

EZDRM GPAC Packager DRM Keys Guide

Table of Contents

EZDRM Universal DRM	1
GPAC Prerequisites	3
Universal DRM (Widevine & PlayReady)	3
Generating Keys	3
Key Value Definitions - Widevine & PlayReady	7
Packaging – Widevine and PlayReady	8
Apple FairPlay Streaming	10
Generating Keys	10
Key Value Definitions – Apple FairPlay	12
Packaging for Apple FairPlay Streaming	13
Testing Playback	13
Additional Information	14

Version 1 (published April 4, 2025)

GPAC Prerequisites

This document outlines the raw key format for GPAC Packager and generating DRM keys with EZDRM.

To download visit: <https://gpac.io/downloads/gpac-nightly-builds/>

Universal DRM (Widevine & PlayReady)

EZDRM Universal DRM is a combination of Google Widevine Modular with Microsoft PlayReady; both using linked CENC keys over DASH streaming. This enables a content owner to encrypt the media once with CENC keys and deliver either a PlayReady License or a Widevine License depending on the player and platform calling for a license.

Generating Keys

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 CPIXv2 Web Service

1. Call the EZDRM web service in a browser:
<http://cpix.ezdrm.com/keygenerator/cpix2.aspx?k=kid&u=username&p=password&c=resourcename>

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>.

