EZDRM
Bitmovin Player Integration
Table of Contents

Introduction ......................................................................................................................................................... 3
Requirements ...................................................................................................................................................... 3
Preparation in Bitmovin .................................................................................................................................... 3
Sample Code Files .............................................................................................................................................. 3
Head Section Code ............................................................................................................................................ 4
Widevine/PlayReady Universal DRM ........................................................................................................... 4
  Obtaining a PX Value ...................................................................................................................................... 4
  LA_ URL - Widevine....................................................................................................................................... 4
  LA_ URL - PlayReady .................................................................................................................................... 5
  Sample Widevine/PlayReady Script ........................................................................................................... 6
Apple FairPlay DRM for HLS .......................................................................................................................... 7
  Sample Fairplay Script ................................................................................................................................ 7
Sample Combined Script ................................................................................................................................ 8
Additional Information .................................................................................................................................. 11

Version 1.0 / Updated August 14, 2020
Introduction

This document outlines the steps necessary to successfully integrate EZDRM with the Bitmovin player.

Bitmovin accepts streams from an MPEG-DASH (*.mpd) file for Microsoft PlayReady and Google Widevine, and/or a *.m3u8 file for Apple FairPlay and HLS.

For information about the Bitmovin web video player, see:

- Bitmovin player homepage: https://Bitmovin.com/
- Bitmovin player online guide: https://bitmovin.com/docs/player

Requirements

1. Basic understanding of HLS, MPEG/DASH, and other streaming protocols and formats.

Preparation in Bitmovin

Log in to your Bitmovin account (bitmovin.com) and perform these steps:

1. Using left side navigation bar, click Player → Licenses → default-licenses
2. Copy your "Player Key" to a convenient place for use later.
3. Add your domain to the "Domains/ Package Name / Bundle Identifiers" list.

Sample Code Files

EZDRM provides complete sample html files that you can configure and use, based on those from Bitmovin.com:

- dash-sample.html [Widevine and PlayReady]
- fairplay-sample.html
- combined-sample.html [Widevine, PlayReady, and Fairplay]

You can download these files at:

https://www.ezdrm.com/downloads/Players/BitMovin/BitMovin-EZDRM-sample.zip

The following sections show how to customize the html files.
Head Section Code

```html
<! -- Base Bitmovin player code -->
<script type="text/javascript" src="https://cdn.bitmovin.com/player/web/8.1.0/bitmovinplayer.js"></script>
```

Widevine/PlayReady Universal DRM

MPEG-DASH requires 3 values:

1. The .mpd URL
2. The Widevine Proxy URL
3. The PlayReady Proxy URL

Obtaining a PX Value

You will need your six-digit PXs values as part of the LA_URLs in the upcoming sections.

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.


LA_URL - Widevine

For all packagers (except AWS, Anevia, and related ones), the base LA_URL is: https://widevine-dash.ezdrm.com/proxy?pX=XXXXXX

Replace XXXXXX with your Widevine PX Value.

Widevine for AWS, Anevia, and other CPIX or SPEKE-based encoders.

The Widevine LA_URL is different for these packagers. The base URL is: https://widevine-dash.ezdrm.com/widevine-php/widevine-foreignkey.php?pX=XXXXXX

Replace XXXXXX with your Widevine PX Value.
**LA_URL - PlayReady**

For PlayReady, the base proxy URL is:  

Replace **XXXXXX** with your PlayReady PX Value.
Sample Widevine/PlayReady Script

```html
<!-- Start the player -->
<script type="text/javascript">
    const container = document.getElementById('my-player');
    const playerConfig = {
        key: "<<YOUR PLAYER KEY>>" <!-- Replace with your URL-->
        drm: {
        },
        source = {
            dash: "<<HTTPS LOCATION OF YOUR MPD FILE>>", <!-- Replace with your URL-->
            drm: {
                widevine: {
                    <!-- Replace “XXXXXX” with your Widevine PX Value -->
                },
                playready: {
                    <!-- Replace “XXXXXX” with your PlayReady PX Value -->
                }
            }
        },
        drm: {
            widevine: {
            }
        },
        playready: {
        }
    },
    player.load(source).then(
        player => {
            console.log('Successfully created Bitmovin Player instance')
        },
        reason => {
            console.log('Error while creating Bitmovin Player instance')
        }
    );
</script>
```
Apple FairPlay DRM for HLS

Apple FairPlay requires 3 values.

