동영상 데이터 보호를 위한 공유 키 풀 기반의 DRM 시스템

김 정 재[†] · 박 재 표[††] · 전 문 석[†††]

요 약

본 논문에서는 동영상 데이터 암호화를 위해 비디오 데이터의 I-프레임 암호화 기법을 제안하고, 시스템 서버에서 암호화된 동영상 데이터를 클라이언트 시스템에서 사용자가 실행할 때 자동으로 사용자 인증과 데이터의 복호화를 수행할 수 있도록 하는 라이선스 에이전트와 동영상 데이터의 실행 시 공유 키 풀(shared key-pool)을 이용한 PKI(Public Key Infrastructure)기반의 사용자 인증과 동영상 데이터의 암호 및 복호화 시스템을 제안한다. 또한 대용량의 동영상 데이터 실행 시 복호화를 수행하면서 발생하는 많은 재생 지연시간을 줄이기 위해 이중 버퍼 제어를 구성하고 효율적인 버퍼 스케줄링을 이용한 실시간 복호화 방식을 제안한다.

A Digital Right Management System based on Shared Key Pool for Video Data Protection

Jung-Jae Kim[†] · Jae-Pyo Park[††] · Moon-Seog Jun[†††]

ABSTRACT

In this thesis, first, we propose I-frame encryption techniques of video data for video data itself encryption and propose license agent that processing user's certification and decryption in client system automatically when user execute encrypted video data in system server. License agent runs user's certification, encryption and decryption of video data based on PKI(Public Key Infrastructure) using shared key-pool when execute of video data. Also, compose duplex buffer control and propose real time decryption method using efficient buffer scheduling to reduce much playing delay times that happen processing decryption when execute of videoa data of high-capacity.

키워드 : DRM, 공유 키 풀(shared key-pool), PKI

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 불법복제 방지를 위한 디지털 저작권 보호문제가 중요한 이슈로 대두되고 있다. 디지털 저작물 보호를 위해서는 안정성과 보안성 확보를 위하여 정보보호 기술이 필요하고, 디지털 저작권과 저작물 유통의 전반을 감시하고 추적하기위한 디지털 저작권 관리(DRM : Digital Right Management) 기술이 필요하다[1]. DRM 기술을 통해 디지털 저작물에 대한 지적재산권 침해 사례로부터 저작권을 보호하고, 유통과정을 관리하기 위한 종합적인 대책이 추진되어 저작물 제작, 유통, 이용 등이 일련의 신뢰할 수 있는 환경에서 이루어질 수 있도록 하는 다양한 연구가 진행 중에 있다[5]. 기존 DRM 솔루션들은 암호화에 사용하는 키로 비밀키를 사용하여 사용자가 파일을 다운로드할 때 암호화를 수행하므로 많은 시간이 소요가 된다. 또한 복호화를 수행하는 경우에도 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. 그리고 기존의 DRM 솔루션들은 동영상 데이터 안에 데이터의 보호조건이나 저작관리 등의 정보를 삽입하여 관리를 수행하는 정적인 저작권 관리를 하기 때문에 저작권에 대한 동적인 제어가 어려울 뿐 아니라, 감시 및 추적 기능의 제약으로 불법적인 복제 등 지적재산권 침해 발생 시 불법행위 입증에 필요한 자료 확보의 어려움 등 해결해야 할 많은 과제를 가지고 있다. 기존의 DRM은 이 문제를 해결하기 위해서 소프트웨어 에이전트를 사용하여 사용자의 데이터 사용을 모니터링 하기도 하지만 오프라인 환경인 경우 그 기능상에

※ 본 논문은 숭실대학교의 연구비의 지원에 의하여 수행되었음.
† 준 회 원 : 숭실대학교 컴퓨터학과 박사과정 수료
†† 정 회 원 : 숭실대학교 컴퓨터학과 공학박사
††† 종신회원 : 숭실대학교 정교수
논문접수 : 2004년 11월 29일, 심사완료 : 2005년 1월 6일

많은 제약을 가지고 있다. 따라서 온라인 및 오프라인 환경에서 모든 저작물 유형에 적용이 가능하면서 동적인 저작권 관리와 실시간 감시 및 추적을 가능하게 하는 디지털저작권 관리 기술의 개발이 필요한 실정이다[4].

본 논문에서는 온라인과 오프라인 상에서 멀티미디어 저작물에 대한 사용자 인증과 데이터 자체의 암호화를 통해 불법적인 실행을 방지할 수 있는 통합적인 DRM 시스템을 제안한다.

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 필요하다. 그러므로 DRM 시스템에 대한 많은 연구와 솔루션 개발이 진행 중에 있다[9, 10].

DRM은 저작권 보호기술을 이용하여 허가되지 않은 사용자로부터 디지털 저작물을 안전하게 보호함으로써 저작권자의 권리 및 이익을 지속적으로 보호하고 관리하는 기술이다[14].

저작물의 기밀성과 무결성을 확보를 위하여 암호기술을 중심으로 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[11]. 또한 전자상거래시스템과 결합하여 다양한 조건에 따른 디지털 저작물의 유통을 통합된 방식으로 제공할 수 있는 솔루션들이 등장하고 있다[6, 7].

