

EZDRM Integration Zencoder

Table of Contents

Introduction	3
Generating Keys - Widevine/PlayReady	3
<i>Key Value Definitions</i>	4
DRM for Widevine/PlayReady CENC	6
Generating Keys - Apple FairPlay HLS	10
<i>Key Value Definitions</i>	10
Apple FairPlay DRM for HLS	12
Additional Information	15

Version 1.0 / October 2020

Introduction

The following document outlines the steps necessary in order to successfully integrate EZDRM with Zencoder.

The following are required for integration:

1. Zencoder account
2. CURL / Postman knowledge (examples utilize Postman)
3. EZDRM account

Generating Keys - Widevine/PlayReady

Below are the steps to create the DRM Keys for encryption for Widevine and PlayReady.

To request the DRM keys from EZDRM to package the media, there are two options, you can call the EZDRM web service in a browser, or you can script this process with curl or other web service calls.

Option 1: Request DRM keys using EZDRM Web Service

1. Call the EZDRM web service in a browser:
<http://cpix.ezdrm.com/keygenerator/cpix.aspx?k=kid&u=username&p=password&c=may2020->

The parameters are as follows:

Parameter	Description
k	kid or Key ID value (client generated) in GUID format*
u	EZDRM username
p	EZDRM password
c	Content ID – generic resource name/identifier (client generated) – passed into id field

* To generate a GUID for the k value, you can use a GUID generator like the one found here: <http://guid-convert.appspot.com>.

Key Value Definitions

Here are the descriptions of the key values returned by EZDRM:

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<cpix:CPIX xmlns:cpix="urn:dashif:org:cpix" xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc" id="may2020-">
  <cpix:ContentKeyList>
    <cpix:ContentKey kid="582de60c-06ab-41ef-a013-ae533cca5ef8" explicitIV="WC3mDAarQe+gE65TP|pe+A==">
      <cpix:Data>
        <pskc:Secret>
          <pskc:PlainValue>2rEs0PBveCztRAh9PHCLpQ==</pskc:PlainValue>
        </pskc:Secret>
      </cpix:Data>
    </cpix:ContentKey>
  </cpix:ContentKeyList>
</cpix:DRMSystemList>
```

- **id** – c value returned, generic resource name/identifier (client generated)
- **kid** – Key ID in GUID format (client generated)*
- **pskc:Secret key**– the Secret Content Encryption Key in Base 64 generated by EZDRM and returned as a plain value

* To generate a GUID for the k value, you can use a GUID generator like the one found here: <http://guid-convert.appspot.com>.

Here is the example XML return:

```
<cpix:CPIX xmlns:cpix="urn:dashif:org:cpix" xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc" id="may2020-">
  <cpix:ContentKeyList>
    <cpix:ContentKey kid="582de60c-XXXX-XXXX-a013-XXX33cca5ef8" explicitIV="WCXXXXXXXX+gE65TXXxe+A==">
      <cpix:Data>
        <pskc:Secret>
          <pskc:PlainValue>2rEsXXXXXXXXXXAh9PHCLpQ==</pskc:PlainValue>
        </pskc:Secret>
      </cpix:Data>
    </cpix:ContentKey>
  </cpix:ContentKeyList>
```

Option 2: Request DRM keys with curl

The second option to request DRM keys from EZDRM is to script the process with curl or another web service call.

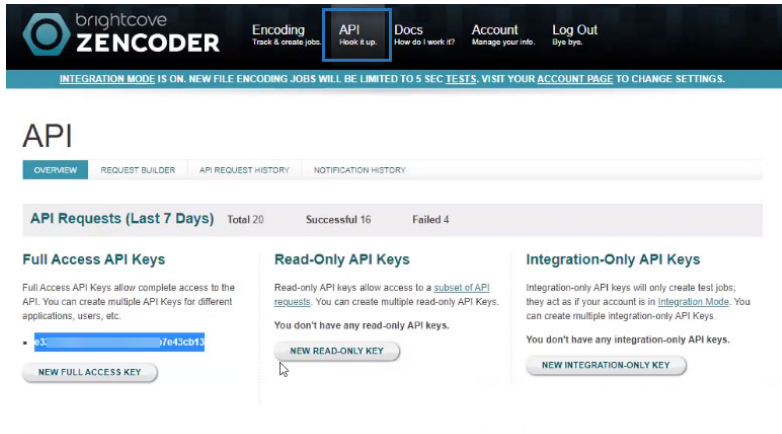
Using EZDRM's web service, the curl script below retrieves the DRM values from the web service.

