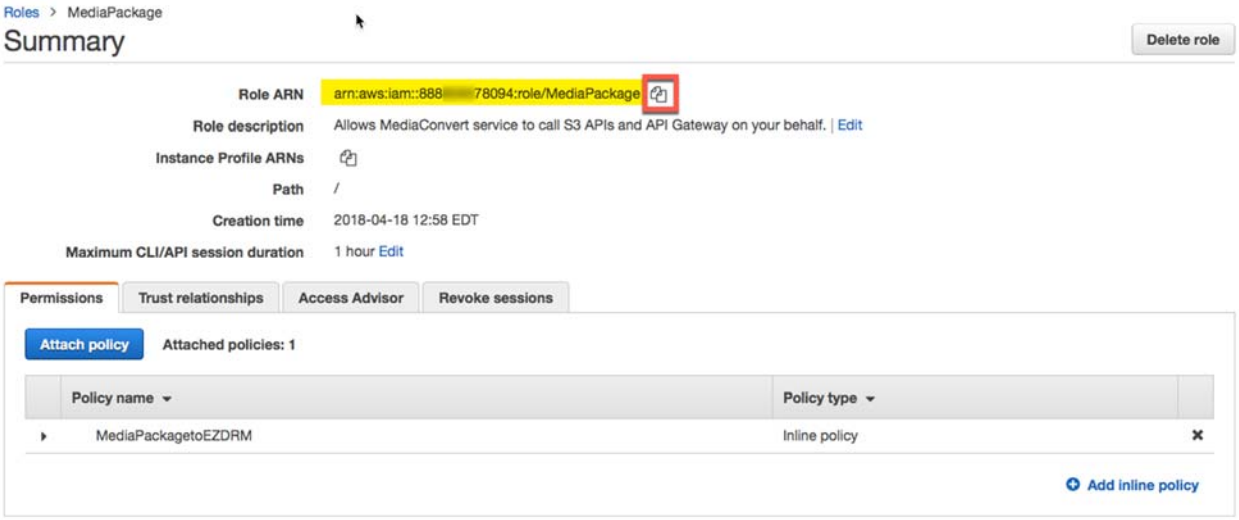
12. Edit line 8 from "mediaconvert.amazonaws.com" to "**mediapackage.amazonaws.com**" and click the **Update trust policy** button.



The Trust relationships tab should be updated as follows:



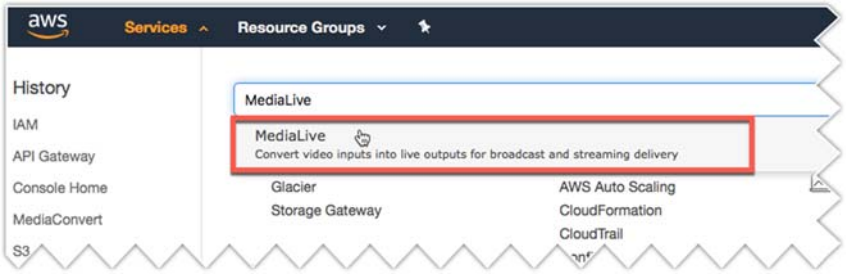
13. Once the MediaPackage role is created, make note of the **Role ARN** value for use in a later step. You can copy this value using the doc copy shortcut.

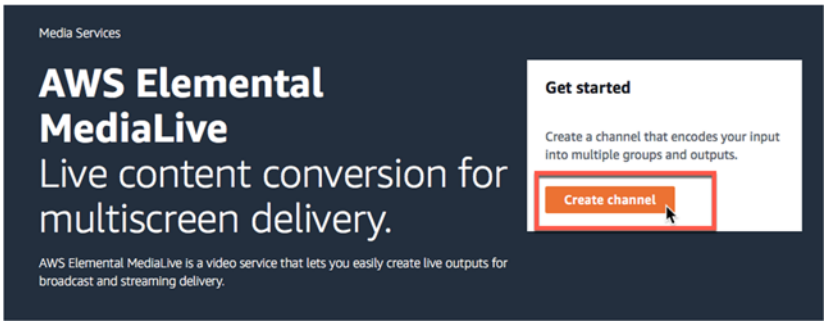


Creating an AWS MediaLive & MediaPackage Job

Create a Channel in MediaLive

1. Through **AWS Services** go to **MediaLive** and under **Get Started**, click **Create Channel**.

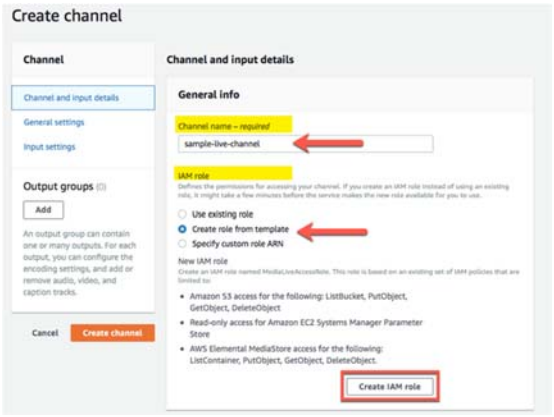


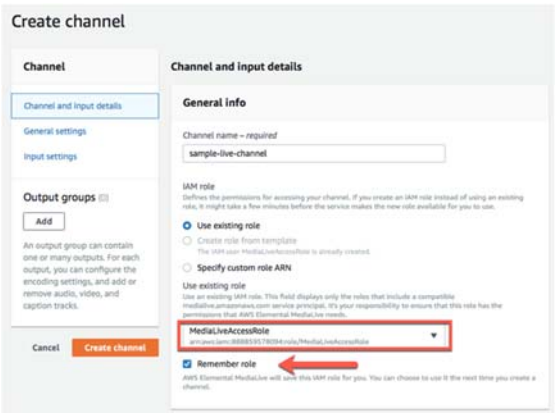


Channel and Input Details

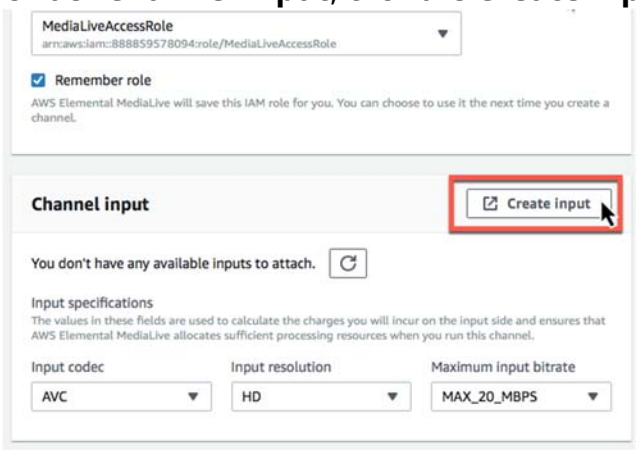
2. The channel is the input for your live broadcast. Enter the **Channel Name** (this is a required value).

3. Under **IAM Role**, the first time you create a channel, you can select **Create Role from Template** and click **Create IAM role**. The **MediaLiveAccessRole** will be created. You can select to **Remember role** and it will be available as the existing role for future channels.

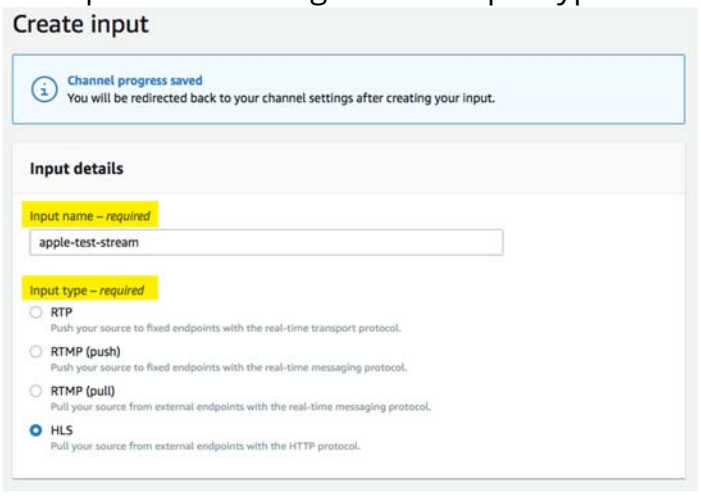




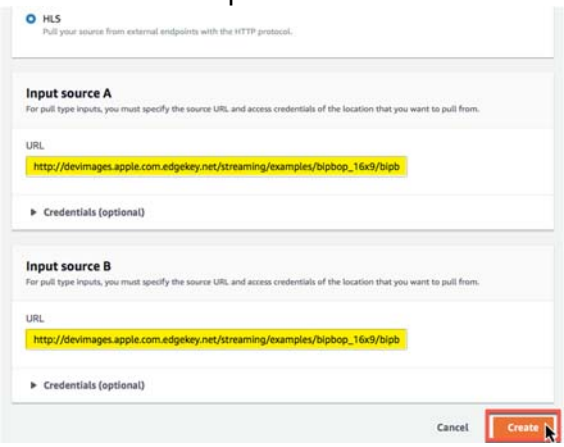
4. Under **Channel input**, click the **Create input** button.



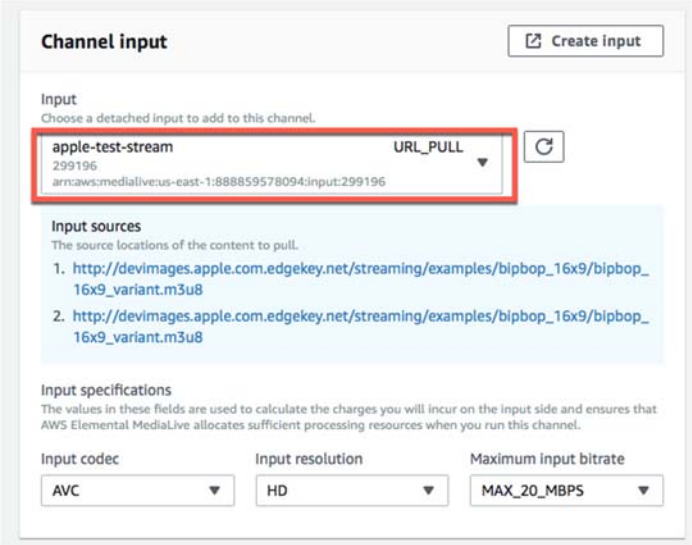
5. This will pull the source and type of stream pushing up to **MediaLive**, for this example we are using the HLS input type.



6. Channel **Input source A** and **Input source B** will be the same for redundancy. For this example, we are using a publically available HLS stream provided by Apple for testing. You will enter your encoders publishing point URL for both Input sources and click **Create** button.



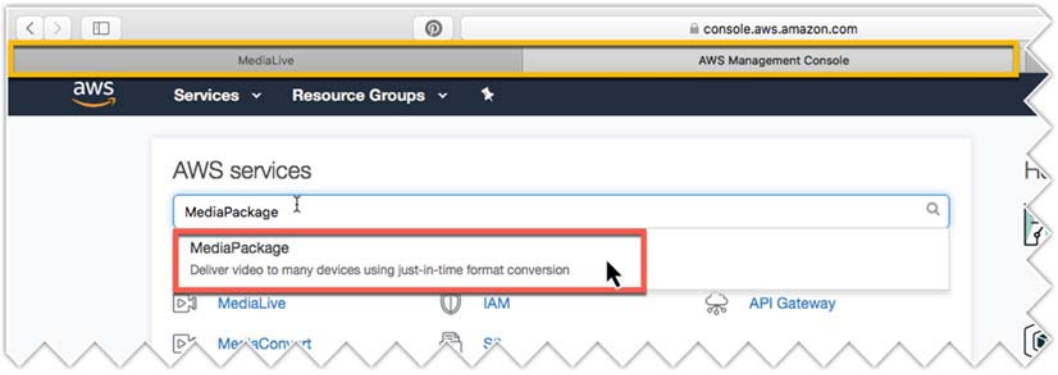
7. Once the Input is created, it can be selected from the Input dropdown menu.



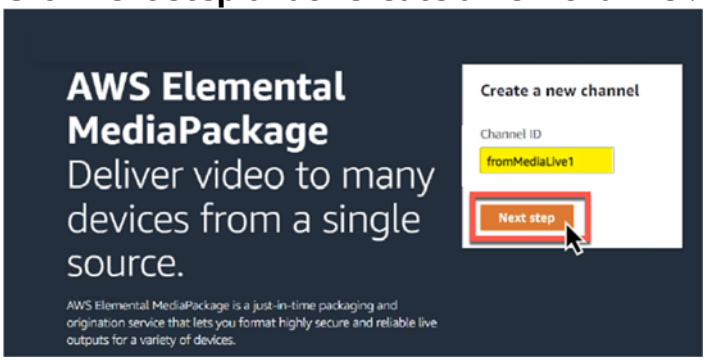
Create Channels in MediaPackage

8. The next step is to create a new channel in **MediaPackage** to ingest the stream that is coming from MediaLive.

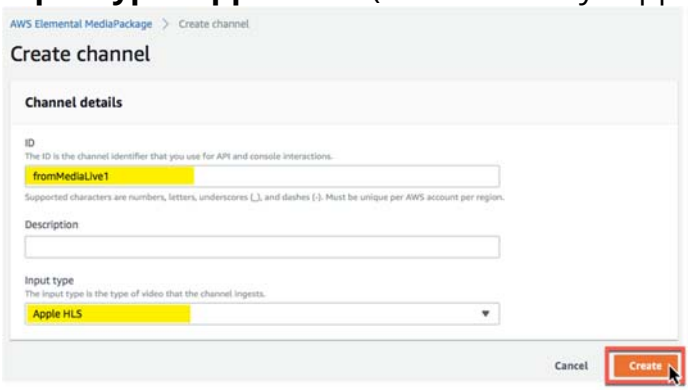
   Note: *It is helpful to have multiple tabs open during this process, for ease of copying settings between MediaLive and MediaPackage.*



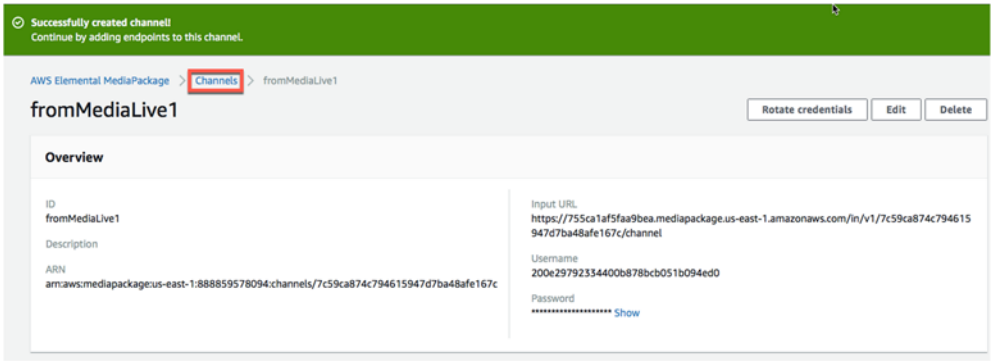
9. Click **Next Step** under **Create a new channel**.



