The Design of a DRM System using PKI and a Licensing Agent

KeunWang Lee[1], JaePyo Park[2], KwangHyoung Lee[2],JongHee Lee[2] and HeeSook kim[3]

[1]Dept.of Science Chungwoon University, Korea
kwlee@cwunet.ac.kr
[2]School of Computing, Soong-sil University, Korea
[3]Asan Information Polytechnic college, Korea

Abstract. As the logistic environment of digital contents is rapidly changing, the protection of digital rights for digital content has been recognized as a very critical issue that needs to be dealt with effectively. Digital Rights Management (DRM) has taken much interest in Internet Service Providers (ISPs), authors and publishers of digital content in order to create a trusted environment for access and use of digital resources. In this paper, PKI (Public Key Infrastructure) and a licensing agent are used in order to prevent illegal use of digital contents by unauthorized users. In addition, a DRM system is proposed and designed which performs proprietary encryption and real-time decoding using the I-frame-under-container method to protect copyright of video data.

1 Introduction

As the distribution environment for digital resources undergoes rapid changes resulting from the proliferation of the Internet and increased interconnection among computers, the demand for multimedia content such as music, images, movies, publications etc. in digital form is rapidly increasing. Since such digital content can be duplicated without deterioration in quality, the protection of digital copyrights for preventing such unauthorized duplication is emerging as an important issue. For content protection and management, information protection, technology for providing stability and security and digital copyright management DRM (Digital Rights Management) technology for management of copyrights and the monitoring/tracing of overall distribution of contents is necessary [1, 2]. DRM can be defined as a management technology which continuously protects and manages the rights and interests of copyright holders [3, 4]. A comprehensive measure for protecting copyrights from attempted copyright infringement against digital contents is being pursued by utilizing DRM technology, and various researches are being carried out to create a trusted environment within which creation, distribution and use of copyrighted media are being performed [5, 6]. Several companies such as InterTrust, ContentGuard etc. are offering various types of DRM solutions. However, in existing DRM technology, static copyright management is performed by inserting protection conditions, authoring management, etc., into the contents; therefore, due to limitations in

monitoring and tracing functionality, not only is dynamic control of copyright difficult to achieve, but there is also difficulty in obtaining proof of illegal conduct should copyright infringement (such as illegal copying) occur. As such, a digital copyright management technology which is applicable in all types of online and offline content and enables dynamic copyright management, as well as real-time monitoring and tracing, must be developed [7].
In this paper, an integrated DRM system is proposed and designed which provides user certification for multimedia contents in online and offline conditions by using PKI and a licensing agent, and prevents illegal use by unauthorized users through encryption of the data itself.

2 Related Works

Existing DRM technology does not take privacy protection into consideration since the protection of user's privacy is not directly necessary for copyright protection. Due to this, user information leaked during the process of user certification for issuing licenses reported usage details for monitoring illegal usage of contents; therefore, problems related to user privacy infringements occurred [8]. Microsoft's WMRM (Windows Media Rights Manager) is an end-to-end DRM system which distributes digital media files to content providers and consumers in a secure manner [9, 10]. WMRM distributes media, such as music or video, to content providers through the Internet in a protected form through the encryption of files. In WMRM, each server or client instance receives a key pair through the individualization process; instances which are considered cracked or unsafe are excluded from service through the certificate cancellation list. WMRM is widely used in incorporated form with the Windows Media Player; however, it only supports a limited number of file formats since its flexibility in dynamic environments is limited–it only supports Windows Media Player. In addition, one disadvantage of the WMRM is since its user certification process for issuing licenses does not use any specific protection mechanism, user information such as user IDs or e-mail addresses are leaked.

3 System Architecture

Data protection of original content and authentication should not be implemented by simple access right control on existing content or password-based authentication but by user authentication using PKI and through inserting related information into original content using data encryption. The proposed system has a client/server architecture and its overall layout is illustrated in Fig. 1.