```
curl -v http://cpix.ezdrm.com/keygenerator/cpix.aspx?k=kid&u=username&p=password&c=may2020-
```

DRM for Widevine/PlayReady CENC

Zencoder API Keys

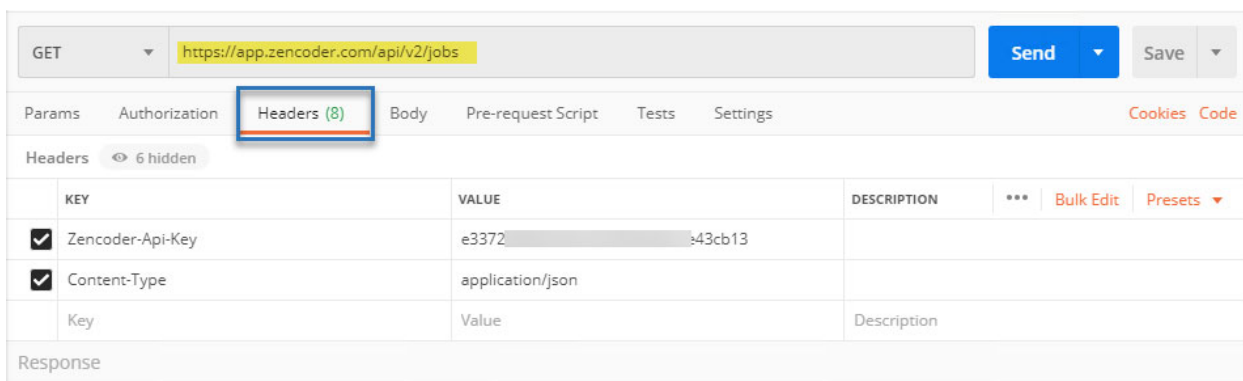
Under API overview for Zencoder, retrieve the API key you will utilize. For this example we utilized the Full Access Key.



POST Request in Postman

In Postman, create POST request to the API <https://app.zencoder.com/api/v2/jobs>

- Under **Headers**, add the following Headers:
 - Zencoder-API-Key – enter your **API Key** in the Value field
 - Content-Type - enter **“application/json”** in the Value field



- Build the Body of the request:
 - Click “Body” Tab and select **raw** and the format is **JSON**.

3. Update the following parameters:

```

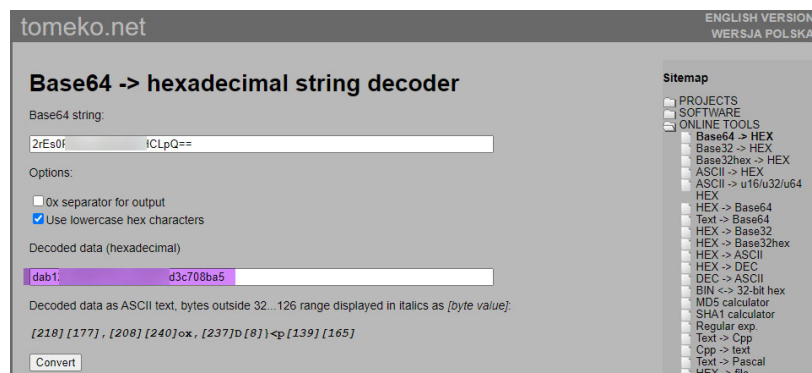
GET https://app.zencoder.com/api/v2/jobs

Params Authorization Headers (10) Body Pre-request Script Tests Settings
none form-data x-www-form-urlencoded raw binary GraphQL JSON

