

# A Study on DRM System for On/Off Line Key Authentication

**Kun-Won, Jang**

School of Information Technology  
Soongsil University  
Seoul, Korea

**Chan-Kil, Park**

School of Information Technology  
Soongsil University  
Seoul, Korea

**Jung-Jae, Kim**

School of Information Technology  
Soongsil University  
Seoul, Korea

**Moon-Seog, Jun**

School of Information Technology  
Soongsil University  
Seoul, Korea

**Abstract** - This paper proposes a Hash Chain algorithm to create a key safer than the existing encryption method, and suggests an encryption method of higher security than the existing system by using the encryption method which applies each key created through the Hash Chain algorithm to each block. Also, after users are authenticated through on/off line key authentication, keys fragmented by using a key fragmentation method are encoded and each fragment is sent to a client. Thus, when this proposed method is used, even when keys are let out, it will be difficult to get the complete set of keys. After the proposed system is designed and realized, an experiment was carried out by using digital contents files of a variety of size for the performance evaluation and it was found that the proposed system was about to do a safer key transmission than the existing system. Here the whole data was encrypted in a way not to be decoded even when key fragments are stolen. Also, in the client system, it was confirmed that when the video and data files are replayed, the times to encrypt and decrypt are similar to those in the existing system.

**Keywords:** DRM, Symmetric Key, Agent, PKI

## 1 Introduction

Due to the generalization of Internet use, changes of digital information have accelerated through cyber space. As the circulation environment for these digital information resources is fast changing, the demand for digital contents such as music, pictures, videos, and publications in digital format has drastically increased. However, one weak point of this transformation to digital contents is that it is possible to reproduce copyrighted digital materials without any damage to their quality, and with this, an issue of protecting digital copyright to prevent illegal reproduction emerges as an important matter. For the protection of copyrighted digital materials, the technology to protect

information is needed in order to ensure stabilization and security, and the Digital Right Management (DRM) technology is needed to generally monitor and trace the flow of digital copyrights and copyrighted materials [2]. Comprehensive measures are promoted to protect copyrights in the cases of the violation of intellectual property for copyrighted digital materials and to manage the distribution process through the DRM technology, and various research is in process to provide a reliable environment for the production, the distribution, and the usage of copyrighted materials [4]. The existing DRM solutions use secret keys as keys used for encryption, so users have to perform encryption when downloading files, which requires a long time to do it. There are some weaknesses in decryption as well. In the case of large-capacity copyrighted materials, the decryption should be performed first for the entire file before the execution, so users cannot play the file in real time; because secret keys used in encryption and decryption are encoded and transmitted only through wire connection, if it gets exposed to other ill-intended users, the protection of the copyrighted materials is no longer ensured [1].

Therefore, this paper suggests a DRM system which uses the transmission method through wire and wireless connections in order to solve the existing DRM system, and proposes a comprehensive DRM system that distributes the decryption key to user's authentication for copyrighted digital materials and the encrypted data themselves and to prevent illegal execution of those materials.

## 2 Relevant research

### 2.1 InterTrust's DRM system

Characteristics of InterTrust's DRM solution are to use password technology and watermarking in order to

protect copyrighted materials and to perform the processes of collecting and recording the information of executions and taking care of bills by implementing the regulations on the usage of copyrighted materials. As copyrighted materials are encrypted prior to the distribution, at the point when users use copyrighted materials at their computer, a license agent confirms the license and sends the payment information so as to conclude a transaction. That is, the transaction can be done by using the payment methods such as credit cards and electronic currency [6, 7, 8]. Also, the copyrighted materials are encrypted and protected, so the Superdistribution of the copyrighted materials to exchange encrypted copyrighted materials among users can be realized [2].

However, the decryption by the InterTrust's DRM system can be reproduced only after the decryption process is finished. Also, since it is encrypted with only one key, when the key information is leaked, it cannot be protected any more; because the whole file is encrypted, it takes more time to encrypt and decrypt files than other systems; and, only after the entire decryption is completed, the file can be executed.

