# AN INTEGRATED APPROACH TO ENCRYPTING SCALABLE VIDEO

*Ahmet M. Eskicioglu*

Dept. of Computer and Information Science,
Brooklyn College of the City Univ. of New York,
2900 Bedford Avenue, Brooklyn, NY 11210

*Edward J. Delp*

Video and Image Processing Laboratory (VIPER),
School of Electrical and Computer Engineering,
Purdue Univ., West Lafayette, IN 47907

**ABSTRACT**

Scalable video compression is the encoding of a single video stream in multiple layers, each layer with its own bit rate. Because of the computational complexity of full video encryption, partial encryption has emerged as a general trend for both standard and scalable video codecs. Depending on the application, a particular layer of the video stream is chosen for encryption. In some applications, however, more than one video layer may need to be protected. This results in a more complicated key management as multiple keys are needed. In this paper, we present an integrated approach to encrypting multiple layers. Our proposal is a prepositioned shared secret scheme that enables the reconstruction of different keys by communicating different activating shares for the same prepositioned information. It presents certain advantages over three other key management schemes.

## 1. INTRODUCTION

The recent phenomenal growth in Internet technologies has led to the development of a vast number of multimedia applications. These applications, which combine audio, video and text, include videoconferencing, on-line video games and pay TV. To reduce the huge bandwidth requirements, video compression is used to remove the spatial and temporal redundancy in source sequences. MPEG-I & II, H.261 & 263 are among the most widely used video standards in the delivery of multimedia content. In heterogeneous networks, there are potential uncertainties:

- the channel capacity may be unknown or time-varying when the video is encoded.
- the clients may display characteristics with different computational and communication capabilities.

Such conditions are inherent in wireless or hybrid (wired/wireless) systems [1]. The video source may simultaneously send a full bandwidth stream to fixed clients on a wired network and a lower bandwidth stream to mobile clients on a wireless network. A transcoder at the sender's site is used to generate lower resolution bit streams.

Scalable video compression allows a single video stream to be encoded in different layers, each layer having its own bit rate. The availability of bit streams of multiple quality makes it possible to adapt to a given set of client capabilities and network conditions.

A scalable codec produces a partitioning of the video data into substreams of varying importance. The type of the codec depends on how the partition is made. There are three types of scalability [2]: (1) SNR, (2) spatial, and (3) temporal.

Security is an increasingly important requirement for multimedia applications where the data has to be protected from unauthorized users. Encryption is an essential tool to provide confidentiality in open public networks like the Internet. Because of the computational complexity of full video encryption, a general trend for both standard and scalable video codecs is to use partial encryption.

Several approaches have been proposed for partial encryption regarding video codecs like MPEG-1 or MPEG-2: the protection of I frames or I blocks only [3]; the permutation of DCT coefficients [4]; and the selection of a particular subset of important DCT coefficients for protection [5]. A recent study [6] compares partial encryption results for a scalable video codec and the non-scalable MPEG-1 codec. It is shown that the protection obtained from simple base layer encryption of a scalable encoded video based on a spatial resolution pyramid is comparable to the best known partial MPEG encryption method.

Another approach for protection is to use progressive encryption (such as cipher block chaining or stream cipher) that allows transcoding with simple packet truncation [1]. This eliminates the need to decrypt the video packets at intermediate network nodes with low complexity. A progressive encryption technique is characterized by the property that the first portion of the data is encrypted independently while the later portions are encrypted based on earlier portions. In the proposed architecture, the header data is left unencrypted, and contains information for the transcoding nodes such as the recommended truncation points within the encrypted packets.

Scalable video codecs generate a base layer and several enhancement layers of video. Depending on the application, all the layers or a particular layer may need to be protected. In a Pay-Per-View application, for example, the full-resolution video of a movie trailer may be multicast without protection. When the actual movie starts, the encrypted base layer would provide sufficient degradation in image quality for the customers who have not obtained the viewing authorization. In other applications, the base layer may need to be encrypted together with an enhancement layer. Furthermore, the business model in use may prevent access to all layers unless the customer agrees with certain conditions. Hence, in a general architecture, each layer can be encrypted independently with a different key. The information about the encrypted layers is carried to the clients in the packet header.

In this paper, we present an integrated approach to encrypt multiple layers. Our proposal is a prepositioned shared secret scheme that enables the reconstruction of different keys by communicating different activating shares for the same prepositioned information.

