

Swinburne Research Bank

<http://researchbank.swinburne.edu.au>



But, J., & Armitage, G. (2005). Implementing encrypted streaming video in a distributed server environment.

Originally published in *Proceedings of the ACM SIGCHI International Conference on Advances in Computer Entertainment Technology, Valencia, Spain, 15–17 June 2005* (pp. 322–325). New York: ACM.

Available from: <http://doi.acm.org/10.1145/1178477.1178537>

Copyright © ACM, 2005. The definitive version was published in Proceedings of ACE (2005).

This is the author's version of the work, posted here with the permission of the publisher for your personal use. No further distribution is permitted. You may also be able to access the published version from your library. The definitive version is available at <http://dl.acm.org/>.

Implementing Encrypted Streaming Video in a Distributed Server Environment

Jason But

Centre for Advanced Internet Architectures
Swinburne University of Technology
John Street Hawthorn
Victoria, 3122, Australia
jbut@swin.edu.au

Grenville Armitage

Centre for Advanced Internet Architectures
Swinburne University of Technology
John Street Hawthorn
Victoria, 3122, Australia
garmitage@swin.edu.au

ABSTRACT

Technical issues are not the only ones preventing large scale introduction of online streaming video services. Unlike generic web browsing applications, streaming video imposes greater demands on network resources. Caching of content through the use of distributed servers has been proposed as a solution to reduce resource requirements and improve scalability. Video caching presents a unique challenge to copyright protection schemes, particularly if we consider provision of functionality such as indexed and high-speed playback modes. This paper discusses the issues involved in implementing copyright protection for cached streaming video, concluding with a set of requirements for any proposed scheme.

Categories and Subject Descriptors

K.4.4 [Computing Milieux]: Electronic Commerce—*intellectual property, security*; K.5.1 [Computing Milieux]: Hardware/Software Protection—*copyrights*

General Terms

Streaming Video, Copyright Protection, Video Encryption

1. INTRODUCTION

Streaming video over the Internet generally consists of low bitrate MPEG-4 or similarly encoded content [20]. The recent availability of broadband Internet is increasing the accessibility of streaming video and other multimedia Internet applications [6]. Regardless of the increase in available bandwidth, the concept of video caching, through a distributed streaming server environment, must be implemented to both improve system scalability and minimise usage of network resources [12, 16].

In this paper we discuss the issue of copyright protection in a distributed streaming server environment. Given the

costs associated with producing and distributing video content, copyright protection is a key issue in the eyes of content owners. However, end users are equally important and expect a certain level of functionality such as provision of indexed and high-speed playback modes. We propose a set of requirements that should be met by any copyright protection scheme in the context of streaming video.

2. INTERNET STREAMING VIDEO

Most video compression algorithms are lossy, playback of compressed video suffers degradation in quality when compared to the original. MPEG-1 was originally developed for Video CD, providing VHS quality at bitrates of 1.2Mbps. At the time, MPEG-1 provided higher compression ratios than existing codecs by removing temporal redundancy between frames as well as spatial redundancy within a frame [14]. Newer codecs, such as MPEG-4, DivX and QuickTime, are able to provide new VHS quality video at bitrates of 200-500kbps [15].

The network bandwidth required by these video compression algorithms is higher than traditional "Last Mile" connections (which typically offer less than 56kbps). Newer broadband customer access technologies such as xDSL and cable provide greater bandwidth, typically ISPs offer 256kbps-1Mbps. While capable capable of supporting video, it requires a large proportion of the total available bandwidth.

A typical streaming video configuration involves a single server streaming content across the network to the end consumer. There is a good likelihood that most streams must traverse the Internet core [4]. While available bandwidth within the core is usually high, the core is also responsible for carrying and routing other traffic, video traffic is subject to interference from these flows [6].

The presence of these other traffic flows impacts on both the available bandwidth and levels of network jitter experienced by the video stream. Changing network conditions have the effect of introducing variations in router transgress times, with resultant network jitter. Also, because the Internet is a packet switched network, two packets in the video stream may take different paths through the network, resulting in different network transmission times and thus jitter.

Content providers must be assured of return on their invest-

ment, and content must be attractive to consumers [3, 7, 13]. Copyright terms offered by the provider must be accepted by recipients before content is made available, and content must only be made available to those who have accepted these terms.

These factors combine to make video streaming difficult in today's Internet environment. Video is typically streamed directly from the server to each client, imposing strict requirements on network resources.

