EZDRM
Digital Rights Management. Simplified.

EZDRM Service Integration
Into Kaltura CE with Nginx

Table of Contents

Version 2.1 / Updated May 14, 2020

Introduction

The following document outlines the steps necessary in order to successfully integrate EZDRM on top of an existing Kaltura “Community Edition” installation.

This guide includes information on how to configure Kaltura’s open-source nginx-vod-module, in particular with the vod_drm_request_uri directive, for on-the-fly repackaging, and encryption.

Requirements

The main three components in this solution are:

1. Backend server with mp4 media files (In our case, the Kaltura system)
2. Nginx server with kaltura-vod-module
3. EZDRM backend server for encryption-keys and license retrieval

The Kaltura nginx-vod-module includes support for on-the-fly DRM encryption. When the client (end-user) requests an HLS playlist or an MPEG/DASH manifest, the nginx kaltura vod module can repackage remotely stored files on-the-fly into HLS or DASH segments. The nginx vod module can also encrypt the chunks and sign them using DRM (in our case EZDRM).

For example: if we upload a video into Kaltura, wait for it to transcode, and open the preview & embed page, we will see a Kaltura player that will play that video. If you’ve configured Kaltura with NGINX then you’ll also be able to play this video using HLS or MPEG/DASH.

This guide will help you configure the NGINX kaltura vod plugin to encrypt the MP4 content on-the-fly using keys provided by EZDRM.

Architecture

The main three components in this solution are:

1. Backend server with mp4 media files (In our example Big Buck Bunny server)
2. Nginx server with kaltura-vod-module

3. EZDRM backend server for encryption-keys and license retrieval

The Kaltura nginx-vod-module includes support for on-the-fly DRM encryption. When the client (end-user) requests an HLS playlist or an MPEG/DASH manifest, the nginx kaltura vod module can repackage remotely stored files on-the-fly into HLS or DASH segments. The nginx vod module can also encrypt the chunks and sign them using DRM (in our case EZDRM).

For example: if we upload an encoded MP4 file to Amazon S3 or Google Cloud Storage, we will then be able to repackage this file on-the-fly to HLS or MPEG/DASH using the Kaltura nginx module. While repackaging, we will be able to use EZDRM services to encrypt this file with DRM.

This guide will help you configure the NGINX kaltura vod plugin to encrypt the MP4 content on-the-fly using keys provided by EZDRM.

Widevine/PlayReady Universal DRM

Setup Instructions

Please follow the steps below, on the machine where the kaltura-nginx package is installed.

Note: *If you've chosen the all-in-one server installation then the nginx should be installed on the same machine as the rest of the Kaltura system. If you've chosen to install a Kaltura cluster, then please go to the machine where the kaltura-nginx package is installed.*

1. Make sure you have root access (sudo)
2. Open the nginx configuration file for the Kaltura server, located at /etc/nginx/conf.d/kaltura.conf with your favorite editor
3. Copy and paste the following lines towards the end of the file, replacing the u=username and p=password highlighted below with your **EZDRM username**, and **password.**
4. The following values are **required** for optimization of the DRM licensing process:
    a. **vod_drm_info_cache drm_cache 128m;**
    b. **vod_drm_single_key on;**

```
        vod_drm_info_cache drm_cache 128m;
        location ~ ^/drmdash/p/\d+/(sp/\d+/)?serveFlavor/.*entryId/((?U).*)/ {
                vod dash;
                vod_segment_duration 4000;
                vod_bootstrap_segment_durations 3500;
                vod_align_segments_to_key_frames on;
                vod_dash_manifest_format segmenttemplate;

                vod_drm_upstream_location /drmdashinfo;
                vod_drm_enabled on;
                vod_drm_single_key on;
                vod_drm_request_uri "u=username&p=password&c=$2";

                add_header Last-Modified "Sun, 19 Nov 2000 08:52:00 GMT";
                add_header Access-Control-Allow-Headers "origin,range,accept-encoding,refe
rer";
```

```
                        add_header Access-Control-Expose-Headers "Server,range,Content-Length,Cont
ent-Range";
                        add_header Access-Control-Allow-Methods "GET, HEAD, OPTIONS";
                        add_header Access-Control-Allow-Origin "*";
                        expires 100d;
                }

                location /drmdashinfo {
                        internal;
                        proxy_pass "http://cpix.ezdrm.com/kaltura.aspx?";
                }
```

