EZDRM Qencode Configuration Transcoding

Table of Contents

Version 1.0

Generating DRM Keys

Below are the steps to create the DRM Keys for CENC-Widevine, CENC-PlayReady and Apple Fairplay Streaming encryption for Qencode.

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 CPIX Web Service

1. Call the EZDRM web service in a browser: https://cpix.ezdrm.com/keygenerator/cpix.aspx?**k=kid**&**u=username**&**p=password**&**c=resourcename**

   The parameters are as follows:

Parameter	Description
k=	Key ID value (client generated) in GUID format
u=	Username
p=	Password
c=	Content Resource ID/name, client generated (such as a stream name or media asset name)

   * To generate a GUID for the k value, you can use a GUID generator like the one found here: https://www.guidgenerator.com/.

2. The response from EZDRM will look like this:

- **id** – c value returned, generic resource name/identifier (client generated)
- **kid** – key_id in GUID format (client generated)*
- **pskc:PlainValue 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
- **PSSH** – The modular specific protection system specific header (PSSH) data for the encryption process; Base 64 encoded.
  - **Widevine PSSH** – under systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed"
  - **PlayReady PSSH** – under systemId="9a04f079-9840-4286-ab92-e65be0885f95"

Here Is the example XML return:

```
<cpix:CPIX xmlns:cpix="urn:dashif:org:cpix" xmlns:pskc="urn:ietf:params:xml:ns:keyprov:pskc" id="1">
<cpix:ContentKeyList>
<cpix:ContentKey kid="b99ed9e5-XXXX-XXXX-XXX8-43692b686ddb" explicitIV="uZ7ZXXXXXXXXXXXXX2ht2w==">
<cpix:Data>
<pskc:Secret>
<pskc:PlainValue>Zf7JmBnxxXXXXXXXMI9lSQ==</pskc:PlainValue>
</pskc:Secret>
</cpix:Data>
</cpix:ContentKey>
</cpix:ContentKeyList>
<cpix:DRMSystemList>
```

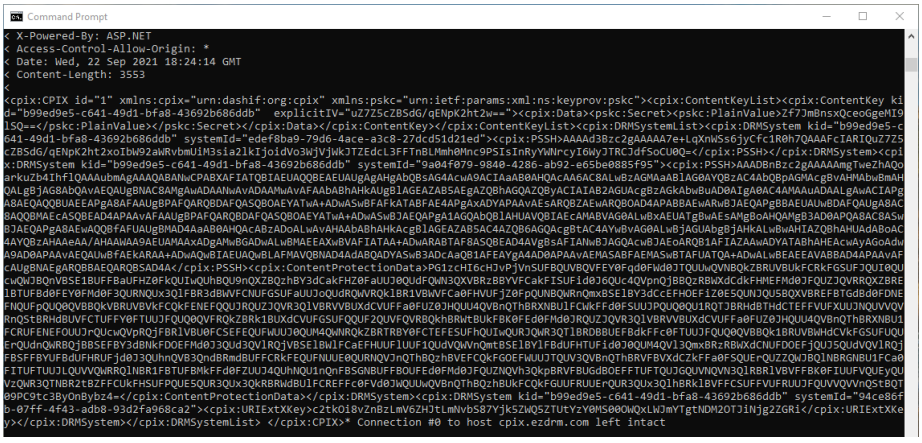
<cpix:DRMSystem kid="b99ed9e5-XXXX-XXXX-XXX8-43692b686ddb" systemId="**edef8ba9-79d6-4ace-a3c8-27dcd51d21ed**"
>
<cpix:PSSH>AAAAd3Bzc2gAAAAA7e+LqXnWSs6jyCfc1R0XXXXXXXXXXXXXXXXXXXXBSdG/qENpK2ht2xoIbW92aWRvbmUiMXXXXXXXXXX
XXXXXXXXXXXEdcL3FFTnBLMmh0Mnc9PSIsInRyYWNrcyI6WyJTRCJdfSoCU0Q=</cpix:PSSH>
</cpix:DRMSystem>
<cpix:DRMSystem kid="b99ed9e5-XXXX-XXXX-XXX8-43692b686ddb" systemId="**9a04f079-9840-4286-ab92-e65be0885f95**"
>
<cpix:PSSH>AAADBnBzc2gAAAAAmgTweZhAQoarkuZb4IhflQAAAubmAgAAAQABANwCPABXAFIATQBIAEUAQQBEAEUAUgAgAHgAbQBsAG4
AcwA9ACIAaAB0AHQAcAA6AC8ALwBzAGMAaABlAG0AYQBzAC4AbQBpAGMAcgBvAHMAbwBmAHQALgBjAG8AbQAvAEQAUgBNAC8AMgAwADAAX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXA
EwARwBJAEQAPgBBAEUAUwBDAFQAUgA8AC8AOQBMAEcASQBEAD4APAAvAFAAUgBPAFQARQBDAFQASQBOAEYATwA+ADwASwBJAEQAPgA1AGQ
AbQBlAHUAVQBIAEcAMABVAG0ALwBxAEUATgBwAEsAMgBoAHQAMgB3AD0APQA8AC8ASwBJAEQAPgA8AEwAQQBfAFUAUgBMAD4AaAB0AHQAc
ABzADoALwAvAHAAbABhAHkAcgBlAGEAZAB5AC4AZQB6AGQAcgBtAC4AYwBvAG0ALwBjAGUAbgBjAHkALwBwAHIAZQBhAHUAdABoAC4AYQB
zAHAAeAA/AHAAWAA9AEUAMAAxADgAMwBGADwALwBMAEEAXwBVAFIATAA+ADwARABTAF8ASQBEAD4AVgBsAFIANwBJAGQAcwBJAEoARQB1A
FIAZAAwADYATABhAHEAcwAyAGoAdwA9AD0APAAvAEQAUwBfAEkARAA+ADwAQwBIAEUAQwBLAFMAVQBNAD4AdABQADYASwB3ADcAaQB1AFE
AYgA4AD0APAAvAEMASABFAEMASwBTAFUATQA+ADwALwBEAEEAVABBAD4APAAvAFcAUgBNAEgARQBBAEQARQBSAD4A</cpix:PSSH>