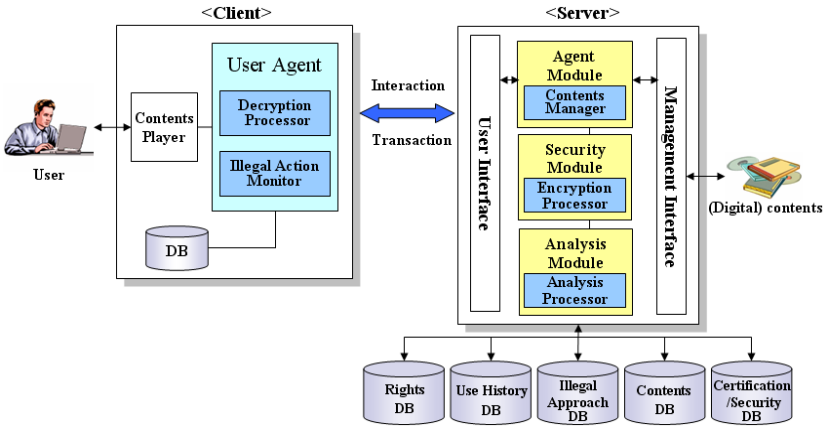

Fig 1. System Architecture

When a content is registered on the server using an external interface, processing for content monitoring is performed by an agent module and encryption is performed on the content. In order to use the content, user authentication is performed by the licensing agent sent from the server: for authorized users, the content is executed by the application program, and for unauthorized users, a warning message is output. Real-time monitoring against illegal activities is performed on the content by the licensing agent, and all illegal activities of all users are stored on the server's database through a monitoring interface.

4 Authentication and Encipherment Mechanism

4.1 Encryption and Decryption of Video Data

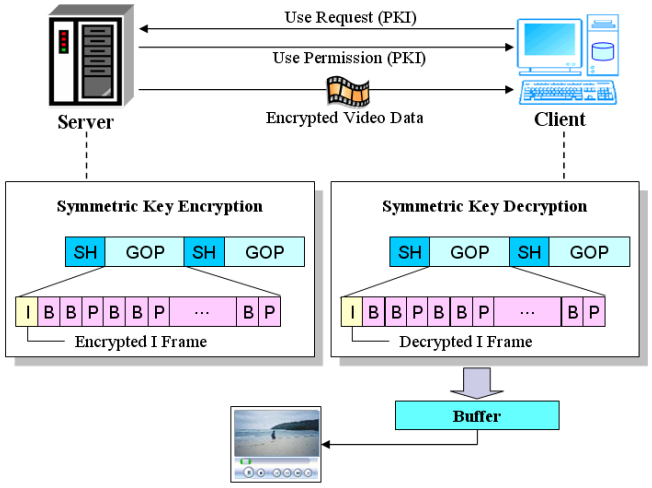

Fig 2. Encryption Decryption Processing

When the server which receives a request for use of content performs user authentication, the symmetric key needed for decoding is encrypted using the user's public key then sent to the client so that the video data's I frame which is encrypted can be decrypted for the client for playback. Figure 2 illustrates the encryption and decryption process of the video data which is the content used in this system.
The video data stored on the server is generated by extracting the I frame of each video content then applying encryption using a symmetric key. Any user can download the server's video data; however, the data cannot be used without proper authorization since the I frame is encrypted. The symmetric key algorithm is used since it can minimize the time required for encryption and decryption. If the downloaded video data is to be played back to the client, the user issues a use request to the server, which then performs authentication depending on whether the user is an authorized one. In this authentication process, a PKI algorithm is used. The symmetric key of the video data requested is encrypted using the user's public key and then sent to the client.
The client's agent decrypts the symmetric keys using the user's private keys, then extracts the I frame of the video data to be played back by using these symmetric keys, then stores it on the buffer together with the B and P frames to perform playback. While all the video data is being played back, the delayed frames should be calculated to determine the initial buffer size.