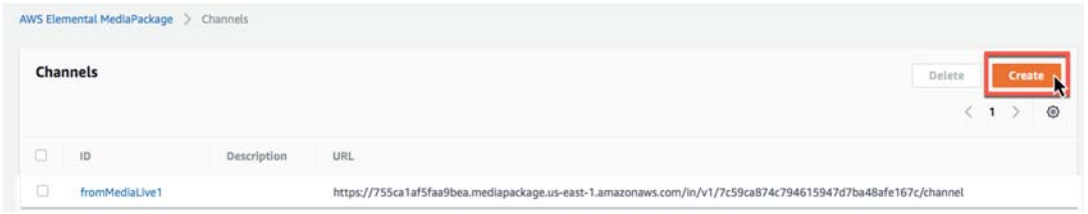
10. Enter the Channel details including the **ID** channel identifier and select the **Input type** "**Apple HLS"** (this is the only supported type). Click **Create**.



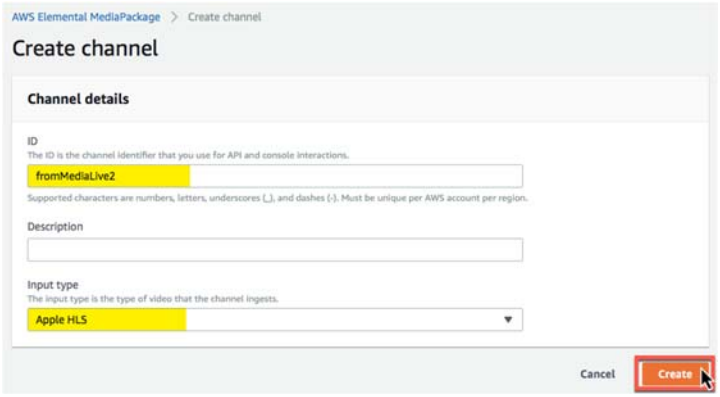
11. This will create the MediaPackage channel. For redundancy, a second channel will need to be created. Select **Channels**.



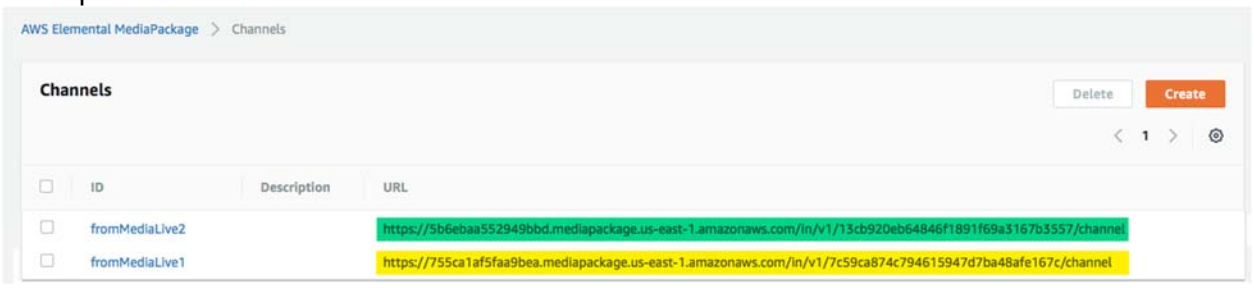
12. Click create on the **Channels** page and click the **Create** button to create the second redundant channel.



13. Enter the **Channel details** and click **Create**.

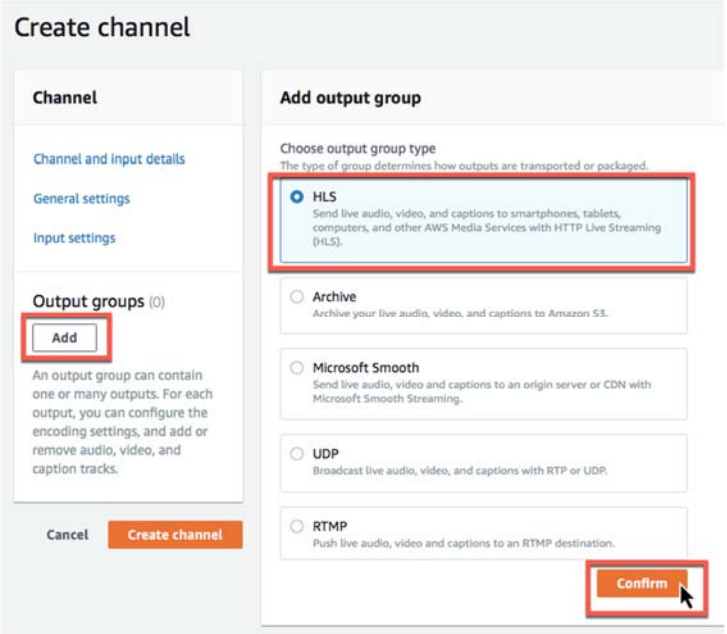


14. Now we have the URL and Channel details we will need for the Output Groups in MediaLive.



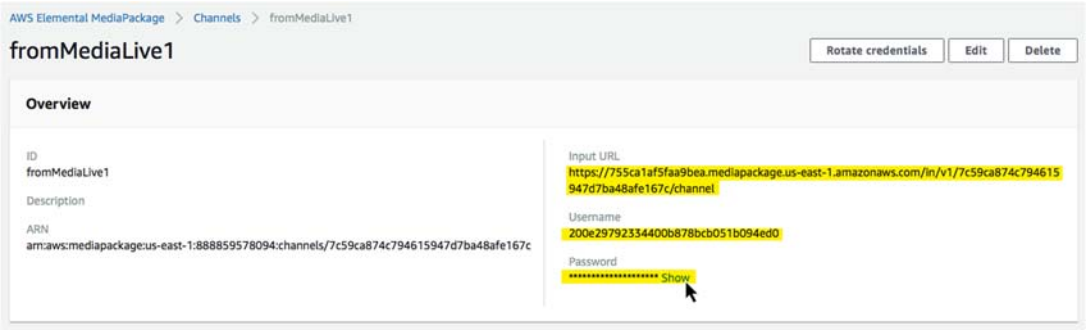
MediaLive Output Groups

15. Back in **MediaLive**, click the **Add** button under **Output groups** and select **HLS**. Click the **Confirm** button.1

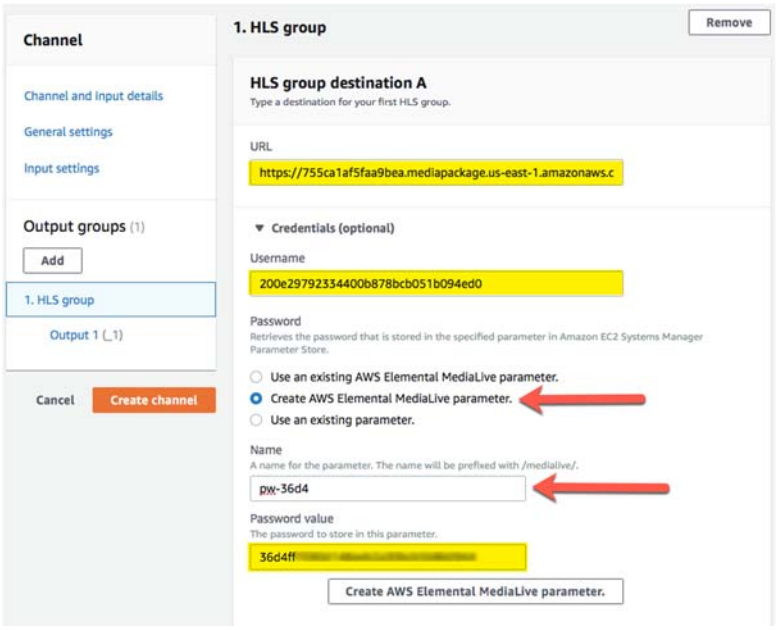


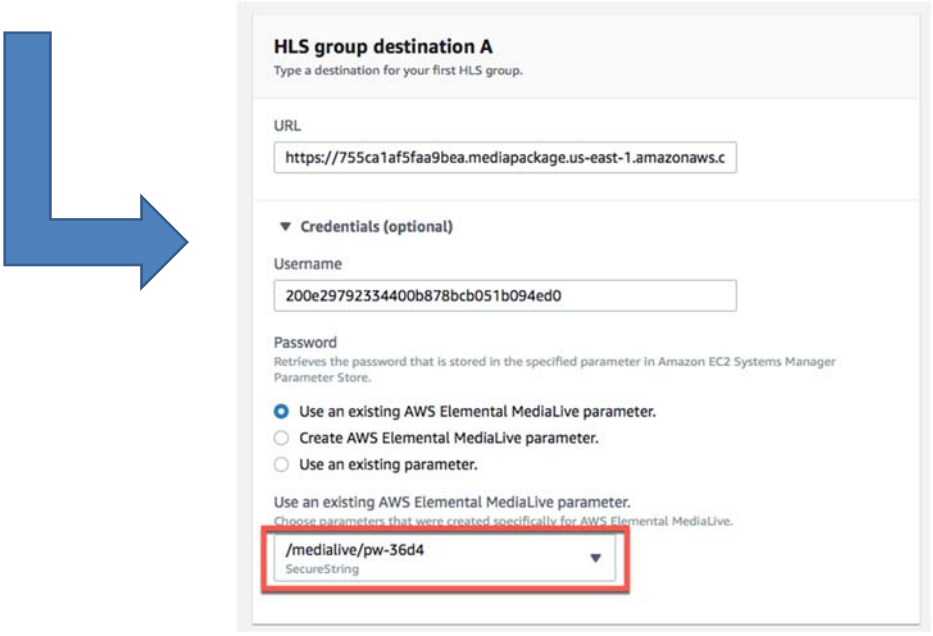
Note: *MediaPackage only accepts HLS streams.*

16. Copy and Paste the **Input URL**, **Username** and **Password** from the first **MediaPackage** channel you created to input in the next step.

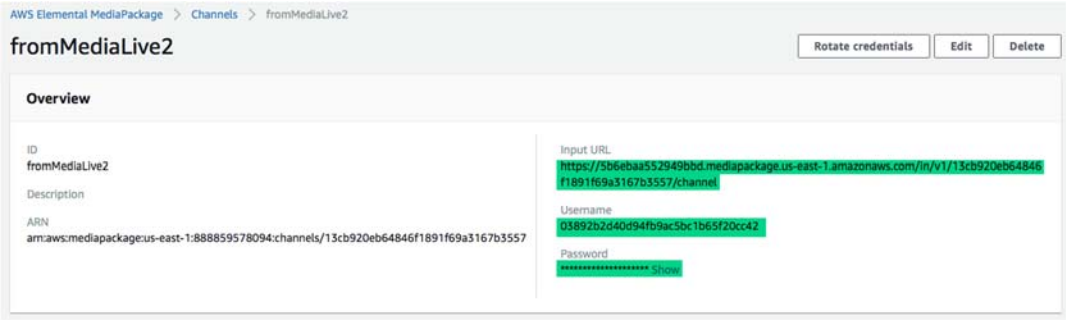


17. In **MediaLive**, in the first **HLS Group destination A**, enter the copied values for **URL** and **Username**. The first time you set up a password in the Output groups, you will select **Create AWS Elemental MediaLive parameter**. This will allow the password to be saved by AWS for future use. We recommend entering the password **Name** with something that will help you select the correct one when you have multiple channels created in the future. Enter the **Password** value and click to **Create the AWS Elemental MediaLive Parameter**.

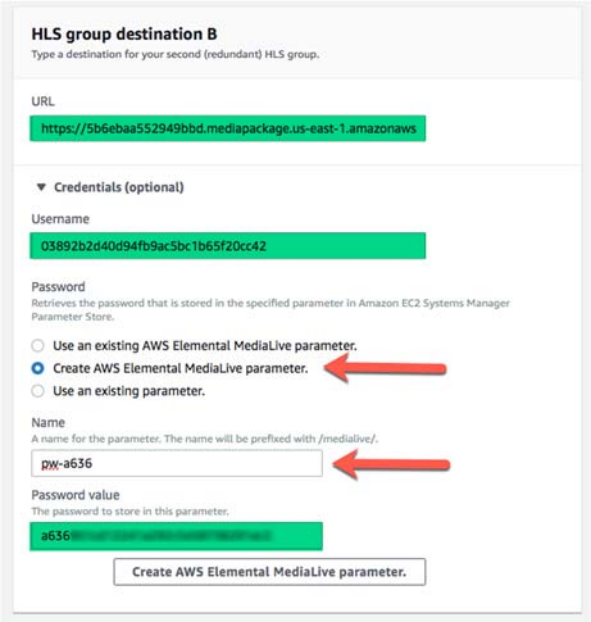


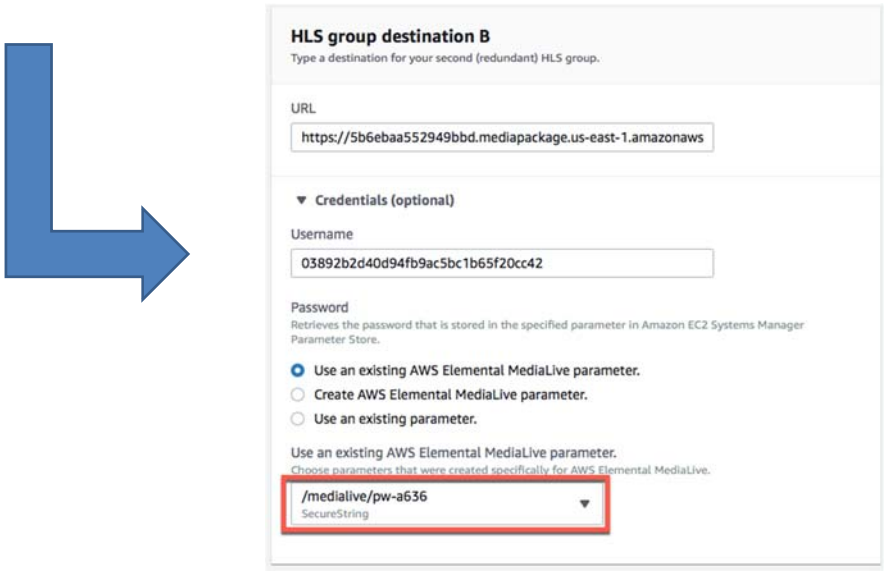


18. Copy and Paste the **Input URL**, **Username** and **Password** from the second **MediaPackage** channel you created to input in the next step.