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.

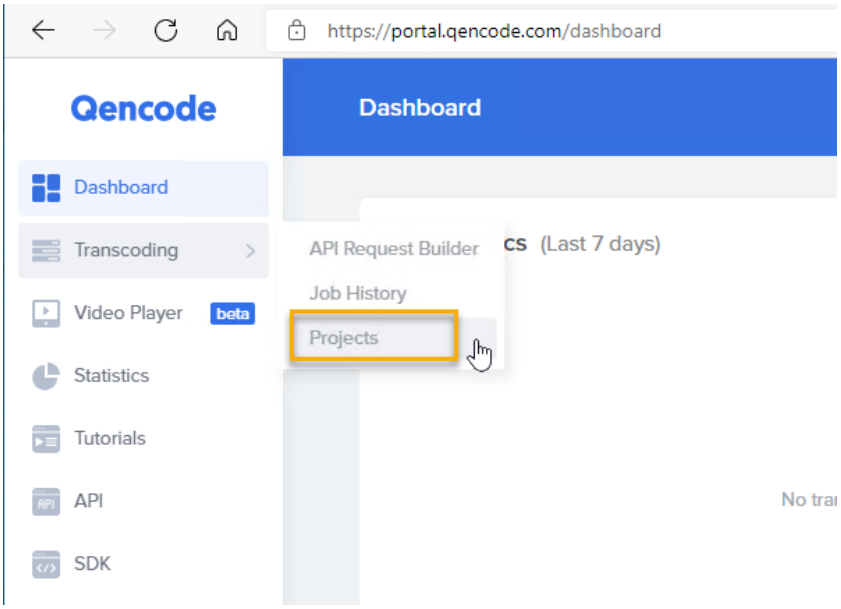
curl -v "https://cpix.ezdrm.com/keygenerator/cpix.aspx?k=kid&u=username&p=password&c=1"



Qencode - Transcoding

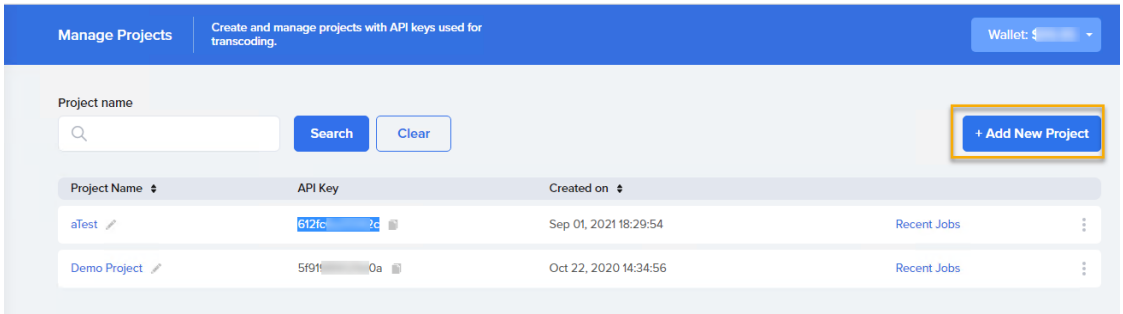
Qencode API Keys

To generate API Requests, copy the **API Key** for authentication from an existing project or create a new project under **Projects** on **Transcoding** Dashboard menu.

