시큐리티 에이전트를 이용한 사용자 인증과 DRM 보안시스템 설계

김 정 재[†] · 이 경 석[††] · 전 문 석[†††]

요 약

본 논문에서는 기존의 암호화 방법보다 다양한 키를 생성하는 알고리즘을 제안하고, 키 생성 알고리즘을 통해 각각 생성된 대칭키를 서버에 저장하지 않는 기존의 시스템보다 보안성이 높은 DRM 암호화 시스템을 제안한다. 또한 클라이언트에서 복호화 할 때 각각의 키를 유추하여 복호화 하는 클라이언트 시큐리티 에이전트 시스템을 제안한다. 제안한 시스템을 설계하고 구현한 후 성능 평가를 위해 다양한 크기의 비디오 데이터 파일을 이용하여 실험을 수행하여 제안한 시스템이 기존 시스템에 비해 비디오 데이터 파일 재생 시 암호화 복호화 시간을 포함한 지연시간을 줄어 든 것을 검증하였다.

키워드 : 디지털 저작권 관리, 대칭키, 시큐리티 에이전트, PKI, 인증

Design of User Authentication and DRM Security System Using Security Agent

Jung-Jae Kim[†] · Kyung-Seog Lee[††] · Moon-Seog Jun[†††]

ABSTRACT

This paper proposes the more various key generation algorithms than existing method and the DRM encryption system supporting the higher security than the existing systems which do not store a symmetric key made by the key generation algorithm in a server. Also, we propose a client security agent system which decrypts a data by analogized key. We designed and implemented the proposed system. And, we tested the video data files with the various sizes to evaluate the performance of our system. Our experiment results show that the delay time which includes an encryption and decryption time was significantly reduced through our proposed scheme.

Key Words : DRM, Symmetric Key, Security Agent, PKI, Authentication

1. 서 론

인터넷의 확산과 컴퓨터 상호연결성의 증대로 디지털 자원에 대한 유통 환경이 급속히 변화함에 따라 디지털 형태의 음악, 화상, 영상물, 출판물 등 멀티미디어 자료에 대한 수요가 급격히 증가하고 있다. 디지털 저작물은 품질의 손상 없이 복제가 가능하기 때문에 디지털 저작권 관리(DRM : Digital Rights Management) 기술이 필요하다. 이러한 DRM 기술을 이용하여 InterTrust사와 Microsoft사 등의 외국 업체와 Digicap와 같은 국내 여러 업체가 다양한 형태의 DRM 솔루션을 제공하고 있다[3].

하지만 기존 DRM 솔루션들은 암호화에 사용하는 키로 비밀키를 사용하여 사용자가 파일을 다운로드할 때 암호화를 수행하므로 많은 시간이 소요되며, 복호화 과정에서도 대용량의 저작물인 경우 전체 파일에 대하여 복호화를 먼저 수행한 후에 실행을 할 수 있으므로 사용자가 실시간으로 파일을 플레이해서 볼 수 없는 문제점이 있었다. 또한 암호화와 복호화에 사용하는 키가 사용자에 의하여 노출이 된다면 해당 저작물에 대한 보호는 더 이상 보장하지 못하는 단점이 있다.

본 연구에서는 기존의 DRM 보안시스템이 가지는 이러한 문제를 해결하기 위해 여러 개의 대칭키를 사용하여 암호화하는 기법을 제안한다. 즉, 하나의 대칭키가 노출 되더러도 저작물 전체에 대한 복호화를 할 수 없는 방법과 암호화 및 복호화 속도를 개선하기 위하여 동영상 전체가 아닌 부분적으로 암호화하는 방법을 제안한다. 또한 동영상 재생 시 대용량의 복호화를 수행하기 위해서 많은 시간이 필요하므로 원활한 동영상의 재생을 위하여 효율적인 보상 이중 버퍼

† 준 회 원 : 숭실대학교 컴퓨터학과 공학박사
†† 종신회원 : 산업연구원 연구위원
††† 종신회원 : 숭실대학교 컴퓨터학과 부교수
논문접수 : 2005년 7월 19일, 심사완료 : 2005년 10월 12일

스케줄링을 사용하여 사용자에게 실시간, 복호화 및 재생을 할 수 있도록 시스템을 제안하고 실험평가를 통해서 암호화 및 복호화 속도의 우수성을 검증하도록 한다.

2. 관련 연구

2.1 DRM 연구 현황

디지털 저작물은 품질의 손상 없이 복제가 가능하기 때문에 불법복제로부터 저작자를 보호하기 위해 안전한 디지털 저작권 보호시스템의 개발이 필요하며, 이를 보완하기 위하여 허가되지 않은 사용자로부터 디지털 저작물을 안전하게 보호함으로써 저작권자의 권리 및 이익을 지속적으로 보호하는 다양한 연구가 진행 중에 있다. 그중에서도 저작물의 기밀성과 무결성을 확보를 위하여 암호기술을 중심으로 많이 발전하여 왔으며 저작권에 대한 내용을 명시하기 위하여 XrML(eXtensible rights Markup Language)을 기반으로 표준화가 진전되고 있으며 식별자 부여를 위해서는 DOI(Digital Object Identifier)를 적극 활용해 나가는 추세이다[5].

2.2 기존의 DRM 시스템

2.2.1 InterTrust의 DRM 시스템

InterTrust사의 DRM 솔루션 특징은 저작물의 보호를 위해서 암호기술과 워터마킹을 사용하며 저작물 사용규칙을 지정하여 사용내역의 수집 및 기록, 과금 처리를 수행하는 것이다. 저작물은 사전에 암호화되어 배포되므로 사용자의 컴퓨터에서 저작물을 사용하는 시점에서 라이선스 에이전트가 라이선스를 확인하고 지불정보를 전송하여 거래를 체결하도록 하였다. 그러므로 신용카드나 전자 화폐 등의 결제 방식을 이용하여 거래할 수 있다[6, 7, 8]. 또한 저작물이 암호화되어 보호되고 있으므로 사용자들 사이에 암호화된 저작물을 주고받을 수 있는 저작물 재분배(SuperDistribution)를 실현하였다[2].