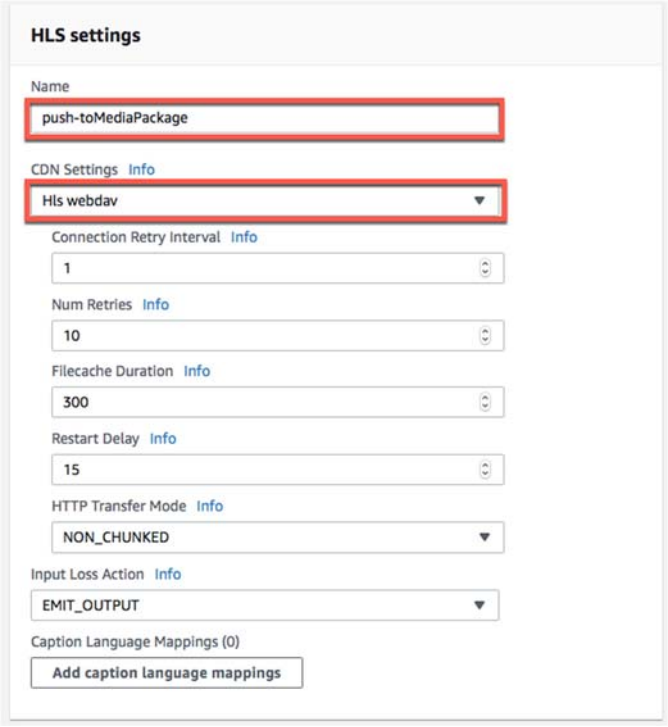


19. Back in **MediaLive**, in **HLS Group destination B** for redundancy, repeat the process in Step 17 to enter parameters for **URL** and **Username**, Password **Name** and **Password** from **MediaPackage**.





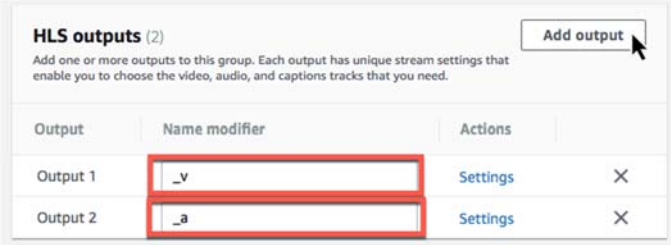
20. Next in **MediaLive** under **HLS Settings** enter a **Name**, and for **CDN Settings** select **Hls webdav**. Keep the other settings as default.



DASH-ISO Output example

21. This is the Output set up for DASH-ISO. See the next section for HLS Output settings. Under **HLS outputs** click the **Add output** button to create Output 2.

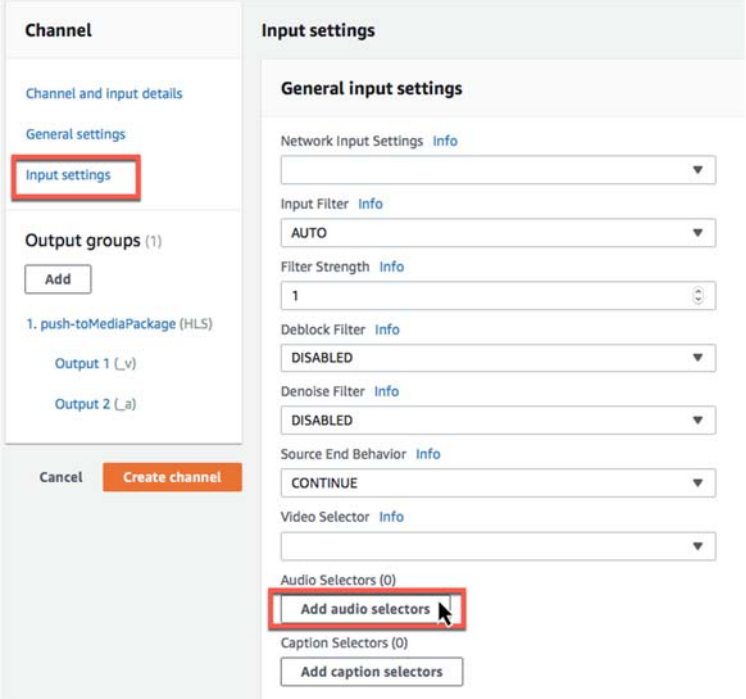
You can name Output 1 to represent the video output, and Output 2 to represent the audio output.



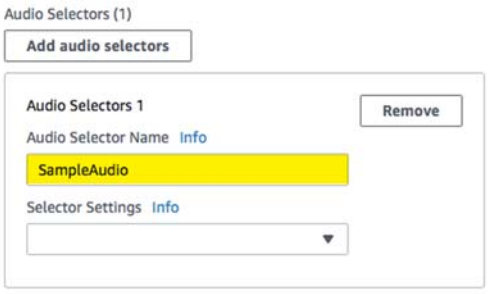
22. The rest of the settings under **Channel and Input Details** keep as default.

Input Settings

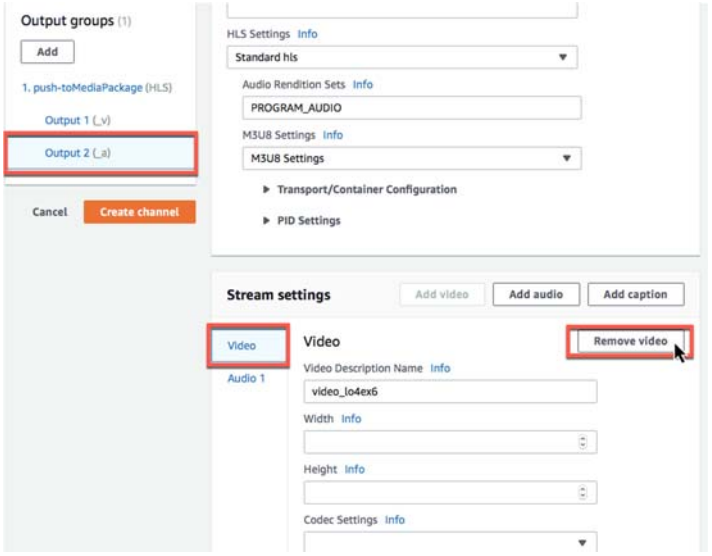
23. Click on the **Input settings** link and click the **Add audio selectors** button.



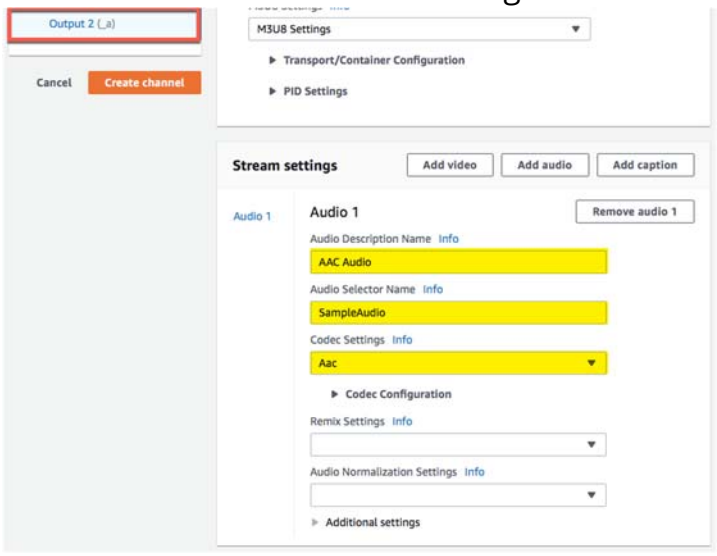
24. Enter the **Audio Selector Name** and copy it to paste in the next section.



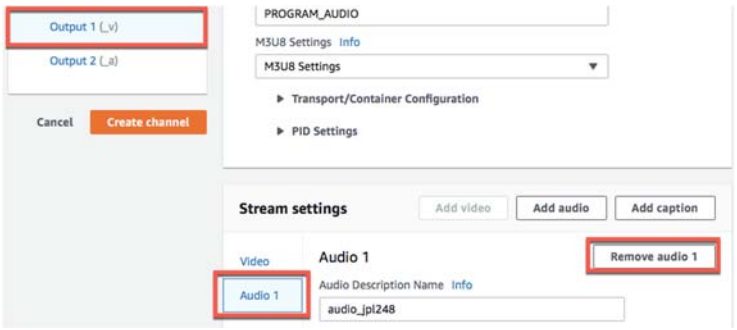
25. Under **Output Groups**, select **Output 2 (_a)** and click on the **Video** tab. Click the **Remove video** button.



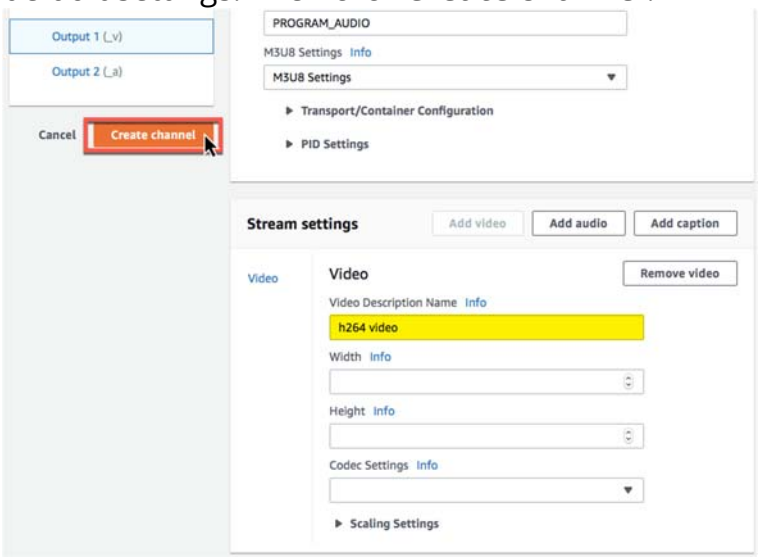
26. In the **Audio 1** section, enter the **Audio Description Name** (we recommend AAC Audio), then paste the **Audio Selector Name** that you entered in Step 24. Select **Aac** under Codec Settings.



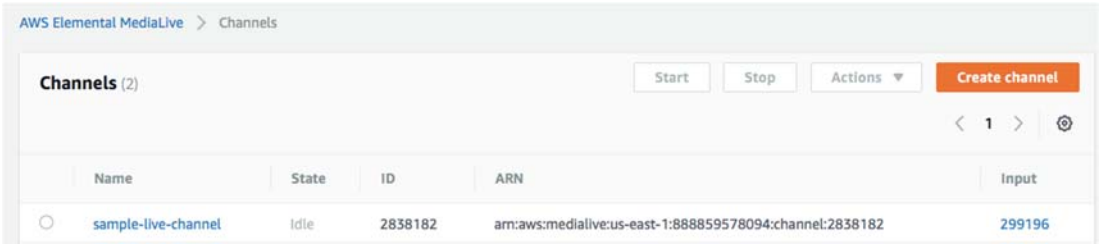
27. Next, select **Output 1 (_v)** and **Remove audio 1**.



28. You can rename the **Video Description Name** if you prefer and leave the default settings. Then click **Create channel**.

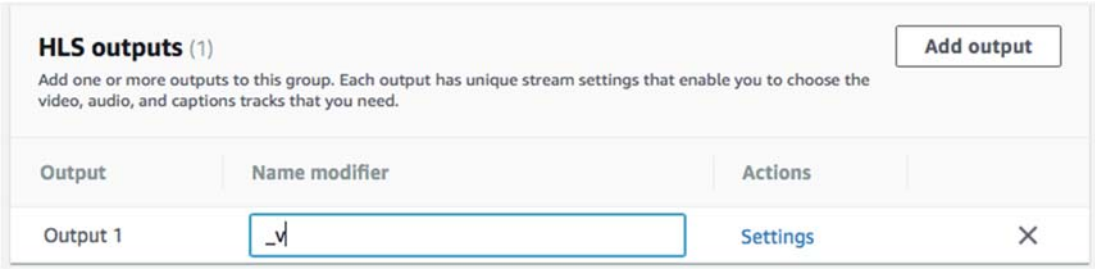


29. The MediaLive channel should now be created.



HLS Output example

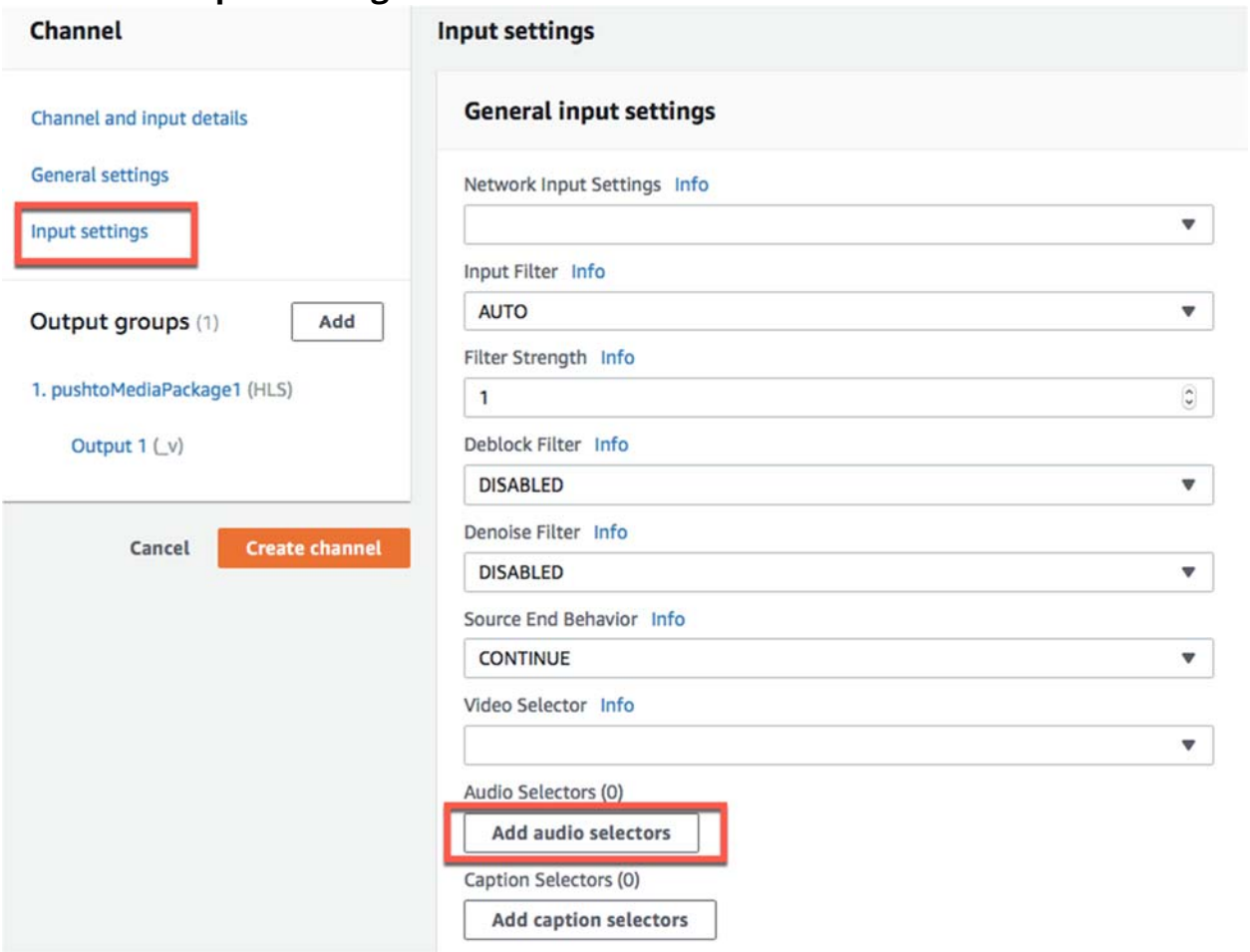
30. This is the Output setup for **HLS**. Under **HLS outputs** rename **Output 1** to represent the video output.



