Content Protection and Delivery System for the Internet

William Macy, Robert Liu, Matthew Holliman, Boon-Lock Yeo and Minerva Yeung
Microcomputer Research Labs, Intel Corporation
Santa Clara, California

Abstract

We describe a multi-level content protection system designed for Internet-based content delivery. Our system consists of a server module and a client module. The delivery of protected content is the same for all clients, but the decoding is dependent on the client's level of authorization. The different levels of protection schemes used by the system include fast encryption exploiting the properties of MPEG streams, perceptual scrambling of visual content, and watermarking for both fingerprinting and copyright protection.

Introduction

The Internet has been growing very rapidly, with over 100 million users by the end of 1998. This growth is due to the wide variety of new opportunities that the Internet brings. Areas that can take advantage of these opportunities include business, science, entertainment, social interaction, publishing, real-time information delivery, transaction processing, libraries, video conferencing and many more.

The bandwidth available to users has increased as the Internet has grown. Digital subscriber lines (DSL), which use regular telephone lines, and cable modems, which use cable networks, are becoming widely available at affordable prices. Currently these services provide bandwidth that is about 30 times that of high speed modems. Carriers are investing in the infrastructure required to support higher bandwidth transmission. Examples include the construction of a global optical-fiber under-sea cable that will transmit data at rates of up to one terabit per second, and development of faster routers and switches that will eventually support terabits per second routing throughput.

As transmission rates increase, the quantity and quality of images, audio and video will increase. However, a major problem faced by content providers and owners is protection of their material. They are concerned about copyright protection and other forms of abuse of their digital content. Unlike copies of analog tapes, copies of digital data are identical to the original. There is no limit to the number of exact copies that can be made. In addition, equipment that can make digital copies is widely available and inexpensive. Content owners need to be assured that their material will have copyright protection and that they will be properly compensated and acknowledged. Copyright protection has a number of aspects. The most prominent of these is the right to limit the distribution of copies. Other aspects include protection of their work from alteration, control over the public display of the material, and rights over derivative works that are modified versions of the original.

Technology can play a major role in providing the infrastructure for content protection. Data encryption and scrambling technology are means of assuring that content is delivered securely and that revenue is collected. Data encryption prevents an unauthorized recipient from viewing images or video or hearing audio. Scrambling allows an unauthorized recipient to view images and video or hear audio in a degraded form. Unfortunately, data encryption and scrambling cannot protect content after it has been decrypted and unscrambled. Digital watermarking is intended to be a solution to this problem that complements encryption and scrambling.

We have addressed the problem of content protection and delivery by developing a comprehensive system that includes an MPEG [1] encoder and player that combines the technologies of encryption, perceptual scrambling, and watermarking to encode both ownership and recipient information. The system can be used for transmission of video, audio or images. We will focus on protection and delivery of video because this is the feature we have implemented.

Our Content Protection and Delivery System

A client-server model can describe the overall architecture of our system. Figure. 1 shows the server part of the content delivery system, where a request for delivery is received. In this setup, an end-user computing system, called the client, issues a request, possibly with identification and order information. The identification and order information are used to determine the protection level. The server will first authenticate the client, possibly by a combination of passwords and user id.

Upon authentication, the server will watermark the video or audio stream to be delivered, visually scramble some data components, and scramble the data bits, which is equivalent to encrypting data by blocks, before sending the data to the user. The watermarking process imprints the recipient information for future tracing, perceptual scrambling achieves intentional quality degradation, and data scrambling, ie. encryption, ensures reliable transmission and disables unauthorized viewing. An additional watermark can be pre-inserted in the content on the server to encode copyright and ownership information. A key, or possibly multiple keys, is produced from a transformation of this information.

The server also provides the capacity to store the requester identification and auxiliary information, which may be in encrypted or hashed state, in a database. The information can later be used for tracing or monitoring purposes, or for subsequent delivery requests. A secure transaction server can also be set up to collect the revenues for content delivery.

Figure 2 shows the content playback system on the client side. Upon receiving the data delivered, either in its entirety or partially if the data is streamed, the system first issues an authentication check to fetch key information from the client computing system. Such information can come from hardware and/or software based identification. The playback system then attempts to perform bit-level descrambling, ie. decryption, on the data. The key for the descrambling or decryption is based on information such as a user id and passwords. The input information is combined and transformed into a key. The transformation used to combine the information is identical to that used by the server.