하지만 InterTrust사의 DRM 시스템의 복호화는 (그림 1)과 같이 복호화가 끝난 후에 재생이 가능하다.

(그림 1) InterTrust사의 DRM 시스템

또한 1개의 키로만 암호화하기 때문에 키가 유출이 될 경우 더 이상 보호를 받지 못한다는 점과 파일 전체를 암호화하기 때문에 암호화/복호화 하는데 시간이 다른 시스템보다 오래 걸리는 점과 재생시 전체 복호화가 끝난 후에야 재생이 되는 단점이 있다.

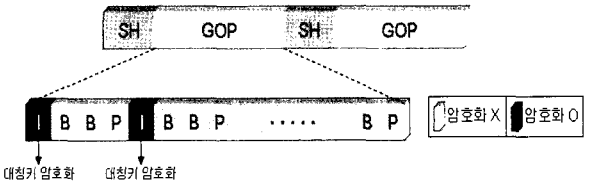
2.2.2 Microsoft의 DRM 시스템

Microsoft의 DRM 시스템은 저작물 제공자와 소비자들에게 디지털 미디어 파일을 안전하게 분배하는 종단 간(end-to-end) DRM 시스템이다[9]. 핵심 제어 부분은 WMRM (Windows Media Rights Manager)으로서 저작물 제공자에게 인터넷상에서 암호화된 파일 형식으로 보호된 음악, 비디오 등의 미디어를 배달한다. WMRM에서 각각의 서버 또는 클라이언트 인스턴스들은 개인화(indivisualization)과정을 통해 키 쌍을 할당받게 되며, 크래킹 되었거나 안전하지 않다고 판단되는 인스턴스에 대해서는 인증서 취소목록을 이용하여 서비스 대상에서 제외시키게 된다. 인증서 취소목록은 마이크로소프트사의 웹사이트를 통해 배포된다. 키는 라이선스에 포함되고, 라이선스와 저작물은 분리되어 분배된다.

하지만 Microsoft사의 DRM 시스템의 경우는 자사의 WMV와 WMA의 파일 포맷만을 지원하기 때문에 암호화시 파일 전체를 인코딩하여 암호화하기 때문에 시간이 오래 걸린다.

2.2.3 I-Frame DRM 시스템

I-Frame DRM 시스템은 (그림 2)와 같이 동영상 GOP (Group Of Picture)의 I-Frame을 대칭키를 이용하여 AES 알고리즘이나 SEED 알고리즘 중에서 하나를 선택하여 암호화한 후 해당 콘텐츠의 ID(CID)와 대칭키의 값을 서버의 데이터베이스에 저장한다[1].

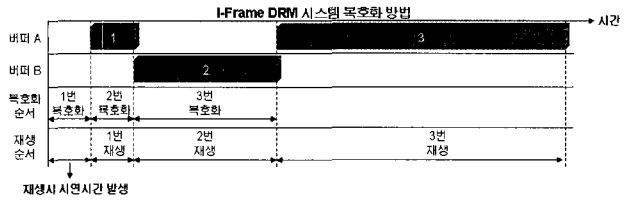


(그림 2) I-Frame DRM 시스템의 암호화 방법

사용자가 암호화된 동영상을 실행시키면 사용자의 인증서를 이용하여 사용자 인증을 수행한 후 서버는 암호화에 사용된 키 값을 사용자의 공개키로 암호화 시키고, 사용자는 개인키를 사용하여 암호화에 사용된 대칭키 값을 획득한 다음, 동영상의 I-Frame만을 다시 복호화 시켜 B, P 프레임과 함께 버퍼에 저장하여 플레이 한다.

I-Frame DRM 시스템은 (그림 3)과 같이 전체 동영상의 복호화가 끝나기 전에 해당 파일을 재생할 수 있는 이중 버퍼 알고리즘을 사용한다. 이 I-Frame DRM 시스템은 MPEG (Moving Picture Expert Group)데이터에서 I-Frame만을 암호화하기 때문에 부분 암호화 시스템에 속하며 이는 암호화 및 복호화 속도가 기존의 다른 시스템보다 향상된 시스템이며 일부분만 복호화 한 후 재생하는 방법으로 실시간적인 서비스를 많이 제공해 주는 시스템이다.

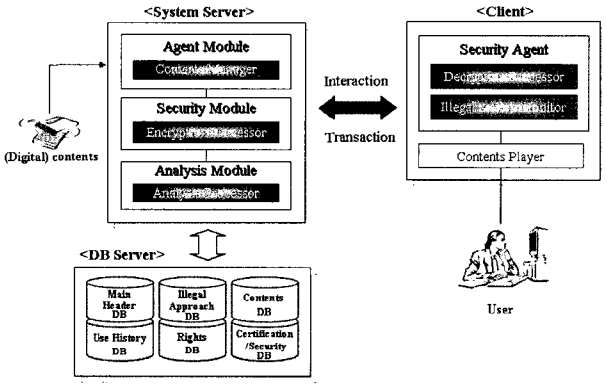
하지만 I-Frame을 추출하기 위하여 GOP(Group of Picture) 그룹의 모든 헤더의 내용을 읽은 다음 I-Frame의 크기를 계산하여 복호화 하는 시스템이기 때문에 모든 GOP 헤더를 읽는데 시간이 많이 소비된다. 기존의 시스템과 같이 한 개의 키 만을 사용하기 때문에 키가 유출이 되면 더 이상 암호화 된 동영상은 보호를 받지 못한다는 단점과, 재생 시 처음 블록을 복호화 하는데 걸리는 재생 지연시간이 발생한다.



