HTTP Adaptive Streaming

Using the Edgeware Video Delivery Appliances

- Microsoft® Smooth Streaming
- Apple HTTP Live Streaming
- Adobe® HTTP Dynamic Streaming
## Table of Contents

1. Confidentiality notice ......................................................................................................................................... 3  
2. About this document ......................................................................................................................................... 3  
3. History ................................................................................................................................................................ 3  
4. Introduction ....................................................................................................................................................... 4  
5. The Issues that HTTP Adaptive Streaming Address ..................................................................................... 4  
6. HTTP Adaptive Streaming Ecosystem and Architecture .................................................................................. 6  
   6.1 Edgeware Video Streaming Servers ................................................................................................................. 6  
   6.2 Support within the Edgeware Asset Caching & Propagation System ................................................................. 7  
   6.3 Edgeware Origin Management System ........................................................................................................... 13  
7. References ....................................................................................................................................................... 13
1. Confidentiality notice

This document is confidential and may not be reproduced, distributed or used for any purpose other than by the recipient for the assessment, evaluation and use of Edgeware products, unless written permission is given in advance by Edgeware AB.

2. About this document

This document provides an overview of adaptive streaming technologies and their application in Edgeware’s video delivery products. The intended readers are operator and CDN product managers and technical architects looking for an overview of Edgeware products and their operation. It is not intended to provide in-depth technical information, which is available in Edgeware technical documentation.

3. History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2010-03-17</td>
<td>First version.</td>
</tr>
<tr>
<td>1.1</td>
<td>2010-03-22</td>
<td>Edited for grammar and spelling. No major changes to content.</td>
</tr>
<tr>
<td>1.2</td>
<td>2010-04-06</td>
<td>Formatting changes only</td>
</tr>
<tr>
<td>1.3</td>
<td>2010-04-07</td>
<td>Corrections to Adobe® HTTP Dynamic Streaming plus addition of new live streaming info.</td>
</tr>
<tr>
<td>1.4</td>
<td>2010-04-09</td>
<td>Simplified Verimatrix solution description. Incorporated modifications from Adobe</td>
</tr>
<tr>
<td>1.5</td>
<td>2010-04-27</td>
<td>Added front cover and finalized for publishing</td>
</tr>
<tr>
<td>1.6</td>
<td>2010-05-05</td>
<td>Corrected Apple Live HTTP Streaming Content Flow</td>
</tr>
<tr>
<td>1.7</td>
<td>2010-10-12</td>
<td>Removed incorrect reference on Live Ingest in Apple HTTP Live Streaming in Section 6</td>
</tr>
<tr>
<td>1.8</td>
<td>2010-10-15</td>
<td>Formatting changes only</td>
</tr>
</tbody>
</table>
4. Introduction

Traditionally, telco and cable operator’s video services have been distributed over managed access lines where the bandwidth required for a good quality of experience has been provisioned and is suitably robust. However, there are now a huge range of Internet connected devices available which are capable of high quality video playback. These include laptops and home media centers, smartphones such as the Apple iPhone, Blueray devices and gaming consoles. These devices are typically connected to un-managed access networks such as 3G, home networks and WiFi hot spots.

In addition, video content owners are increasingly choosing to make their content available directly on the Internet via massively popular services such as the BBC iPlayer and Hulu™. Delivery of these services is typically handled by Content Delivery Networks (CDNs) such as Akamai and Limelight Networks® that deliver “Over The Top” of operator networks, leading to the description “OTT video services”. Although CDNs optimize delivery over the transit network, these services are all affected by varying degrees of congestion when they reach the local operator’s network – the so called “middle and last mile”.

However, demand from consumers to watch video anytime, anywhere has lead to an urgent requirement for operators and CDNs to be able to deliver video services to these devices with a high Quality of Experience (QoE). A number of leading companies have developed HTTP Adaptive Streaming technologies to specifically enable this including Microsoft®, Apple and most recently Adobe®.

This application note provides an overview of the issues that these technologies address and describes how they are incorporated with Edgeware WTV servers to create a massively scaleable video delivery solution for these devices.

