An Empirical Study of Secure MPEG Video Transmissions

Iskender Agi and Li Gong

SRI International
Computer Science Laboratory
333 Ravenswood Avenue
Menlo Park, California 94025, U.S.A.
e-mail: *iskender,gong@csl.sri.com*
WWW: http://www.csl.sri.com

Abstract

MPEG (Moving Pictures Expert Group) is an industrial standard for video processing and is widely used in multimedia applications in the Internet. However, no security provision is specified in the standard. We conducted an experimental study of previously proposed selective encryption schemes for MPEG video security. This study showed that these methods are inadequate for sensitive applications. We discuss the tradeoffs between levels of security and computational and compression efficiency.

1 Introduction and Motivation

MPEG (Moving Pictures Expert Group) is an industrial standard for video processing. As network bandwidth becomes more abundant, MPEG and its variant video coding techniques are increasingly used in applications over the Internet, such as video dissemination and desktop conferencing. Awareness of security threats and incidents have prompted researchers to examine secure ways of encrypting MPEG traffic.

No provision for encryption was included for MPEG transmissions. The very nature of MPEG encoding—the encoding of inherently spatially and temporally correlated video—makes the encryption difficult[4]. The MPEG bit streams also contain resynchronization signals, for correcting transmission errors that also reduce the security of the transmission.

MPEG uses an asymmetric coding model where encoding requires substantially more computational power than decoding. Adding security to MPEG transmission usually involves encrypting parts or the whole MPEG bit stream. It has been recognized that commonly available software and hardware encryption mechanisms often cannot encrypt entire MPEG streams without severely degrading performance and quality of service.

In this paper, we examine previously proposed selective encryption schemes for secure MPEG transmission. We find that these selective encryption schemes offers a limited level of security, especially for sensitive applications. For example, certain objects in the scene are clearly visible even if the video sequence is "encrypted". We suggest improvements to such schemes, evaluate their security, and discuss tradeoffs in higher encoding complexity and lower compression rate.

2 MPEG and Its Security

A video sequence consists of a string of time-indexed frames. Each frame is a still-color image of a scene taken at a particular instant in time. The MPEG encoder takes a color video sequence, compresses that sequence using a combination of compression schemes, and produces a serial bit-stream that can be sent over a communication medium. The conversion of the three-dimensional video sequence into a one-dimensional bit stream allows the communication medium to be any of a number of possibilities including satellite, coaxial cable or even CD-ROM. An MPEG decoder receives this bit stream and decodes it to reproduce a facsimile of the original video sequence. We discuss MPEG-1 encoding, but based on our experiments these results also apply to MPEG-2.

MPEG-1 video sequences typically use a frame size of 352×288 pixels displayed at a rate of 24-30 frames per second (FPS). Once encoded, the bit-stream of this typical video sequence is designed to reproduce a VHS-quality video at a transmission rate of 1.5Mb/sec. The MPEG encoder is not limited to these typical values and can encode larger or smaller image sizes and at varying quality settings, which affect the transmission rate accordingly.

MPEG encoding of a video sequence requires many compression steps. Each frame in the video sequence is compressed using a combination of block-based motion compensation to take advantage of interframe temporal redundancy, and discrete cosine transform (DCT)-based compression to take advantage of intraframe spatial redundancy [2]. Each frame (or picture) in the sequence is broken into 8×8 pixel blocks for interframe DCT compression and 16×16 pixel macroblocks for intraframe motion compensation. A 16×16 macroblock also includes two 8×8 chrominance blocks in addition to the four luminance blocks for a total of six blocks per macroblock.

A number of these blocks strung together form a slice (of a frame) that acts as a resynchronization unit. For standard size frames the number of slices per frame is one. Larger frame sizes may use more slices per frame. Then these slices combine to form a picture.

A number of these pictures are grouped together to form a random access unit so that the video can be viewed either forward or backward. This group of pictures (GOP) has little or no dependence on the pictures in adjacent GOPs. A GOP consists of three types of frames: intracoded frames (I), motion-estimated forward predicted frames (P), and motion-estimated bidirectional predicted frames (B) whose frame dependency is shown in Figure 1.