(그림 3) I-Frame DRM 시스템의 이중 버퍼를 사용한 복호화 방법

3. 제안 시스템 구조

제안하는 시스템은 (그림 4)와 같이 클라이언트/서버 구조로 구성되어 운용되고, 서버는 에이전트 모듈과 암호화 모듈, 분석 모듈과 데이터베이스로 구성되며 클라이언트는 복호화 처리기와 저작물 실행기로 구성된 시큐리티 에이전트가 있다.



(그림 4) 제안하는 개선된 DRM 시스템 구성도

3.1 서버의 에이전트 모듈

서버의 에이전트 모듈은 CP(Content Provider)에 의해 등록된 컨텐츠를 서버의 암호화 모듈로 전송시켜주는 역할과 에이전트 모듈에서 컨텐츠 관리자는 CP에서 들어오는 컨텐츠의 ID를 만들어서 컨텐츠 데이터베이스에 등록을 하는 역할을 담당하고 있으며, 클라이언트의 시큐리티 모듈과 SSL (Secure Socket Layer)로 접속하여 세션값을 유지하면서, 클라이언트의 시큐리티 모듈에서 들어오는 모든 값들을 통계 분석 모듈과 직접 통신하여 처리 및 관리하는 모듈이다.

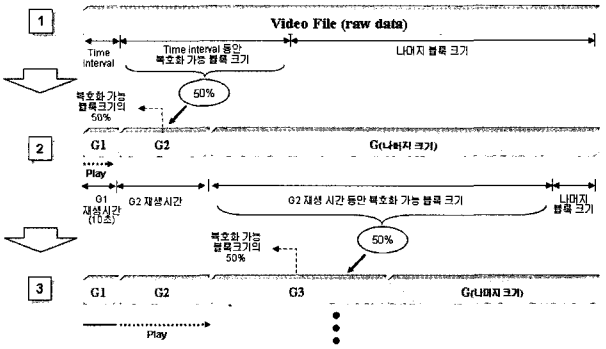
3.2 서버의 암호화 모듈

CP가 등록한 컨텐츠를 에이전트 모듈에서 받으면, 서버의 시큐리티 에이전트는 전처리 단계인 슬라이스 레이어로 나주어 주는 작업을 수행한다.

슬라이스 레이어 작업은 서버에서 받은 해당 저작물에 대한 시간과 화면 사이즈를 획득한 후, 타임 인터벌 값(약 10초)에 해당되는 동영상 파일의 크기를 먼저 계산한 후, 이 타임 인터벌 부분이 재생됨과 동시에 다음 블록이 복호화 될 수 있는 사이즈 크기의 50%~95% 양을 구하게 된다.

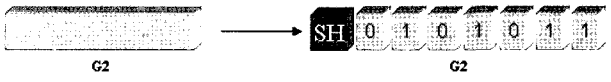
이유는 복호화만 시킬 때 100%라고 가정을 하고, 이 과정은 이전 블록이 재생됨과 동시에 복호화를 해야 하기 때문에 비중을 CPU 사용률을 고려하여 최대 복호화 양의 일부분으로 정하게 된다. 만약 이 CPU 성능이 우수한 컴퓨터일수록 복호화 양의 비율을 높일 수 있으며 반대로 CPU 성능이 낮을수록 복호화 비율을 낮출 수가 있다.

이러한 방법으로 다음 동영상 데이터가 플레이 되는 동안, 해당 동영상 파일의 복호화 할 수 있는 동영상의 사이즈를 구하는 작업을 반복하게 된다. 이러한 방법으로 해당 동영상의 그룹을 n개로 나누어 저장을 하게 된다. (그림 5)는 G1, G2, G3, G4와 같이 4개로 분할된 슬라이스 레이어 그룹을 볼 수가 있다.



(그림 5) 제안하는 동영상의 슬라이스 레이어

슬라이스 레이어 작업을 거친 후 암호화 작업으로 넘어간다. 이 암호화 작업시 타임 인터벌에 해당되는 G1 블록은 암호화를 하지 않고, 다음 G2 블록부터 암호화를 시킨다. 동영상 실행 시 G1 블록은 암호화가 되어 있지 않기 때문에 바로 재생을 할 수가 있으며, 이때 G2 블록을 동시에 복호화 시킬 수 있기 때문이다. G2의 슬라이스 레이어 블록부터 암호화 작업을 수행하는데, Gn의 각각의 슬라이스 레이어에 랜덤수 5 ~ 15까지 발행한 후, 해당 수만큼 그 동영상을 다시 나누게 된다. 만약 해당 랜덤수가 7이 나왔을 경우, 동일한 사이즈로 나누어 주며, 암호화 시킬 블록과 그렇지 않은 블록을 나누어 주어야 한다.



(그림 6) 슬라이스 레이어의 암호화 블록 매핑

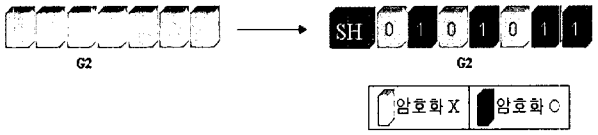
암호화 조건은 연속적으로 암호화를 시키지 않는 블록은 없어야 하며, 전체 암호화된 블록은 50% 이상이어야 한다. 그리고 암호화 할 슬라이스 레이어 블록 부분을 (그림 6)과 같이 1로 매핑을 시키고, 암호화 하지 않을 슬라이스 레이어 부분을 0으로 매핑 시킨다.