2. The response from EZDRM will look like this:

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" contentId="" version="2.3">
  <cpix:ContentKeyList>
    <cpix:ContentKey explicitIV="Zuk4yHtHTPevvByJiePx5A==" kid="654938c8-afc0-1c8921e3f1e4" commonEncryptionScheme="cenc">
      <cpix:Data>
        <pskc:Secret>
          <pskc:PlainValue>HGRQkudkLMX9shmg==</pskc:PlainValue>
        </pskc:Secret>
      </cpix:Data>
    </cpix:ContentKey>
  </cpix:ContentKeyList>
  <cpix:DRMSysSystemList>
    <cpix:DRMSysSystem kid="654938c8-afc0-1c8921e3f1e4" systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed">
      <cpix:PSSH>AAAP3Bcz2gAAAA7e+LxXnSs6jycFc1P0h7QAAABBSegH750Pc1Zg6</cpix:PSSH>
      <cpix:ContentProtectionData>PHBzc2ggG1sbN91nVybJptcGVn0jMdBQUBQTD1K0xxwG5XU3M2an1DZmHxUjBoN1FBQUFCOFNFR1ZKT01qY1RveJHyOEFjaVNIajh1UwFCV1Y2WkhKdFHPUGNswNHPc9w3HoPg==
      </cpix:ContentProtectionData>
      <cpix:HLSSignalingData>
        <playlist="media">I0VYVC1YLUFwTpnRVRIT0Q9U0FNUExLUFFUy1DVFIa50VZSUQImRhDGEdGV4dC9ubGFpbjtiYXNlInJ0sQUFBQVazQnpjMmDBQUBQTD1K0xxwG5XU3M2an1DZmHx
        </cpix:HLSSignalingData>
        <playlist="master">I0VYVC1YLVNFUJNJT04t50VZOK1FVEHPRD1TQU1QTEUtQUVTLUjMmDBQUBQTD1K0xxwG5XU3M2an1DZmHxUjBoN1FBQUFCOFNFR1ZKT01qY1RveJHyOEFjaVNIajh
      </cpix:DRMSysSystem>
    <cpix:DRMSysSystem kid="654938c8-afc0-1c8921e3f1e4" systemId="9a04f079-9840-4286-ab92-e65be0885f95">
      <cpix:PSSH>AAACuHbz2gAAAAng1wezHAQoarkuZb41hF1QAAAP1YgAAAQABAI4CPABXAF1ATQ81AEUAQQBEAEUAUgAgAHBQBSAG4AcwA9ABzAC4ABQBpAGNacgBvAHMabwBMAHQALg8jAG8ABQAAEAQAUg
      </cpix:ContentProtectionData>
      <cpix:HLSSignalingData>
        <playlist="master">I0VYVC1YLVNFUJNJT04t50VZOK1FVEHPRD1TQU1QTEUtQUjMmDBQUBQTD1K0xxwG5XU3M2an1DZmHxUjBoN1FBQUFCOFNFR1ZKT01qY1RveJHyOEFjaVNIajh
        </cpix:HLSSignalingData>
        <playlist="media">I0VYVC1YLUFwTpnRVRIT0Q9U0FNUExLUFFUy1DVFIa50VZSUQjFBQUBQVFDt0Fad0Fw03TQUUwQNBQKZBRVUBKFCRKFgSUFJQU10QcWjBQnVBE1BUFFB
      </cpix:SmoothStreamingProtectionHeaderData>
    </cpix:DRMSysSystem>
  </cpix:DRMSysSystemList>
</cpix:CPiX>
```

- **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.
- **PSSH** – Protection System Specific Header value containing DRM specific metadata. Select the value for Widevine or PlayReady using the systemId:
 - PlayReady: 9a04f079-9840-4286-ab92-e65be0885f95
 - Widevine: edef8ba9-79d6-4ace-a3c8-27dcd51d21ed

* 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" contentId="
" version="2.3">
<cpix:ContentKeyList>
<cpix:ContentKey explicitIV="ZUK4yNtXXXXXwByJIePx5A==" kid="654938c8-xXxX-XxXx-afc0-1c8921e3f1e4" commonEn
ryptionScheme="cenc">
<cpix:Data>
<pskc:Secret>
<pskc:PlainValue>hGR0kwdKn/4JXXXXX9shme==</pskc:PlainValue>
</pskc:Secret>
</cpix:Data>
</cpix:ContentKey>
</cpix:ContentKeyList>
<cpix:DRMSysTemList>
<cpix:DRMSysTem kid="654938c8-xXxX-XxXx-afc0-1c8921e3f1e4" systemId="edef8ba9-79d6-4ace-a3c8-27dcd51d21ed"
>
<cpix:PSSH>AAAAP3Bzc2gAAAAA7e+LqXnXXXXXXXXXXR0h7QAAAB8SEGVJOMjBUz3r8AcIShjXXXXXXXXXXJtSOPclzSg</cpix:PSSH
>
<cpix:ContentProtectionData>PHBzc2ggeG1sbnM9InVybjptcGVnOmNlbnM6MjAxXXXXXXXXxVAzQnpjMmdBQUFBQdTd1K0xxwG5XU3
M2an1DZmMxUjBoN1FBQUFCOFNFR1ZKT01qY1RveXXXXXXXXNIajh1UWFVCV1Y2WkhKdFNPUgNswnNHPC9wc3NoPg==</cpix:ContentPr
otectionData>