5. The Issues that HTTP Adaptive Streaming Address

**Network Connectivity and Traversal Assurance Requirement**

When attempting network connectivity in an un-managed network, routers, firewalls and which ports are open are unknown. In a home network, there are personal firewalls, possible routers and security software scanning port activity. In a WiFi hot spot, the port access can be extremely limited due to security concerns.

This is a well known hurdle with network applications and is overcome by using the HTTP protocol for communication. HTTP uses port 80 for requests. Requests to this port are most likely port to be allowed through any firewall or router as they are used for all web surfing. As HTTP uses a state-full TCP connection, any issues that can be incurred by NAT based networks are also overcome.

**Bandwidth Management Requirement**

Bandwidth consistency is a major issue. If a user is watching a video and someone else on the same network suddenly decides to perform a file transfer, the available bandwidth for the video can be severely impacted. In order to maintain a good Quality of Experience, content therefore needs to be encoded at different bit rates.
and the delivery protocol needs to be able to dynamically switch the bit rate with no interruption in playback or action by the user.

The HTTP protocol is a synchronous client-to-server protocol so only one request can be made for a video file. Switching bit rates on the fly is therefore not possible in the middle of an HTTP transaction. To overcome this problem, the video must be sliced up into “chunks.” Each chunk is typically between two to ten seconds of video. The chunk sizes are such that the reference IFrame at the beginning of each chunk is synchronized.

The delivery server can host several different bit rate encodings of the same video content. Each bit rate encoding has a separate play list, which is defined by a master playlist. These playlists are typically in an M3U format and contain the list of chunks in order. When the client detects either insufficient bandwidth or more available bandwidth, it can switch to either the lower or higher bit rate playlist and download the chunks in that list. Since each chunk is synchronized with the other bit rate streams, there is a seamless transition between them so that the video playback is not interrupted. This maintains a high quality user experience.

**Multiple Clients and Resolutions**

As more and more Internet connected devices appear on the market every month, with greater and greater capabilities for video browsing and playback, the number of resolutions and bit rates required to support these devices increases exponentially. Historically, many devices have also communicated via a proprietary protocol. It is not viable to have separate encoders, DRM systems and delivery servers for each device that needs to be supported.

HTTP Adaptive streaming enables delivery of multiple resolutions and bit rates over a common, open protocol. This enables consolidation of the encoder, encryption and delivery server infrastructure to a single manageable system. New devices can be added simply via a new encoding profile.

**Content Security**

The traditional broadcast method of restricting access to live content is achieved by encrypting the transmission stream. This method has been successfully applied to delivery of live video over the Internet: content is encrypted between the encoder and the server and then again between the server and the client. However, the risk with this approach is that content is being temporarily written to cache in the server before immediate retransmission and again is written to the client cache before immediate presentation and deletion. During these short periods, content is sitting ‘in the clear’ and this presents a security risk. Also, for VOD and catchup TV services, content will be stored on servers in the network or client devices and must remain protected to avoid piracy.

HTTP Adaptive streaming enables encryption of the individual chunks of content for both live streaming and VOD / catchup TV downloads. The content itself is encrypted during or immediately after encoding and remains so during transmission across networks and when stored on servers or client devices.
6. HTTP Adaptive Streaming Ecosystem and Architecture

Edgeware provides highly optimized video server technology for integration into a solution for delivery of video over the Internet. Edgeware’s strategy is to integrate and ensure interoperability with the best-in-class providers of other components of the ecosystem including client devices, conditional access systems, web portals, encoders and content management systems etc. The elements that Edgeware provides are the video streaming servers, which are optimized to be highly distributable and scalable. Edgeware also provides an asset caching and propagation system, which dynamically ensures that the most popular assets are distributed to servers as close to the subscribers as possible. Finally, Edgeware provides a sophisticated management system, for configuration, management and monitoring of a complete network of servers.

6.1 Edgeware Video Streaming Servers

The Edgeware WTV server is a combination of an advanced and highly integrated network device and a complete TCP service delivery platform that includes modular built-in solid state flash storage, hosting up to 6 TB of content and delivering 20Gbps from just 1RU unit. In contrast to generic servers, the product is a small, highly integrated, high performance and ultra low power appliance, offering a step-change in cost effectiveness. It is purpose designed to work as both a centralized streaming server or as a distributed cache for TV content anywhere in a network and offers efficient protocol support for any transmission mode including all variants of HTTP Adaptive Streaming.