G2 슬라이스 레이어 헤더에는 랜덤수(n), 즉 블록의 개수와 암호화 시킬 블록(0101011), 세분화된 블록 시작 바이트 (S_b)를 담고 있으며, 슬라이스 헤더(SH)는 CID 값으로 다시 암호화 시켜 열어볼 수 없도록 만들어 둔다. 다음 슬라이

스 레이어에서 나누어 놓은 부분 슬라이스 레이어 중 1로 매핑된 블록을 암호화 하는데, 암호화 키는 (식 1)과 같이 생성한다.

$$KEY = H(CID \| S_b \| n \| EB) \quad \text{(식 1)}$$

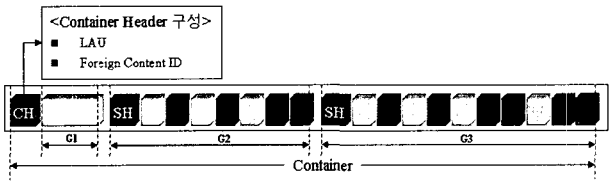
헤더 정보(SH)와 CID로 해쉬한 값을 부분 슬라이스 레이어의 1로 매핑된 부분만을 (식 1)의 키를 이용한 대칭키 암호 방법으로 암호화를 시키며, 헤더 정보 역시 CID 값으로 암호화를 시킨다(그림 7).



(그림 7) 암호화 한 슬라이스 레이어

여기서 해쉬 함수(H)는 128Bit MD5를 사용하였고, 암호화는 AES 암호화 방법을 사용한다.

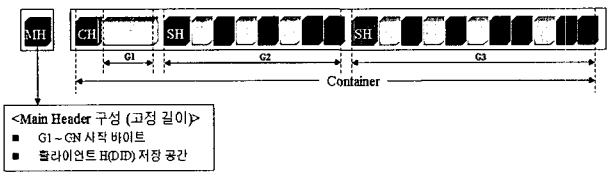
이와 같은 슬라이스 블록 암호화 방법으로 각 슬라이스 레이어의 난수로 발생된 1로 매핑된 블록을 암호화 하는 방법으로 암호화 한 다음, (그림 8)과 같이 슬라이스 레이어의 모든 블록을 하나로 합치는 과정을 거치게 된다.



(그림 8) 전체 동영상의 Container와 Container 헤더

암호화 과정을 거친 슬라이스 레이어를 묶어 '컨테이너'라고 불리우며, 컨테이너 헤더(CH)는 LAU(License Acquisition URL)와 컨텐츠의 2차 ID (Foreign Content ID)로 구성이 되어 있으며 이 컨테이너는 웹사이트를 통해서 사용자가 받아갈 수가 있다.

LAU에는 라이선스를 획득할 수 있는 URL이 들어가 있으며, 암호화된 컨텐츠를 재생시킬 때 해당 컨텐츠의 2차 ID의 값으로 해당 LAU로 가서 라이선스가 있는지 확인을 하고, 만약 라이선스가 없다면 라이선스를 받을 수 있는 웹페이지 URL로 이동하기 위하여 넣어둔 값이며, 컨테이너 헤더는 암호화를 시키지 않는다.



(그림 9) 전체 동영상과 메인 헤더

클라이언트에서 LAU를 통하여 라이선스를 획득한 다음 동영상을 복호화 하기 위해서는 각각의 슬라이어 레이어가 몇 바이트인지 알아야 하므로, 메인 헤더(MH)를 따로 구성을 해야 하는데, (그림 9)와 같이 메인 헤더(MH)는 클라이언트의 DID를 해쉬한 값을 저장할 공간과 각각의 $G_1 \sim G_n$의 시작 바이트를 기록해 놓은 파일이다.

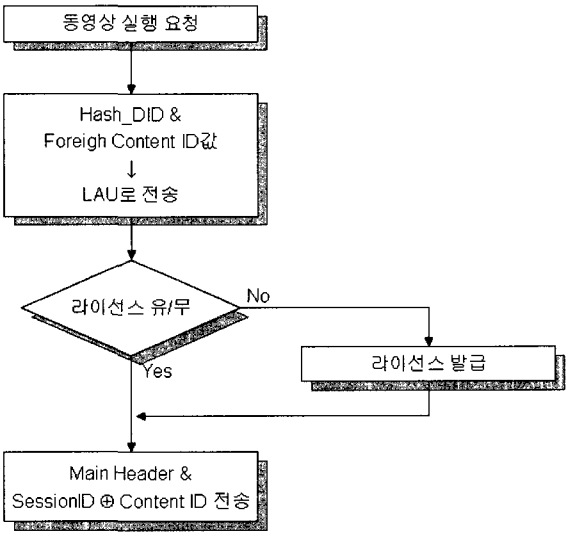
3.3 서버의 통계 분석 모듈

서버의 통계 분석 모듈은 에이전트 모듈로부터 받은 클라이언트의 모든 행위에 대한 정보를 데이터베이스 시스템과 연동하여 정보를 얻어내고, 분석하는 모듈이다. 또한 암호화 모듈에서 생성된 메인 헤더 데이터베이스를 관리하며, 클라이언트 사용자의 라이선스도 직접 관리하는 역할도 담당하고 있다.

3.4 클라이언트의 시큐리티 에이전트

클라이언트의 암호화 에이전트는 암호화된 컨텐츠를 복호화 하기 위하여 무조건 설치하여야 하며, 클라이언트 사용자가 서버에 처음으로 접속하였을 때 서버로부터 다운로드하여 설치된다.

동영상 컨테이너를 다운로드 받은 클라이언트에서 컨테이너를 실행시키면 CH에 있는 LAU를 통하여 라이선스를 확인한 후에 클라이언트의 DID의 해쉬 값을 보내주면, 서버의 에이전트에서 해당 DID 해쉬 값을 해당 컨테이너의 MH에 포함하여 사용자의 공개키로 암호화 하여 클라이언트로 전송해 준다. 그다음 클라이언트 에이전트는 사용자의 인증서를 바탕으로 컨텐츠의 메인 헤더파일 복호화 작업을 수행한다. 클라이언트에서는 자신의 개인키로 암호화된 헤더를 복호화 하더라도 슬라이스 레이어의 헤더와 각각의 슬라이스 레이어의 난수 블록은 해당 컨텐츠의 CID를 알 수가 없기 때문에 복호화 시킬 수가 없다. MH를 복호화 한 후, CID 값을 얻기 위해 서버에서 SSL을 통해 클라이언트 세션 ID값

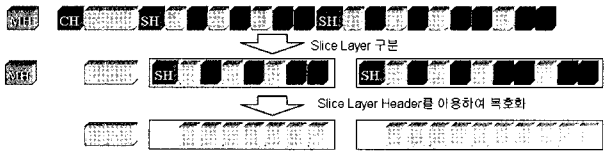


(그림 10) 복호화 전처리 작업

과 컨텐츠 헤더 파일을 XOR한 값, 즉 Temp ID(임시 ID)값을 전송해 주게 되며, 클라이언트에서는 Temp ID값을 다시 Session ID 와 XOR 시켜 CID값을 추출하게 된다.