## 2. SCALABLE VIDEO ENCRYPTION SCHEME BASED ON SECRET SHARING

A $(t, n)$ *threshold scheme* $(t \leq n)$ [7] is a method by which $n$ secret shares $S_i, (1 \leq i \leq n)$, are computed from a secret $S$ in such a way that at least $t$ shares are required to reconstruct $S$. In a $(2, 5)$ threshold scheme, the secret is divided into five pieces, and any two of the five pieces can be used to reconstruct the secret. In a *perfect* threshold scheme, a knowledge of $(t-1)$ or fewer shares does not change the probability distribution of the possible values of the secret. Shamir's $(t, n)$ threshold scheme [8] uses a random $(t-1)$-degree polynomial over the finite Galois Field $GF(p)$, i.e., $f(x) = (a_{t-1}x^{t-1} + ... + a_1x + a_0) \; mod \; p$:

1. Choose a prime $p$ larger than $n$ and the secret $S$.
2. Define $S$ to be the constant term $a_0$.
3. Construct $f(x)$ by selecting $(t-1)$ random coefficients $a_1, ..., a_{t-1}$.
4. Compute the shares by evaluate $f(x)$ at $n$ distinct points, and distribute them to $n$ members.

The secret $S$ can be computed by constructing the polynomial from any $t$ of the $n$ shares.

Figure 1 shows three encrypted layers of a scalable video stream. Suppose each layer is encrypted using a different key. We will later discuss key diversification. The following notation is used for the encryption process:

$\{M\}_k$ Message $M$ is encrypted with the symmetric key $k$.

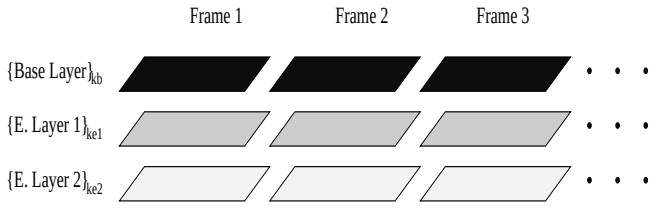


Figure 1: Encryption of multiple layers of a scalable video stream

A traditional approach for encrypting multiple layers is to use three different symmetric keys. A symmetric group key can be established between a sender and the receivers in a multicast architecture in several different ways [9, 10]. Efficiency is achieved with hierarchical key distribution trees. A discussion of the group key management is beyond the scope of our work. Once the group key is established, it is used until a member joins or leaves the multicast group. In the multimedia applications where the data has a high commercial value, the group key ought to change frequently. The frequency selected in some of the conditional access systems today is as few as a couple of seconds [11]. If more than one layer of a scalable video stream needs to be encrypted, we need as many simultaneous group keys as the number of layers.

A recent paper [12] introduces a new approach based on secret sharing in which the group manager (GM) assigns unique secret shares to the nodes in the distribution tree. Called the Centralized Key Management with Secret Sharing (CKMSS), this is a prepositioned shared secret scheme that allows the reconstruction of different keys by communicating different activating shares for the same prepositioned information. For a given node in the tree, the activating share and the shares assigned to the node define a unique polynomial. By a proper assignment of the shares, different key encryption keys are generated for different nodes. A comparison of the scheme with the Wong et al method [13] shows that both computational and communication costs are comparable. A major advantage of using shares instead of keys is that for each new activating share, a new set of keys are generated for the nodes.

Now, we will extend this scheme to the encryption of multiple layers of scalable video. Given the 3 layers in Figure 1, the members of the multicast group need to have three simultaneous group keys for decrypting the 3 layers of video, and these keys will have to be renewed for each join and leave. Furthermore, depending on the application, the frequency of periodic group key change may be quite high. Following our example of a 3-layer video stream, three separate keys would be generated by any centralized key management scheme. Generation of multiple keys in the CKMSS scheme is straightforward. Two alternatives can be considered:

1. Only one activating share is multicast by the GM, and it is used together with the prepositioned information to generate three simultaneous keys. In this case, the shares assigned to the root are used to define three subsets, one subset for each key. As each node can be assigned a different number of shares, such an arrangement is trivial.

2. Three activating shares are multicast by the GM, and they are used together with the prepositioned information to generate three simultaneous keys.