5. Run the following command:

```
sudo service kaltura-nginx restart
```

6. The final step is to modify the Kaltura Delivery profile URL by going to the Admin Console - Delivery Profiles. The link to this page is: http://<your_kaltura_server_url>/admin_console/index.php/delivery/delivery-profiles-configuration/filter_type/byid/filter_input/0
   On this page, find the delivery profile named “Kaltura Packager Dash segmentation”. Edit the profile and edit the last part of the URL.
   Change it from /dash to **/drmdash**.
7. To test your configuration, you will need to load a mpeg/dash kaltura playManifest URL in a DRM compatible player such as Bitmovin or Shaka player.

   An example playManifest URL in mpeg/dash format is:
   https://<your_kaltura_url>/p/<partnerId>/sp/<partnerId>00/playManifest/entryId/<entryId>/format/mpegdash/protocol/https/a.mpd
   Note: *make sure to replace the highlighted parts with the relevant values.*

Microsoft Smooth Streaming

Setup Instructions

Please follow the steps below, on the machine where the kaltura-nginx package is installed.

Note: *If you've chosen the all-in-one server installation then the nginx should be installed on the same machine as the rest of the Kaltura system. If you've chosen to install a Kaltura cluster, then please go to the machine where the kaltura-nginx package is installed.*

1. Make sure you have root access (sudo)
2. Open your nginx configuration file and add the following configuration, replacing the **u=username** and **p=password** highlighted below with your EZDRM username and password, and the **hostname** with your files storage backend hostname
3. The following values are **required** for optimization of the DRM licensing process:
    a. **vod_drm_info_cache drm_cache 128m;**
    b. **vod_drm_single_key on;**

```
                location ~ ^/drmmss/ {
                    vod mss;
                    vod_mode remote;
                    vod_upstream_location /vod_upstream;
                    vod_segment_duration 4000;
                    vod_bootstrap_segment_durations 3500;
                    vod_drm_upstream_location /drmmssinfo;
                    vod_drm_enabled on;
                    vod_drm_single_key on;

                    vod_drm_request_uri "u=username&p=password&c=";

                    add_header Last-Modified "Sun, 19 Nov 2000 08:52:00 GMT";
                    add_header Access-Control-Allow-Headers "origin,range,accept-encoding,refe
rer";
```

```
                        add_header Access-Control-Expose-Headers "Server,range,Content-Length,Cont
ent-Range";
                        add_header Access-Control-Allow-Methods "GET, HEAD, OPTIONS";
                        add_header Access-Control-Allow-Origin "*";

                        expires 100d;
                }

                location /vod_upstream {
                        internal;
                        rewrite /drmmss/(.*) /$1 break;
                        proxy_pass http://hostname;
    }

                location /drmmssinfo {
                        internal;
                        proxy_pass "http://cpix.ezdrm.com/kalturamss.aspx?";
                }
```

4. Run the following command to restart:

```
sudo service kaltura-nginx restart
```

1. To test your configuration, you will need to load an MSS manifest in a DRM compatible player such as Bitmovin, on a Windows machine in Internet Explorer or EDGE.
2. An example manifest URL in MPEG DASH format is:
   https://<your_nginx_server_hostname_url>/drmmss/path/to/video/file.mp4/manifest
   Note: *make sure to replace the highlighted parts with the relevant values.*

Apple FairPlay DRM for HLS

Obtaining an Asset ID

Option 1: Request DRM keys using EZDRM URL for Kaltura

To obtain an AssetID, use the following URL replacing **u=** with your EZDRM **username** and **p=** with your EZDRM **password**.

http://cpix.ezdrm.com/kalturagetasset.aspx?u=username&p=password

Option 2: Request DRM keys with curl

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

```
curl -X POST 'http://cpix.ezdrm.com/kalturagetasset.aspx?U=Username&P=Password' -d ' '
```

Note: there is a space between the single quotes at the end of the line.