사용자의 개인키로 복호화된 MH를 통해서 클라이언트의 DID값과 비교를 한 후, DID값이 맞지 않는다면 복호화 작업을 중단하고, 새로운 MH를 부여 받도록 한다(그림 10).

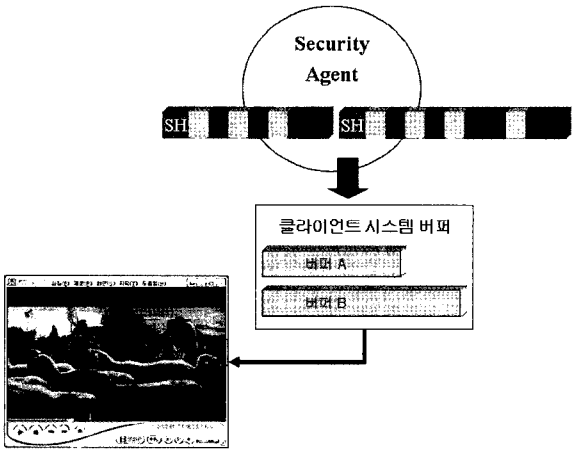
MH와 CID를 전부 획득하였다면, MH에 있는 $G_1 \sim G_n$의 사이즈를 획득한 후, 합쳐진 컨테이너를 슬라이스 레이어로 나눈 후, CID로 슬라이스 헤더 파일(SH)을 복호화 한 후, 슬라이스 헤더(SH)를 이용하여 각각의 슬라이스 조각 레이어의 키를 생성하여 암호화된 블록을 복호화 시킨다(그림 11).



(그림 11) 제안하는 시스템의 복호화 과정

기존의 시스템은 전체 동영상의 복호화가 끝난 후 실행을 하거나, 복호화가 10초 분량이 됐을 때 재생을 시키는 방법도 있지만, 이러한 방법은 일단 재생을 위해 일정시간을 기다려야 하는 문제점이 있다. 그러나 제안하는 방법은 재생과 동시에 컨텐츠 헤더를 복호화만 시키면 재생 지연시간 없이 바로 해당 컨텐츠를 실행할 수가 있다.

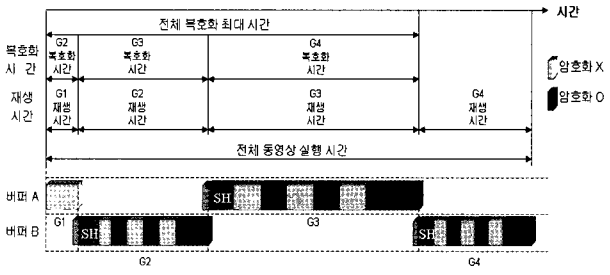
클라이언트에 위치한 시큐리티 에이전트는 (그림 12)와 같이 복호화를 수행하기 위하여 암호화된 동영상의 슬라이스 레이어를 추출하여 비밀키로 복호화를 수행한 후 버퍼 A와 버퍼 B에 번갈아 가며 저장하여 재생한다.



(그림 12) 보상 이중버퍼를 사용한 동영상 복호화 과정

초기 재생 시간을 확보하기 위하여 첫 번째 슬라이스 레이어 G1은 암호화 하지 않았기 때문에 G1은 바로 재생이 가능하며, G1을 재생할 때 슬라이스 레이어 G2의 복호화 과정이 동시에 수행이 된다. 이러한 방법으로 G2가 재생이 될 때 슬라이스 레이어 G3에 대한 복호화 과정이 반복되는 방법이다.

버퍼에는 전체 동영상이 플레이 되는 동안 지연되는 프레임을 계산하여 초기에 버퍼 사이즈를 결정한 후 플레이 하도록 하며, (그림 13)과 같이 2개의 버퍼를 사용하는 보상 이중버퍼 시스템을 사용한다.



(그림 13) 동영상 실행을 위한 보상 이중버퍼 시스템

원활한 재생을 위하여 초기에는 G1(타임 인터벌값 : 약 10초 분량) 슬라이스 레이어를 재생하기 위해 버퍼 A에 저장하여 실행되어 지면, 슬라이스 레이어 G2 분량의 데이터를 복호화하여 버퍼 B에 저장한다. 버퍼 A에서 재생이 끝나면 에이전트는 버퍼 B의 데이터가 실행될 수 있도록 버퍼 B로 옮겨준다. 버퍼 A에서 버퍼 B로 바뀔때 화면의 끊김현상이 발생하는데 G2, G3, G4의 첫 프레임이 임의의 수로 나누어진 완전치 않은 프레임이기 때문이다. 이를 방지하기 위해서 G1, G2, G3의 마지막 프레임의 값을 버퍼 B에 붙여서 완전한 프레임으로 바꿔주기 위한 작업이 필요하다.

시큐리티 에이전트에 있는 불법행위 감시기는 사용자의 정보와 실행하고자 하는 동영상의 정보를 서버로 보내게 된다. 사용자의 불법적인 행위는 감시 인터페이스를 통해 서버의 데이터베이스에 저장된다. 인증된 사용자라 할지라도 사용권한에 따라 제한적인 사용을 위해 저작물 자체 암호화에 의해 저작물을 보호하게 된다. 사용자가 저작물에 대한 라이선스를 초과하여 사용하려고 시도하거나 사용자 임의의 복호화를 시도하는 등의 불법적인 사용 행위를 시도할 경우 이를 봉쇄하도록 저작물에 대한 지속적인 모니터링을 수행한다. 사용자가 저작물에 대해 불법적인 사용을 몇 회나 시도 했는지, 또한 사용권한 범위가 어느 정도인지 등 저작권 위배사례 수집 및 분석을 통하여 사용자에 대한 블랙리스트 관리와 각종 통계 정보를 계산하여 그 정보를 갱신 및 유지하는 작업을 수행한다.