Based on the authorization level, various qualities of video can be played. If proper authorization is not obtained at the basic level, the data stream cannot be descrambled for playback. In this case either no video is played back, or a few unencrypted frames which serve as a preview can be displayed. Figure 3 shows the response of the player for the case in which there is no authorization at the basic level.

The second level uses auxiliary user-specific access control data such as user identification information and passwords. If the second level of authorization is not matched, a lower quality video/audio/image is played back. This is because the second key cannot be properly generated to visually descramble the video/image, or perceptually descramble the audio. An example of a visually scrambled video frame is shown in Figure 4. Compared with the original frame shown in Figure 5, the visually scrambled version is significantly degraded yet perceptible.

The third level of protection uses digital watermarking. If there is misappropriation of the video, the hidden watermark can be extracted to identify the recipient who initially requested the data and/or the owner of the source data.

The watermark can be a combination of server-specific information and requester information. This allows tracing of the abuser if the recipient can be properly identified. The ownership data embedded by digital watermarking technology can be used to identify the owner of copyrighted material, and can allow copy control in systems that detect a watermark first to check the authorization code before any recording action.

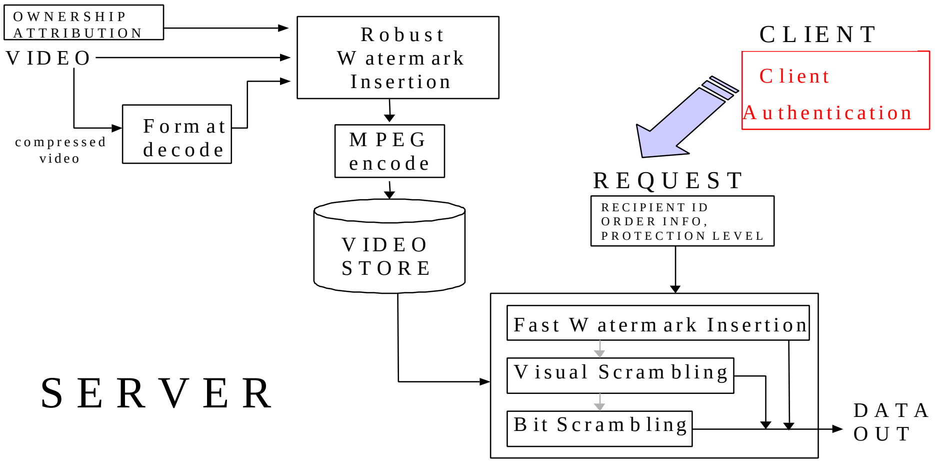


Figure 1 The Content Delivery System Components

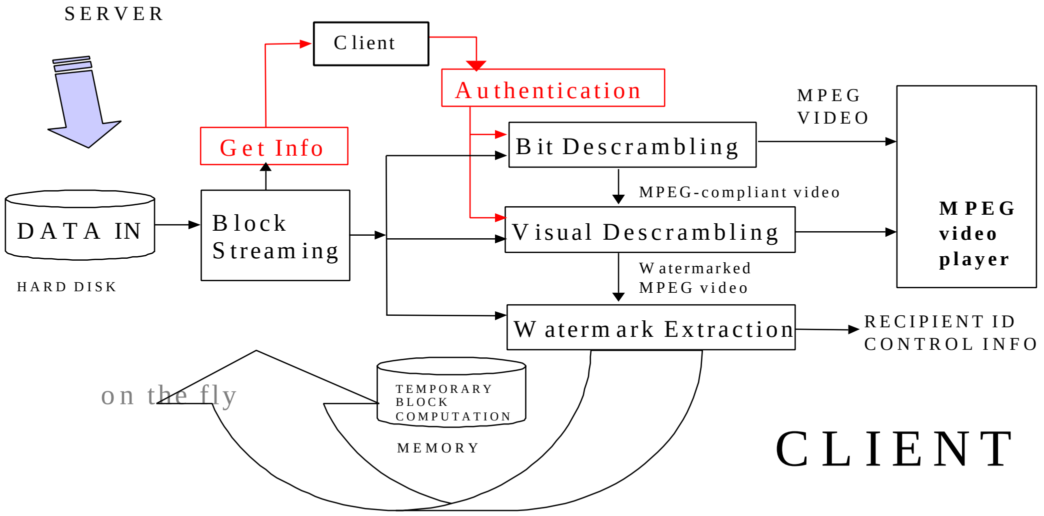
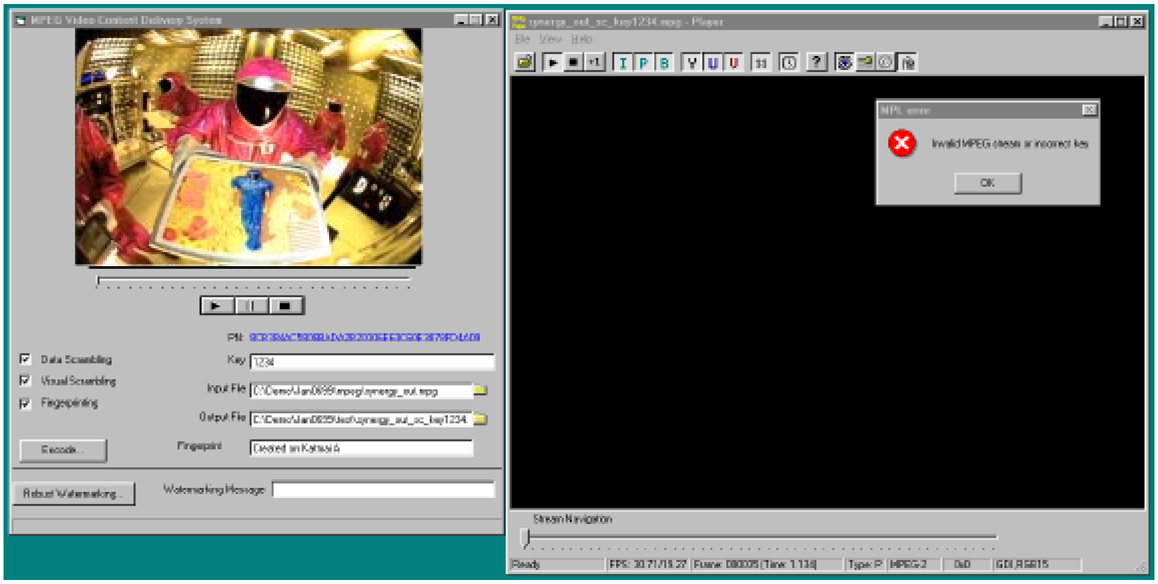


Figure 2 The Content Playback System


MPEG Video Content Delivery System
synergy_out_sc_key1234.mpg - Player
File View Help
Encode...
Robust Watermarking...
OK
MPL error

PIN: [illegible]

☑ Data Scrambling

☑ Visual Scrambling

☑ Fingerprinting

Key: 1234

Input File: C:\Demo\Jan0699\mpeg\synergy_out.mpg

Output File: C:\Demo\Jan0699\test\synergy_out_sc_key1234

Fingerprint: Created on Kaitai A

Watermarking Message:

Invalid MPEG stream or incorrect key

Stream Navigation

Ready | FPS: 30.71/18.27 | Frame: 000035 (Time: 1.134) | Type: P | MPEG-2 | 0x0 | GDI,RGB15

Figure 3 (left) Content delivery system (right) Content playback system - authentication failed.

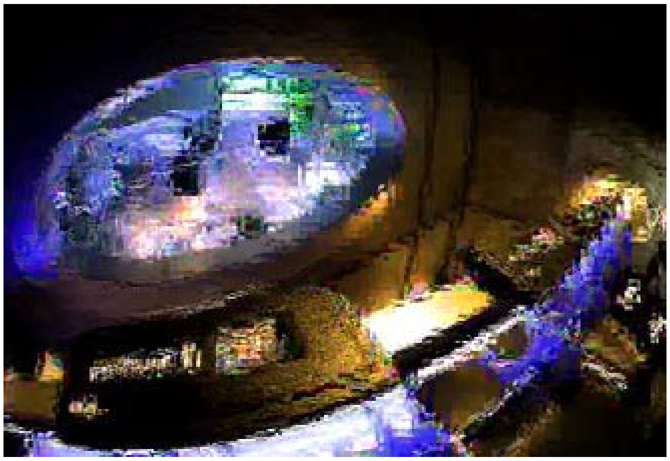


Figure 4 Visually scrambled image.