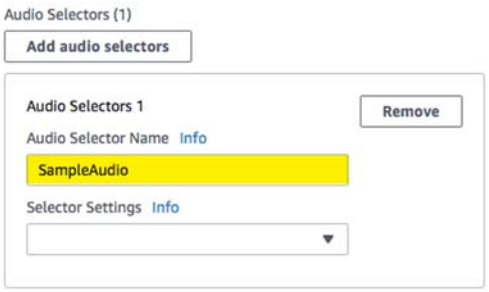
31. The rest of the settings under **Channel and Input Details** keep as default.

Input Settings

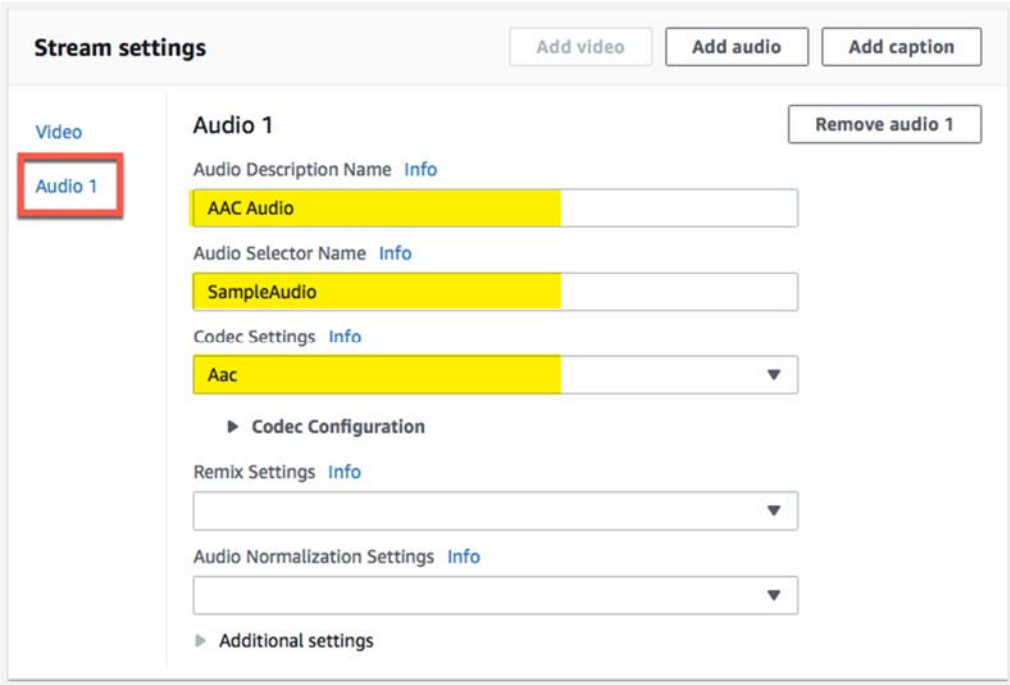
32. Click on the **Input settings** link and click the **Add audio selectors** button.



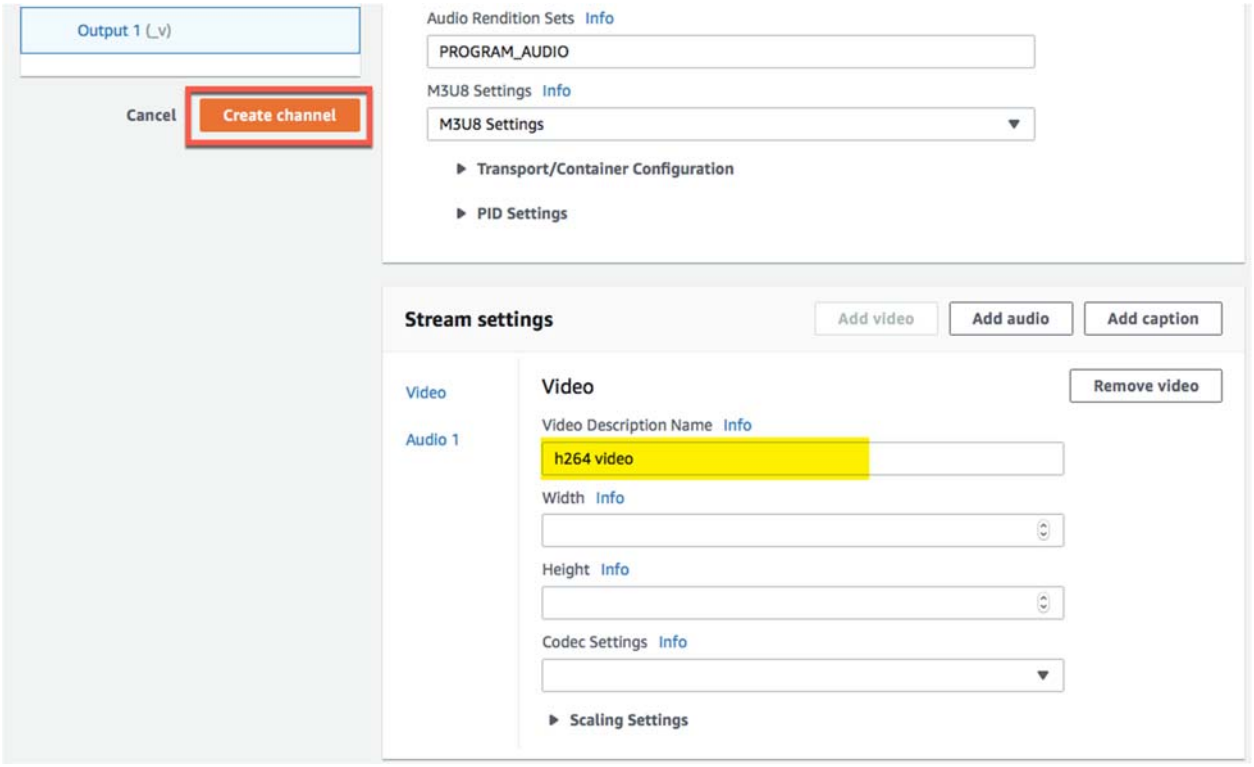
33. Enter the **Audio Selector Name** and copy it to paste in the next section.



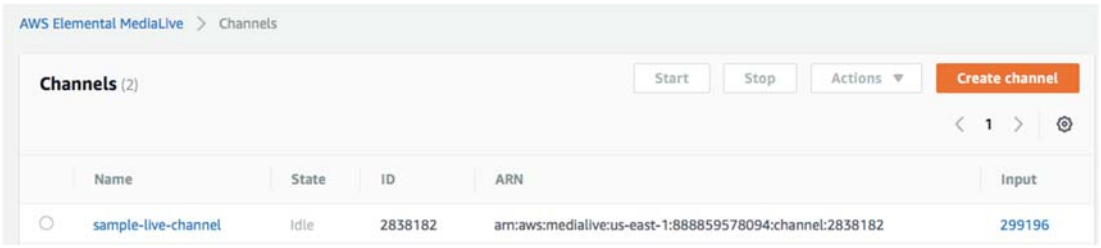
34. Under **Output Groups**, In the **Audio 1** section, enter the **Audio Description Name** (we recommend AAC Audio), then paste the **Audio Selector Name** that you entered in Step 33. Select **Aac** under Codec Settings.



35. Next, select **Output 1 (_v)**. You can rename the **Video Description Name** if you prefer and leave the default settings. Then click **Create channel**.



36. The MediaLive channel should now be created.

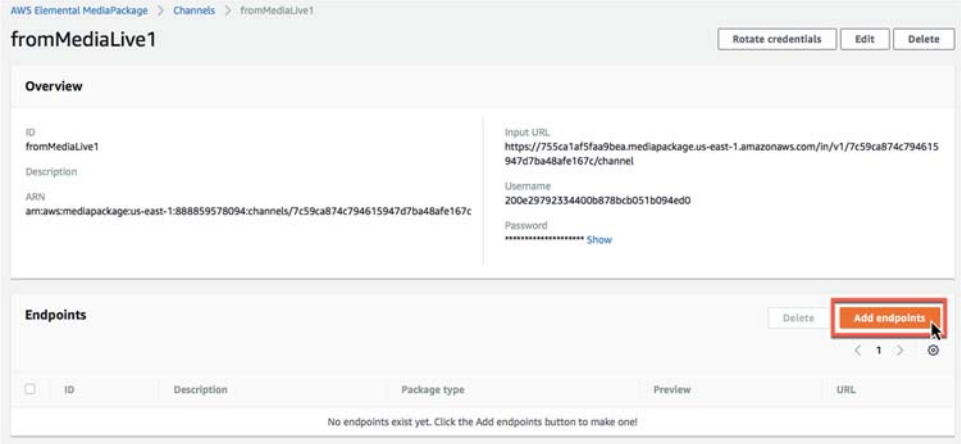


Create Endpoints in MediaPackage

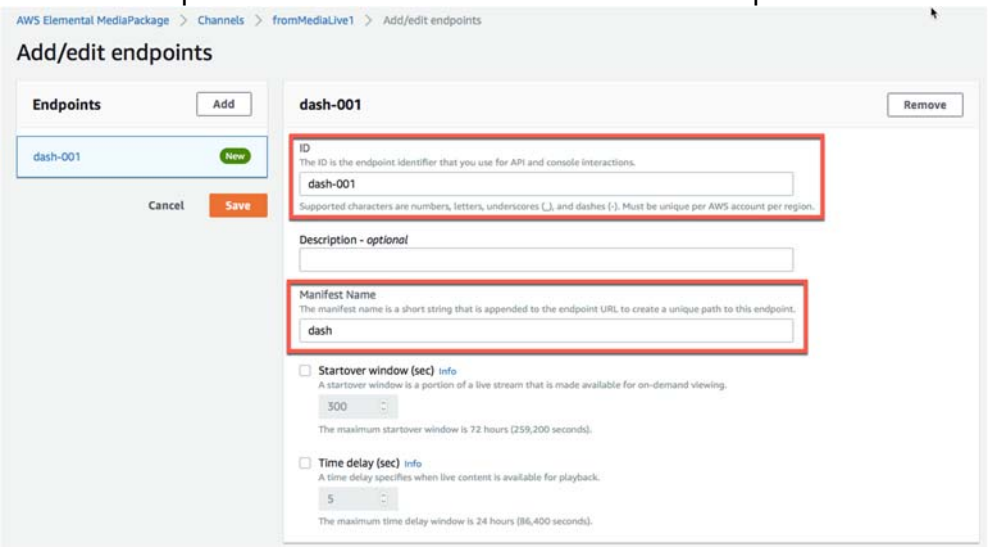
Endpoints are the outputs for the live stream for viewing. You can have multiple endpoints for each channel.

DASH-ISO Endpoint example

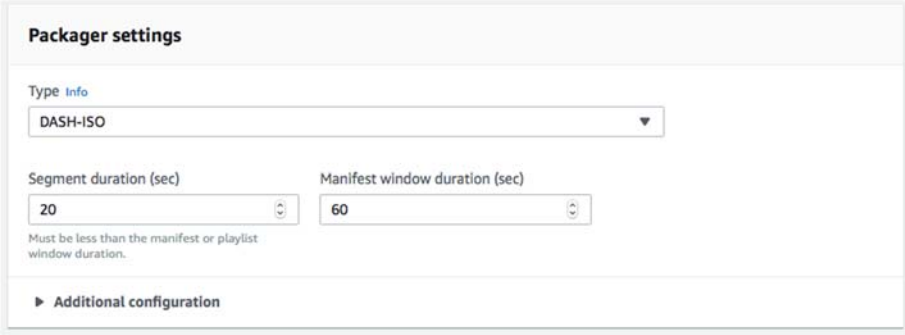
1. In **MediaPackage**, from the first MediaLive channel you created, click the **Add endpoints** button.



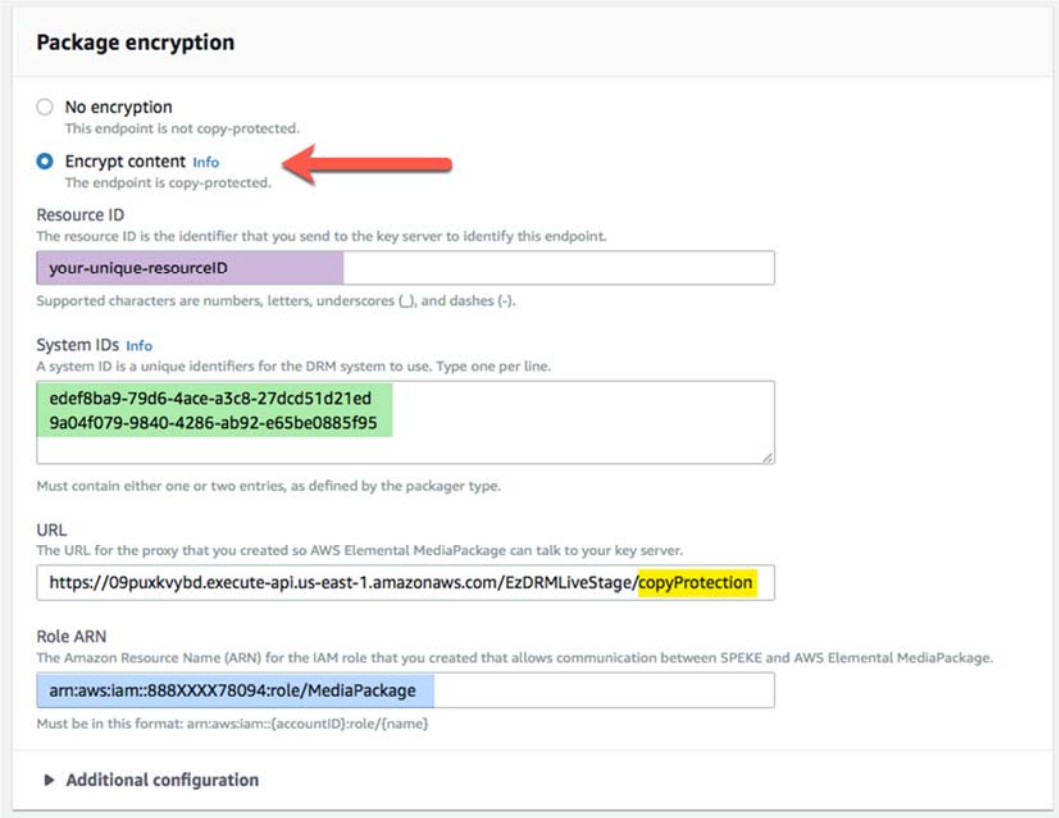
2. Edit the Endpoint **ID** and **Manifest Name** to a unique identifier.