Edgeware appliances enable operators to build a network of highly accelerated video servers which can be distributed deep into the operators’ networks. Compared with traditional server technologies, Edgeware appliances provide:

- The smallest form factor and highest reliability available
- Instant 10x peak bandwidth scaling
- Rapid drop-in acceleration at network bottlenecks
- Support for all video business models – Live, nPVR, catch-up TV etc
- Support for all video devices – STBs, Internet TVs, gaming consoles etc.
6.2 Support within the Edgeware Asset Caching & Propagation System

Microsoft® Silverlight®, Apple Quicktime and Adobe® Flash® are the leading frameworks for the creation and playback of Rich Internet Applications incorporating video. These include server programming tools for the creation of the presentation portals, tools for the creation of encoders to compress and format video for Internet delivery and client programming tools for creation of video players on different Internet connected devices.

All frameworks now support variants of HTTP Adaptive Streaming. The following describes how the Edgeware asset caching and propagation system supports these variants:

**Microsoft® Smooth Streaming (WTV)**

The WTV server acts as a reverse (server side) proxy to a Windows Server® running Internet Information Service 7 with the Media Service 3.0 extension. When running in this mode, caching replaces the need for Edgeware's Convoy asset propagation and this is therefore disabled.

The following is a description of the Smooth ecosystem and content flow:

Microsoft® or 3rd party encoders generate a single contiguous file per bit rate according to the ISO/IEC 14496-12 ISO Base Media File Format (MP4) specification. The files have the *.ismv extension and contain MP4 video fragments (and audio fragments if the video source also contains audio).

An XML-based server manifest file and a client manifest file is also generated. These describe available bit rates and other information required by the IIS 7 server and the Silverlight clients.

For on-demand Smooth Streaming, following encoding, all these files are copied to the IIS 7 server or are published if WebDAV is enabled. For Live content, rather than storing the fragments in MP4 containers, encoders deliver the fragments directly to Live Smooth Streaming publishing points on the IIS 7 server. The server itself then generates a manifest for clients, based on information provided to it by the encoder.
Requests for on-demand content stored on the IIS 7 server or for live content that is being delivered to a Live Smooth Streaming publishing point on the IIS 7 server should be directed to the Edgeware WTV server. E.g. a request for an on-demand asset stored on the IIS 7 server with the following URL http://iismedia7/BigBuckBunny/default.html should be directed to the IP address of the Edgeware server.

In addition, the Client Cache Settings of the IIS 7 server should be enabled to specify the Cache-Control header. This is used in the IIS 7 server to specify any intermediate caches the conditions and restrictions for caching of content.

Requests for live and on demand Smooth streams on the IIS 7 server are proxied by the Edgeware WTV server, ingested via HTTP and cached according to the specified cache Control directives. For a fragment that has been cached, subsequent requests are served directly from the WTV.

**Apple Live HTTP Streaming**

There are two modes of operation when supporting Apple Live HTTP streaming in the WTV servers: One for non-encrypted content and one for encrypted content.

For encrypted content the WTV server does not act as a reverse proxy but instead receives segmented content pushed directly from an encoder or content store. The architecture is very similar to Smooth Streaming except the (f)MP4 files are replaced with MPEG-2 Transport Streams (*.ts extension).

A segmenter creates and maintains an index file (*.M3U8 extension) containing a list of the media files. Apple provides a software segmenter but typically this is included as part of the encoder functionality. The index file and the stream segments are all pushed to the first WTV server using WebDAV.

WTV Servers can be added as edge caches to this first server and Convoy can be enabled to download transport stream segments according to their popularity.
For non-encrypted content, encoders deliver a single Transport Stream per bit rate without segmentation but with synchronized iframes. These can be ingested by the WTV server via FTP, HTTP or even scheduled UDP multicast for live transmissions. This option significantly reduces core network bandwidth, especially if live content is to be distributed to multiple servers.