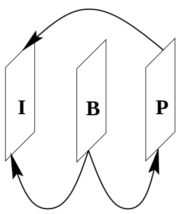

Figure 1: MPEG frame dependency

I-frames are encoded block by block, without regard to previous or future frames. The encoder takes the DCT of each 8×8 block, transforming that block into its frequency-based representation. Since spatial signals have high local autocorrelation, most of the frequency information is located in the lower frequency terms. The encoder eliminates many of the higher-order terms using quantization. The use of lossy compression increases the compression tremendously. The I-frame encoding uses the same scheme as JPEG (Joint Pictures Expert Group) encoding of still frames. This encoded I-frame is used as the motion-estimation reference for the other forward- and bidirectional-predicted frames.

Forward-predicted (P) frames are encoded with reference to the most recent previous I- or P-frame. Each macroblock undergoes motion-estimation search to find the so-called *best-fit* vector relative to the reference frame. This vector is then transmitted along with the error between the macroblock and the motion-compensated macroblock from the previous frame encoded using the DCT. The use of exhaustive search algorithms for motion estimation makes MPEG encoding much more computationally demanding than decoding. Macroblocks with no motion and no error are marked as *skipped* blocks, and no further coding is performed.

Bidirectional-predicted (B) frames are coded with reference to both the immediately previous I or P reference frame and the immediately future I or P reference frame. The motion-estimation and encoding procedure for B-frames is similar to the procedure used to encode the P-frames: it includes the calculation of the best-fit motion vector and the coding of the error terms using the DCT.

The sequence of I-, P- and B-frames defines what is called the *frame pattern* of a GOP. The GOP always begins with an I-frame. Figure 2 shows the frame pattern of a typical MPEG sequence. The frame pattern length is the number of P- and B-frames until the next I-frame. A GOP that ends with a P-frame has no encoding dependence on the frames in the next GOP. However, if a GOP ends in a B-frame, then the encoding of that B-frame requires access to the first frame of the next GOP in the sequence.

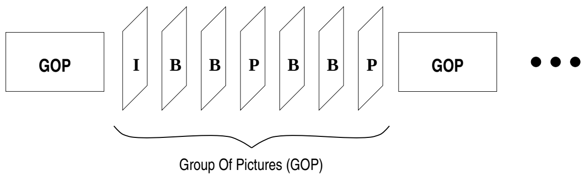

Figure 2: MPEG frame pattern

The compression, using spatial and motion techniques, yields a bit stream that is compressed even further by the encoder using a variable-length (Huffman) coding. It is the combination of the quantization and the run-length encoding that yields the greatest compression in MPEG. This resulting bit stream becomes the MPEG sequence with the addition of the necessary sequence and GOP headers.

2.1 Naive Encryption

The most straight-forward method to achieve secure MPEG transmission would be to encrypt the entire MPEG bit-stream. Once received, the bit-stream would be decrypted and decoded for display. While wholesale encryption is arguably the most secure encryption approach, selective encryption and decryption, exploiting the MPEG structure, seems to offer an encryption solution that reduces the computation considerably without compromising the security of the transmission.

One of these selective encryption schemes uses the fact that the decoder depends on the integrity of the I-frame to decode the P- and B-frames. Encryption of the I-frame alone, in theory, would render the information in the P- and B-frames useless. Encrypting base frames alone allows the encryption to be implemented in software in real-time[6]. Wholesale encryption and decryption of the MPEG bit-stream would require much faster processing and possibly dedicated hardware.

This base-frame encryption scheme does not account for I-blocks (intracoded 8×8 blocks) in the P- and B-frames. Under certain circumstances the security of this scheme falls apart. We performed one experiment where text in the scene undergoes translation. The resulting frame sequence was encoded and the I-frames were encrypted. However, the text was clearly readable when viewing the encrypted stream on a standard MPEG decoder.

2.2 Effects of I-Blocks in P- and B-Frames

The MPEG standard allows encoders to include I-blocks in the P- and B-frames. Encoders can choose to encode certain blocks as I-blocks when the block cannot be adequately motion-compensated. Frame sequences with a large amount of motion often require more I-blocks in motion-compensated frames.