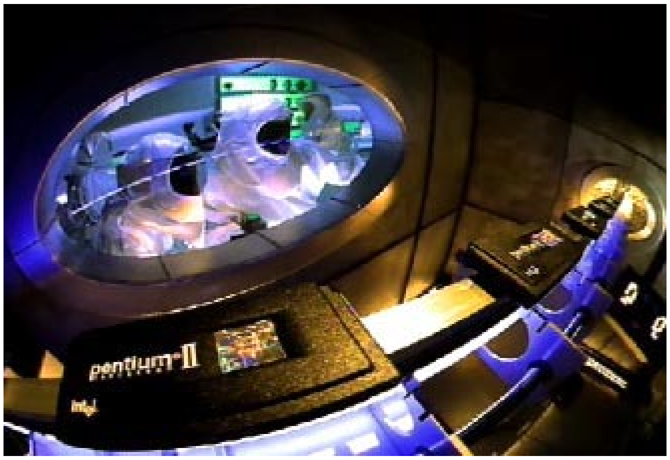


Figure 5. Original image

Techniques for Content Protection

1.Encryption

Real-time encryption and decryption of digital video can be problematic, due to the fact that conventional cryptographic approaches [2] are often not fast enough to adequately support the high data rates involved, particularly when implemented in software. In practice, digital video is typically stored and transmitted in a compressed format that removes much of the redundancy in the source data. For example, digital video may often be coded with MPEG-1 (about 1.5 Mbps) or with MPEG-2 (5-8 Mbps) which is more suitable for higher bandwidth applications. However, traditional cryptosystems may still not be able to handle such reduced data rate.

An MPEG video stream consists largely of variable-length Huffman codewords (VLCs). The encoding process has therefore greatly reduced the structure of the input stream, i.e. the plaintext video. Furthermore, little useful information is found in the stream headers, so that essentially all that need be encrypted to protect the stream is the VLC data in each coded picture. In terms of MPEG coded video, this implies we need only encrypt from the slice layer onwards. In addition to reducing the workload of the encryption process, the amount of known plaintext (e.g. system start codes) available to an attacker is diminished when adopting this approach.

Our particular implementation divides the coded bitstream of a picture into disjoint blocks, $B_0 \ldots B_n$. The ciphertext of the first block, C_0, is computed as $E_K(B_0,$ *block location)*, the block location is defined in terms of absolute position from the start of the sequence. E_K may be drawn from a symmetric or asymmetric cryptosystem depending on the application, and encrypts its argument with the key K. The ciphertext of the remaining blocks in the image is then computed as $C_i = B_{i-1}\ XOR\ B_i$, $1 \leq i \leq n$. In an analogous fashion, the decryption process can be described as $B_0 = D_{K'}(C_0,$ *block location);* $B_i = B_{i-1}\ XOR\ B_i$, $1 \leq i \leq n$. $D_{K'}$ represents the complementary decryption process to E_K, using a key K'. Since the Huffman coding resulted in essentially random bit strings B_i from a cryptographic point of view, this scheme essentially amounts to a one-time pad, where the security of the scheme depends on the security of the encryption algorithm applied to the initial block and the size of each block. Such an approach appears to represent a reasonable compromise between speed of execution and required security.

2. Visual Scrambling

We employ a method to visually scramble video by changing DCT coefficients. Both MPEG-1 and MPEG-2 encode quantized DCT AC coefficients using a combination of run-length and Huffman coding, in a manner similar to that of the JPEG still image compression standard. Specifically, non-zero AC coefficients are paired with an associated run of zero values and the combination is encoded using Huffman coding. The VLC for a run-length/coefficient pair is determined as a function of the magnitude of the non-zero coefficient and the length of the zero run; the sign of the coefficient is encoded as a separate bit of information. Flipping the sign bits of encoded coefficients has the effect of scrambling a DCT block. In addition to its efficiency, this operation has no effect on the bit rate of the compressed sequence.