## 2.2 Microsoft's DRM system

The Microsoft's DRM system is the end-to-end DRM system which safely distributes digital media files to providers and customers of copyrighted materials [9]. The key control part is the WMRM (Windows Media Rights Manager), to deliver the media files such as music and videos, which is protected in a file format encrypted on the Internet, to the providers of copyrighted materials. In the WMRM, a pair of keys are allocated to each server and client instances through individualization; and the instances which are assessed to be cracked or to be unsafe are excluded from a pool of service targets with the certificate cancellation list that is distributed by the Microsoft company. A key is included in the license and the license is distributed separately from the copyrighted materials.

However, because the Microsoft's DRM system only provides the company's WMV and WMA file formats, for the encryption, the entire file is encoded so it takes a long time to encrypt a file.

## 2.3 I-Frame DRM system

As shown in Figure 1, in the I-Frame DRM system, an I-Frame of video GOP (Group of Pictures) is encrypted by using a symmetric key to select either AES algorithm or SEED algorithm, and then by using a symmetric key, and then the ID (CID) of appropriate contents and the value of the symmetric key are saved in the server's database [1].

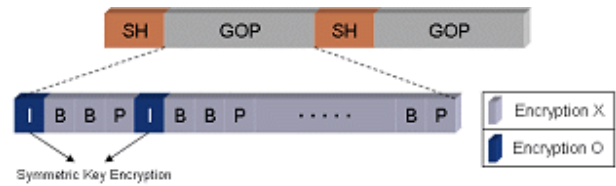


Figure 1. I-Frame DRM system encryption method

When a user execute the encrypted video file, user authentication is performed with the user's certificate, and then the server encodes the key used for the encryption to be the user's public key. After the user acquires the value of symmetric key used for the encryption with his/her individual key, only the I-Frame of video file is decrypted and played, saving the B and P frames together in the buffer.

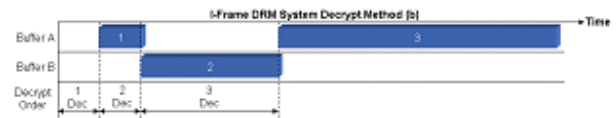


Figure 2. Decryption method of I-Frame DRM system

The I-Frame DRM system in Figure 2 uses a dual buffer algorithm which can execute a file before completing the decryption of the whole video. Since this I-Frame DRM system encrypts only I-Frames from the MPEG (Moving Picture Expert Group) data, it is one of the partial encryption systems. The system's encryption and decryption time has improved more than others, and it allows the file to be executed after decrypting only a part of it, hence the system provides a real-time service.

However, in order to extract I-Frames, the system needs to calculate the size of I-Frame and decrypt it only after reading the contents of all headers of the GOP, so a lot of time is spent to read all GOP headers. The system's weaknesses are: since it only uses one key like the existing systems, once the key information is let out, the encrypted video can no longer be protected, and there is a delay time in decrypting the first block of the file when it is played.

## 3 Encryption / Decryption techniques

### 3.1 InterTrust's DRM system

As shown in Figure 3, before digital contents are encrypted, a preprocessing operation should be performed, in which the raw data are fragmented into blocks so that each block can be encrypted. During this preprocessing operation, the size of the first encryption block should be set as much as the TI (Time Interval) before starting the raw data, and the size of the second block should be 100 ~ 200% of that of the first one. Setting up the block size as 100 ~ 200% of that of the previous one prevent unlimited convergence of the data, and depending on the size of the

raw data, the reasonable number of blocks can be created, which make the execution time of the decryption most stable. In creating groups, one group consists of several blocks, and each group is processed to have a size which is within 12 times of that of the first block. When the group with its size between 12 and 13 times of the first block size was decrypted and executed with dual buffers, no TI was generated. From the second to the last groups, the same method to make each group with several blocks to encrypt and to decrypt with compensatory dual buffers, and it showed the improvement in processing time during encryption and decryption, providing a more stable encryption technique.