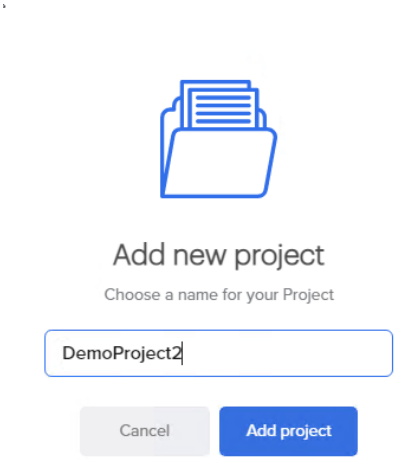


To add a new Project:

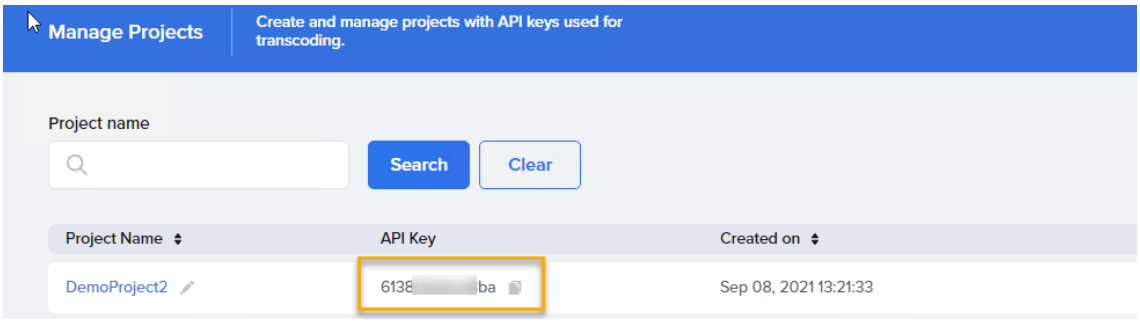
1. In **Manage Projects** page, click **Add New Project**.



