



# Packet Ship *Streamline* Media Server

## Release Note

### 3.1 “Antigua” Major Release

Package	Version	Revision
ps-streamd	3.1.0	1
ps-index-mpeg2ts	3.1.0	1
ps-analyse-mpeg2ts	1.2.1	1



## Release Note

The 3.1 “Antigua” release of the Packet Ship Streamline media server is a major architectural revision of the server with much higher performance, new protocols and features and a much more modular design.

The new features include:

- The server is now a single daemon, `ps-streamd`
- There is a new index format “PSI2” which references I-frame data in the original asset file rather than (as previously) copying it, making the index much smaller.
- The server is much higher performance, making better use of multi-core CPUs, and is capable of multi-gigabit streaming on suitable hardware.
- RTSP/TCP is now supported, for Over The Top (OTT) public Internet streaming
- Apple's HTTP Live Streaming (HLS) is experimentally supported, allowing adaptive bit-rate streaming to iPhones and iPads, using a single asset file rather than the normal method of splitting into hundreds of chunks.
- Standard HTTP progressive streaming is also supported.
- Interface to Timeline IPTV recorder (a separate product) to provide catch-up TV and pause live TV services.
- Modular configuration allows variant configuration for different asset types and protocols.
- Seamless support for playlists with trick mode across playlist boundaries

The main documentation for the 3.1 Antigua release is in the fully updated Packet Ship Streamline Installation & Configuration Guide (ICG-SL). This Release Note just highlights changes for users who are already using previous releases.



---

## Single daemon

The three-way split – `ps-rtspd`, `ps-masterd`, `ps-pumpd` – of previous versions was designed for flexibility and to provide a measure of clustering functionality with multiple pumps. However the cross-process communication itself became a bottleneck for very large numbers of streams, and the split of the RTSP receiver from the data generating pump prevented implementation of RTSP/TCP. The complexity of managing three daemons also gave both us and our customers occasional headaches!

In the Antigua release everything has been combined into a single daemon, `ps-streamd`. This is highly modular, so that functions that are not required can be turned off. The server can now support much higher numbers of streams and can also support TCP protocols such as RTSP/TCP and HTTP Live Streaming.

There is now a single configuration file, `/etc/packetship/streamd.cfg.xml`, and a single log file, `/var/log/packetship/streamd.log`.

---

## New index format: PSI2

The Antigua release uses a new “index file” format, PSI2. The default suffix for these files is “.psi2”. The new format has a number of changes from the previous one:

### Referenced data

The data for the “index frames” used in trick mode is no longer copied into the index file but referenced at its location in the original asset. This makes the index files much smaller – less than 1% of the original asset, typically.

### VBR pacing

The Antigua video server now uses the index file for VBR rate pacing rather than scanning the asset itself; this improves performance and also allows for other formats to be handled in the indexer in the future without needing a video server update.

This means that for VBR streaming the index file is now essential (in its absence the server will revert to CBR only if the rate is specified manually), and hence we recommend indexing is now a standard part of content ingest.

---

## Higher performance

As well as the higher performance brought about by the combination into a single daemon, the Antigua release is able to make better use of multiple CPUs, by distributing the stream generation load evenly between a number of realtime threads, ideally one for each core.

The number of realtime threads to use is configured in the **<threads>** element of the 'realtime' **<group>** in `streamd.cfg.xml`.



---

## RTSP/TCP

While UDP/IP streaming is ideal for managed networks, in the public Internet UDP traffic often suffers packet loss, and may be difficult to pass through firewalls. The alternative is to stream the data with TCP/IP across the same connection used for the RTSP control itself.

In the previous version, the split of `ps-rtspd` and `ps-pumpd` meant that we could not implement RTSP/TCP. However with the new single daemon architecture this becomes possible, and Antigua fully supports RTSP/TCP.

RTSP/TCP is enabled by the client requesting Transport “RTP/AVP/TCP”. VLC can be forced to do this if you check the “Use RTP over RTSP (TCP)” option in the RTP/RTSP demuxer preferences.

---

## HTTP Live Streaming

Another way of streaming video over the public Internet is to use HTTP instead of RTSP. Simple progressive download (which Antigua also supports) is fine for simple linear streaming, but Apple (among others) have come up with a way of using HTTP for streaming, HTTP Live Streaming (HLS). This splits the files into lots of tiny chunks (typically 2-10 seconds long), which gives both ability to accurate seek (but in current clients, not trick play) and adaptive bitrate streaming where the client can switch between multiple playlists of chunks to adapt to the available bandwidth.

While splitting files up like this allows use of standard Web servers, large assets become difficult to manage, and because Web servers are stateless there is no admission control and hence no guaranteed quality of service. Because the Streamline video server already has all the stream state and admission control infrastructure, we decided to implement it slightly differently...

The Antigua server is able to take a standard Transport Stream file and index file – as used for RTSP streaming – and make it appear as if it has been split into hundreds or thousands of chunks, and generate the associated playlists for it. We call this “virtual playlists”. If multiple versions of the asset are available (multiple assets with the same ID) it will offer rate-adaptive alternatives.

The current support for HLS should be regarded as an experimental proof of concept; future versions will optimise and improve it. In particular, we are interested in seeing how trick mode can be supported over HLS, which it currently lacks.

The HLS support is configured in the **<hls>** controller section of `streamd.cfg.xml`.

---

## Catch-up TV

Alongside the Antigua release, Packet Ship is launching a new product, Timeline IPTV Recorder. This is a “circular buffer” capture daemon which can record a continuous loop of video from multicast IPTV channels, along with their associated EPG data.

The Streamline video server now provides an interface to the recorded video, mapping the recordings into a virtual URL space by channel and start time (ISO time stamp). For example:

`rtsp://server/tv/BBC1/20110129T183000`



Starts a stream from 18:30 on 29<sup>th</sup> January 2011. The client can then pause, fast-forward and rewind from there. For features such as pausing live TV, the client can obtain a stream from the current point of recording by simply leaving off the timestamp part of the URL:

```
rtsp://server/tv/BBC1
```

---

## Modular configuration

The new architecture of the video server in Antigua is much more modular than before, and offers the choice of multiple instances of each module. The effect of this is that different types of content or different access protocols can be configured entirely differently throughout the streaming data flow, making it much more flexible and able to handle a wider range of content. For example, the buffer sizes needed for large numbers of low-bitrate mobile streams are likely to be a lot smaller than for HD VOD streams.

The module architecture of the new server is described in the first chapter of the new Packet Ship Streamline Installation & Configuration Guide (ICG-SL). This is worth reading even if you just use the configuration files for guidance after that!

---

## Seamless playlists

The way that playlists are implemented in the Antigua video server has changed considerably from the previous version. Previously, each element of a playlist was effectively an independent stream, and rate pacing and buffering was restarted as the server moved from one element to the next. Also, trick mode didn't work across playlist boundaries.

In Antigua, the rate-pacing and buffering is continued across playlist boundaries, making transitions from one to the other seamless. In conjunction with the 'ts-timing-adjust' and 'ts-cc-adjust' Output Filters, you can now make a playlist look exactly like a single asset to the client.

Trick mode is also now continuous over playlist boundaries, allowing you to fast-forward and rewind across an entire playlist as if it was a single asset. This is what enables seamless fast-forward/rewind of the multiple recorded files produced by Timeline IPTV Recorder.