1. The .m3u8 URL
2. The license_url
   - The base license_url is https://fps.ezdrm.com/api/licenses/<<ASSET-ID>>
3. The FPS cert URL
   - This is the .cer file provided by Apple. This URL should be either on your Website or CDN.

Sample Fairplay Script

```html
<script type="text/javascript">
    const container = document.getElementById('my-player');
    const playerConfig = {
        key: '<<YOUR PLAYER KEY>>' // Replace with your URL-->
    };
    const source = {
        hls: '<<HTTPS_LOCATION_OF_YOUR_M3U8_FILE>>', // Replace with your URL-->
        drm: {
            fairplay: {
                LA_URL: 'https://fps.ezdrm.com/api/licenses/<<ASSET-ID>>', // Insert Asset ID -->
                certificate_url: '<<HTTPS_LOCATION_OF_YOUR_CERTIFICATE>>', // Replace with your URL-->
            },
        },
        prepareContentId: function(contentId) {
            var uri = contentId;
            var uriParts = uri.split('://', 1);
            var protocol = uriParts[0].slice(-3);
            uriParts = uri.split(';', 2);
            contentId = uriParts.length>1?uriParts[1]:'';
            return protocol.toLowerCase()=='skd'?contentId:'';
        },
        prepareLicenseAsync: function(ckc) {
            return new Promise(function(resolve, reject) {
                var reader = new FileReader();
                reader.addEventListener('loadend', function() {
                    var array = new Uint8Array(reader.result);
                    resolve(array);
                });
                reader.addEventListener('error', function() {
                    reject(reader.error);
                });
            });
        },
    },
</script>
```
Sample Combined Script
For Widevine, PlayReady, and Apple Fairplay

```
<script type="text/javascript">
const container = document.getElementById('my-player');
const playerConfig = {
    key: '<<YOUR PLAYER KEY>>' <!-- Replace with your URL-->
};
const source = {
    dash: '<<HTTPS_LOCATION_OF_YOUR_MPD_FILE>>', <!-- Replace with your URL-->
    hls: '<<HTTPS_LOCATION_OF_YOUR_M3U8_FILE>>', <!-- Replace with your URL-->
    ...}
</script>
```
drm: {
  widevine: {
  },
  playready: {
  },
  fairplay: {
    LA_URL: 'https://fps.ezdrm.com/api/licenses/<<ASSET-ID>>',
    certificateURL: '<<HTTPS LOCATION OF YOUR CERTIFICATE>>'
    prepareContentId: function(contentId) {
      var uri = contentId;
      var uriParts = uri.split('://', 1);
      var protocol = uriParts[0].slice(-3);
      uriParts = uri.split(';', 2);
      contentId = uriParts.length > 1 ? uriParts[1] : '';
      return protocol.toLowerCase() == 'skd' ? contentId : '';
    },
    prepareLicenseAsync: function(ckc) {
      return new Promise(function(resolve, reject) {
        var reader = new FileReader();
        reader.addEventListener('loadend', function() {
          var array = new Uint8Array(reader.result);
          resolve(array);
        });
        reader.addEventListener('error', function() {
          reject(reader.error);
        });
        reader.readAsArrayBuffer(ckc);
      });
    },
    prepareMessage: function(event, session) {
      return new Blob([event.message], {type: 'application/octet-binary'});
    },
    headers: [{
      name: 'content-type',
      value: 'application/octet-stream'
    }],
    useUint16InitData: true,
    licenseResponseType: 'blob'
  }
}
```javascript
};
const player = new bitmovin.player.Player(container, playerConfig);
player.load(source).then(
    player => {
        console.log('Successfully created Bitmovin Player instance')
    },
    reason => {
        console.log('Error while creating Bitmovin Player instance')
    }
);
</script>
Additional Information

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