2. Enter Project Name and click **Add project**.



3. Copy the **API Key** for the new Project.

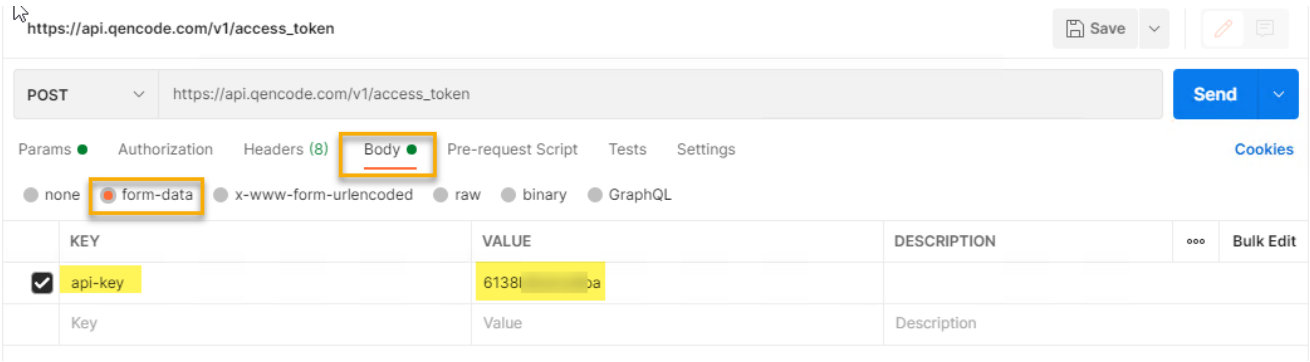


Step 1 - Access Token

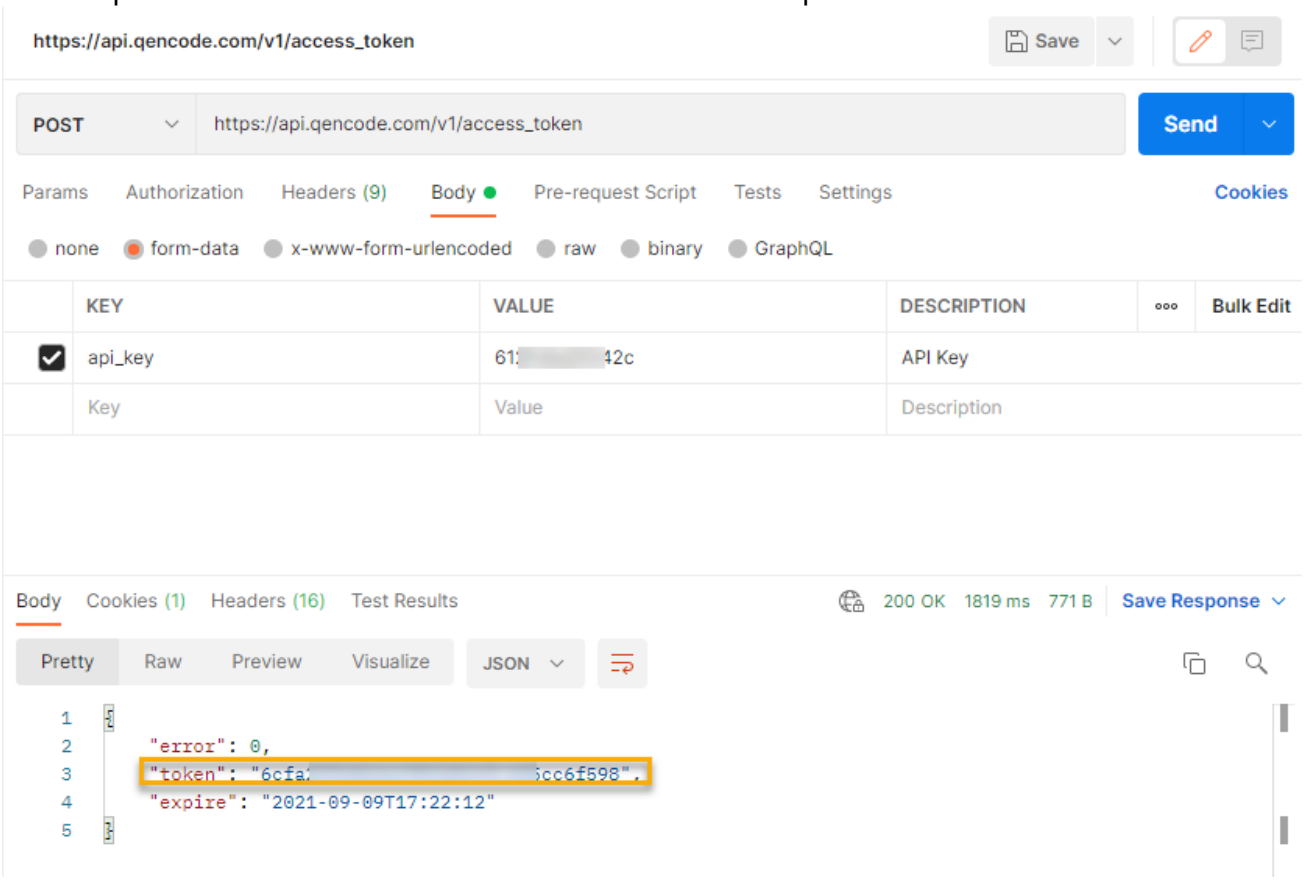
In Postman, create **POST** request to the API
https://api.qencode.com/v1/access_token

Under **Body**, select **form-data** and add the following values:

Key	Value
api_key	The Project **API Key** value



The response will be the Authentication **token** and expiration.



This **token** will be used to create a Task.