<cpix:HLSSignalingData playlist="media">I0VYVC1YLUtFWtPnRVRIT0Q9U0FNUEXFLUFFUy1DVFISS0VZSUQ9MHg2NTQ5MzhDOE
RCNEQ0Q0Y3QUZDMDFDODkyMUUZrjFFNCxVUkk9ImRhdGE6dpbjtiYXNlNXXXXXXXjQsQUFBQVAzQnpjMmdBQUFBQdTd1K0xxwG5XU3M2
an1DZmMxUjBoN1FBQUFCOFNFR1ZKT01qY1RvejNyOEFjaVNIajh1UWFVCV1Y2WkhKdFNPUgNswnNHIXLRV1GT1JNQVQ9InVybjp1dWlkOm
VkZWY4YmE5LTc5ZDYtNGFjZS1hM2M4LTI3ZGNkNTFkMXXXXXXXZrK9STUFUVkVSU01PT1M9IjEi</cpix:HLSSignalingData>
<cpix:HLSSignalingData playlist="master">I0VYVC1YLUtFWtPnRVRIT0Q9U0FNUEXFLUFFUy1DVFISS0VZSUQ9MHg2NTQ5MzhDO
ERCNEQ0Q0Y3QUZDMDFDODkyMUUZrjFFNCxVUkk9ImRhdGE6dpbjtiYXNlNXXXXXXXjQsQUFBQVAzQnpjMmdBQUFBQdTd1K0xxwG5XU3M
2an1DZmMxUjBoN1FBQUFCOFNFR1ZKT01qY1RvejNyOEFjaVNIajh1UWFVCV1Y2WkhKdFNPUgNswnNHIXLRV1GT1JNQVQ9InVybjp1dWlkO
mVkZWY4YmE5LTc5ZDYtNGFjZS1hM2M4LTI3ZGNkNTFkMXXXXXXXZrK9STUFUVkVSU01PT1M9IjEi</cpix:HLSSignalingData>
</cpix:DRMSysTem>
<cpix:DRMSysTem kid="654938c8-xXxX-XxXx-afc0-1c8921e3f1e4" systemId="9a04f079-9840-4286-ab92-e65be0885f95"
>
<cpix:PSSH>AAACuHBzc2gAAAAmgTweZhaQoarkuZb4Ihf1QAAAPiYAgAAQAABAI4CPABXAFIATQBIAEUAQQBEAEUAUgAgAHgAbQBsAG4
AcwA9ACIAaAB0AHQAcaA6AC8ALwBzAGMAaABLAG0AYQBzAC4AbQBpAGMACgBvAHMAbwBmAHQALgBjAG8AbQAvAEQAUGBNAC8AMgAwADAAN
wAvADAAMwAvAFaAbABhAHKAUGBlAGEAZAB5AEgAZQBhAGQAZQBvACIAIAB2AGUAcgBzAGkAbwBuAD0AIgA0AC4AMAAuADAALgAwACIAPgA
8AEQAQQBUAEeAPgA8AFAAUgBPAFQARQBDAFQASQBOAEYATwA+ADwASwBFAFkATABFAE4APgAxADYAPAAVAEsARQBZAeWARQBOAD4APABBA
EwARwBJAEQAPgBBAEUAUwBDAFQAUgA8AC8AQQBMAEAcASQBEAD4APAAvFAAUgBPAFQARQBDAFQASQBOAEYATwA+ADwASwBJAEQAPgB5AEQ
AaABKAFoAVQAZAGIAOQAwhkAdgB3AEIAEQBKAekAZQBhAGQANQBBDAD0APQAS8AC8ASwBJAEQAPgA8AEMASABFAEMASwBTAFUATQA+AEgAU
gBPAEUANGA0AFgAUAB5AGQAbwA9ADwALwBDAEgARQBDAEsAUwBVAE0APgA8AEwAQQBFAFUUAUGBMAD4AaAB0AHQAcaBzADoALwAvAHAaAB
hAHKAcbIAGEAZAB5AC4AZQB6AGQAcgBtAC4AYwBvAG0ALwBjAGUAbGJjAHkALwBwAHIAZQBhAHUAdBoAC4AYQBzAHAAEA/AAAAWA9A
EUAMAAxADgAMwBGADwALwBMAEEAXwBVAFIATAA+ADwALwBEAEEXXXXXXXXAvAFcAUgBNAEgARQBBAEQARQBSAD4A</cpix:PSSH>
<cpix:ContentProtectionData>PHBzc2ggeG1sbnM9InVybjptcGVnOmNlbnM6MjAxMyI+QUFBQ3VIQnpjMmdBQUFBQW1nVHdlwHmBUW
9hcmt1WmI0SWhmbBFbQFUFwv1BZ0FBQVFBQkFJNENQQUJYQUZjQVRRQk1BRVwBVUFVCRUFFVUFVZ0FnQUhnQWRJRNBRzRBY3dBOUFDUSFh
QUIwQUhrRQwNBQTZBQzhBTHdCekFHTUFhQUJzQUcwQV1RQnpBQzRBY1FCcEFHTUFjZ0J2QUhNQWJ3Qm1BSFbTGDcAKFH0EFiUUF2QVURQV
```

```
VnQk5BQzhBTWdBd0FEQUF0d0F2QURBQU13QXZBRkFbYkFCaEFiA0FVZ0JsQUdFQVpBQjVBRWdBWlFCaEFHUUFaUUJ5QUNJQU1BQjJBR1VB
Y2dCekFHa0Fid0J1QUQwQUInQTBBQzRBTUFBdUFEQUFMZ0F3QUNJQVBNQThBRVFBVUFVCUUFFRUFQZ0E4QUZBQVnQ1BBR1FBUIFCREFGUU
FTUJJPQUVZQVR3QStBRHdBu3dCRkFGa0FUQUJGQUU0QVBNQXhBRF1BUEFBdkFFc0FSUUJaQUV3QVJRQk9BRDRBUeFCQkFFd0FSd0JKQVVR
QVBNQk1BRVVBVXdCREFGUUFVZ0E4QUM4QVFRQk1BRWNBU1FCRUFENEfQQUF2QUZBQVnQ1BBR1FBUIFCREFGUUFUUJPPQUVZQVR3QStBRH
dBu3dCSkFFUUFQZ0I1QUVRQWFBQktBRm9BV1FBekFHSUFPUUF3QUhrQWRnQjNBRU1BZVFCs0FFa0FaUUJRQUhnQU5RQk1BRDBBUFB0EFD
OEFTd0JKQVVRQVBNQThBRU1BU0FCRkFFTUFTd0JUQUZVQVRRQStBRWdBVWdCUEFFVUF0Z0EwQUZnQVVBQjVBR1FBYndB0UFEEd0FMd0JEQU
VnQVJRQkBRXNBVXdCVkFFMEFQZ0E4QUV3QVFRQmZBR1VBVWdCTUFENEfQhQUIwQUhrQWNBQnpBRG9BTHdBdkFIQUFiQUJoQUhrQWnNmxB
R0VBWkFCNUFDNEFaUUJ2QUdRQWnQnRBQzRBWXdCdKfHMEFMd0JqQUdVQWJnQmpBSGtBTHdCd0FISUFaUUJoQUhVQWRBQm9BQzRBWVFCek
FIQUFIQUEvQUhBQVdBQT1BRVVBTUFB0E4FEZ0FNd0JHQR3QUx3Qk1BRUVBWHdCVkFGSUFUQUERUR3QUx3QkVBRUVBVKFCQkFENEfQQUF2
QUZjQVnQk5BRWdBUIFCQkFFUUFUUJQUUQ0QTWvcHNzaD48cHJvIHhtbG5zPSJ1cm46bWljcm9zb2Z0OnBsYX1yZWfkeSI+bUFJQUFBRC
cpix:ContentProtectionData>
```