A centralized WTV server can then perform the segmentation and create the index file. WTV Servers can be added as edge caches to this first server and Convoy can be enabled to download transport stream segments according to their popularity. Alternatively, each edge WTV server can ingest complete transport streams via scheduled multicast or according to popularity via Convoy and it can then perform segmentation on the fly.

Requests for Apple Live HTTP files should be directed to the IP address of the appropriate Edgeware server. Client software first reads the index file, based on a URL identifying the stream. This index specifies the location of the available media files, decryption keys (if applicable), and any alternate streams available. For the selected stream, the client downloads each available media file in sequence.

This process continues until the client encounters the #EXT-X-ENDLIST tag in the index file. If no #EXT-X-ENDLIST tag is encountered, the index file is part of an ongoing broadcast. The client loads a new version of the index file periodically. The client looks for new media files and encryption keys in the updated index and adds these URLs to its queue.

**Verimatrix VCAS Solution for Enhanced Apple HTTP Live Streaming**

The Apple HTTP Live Streaming protocol provides optional encryption of the video chunks using the AES-128-CBC block encryption algorithm. During encryption a 128-bit key is generated and placed in a “key file” on a server so it can be downloaded by the client. The location of the key file is given in the playlist. A new key can be given at any time. When a key file is encountered in the playlist, this key must be used to decrypt each subsequent chunk until another key file is encountered.
The URL provided to the client to retrieve the key file is HTTPS. This ensures that the connection to the server will at least be encrypted, even if it does not have server-side authentication. However, the protocol does not provide a way to ensure mutual authentication of the HTTPS session. There is no client-side certificate provisioning built into the protocol and there is no mechanism for the reporting of a unique client identifier from the device to the server. The lack of either of these features does not allow for enforceable entitlements against the video content encryption key.

To overcome these security limitations and enable HTTP live streaming video delivery suitable for a secured pay-TV service, Edgeware has confirmed interoperability with a security solution provided by Verimatrix. Using the Verimatrix VCAS 3 system, the 128-bit keys are managed and selectively distributed to the encoder and authorized clients only. VCAS 3 can support HTTP live streaming in a standalone OTT service configuration, or can be utilized as part of a unified security head-end supporting multi-screen deployments pairing OTT delivery alongside IPTV and DVB content distribution. Currently, to support this additional security, the encoder must perform the segmenting of the chunks prior to encryption and distribution to Edgeware servers (Mode 1).

Verimatrix VCAS Solution for Enhanced Apple HTTP Live Streaming (picture courtesy of Verimatrix)

There are four major components of the enhanced security solution. These pieces are the Edgeware WTV server, Live Streaming Encoder, Verimatrix VCAS Server and the Verimatrix Client.
The Live Streaming Encoder is responsible for receiving the video and encoding it into different bit-rates. The Envivio 4Caster™ C4 has been tested for interoperability with the other components of this system. Once the different bit-rates are encoded, they are sliced up into chunks which are synchronized and assigned sequence numbers. At this point, the chunks are ready for encryption. The encoder, or other sub-system responsible for encryption, will contact the Verimatrix VCAS Server and request the key for this particular piece of content at the current “now” time. The encoder then takes this key and encrypts each chunk with AES-CBC-128 encryption. The encoder will request the key from the VCAS Server with the “now” time for every chunk. This will ensure the key mutation interval is enforced. If the key is new, the key file URL will be written to the play list. The sequence number of each chunk is used as the initialization vector of the CBC encryption. These encrypted chunks are then placed on the Edgeware WTV Server.

While the chunks are being encrypted, the play lists for each bit-rate are being generated at the same time. These play lists contain the URLs to the chunked files, in sequence, on the Edgeware server. For each new key that is received from VCAS, the encoder will insert a “key file” URL. This key file URL points to the VCAS Server and contains the channel identifier and the time that they key was retrieved from the VCAS Server by the encoder.

The Verimatrix VCAS Server is responsible for the secure generation and authorized distribution of the keys used for chunk encryption. The keys to be used for encryption (typically performed by the encoder device responsible for segmentation) are provided to the encoder via a pre-defined interface [1]. The request for the key will be accompanied by the channel identifier and the key time, which is supplied by the play list written by the Live Streaming Encoder.