Note: *Every Job needs a Task. You cannot run multiple Jobs using the same Task.*

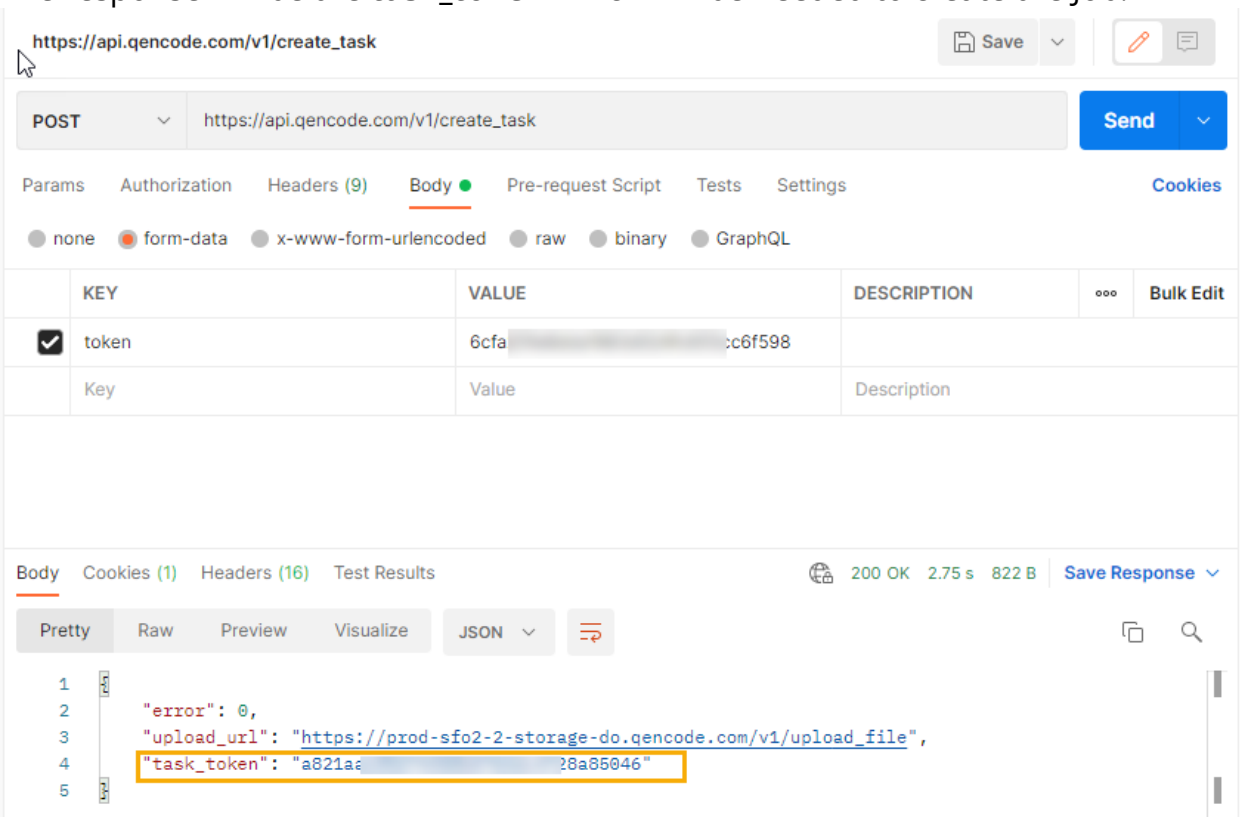
Step 2 - Create Task

In Postman, create **POST** request to the API https://api.qencode.com/v1/create_task

Under **Body**, select **form-data** and add the following values:

Key	Value
token	The **token** value from the Access Token POST

The response will be the **task_token** which will be needed to create the Job.