1 {
2   "input": "https://demo.s3.amazonaws.com/zencoder/BigBuckBunny_320x180.mp4",
3   "outputs": [
4     {
5       "streaming_delivery_format": "dash",
6       "video_bitrate": 700,
7       "type": "segmented",
8       "drm": {
9         "method": "cenc",
10        "content_id": "may2020-",
11        "content_key": "dab1XXX-f06f-XXX-ed44-XXX3c708ba5",
12        "key_id": "582dXXX-06ab-XXX-a013-XXX3cca5ef8",
13        "schemas": [
14          {
15            "type": "playready",
16            "license_acquisition_url": "https://playready.ezdrm.com/cency/preauth.aspx?px=XXXXXX"
17          },
18          {
19            "type": "widevine",
20            "provider": "ezdrm",
21            "license_acquisition_url": "https://widevine-dash.ezdrm.com/widevine-php/widevine-foreignkey.php?px=XXXXXX&id=582dXXX-06ab-XXX-a013-XXX3cca5ef8"
22          }
23        ]
24      }
25    }
  }

```

- **input**: location of the file to be encoded (FTP site, S3 bucket, etc. where the .mp4 can be downloaded)
- **content id**: enter the Content ID for the stream (your generic resource/identifier name)
- **content key**: use the **pskc:Secret key** value and decode from Plain Value Base 64 to HEX format in lowercase. An example decoder can be found at: https://tomeko.net/online_tools/base64.php?lang=en



pskc:Secret key (Base 64) = 2rEsXXXXXXXXXXAh9PHCLpO==



(HEX lower case) = dab1XXXXf06fXXXXed4XXXXd3c708ba5



Add in hyphens between characters: *format 8-4-4-4-12*

content key = dab1XXXX-f06f-XXXX-ed4X-XXXd3c708ba5

- **kid**: Key ID in GUID format (client generated)
- **license_acquisition_url**: provided by EZDRM including your **PX Value**.

PlayReady:

<https://playready.ezdrm.com/cency/preauth.aspx?pX=XXXXX>

Widevine:

<https://widevine-dash.ezdrm.com/widevine-php/widevine-foreignkey.php?px=XXXX&kid=582dXXXX-06ab-XXXX-a013-XXXX3cca5ef8>

Note: Your Widevine **PX value** is the last six characters of your Widevine Profile ID. Your PlayReady **PX value** is the last six characters of your PlayReady Profile ID. The appropriate one is required for all packagers you use. For more details on how to find your PX value visit: https://www.ezdrm.com/Documentation/EZDRM_Testing_Playback_v2.pdf.

```
{
  "input": "https://demo.s3.amazonaws.com/zencoder/BigBuckBunny_320x180.mp4",
  "outputs": [
    {
      "streaming_delivery_format": "dash",
      "video_bitrate": 700,
      "type": "segmented",
    }
  ]
}
```

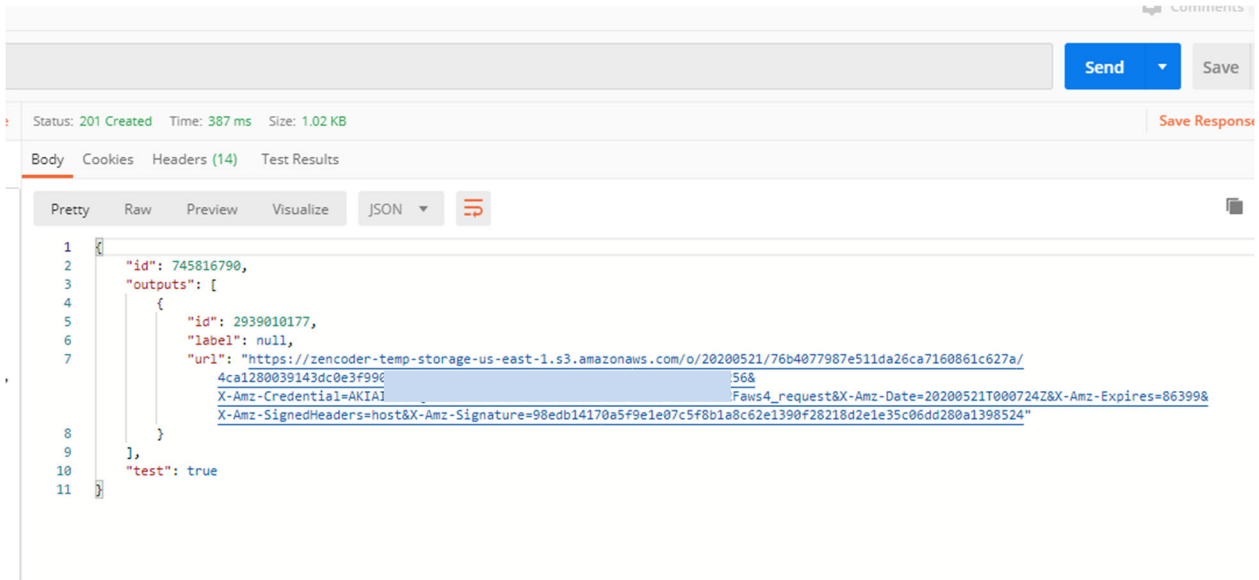