3. HIGH QUALITY STREAMING VIDEO

Due to the long duration of feature length movies, a typical consumer would prefer to watch them in comfort on a large display, such as a TV. High quality movies mean increasing the encoded bitrate.

3.1 Network Issues

Compressed video is encoded at a variable bitrate and must be presented to the decoder at this variable rate [14]. To simplify implementation, video is typically streamed at the average encoded rate and buffered at the client which then feeds the decoder at the required rate. This requires regular arrival of data to ensure against an empty buffer [14]. The more hops in the link between the streaming server and the client, the greater the likelihood that transient network conditions will cause periods where the required bandwidth is not available. This is evident even today when streaming low bitrate Internet video [20]. Streaming video can be more sensitive to variations in inter-packet arrival times (jitter), than network round trip time [21]. Jitter affects streaming video by potentially delaying the arrival of individual packets at the client. The effects can be minimised by maintaining a larger client buffer, effectively increasing the allowable time by which a packet can be late.

3.2 Caching Video Streams

The major network requirements for streaming high bitrate video are available bandwidth and minimal jitter. Caching of video can improve performance (Figure 1). Caching of web pages is often implemented by managers of LANs or ISPs, where copies of common web pages are stored on a local server. When one of these pages is requested, it is served from the cache rather than fetching it from the source. Benefits include reduced core network traffic and faster user response time [19].

The same benefits could be realised with streaming video. A cached video streaming service would ensure that as the proportion of video traffic increases, the growth in demands on core network resources is minimised [12, 16]. However, interactive streaming video (offering features such as pause and indexed playback) cannot be managed by a cache that simply stores and replays content, video caches must also provide these services. One possible solution is through the use of distributed servers [16, 4, 5].

3.3 Distributed Servers

We configure a number of distributed streaming servers, each servicing a local area via a fast, over-engineered network which would generally not suffer from the bandwidth or jitter issues which affect the delivery of video services. As

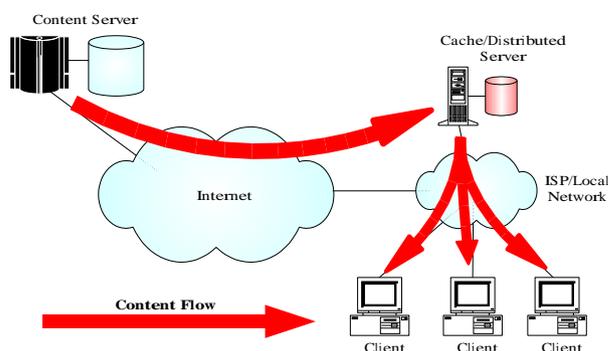


Figure 1: Cached Content Delivery Model

with web caches, these servers would not store all available content. A central server stores the original content, which is then delivered to the distributed servers for caching and ultimate delivery, limiting the use of the network core for video services to one-off transfers of content [16, 4].

4. COPYRIGHT CONCERNS

Copyright ownership enables content owners to enforce restrictions, such as payment of fees, on every screening of their asset. In practice, streaming of commercial content to a consumer will involve some payment back to the copyright owner [3, 7, 13]. The recent legal battles fought over digital and streaming audio with respect to Napster and similar businesses demonstrate just how seriously content owners consider the issue of copyright to be [2].

One of the primary requirements for a commercially viable streaming video service is the availability of content that consumers are willing to pay to access. Commercially successful content will largely be feature films and sporting broadcasts. Unless concerns about copyright violation are addressed, this content will not be made available, and the streaming video service will not succeed commercially.

4.1 Protection of Content

With the advent of better encoding algorithms, and the widespread availability of broadband Internet, downloading (and streaming) of audio content became feasible and widespread copyright violation began to occur [2]. This used to be a minor issue as the equipment to make copies in bulk was expensive and difficult to acquire. Today however, CD writers are ubiquitous and small MP3 files are easily transmitted over networks. Copyright violations of audio content has made video content owners more wary - DVDs have a built-in encryption mechanism [3]. Similarly, any future content delivery systems, such as video streaming, must take copyright protection into account.

4.2 Watermarking

Watermarking involves encoding extra information within the digital bitstream. Physical watermarking can often be seen on television broadcasts in the form of a station logo. Digital watermarking uses the theory of information hiding to conceal a message within a digital bitstream [1]. Key features include it being impossible, or at least very difficult,

to change or remove the encoded watermark and that even if the content is edited, the original watermark is retrievable from the newly created image.