The main factor that controls the quality of the scrambled video is the choice of the set of coefficients whose signs are flipped. MPEG encodes DCT coefficients in zig-zag scan order, providing a convenient heuristic measure of coefficients' significance. The choice of point in the zig-zag scan order beyond which signs of coefficients are pseudo-randomly changed therefore determines the strength of the scrambling; a point close to the beginning of the scan results in a greater degree of scrambling than one later in the scan.

The other important factor determining the quality of the scrambled video is the set of blocks chosen for modification. Typically the choice is made from intra-coded, nonintra-coded, or both. The degradation in the coded signal can generally be made substantially more severe by modification of intra-coded blocks than is possible by modification of nonintra-coded blocks only. However, scrambling of both kinds of blocks is advantageous as the degradation of nonintra-coded blocks can potentially maintain more consistent error propagation throughout the video.

3. Watermarking

A digital watermark is a signal that is added to digital content such as audio, video, and images. Watermarks are either visible or invisible. Visible watermarks typically display a message, such as a copyright logo. On the other hand, invisible watermarks should not change the appearance of the original. An invisible watermark signal can be detected or extracted from a watermarked image to determine information about the digital content in which it is embedded. Purposes of the information include protection, control, and description of the content. Of these, protection in particular has attracted a great deal of attention recently. One form of protection is the embedding of ownership information for copyright enforcement. Another is that of fingerprinting for recipient tracing, in which a unique watermark is inserted in each recipient's copy for the purposes of tracking illegal copies. Still another is authentication of content to verify that it has not been modified by an unauthorized user.

The characteristics required of a watermark depend on the particular application. For example, watermarks intended to establish ownership or tracing information should be robust to typical processing operations such as low pass filtering, scaling, and lossy compression, as well as other malicious attacks intended to remove the watermark. In contrast, fragile watermarks are designed to change if the content has been altered, and are therefore intended for purposes of authentication. In this paper, we focus exclusively on robust watermarking of video.

One difference between watermarking of video and information embedding in still images is that there is a much larger amount of data in video, so that it is desirable to use a relatively efficient method to watermark video. Another difference is that adjacent frames in video are generally highly correlated, which can potentially be used to improve detection reliability. A third difference is that in video frames can be easily dropped or edited out, so that it is desirable to be able to detect or extract a watermark from a segment of video without resorting to a brute force search.

We insert a watermark in both the spatial domain and the frequency domain. We shall focus on the spatial domain watermark here. The watermark is embedded in the video on the server. The watermark contains a sequence of signal bits denoted by B_j, where $B_j \in \{-1, 1\}$. Typical signal bit strings might include ownership and/or recipient information.

The scheme encodes a bit string using a spread spectrum technique[3-4]. Decoding for this method is very efficient and is consequently appropriate for video. The basic idea behind the spread spectrum method is to spread each signal bit, B_j, N times resulting in a spread sequence $S_i = B_j$, where $i = [j \cdot N, (j+1) \cdot N)$. Here the factor N is called the chip rate, and the index i refers to the position in the spread sequence of the bit j. The spread sequence S_i is modulated by a pseudo-random number sequence R_i, where R_i can have values of 1 or –1. The amplitude of the modulated spread sequence R_i S_i is A_i. The resulting watermark W_i equals $R_i \cdot S_i \cdot A_i$. The watermarked frame is computed as $I' = I + \ W_,$ where I is the original frame.

The values for the amplitude A_i are determined from the contrast and the intensity of 2x2 blocks of pixels. The contrast is estimated using the absolute value of the difference of the maximum and minimum pixel values of the block, and the intensity is given by the sum of the pixel values. Values for A_i increase with increasing values of block contrast and intensity.

The pseudo-random watermark signal can be shaped by treating each element of the watermark, W_i, as an n x n block, where n is the number of pixels along each side of the block. As the value of n increases, the frequency of the watermark signal decreases. Advantages of larger block sizes include greater robustness to video processing operations such as filtering, lossy compression, and digital-to-analog-to-digital conversion. Disadvantages include lower chip rates and the need for smaller watermark amplitude values A_i in order to avoid visual artifacts. A watermark block size of 4x4 has been found to represent a good compromise between these conflicting factors. Watermark blocks can be tiled in an image, or for greater security, their positions can be randomized.

