**Streaming**

IP-516\_AppleTV App

**Version number:** 0.1  
**Main author:** AZ, JB  
**Last revision:** 21.10.2016  
**Status:** in work  
**Filename:** Streaming  
**Client:** Michael Keller, UPC

Authors

|  |  |
| --- | --- |
| **Name** | **Initials** |
| Andrea Zirn | AZ |
|  |  |
|  |  |

Revision list

| **Version** | **Date** | **Comment** | **Author** |
| --- | --- | --- | --- |
| 0.1 | 21.10.2016 | Initial version | AZ |
| 0.2 | 28.10.2016 | Add content | AZ |
|  |  |  |  |

Table of Content

Authors 2

Revision list 2

Table of Content 3

1 Heading 1 3

1.1 Heading 2 3

1.1.1 Heading 3 3

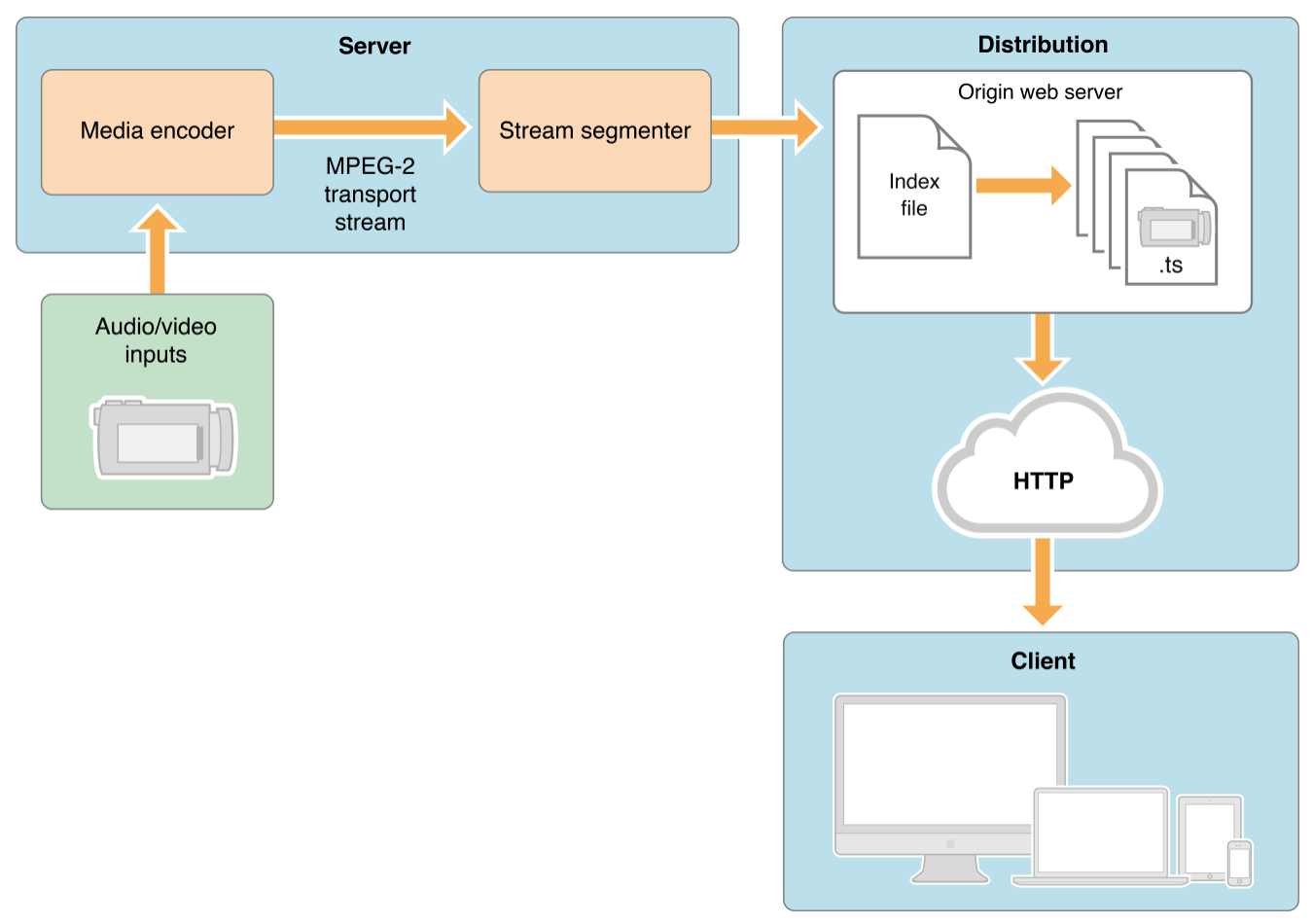
# HTTP Live Streaming[[1]](#footnote-1)

HTTP Live Streaming lets you send audio and video over HTTP from an ordinary web server for playback on iOS-based devices. HTTP Live Streaming supports both live broadcasts and rerecorded content (video on demand).