```

"drm": {
  "method": "cenc",
  "content_id": "may2020-",
  "content_key": "dab1XXXX-f06f-XXXX-ed4X-XXXd3c708ba5",
  "key_id": "582dXXXX-06ab-XXXX-a013-XXXX3cca5ef8",
  "schemas": [
    {
      "type": "playready",
      "license_acquisition_url": "https://playready.ezdrm.com/cency/preauth.aspx?
      pX=XXXXX"
    },
    {
      "type": "widevine",
      "provider": "ezdrm",
      "license_acquisition_url": "https://widevine-dash.ezdrm.com/widevine-php/wi
      devine-foreignkey.php?px=XXXX&kid=582dXXXX-06ab-XXXX-a013-XXXX3cca5ef8"
    }
  ]
}

```

4. Click SEND and a temporary URL will be returned by Zencoder.



5. Download the ZIP file and extract to your webserver for playback.

Generating Keys - Apple FairPlay HLS

To request the DRM keys from EZDRM to package the media, there are two options, you can call the EZDRM web service in a browser, or you can script this process with curl or other web service calls.

Option 1: Request DRM keys using EZDRM Web Service

1. Call the EZDRM web service in a browser:

<http://cpix.ezdrm.com/keygenerator/cpix.aspx?k=kid&u=username&p=password&c=may2020->

The parameters are as follows:

Parameter	Description
k	kid or Key ID value (client generated) in GUID format*
u	EZDRM username
p	EZDRM password
c	Content ID – generic resource name/identifier (client generated) – passed into id field

* To generate a GUID for the k value, you can use a GUID generator like the one found here: <http://guid-convert.appspot.com>.

Key Value Definitions

Here are the descriptions of the key values returned by EZDRM:

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<cpix:CPiX xmlns:cpix="urn:dashif:org:cpix" xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc" id="may2020-1">
  <cpix:ContentKeyList>
    <cpix:ContentKey kid="582de60c-06ab-41ef-a013-ae53cca5ef8" explicitIV="wC3nDAarQe+gE65TPWpe+A=">
      <cpix:Data>
        <pskc:Secret>
          <pskc:PlainValue>2rEe0PBveC2tRAh9PHCLpQ=

```

- **id** – c value returned, generic resource name/identifier (client generated)
- **kid** – Key ID in GUID format (client generated)*

- **pskc:Secret key**– the Secret Content Encryption Key in Base 64 generated by EZDRM and returned as a plain value
- **explicitIV** – the Apple FairPlay explicit IV value

* To generate a GUID for the k value, you can use a GUID generator like the one found here: <http://guid-convert.appspot.com>.

Here is the example XML return:

```
<cpix:CPIX xmlns:cpix="urn:dashif:org:cpix" xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc" id="may2020-"
>
<cpix:ContentKeyList>
<cpix:ContentKey kid="582de60c-XXXX-XXXX-a013-XXX33cca5ef8" explicitIV="WCXXXXXXXXX+eF65TXXXe+A==">
<cpix:Data>
<pskc:Secret>
<pskc:PlainValue>2rEsXXXXXXXXXXAh9PHCLp0==</pskc:PlainValue>
</pskc:Secret>
</cpix:Data>
</cpix:ContentKey>
</cpix:ContentKeyList>
```

Option 2: Request DRM keys with curl

The second option to request DRM keys from EZDRM is to script the process with curl or another web service call.