2.2 기존의 DRM 시스템

2.2.1 InterTrust의 DRM 시스템

InterTrust 사의 DRM 솔루션 특징은 저작물의 보호를 위해서 암호기술과 워터마킹을 사용하며 저작물 사용규칙을 지정하여 사용내역의 수집 및 기록, 과금 처리를 수행하는 것이다. 사용자 컴퓨터에 에이전트를 실행하여 라이선스와 과금 처리, 저작물의 실행을 에이전트를 통하여 처리하도록 하였다. 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[13, 14]. 또한 저작물이 암호화되어 보호되고 있으므로 사용자들 사이에 암호화된 저작물을 주고받을 수 있는 저작물 재분배(SuperDistribution)를 실현하였다[8].

2.2.2 Microsoft의 DRM 시스템

Microsoft의 DRM 시스템은 저작물 제공자와 소비자들에게 디지털 미디어 파일을 안전하게 분배하는 종단 간(end-to-end) DRM 시스템이다[15]. 핵심 제어 부분은 WMRM (Windows Media Rights Manager)으로서 WMRM의 Rights Manager는 저작물 제공자에게 인터넷 상에서 암호화된 파일 형식으로 보호된 음악, 비디오 등의 미디어를 배달한다. WMRM에서 각각의 서버 또는 클라이언트 인스턴스들은 개인화(individualization)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.

2.3 기존의 DRM 시스템 분석

InterTrust사와 Microsoft사의 DRM 솔루션에 대해 저작물의 생성규칙, 암호화와 키의 분배, 인증 및 사용 내역 관리, 유통 및 추적기능 등에 대하여 <표 1>과 같이 비교 분석한다[3, 12, 14].

저작물 사용규칙의 정의는 사용 규칙을 얼마나 다양하게 제어 할 수 있는가 하는 것이 중요한 문제이다. 현재 개발된 기술들의 대부분은 사용횟수, 사용기간(사용개시일, 사용만료일), 사용자의 나이 등에 따라 저작물을 실행할 수 있도록 허용하는 저작물의 사용규칙을 정의한다.

<표 1> DRM 시스템 기술 분석 비교

DRM 시스템 / 기술 항목	InterTrust DRM	Microsoft DRM
사용 규칙 정의	XrML(비공개)	XrML 미사용
저작물 패키징	DigiBox Container를 통하여 처리	WMRM을 통하여 처리 암호화된 저작물 라이선스 키
주요기술	암호화, 워터마킹	암호화
암호화 방식	공개키/비밀키	비밀키
키의 저장위치	서버에 저장	클리어링하우스에 저장
인증처리	배포자로부터 InterRights Point를 다운받아 처리	WMRM이 클리어링하우스를 통해 라이선스 키 획득
사용내역 저장	사용자 컴퓨터	클리어링하우스
결제 방식	신용카드, 전자화폐	클리어링하우스에 미리 등록
유통방식	온라인 및 오프라인	오프라인
추적 기능	제한적	불가능
복호화 장소 및 수단, 독립성	전용 플레이어 플레이어에 종속	Microsoft Media Player 플레이어에 종속

저작물의 패키징 작업에 대해서는 두 회사 모두 원 저작물에 저작권 정보를 삽입하는 기법을 취하기 때문에 해당 저작물 생성 환경에 종속적인 특성이 있으므로 특정 파일 형식을 사전에 알아야 하는 문제점이 있다. 또한 저작물에 저작권 정보를 삽입하는 방식이므로 패키징 작업이 끝난 후에는 보호 조건의 변경이 필요한 경우 변경할 수 없으므로 필요시 재 패키징을 해야 하는 문제점이 존재한다. 그러므로 저작권 정보를 라이선스에 포함하여 라이선스에 통해 관리를 하는 것과 비교해서 저작권의 정적인 관리가 불가피하다.

DRM 솔루션에 적용된 주요기술 역시 적법한 사용자가 플레이어를 통하여 평문을 획득했을 경우, 이를 무단으로

복사하여 배포하는 것을 막을 수가 없는 문제점이 있다.

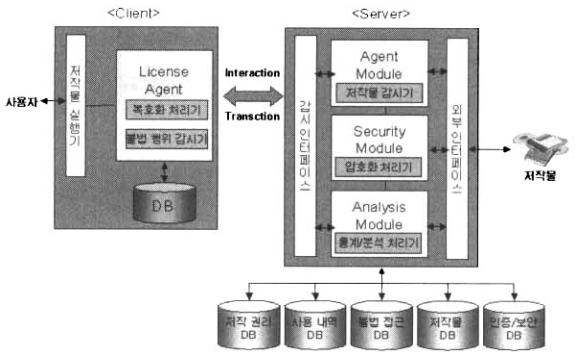
저작물의 암호화 방식은 전제 시스템 특성에 따라 많은 영향을 주게 된다. 두 시스템 모두 사용자가 키를 노출했을 경우에는 심각한 보안 위협이 된다.

추적 기능 제공 측면에서 기존 DRM 솔루션의 경우 특정 저작물에 대한 법적 분쟁이 발생했을 때 불법 사례 입증에 필요한 기초자료 제공을 위해 추적 기능을 충분히 제공하지 못하고 있다.

복호화의 장소는 소비자가 저작물을 이용할 때 두 업체 모두 각 업체별로 제공하고 있는 별도의 플레이어를 통해서만 복호화를 수행할 수 있으므로 플레이어에 의존해야 하는 문제점이 있다.