4. 실험 평가

서버에서는 클라이언트 측의 저작물의 사용실태를 파악할 수 있는 정보를 서버측 화면에 출력할 수 있는 인터페이스를 제공하며, 암호화 과정을 수행할 때 동영상에 대한 각종 정보를 볼 수 있도록 하였다. 클라이언트 측에서는 해당 서버로 접속을 한 다음, 암호화된 콘텐츠를 리스트를 확인한 후 다운로드 받을 수 있도록 구성하였고, 복호화 과정과 함께 콘텐츠를 재생 할 수 있도록 구성하였다.

4.1 기존 시스템과의 비교 분석

DRM시스템에서 가장 많은 시간을 소요하는 것은 저작물에 대한 암호화와 복호화 시간이므로 암호화 및 복호화 방식에 대한 시간과 시스템 사양별로 인한 재생시간을 분석한다.

4.1.1 암호화에 대한 비교 분석

암호화에 대한 비교분석은 <표 1>과 같이 암호화 기법과 키의 노출 가능성, 동영상 자체의 암호화 방식과 적용파일에 대하여 비교해볼 수 있다. 암호화 방법은 대칭키 암호화 방식을 사용한다.

〈표 1〉 기존 DRM 시스템과의 암호화에 대한 분석

비교 항목	기존의 DRM 시스템	제안하는 개선된 DRM 시스템
암호화 기법	한 개의 대칭키	복수개의 대칭키
키의 노출 가능성	높음	낮음
동영상 자체 암호화 방식	파일의 전체나 일부분	파일의 일부분

가. 대칭키 암호화 방법의 장점

저작물을 패키지화할 때 암호화를 수행하므로 다운로드 시 암호화를 수행하는 것보다 다운로드 시간을 줄일 수 있으며, 같은 저작물에 대하여 같은 대칭키로 암호화를 수행하므로 한 사용자가 다른 사용자에게 저작물을 전달할 수 있다. 그리고 저작물을 전달 받은 사용자가 저작물에 다른 사용자에게 복사하여 이를 전달받은 제 삼자 역시 저작물에 대한 라이선스를 서버로부터 새로이 발급받는 경우에 해당 저작물을 사용할 수 있는 저작물 재분배가 가능하다.

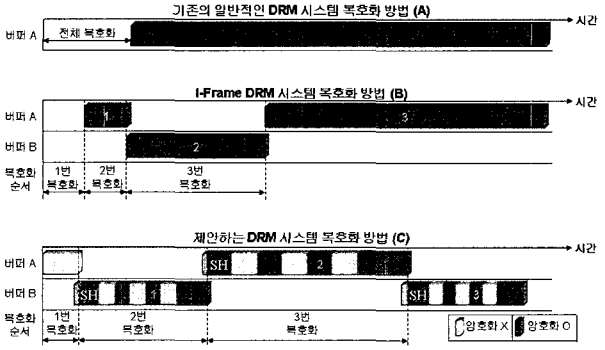
나. 대칭키 암호화 방법의 단점

저작물에 대한 암호화를 미리 수행할 경우 하나의 저작물을 하나의 대칭키로 암호화하는 것이므로 사용자가 해당 대칭키를 노출시킨다면 더 이상 해당 저작물에 대한 안전은 보장받지 못하며, 또한 키를 노출시킨 사용자가 누구인지 알 수 없어서 해당 사용자를 추적할 수 있는 방법이 없다.

제안하는 방식은 하나의 대칭키만을 사용하는 것이 아니라 복호화 에이전트에 의해 여러 개의 대칭키를 유추하기 때문에 사용자가 임의로 키를 노출시킬 위험이 낮다. 그리고 만약 사용자에 의한 하나의 키가 노출된다 할지라도 나머지 키 값을 모르기 때문에 전체 동영상의 복호화가 불가능하다. 또한 기존의 기법은 동영상 전체에 대한 암호화를 수행하는 방식이므로 대용량의 동영상에 적용할 경우 많은 시간이 소요된다. 그러나 제안하는 기법은 동영상 파일의 일부분만을 암호화 수행하기 때문에 상대적으로 적은 양의 데이터를 암호화하면서 효율은 높인 방식으로 대용량의 동영상에 대해서도 적용이 가능하다.

4.1.2 복호화에 대한 비교 분석

DRM 시스템에서 파일은 암호화된 상태로 사용자의 컴퓨터에 저장되어 있다. 만약 암호화되지 않은 상태로 저장이 된다면 사용자에 의한 저작물의 불법복제와 노출이 발생할 수 있으므로 암호화하여 저장하여야 한다. 그러므로 사용자가 저작물을 실행할 경우 사용자 에이전트에 의하여 복호화가 이루어진다. 그러나 대용량 파일인 경우 복호화에 많은 시간이 소요되므로 사용자는 오래 기다려야 한다. 기존의 DRM 시스템의 기법과 제안하는 DRM 시스템 기법의 차이는 (그림 14)와 같다.



(그림 14) 기존 시스템과 제안하는 시스템의 복호화 기법

사용자가 동영상을 실행하면 에이전트는 해당 저작물에 대한 라이선스의 유효성을 서버에 접속하여 검증하여 정당한 사용자인 경우 복호화를 수행하여 해당 저작물을 재생한다. 복호화 과정을 수행을 하기 때문에 (A), (B), (C) 3개의 방법이 모두 동일하다.