Released content is encoded with different watermarks for each distributor. It is then possible to trace the source of stolen content by extraction of the encoded watermark [1]. Digital watermarking provides passive protection against copyright violation. It does not stop theft, but assists in tracking down the perpetrator after the fact. Subsequent legal action can discourage making of illegal copies [1].

4.3 Encryption

A more active form of content protection uses encryption, which modifies the original bitstream such that it cannot be viewed unless a decryption key is known. The original bitstream is passed through a cipher, the resultant bitstream is statistically random and provides no clues as to the nature of the original data. Different ciphers provide differing levels of security (defined by how difficult it is to retrieve the original bitstream) [17].

To break any cipher, an attacker can try each possible key (known as a "Brute Force" attack), until the correct key is found. For N keys, it will take on average $N/2$ attempts to find the key. For each extra bit in key length, the key space doubles. Today's computing environment requires key lengths greater than 128 bits to be secure against a "Brute Force" attack [17]. Further, the size of the key space is irrelevant if there is a weakness in the cipher such that it is possible to determine the key through a shortcut.

Key Management defines the procedures and protocols used to protect and transfer secret keys between parties. It is likely that a Public Key scheme might be used in implementation of Key Management for streaming of encrypted video. Public Key ciphers are too slow for general purpose use [17], even more so with the high bitrate of encoded video. However, Public Key schemes would enable secure delivery of cipher keys to authorised users, and are already used in key exchange roles in the `ssh` and `https` protocols [11, 10].

Encryption of content provides active protection against copyright violations. Following theft, the attacker must still circumvent the cipher prior to gaining access to the content. Encryption does not protect against distribution of stolen content once the cipher has been broken, ideally our system should combine both passive and active techniques.

5. ENCRYPTED VIDEO IN A DISTRIBUTED SERVER ENVIRONMENT

When considering content protection, we should do so in a distributed server context. Watermarking is a transparent technology. All watermarked digital video will function with all video streaming tools. The hidden data does not affect the formal video bitstream definition and will not interfere with existing systems [1]. An encrypted video bitstream is non-standard. When considering ciphers, we must take the design and implementation of both streaming servers and client playback tools into account. Either the cipher should be compatible with existing products, or the streaming video products must be modified to support the cipher.

5.1 Commercial Issues

The cost of implementing a true distributed server system can be prohibitive. A working system may require many distributed servers to service a large city, for a more widespread service, the installation costs will be extremely high [5]. It may be more economically feasible to spread these costs over a number of companies, in effect numerous operators implement a small section of a distributed video server network. This network is then utilised by content owners and their customers as a means of content delivery [5].

What type of cipher is required to protect content streamed from a multi-platform distributed server system? First consider the option of modifying streaming video products to comply with the cipher. This requires each streaming server supplier to modify their product to support streaming encrypted video. The usual approach is to install the video on the server, which encrypts the data as it is being streamed.

This does not sit well with content owners, especially if the server network is maintained by independent operators. The content owner must place their trust in both the security of each server to protect the content and in each individual server operator not to steal content [4, 5]. There is also the difficulty in getting all streaming server developers to support the selected cipher.

Now consider the option whereby the cipher is compatible with existing streaming video products. The cipher must be selected such that the encrypted bitstream can be successfully installed on and streamed from existing streaming servers. It also requires that the bitstream can be decrypted and played back on a multitude of playback devices.

This is more comforting to content owners. All copies of the content throughout the network will be stored in encrypted form, and will remain protected in the case of an individual server being poorly secured, or against unscrupulous server operators [13, 5]. We also acquire the following benefits:

- Cipher code can be produced by cipher developers rather than streaming server developers.
- Cipher upgrades can be made without re-writing the streaming server code.
- New streaming server products can be developed without considering the cipher implementation.
- Competition exists between server developers as products are not differentiated based on their content protection features.

5.2 Required Cipher Properties

For a cipher to correctly function with existing streaming video products, it is necessary to consider how streaming video server and client software is implemented. The different playback modes we need to consider include normal streaming, pause, indexed playback and high-speed playback (not supported by all servers).

To support these playback modes, a streaming server must partially decode the video bitstream [8]. The streaming

server must be able to locate key reference points within the encrypted bitstream in order to provide these playback modes [9, 18]. The cipher must successfully resynchronise during playback to support all playback modes. For indexed playback, this requires resynchronisation at each possible index point [9, 18]. For high-speed playback the delivered stream is typically generated by extraction of key frames from the source bitstream, this requires decryption of each of these individual frames [9, 18, 8]. Finally, the decryption module must not be embedded into the video decoder, providing similar benefits as keeping the cipher independent of the server implementation.