The signal is decoded using the difference between the watermarked frame to be decoded, I', and a previous frame, I' , and the random sequence R_i that was used by the encoder to watermark frame I'. This approach requires that the random sequence used for watermarking I' be different from the random sequence used for watermarking *I'* . The equation to compute B_j is given by:

$$B_j = sign(\Sigma(I'_i - I'_i) R_i - mean(I'_i - I'_i) \Sigma R_i), \quad (1)$$

where only intensity differences that satisfy the relationship

$$|I'_i - I'_i| < Threshold \quad (2)$$

are used for the calculation of B_j. The purpose of the second term in equation (1) is to correct for the fact that ΣR_i is generally non-zero.

Expansion of the I'_i term helps to illustrate how the decoding equation correctly returns the sign of B_j:

$$B_j = sign(\Sigma (I_i - I'_i) R_i + \Sigma S_i A_i R_i^2 - mean(I'_i - I'_i) \Sigma R_i) \quad (3)$$

The middle term, $\Sigma S_i A_i R_i^2$, has the sign of B_j because S_i has the sign of B_j , and A_i and R_i^2 are always positive. The absolute value of the middle term increases as the number of products increases because the sign of every product is the same. The other two terms are sums of values that are randomly negative or positive. The correct sign is decoded when the chip rate is sufficiently large that the magnitude of the middle term dominates the other two. It is not necessary to use frame differences to decode bit values, but using thresholded frame differences reduces the chip rate necessary for reliable detection.

For fingerprinting, we have to employ fast watermarking techniques, preferably on MPEG video directly. The fingerprinting watermarking algorithm implements a similar algorithm in the compressed domain.

Discussion and Conclusion

A number of further steps can be taken to enhance the system in terms of its performance, security and additional capabilities. An example of the performance of the system when decrypting and descrambling MPEG-2 video in real time on a Pentium® III 400 MHz processor is given in Table 1. The results are based on un-optimized decryption and descrambling code, and significant performance improvements are possible.

Playback mode	Frames/sec
Original video	37
Decryption	36
Visual descrambling	30
Decryption and visual descrambling	28

Table 1. Impact of content protection methods on decoding performance for a 6 Mbits/sec MPEG-2 stream.

System security can be enhanced through continuous authentication as the video plays. The checking can be performed at regular intervals or at random intervals. For video, the authentication is performed from between every few frames to every few seconds. A tamper-resistant software implementation [5] can be incorporated into the client system as an additional layer of software protection to make it more difficult for an attacker to determine the correct keys used for descrambling by hacking into the client playback software.

The client can also be required to authenticate over the network with a secure connection to the server. This can be carried out periodically in the background throughout the viewing of a given piece of content. The server would repeatedly authenticate the client and the authentication results would be needed by the playback system for continued playback of the protected content. The content would fail to play or would not play in full quality if the client were not periodically authenticated with the server. This is particularly applicable to the case of streaming media delivered over the network. However, it also pertains to the media already available on the client's machine.

Many of the methods and concepts in building the video system can be extended to audio also, including encryption, audio scrambling, and watermarking.

References

1. B. G. Haskell, A. Puri, and A. N. Netravali, "Digital video: an introduction to MPEG-2," Chapman and Hall, New York, 1997.
2. National Bureau of Standards, PUB 46, "Data Encryption Standard," National Bureau of Standards, US. Dept of Commerce (1977).
3. F. Hartung and B. Girod, "Digital watermarking of raw and compressed video," Proc. SPIE, **2952,** 205— 213, (1996).
4. W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding", *IBM Syst. J.*, **35**, 3-4, (1996)
5. D. Aucsmith and G. Graunke, "Tamper Resistant Software: An Implementation," Proc. Intel Software Developer's Conference, **2**, (1996).
6. S. Craver, B.-L. Yeo, and M. Yeung, "Technical Trials and Legal Tribulations," *Communications of the ACM* special issue on digital watermarking, **7**, 45— 54, (1998).