Consider the $k$-ary tree in Figure 2. The value of $k$ is 4*, and the tree has 16 members. For each join/leave and

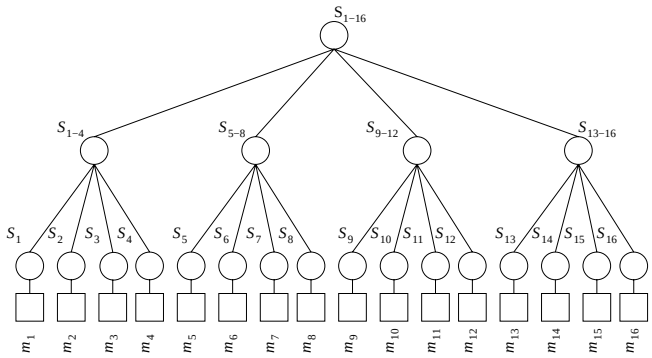


Figure 2: Hierarchical tree for secret sharing

each periodic key change, the shares will change. Choosing Alternative 1 for multiple key generation and using group-oriented strategy [13], we will see the effect of these operations on the tree. A simple partitioning of the set $s_{1-15}$ is to have three disjoint sets. Other definitions are also possible depending on the cardinality of the set.

Leaving the tree

The leaving member will be $m_{16}$. The GM deletes the member node and the set node for the individual set from the key graph, and replaces $s_{13}$-16 at the "leaving point" by $s_{13-15}$ and $s_{1-16}$ by $s_{1-15}$. It then constructs and multicasts the following message to the remaining fifteen members:

$$L_0 : \{s_{1-15}\}_{k_{1-4}}, \{s_{1-15}\}_{k_{5-8}}, \{s_{1-15}\}_{k_{9-12}}, \{s_{1-15}\}_{k_{13-15}}$$
$$L_1 : \{s_{13-15}\}_{k_{13}}, \{s_{13-15}\}_{k_{14}}, \{s_{13-15}\}_{k_{15}}$$
$$GM \rightarrow \{m_1, \cdots, m_{15}\} : AS, L_0, L_1$$

where $AS$ is the activating share, and the fresh keys $k_{1-4}$, $k_{5-8}$, $k_{9-12}$, $k_{13-15}$, $k_{13}$, $k_{14}$ and $k_{15}$ are obtained using the activating share and the sets $s_{1-4}$, $s_{5-8}$, $s_{9-12}$, $s_{13-15}$, $s_{13}$, $s_{14}$ and $s_{15}$, respectively. The members construct the next set of group keys $k'_{1-15}$, $k''_{1-15}$, $k'''_{1-15}$ when the new activating share is multicast with the encrypted content.

Joining the tree

The joining member will be labeled $m_{16}$. The GM establishes $s_{16}$ with the member, creates a new member node and a new set node, and attaches the set node to the existing "joining point." After changing $s_{1-15}$ to $s_{1-16}$ and $s_{13-15}$ to $s_{13-16}$, it constructs and sends the following two messages (The first is multicast to members $1-15$, the second is unicast to member 16):

$$GM \rightarrow \{m_1, \cdots, m_{15}\} : AS, \{s_{1-16}\}_{k_{1-15}}, \{s_{13-16}\}_{k_{13-15}}$$
$$GM \rightarrow m_{16} : AS, \{s_{1-16}, s_{13-16}\}_{k_{16}}$$

where $AS$ is the activating share, and the fresh keys $k_{1-15}$, $k_{13-15}$ and $k_{16}$ are obtained using the activating share and the sets $s_{1-15}$, $s_{13-15}$ and $s_{16}$, respectively. The members construct the next set of group keys $k'_{1-16}$, $k''_{1-16}$, $k'''_{1-16}$ when the new activating share is multicast with the encrypted content.

Periodic key change

For periodic group key change, the group manager constructs and multicasts the following message to the entire group:

$$GM \rightarrow \{m_1, \cdots, m_{16}\} : AS,$$

where $AS$ is the activating share.

The activating share is used by the group members to construct the new set of group keys $k'_{1-16}$, $k''_{1-16}$, $k'''_{1-16}$.