6. CONCLUSION

Digital streaming video is available on the Internet today, provided that network and server capacity are not overloaded. In considering widespread availability of streaming video encoded at high visual quality, it is imperative to both decrease the demands on network resources while at the same time improving system scalability. This is best achieved through the adoption of a video caching scheme.

In today's environment, copyright protection of content with commercial value is an important issue that must be addressed. However, end users also demand a certain level of functionality, particularly the provision of paused, indexed and high-speed playback modes. This is problematic when copyright protection must also be provided.

While watermarking is essentially transparent, it is imperative that a video cipher functions in the context of providing this functionality. A suitable cipher must be independent of the streaming server and decoder implementations, stream and decrypt the encrypted bitstream in all playback modes and be secure against attack.

7. ACKNOWLEDGMENTS

The work presented in this paper has been developed as part of Jason But's PhD studies at Monash University, Australia.

8. REFERENCES

- [1] N. Abdulaziz. *Digital Watermarking and Data Hiding in Multimedia*. PhD Thesis, Monash University, Australia, 2001.
- [2] A. Berschadsky. RIAA vs. Napster: a window onto the future of copyright law in the internet age. *Journal of Computer & Information Law*, 18(3):755–789, Spring 2000.
- [3] J. Bloom, I. Cox, T. Kalker, J.-P. Linnartz, M. Miller, and B. Traw. Copy protection for dvd video. *Proceedings of the IEEE*, 87(7):1267–1276, 1999.
- [4] J. But and G. Egan. Designing a scalable video on demand system. In *International Conference on Communications, Circuits and Systems (ICCCAS'02)*, pages 559–565, 2002.
- [5] J. But and G. Egan. Designing an affordable scalable video on demand system. In *2nd ATCRC Telecommunications and Networking Conference and Workshop*, pages 16–21, 2002.
- [6] D. Cop. The broadband boom. *Telephony*, 239(12):60–66, September 2000.
- [7] S. Fist. Dial m for movie: Video-on-demand. *Australian Communications*, pages 65–72, August 1994.
- [8] E. Frimout, J. Biemond, and R. Lagendick. Extraction of a dedicated fast playback mpeg bit stream. *Proceedings of the SPIE*, 2501:76–87, 1995.
- [9] D. Gemmell, H. Vin, D. Kandlur, P. Rangan, and L. Rowe. Multimedia storage servers: A tutorial. *IEEE Computer*, 28(5):40–49, May 1995.
- [10] IETF. *HTTP Over TLS*. IETF Network Working Group Request for Comments RFC2818, May 2000.
- [11] IETF. *SSH Protocol Architecture*. IETF Network Working Group Internet Draft, October 2003.
- [12] J. Kangasharju, F. Hartanto, M. Reisslein, and K. Ross. Distributing layered encoded video through caches. In *Proc. 20th INFOCOM Conference*. IEEE, 2001.
- [13] N. Memon and P. Wong. Protecting digital media content. *Communications of the ACM*, 41(7):35–43, 1998.
- [14] J. Mitchell, W. Pennebaker, C. Fogg, and D. LeGall. *MPEG Video Compression Standard*. Chapman & Hall, ISBN 0-412-08771-5.
- [15] F. Pereira. Mpeg4: A new challenge for the representation of audio-visual information. In *International Picture Coding Symposium*, pages 7–16, March 1996.
- [16] M. Reisslein, F. Hartanto, and K. Ross. Interactive video streaming with proxy servers (extended version). In *Technical Report*. GMD FOKUS, June 1999.
- [17] B. Schneier. *Applied Cryptography: Protocols, Algorithms and Source Code in C*. John Wiley & Sons, ISBN 0-471-11709-9.
- [18] D. Wu, Y. Hou, W. Zhu, Y.-Q. Zhang, and J. Peha. Streaming video over the internet: Approaches and directions. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(3), 2001.
- [19] S. Zander and G. Armitage. Dynamics and cachability of web store: Implications for inverted capacity networks. In *11th IEEE International Conference on Networks (ICON 2003)*. IEEE, September 2003.
- [20] Y.-Q. Zhang. Streaming video over internet - issues and new developments. In *IEEE Workshop on Signal Processing Systems*, page 23. (SiPS 99): Design and Implementation, October 1999.
- [21] L. Zheng, L. Zhang, and D. Xu. Characteristics of network delay and delay jitter and its effect on voice over ip (voip). In *IEEE International Conference on Communications (ICC 2001)*, pages 122–126. IEEE, June 2001.