3. 제안 시스템 구조

제안하는 시스템은 (그림 1)과 같이 클라이언트/서버 구조로 구성되어 운용된다. 서버는 외부인터페이스, 저작물 감시기, 암호화 처리기, 통계분석 처리기, 감시 인터페이스와 데이터베이스로 구성되며 클라이언트는 라이선스 에이전트와 저작물 실행기로 구성된다. 시스템 서버에 외부 인터페이스를 통해 저작물이 등록되면 저작물 감시기의 에이전트 모듈에 따라 저작물은 사용횟수, 사용기간(사용개시일, 사용만료일), 사용자의 나이 등에 따라 저작물을 실행할 수 있도록 허용하는 저작물의 사용규칙 등에 대해 저작물 감시 처리가 이루어지고 저작물에 대한 암호화 과정이 수행된다.



(그림 1) DRM 시스템 구성도

동영상 전체에 대한 암호화를 수행할 경우 대용량으로 인하여 암호화에 많은 시간이 소요될 뿐만 아니라 복호화에도 많은 시간이 소요되어 실시간 재생에 문제점이 발생하므로 동영상의 I-프레임 부분만을 암호화한다. 또한 사용자에 의해 비밀키의 누출을 막기 위해서 해당 동영상 저작물에 대해 공유 키 풀을 생성한다. 에이전트 모듈의 저작물 감시기는 사용자 에이전트와 통신하며 사용자의 행위와 사용자가 다운로드 받은 동영상에 대해 감시를 수행한다. 분석 모듈의 통계분석 처리기는 감시 인터페이스를 통해 사용자의 행위와 저작물의 사용상황을 감시하고 그 결과를 분석하여 데이터베이스에 저장한다.

사용자가 보호된 동영상 데이터를 사용하기 위해서는 등록과 인증과정을 거쳐야 한다. 초기 사용자 등록을 하면 서버는 PKI의 인증서를 바탕으로 사용자 인증을 수행한다. 인증된 사용자는 자신의 시스템에 사용자 에이전트를 설치하고 에이전트를 통하여 보호된 동영상 데이터를 실행하기 위해 라이선스를 발급받게 된다. 인증과정에서 전송되는 데이터의 보호를 위해서는 PKI 인증서의 공개키를 가지고 암호화를 수행하며 알고리즘으로는 RSA 암호화 알고리즘을 사용한다. 사용자 등록 및 인증이 이루어지면 라이선스 에이전트는 사용자 행위를 감시하여 보호된 동영상 실행을 감지한다. 사용자가 저작물을 실행하면 사용자 에이전트는 사용자의 요구에 맞는 라이선스를 확인하여 인증된 사용자라면 복호화를 수행하여 저작물을 실행할 수 있도록 비밀키를 생성하고 복호화 작업을 수행한다. 만약 인증되지 않은 사용자라면 안내 메시지와 함께 라이선스를 획득하도록 유도한다.

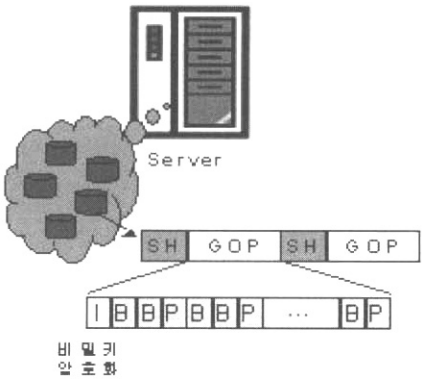
4. 사용자 인증과 복호화 기법

4.1 암호화 과정

저작물 저작자는 저작물을 서버에게 전송한다. 그러면 서버는 해당 저작물을 임의의 비밀키 Ks로 암호화하여 암호화된 저작물 C를 서버의 저작물 데이터베이스에 비밀키 Ks와 같이 저장한다.

$$C = E_{Ks}[data]$$

사용자는 원하는 저작물을 인증과정을 통하여 서버에서 다운로드 받거나 다른 사용자로부터 복사해서 사용할 수 있다. 그러나 다운로드 된 저작물은 암호화 되어 있으므로 에이전트를 통하여 실행할 수 있다. 서버는 사용자에 의한 비밀키 누출을 막기 위해서 비밀키를 암호화할 수 있는 공유 키 풀을 생성한다.



(그림 2) 암호화 과정

서버는 (그림 2)와 같이 동영상 GOP의 I-프레임을 비밀키를 이용하여 AES 알고리즘이나 SEED알고리즘 중에서 하나를 선택하여 암호화한 후 해당 저작물의 ID와 비밀키의

값을 서버의 데이터베이스에 저장한다. 그리고 암호화된 동영상에 적용할 수 있는 임의의 공유 키 풀을 생성하여 역시 데이터베이스에 저장한다.

사용자의 등록이 이루어지면 서버는 사용자 인증서를 이용하여 사용자 인증을 수행한 후 사용자의 고유한 키 생성을 위하여 공유 키 풀에서 사용자의 인증서 정보를 이용하여 개인 정보를 추출한다. 또한 비밀키(Ks)와 공유 키 풀의 각 열을 비트 단위로 배타적 논리합(bit-wise XOR) 연산을 수행하여 암호화된 공유 키 풀을 생성한다. 개인정보와 암호화된 공유 키 풀은 서버의 데이터베이스에 저장되고 사용자 에이전트를 통하여 사용자의 데이터베이스에도 저장한다.