그러나 기존의 일반적인 DRM 시스템 복호화 방법(A)은 전체 동영상에 대한 복호화가 끝난 후에 재생을 수행하므로 사용자는 복호화가 끝날 때 까지 긴 시간을 기다려야 하므로 대용량 동영상 파일인 경우 복호화에 많은 시간이 소요되므로 실시간적인 서비스를 제공할 수 없다. I-Frame DRM 시스템 복호화 방법(B)의 경우 이중 버퍼 알고리즘을 사용하였지만 동영상의 모든 프레임이 암호화 되어 있기 때문에 바로 시작하지 못한다는 단점이 있다. 제안하는 DRM 시스템 복호화 방법(C)에서는 재생시 지연시간이 없도록 개선하여 실시간적인 서비스를 가능하게 하였다. 기존의 DRM 시스템과 제안하는 DRM 시스템에 대한 항목별 비교분석은 <표 2>와 같다.

〈표 2〉 기존 DRM 시스템과의 복호화에 대한 분석

비교 항목	Microsoft DRM 시스템	I-Frame 암호화 DRM 시스템	제안하는 DRM 시스템
데이터 파일 재생 시작 시간	전체 복호화 완료 후	부분 복호화 완료 후	실시간
복호화 수행 시기	데이터 파일 재생 전	데이터 파일 재생 전	데이터 파일 재생과 동시 수행
재생 지연 소요 시간	상대적으로 큼 (제안 시스템과 비교)	상대적으로 중간 (제안 시스템과 비교)	없음
이중 버퍼	미사용	사용	사용

4.2 성능 비교

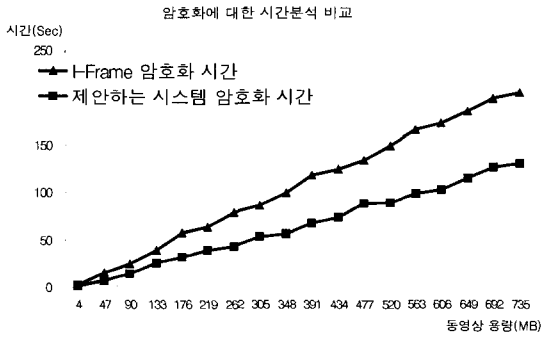
본 논문에서 실험 평가를 하기위해 사용한 비교 DRM 시스템은 Microsoft사의 DRM 시스템과 I-Frame DRM 시스템을 가지고 비교 분석하였다. 실험 데이터 샘플은 18개의 서로 다른 파일크기를 가지고 있는 동영상 데이터를 사용하였고, Microsoft사의 DRM 시스템의 경우는 Version 1 Key ID를 사용을 하였으며, I-Frame 암호화 DRM 시스템의 경우는 AES 암호화 방법을 128비트에서 256비트로 수정하여 평가를 수행하였다. 본 논문에서 사용한 MD5 해쉬 알고리즘은 128비트 이다. 기존의 논문은 I-Frame을 가지고 암호화 하는 방법이며, 제안하는 시스템은 I-Frame과는 무관하다.

암호화에 대한 시간을 비교 분석한 결과는 제안한 시스템이 <표 3>과 같이 I-Frame DRM 시스템 보다 약 1.56배 향상되었다.

〈표 3〉 기존 시스템과의 암호화에 대한 시간

파일크기(MB)	동영상 재생 시간 (Sec)	I-Frame 갯수	Microsoft DRM 암호화 시간 (Sec)	I-Frame DRM 암호화 시간 (Sec)	제안하는 시스템 암호화 시간 (Sec)
4.2	20	610	99.269	1.416	0.937
12.2	60	1,821	249.413	4.961	3.109
120.5	600	18,001	2122.847	47.957	30.906
717.078	5,505	139,129	9468.497	204.670	131.657

Microsoft DRM 시스템은 암호화시 동영상 콘텐츠를 WMV 파일로 인코딩 작업을 수행한 후 암호화 작업을 하기 때문에 암호화에 대한 시간 분석 비교 그래프에서 제외시켰다(그림 15).



(그림 15) 암호화에 대한 시간분석 비교 그래프

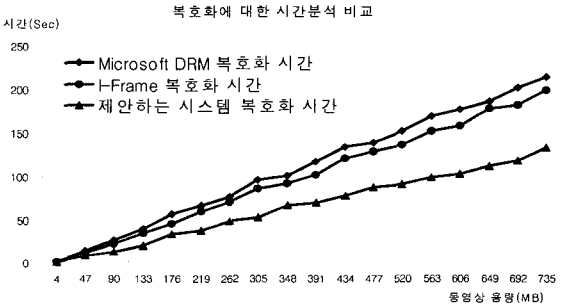
그리고 I-Frame 만을 암호화하는 시스템은 동영상 전체를 암호화 하지 않고, 동영상의 I-Frame만을 암호화하기 때문에 부분 암호화 시스템에 속하지만, I-Frame을 추출하기 위하여 GOP(Group of Picture) 그룹의 모든 헤더의 내용을 읽은 다음 I-Frame의 크기를 계산하여 추출하기 때문에 제안하는 시스템 암호화 방법보다 느리다.

복호화에 대한 시간을 비교 분석한 결과는 <표 4>와 같이 기존의 I-Frame DRM 시스템 보다 약 1.61배 향상되었다. Microsoft사의 DRM 시스템의 경우는 파일전체를 암호화하였기 때문에 복호화 역시 가장 늦으며, I-Frame DRM 시스템은 암호화 방법과 마찬가지로 GOP 그룹의 모든 헤더를 읽어 I-Frame을 얻어내야 하기 때문에 제안한 시스템보다는 복호화 속도가 느리다.

〈표 4〉 기존 시스템과의 복호화에 대한 시간

파일크기(MB)	동영상 재생 시간 (Sec)	I-Frame 갯수	Microsoft DRM 복호화 시간 (Sec)	I-Frame DRM 복호화 시간 (Sec)	제안하는 시스템 복호화 시간 (Sec)
4.2	20	610	2.084	1.764	0.969
12.2	60	1,821	3.831	3.505	2.234
120.5	600	18,001	35.794	32.651	20.791
717.078	5,505	139,129	211.279	194.043	129.482

(그림 16)은 복호화에 대한 시간 분석을 비교 분석한 값을 그래프로 나타낸 것이다.