If a frame contains an unencrypted I-block, blocks in subsequent frames using that block as a reference would be correctly decoded until the next encrypted I-frame. The inclusion of the I-blocks in P- and B-frames compromises the security of the transmission. There are two possible solutions to eliminate this problem: force the encoder not to generate I-blocks outside the I-frame, or encrypt all I-blocks. We will explore the latter option in a later section.

2.3 Previous Work on Secure MPEG

There have been some attempts at securing MPEG transmission. For example, Maples and Spanos perform real-time, software encryption using selective encoding of the I-frames. The encryption was performed on a 486-class machine for ATM transmission[5]. In this work, the headers and the I-frames in the MPEG stream are encrypted using DES encryption in the cipher block chaining (CBC) mode[8]. The CBC mode is a commonly used operation mode of DES and has the ability to recover property from bit-errors.

In another example, Meyer and Gadegast have designed a new MPEG-like bit-stream called *SECMPEG* that incorporates selective encryption, additional header information to aid in bit-error recovery, and has high-speed software execution[6]. SECMPEG also allows the use of both DES and RSA [7] encryption algorithms.

SECMPEG implements four levels of security. At the first level, SECMPEG encrypts the headers from the sequence layer down to the slice layer. The motion vectors and the DCT-coded blocks are unaltered. At the second level of security, SECMPEG encrypts parts of the I-blocks (presumably the DC and the lower AC terms of the DCT-encoded I-blocks) in addition to the encryption performed in the first level. At the third level of security, SECMPEG encrypts I-frames and all I-blocks. At the fourth level, SECMPEG performs wholesale encryption.

Since SECMPEG requires major additions and changes to the standard MPEG bit stream headers, a SECMPEG bit stream is not compatible with a standard MPEG bit stream. A special encoder and decoder would be required to view unencrypted SECMPEG bit streams.

3 Weakness in Selective Encryption

As mentioned earlier, one proposed selective encryption algorithm encrypts only the I-frames of MPEG video [5]. This selective method appears attractive because it can significantly reduce the amount of encryption and decryption. The percentage of reduction depends on many factors, such the frequency of I-frames, the particular encoder used, and the actual video sequence. In a typical frame sequence of length 10, the I-frames and I-blocks can take from 30 to 60 % of bit stream, depending on the amount of motion in the scene and the type of encoder.

The security rational behind this proposal is that since I-frames carry the basic information, if they are unavailable to an eavesdropper, then the P- and B-frames become useless even though they are transmitted unencrypted. By restricting the encryption and decryption to the I-frames, this method reduces the cryptographic computation by half. This reduction is significant because the decryption of the I-frames and I-blocks can take as much as one-third of the decoding time[6].

However, a little cryptanalysis (e.g., [3]) would reveal that, although a single P- or B-frame on its own is meaningless without the corresponding I-frame, a series of P- or B-frames carries much information if their base frames are correlated. For example, in a MPEG sequence of an ongoing meeting, the base frames (the I-frames) often contain images of the same people in the same surrounding, which are clearly related. Thus we expect that for such MPEG sequences the simple selective encryption method [5, 6] would not be able to provide a high level of security.

To verify our hypothesis, we conducted a series of experiments. The experiment environment and set up is as follows. The structure of our experimental system is as shown in Figure 3. We wrote the encryption routine that encrypts video output from a MPEG encoder and modified the MPEG encoder code accordingly. More specifically we used the parallel MPEG-1 encoder developed at SRI [1] and the MPEG-2 encoder by Chad Fogg, used DES in cipher-feedback mode to generate a cipher stream, and exclusive-or the cipher stream with the MPEG stream.[1] The routine can optionally encrypt I-frames only, or including all I-blocks also, or all P-frames in addition to all I-blocks.