4.2 공유 키 풀

공유 키 풀(Shared Key-Pool)을 구성하기 위해서 저작물 제작자는 분배할 동영상을 비밀키 Ks를 사용하여 암호화한다. 비밀키 Ks는 다음과 같이 k개의 비트열로 나눌 수 있다.

$$Ks = Ks_1 | Ks_2 | \cdots | Ks_k \quad \text{(식 1)}$$

일반적으로 비밀키 암호화에서 비밀키 Ks는 128 비트를 사용한다. 암호화에 사용한 비밀키의 보안성을 높이기 위해서 비밀키를 암호화한다. 비밀키를 암호화하여 사용자가 접근할 수 없도록 하기 위하여 공유 키 풀(shared key pool)을 사용한다. 공유 키 풀은 $k * 2^{\frac{n}{k}}$개의 비트로 구성되며 다음과 같이 생성한다.

$$\left\{a_1^0, a_1^1, a_1^2, \cdots, a_1^{2^{\frac{n}{k}}-1}, \cdots, a_k^0, a_k^1, a_k^2, \cdots, a_k^{2^{\frac{n}{k}}-1}\right\} \quad \text{(식 2)}$$

〈표 2〉 암호화 키 Ks에 대한 공유 키 풀

a_1^0	a_1^1	⋯	$a_1^{2^{\frac{n}{k}}-1}$
a_2^0	a_2^1	⋯	$a_2^{2^{\frac{n}{k}}-1}$
⋮	⋮	⋮	
a_k^0	a_k^1	⋯	$a_k^{2^{\frac{n}{k}}-1}$

키의 크기에 맞게 적용할 수 있도록 k개의 행과 $2^{\frac{n}{k}}$개의 열을 갖는 행렬 <표 2>과 같이 나타낼 수 있다.

각 사용자에 대한 개인용 키 Kp는 k비트로 이루어진 다음과 같은 비트열의 집합이다.

$$Kp = a_1^{b_1} | a_2^{b_2} | \cdots | a_k^{b_k} \quad \text{(식 3)}$$

이 때 b_i는 키 풀에서 각 i 번째 행의 값으로서 사용자의 개인용 키를 결정하는 중요한 값이다. b_i는 사용자 인증서의 공개키에서 추출한다.

$$B = b_1 | b_2 | \cdots\cdots | b_k \quad \text{(식 4)}$$

사용자의 공개키는 일반적으로 비도가 낮은 경우에는 512비트의 값을 사용하고 비도가 큰 중요한 정보인 경우에는 1024비트의 값을 사용한다. 일반적으로 n비트의 공개키를 사용하는 경우 개인용 키의 길이가 k비트라면 이 때 키 풀의 각 항목 값들은 $2^{\frac{n}{k}}(0 \sim 2^{\frac{n}{k}}-1)$의 범위를 가지게 된다. 예를 들어, 공개키의 길이가 512비트이면서 개인용 키의 길이가 128비트라면 512/128=4이므로 각 항목의 값은 24=16이 되어 16진수로 0~F의 값을 가지게 된다. 그러므로 공개키의 값 0~F에 따라서 실제 개인용 키의 각 행이 결정되게 된다. 공개키의 길이와 비밀키의 길이에 따른 각 키의 값의 범위는 <표 3>와 같다.

〈표 3〉 키의 길이에 따른 키 풀 값의 범위

공개키의 길이 \ 비밀키의 길이	128	256	512
512	0~F	0~3	0, 1
1024	0~255	0~F	0~3
2048	0~65535	0~255	0~F

그러므로 각 사용자들의 개인용 키는 자신들의 유일한 공개키 값에 의해서 결정되므로 공개키에 의해서 각 행에서 선택된 키 값은 모든 사용자에게 다른 키가 배정되는 것을 보장해준다.

공유 키 풀이 생성되면 비밀키 Ks를 암호화하기 위하여 공유 키 풀의 각 i번째 행에 대하여 $2^{\frac{n}{k}}$개 비트인 $a_i^0, a_i^1, \cdots, a_i^{2^{\frac{n}{k}}-1}$에 각각 Ks_i와 비트 단위의 배타적 논리합(bit-wise XOR)을 하여 식 4와 같이 구한다.

$$\begin{aligned} a'^0_i &= Ks_i \oplus a_i^0 \\ a'^1_i &= Ks_i \oplus a_i^1 \\ &\vdots \\ a'^F_i &= Ks_i \oplus a_i^F \end{aligned} \quad \text{(식 5)}$$

위의 연산을 거쳐서 암호화된 공유 키 풀을 <표 4>와 같이 구할 수 있다.