3. Under **Packager Settings**, select the **Type DASH-ISO**, and update **Segment duration (sec)** to **20** seconds.



4. Scroll down and select the toggle for **Encrypt Content**.



The parameters are as follows:

- **ResourceID**: this will be the ID that references your DRM Keys. This is a required field.

  Note: *The first time you send a ResourceID to run a job, the ID will be tied to the DRM keys for that job. Jobs can use the same ResourceID to reference the same keys or for new DRM Keys send a new ResourceID. It is best not to use a ResourceID from a failed job.*

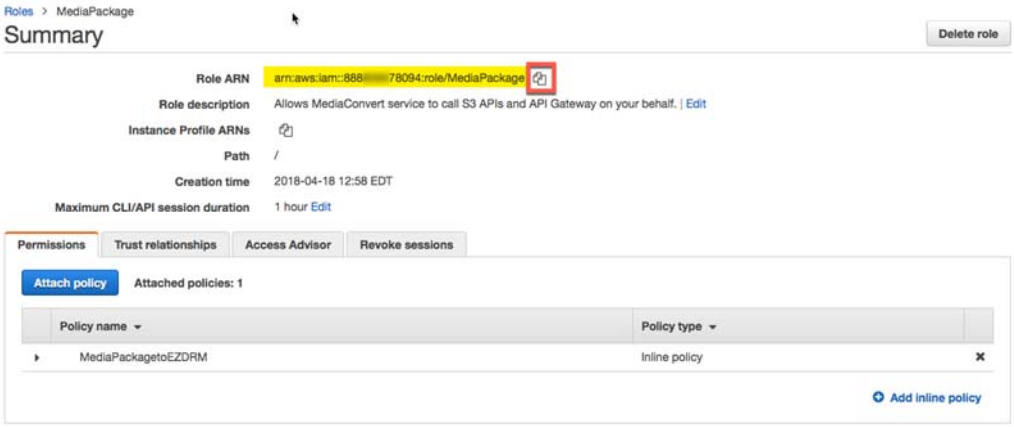
- **System ID**: Unique identifiers for the DRM system to use. Insert the System ID's for Widevine and PlayReady, one ID per line:
  (Widevine) **edef8ba9-79d6-4ace-a3c8-27dcd51d21ed**
  (PlayReady) **9a04f079-9840-4286-ab92-e65be0885f95**

  Note: *The System ID values need to be lowercase.*

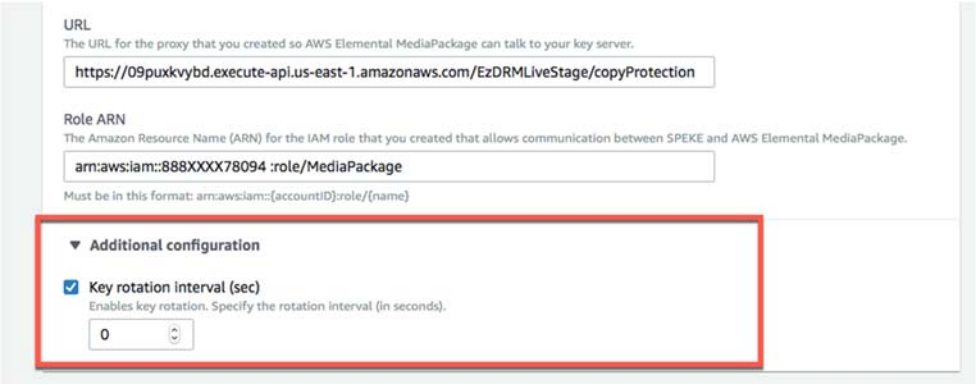
- **URL:** The URL is the **API URL** copied from **Step 5** above, except to the end of the URL add "**/copyProtection**". (This is case sensitive, be sure **capitalize the P** in Protection.)

  Sample URL:
  https://09puxkvybd.execute-api.us-east-1.amazonaws.com/EzDRMLiveStage/copyProtection

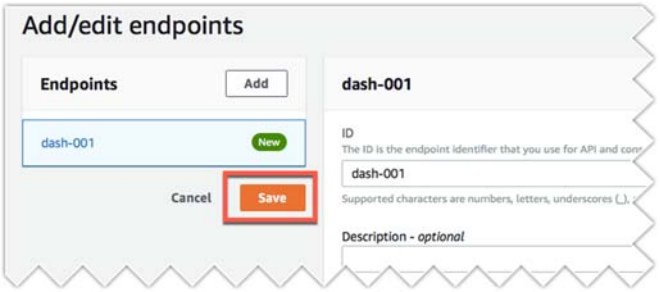
- **Role ARN:** This value is from the **MediaPackage Role ARN** created in **Step 6**.



5. Under **Additional configuration** (Package Encryption), select the **Key rotation interval (sec)** and enter a value of **0**. (This solves a current issue with Key rotation not saving as off.)



6. Once these settings are completed, click the Save button to create the endpoint.

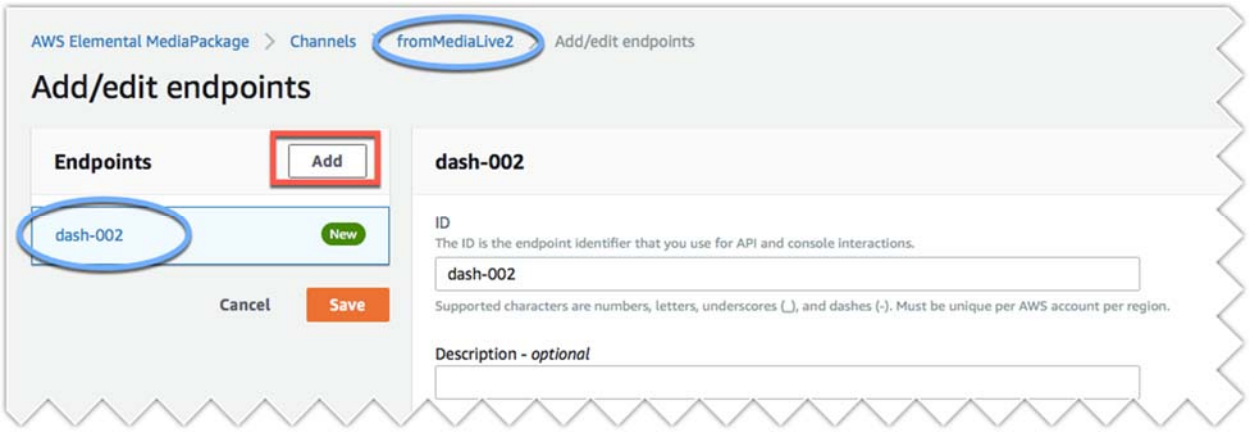


7. Now for redundancy, from your second MediaLive channel, create a DASH-ISO endpoint with the same settings as the one we just created, but change the **ID** name to indicate the redundant endpoint.

   For this example, we called our first channel **MediaLive1** and created the DASH-ISO endpoint **dash-001**. Under **MediaLive2** we will create a duplicate DASH-ISO endpoint but name it **dash-002**.

   Duplicate ALL the same settings for the second DASH-ISO endpoint under the second channel and click **Save**.

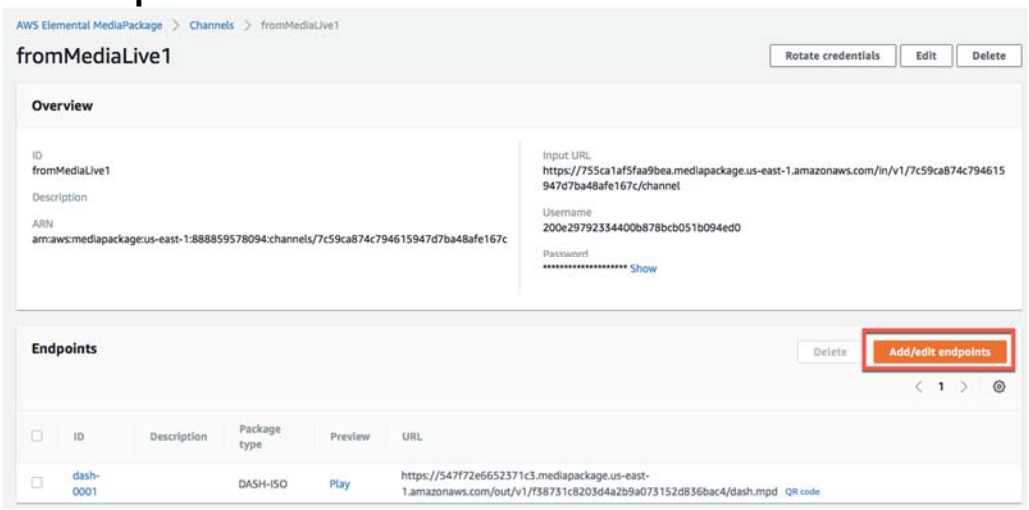
   Note: *It is helpful to have multiple tabs open during this process, for ease of copying settings from one channel to the other.*



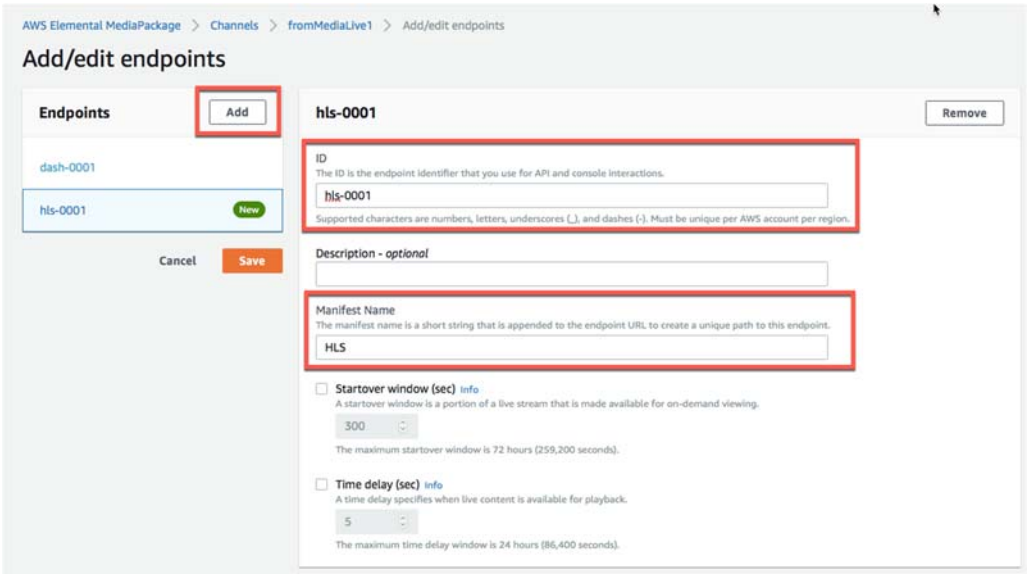
8. Once **MediaLive** is running and publishing to **MediaPackage**, you will be able to access the URL created to play the encrypted Media.

Apple HLS Endpoint example

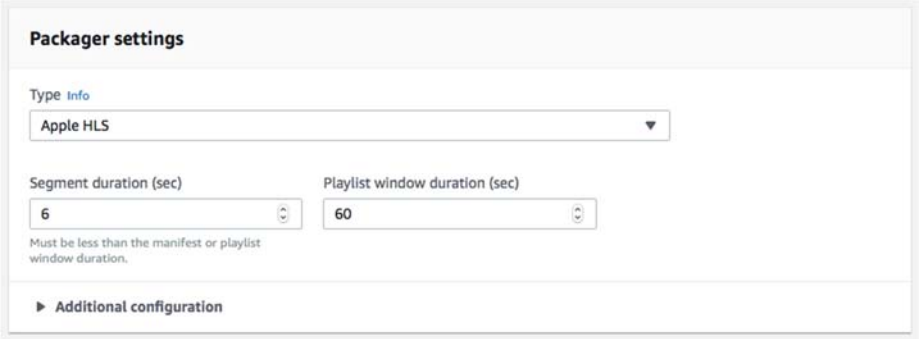
1. In **MediaPackage**, from the first MediaLive channel you created, click the **Add endpoints** button.



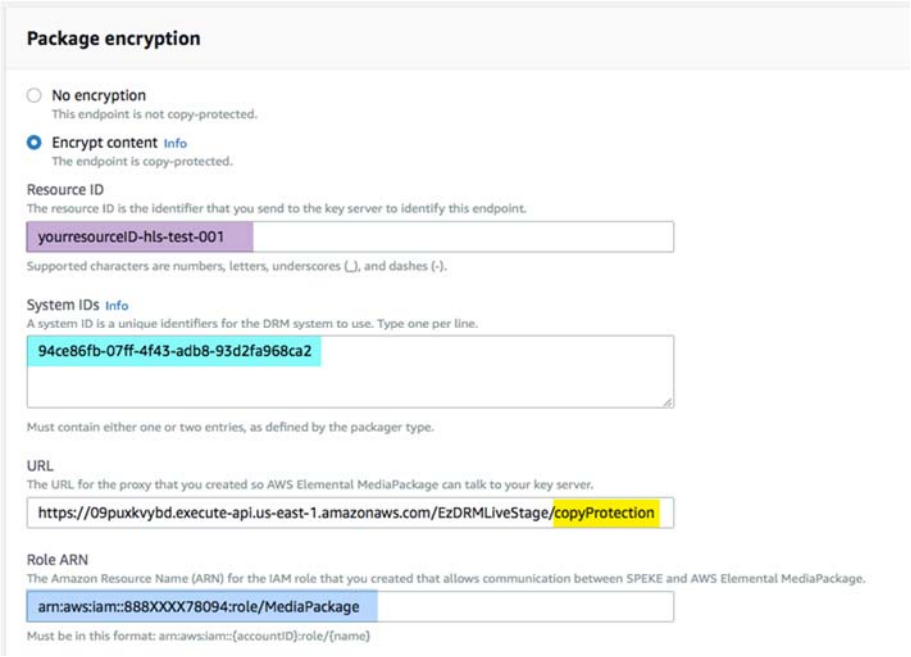
1. Click the **Add** button. Edit the Endpoint **ID** and **Manifest Name** to a unique identifier.



2. Under **Packager Settings**, select the **Type HLS** and leave the other settings as default.



3. Scroll down and select the toggle for **Encrypt Content**.



The parameters are as follows:

- **ResourceID**: this will be the ID that references your DRM Keys. This is a required field.