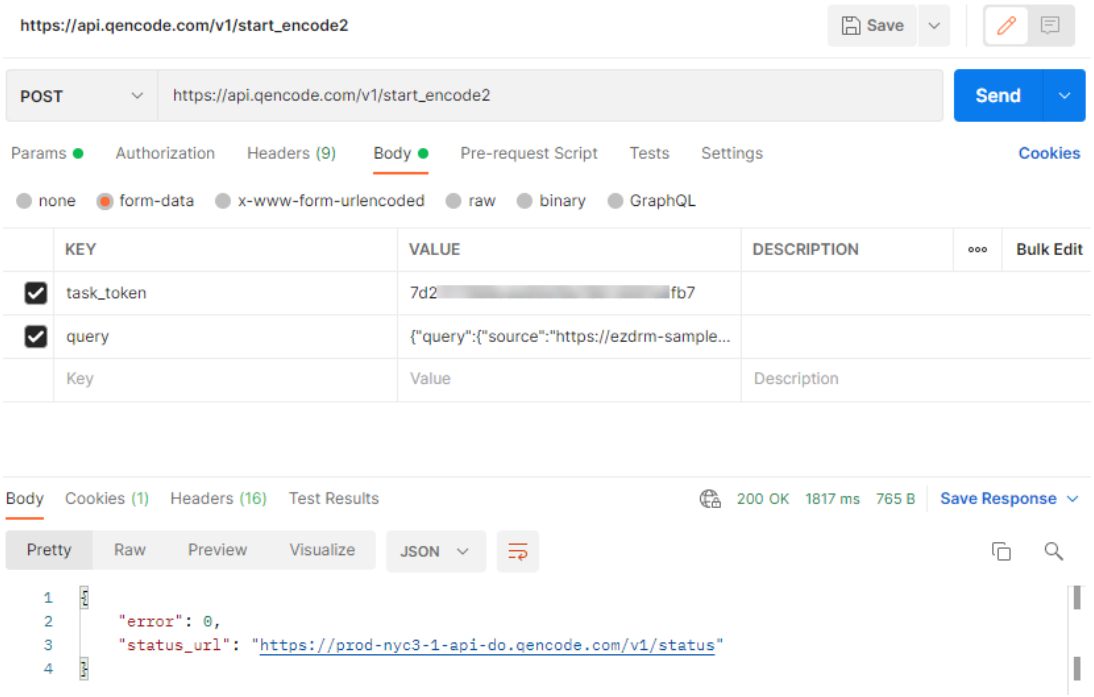


Step 3 - Create Job

In Postman, create **POST** request to the API
https://api.qencode.com/v1/start_encode2

Under **Body**, select **form-data** and add the following values:

Key	Value
task_token	The **task_token** value from Create Task POST
query	JSON format for Job Details



The JSON query format for Job Details is outlined at
https://portal.qencode.com/request_builder.

The sample base output format includes:

```
{"query":{"format":[{"output":"mp4","video_codec":"libx264","height":2160,"quality":23,"audio_bitrate":128
}],"encoder_version":"2","source":"https://nyc3.s3.qencode.com/qencode/samples/1080-sample.mov"}}
```

The next sections of this document detail the **drm** object formatting for Widevine, PlayReady and FairPlay.

Qencode DRM Encoding - Widevine

For DRM encoding, add a '**cenc_drm**' object to your Qencode API request JSON query as shown in this example:

```
"cenc_drm": {
  "key_id" : "b99ed9e5XXXXXXXXXXX843692b686ddb",
  "key"    : "65fecXXXXXecc5071xxXXXXe308f6549",
  "pssh" : "AAAAd3Bzc2gAAAAA7e+LqXnWSs6jyCfc1R0XXXXXXXXXXXXXXXXXXXXBSdG/gENpK2ht2xoIbW92aWRvbmUiMXXXXXXXXXXXXXXXXXXXXXEdcL3FFTnBLMmh0Mnc9PSIsInRyYWNrcyI6WyJTRCJdfSoCU0O="
}
```

Key Value Definitions

- **key_id:** The **kid** used for encryption (also known as KID); Base 64 encoded with no dashes.

  Note: Please make sure you remove dash - characters from key_id value when submitting in Qencode API request since it should be in hex format.

- **key:** The DRM content encryption key (128 bit key); Base 64 encoded.

  For the **key** value use the **pskc:Secret key** value and decode the Plain Value tag from Base 64 to HEX format in lowercase (no dashes). An example decoder can be found at: https://tomeko.net/online_tools/base64.php?lang=en

  pskc:Secret key (Base 64) = Zf7JmBnxxXXXXXXXMI9lSQ==

  ↓

  (KeyHEX) = 65fecXXXXXecc5071xxXXXXe308f6549

- **PSSH** – The modular specific protection system specific header (PSSH) data for the encryption process; Base 64 encoded.

  Note: *For Widevine use cpix:**PSSH** xml tag from systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed"*

You need to add 'cenc_drm' object as a direct child of the format object. Here's the full example:

```
{
  "query": {
    "source": "https://yourserver.com/video.mp4"
    "format": [
      {
        "output": "advanced_dash",
        "destination": {
          ...
        },
        "stream": [
          {
            "video_codec": "libx264",
            "height": 720,
            "width": 1280,
            "audio_bitrate": 128,
            "keyframe": 25,
            "bitrate": 3100
          }
        ],
        "cenc_drm": {
           "key_id" : "b99ed9e5XXXXXXXXXXX843692b686ddb",
           "key"    : "65fecXXXXXecc5071xxXXXXe308f6549",
           "pssh" : "AAAAd3Bzc2gAAAAA7e+LqXnWSs6jyCfc1R0XXXXXXXXXXXXXXXXXXXXBSdG/gENpK2ht2xoIbW92aWRvbmUiM
XXXXXXXXXXXXXXXXXXXXXXEdcL3FFTnBLMmh0Mnc9PSIsInRyYWNrcyI6WyJTRCJdfSoCU0O="
        }
      }
    ]
  }
}
```