〈표 4〉 Ks로 암호화된 공유 키 풀

a'^0_1	a'^1_1	⋯	$a'^{2^{\frac{n}{k}}-1}_1$
a'^0_2	a'^1_2	⋯	$a'^{2^{\frac{n}{k}}-1}_2$
⋮	⋮	⋮	
a'^0_k	a'^1_k	⋯	$a'^{2^{\frac{n}{k}}-1}_k$

암호화된 공유 키 풀은 이제 네트워크를 통하여 사용자 에이전트에게 전달된다. 에이전트는 암호화된 동영상 파일을 복호화하기 위하여 암호화된 키 풀에서 사용자의 비밀키

Kp를 이용하여 비밀키 Ks를 찾아낸다. 이 비밀키 Ks를 이용하여 동영상 파일을 복호화하여 사용자에게 보여준다. 에이전트가 암호화된 키 풀과 개인용 키 Kp를 가지고 비밀키 Ks를 찾는 방법은 다음과 같다.

$$Kp = a_1^{b_1} | a_2^{b_2} | \cdots | a_k^{b_k} \text{ 이고 } a'^{j}_{i} = Ks_i \oplus a^{j}_{i} \text{ 이므로}$$

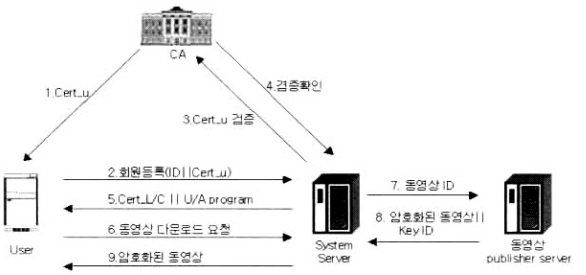
$$a_1^{b_1} \oplus a'^{b_1}_{1} = a_1^{b_1} \oplus Ks_i \oplus a_1^{b_1} = Ks_i \qquad \text{(식 6)}$$

4.3 인증 기법

4.3.1 사용자 인증 프로토콜

사용자인증은 회원 가입을 통해 이루어지며, 회원은 로그인 과정을 거쳐서 파일을 다운로드 받을 수 있다. 그러나 다른 사용자로부터 동영상 데이터를 전달 받은 사용자도 동영상 데이터를 사용할 수 있는 저작물 재분배가 가능하므로 누구든지 회원에 가입할 수 있으며 이를 위해서 PKI기반의 인증서를 통한 로그인을 지원한다. 별도의 ID와 암호 기반의 로그인도 지원을 하지만 그 과정에서도 반드시 PKI기반의 인증서를 확인하여 로그인을 한다. ID와 암호기반의 로그인이나 인증서 기반의 로그인 과정을 통하여 사용자 인증을 받으면 동영상에 대한 다운로드가 가능하다.

사용자는 시스템 서버에 접속하여 자신의 인증서 Cert_u를 전송한다. 시스템 서버는 사용자 인증서 Cert_u를 인증경로를 통하여 CA서버에서 검증하고 올바른 인증서이면 사용자 에이전트 프로그램과 서버의 인증서를 전송하여 인증이 성공적이라는 것을 사용자에게 알린다.



(그림 3) 사용자 인증 프로토콜

이제 사용자는 자신이 필요한 동영상 데이터를 다운로드 받을 수 있다. 그러나 동영상 데이터는 암호화가 되어 있으므로 다운로드 받더라도 바로 실행할 수 없으며 인증과정에서 다운로드 받은 에이전트를 설치하고 에이전트를 통하여 동영상의 실행을 요청할 수 있다.

사용자가 암호화된 저작물을 실행할 경우 라이선스 에이전트는 사용자의 라이선스를 확인한다. 라이선스가 있다면 서버를 통하여 라이선스를 인증하고 라이선스가 없다면 라이선스를 발급받는다.

4.3.2 라이선스 발급 프로토콜

사용자는 라이선스 에이전트(LA) 프로그램을 설치하고 라이선스 에이전트를 실행한다. 사용자의 PC에 탑재된 라이선스 에이전트는 사용자가 암호화된 저작물을 실행하면 사용자의 라이선스를 확인한다. 라이선스가 있다면 서버를 통하여 라이선스를 인증하고 라이선스가 없다면 시스템 서버에 접속하여 라이선스를 발급받는다.

서버는 해당 인증서의 정보에 의하여 공유 키 풀에서 개인정보를 추출하고 추출된 개인정보와 암호화된 동영상의 비밀키(Ks)를 공유 키 풀에 저장한 암호화된 공유 키 풀을 라이선스와 함께 사용자에게 전송한다.

라이선스는 라이선스 ID, 사용자 ID, 저작물 ID, 유효기간, 라이선스 검증주기, 사용자의 공개키 정보, 서버 식별 정보 및 권한 등이 저장되어 있는 디지털 문서로서 서버가 자신의 개인키로 서명하여 유효성을 보장한다.

4.3.3 라이선스 인증 프로토콜

라이선스 에이전트는 사용자가 암호화된 저작물을 실행하면 라이선스가 있는지 확인한다. 만약 라이선스가 없다면 라이선스 발급 프로토콜에 따라서 라이선스를 발급받고 라이선스가 있다면 온라인인 경우에는 다음과 같이 해당 라이선스에 대한 인증을 시스템 서버에 요청한다. 만약 오프라인인 경우에도 특정 횟수까지는 실행을 지원하기 위하여 사용자의 데이터베이스 정보에 의하여 라이선스를 자체적으로 인증하고 파일을 실행한다.

시스템 서버는 라이선스 에이전트로부터 라이선스 인증 요청을 받으면 데이터베이스에서 해당 라이선스의 정보와 클라이언트의 라이선스 정보를 비교한 후 라이선스 정보를 수정하고 인증을 하게 된다.

사용자의 라이선스 시스템 정보를 확인하여 서버의 데이터베이스에 추가하고 라이선스가 특정일까지의 시간 라이선스이면 해당 시간이 경과 되었는지 확인하고 사용횟수에 대한 라이선스라면 라이선스 정보를 수정한 후에 수정된 라이선스를 사용자의 공개키로 암호화하여 전송한다.

