Manage CMAF playback from Wowza Streaming Engine

Originally Published on 06/04/2019 | Updated on 11/05/2019 6:48 pm PST

When you create Apple HLS and MPEG-DASH live streams with Wowza Streaming Engine™ media server software version 4.7.8 or later, you can use the CMAF packetizer independently or in tandem with the server software’s standalone HLS and MPEG-DASH packetizers. Depending on the packetizer or packetizers you use, Wowza Streaming Engine produces different types of segments or formats, and correspondingly different playback URLs. An understanding of how the packetizers construct playback URLs can help you manage and control stream playback.

Player support for Wowza Streaming Engine CMAF live streams

When producing CMAF live streams, you should be aware of which players support them. Player support, which varies for HLS and DASH playback, is evolving.

Wowza Streaming Engine CMAF streams have been tested on and can be played on the desktop and on mobile devices as follows:

<table>
<thead>
<tr>
<th>Stream type</th>
<th>Player support</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMAF HLS</td>
<td>hls.js, including <a href="http://players.akamai.com/hls/">http://players.akamai.com/hls/</a>; and JW Player 8</td>
</tr>
<tr>
<td>CMAF DASH</td>
<td>Any third-party player that supports MPEG-DASH, including the latest version of Google Shaka Player</td>
</tr>
</tbody>
</table>

About playback URLs for HLS live streams

Wowza Streaming Engine can produce HLS live streams by using either of two packetizers: the "Cupertino" packetizer, [cupertinostreamingpacketizer](http://cupertinostreamingpacketizer), or the CMAF packetizer, [cmafstreamingpacketizer](http://cmafstreamingpacketizer).

The Cupertino packetizer, the original HLS packetizer in Wowza Streaming Engine, generates
segments that use the MPEG-TS container format.

The newer CMAF packetizer generates segments that use the fragmented MP4 (fMP4) container format.

You can use either or both packetizers in a live streaming application.

If you use one of them—either one—Wowza Streaming Engine generates a master playlist URL for the stream that uses the format:

http://[wowza-ip-address]/[application]/[application-instance]/[stream-name]/playlist.m3u8

Where:

- **[wowza-ip-address]** is the IP address or domain and port for Wowza Streaming Engine (default port 1935)
- **[application]** is the application name
- **[application-instance]** is the name of the application instance (if omitted, defaults to _definst_)
- **[stream-name]** is the stream name

So, for example, the playlist URL for an HLS live stream **myStream** from Wowza Streaming Engine at the address **mycompany.com** using the **live** application is:

http://mycompany.com:1935/live/myStream/playlist.m3u8

If you use both HLS packetizers, Wowza Streaming Engine generates two URLs, one for each packetizer.

When both HLS packetizers are enabled, the Cupertino stream playback URL uses the format:

http://[wowza-ip-address]/[application]/[application-instance]/[stream-name]/playlist.m3u8

or

http://[wowza-ip-address]/[application]/[application-instance]/[stream-name]/playlist_sfts.m3u8

where _sfts_ indicates that the media playlist contains .ts segments.

When both HLS packetizers are enabled, the CMAF HLS stream playback URL uses the format:
where _sfm4s_ indicates that the media playlist contains fMP4 segments.

**About playback URLs for MPEG-DASH live streams**

Wowza Streaming Engine can produce MPEG-DASH live streams by using either of two packetizers: the original MPEG-DASH packetizer, `mpegdashstreamingpacketizer`, or the newer CMAF packetizer, `cmafstreamingpacketizer`.

Enabling `mpegdashstreamingpacketizer` and `cmafstreamingpacketizer` is valid but redundant, as both create streams whose segments use the fMP4 container format. If you configure a live stream that uses both packetizers, Wowza Streaming Engine ignores `mpegdashstreamingpacketizer` and produces one set of fMP4 segments using `cmafstreamingpacketizer`.

Both DASH packetizers generate a list of available media segment URLs in a Media Presentation Description (.mpd) manifest file. An MPEG-DASH manifest (playback) URL for a Wowza Streaming Engine stream, no matter how it's packetized, uses the format:

http://[wowza-ip-address]/[application]/[application-instance]/[stream-name]/manifest.mpd

Where:

- `[wowza-ip-address]` is the IP address or domain and port for Wowza Streaming Engine (default port 1935)
- `[application]` is the application name
- `[application-instance]` is the name of the application instance (if omitted, defaults to `_definst_`)
- `[stream-name]` is the stream name

For example, the playback URL for an MPEG-DASH live stream `myStream` from Wowza Streaming Engine at the address `mycompany.com` using the `live` application is:


**Differences between DASH and CMAF DASH manifests**

Under the hood, the MPD manifests for `mpegdashstreamingpacketizer` and `cmafstreamingpacketizer` are similar but not identical.