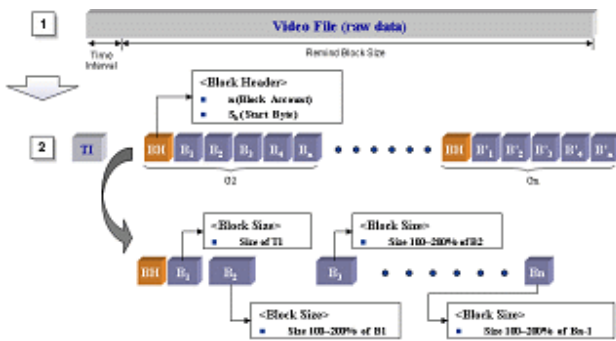


Figure 3. Separate handling of digital content block

When digital content is received, its file size should be confirmed, its arrangement should be set at default, and it should be prepared to be treated by block unit. Before processing a video file, the TI should be checked and its size saved. Then, the size of the first block should be divided into the size of the TI, and the same size is applied during the encryption. After encrypting the first block, it should be checked if there is any remaining data. If there is none, the job to fragment into block units can be concluded. If there is any remaining data, it should be fragmented into blocks by applying a random function to make its size 100 ~ 200 % of the size of the previous block. Making blocks should be repeated until there is no remaining data. During the decryption process using dual buffers, the operation should be repeated, in which a group is made by setting the group size to be 12 times of the TI size, and of the rest of blocks, the second group is made by tying blocks up to 12 times of the size of the last block of the previous group. It should be repeated until there are no remaining blocks and during the decryption process, mutual dual buffers are applied to group unit, which decreases the TI in decryption, preventing any interruption during the data execution. In Table 1, by using the dual buffers in decryption, the execution and the decryption are carried out simultaneously, and the operation can be done without any interruptions.

Table 1. Comparance decryption time with playing time

Interval	Decryption time		Playing time	
	Time (second)	Size (Kbyte)	Time (second)	Image size (Kbyte)
G1	0.1	508	0.1	40
G2	1.238	6287	1.238	508
G3	15.328	77,841	15.328	6,287
G4	189.785	963,752	189.785	77,841
G5	2349.720	11,932,174	2349.720	963,752

The preprocessing process allowed to check the initial file size and, with an arrangement, it was designed to store the size of blocks in processing block units. Also, it allowed to check the TI in executing the contents, and to allocate the next block size by calculating the size of the remaining digital contents. Fragmented blocks were grouped, which was repeated until no blocks were left, and each group was processed.

### 3.2 Designing to create an encryption key using the Hash Chain Method

In order to encrypt the fragmented blocks, two different Hash Functions were used: a key was created with user authentication number in the first Hash Function (H1), and this key was used to encrypt the first block.

By hashing the first key value, the second key (H2) is created, and this key is used to encrypt the second block. The key created from the second key (H2) is sent back to the H1 to create the third key, and this key will be used to encrypt the third block.

This process continues to repeat until no contents are left. Keys are created by using dual Hash Functions and the encryption processes are carried out until all blocks are encrypted. The encryption of digital contents by using two Hash Functions improves the encryption safety. Even when one key attribute is leaked, because Hash Function algorithm is not known, other blocks cannot be decrypted.

In this study, as shown in Figure 4, from the data divided by the fragmentation algorithm, keys are created by using dual hash algorithms, and each block is encrypted by using the created keys. In the encryption, block Header (BH) containing the information on the location address and the size of blocks within a group and the information to control all groups is made up with Container Header (CH) which is composed of LAU (License Acquisition URL) and Contents ID. And, Main Header (MH) has the information on group size and the values of hashed DID.



Figure 4. Each block is encrypted with the value of hash chain

The information on each encrypted block is placed in the BH so that it can have the information on the number of blocks in each group and on the initial bit of each block. Also, the CH is placed to manage the entire data so as to store the contents ID and the information on the starting point of each BH in order to improve the procession time in the encryption.