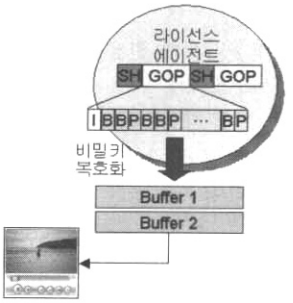
$$Eku\_u[Licence'] \qquad \text{(식 7)}$$

서버로부터 수정된 라이선스를 받은 사용자 에이전트는 해당 라이선스를 바탕으로 클라이언트 측의 데이터베이스 정보를 수정하고 라이선스의 개인정보와 공유키 풀의 값을 바탕으로 비밀키(Ks)를 생성한다. 그리고 비밀키(Ks)를 가지고 암호화된 저작물을 복호화하여 사용자에게 보여준다.

4.3 복호화 기법

클라이언트 에이전트는 사용자의 인증서와 라이선스를 바탕으로 비밀키 Ks를 복구하여 동영상의 복호화를 수행한다. 기존의 시스템은 전체 동영상의 복호화가 끝난 후 실행을 하므로 대용량의 동영상의 경우 사용자가 복호화가 끝나는 시간까지 오랜 시간을 기다려야 한다. 그러나 제안하는 기법은 전체동영상의 복호화가 끝나기 전에 해당 파일을 실행할 수 있다.

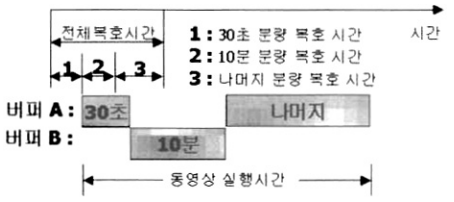
클라이언트 에이전트는 (그림 4)와 같이 복호화를 수행하기 위하여 암호화된 동영상의 I-프레임을 추출하여 비밀키로 복호화를 수행한 후 B, P 프레임과 함께 버퍼에 저장하여 플레이 한다. 버퍼에는 전체 동영상이 플레이 되는 동안 지연되는 프레임을 계산하여 초기에 버퍼 사이즈를 결정한 후 플레이 하도록 한다.



(그림 4) 이중버퍼를 사용한 동영상 복호화 과정

이를 위해서 (그림 5)와 같이 2개의 버퍼를 사용하는 이중버퍼 시스템을 사용한다.

지연프레임을 계산하기 위하여 초기에는 30초 분량의 프레임을 먼저 복호화한 후 이를 버퍼 1에 저장하여 실행을 시작한다. 그리고 동영상이 실행되는 30초 동안에 10분 분량의 데이터를 복호화하여 버퍼 2에 저장한다.



(그림 5) 이중버퍼의 구성

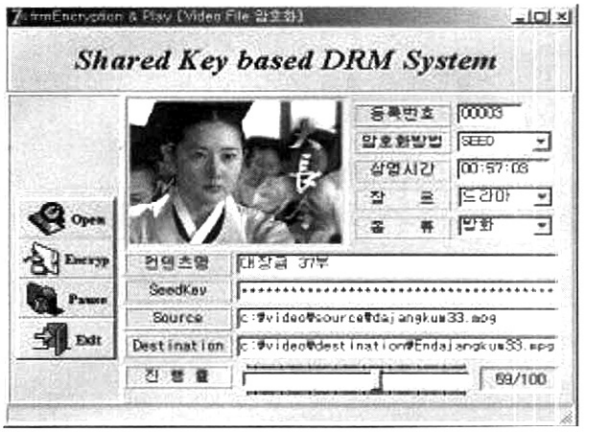
버퍼 1에서 실행이 끝나면 에이전트는 버퍼 2의 데이터가 이어서 실행될 수 있도록 버퍼 2의 메모리 참조 값을 저장한다. 계속해서 동영상 플레이어가 버퍼 2의 데이터를 실행하는 동안에 나머지 데이터를 다시 버퍼 1에 복호화하여 전체적으로 화면이 끊기는 현상이 발생하지 않으면서 동영상을 실행할 수 있다.

5. 실험 평가

5.1 시스템 구현

5.1.1 동영상 암호화 인터페이스

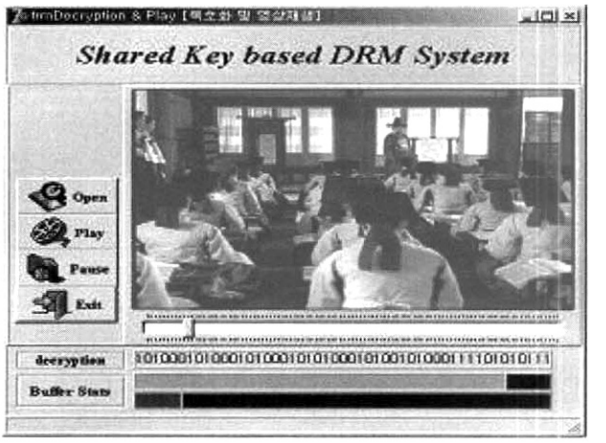
(그림 6)은 동영상에 대한 암호화를 진행한다. 암호화에 사용하는 키와 암호화 하고자 하는 동영상의 위치 정보를 입력하고 암호화를 수행한다. 암호화에 사용한 정보는 데이터베이스로 저장된다. 제안하는 시스템은 SEED암호 알고리즘과 AES암호 알고리즘을 선택하여 암호화를 수행할 수 있도록 구성하였다.