  Note: *The first time you send a ResourceID to run a job, the ID will be tied to the DRM keys for that job. Jobs can use the same ResourceID to reference the same keys or for new DRM Keys send a new ResourceID. It is best not to use a ResourceID from a failed job.*

- **System ID**: Unique identifiers for the DRM system to use. Insert the System ID for Apple FairPlay, one ID per line:

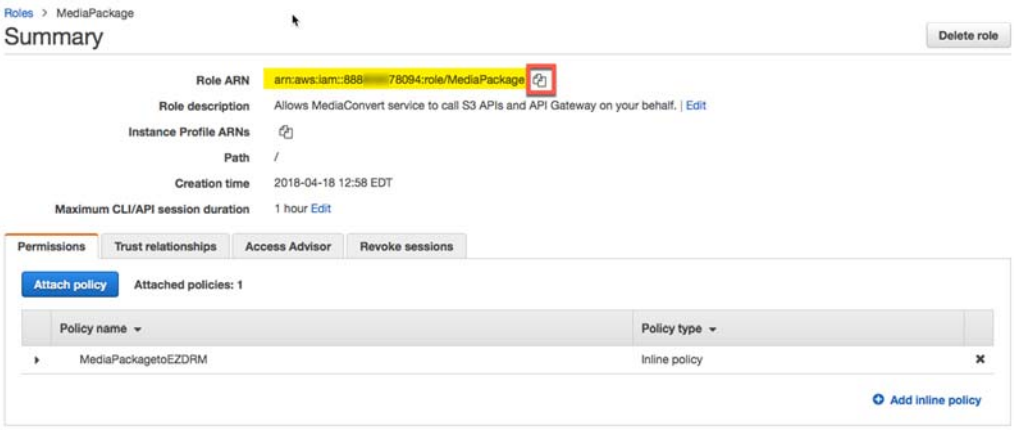
94ce86fb-07ff-4f43-adb8-93d2fa968ca2

Note: *The System ID values need to be lowercase.*

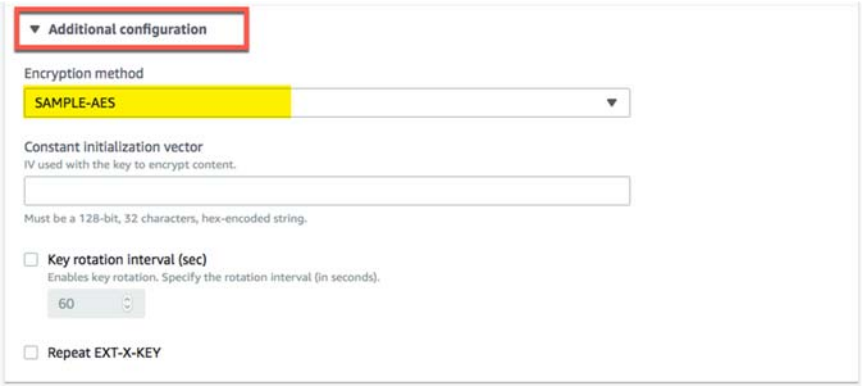
- **URL:** The URL is the **API URL** copied from **Step 9** above, except at to the end of the URL add "**/copyProtection**". (This is case sensitive, be sure **capitalize the P** in Protection.)

  Sample URL:
  https://09puxkvybd.execute-api.us-east-1.amazonaws.com/EzDRMLiveStage/copyProtection

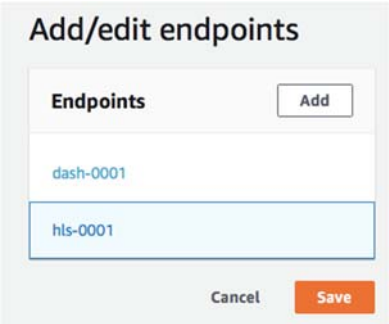
- **Role ARN:** This value is from the **MediaPackage Role** created in Step 6.



4. Under **Additional configuration**, select the **Encryption method: SAMPLE-AES**.



9. Once these settings are completed, click the **Save** button to create the endpoint.

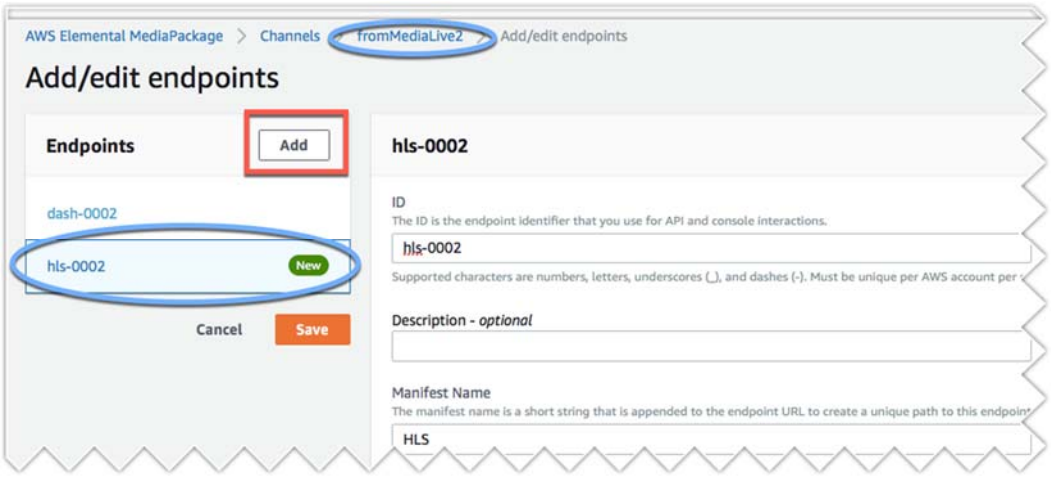


10. Now for redundancy, from your second MediaLive channel, create an Apple HLS endpoint with the same settings as the one we just created, but change the **ID** name to indicate the redundant endpoint.

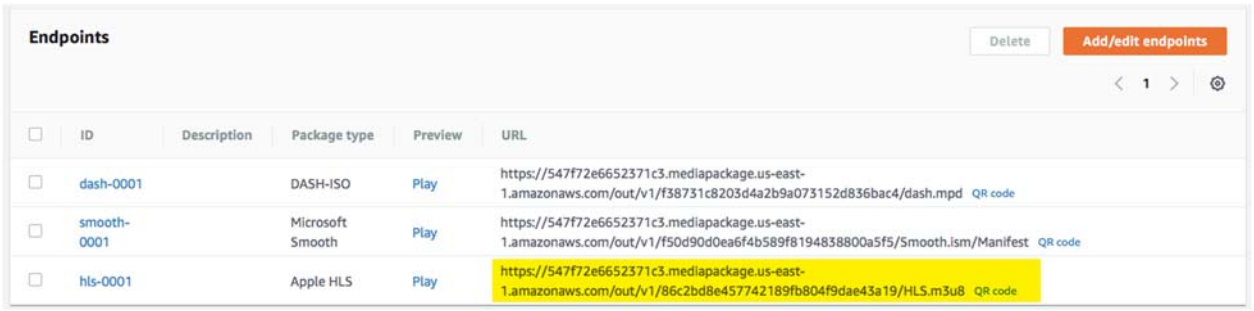
    For this example, we called our first channel **MediaLive1** and created the Apple HLS endpoint **hls-0001**. Under **MediaLive2** we will create a duplicate Apple HLS endpoint but name it **hls-0002**.

    Duplicate ALL the same settings for the second Apple HLS endpoint under the second channel and click **Save**.

    Note: *It is helpful to have multiple tabs open during this process, for ease of copying settings from one channel to the other.*

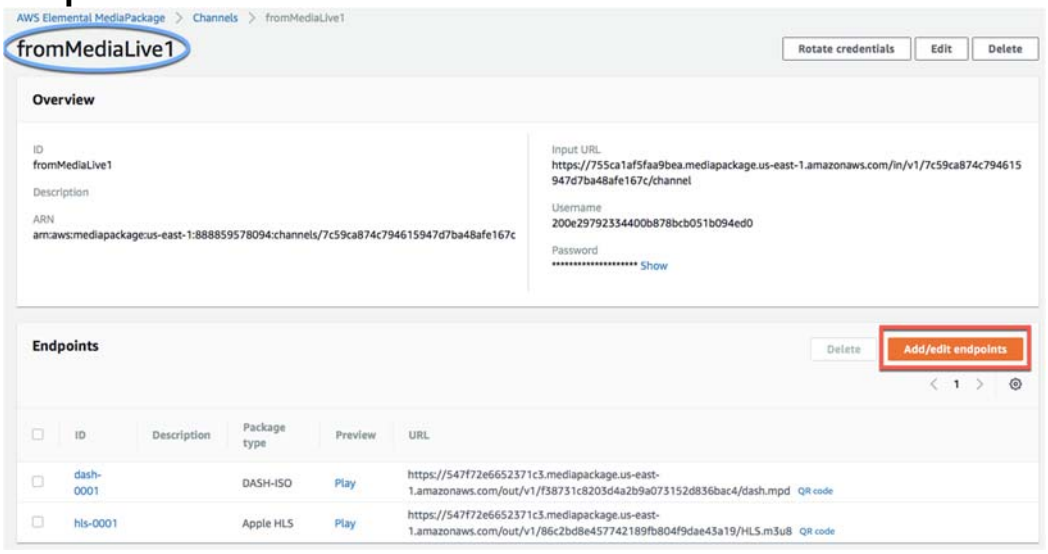


11. Once **MediaLive** is running and publishing to **MediaPackage**, you will be able to access the URL created to play the encrypted Media.

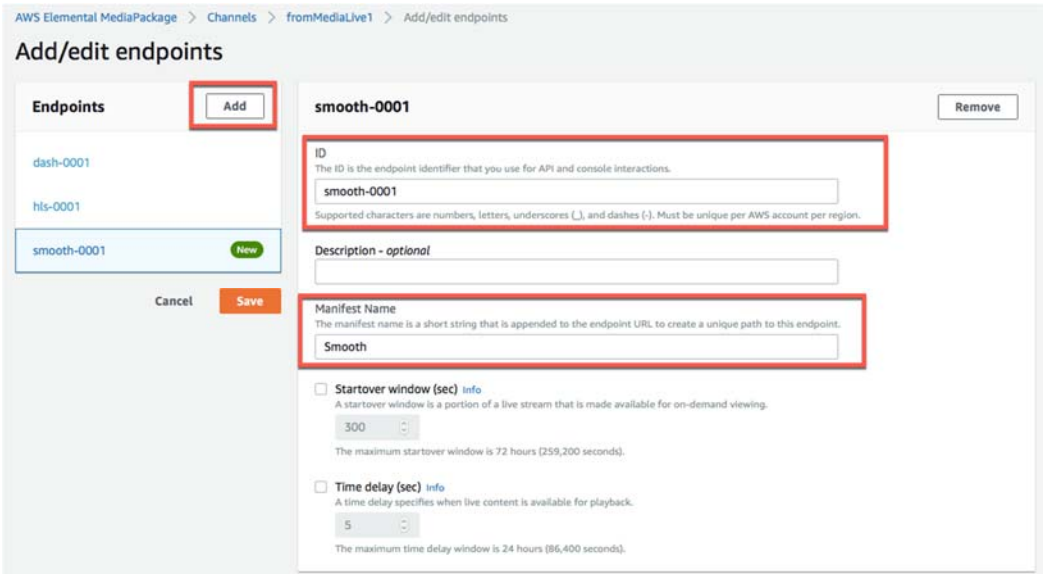

Endpoints
Delete
Add/edit endpoints
1
ID
Description
Package type
Preview
URL
dash-0001
DASH-ISO
Play
https://547f72e6652371c3.mediapackage.us-east-1.amazonaws.com/out/v1/f38731c8203d4a2b9a073152d836bac4/dash.mpd QR code
smooth-0001
Microsoft Smooth
Play
https://547f72e6652371c3.mediapackage.us-east-1.amazonaws.com/out/v1/f50d90d0ea6f4b589f8194838800a5f5/Smooth.ism/Manifest QR code
hls-0001
Apple HLS
Play
https://547f72e6652371c3.mediapackage.us-east-1.amazonaws.com/out/v1/86c2bd8e457742189fb804f9dae43a19/HLS.m3u8 QR code

Microsoft Smooth Streaming

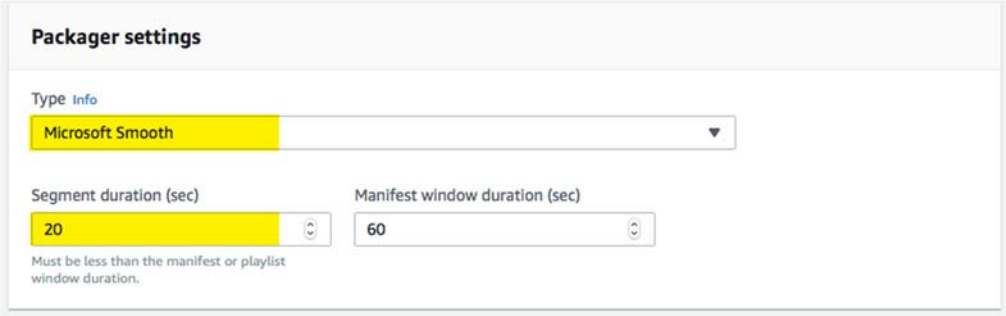
1. In **MediaPackage**, from the first MediaLive channel you created, click the **Add endpoints** button.



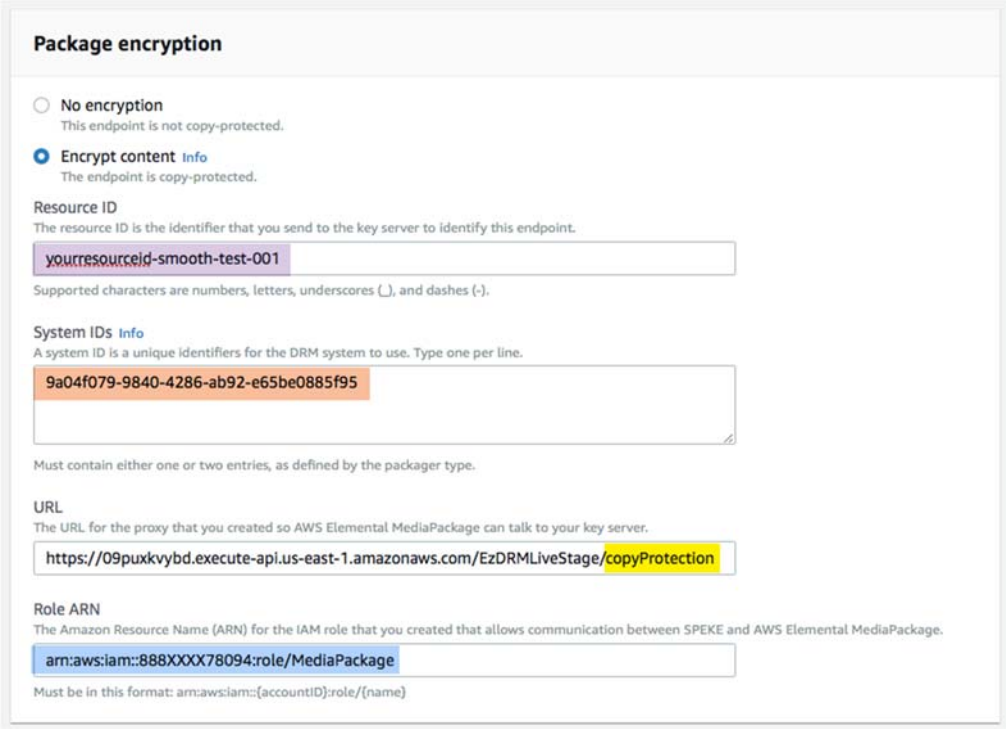
2. Click the **Add** button. Edit the Endpoint **ID** and **Manifest Name** to a unique identifier.