4.2 License Certification Method

The author of the content sends the created content to a content publisher. Then, the content publisher encrypts (E) the content using an arbitrary symmetrical key Ks to generate the encrypted content C, which is then sent to the content provider to be stored on the content provider's server.

$$C = EKs[data]$$

The user can download a desired content from the content provider's server. However, the user cannot execute the downloaded content arbitrarily because it is encrypted.

Step 1. User registration protocol

The user has to register first in order to use the content. The user registration process is shown in Fig. 3.
The user connects to a system server which functions as a license server and sends the user's certificate cert_u. The system server verifies the user's certificate cert_u through its specified certification path; if the certificate is correct, it sends the user's agent.

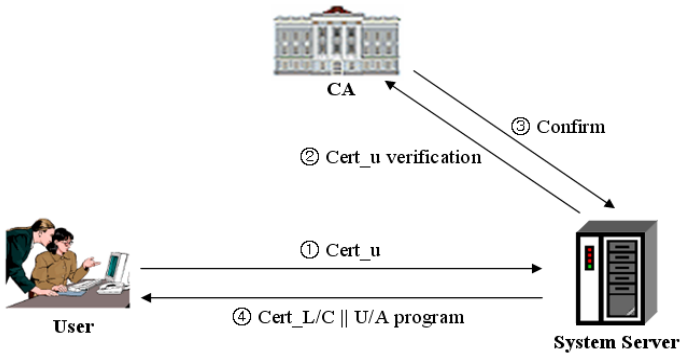

Fig 3. User Registration Protocol

Step 2. License issuing protocol

The user installs a license agent (LA) program which is then executed. When the user executes an encrypted content, the license agent installed on the user's PC connects to the system server and obtains a license, as shown in Fig. 4.

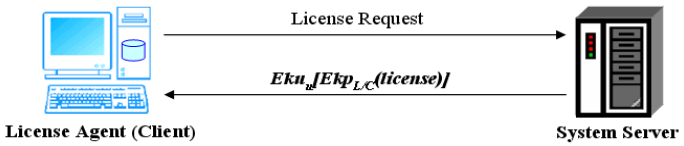

Fig 4. License Issuance Protocol

The license agent connects to the system server and requests a license for the desired content. The system server issues a license including the license ID, user ID, content ID and privileges. Here, the license is encrypted using the user's public key for security reasons (as shown below), and signed using the user's private key and then transmitted.

$$EKuu[EkpL/C(license)]$$

Here, ku is the public key while kp is the private key. Therefore, kuu is the user's public key and kpL/C is the private key of the license clearing house (L/C).

Step 3. License certification protocol

When the user executes an encrypted content, the license agent checks whether there is a license present. If there is no license present, a license is requested according to Step 2 above; if there is a license present, the authentication for that license is requested to the license clearing house which resides on the system

server, as illustrated in Fig. 5 below.

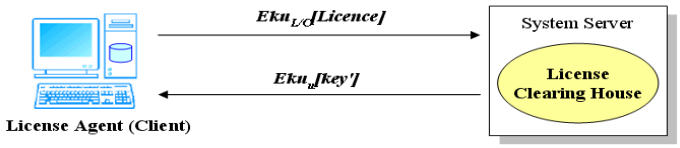

Fig 5. License certification Protocol

When the license clearing house receives an authentication request for the license from the license agent, it checks the privileges from the license information list and then performs authentication. If the user's license is valid for the time up to a specific date, it checks whether that time has expired; if it is a license for the allowed number of usage, the number is decremented by 1. Then, after performing operations on the key value using the user's ID (as shown below), encryption is done using the user's public key then the key is sent to the user.

$$Ekuu[key'] \qquad (where \;\; key' = ks \oplus user\_ID)$$

The user agent, which has received the encrypted key, decrypts that key by using the user's private key to extract key, which is then calculated with the user_ID in the user's license to get the key needed for decrypting the encrypted content, which is then shown to the user.