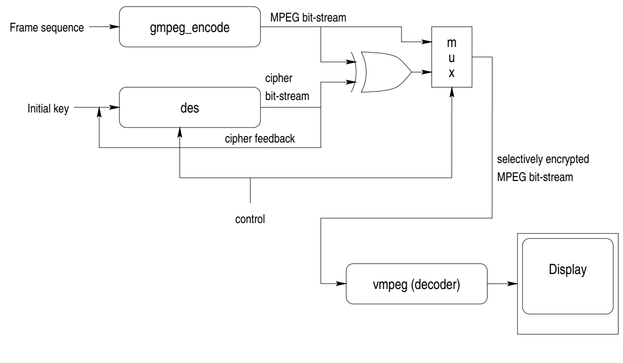

Figure 3: System architecture of experiment

For convenience, this study would need a video decoder to play encrypted video sequences, and we decided to use off-the-shelf products to keep customization at a minimum. This decision brings certain constraints. For example, we cannot easily experiment with the effect of encrypting header information.[2] Eventually, we chose the player for Windows, VMPEG (by Stefan Eckart, version 1.6a "lite", 1995), and run it under MS WFW3.1 on an Intel 486 at 100 MHz. We chose VMPEG because it had the best tolerance to bit-errors of all the decoders that we sampled. It recovers from bit-errors, by making reasonable assumptions on what may have been corrupted. The MPEG sequences were encoded on Sun SparcStations.

We encoded a few different types of sequences, encrypted them while adjusting the parameters to various values. It is impossible to report all the details in this paper, but the two examples we chose are sufficient to illustrate the general conclusions. Table 1 shows the number of bits per frame type in the two example sequences. The absolute numbers can change based on many factors, but the important thing to note is that the percentage of bits in the P- and B-frames increases as the amount of motion in the scene increases.

Table 1: Bits per frame type (percentage of total) with I-frame frequency = 1/10

Frame type	"Miss America"	"Flowers"
I	669186 (57.77%)	3267000 (37.24%)
P	230202 (19.88%)	4855320 (55.34%)
B	258804 (22.35%)	650888 (7.42%)

As one example, we encoded a MPEG video of Miss America sitting behind a desk and speaking to the camera. Figure 4 shows a typical decoded frame from the "Miss America" video sequence. We chose the Miss America sequence because it is a good representation of video conferencing, where a single person is seated in the scene and there is relatively small amount of movement.

We then modified the MPEG encoder to encrypt the I-frames and played the ciphertext directly on a MPEG player. Because of the interframe correlation, we can clearly see that the video sequence contains a person speaking and can also make out some of the features of the person. A revealing frame from the

[1]Strengthening the crypto algorithm and the encryption method does not affect the conclusions of our experiments because we do not exploit weaknesses in the underlying encryption algorithm and method.

[2]It seems unwise to depend security on hiding the header information, because the header information may be available through other means, tends to be guessable, can be exhaustively searched, or may be broken by plaintext-ciphertext pairs.

Figure 4: Miss America I: unencrypted

encrypted sequence is shown in Figure 5.

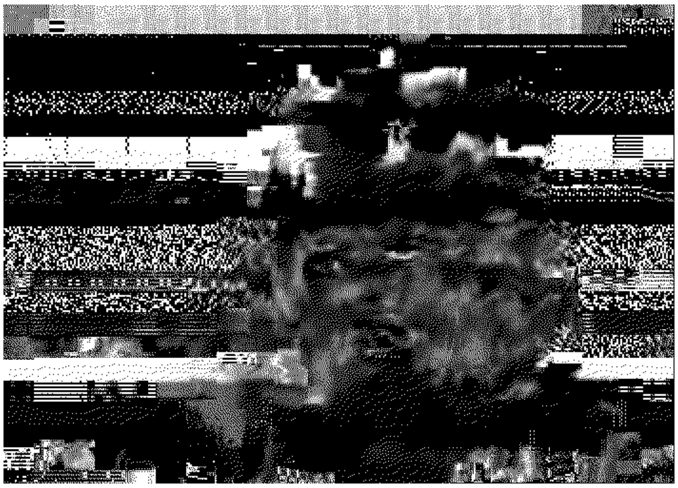

Figure 5: Miss America II: "encrypted"

The lack of security can in fact be much worse under some circumstances, for example, if there is a great deal of motion in the scene. The next video example illustrates how the selective encryption scheme can fall apart. Figure 6 shows a frame from the "flowers" video sequence taken on a moving platform at the bottom of a slope panning across a scene that contains houses and a flower garden. The effect appears to be trees moving towards left against a background of flowers and terrace houses. The "flowers" video contains object in the scene that move in both directions simultaneously.