Qencode DRM - PlayReady

For DRM encoding, add a '**cenc_drm**' object to your Qencode API request JSON query as shown in this example:

```
"cenc_drm": {
  "key_id" : "b99ed9e5XXXXXXXXXXX843692b686ddb",
  "key"    : "65fecXXXXXecc5071xxXXXXe308f6549",
  "pssh" : "AAADBnBzc2gAAAAAmgTweZhAQoarkuZb4IhflQAAAubmAgAAAQABANwCPABXAFIATQBIAEUAQQBEAEUAUgAgAHgAbQBsAG
4AcwA9ACIAaAB0AHQAcAA6AC8ALwBzAGMAaABlAG0AYQBzAC4AbQBpAGMAcgBvAHMAbwBmAHQALgBjAG8AbQAvAEQAUgBNAC8AMgAwADAA
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
AEwARwBJAEQAPgBBAEUAUwBDAFQAUgA8AC8AQQBMAEcASQBEAD4APAAvAFAAUgBPAFQARQBDAFQASQBOAEYATwA+ADwASwBJAEQAPgA1AG
QAbQBlAHUAVQBIAEcAMABVAG0ALwBxAEUATgBwAEsAMgBoAHQAMgB3AD0APQA8AC8ASwBJAEQAPgA8AEwAQQBfAFUAUgBMAD4AaAB0AHQA
cABzADoALwAvAHAAbABhAHkAcgBlAGEAZAB5AC4AZQB6AGQAcgBtAC4AYwBvAG0ALwBjAGUAbgBjAHkALwBwAHIAZQBhAHUAdABoAC4AYQ
BzAHAAeAA/AHAAWAA9AEUAMAAxADgAMwBGADwALwBMAEEAXwBVAFIATAA+ADwARABTAF8ASQBEAD4AVgBsAFIANwBJAGQAcwBJAEoARQB1
AFIAZAAwADYATABhAHEAcwAyAGoAdwA9AD0APAAvAEQAUwBfAEkARAA+ADwAQwBIAEUAQwBLAFMAVQBNAD4AdABQADYASwB3ADcAaQB1AF
EAYgA4AD0APAAvAEMASABFAEMASwBTAFUATQA+ADwALwBEAEEAVABBAD4APAAvAFcAUgBNAEgARQBBAEQARQBSAD4A"
}
```

Key Value Definitions

- **key_id:** The **kid** used for encryption (also known as KID); Base 64 encoded with no dashes.

  Note: Please make sure you remove dash - characters from key_id value when submitting in Qencode API request since it should be in hex format.

- **key:** The DRM content encryption key (128 bit key); Base 64 encoded.

  For the **key** value use the **pskc:Secret key** value and decode the Plain Value tag from Base 64 to HEX format in lowercase (no dashes). An example decoder can be found at: https://tomeko.net/online_tools/base64.php?lang=en

pskc:Secret key (Base 64) = Zf7JmBnxxXXXXXXXMI9lSQ==

(KeyHEX) = 65fecXXXXXecc5071xxXXXXe308f6549

- **PSSH** – The modular specific protection system specific header (PSSH) data for the encryption process; Base 64 encoded.

  Note: *For PlayReady use cpix:**PSSH** xml tag with systemId="9a04f079-9840-4286-ab92-e65be0885f95"*