5 Performance Evaluation

To evaluate the performance of the system proposed in this paper, we implemented a prototype version of the system using Visual C++ 6.0 and MS-SQL 2000. The encryption time for the video data and the initial playback delay time due to decoding time were both measured at a PC that have Intel(R) P-IV, CPU 2.4GHz and 512M RAM. The result of comparison between the conventional method of playing back an already encrypted video data file after decoding (non-realtime decoding method), and the method of playing back while decoding in real time, which is proposed in this paper, is shown in Fig. 6. To measure time accurately, a total of 30 video data files were segmented in minute units for the comparison. As the test result shows, in the conventional method of playback after decoding the entire video data file, the larger the file size, the longer the initial delay time for playback; whereas in the proposed method, it has been shown that the initial delay time has been reduced significantly.
The result shows that, in the proposed method, the delay time until the start time of video data playback (including decoding time) is much shorter than that of the conventional method. In addition, even with real-time decoding, stable playback was demonstrated without interruption of playback or noise.

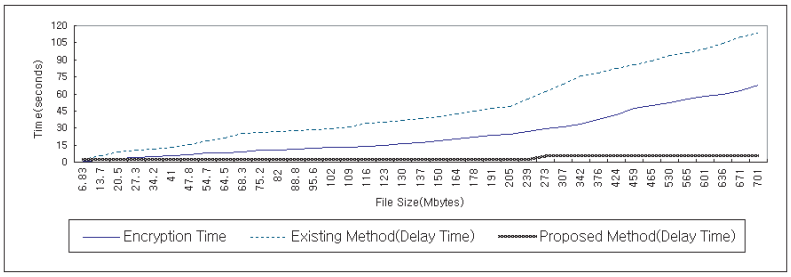

Fig 6. Encryption Time and Delay Time Comparison

6 Conclusion

In this paper, a DRM system for digital copyright protection using a licensing agent which is based on PKI has been proposed and designed. The licensing agent performs user authentication using PKI methodology, performs encryption of the data itself at the system server using container methodology, and decoding is performed in real time for the client under copyright protection of multimedia data. As a follow-up, complete implementation of the proposed system and a safety evaluation of the user authentication process are necessary.

References

1. James Cannady, Jay Harrell, "A Comparative Analysis of Current Intrusion Detection Technologies," http://iw.gtri.gatech.edu/Papers/ids_rev.html, Feb., 1998.
2. Jai Sundar B., Spafford E., "Software Agents for Intrusion Detection," Technical Report, Department of Computer Science, Purdue University, 1997.
3. J.Dubl,"Digital Rights Management: A Defination", IDC 2001.
4. J.Dubl, S.Kevorkian, "Understanding DRM system: An IDC White paper", IDC, 2001.
5. Kentaro Endo, The Building up of national Regional and International Registers for works and objects of related rights," Proc. of International Conference on WIPO, Seoul, Korea October 25-27, 2000.
6. V. K Gupta, "Technological measures of protection," Proc. of International Conference on WIPO, Seoul, Korea October 28-29, 2000.
7. P. Vora, D, Reynolds, L. Dickinson, J. Erickson, D. Banks, "Privacy and Digital Rights Managements", A Position paper for the W3C Workshop on Digital Rights Management, January 2001.
8. D. K. Mulligan and A. Burstein, Implementing Copyright Limitations in Rights Expression Languages, in 2002 ACM Workshop on Digital Rights Management, Washington DC, November 18 2002.
9. J. S. Erickson, Fair use, DRM, and trusted computing, Communications of the ACM, vol. 46, no. 4, pp. 34-39, April 2003.
10. Microsoft's press releases of the PocketPC 2002 launch, Oct 8, 2001. Available at ww.microsoft.com/presspass/events/pocketpc2002/default.asp.