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/cpix2.aspx?k=kid&u=username&p=password&c=drm-001
```

Key Value Definitions - Widevine & PlayReady

- **key_id**: The **kid** used for encryption (also known as KID); Base 64 encoded with no dashes.
- **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) = ZUk4yNtXXXXXwByJlePx5A==



(KeyHEX) = 8464509XXXXa9ffXXXXe4d4c5fdb219a

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

For the **PSSH** value use the **cpix:PSSH** value and decode from Base 64 to HEX format in lowercase (no dashes).

Select the value for Widevine or PlayReady using the systemId:

- PlayReady: 9a04f079-9840-4286-ab92-e65be0885f95
- Widevine: edef8ba9-79d6-4ace-a3c8-27dcd51d21ed

cpix:PSSH(Base64)=AAAAP3Bzc2gAAAAA7e+LqXnXXXXXXXXXXR0h7QAAAB8SEGVJOMjbTUz3r8AciSHjXXXXXXXXXXJtSOPclZsG



Cpix:PSSH(HEX)=0000003F7073736800000000EDEF8BA979D64ACEA3C827DCD51D21ED000001XXXXXXXXXX8C8DB4D4CF7AFC01C8921E3F1E41A05657A64726D48E3DC959B06

Packaging – Widevine and PlayReady

Place values in an XML document “**cenc.xml**”. Below is the example:

```
<?xml version="1.0" encoding="UTF-8" />
<GPACDRM type="CENC AES-CTR">

<!-- Widevine -->
<DRMInfo type="pssh" version="0">
  <BS ID128="edef8ba979d64acea3c827dcd51d21ed"/> <!-- System ID -->
  <BS data="0000003F7073736800000000EDEF8BA954938C8DB4D4CF7AFC01C8921E3F1E41A05657A647" />
</DRMInfo>

<!-- Playready -->
<DRMInfo type="pssh" version="0">
  <BS ID128="9a04f07998404286ab92e65be0885f95"/> <!-- System ID -->
  <BS
data="000002B87073736800000009A04F07998404286AB92E65BE570052004D0048004500410044004
6F0073006F00660074002E0063006F006D002F00440052004D002FC00610079005200650061006400790
00540041003E003C00500052004F00540045004300540049004E004
52004F00540045004300540049004E0046004F003E003C004B00490044003E007900440068004A005A00550033006200390030007900760077004200790
003600340058005000790064006F003D003C002F0043004800450043004800530055004D003E003C004C0041005F00550052004C003E006800740074007
79002F0070007200650061007500740068002E0061007300700078003F00700058003D004500300031003800330046003C002F004C0041005F0055005200
</DRMInfo>










<CrypTrack trackID="1" IsEncrypted="1" IV_size="16" first_IV="0x654938C8DB4D4CF7AFC01C8921E3F1E4" saiSavedBox="senc">
<key KID="0x654938C8DB4D4CF7AFC01C8921E3F1E4" value="0x84645C5FDB219A"/>
</CrypTrack>

</GPACDRM>
```

To package for playback for DASH use the following CLI:

```
gpac -i d:\gpac\bunny.mp4 cecrypt:cfile=d:\gpac\cenc.xml -o d:\gpac\dash.mpd
```

```
C:\Program Files\GPAC>gpac -i c:\gpac\bunny.mp4 cecrypt:cfile=c:\gpac\cenc.xml -o c:\gpac\dash.mpd
[CENCrypt] Missing track crypt info in DRM config file, PID A2 will not be crypted
[Dasher] Segment 1 duration 10 more than 3/2 DASH duration, consider reencoding or using segment timeline
[Dasher] First CTS 900000 in segment 2 drifting by 9 (more than half a segment duration) from segment time, consider ree
ncoding or using segment timeline
[Dasher] Segment 2 duration 10 more than 3/2 DASH duration, consider reencoding or using segment timeline
[Dasher] First CTS 1800000 in segment 3 drifting by 18 (more than half a segment duration) from segment time, consider r
eencoding or using segment timeline
[Dasher] Segment 3 duration 2.70833 more than 3/2 DASH duration, consider reencoding or using segment timeline
[Dasher] First CTS 2043750 in segment 4 drifting by 19.7083 (more than half a segment duration) from segment time, consi
der reencoding or using segment timeline
```


 bunny_dash_track2_114.m4s	3/25/2025 9:55 AM	M4S File	21 KB
 bunny_dash_track2_115.m4s	3/25/2025 9:55 AM	M4S File	20 KB
 bunny_dash_track2_116.m4s	3/25/2025 9:55 AM	M4S File	20 KB
 bunny_dash_track2_117.m4s	3/25/2025 9:55 AM	M4S File	20 KB
 bunny_dash_track2_118.m4s	3/25/2025 9:55 AM	M4S File	21 KB
 bunny_dash_track2_119.m4s	3/25/2025 9:55 AM	M4S File	20 KB
 bunny_dash_track2_120.m4s	3/25/2025 9:55 AM	M4S File	21 KB
 bunny_dash_track2_init	3/25/2025 9:54 AM	MP4 File	1 KB
 bunny_protected	2/7/2025 10:56 AM	MP4 File	32,627 KB

Apple FairPlay Streaming

EZDRM Apple FairPlay DRM is a hosted Apple FairPlay Streaming (DRM). This enables a content owner to encrypt the media with Apple FPS DRM keys and deliver content Apple devices with native support MAC Safari browser via HTML 5 player or iOS via native App or Safari 11.3.

The packaging process encrypts the media. This is accomplished via a secure web call to the EZDRM Key Servers API. The Key Server API will return an XML response with the DRM key structure.

Generating Keys

Option 1: Request DRM keys using EZDRM CPIX Web Service

3. Call the EZDRM web service in a browser:
<http://cpix.ezdrm.com/keygenerator/cpix2.aspx?k=kid&u=username&p=password&c=resourceName>

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>.

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="hyb-001">
  <cpix:ContentKeyList>
    <cpix:ContentKey kid="000ae000-533cca5ef8" explicitIV="AArgA" >?Hpe+A==">
      <cpix:Data>
        <pskc:Secret>
          <pskc:PlainValue>q4sp <GmTQ==</pskc:PlainValue>
        </pskc:Secret>
      </cpix:Data>
    </cpix:ContentKey>
  </cpix:ContentKeyList>
  <cpix:DRMSysList>
    <cpix:DRMSys kid="000ae000-533cca5ef8" systemId="edef8ba9-79c;-27dcd51d21ed">
      <cpix:PSSH>AAAAAdnBzc2j,XnwSs6jyCfc1R0h7QAAAFYIARIQAArgAAarQe+oIbW92awRvbmUiMnsia2lkIjoiQUFyZ0FBYXJRZStnRTY1VFBNcGUrQ
      <cpix:ContentProtectionData/>
    </cpix:DRMSys>
  </cpix:DRMSysList>
</cpix:CPIX>
```