When accessing the VCAS key server for decryption keys, the Verimatrix ViewRight client provides a unique device identifier. This information is used by the VCAS Server to determine entitlement to the key for the particular channel. This entitlement check is performed against the middleware server. Upon determining that the client device is entitled to access to the content (e.g., a paid channel subscription, VoD asset, etc.), the VCAS server will securely transfer the key to be used for chunk decryption.

The Verimatrix ViewRight Web Client (for PC/Mac, iPhone, iPad, iPod Touch, STBs, etc.) executes on the device that will be registered with the middleware server. The Verimatrix ViewRight Client will provide a unique identifier of the device and thus ensure proper device authentication prior to entitlement checking and key distribution by the VCAS server. This unique identifier will need to be registered with the middleware / billing server via a process determined by the operator. After entitlement verification and secure key download to the device, the ViewRight client will provide the required keys to the decryption engine, resulting in a decrypted stream to be displayed by the player application.
Adobe® HTTP Dynamic Streaming

The WTV server acts as a reverse (server side) proxy to an Apache Server running the Adobe® Origin Module (VOD) or an Adobe® Flash® Media Server 4.0. When running in this mode, caching replaces the need for Convoy asset propagation and this is therefore disabled.

The following is a description of the Adobe® HTTP Dynamic Streaming ecosystem and content flow:

Adobe® or 3rd party encoders generate a single contiguous file per bit rate using the following file types: F4V/MP4 compatible files and FLV. For VOD content, an Adobe® File Packager command line tool is used to parse the file and translate it into fragments. This tool can also be used to encrypt files for use with Adobe® Flash® Access 2.0 DRM. Once created (and encrypted if used) fragments are written to files with the *.f4f extension. Each file can contain multiple fragments and quantity and duration of fragments per file can be optimized.

An XML-based client manifest file is created with the filename of the input file. This contains information about the codec, resolution and the availability of multi-bitrate files. In addition, a server index file is also generated. This contains information on the specific location of fragments within a file for the Origin Module to translate to byte range requests.

Adobe® HTTP Dynamic Streaming Ecosystem and Content Flow

For VOD, requests for content stored on the Apache Origin server should be directed to the Edgeware WTV server. E.g. a request for an on-demand asset stored on the server with the following URL http://localhost/media/webplayer.html should be directed to the IP address of the Edgeware server.

In addition, the Client Cache Settings of the Apache server should be enabled to specify the Cache-Control header. This is used in the Apache server to specify any intermediate caches the conditions and restrictions for caching of content.

Requests for on demand streams on the Apache Origin server are proxied by the Edgeware WTV server, ingested via HTTP and cached according to the specified cache Control directives. For a fragment that has been cached, subsequent requests are served directly from the WTV.
For live streams, the Apache Origin is incorporated into the Flash® Media Interactive Server (FMIS). Live streams are ingested over RTMP and segmented into *.f4f files. The FMIS server has a built-in Apache HTTP Server and uses the Origin Module to deliver the live content over HTTP. Requests for live streams are proxied by the Edgeware WTV server and cached in the same way as on-demand streams.

### 6.3 Edgeware Origin Management System

The Origin Management System provides a centralized tool for monitoring and configuring Edgeware video server systems. A web-based user interface provides status and statistics for a system in operation as well as a central portal for accessing the configuration tools of each server. It is possible to look at information from individual servers or aggregated data from a multi-server system. Since a web-based interface is used, the management tools can be accessed from any computer with a web browser and network access to the Origin server.

Origin is built around a database, where status and statistics data is stored. All servers send status information to Origin using standard syslog messages. The user interface provides both live status views and historical views, as graphs and lists.

Origin Management System

Origin is an optional tool that simplifies monitoring and management of Edgeware video servers. However, no functionality in the video servers relies on the use of Origin. Configuration of servers can be done by directly accessing the web configuration tool in each video server. Monitoring can be provided by third-party tools using either the SNMP interface available in all servers or by listening to the syslog messages.

### 7. References