We encrypted the I-frames in this sequence and played the encrypted stream directly on the VMPEG

Figure 6: Flowers I: unencrypted

player. Figure 7 clearly shows that parts of that frame are effectively in plain view. Great portions of the video are visible partly due to interframe correlation and mainly from unencrypted I-blocks in the P- and B-frames. Therefore, encrypting I-frames only may leave visible I-blocks in other frames.

Figure 7: Flowers II: "encrypted"

These two examples have demonstrated clearly that, contrary to some previous proposals, MPEG video cannot be secured by simply encrypting the encoded I-frames. Of course, security measure is largely a matter of the system requirements. So, simple selective encryption may be sufficient for certain applications, such as cable television broadcast of pay channels, where quality of service supersedes high security as the driver. However, we feel that a wide range of applications would require a much higher level of security for MPEG video. For example, suppose a person in Cupertino, California, is giving a confidential presentation on Apple Computer's finances, which is transmitted to a group of investors in New York City through the Internet using MPEG encoding. Using I-frame selective encryption, it is likely that some portions of the text used in the presentation (e.g., dollar numbers projected overhead) would be visible to an eavesdropper, and this level of security is clearly unacceptable.

4 Tradeoffs in Encrypting MPEG

In this section, we suggest a few methods to improve MPEG security and discuss trade-offs between performance and the level of security.

4.1 Changing I-Frame Frequency

A simple security improvement in selective MPEG encryption is to increase the frequency of the I-frames. The increase in encrypted I-frames would reduce the interframe correlation. At the maximum frequency the entire video sequence is composed only of I-frames resulting in the encryption of the entire sequence. This scheme is obviously quite secure but compression and speed would be much degraded. Selective encryption is intended to add security without reducing the compression enjoyed by using P- and B-frames.

We encrypted a few sequences with increasing I-frame frequencies. By viewing these encrypted videos, we observed that as the I-frame frequency increases the amount of visible information decreases. We also noticed that often the encrypted sequence appears quite secure well before the I-frame frequency reaches maximum. For example, as the frequency of I-frames increases to 1 in every 3 frames in encoding the "Flowers", we found that the most revealing frame looks like the one shown in Figure 8.

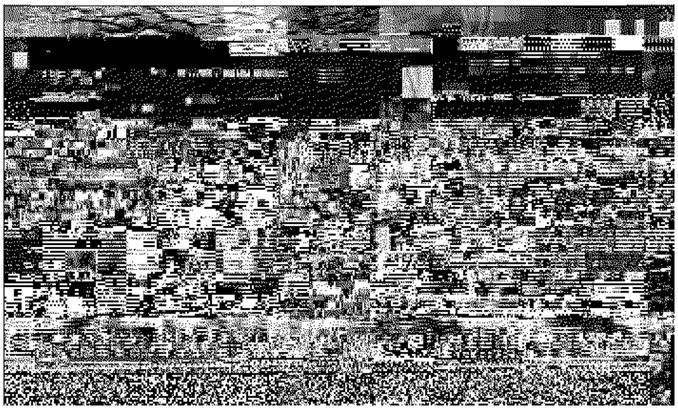

Figure 8: Flowers III: higher I-frame frequency

Here the frame looks random except certain small areas (e.g., tree branches in the upper-left corner and flowers in the lower-right corner) appear unaffected because we did not encrypt the I-blocks.

As expected, when we increased the I-frame frequency to the maximum, the whole sequence appeared random. The case is the same with the "Miss America" example, as shown in Figure 9.