(그림 6) 동영상에 대한 암호화 진행 과정

5.1.2 동영상 복호화 인터페이스

(그림 7)은 동영상에 대한 복호화를 진행하는 화면이다. 복호화 화면은 사용자가 동영상을 실행 시 에이전트에 의하여 실행이 된다. 사용자 에이전트는 사용자의 저작물 실행 시 키를 가지고 동영상 복호화를 수행한다.



(그림 7) 동영상에 대한 복호화 진행화면

5.2 실험 평가

본 논문에서 제안하는 시스템에 대한 성능 평가를 위해 비디오 데이터 자체 암호화 시간과 복호화 시간에 따른 초기 재생 지연 시간을 측정하였다.

실험 환경은 시스템 Intel(R) Pentium-IV CPU 2.4GHz와 512M의 RAM, 그리고 MS-Windows 2000 서버 운영체제를 사용하였으며, 50분 재생시간을 갖는 약450Mbyte 용량의 MPEG 동영상 데이터를 사용하였다.

일반적인 방식인 이미 암호화되어있는 비디오 데이터 파일의 복호화를 먼저 수행한 후 재생(비실시간 복호화 방식)하는 방식과 본 논문에서 제안한 방식인 실시간으로 복호화를 하면서 재생(실시간 복호화 방식)하는 방식을 비교하여 각각 시간을 측정하여 <표 5>와 같은 결과를 도출하였다. 정확한 시간을 측정하기 위해 비디오 데이터 파일을 분(minutes) 단위로 분할하여 실험 데이터로 이용하였다.

〈표 5〉 실행 시 지연시간의 비교

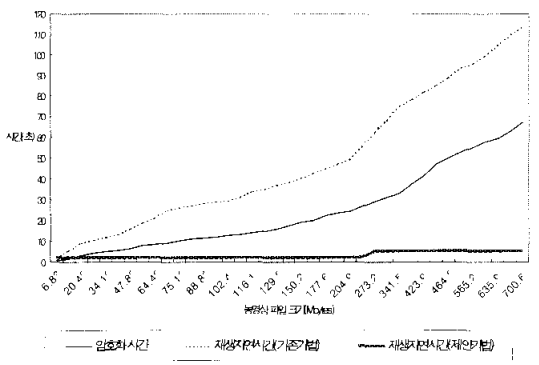
파일크기 (MB)	실행시간 (Min)	복호화시간 (Sec)	기존 기법의 지연시간 (Sec)	제안하는 기법의 지연시간 (Sec)
6.83	1	0.76	2.42	2.42
13.66	2	1.57	5.50	2.42
20.49	3	3.00	9.24	2.42
68.31	10	9.05	25.01	2.42
204.94	30	24.31	49.11	2.42
423.94	60	41.62	82.06	5.50
635.92	90	59.46	104.72	5.50

각 지연시간은 암호화된 비디오 데이터 파일을 복호화한 후 재생하기까지의 소요되는 시간이며, 비디오 데이터 파일의 복호화 시간에 로딩 시간이 더해진 시간이다. 일반적으로 플레이어에 따라 비디오 파일의 로딩 시간이 달라지므로 실험에서는 로딩 시간을 복호화 시간과 같이 처리하여 지연시간을 계산하였다.

동영상 전체에 대하여 동일한 암호화 기법을 사용하여 실험을 하였으므로 기존 기법과 제안하는 기법에서의 암호화 시간은 동일하다. 그러나 복호화 시간을 비교해 보면 기존의 기법에서는 데이터 전체를 복호화한 후 비디오 데이터 파일을 재생하므로 복호화가 실행되어 종료된 후 재생하기까지의 재생 지연 시간이 많이 소요되었으며, 제안하는 기법에서는 제안하는 기법과 알고리즘에 따라 복호화를 수행하면서 이중 버퍼 스케줄링에 의해 동시에 비디오 데이터 파일을 재생하므로 기존의 기법보다 현저하게 재생 지연시간이 감소하였다.

(그림 8)은 39개의 비디오 데이터를 이용하여 기존기법과 제안하는 기법의 암호화 시간과 재생 지연시간을 측정한 후 그래프로 나타내었다.

기존의 기법은 동영상의 크기에 비례하여 암호화 시간이 늘어나며 지연시간 역시 비례하여 늘어나는 것을 알 수 있다. 그러므로 기존의 기법을 대용량 동영상 데이터에 사용할 경우 많은 지연시간이 발생하여 사용자에 대한 서비스 지연 시간이 늘어나게 된다. 이에 반하여 제안하는 기법은 동영상의 크기에 비례하여 암호화 시간이 늘어나지만 복호화를 수행하면서 동시에 재생을 하므로 로딩 시간이 줄어들어 실제 지연시간은 줄어드는 것을 알 수 있다.



(그림 8) 암호화 시간과 지연시간의 비교

비디오 파일 전체를 복호화한 후 재생하는 기존의 방식에서는 동영상 파일 크기가 커질수록 복호화와 파일의 재생을 위한 로딩 시간이 길어졌으나 제안하는 방식에서는 복호화와 동시에 재생을 수행하므로 로딩 시간이 줄어들었다. 그러므로 제안하는 기법이 지연시간이 현저히 짧아지는 것을 알 수 있다. 또한 이중버퍼를 이용하여 실시간으로 복호화를 수행하는 경우에도 재생 중에 끊기는 현상이나 노이즈가 나타나지 않고 안정적인 재생을 보였다.