The MH will be stored in the server to store each group size of contents and the device ID, which are provided to an authorized user. Even when the contents information is leaked, the authorized user will be provided with the MH information, which ensures safety.

### 3.3 Design of user authentication and key transmission method

The overall system outline is that after confirming the user's authentication, the server provides the User Authentication Number (UAN) in order to identify the user and to prevent the information leakage. The user inputs the UAN as a key value and requests the decryption key through wire connection. The agent which confirms the UAN creates a decryption key with the One Time Password (OTP), and provides the key to the user through an algorithm that safely delivers the created key.

After the created key is divided into two keys (Keys\_1 and Keys\_2) by using the key fragmentation algorithm, Keys\_1 and Keys\_2 will be each encrypted by using an agent, and be transmitted to the user through the transmission process as shown in Figure 5.

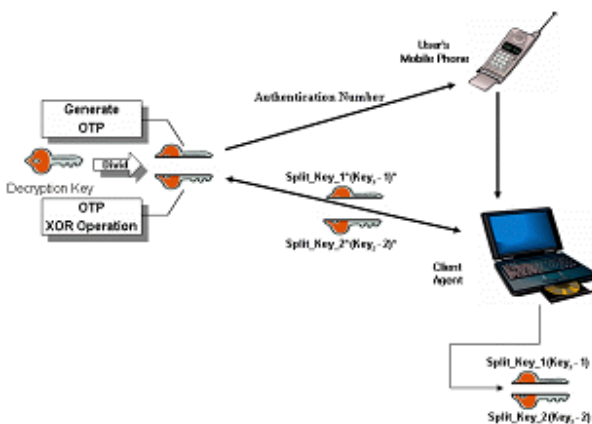


Figure 5. Key transmission method

Key Fragmentation Algorithm is (OTP(Ka) is Keys\_1 and Temp(Kb) is Keys\_2) :

$$Key \oplus OTP = Temp \quad (1)$$

For safe key transmission, the key transmission protocol, with which the user is confirmed through the user authentication process, and a key to be transmitted to the user is created by using the OTP and transmitted, is proposed as the following (Figure 6).

Keys\_1 is set as Ka and Keys\_2 as Kb to be provided in two occasions.

First, the UAN (ex. 101023) is created at the server and is provided to the user through a mobile phone via the SSL channel to ensure the safety of key transmission.

Second, with the UAN, the decryption key can be requested. The first-time user can request the decryption key with the UAN, and then the server create a key with the key fragmentation algorithm (Formula 1) and safely send the key with the key transmission protocol.

Third, the server transmits the value of encrypted Ka, which uses the sum of the increment value in session ( $\Delta i$ ) and the UAN as the encryption key, to a client.

Fourth, the user decrypts Ka and gets a random number, r. Then, the Ka value and the r value are concatenated, hashed, and sent back to the server.

Lastly, at the server, the sum of the UAN and a random number, r, generated from the client, is encrypted again and Kb is transmitted.

$$K_a = \text{Keys}_1$$

$$K_b = \text{Keys}_2$$

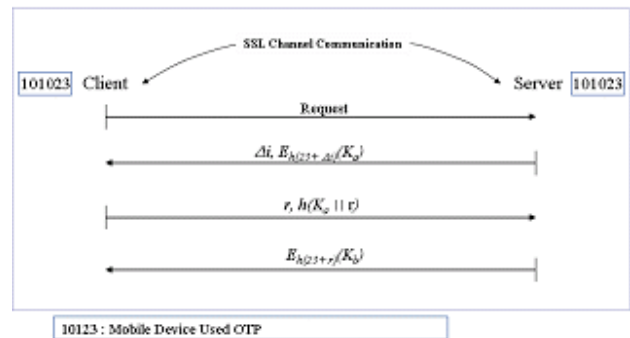


Figure 6. Key transmission protocol

A new user and an old user should be distinguished so that the old user is not asked for a key again. For this, it is designed that a session value, i, which was previously

used, is stored and confirmed at the server to allow the user to continue to use the program. The session value is represented by  $i$ , and  $\Delta i$  is defined as the increment value of a session,  $i^*$  as the previous session value, and  $i$  as the current session value. For old users, session increment values are checked to allow them to continue to use the service. But, in the case of new users, the increment value is zero, so they will be provided with the key value.