The CMAF packetizer also uses different segment format indicators and file extensions:

- .cmfv for video
- .cmfa for audio

For example:

- segment_ctaudio_cfcmfa_rid$RepresentationID$_cinit_w1839415535_mpd.cmfa
- segment_ctvideo_cfcmfv_rid$RepresentationID$_cinit_w1839415535_mpd.cmfv

When mpegdashstreamingpacketizer is used, all segments in the manifest have the suffix .m4s and you must rely on the _ctvideo_ or _ctaudio_ component near the beginning of the string to distinguish video from audio. For example:

- segment_ctvideo_cfm4s_rid$RepresentationID$_cinit_w99394109_mpd.m4s
- segment_ctaudio_cfm4s_rid$RepresentationID$_cinit_w99394109_mpd.m4s

**Note:** Wowza Streaming Engine CMAF DASH segments pass the Dash Industry Forum validation tool.

### About playback URLs for adaptive bitrate CMAF streams

Wowza Streaming Engine generates HLS playback URLs and DASH manifests for adaptive bitrate streams using the same methodology as it does for single bitrate HLS and DASH streams. The difference is that playback URLs and manifests for adaptive bitrate CMAF streams include the smil: prefix before the stream name.

If only cmafstreamingpacketizer is enabled, the HLS adaptive bitrate playback URL uses the format:

http://[wowza-ip-address]/[application]/[application-instance]/smil: [stream-name]/playlist.m3u8

When both HLS packetizers are enabled, the Cupertino stream playback URL uses the format:

http://[wowza-ip-address]/[application]/[application-instance]/smil: [stream-name]/playlist.m3u8
When both HLS packetizers are enabled, the CMAF HLS stream playback URL uses the format:

```
http://[wowza-ip-address]/[application]/[application-instance]/smil:
[stream-name]/playlist_sfm4s.m3u8
```

The manifest (playback URL) for an adaptive bitrate MPEG-DASH live stream, no matter how it's packetized, uses the format:

```
http://[wowza-ip-address]/[application]/[application-instance]/smil:
[stream-name]/manifest.mpd
```

**Manually select the HLS segment format to play**

If both the Cupertino and CMAF packetizers are enabled for your Wowza Streaming Engine live stream, you can manually invoke playback of either stream by appending the segment format key to the playlist URL.

**Play the Cupertino .ts stream:**

```
http://[wowza-ip-address]/[application]/[application-instance]/[stream-name]/playlist_sfts.m3u8
```

**Play the CMAF fMP4 stream:**

```
http://[wowza-ip-address]/[application]/[application-instance]/[stream-name]/playlist_sfm4s.m3u8
```

Where:

- **sf** indicates the segment format key
- **ts** indicates the MPEG-TS segments in the Cupertino HLS stream
- **m4s** indicates the fMP4 segments in the CMAF HLS stream

You can also select the segment format to play using a query parameter.

**Play the Cupertino .ts stream using a query parameter:**

```
http://[wowza-ip-address]/[application]/[application-instance]/[stream-name]/playlist.m3u8?wowzasegmentformat=ts
```

**Play the CMAF fMP4 stream using a query parameter:**
For more on CMAF query parameters, see View CMAF HLS stream playlists from Wowza Streaming Engine.

Play CMAF streams over SSL

To enable CMAF stream playback over SSL, first install an SSL certificate from a certificate authority. For instructions, see Request an SSL certificate for Wowza Streaming Engine from a certificate authority.

Then, to play the stream, use the HTTPS protocol and replace the [wowza-ip-address] portion of the stream’s playback URL with the SSL certificate domain and port 443. In other words, use the format:

https://[ssl-certificate-domain-name]:443/[application]/[application-instance]/[stream-name]/[playlist or manifest filename]

For example, the basic playback URL for the CMAF HLS live stream myStream from the SSL domain mySSLcertificate.com using the live application is:

https://mySSLcertificate.com:443/live/myStream/playlist.m3u8

or

https://mySSLcertificate.com:443/live/myStream/playlist_sfm4s.m3u8

And the basic playback URL for the CMAF MPEG-DASH live stream myStream from the SSL domain mySSLcertificate.com using the live application is:

https://mySSLcertificate.com:443/live/myStream/manifest.mpd