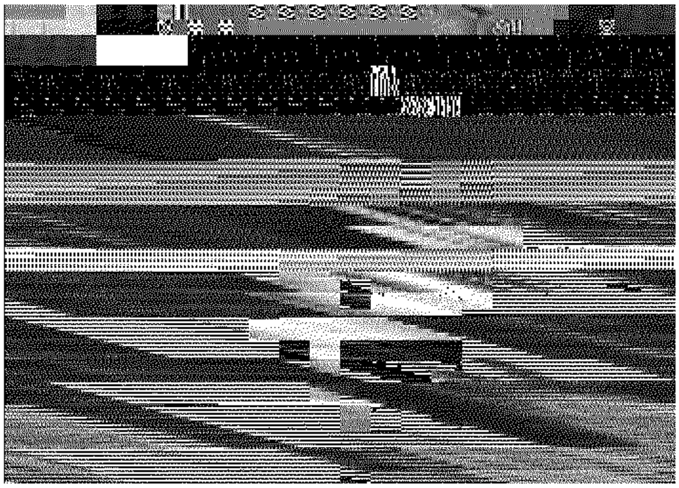

Figure 9: Miss America IV: fully encrypted

Based on our experiments, we feel that it is probably infeasible to come up with a set of parameters that will work reasonably well across a wide range of different types of video that are common in various applications using MPEG. Thus it would be desirable to build into MPEG encryption facility a control mechanism to dynamically change I-frame frequency based on video types. It would also be very useful to conduct further experiments, using extensive video footage, to obtain for various types of video streams reference points (that correspond to different security levels) and performance benchmarks. For example, a low level of encryption may be sufficient for scrambling videos of ice-hockey matches, because of the rapidly changing background and fast moving players. On the other hand, a snooker match (pool, in American terms) may require a different set of parameters because the snooker table is often shot from a few fixed angles and typically only one or two balls would move at one time. These data could guide further secure MPEG development and fine tuning of application-specific MPEG security parameters.

4.2 Encrypting I-Blocks and P-Frames

To further improve security, we experimented with encrypting all of I-blocks in the P- and B-frames in addition to encrypting all I-frames. We found that this increases security significantly, making it possible to reduce the I-frame frequency and improving compression while maintaining the same level of security. Figure 10 shows that by encrypting I-blocks, the following quality of encryption is obtained even though the I-frame frequency is only 1 in every 6.

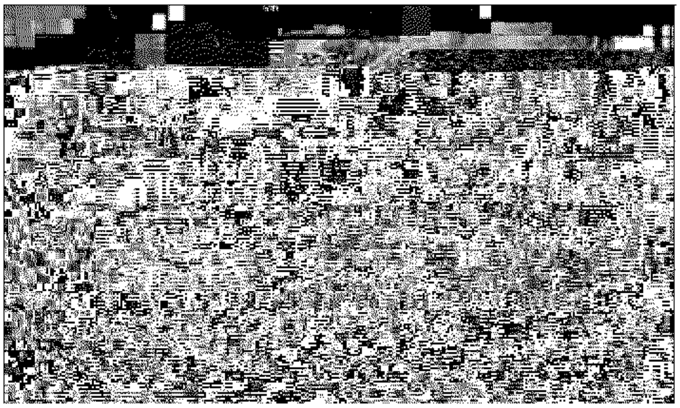

Figure 10: Flowers IV: I-blocks encrypted

Another improvement we experimented was to encrypt the P-frames. Our intuition was that encrypting the P-frames is more economical than increasing the encrypted I-frame frequency because the former does not cause the video size to increase. Both methods would increase the computation load, although the former is less costly because it results in smaller encoded streams.

We encrypted a series of video sequences, with different I-frame frequencies and different encryption options. We found that encrypting the P-frames does not appear to add significant security value, if we already encrypt all I-blocks in all frames. We suspect that this observation might be due to other reasons, including the precision and sensitivity of the observation instruments (in our case, our eyes) and the performance of the video player we used. More powerful cryptanalysis tools may discover improvement in security levels when P-frames are encrypted, and further study in this direction is necessary.

4.3 Trade-Offs

Increasing the I-frame frequency not only means that there are more frames to encrypt but, because of the nature of MPEG, also affects the effectiveness of video compression (thus increasing the total size of the MPEG bit-stream). For example, in the "Flowers" sequence, increasing from 1 in every 12 frames to 1 in every 3 nearly doubles the video size. Table 2 below details the changes in video sizes with different I-frame frequencies.

Table 2: Video sizes (in bytes) at different I-frame frequencies

I-frame frequency	"Miss America" sequence	"Flowers" sequence
1/12	134470	1077688
1/9	156837	1167172
1/6	199582	1362442
1/3	329719	1952627
1/1	859617	4118235

Therefore, there is a clear trade-off between increasing the I-frame frequency (and thus security level) and increasing encryption performance and compression ratio (or in other words, reducing network bandwidth consumption). Our experiments generally confirm the trend illustrated in Figure 11.