## 3. CONCLUSIONS

We have presented an integrated scheme for encrypting scalable video streams. For a $n$-layer stream, $n$ group keys are needed in a multicast architecture. The CKMSS scheme can be used in a straightforward manner for the simultaneous generation of these $n$ keys. The set of secret shares assigned to the root of the tree can be chosen in such a way that a disjoint subset can be used for the generation of each key. Other advantages of this extension are:

- The keys used in protecting the new secret shares after each join/leave operation are always fresh keys. This is in contrast with the Wong et al scheme where only the compromised keys are replaced.

- For periodic group key change, the only data in the multicast message is the activating share. In comparison, three well-known schemes have certain overheads or weaknesses.

  – For a $k$-ary tree, the Wong et al scheme requires $k$ messages to be multicast.

*Wong et al have found that the optimal key tree degree is around four

  - In Iolus [14], the new key for each subgroup is multicast to the subgroup encrypted with the old subgroup key. This presents a weakness because it sets up a chain of keying material. A compromise in one link of the chain results in a compromise of all the keying material in the remaining part of the chain. The alternative suggested for key change is to wait until a member leaves.
  - In the dual encryption protocol (DEP) [15], two types of keys are defined: the key encryption keys (KEKs) are used to hide data encryption keys (DEKs) from the participants (nodes who are not entitled to the multicast data) and the local subgroup keys are used by subgroup managers (SGMs) to distribute encrypted DEKs to their subgroup members. For periodic rekeying, a SGM signs the new subgroup key and encrypts it with the public keys of all the subgroup members. It then multicasts the protected key to its subgroup members. Refreshing the key encryption keys (KEKs) is a costly procedure, and should be done infrequently.

- The difficulty of finding the key of a node depends on the degree of the polynomial defined for the node, and hence, the number of shares stored as prepositioned information. This parametrization is useful in adjusting the scheme to the security requirements of the applications.

## 4. REFERENCES


[1] S. J. Wee and J. G. Apostolopoulos, "Secure scalable video streaming for wireless networks," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, Salt Lake City, UT, May 2001.

[2] ISO/IEC, *IS 13818-2: generic coding of moving pictures and associated audio information: Video*, 1996.

[3] T. B. Maples and G. A. Spanos, "Performance study of selective encryption scheme for the security of networked real-time video," in *Proceedings of the 4th International Conference on Computer and Communications*, Las Vegas, NV, 1995.

[4] L. Lang, "Methods for encrypting and decrypting mpeg video data efficiently," in *Proceedings of the 4th ACM International Multimedia Conference*, Boston, MA, 1996.

[5] T. Kunkelmann and R. Reineman, "A scalable security architecture, for multimedia communication standards," in *Proceedings of the 4th IEEE International Conference on Multimedia Computing and Systems*, Ottawa, Canada, 1997.

[6] T. Kunkelmann and U. Horn, "Partial video encryption based on scalable coding," in *5th International Workshop on Systems, Signals and Image Processing (IWSSIP'98)*, Zagreb, Croatia, June 1998.

[7] J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.

[8] A. Shamir, "How to share a secret," *CACM*, vol. 22, no. 11, pp. 612–613, November 1979.

[9] M. J. Moyer, J. R. Rao, and P. Rohatgi, "A survey of security issues in multicast communications," *IEEE Network*, pp. 12–23, November/December 1999.

[10] T. Hardjono and G. Tsudik, "IP multicast security: Issues and directions," *Annales de Telecom*, pp. 324–334, July-August 2000.

[11] A. M. Eskicioglu, "A key transport protocol for conditional access systems," *Proceedings of SPIE Security and Watermarking of Multimedia Content III*, vol. 4314, pp. 139–148, January 22-25 2001.

[12] A. M. Eskicioglu and M. R. Eskicioglu, "Multicast security using key graphs and secret sharing," *University of Manitoba Computer Science Department, Technical Report 02-01*, February 2002.

[13] C. K. Wong, M. G. Gouda, and S. S. Lam, "Secure group communications using key graphs," *IEEE/ACM Transactions on Networking*, vol. 8, no. 1, pp. 16–30, Feburary 2000.

[14] S. Mittra, "Iolus: A framework for scalable secure multicasting," *Proceedings of the ACM SIGCOMM '97*, pp. 277–288, September 1997.

[15] L. R. Dondeti, S. Mukherjee, and A. Samal, "A dual encryption protocol for scalable secure multicasting," in *Fourth IEEE Symposium on Computers and Communications*, Red Sea, Egypt, July 6-8 1999.