(그림 16) 복호화에 대한 시간분석 비교 그래프

기존의 시스템과 제안하는 시스템과의 전반적인 사항을 비교 분석 해 보면 제안하는 시스템은 기존의 시스템과 같이 모든 동영상 파일을 지원하며 동영상 전체에 대한 암호화를 수행하지 않기 때문에 암호화 및 복호화에 대한 속도가 기존의 시스템보다 더 향상되었다.

〈표 5〉 기존 시스템과 제안하는 시스템의 비교 분석

비교	Microsoft DRM	I-Frame DRM	제안하는 시스템
적용 파일	ASF / WMV 파일	모든 동영상 파일	모든 동영상 파일
파일 암호화 방법	파일 전체	파일 일부분	파일 일부분
동영상 암호화 속도	상대적으로 느림	상대적으로 빠름	상대적으로 빠름
동영상 복호화 속도	느림	느림	빠름
동영상 암호화 키	1개	1개	n개
키 유출	취약	취약	안전

기존의 시스템에서는 암호화에 쓰인 동영상 콘텐츠 암호화 키는 1개만을 사용하였지만 <표 5>와 같이 제안하는 시스템에 대해서는 n개를 사용하였기 때문에 1개의 키가 유출

이 되더라도 동영상 전체가 복호화 되지 않아 더 안전하다. 그리고 제안하는 시스템은 전송되는 키 값이 없기 때문에 안전하며, 기존의 시스템에서는 아직까지 PKI 시스템이 안전하기 때문에 안전하다고 볼 수가 있다.

5. 결 론

본 논문에서는 시큐리티 에이전트를 이용한 사용자 인증과 다중키를 사용한 DRM 보안 시스템에 대하여 제안하였다.

제안한 시스템은 디지털 콘텐츠 사용에 대한 인증에 있어서 불법적인 사용자에 의한 비밀키 유출을 막기 위하여 서버에서 시큐리티 모듈에서 다수의 비밀키를 사용하여 부분적으로 암호화하여 하나의 비밀키가 노출 되더러도 저작물 전체에 대한 복호화를 사전에 봉쇄함으로써 저작물을 재생할 수 없도록 한다. 또한, 파일 전체가 아닌 부분적으로 암호화를 수행하므로 다른 기존의 시스템보다 암호화 및 복호화 속도를 개선하도록 설계하였다.

클라이언트의 시큐리티 에이전트는 동영상 실행 시 대용량의 복호화를 수행하기 위해서는 많은 시간이 필요하므로 스트림 방식으로 원활한 동영상의 재생을 위해 보상 버퍼제어 방식을 제안하여 효율적인 버퍼 스케줄링을 수행하여 사용자에게 실시간 복호화 및 재생을 할 수 있도록 하였으며, 동영상 재생을 위한 버퍼 스케줄링 시 끊김 현상이 없도록 시스템을 설계하였다.

향후 과제는 휴대폰 및 PDA와 같은 개인 이동식 휴대 단말기에서 활용할 수 있도록 시스템을 개선할 계획이다.

참 고 문 헌

[1] 김정재 외 2명, "동영상 데이터 보호를 위한 공유키 풀 기반의 DRM 시스템," 정보처리학회논문지C, Vol.12-C No.02, pp. 0183~0190, April, 2005.

[2] 김지홍 외 5명, '전자상거래 보안기술', 생능출판사, 2001.

[3] Brad Cox, Superdistribution : Objects As Property on the Electronic Frontier, Addision-Wesley, May, 1996.

[4] Sung, J Park, "Copyrights Protection Tech niques," Proceedings International Digital Content Conference, Seoul Korea, Nov., 28~29, 2000.

[5] V.K. Gupta, "Technological Measures of Protection," Proceedings of International Comference on WIPO, Seoul Korea, October, 25~27, 2000.

[6] Whitfield Diffie and Martin Hellman, "New Directions in Cryptography, " IEEE Transaction on Information Theory, Vol.IT-22, No.6, pp.644~654, November, 1976.

[7] Intertrust : http://www.intertrust.com/main/ overview/ drm.html

[8] Joshua Duhl and Susan Kevorkian, "Unde rstanding DRM system : An IDC White paper," IDC, 2001.

[9] Joshua Duhl, "Digital Rights Management : A Definition," IDC 2001.

[10] Microsoft : http://www.microsoft.com/windows/window-smedia/drm.asp

김 정 재

e-mail : argniss@empal.com

1999년 영동대학교 컴퓨터공학과(공학사)

2001년 숭실대학교 컴퓨터학과(공학석사)

2005년 숭실대학교 컴퓨터학과(공학박사)

관심분야 : 멀티미디어 보안, 멀티미디어 데이터베이스, DRM

이 경 석

e-mail : kslee@kiet.re.kr

1978년 숭실대학교(학사)

1981년 성균관대학교(석사)

1983년~1986년 Univ. Paris 7 연구소 (ITODYS) 연구원

1986년 University Paris 7, 박사

1987년~현재 산업연구원 연구위원

2001년~현재 건국대학교 정보통신대학원 겸임교수

관심분야 : 데이터베이스, 네트웍 보안, 정보보안표준, 정보보안 알고리즘

전 문 석

e-mail : mjun@computing.ssu.ac.kr

1981년 숭실대학교 전자계산학과(공학사)

1986년 University of Maryland Computer Science(공학석사)

1989년 University of Maryland Computer Science(공학박사)

1989년~1991년 New Mexico State University Physical Science Lab 책임연구원

1991년~현재 숭실대학교 컴퓨터학과 정교수

관심분야 : 전자상거래 보안, 인터넷 보안, 멀티미디어 보안, 인증시스템