EME in WebKit with GStreamer
Quick status and challenges

Xabier Rodríguez Calvar

calvaris@igalia.com

GStreamer conference

Edinburgh

25th-26th October 2018

Who am I?

- Fiddling with WebKit and GStreamer

- WebKit committer and reviewer

- GStreamer contributor

- Partner at Igalia

  - Worker-owned coop, currently around 70 happy Igalians around the world

  - Provides consulting services for various Free Software projects

igalia

Encrypted Media Extensions

- W3C standard that brings DRM to the web

  - ClearKey

- Implemented and in WPEWebKit downstream

  - PlayReady

  - Widevine

- Upstreaming is not trivial legally

- Works with Media Source Extensions and regular playback

igalia

Demuxers

- GStreamer demuxers need to be aware of encryption

  - qtdemux quite stable for MP4

  - matroskademux just landed for WebM

- Send protection events with initialization data

- Configure encrypted caps

  - A key system is selected according to the available decryptors

  - playbin does the rest

igalia

Decryptor selection (1)
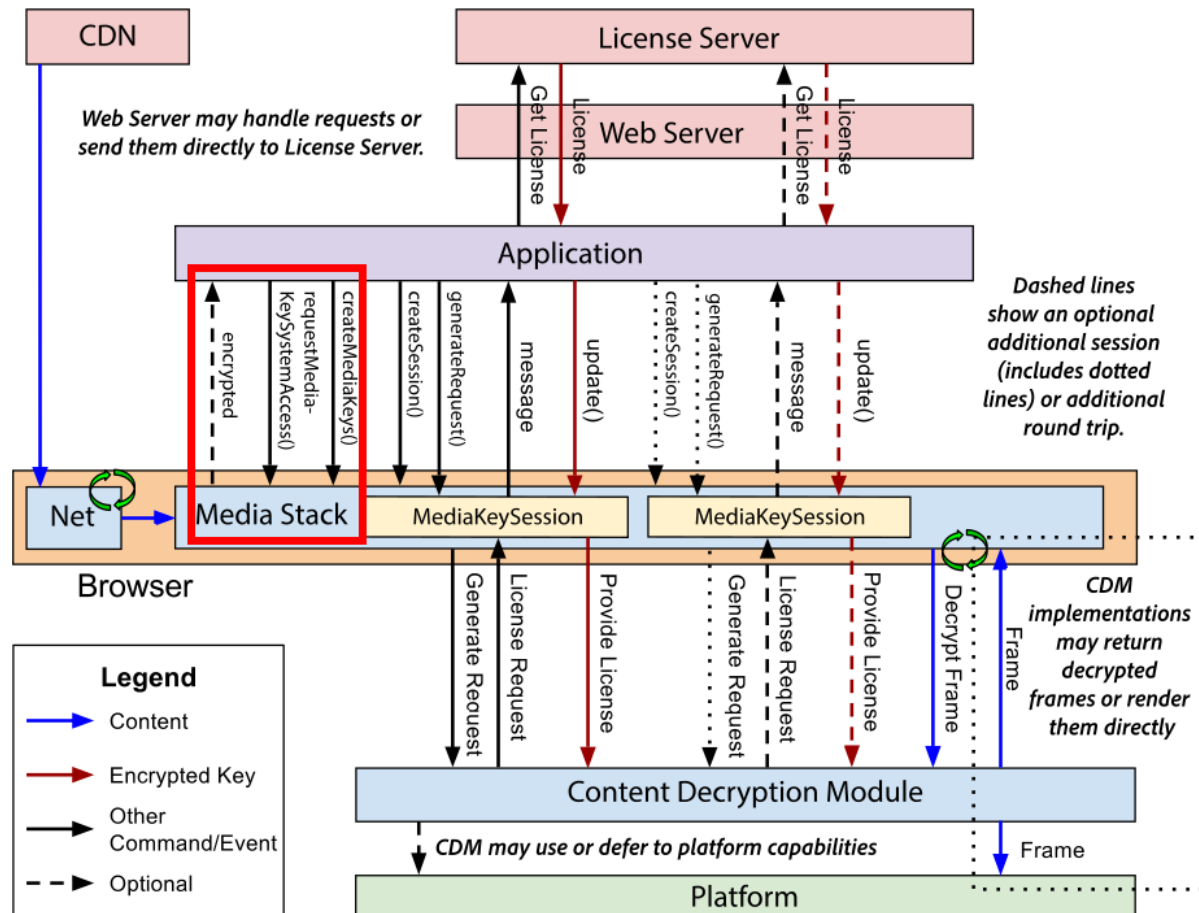
Decryptor selection (and 2)

- Decryptors need to let the app select the encryption system
  - qtdemux runs a query for preferred key system
  - matroskademux does nothing?
  - Should be answered synchronously
- Single decryptor for all systems
  - Works
  - Wrong caps
- Room for improvement
  - Depends on the demuxer
  - No common API for this

igalia

Protection GstEvent

```
GstEvent *
gst_event_new_protection (const gchar *system_id,
                          GstBuffer *data,
                          const gchar *origin);
```

- system_id is non-nullable
- WebM does not report key systems
- API cannot be changed
  - define for undefined systems

Future

- Upstream, upstream, upstream

- Improve decryptors arch to handle decoder/decryptor/sink elements

  - Base class?