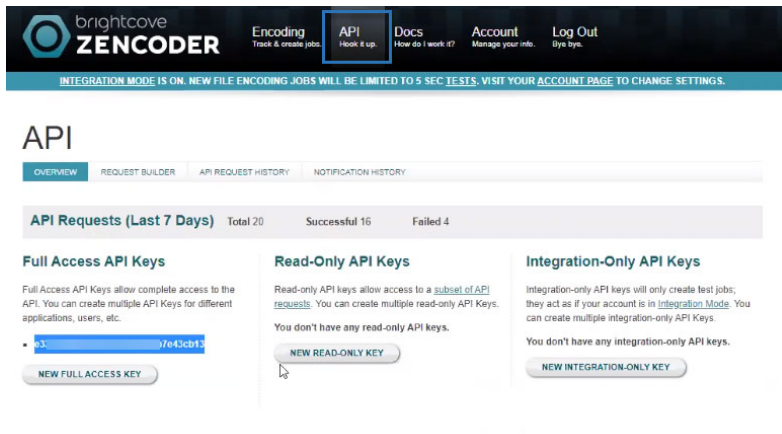
Using EZDRM's web service, the curl script below retrieves the DRM values from the web service.

```
curl -v http://cpix.ezdrm.com/keygenerator/cpix.aspx?k=kid&u=username&p=password&c=may2020-
```

Apple FairPlay DRM for HLS

Zencoder API Keys

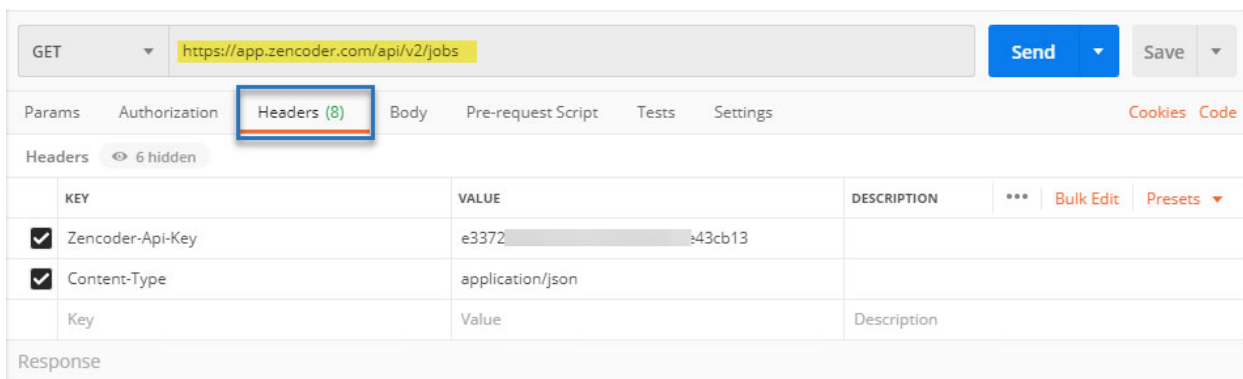
Under API overview for Zencoder, retrieve the API key you will utilize. For this example we utilized the Full Access Key.



POST Request in Postman

In Postman, create POST request to the API <https://app.zencoder.com/api/v2/jobs>

- Under **Headers**, add the following Headers:
 - Zencoder-API-Key – enter your **API Key** in the Value field
 - Content-Type - enter **“application/json”** in the Value field



- Build the Body of the request:
 - Click “Body” Tab and select **raw** and the format is **JSON**.

3. Update the following parameters:

```

1  {
2  "input": "https://demo.s3.amazonaws.com/zencoder/BigBuckBunny_320x180.mp4",
3  "outputs": [
4  {
5  "type": "segmented",
6  "drm": { "method": "fairplay" },
7  "encryption_key_url": "skd://fps.ezdrm.com/;582de60c-XXXX-XXXX-a013-XXX33cca5ef8",
8  "encryption_key": "dab1XXXXf06fXXXXed4XXXXd3c708ba5",
9  "encryption_iv": "582DXXXX06ABXXXXA013XXXXXCCA5EF8"
10 }
11 ]
12 }
13