6. 결 론

본 논문에서는 라이선스 에이전트를 이용하여 디지털 저작권 보호를 위한 동영상 데이터 관리 및 감시 시스템에 대하여 제안하였다.

본 논문에서는 첫째, 동영상 데이터 암호화를 위해 비디오 데이터의 I-프레임을 암호화하는 새로운 암호화 기법을 제안하였다. 둘째, 시스템 서버에서 암호화된 동영상 데이터를 클라이언트 시스템에서 사용자가 실행할 때 자동으로 사용자 인증과 데이터의 복호화를 수행할 수 있도록 하는 라이선스 에이전트를 제안하였다. 셋째, 대용량 동영상 데이터의 실행 시 복호화를 수행하면서 발생하는 많은 재생 지연시간을 줄이기 위해 이중 버퍼 제어를 구성하고 효율적인 버퍼 스케줄링을 이용한 실시간 복호화 방식을 제안하였다.

제안한 시스템이 기존 시스템에 비해 클라이언트 시스템에서 비디오 데이터 파일 재생 시 대용량의 동영상에 대해서 복호화 시간을 포함한 지연시간을 현저히 줄일 수 있는 것을 실험을 통해 확인하였다.

제안하는 시스템은 동영상에 대한 DRM 시스템으로서 인터넷을 통한 영화 및 뮤직비디오, CF 등에 대한 동영상 서비스를 저작권 보호에 사용된다면 좋은 효과를 기대할 수 있다. 또한 본 논문은 향후 PDA와 같은 무선망에서 활용할 수 있도록 개선하는 것이 필요하다.

참 고 문 헌

[1] 김지홍, 이만영, 류재철, 송유진, 염홍렬, 이임영, 전자상거래 보안기술, 생능출판사, 2001.

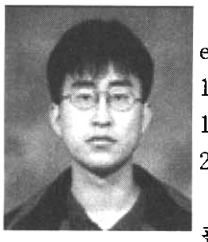
[2] 박재표, 이광형, 김원, 전문석, "라이센스 에이전트를 이용한 디지털 저작권 보호 를 위한 멀티미디어 데이터 관리 및 감시 시스템의 설계," 컴퓨터산업교육학회 논문지, 제5권, 제2호, pp.281-292, 2004.

[3] 배민오, 조규곤, "디지털 컨텐츠 저작권 보호기술동향," 한국정보과학회지, 제18권, 제 7호, pp.0043-0051, 2000.

[4] 이덕규, 박희운, 이임영, "Agent 기반 불법 복제 방지 DRM모델," 정보과학회 2001년 추계학술대회, 제28권, 제2호, pp.682-684, 2001.

[5] 이용효, 황대준, "에이전트 기반의 동적 디지털저작권관리 시스템 설계 및 구현," 정보처리학회논문지D, 제8-D권 제5호, pp.613-622, 2001.

[6] 한국교육학술정보원, "디지털정보에 대한 식별자 부여 및 전자상거래 등 메타데이터 모델에 관한 연구," 1999.
[7] 한국교육학술정보원, "디지털 자원 보호체제 구축 방안에 관한 연구," 2000.
[8] Brad Cox, Superdistribution : Objects As Property on the Electronic Frontier, Addision-Wesley, May, 1996.
[9] Sung, J Park, "Copyrights Protection Tech niques," Proceedings International Digital Content Conference, Seoul Korea, Nov., 28-29, 2000.
[10] V.K. Gupta, "Technological Measures of Protection," Proceedings of International Comference on WIPO, Seoul Korea, October, 25-27, 2000.
[11] Whitfield Diffie and Martin Hellman, "New Directions in Cryptography," IEEE Transac tion on Information Theory, Vol.IT-22, No.6, pp.644-654, November, 1976.
[12] Intertrust : http://www.intertrust.com/main/ overview/drm.html
[13] Joshua Duhl and Susan Kevorkian, "Unde rstanding DRM system: An IDC White paper," IDC, 2001.
[14] Joshua Duhl, "Digital Rights Management : A Definition," IDC 2001.
[15] Microsoft : http://www.microsoft.com/windows/windows media/drm.asp

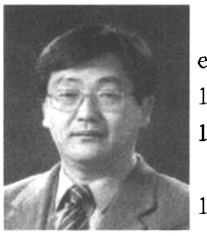


김 정 재

e-mail : argniss@empal.com
1995년 영동대학교 컴퓨터공학과(공학사)
1999년 숭실대학교 컴퓨터학과(공학석사)
2001년 숭실대학교 컴퓨터학과(박사과정 수료)
관심분야 : 멀티미디어 보안, 멀티미디어 데이터베이스, DRM

박 재 표

e-mail : pjerry@dreamwiz.com
1996년 숭실대학교 컴퓨터학부(공학사)
1998년 숭실대학교 컴퓨터학과(공학석사)
2004년 숭실대학교 컴퓨터학과(공학박사)
관심분야 : 멀티미디어 보안, DRM, PKI



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)
1991년~현재 숭실대학교 정교수
관심분야 : 전자상거래 보안, 인터넷 보안, 멀티미디어 보안, 인증 시스템