- **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" contentId="
" version="2.3">
  <cpix:ContentKeyList>
    <cpix:ContentKey explicitIV="Zuk4yNtXXXXXwBvJIePx5A==" kid="654938c8-xXxX-xXxX-afC0-1c8921e3f1e4" commonEn
    cryptationScheme="cenc">
      <cpix:Data>
        <pskc:Secret>
          <pskc:PlainValue>hGR0kwdKn/4JXXXXX9shmg==</pskc:PlainValue>
        </pskc:Secret>
      </cpix:Data>
    </cpix:ContentKey>
  </cpix:ContentKeyList>
  <cpix:DRMSysList>
```

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/cpix2.aspx?k=kid&u=username&p=password&c=hyb-001
```

Key Value Definitions – Apple FairPlay

Here are the descriptions of the key values returned:

- **key_id**: The **kid** used for encryption (also known as KID); Base 64 encoded with no dashes (-).
- **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) = sU+A8UXXXXvSN0aXXXXwcg==



(KeyHEX) = b14fXXXX41b77XXXX2374XXXXe2f30XX

- **iv: explicitIV** decoded from Base 64 to HEX) combined.

Decode the **explicitIV** Plain Value Base 64 to HEX format.

explicitIV (Base 64) = WCq8DXXXXX+gE65TXXXX+A==



iv (HEX no dashes) = 582aXXXXXXXXX41efXXXae533ccaXXXX

- **KeyURI** - Use the command line option **--encryption-key-uri** to specify the license URL for encryption. Build by appending the **kid** value to base URL "skd://fps.ezdrm.com/;" for example:

<skd://fps.ezdrm.com/582de60c-XXXX-XXXX-a013-XXX33cca5ef8>

Packaging for Apple FairPlay Streaming

Place values in an XML document “cbcs.xml”. Below is the example:

```
<?xml version="1.0" encoding="UTF-8" />
<GPACDRM type="CENC AES-CBC">

<CrypTrack trackID="1" IsEncrypted="1" IV_size="16" first_IV="0x65... )1C8921E3F1E4" saiSavedBox="senc">
<key KID="0x654938: \FC01C8921E3F1E4" value="0x8464509: 7E4D4C5FDB219A" hlsInfo="URI="skd://654938c8-
afc0-1c8921e3f1e4:654938C: 21E3F1E4",KEYFORMAT="com.apple.streamingkeydelivery",KEYFORMATVERSIONS="1"/>
</CrypTrack>

</GPACDRM>
```

The example to package for playback in HLS:

```
gpac -i d:\gpac\bunny.mp4 cecrypt:cfile=d:\gpac\cbcs.xml -o d:\gpac\hls.m3u8
```

```
C:\Program Files\GPAC>gpac -i c:\gpac\bunny.mp4 cecrypt:cfile=c:\gpac\cbcs.xml -o c:\gpac\hls.m3u8
[CENCrypt] Missing track crypt info in DRM config file, PID A2 will not be crypted
[Dasher] Segment 1 duration 10 more than 3/2 DASH duration, consider reencoding or using segment timeline
[Dasher] Changing HLS target duration from 1 to 10, either increase the segment duration or re-encode the content
[Dasher] First CTS 900000 in segment 2 drifting by 9 (more than half a segment duration) from segment time, consider ree
ncoding or using segment timeline
[Dasher] Segment 2 duration 10 more than 3/2 DASH duration, consider reencoding or using segment timeline
[Dasher] First CTS 1800000 in segment 3 drifting by 18 (more than half a segment duration) from segment time, consider r
eencoding or using segment timeline
```

bunny_dash_track2_118.m4s	3/25/2025 9:56 AM	M4S File	21 KB
bunny_dash_track2_119.m4s	3/25/2025 9:56 AM	M4S File	20 KB
bunny_dash_track2_120.m4s	3/25/2025 9:56 AM	M4S File	21 KB
bunny_dash_track2_init	3/25/2025 9:56 AM	MP4 File	1 KB
hls	3/25/2025 9:56 AM	M3U8 File	1 KB
hls_1	3/25/2025 9:56 AM	M3U8 File	2 KB
hls_2	3/25/2025 9:56 AM	M3U8 File	6 KB

Testing Playback

For more information on testing playback, please refer to documentation at <https://ezdrm.com/documentation> > **EZDRM Testing Playback**.

Additional Information

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