```

- **input**: location of the file to be encoded (FTP site, S3 bucket, etc. where the .mp4 can be downloaded)
- **encryption_key_url**: build by appending the **kid** value to "skd://fps.ezdrm.com/;" for example: [skd://fps.ezdrm.com/;582de60c-XXXX-XXXX-a013-XXX33cca5ef8](#)
- **encryption_key**: use the **pskc:Secret key** value and decode from Plain Value Base 64 to HEX format in lowercase. An example decoder can be found at: https://tomeko.net/online_tools/base64.php?lang=en

[pskc:Secret key \(Base 64\) = 2rEsXXXXXXXXXXAh9PHCLpO==](#)



[encryption key \(HEX lower case\) = dab1XXXXf06fXXXXed4XXXXd3c708ba5](#)

- **encryption_iv**: use the **explicitIV** and decode from Plain Value Base 64 to HEX format.

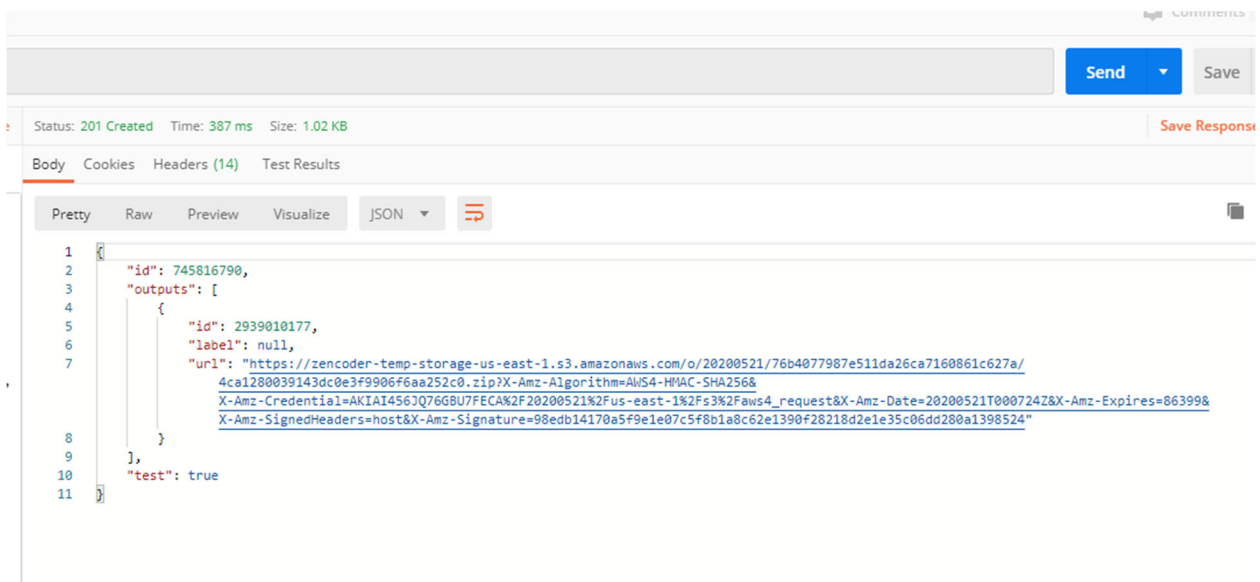
explicitIV (Base 64) = WCXXXXXXXXX+gE65TXXXe+A==



encryption iv (HEX) = 582DXXXX06ABXXXXA013XXXXCCA5EF8

```
{
  "input": "https://demo.s3.amazonaws.com/zencoder/BigBuckBunny_320x180.mp4",
  "outputs": [
    {
      "type": "segmented",
      "drm": { "method": "fairplay" },
      "encryption_key_url": "skd://fps.ezdrm.com/;582de60c-XXXX-XXXX-a013-XXX33cca5ef8",
      "encryption_key": "dab1XXXXf06fXXXXed4XXXXd3c708ba5",
      "encryption_iv": "582DXXXX06ABXXXXA013XXXXCCA5EF8"
    }
  ]
}
```

4. Click SEND and a temporary URL will be returned by Zencoder.



5. Download the ZIP file and extract to your webserver for playback.

Additional Information

For additional questions and comments please contact: simplify@ezdrm.com