3. Under **Packager Settings**, select the **Type Smooth** and **Segment duration (sec)** to **20** seconds.



4. Scroll down and select the toggle for **Encrypt Content**.



The parameters are as follows:

- **ResourceID**: this will be the ID that references your DRM Keys. This is a required field.

  Note: *The first time you send a ResourceID to run a job, the ID will be tied to the DRM keys for that job. Jobs can use the same ResourceID to reference the same keys or for new DRM Keys send a new ResourceID. It is best not to use a ResourceID from a failed job.*

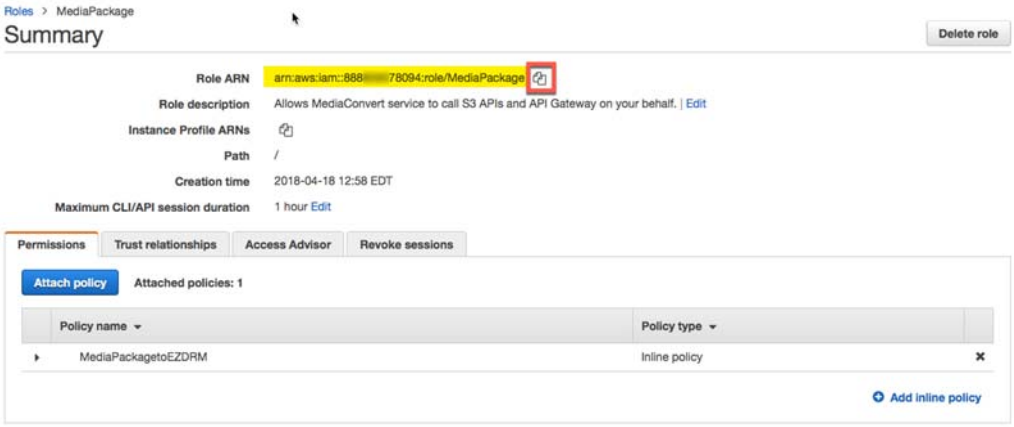
- **System ID**: Unique identifiers for the DRM system to use. Insert the System ID for Smooth Streaming, one ID per line:
  9a04f079-9840-4286-ab92-e65be0885f95

  Note: *The System ID values need to be lowercase.*

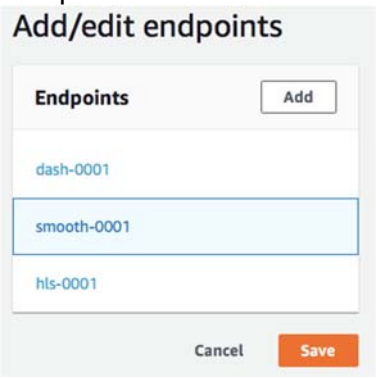
- **URL:** The URL is the **API URL** copied from **Step 9** above, except at to the end of the URL add "**/copyProtection**". (This is case sensitive, be sure **capitalize the P** in Protection.)

  Sample URL:
  https://09puxkvybd.execute-api.us-east-1.amazonaws.com/EzDRMLiveStage/copyProtection

- **Role ARN:** This value is from the **MediaPackage Role** created in Step 6.



5. Once these settings are completed, click the **Save** button to create the endpoint.

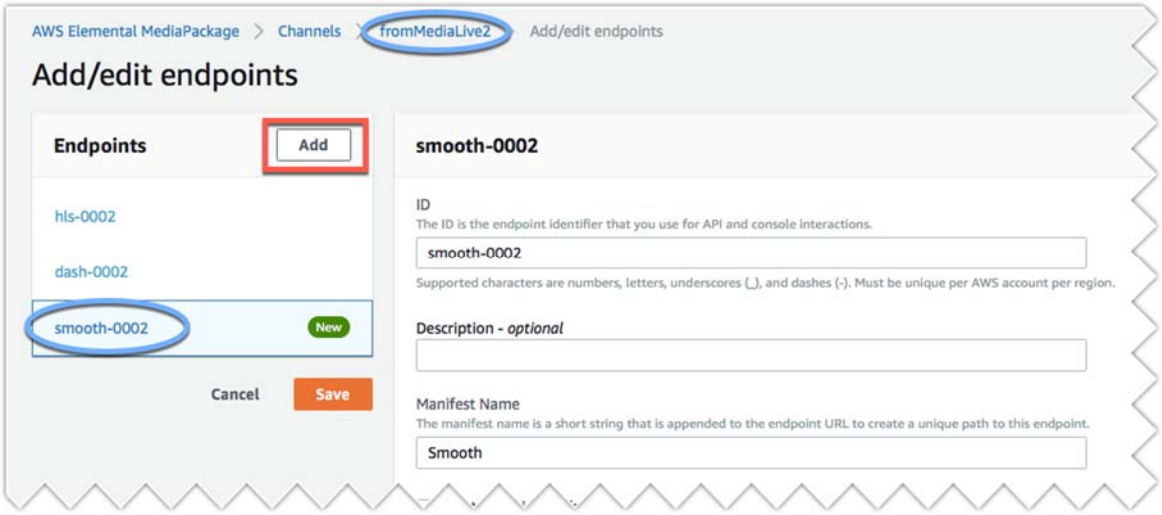


6. Now for redundancy, from your second MediaLive channel, create an Smooth Streaming endpoint with the same settings as the one we just created, but change the **ID** name to indicate the redundant endpoint.

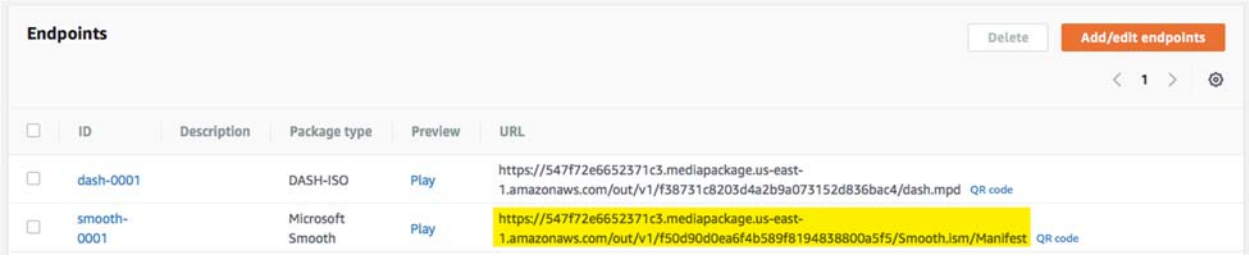
   For this example, we called our first channel **MediaLive1** and created the Smooth Streaming endpoint **smooth-0001**. Under **MediaLive2** we will create a duplicate Smooth Streaming endpoint but name it **smooth-0002**.

   Duplicate ALL the same settings for the second Smooth Streaming endpoint under the second channel and click **Save**.

   Note: *It is helpful to have multiple tabs open during this process, for ease of copying settings from one channel to the other.*



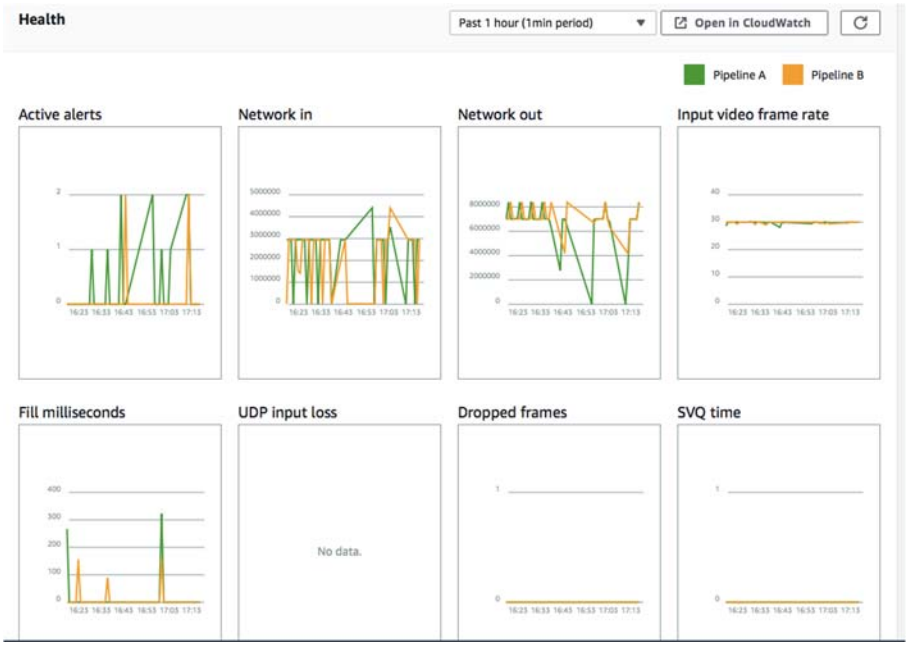
7. Once **MediaLive** is running and publishing to **MediaPackage**, you will be able to access the URL created to play the DRM encrypted Media.



Starting a MediaLive Channel

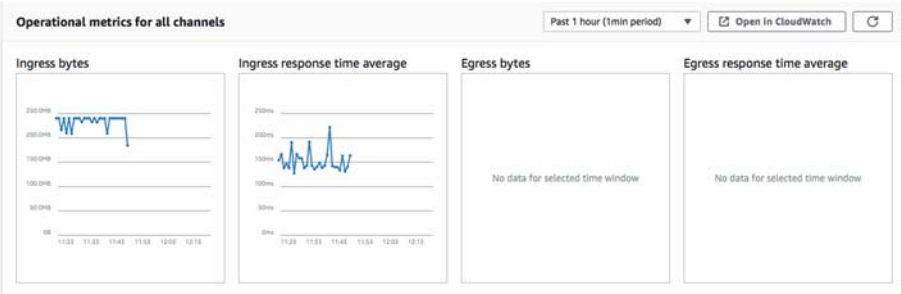
Open **MediaLive** and select the channel. Click the **Start** button to start the channel.

Once the channel is started, data for the stream will be shown in the Health section.



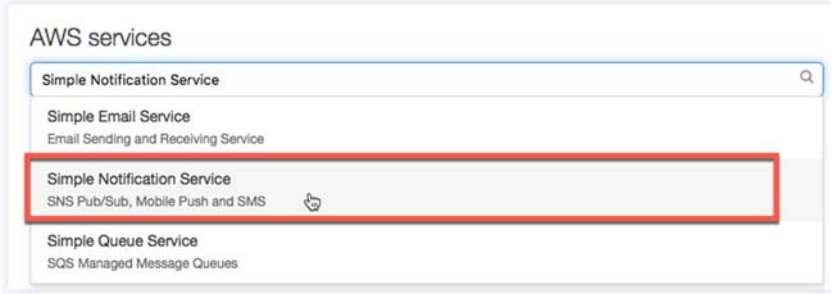
If Input video frame rate is ever not running, you know that there is a problem with the stream.

Same on the **MediaPackage** side, there will be data showing under Operational metrics.

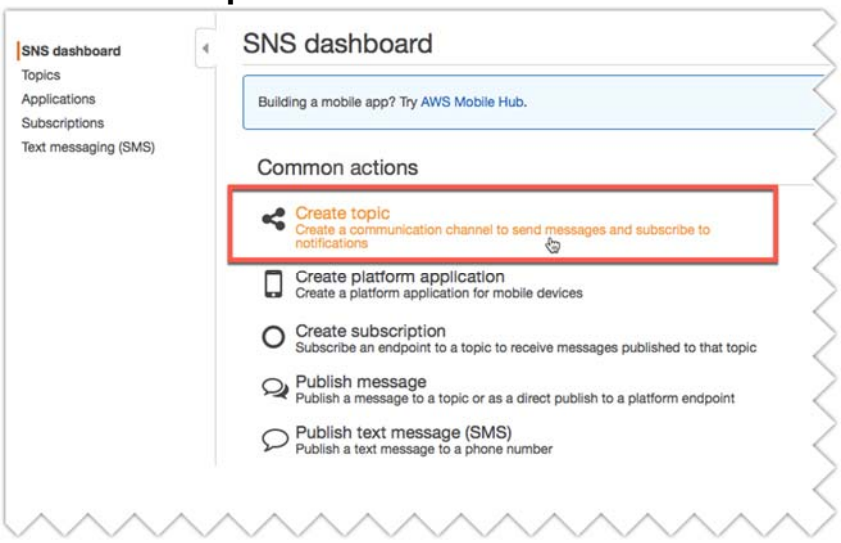


Appendix 1 – Error Log Set-up

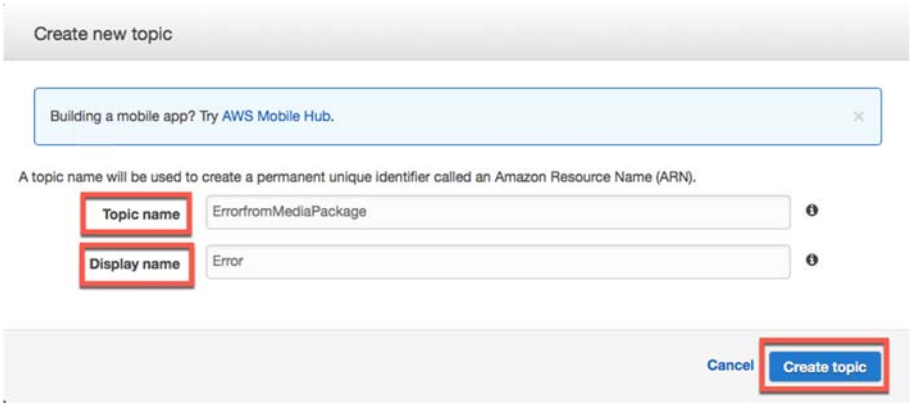
1. To set up an error log, go to **Simple Notification Service** in AWS.



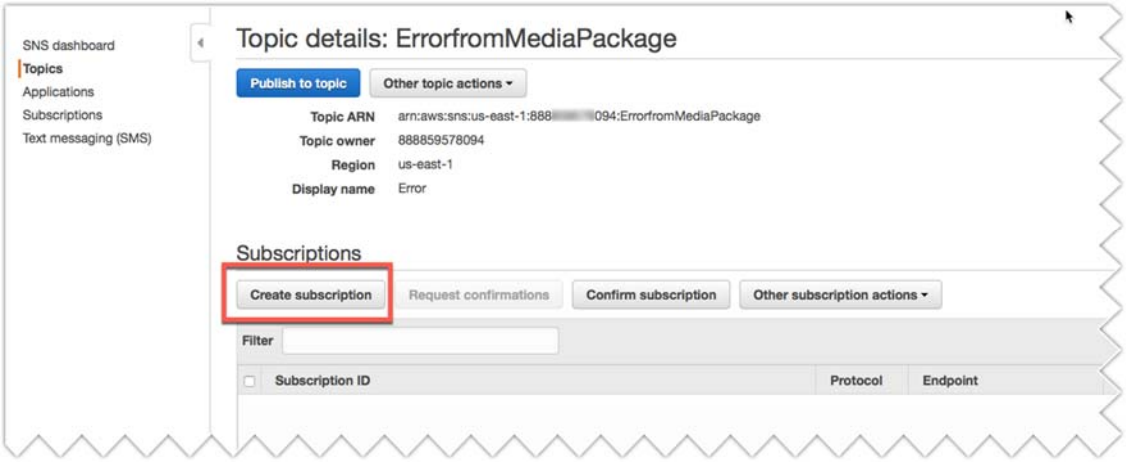
2. Click **Create topic** from the SNS dashboard.