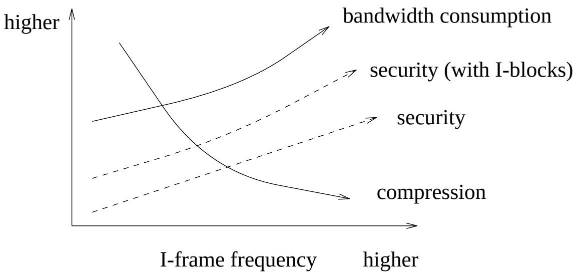

Figure 11: Trade-off between security, performance, and bandwidth consumption

5 Summary and Future Work

MPEG-1 (and MPEG-2) is a popular standard for video processing and is widely used in applications such as desk-top video conferencing [2]. Given currently available hardware and software encryption devices and the demand in MPEG usage, it is commonly accepted that encrypting the full MPEG stream would be too slow for real-time applications, especially in an environment such as the Internet. The selective encryption method [5, 6] where the I-frames (and sometimes I-blocks also) are to be encrypted, has been proposed to be an economical and secure substitute for full encryption.

In this paper we have reported an empirical study of MPEG video encryption. We found that these methods are not adequate for sensitive applications. Specifically, our experiments confirmed our intuition that encrypting I-frames alone may not be sufficiently secure for some types of video. We have given examples where by playing back "encrypted" video sequences on regular MPEG players, one can clearly see patterns of movement and sometimes even large chunks of plaintext video. We conducted experiments with both MPEG-1 and MPEG-2 and obtained similar results.

There are several obvious ways of improvement. First, by increasing the I-frame frequency, better security can be obtained but at the expense of increased video size, a higher network bandwidth consumption, and higher computational complexity for encryption and decryption. Moreover, encrypting the I-blocks in all frames improves security, does not affect the bandwidth consumption, and should be considered an alternative to increasing the I-frame frequency. Additional encryption of the P-frames does not seem to improve security, but warrants further investigation.

There are a number of interesting issues for further study. One is to investigate other ways in which a video stream can be broken into frames or layers, and the applicability and the security of selective encryption in such situations. Another useful effort is to collect data on a wide range of video samples and obtain approximate parameters for effective and efficient encryption. Analytical results will provide deeper insight into the (in)security of selective encryption. Finally, it may be worthwhile to examine the different orders in which compression, error correction coding, and encryption are performed, and see if some arrangements substantially improve security and performance.

References

[1] Iskender Agi and R. Jagannathan. A Portable Fault-Tolerant Parallel Software MPEG-1 Encoder. In B. Furht and M.H. Hamza, editors, *Proceedings of the Second IASTED/ISMM International Conference on Distributed Multimedia Systems and Applications*, Stanford, CA, August 1995.

[2] D. Le Gall. MPEG: A video compression standard for multimedia applications. *Communications of the ACM*, 34(4):46–58, April 1991.

[3] A.G. Konheim. *Cryptography: A Primer.* John Wiley & Sons, Inc., New York, 1981.

[4] Benoit M. Macq and Jean-Jaques Quisquater. Cryptology for Digital TV Broadcasting. *Proceedings of the IEEE*, 83(6):944–957, June 1995.

[5] T.B. Maples and G.A. Spanos. Performance Study of a Selective Encryption Scheme for the Security of Networked, Real-Time Video. In *Proceedings of 4th International Conference on Computer Communications and Networks*, Las Vegas, Nevada, September 1995.

[6] J. Meyer and F. Gadegast. Security Mechanisms for Multimedia-Data with the Example MPEG-I-Video. Project description of SECMPEG, Technical University of Berlin, Germany, May 1995.

[7] R.L. Rivest, A. Shamir, and L. Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*, 21(2):120–126, February 1978.

[8] *Data Encryption Standard.* (U.S.) National Bureau of Standards, January 1977. (U.S.) Federal Information Processing Standards Publication, FIPS PUB 46.