### 3.4 Decryption process

The process to decrypt digital contents entails to check the existence of a license same as the Content ID of the Container. If there is no license, it is moved to the LAU (License Acquisition URL) of the Container Header to acquire a license and the user's DID (Device ID: Mac Address) is saved as a hash value.

As in Figure 7, when requesting to play a video, the first-time user should be granted a license for decryption. In acquiring a key, a decryption key required for the user authentication process and decryption is received. With a Content ID, the MH of the appropriate digital contents is requested; and the user's hashed DID value and the MH are sent to the MH as the user's public key. After decrypting the MH with an individual key, it is checked if the hashed values at the user's computer and at the MH are same, and then the file is played.

The MH is encrypted with the user's public key (PU) and is decrypted with the user's individual key. After acquiring the location of the BH, a key is acquired with the user authentication method, and  $G_k (B_1 \sim B_n)$  is decrypted with the BH content as the key.

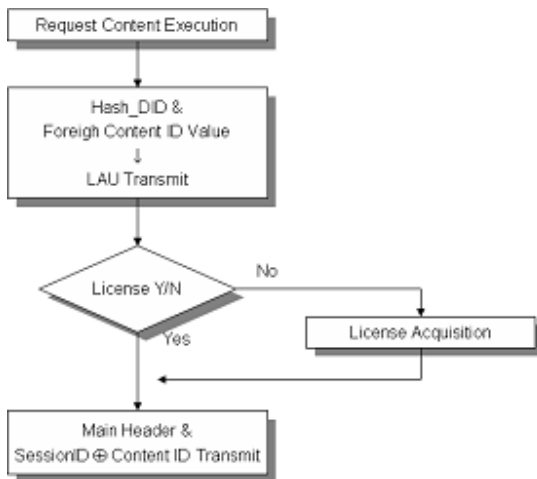


Figure 7. Preprocessing for a decryption

## 4 Experiment evaluation

When the encrypted digital content is played, the decryption time is analyzed and its play time is compared. Compared with the delay time of the existing decryption algorithm, that of the proposed algorithm is analyzed and shown in Figure 8.

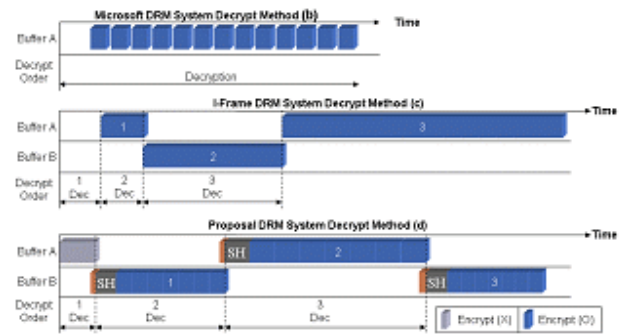


Figure 8. Comparison the delay time of our system with conventional system

The Microsoft's DRM system decrypts the encrypted data as a whole and then plays it, so there is a weakness of a long processing time. The decryption of I-Frame system that extracts only I frames from I, B, and P frames composed of a video and encrypts them has a less processing delay time for it uses a dual buffer after partial decryption. Here the proposed encryption algorithm decrypts and plays a part of the file during the time interval before starting to play it; and, the next block is decrypted while the previous ones are played, so it is designed not to have any delay time or interruptions.