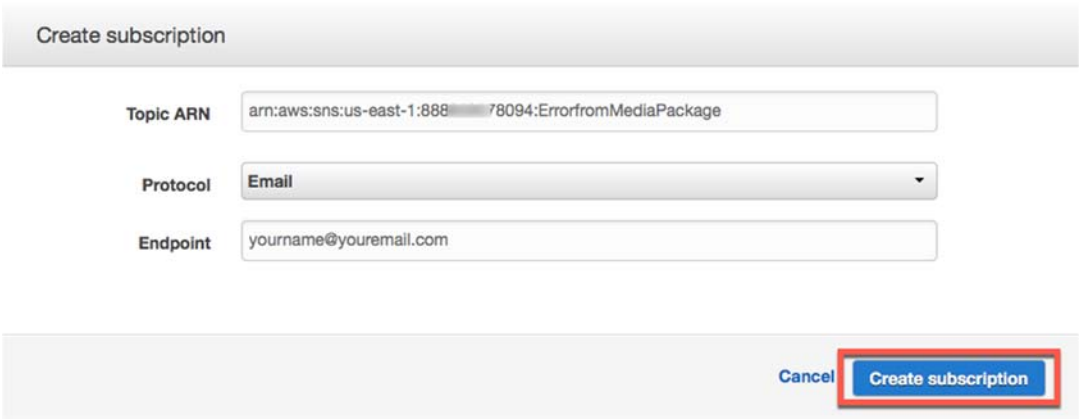
3. Enter the **Topic name** and **Display name** and click **Create topic**.



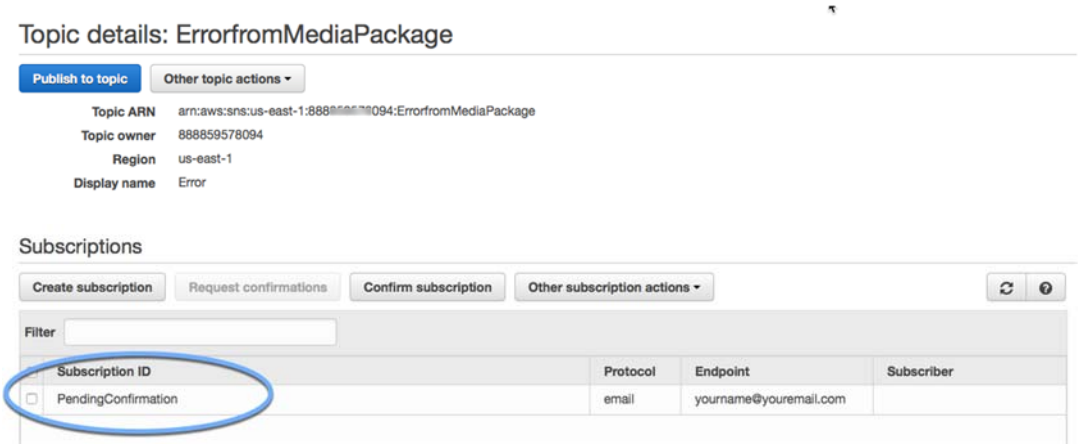
4. The Topic details will open, then click **Create subscription**.



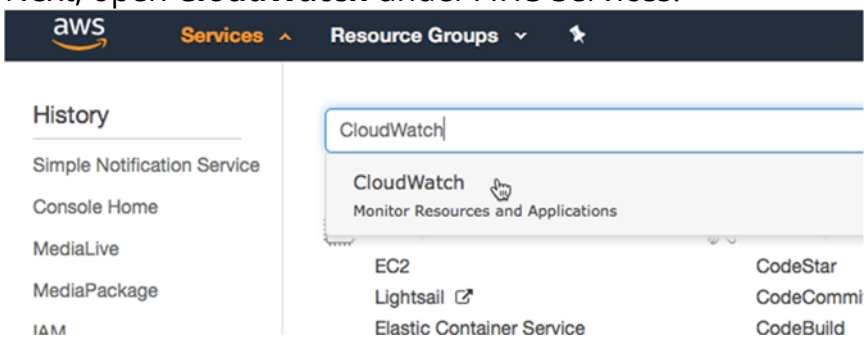
5. Change the **Protocol** to **Email** and enter the **email address** in the **Endpoint** field. Click **Create subscription**.



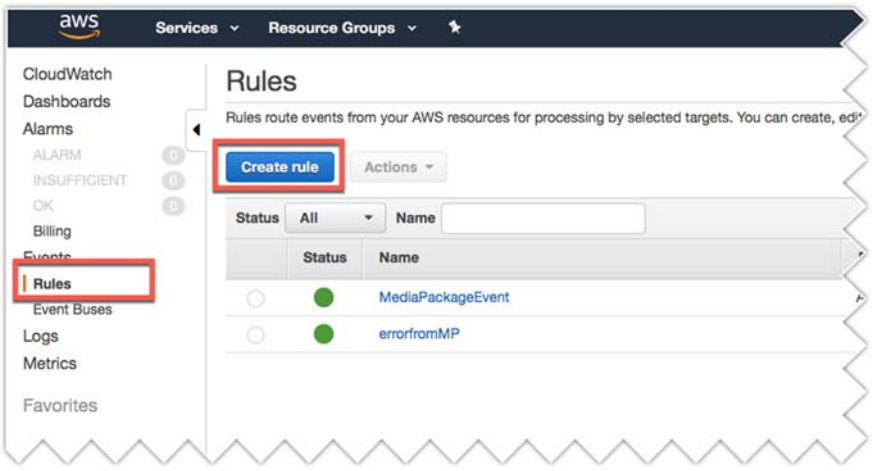
6. There will now be a **Pending Confirmation** line item, and an email will be sent to confirm the subscription.



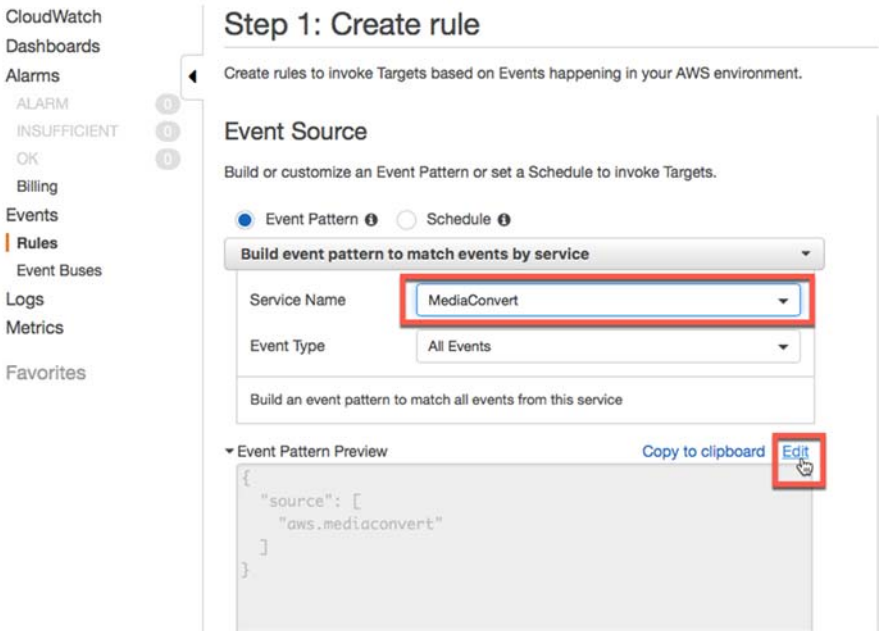
7. Next, open **CloudWatch** under AWS Services.



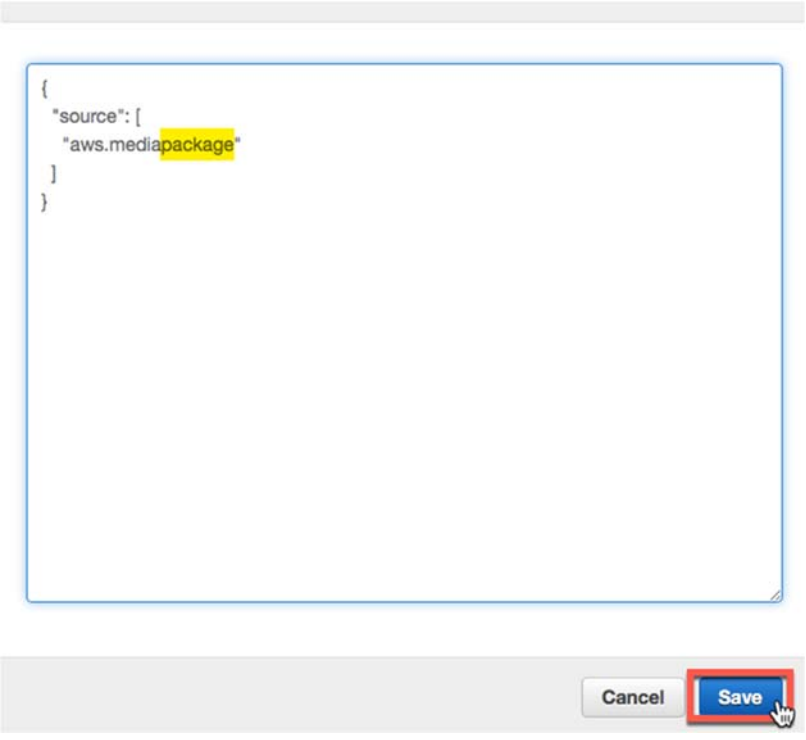
8. Under the **Rules** menu, click **Create rule**.



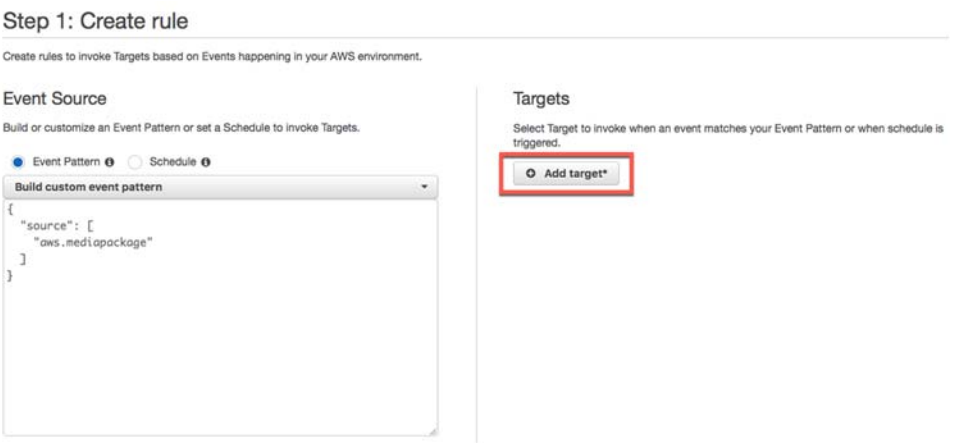
9. Select the **Service Name: MediaConvert** (there isn't currently an option for MediaPackage) and click the **Edit l**ink.



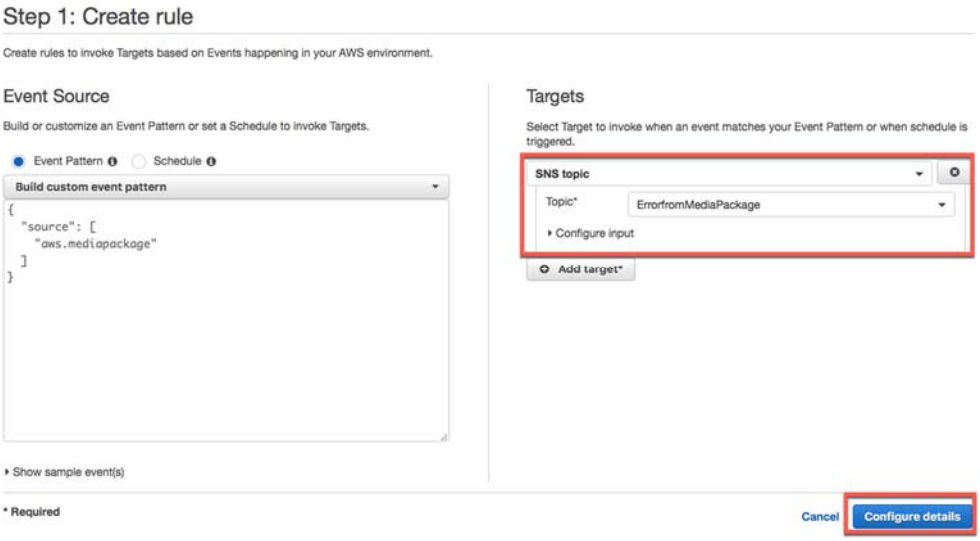
10. Update "aws.mediaconvert" to **"aws.mediapackage"** and click **Save**.



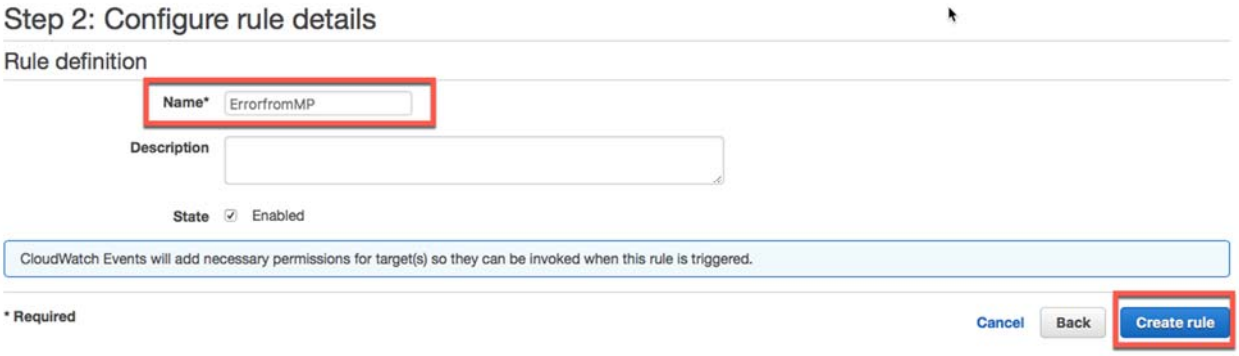
11. Under Targets, click **Add target**.



12. Select **SNS Topic** from the dropdown and select the **Topic** you created in Step 3, for this example "ErrorfromMediaPackage". Then click the **Configure details** button.



13. Enter a **Name** for the rule and click **Create rule**.



You will now get an error message in the event that there is a connection issue.