You need to add 'cenc_drm' object as a direct child of the format object. Here's the full example:

```
{
  "query": {
    "source": "https://yourserver.com/video.mp4"
    "format": [
      {
        "output": "advanced_dash",
        "destination": {
          ...
        },
        "stream": [
          {
            "video_codec": "libx264",
            "height": 720,
            "width": 1280,
            "audio_bitrate": 128,
            "keyframe": 25,
            "bitrate": 3100
          }
        ],
        "cenc_drm": {
          "key_id" : "b99ed9e5XXXXXXXXXXX843692b686ddb",
          "key"    : "65fecXXXXXecc5071xxXXXXe308f6549",
          "pssh" : "AAADBnBzc2gAAAAAmgTweZhAOoarkuZb4IhflOAAAubmAgAAAOABANwCPABXAFIATOBIAEUAOOBEAEUAUgAgA
HgAbOBsAG4AcwA9ACIAaAB0AHOAcAA6AC8ALwBzAGMAaABlAG0AYOBzAC4AbOBpAGMAcgBvAHMAbwBmAHOALgBjAG8AbOAvAEOAUgBNAC8
AMgAwADAAXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXAEwARwBJAEOAPgBBAEUAUwBDAFOAUgA8AC8AOOBMAEcASOBEAD4APAAvAFAAUgBPAFOAROBDAFOASOBOAEYATwA+ADwASwBJA
```

```
EQAPgA1AGQAbQBlAHUAVQBIAEcAMABVAG0ALwBxAEUATgBwAEsAMgBoAHQAMgB3AD0APQA8AC8ASwBJAEQAPgA8AEwAQQBfAFUAUgBMAD4
AaAB0AHQAcABzADoALwAvAHAAbABhAHkAcgBlAGEAZAB5AC4AZQB6AGQAcgBtAC4AYwBvAG0ALwBjAGUAbgBjAHkALwBwAHIAZQBhAHUAd
ABoAC4AYQBzAHAAeAA/AHAAWAA9AEUAMAAxADgAMwBGADwALwBMAEEAXwBVAFIATAA+ADwARABTAF8ASQBEAD4AVgBsAFIANwBJAGQAcwB
JAEoARQB1AFIAZAAwADYATABhAHEAcwAyAGoAdwA9AD0APAAvAEQAUwBfAEkARAA+ADwAQwBIAEUAQwBLAFMAVQBNAD4AdABQADYASwB3A
DcAaQB1AFEAYgA4AD0APAAvAEMASABFAEMASwBTAFUATQA+ADwALwBEAEEAVABBAD4APAAvAFcAUgBNAEgARQBBAEQARQBSAD4A"
        }
      }
    ]
  }
}
```


Qencode DRM - Apple Fairplay

For DRM encoding, add a '**fps_drm**' object to your Qencode API request JSON query as shown in this example:

```
"fps_drm" : {
  "key" : "65fecXXXXXecc5071xxXXXXe308f6549",
  "iv" : "b99ed9e5cXXXXxxXXXXxx692b686ddb",
  "key_url" : "skd://fps.ezdrm.com/;b99ed9e5-xXXX-XXxX-xxxX-43692b686ddb"
}
```

Key Value Definitions

Here are the descriptions of the items returned:

- **key**: This is the common content encryption key in HEX format (32 hexadecimal characters), the EZDRM **KeyHEX** value.
    - Use the **pskc:Secret key** value and decode from Base 64 to HEX format in lowercase (no dashes). An example decoder can be found at: https://tomeko.net/online_tools/base64.php?lang=en

pskc:Secret key (Base 64) = Zf7JmBnsxXxxxXgeMI9lSQ==

key (HEX) = 65fecXXXXxxxXX071ea0681e308f6549

- **iv**: the initialization vector (IV) used for encryption. This is **explicitIV** value in HEX with no dashes.

  Decode the **explicitIV** Plain Value Base 64 to HEX format. An example decoder can be found at: https://tomeko.net/online_tools/base64.php?lang=en

  explicitIV (Base 64) = uZ7Z5cXXXXXXXXXpK2ht2w==

  iv (HEX no dashes) = b99ed9e5cXXXXxxXXXXxx692b686ddb

- **key_url:** the License URL used for the license key call, this is the EZDRM **KeyURI**.

  Build by appending the **kid** value to base URL "**skd://fps.ezdrm.com/:<kid>**" for example:

  skd://fps.ezdrm.com/:b99ed9e5-xXXX-XXxX-xxxX-43692b686ddb

You need to add 'fps_drm' object as a direct child of the format object. Here's the full example:

```
{
  "query": {
    "source": "https://yourserver.com/video.mp4"
    "format": [
      {
        "output": "advanced_hls",
        "destination": {
          ...
        },
        "stream": [
          {
            "size": "640x360",
            "audio_bitrate": "128"
          },
          {
            "size": "852x480",
            "audio_bitrate": "128"
          },
          {
            "size": "1280x720",
            "audio_bitrate": "320"
          }
        ],
        "fps_drm" : {
          "key" : "65fecXXXXXecc5071xxXXXXe308f6549",
          "iv" : "b99ed9e5cXXXXxxXXXXxx692b686ddb",
          "key_url" : "skd://fps.ezdrm.com/;b99ed9e5-xXXX-XXxX-xxxX-43692b686ddb"
        }
      }
    ]
  }
}
```

Additional Information

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