In the DRM system, files are saved at the user's computer in the encrypted state. If it is saved without being encrypted, a user may illegally duplicate or leak the copyrighted materials, so it should be stored encrypted. Therefore, when the user executes a copyrighted material, the decryption can be performed by the user agent. However, when it is a large-sized file, it takes a long time to decrypt the file so the user has to wait long.

The differences between the existing DRM system method and the proposed DRM system method are as followed (Figure 8).

1) In the case of the existing method, the decryption is performed only if a user executes the file. That is, when the user executes the copyrighted material, an agent contacts a server to certify the validity of the license for the said material, and when the user is legitimate, the decryption is carried out and the said material is executed.

2) The existing general decryption system method executes the digital contents after the whole content is completely decrypted, so a user has to wait long until the decryption is finished. Especially in the case of large-capacity digital content file, the decryption takes a long time and a real-time service cannot be provided.

3) The conventional I-Frame decryption system also uses dual buffer algorithm as this paper proposes, but because all frames of digital contents is encrypted, it has a weakness that an execution cannot be started right away.

## 5 Conclusion

This paper proposed a mutual authentication protocol and the encryption method designed to protect digital contents by using wire and wireless connections.

The DRM system is a management technology to allow copyrighted digital materials to be used and distributed in a reliable environment and to protect the intellectual property of the materials. In other words, it is a technology to consistently protect and manage the rights and profits of copyright holders by protecting digital contents from unauthorized users.

However, as the existing DRM system uses the secret key password algorithm as an encryption algorithm, the encryption cannot be performed in advance. So, the encryption is performed when a user downloads a data file with digital contents, so it takes a long time to download a file. Also, when the secret key is leaked by the user, a serious issue arises for the safe of the copyright cannot be guaranteed. Therefore, in order to overcome these disadvantages, for the existing DRM system, a public key password algorithm is used, or research to mix secret key and public key algorithms for the encryption is ongoing. But, an impact on encryption and decryption time is significant and little satisfactory outcome is produced. An agent is used in encryption, decryption, and copyright management to prevent the leakage of key information by a user, but it was pointed out that there are a lot of limitations in its function and process in off-line situations.

This paper proposes a method to encrypt the entire data by using several secret keys as security agents in order to prevent the leakage of secret key by a user vis-a-vis the user authentication of digital contents. So, even if one secret key is let out, the decryption of the entire copyrighted material is not possible. Also, since it is encrypted in advance, the proposed method improves the transmission speed so that files can be downloaded and played immediately.

After the proposed system is designed and materialized, an experiment is carried out with the video files of diverse sizes for performance evaluation. It was

confirmed that, compared to the existing systems, the proposed system can drastically reduce a delay time including the decryption time for large-capacity digital contents when a video file is played at a client system.

## 6 References

- [1] J. Kim, J. Park, and M. Jun, "DRM system based on public key pool for the security of movie data," *Korea Information Processing Society journal*, Vol. 12-C, No. 02, pp. 183-190, April, 2005.
- [2] Brad Cox, *Superdistribution : Objects As Property on the Electronic Frontier*, Addison-Wesley, May 1996.
- [3] Sung, J Park, "Copyrights Protection Techniques," Proceedings International Digital Content Conference, Seoul Korea, November 28-29, 2000.
- [4] V.K Gupta, "Technological Measures of Protection," Proceedings of International Conference on WIPO, Seoul Korea, October 25-27, 2000.
- [5] Whitfield Diffie and Martin Hellman, "New Directions in Cryptography," *IEEE Transaction on Information Theory*, Vol. IT-22, No.6, pp. 644-654, November 1976.
- [6] Intertrust: <http://www.intertrust.com/main/overview/drm.html>
- [7] Joshua Duhl and Susan Kevorkian, "Understanding DRM system: An IDC White paper," IDC, 2000.
- [8] Joshua Duhl, "Digital Rights Management: A Definition," IDC 2001.
- [9] Microsoft: <http://www.microsoft.com/windows/windowsmedia/drm.asp>