## General information

* Streaming audio or video to iPhone, iPod, iPad or Apple TV
* Streaming live events without special server software
* Sending video on demand with encryption and authentication
* Apple TV 2 and later includes an HTTP Live Streaming client

### Functionality



You can serve HTTP Live Streaming audio and video from an ordinary web server. HTTP Live Streaming is a way o send audio and video over HTTP from a web server to client software on the desktop or to iOS-based devices.

HTTP Live Streaming sends audio and video as a series of small files, typically of about 10 seconds duration, called media segment files.

### Live Stream

For live streams, Apple provides a free tool to make media segment files and playlists from live MPEG-2 transport streams carrying H.264 video, AAC audio, or MP3 audio. There are a number of hardware and software encoders that can create MPEG-2 transport streams carrying MPEG-4 video and AAC audio in real time.

## HTTP Streaming Architecture[[2]](#footnote-2)

### Overview

HTTP Live Streaming consist of three parts:

1. The server component:  
   is responsible for taking input streams of media and encoding them digitally, encapsulating them in a format suitable for delivery and preparing the encapsulated media for distribution
2. The distribution component:  
   consists of standard web servers. They are responsible for accepting client requests and delivering prepared media and associated resources to the client.
3. The client software:  
   is responsible for determining the appropriate media to request, downloading those resources, and then reassembling them so that the media can be presented to the user in a continuous stream.

Input can be live or from a prerecorded source. It is typically encoded as MPEG-4 (H.264 video and AAC audio) and packaged in an MPEG-2 Transport Stream by off-the-shelf hardware. The MPEG-2 transport stream is then broken into segments and saved as a series of one or more .ts media files. This is typically accomplished using a software tool such as the Apple stream segmenter.

### Server Components

The server requires a media encoder, which can be off-the-shelf hardware, and a way to break the encoded media into segments and save them as files, which can either be software such as the media stream segmenter provided by Apple or part of an integrated third-party solution.

**Media Encoder**

The media encoder takes a real-time signal from an audio-video device, encodes the media, and encapsulates it for transport. Currently, the supported delivery format is MPEG-2 Transport Streams for audio-video.

The encoder delivers the encoded media in an MPEG-2 Transport Stream over the local network to the stream segmenter. The transport stream is a packaging format that can be used with a number of different compression formats. For video: The UIKit UIIMagePickerController, AVKit, AV Foundation, Core Media. (for more information see: <https://developer.apple.com/library/content/documentation/Miscellaneous/Conceptual/iPhoneOSTechOverview/MediaLayer/MediaLayer.html#//apple_ref/doc/uid/TP40007898-CH9-SW6>)

**Stream Segmenter**

The stream segmenter is a process that reads the Transport Stream from the local network and divides it into a series of small media files of equal duration. The segmenter also creates an index file containing references to the individual media files. Each time the segmenter completes a new media file, the index file is updated. The index is used to track the availability and location of the media files. Media segments are saved as .ts-files and index files are saved as .M3U8-playlists.

**File Segmenter**

If you already have a media file encoded using supported codecs, you can use a file segmenter to encapsulate it in an MPEG-2 transport stream and break it into segments of equal length. The file segmenter allows you to use a library of existing audio and video files for sending video on demand via HTTP Live Streaming. The file segmenter performs the same tasks as the stream segmenter, but it takes files as input instead of streams.

### Distribution Components

The distribution system is a web server or a web caching system that delivers the media files and index files to the client over HTTP.

### Client Components

The client software begins by fetching the index file, based on a URL identifying the stream. The index file in turn specifies the location of the available media files, decryption keys, and any alternate streams available. For the selected stream, the client downloads each available media file in sequence. Each file contains a consecutive segment of the stream. Once it has a sufficient amount of data downloaded, the client begins presenting the reassembled stream to the user.

The client is responsible for fetching any decryption keys, authenticating or presenting a user interface to allow authentication, and decrypting media files as needed.

## Using HTTP Live Stream

The HTTP Live Streaming Tools package installs prerelease command-line tools that are used for deployment and validation of HTTP Live Streaming solutions. The tools are:

* Media Stream Segmenter
* Media File Segmenter
* Media Subtitle Segmenter
* Variant Playlist Creator
* Media Stream Validator
* HLS Report
* ID3 Tag Generator (or a variant playlist generator)

This tools can be downloaded from the Apple Developer website.[[3]](#footnote-3)

# JW Player

# ffmpeg

1. <https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/Introduction/Introduction.html> (21.10.2016) [↑](#footnote-ref-1)
2. <https://developer.apple.com/library/content/documentation/NetworkingInternet/Conceptual/StreamingMediaGuide/HTTPStreamingArchitecture/HTTPStreamingArchitecture.html#//apple_ref/doc/uid/TP40008332-CH101-SW2> (21.10.2016) [↑](#footnote-ref-2)
3. <https://developer.apple.com/programs/how-it-works/> (28.10.2016) [↑](#footnote-ref-3)