The parameters are as follows:

Parameter	Description
U	EZDRM username
P	EZDRM password

The following is returned:

```
[{"asset": "9clXXXX4-6d6e-4773-bXX2-c86XXXXXX5cl "}]
```

Setup Instructions

Please follow the steps below, on the machine where the kaltura-nginx package is installed.

Note: *If you've chosen the all-in-one server installation then the nginx should be installed on the same machine as the rest of the Kaltura system. If you've chosen to install a Kaltura cluster, then please go to the machine where the kaltura-nginx package is installed.*

1. Make sure you have obtained an Asset ID
2. Make sure you have root access (sudo)
3. Open the nginx configuration file for the Kaltura server, located at /etc/nginx/conf.d/kaltura.conf with your favorite editor
4. Copy and paste the following lines towards the end of the file, replacing the u=username and p=password highlighted below with your **EZDRM username** and **password**, and replace the **$hls_drm_asset_id** value with the correct guid for your Asset ID
5. The following values are **required** for optimization of the DRM licensing process:
    a. **vod_drm_info_cache drm_cache 128m;**
    b. **vod_drm_single_key on;**

```
vod_drm_info_cache drm_cache 128m;

location ~ ^/drmhls/p/\d+/(sp/\d+/)?serveFlavor/.*entryId/((?U).*)/ {
                        vod hls;
                        vod_bootstrap_segment_durations 2000;
                        vod_bootstrap_segment_durations 2000;
                        vod_bootstrap_segment_durations 2000;
                        vod_bootstrap_segment_durations 4000;

                        set $hls_drm_asset_id "9clXXXX4-6d6e-4773-bXX2-c86XXXXXX5cl";

                        vod_hls_encryption_method sample-aes;
                        vod_hls_encryption_key_uri "skd://fps.ezdrm.com/;$hls_drm_asset_id";
                        vod_hls_encryption_key_format "com.apple.streamingkeydelivery";
```

```
                        vod_hls_encryption_key_format_versions "1";

                        vod_drm_enabled on;
                        vod_drm_single_key on;
                        vod_drm_upstream_location /hlsdrminfo;
                        vod_drm_request_uri "a=$hls_drm_asset_id&u=username&p=password";
                        add_header Last-Modified "Sun, 19 Nov 2000 08:52:00 GMT";
                        add_header Access-Control-Allow-Headers "*";
                        add_header Access-Control-Expose-Headers "Server,range,Content-Length,C
ontent-Range";
                        add_header Access-Control-Allow-Methods "GET, HEAD, OPTIONS";
                        add_header Access-Control-Allow-Origin "*";
                        expires 100d;

                }


                location /hlsdrminfo {
                        internal;
                        proxy_pass https://cpix.ezdrm.com/kalturafps.aspx?;
                }
```

```

```

```

```

6. Run the following command

```
sudo service kaltura-nginx restart
```

7. The final step is to modify the Kaltura Delivery profile URL by going to the Admin Console - Delivery Profiles. The link to this page is: http://<your_kaltura_server_url>/admin_console/index.php/delivery/delivery-profiles-configuration/filter_type/byid/filter_input/0
   On this page, find the delivery profile named “Kaltura Packager Hls segmentation”. Edit the profile and edit the last part of the url.
   Change it from /hls to **/drmhls**
8. To test your configuration, you will need to load an mpeg/dash kaltura playManifest URL in a Fairplay player that can be provided by EZDRM. An

example playManifest URL in mpeg/dash format is:
https://<your_kaltura_url>/p/<partnerId>/sp/<partnerId>00/playManifest/entryId/<entryId>/format/applehttp/protocol/https/a.m3u8
Note: *make sure to replace the highlighted parts with the relevant values.*

References

NGINX VOD Module Github Page: https://github.com/kaltura/nginx-vod-module

Additional Information

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