디지털 저작권 관리에서 사용자의 프라이버시 보호를 제공하는 라이센스 관리 프로토콜

(A License Administration Protocol Providing User Privacy in Digital Rights Management)

박 복 녕 † 김 태 윤 ††
(Bok-Nyong Park) (Tai-Yun Kim)

요 약 개인 정보 유출로 인한 피해사례가 늘어나면서 사용자 프라이버시 침해에 대한 인식이 높아지고 있다. 그러나 기존의 DRM 시스템은 사용자의 프라이버시 보호가 저작권을 보호하는데 직접적으로 필요하지 않다는 이유로 사용자의 프라이버시 보호에 대해 고려하지 않았다. 본 논문에서는 DRM에서 사용자 프라이버시 보호 측면이 강조된 라이센스 관리 프로토콜을 제안한다. 제안한 프로토콜은 사용자 식별 정보의 노출을 보호하기 위해 임시 ID와 token을 사용함으로 익명성을 보장하고 *ECDH* 세션키와 공개키 암호 시스템을 이용하여 메시지를 암호화함으로써 사용자 정보의 유출을 방지하여 사용자의 프라이버시를 보호한다.

키워드 : 디지털 저작권 관리, 공개키 암호 시스템, 익명성, 프라이버시, 라이센스

Abstract As people are damaged increasingly by personal information leakage, awareness about user privacy infringement is increasing. However, the existing DRM system does not support the protection of user's personal information because it is not necessary for the protection of copyrights. This paper is suggesting a license administration protocol which is more powerful to protect personal information in DRM. To protect the exposure of users identifier, this protocol uses temporary ID and token to guarantee anonymity and it uses a session key by *ECDH* to cryptography and Public-Key Cryptosystem for a message so that it can protect the exposure of personal information and user's privacy.

Key words : DRM, Public-Key Cryptosystem, Anonymity, Privacy, License

1. 서 론

인터넷의 발전을 통해 대량의 디지털 정보를 활용할 수 있는 기반이 형성됨에 따라 다양한 컨텐츠들이 인터넷 환경에서 이용 가능한 디지털 형태로 제작되어 활발하게 유통되고 있다. 그러나 디지털 컨텐츠의 특성상 불법 복제와 유통으로 인한 저작권 침해 문제가 발생하였고, 이를 해결하기 위해 DRM(Digital Rights Management) 기술을 이용하게 되었다. DRM은 저작권 보호 기술로 암호화 기술을 이용하여 허가되지 않은 사용자로부터 디지털 컨텐츠를 안전하게 보호함으로써 저작권 관련 당사자의 권리 및 이익을 지속적으로 보호 및 관리하는 저작권 관리 기술로 정의할 수 있다[1][2].

DRM은 컨텐츠 저작권 보호 기술에서 활용 가치를 인정받고 있고, 그 개발도 MS[3], Intertrust[4] 등 다수의 업체에서 활발히 진행되고 있다. 그러나 기존의 DRM은 사용자의 프라이버시 보호가 저작권 보호에 직접적으로 필요하지 않는다는 이유로 사용자의 프라이버시 보호에 대해서는 고려하지 않았다. 이러한 영향으로 라이센스 발급시의 사용자 인증과 컨텐츠의 불법 사용 감시를 위한 사용내역 보고 과정에서 사용자 정보가 유출되는 문제점이 발생하였고, 이로 인해 사용자 프라이버시 침해 문제가 발생하게 되었다[5].

† 학생회원 : 고려대학교 컴퓨터학과
happy@korea.ac.kr
†† 종신회원 : 고려대학교 컴퓨터학과 교수
tykim@netlab.korea.ac.kr
논문접수 : 2002년 9월 26일
심사완료 : 2002년 12월 18일

사용자 프라이버시 침해는 사용자의 동의나 인가를 받지 않은 상태에서 한 개인의 정보를 수집하고, 인증에 필요한 정보 외에 개인을 식별할 수 있는 불필요한 정보를 수집한다든지 혹은 정보 이용에 대한 충분한 동의나 사전인지 없이 이를 사용하거나, 저장된 정보를 무단으로 공개하는 것으로 인해 발생한다. 최근 스팸 메일 등과 같은 여러 피해사례가 높아짐에 따라 사용자 프라이버시에 대해 사용자들이 민감하게 느끼게 되면서 소비자들 사이에 프라이버시 침해에 대한 인식이 증가하고 있다. 이에 따라 DRM에서도 기존의 저작권 보호뿐만 아니라 사용자의 프라이버시 보호가 DRM 분야의 새로운 연구 과제가 되고 있다.

DRM에서 프라이버시 강화 기술은 암호화(*encryption*)와 익명성(anonymity)이다[5][6]. 본 논문에서는 DRM에서 사용자의 프라이버시 보호를 제공하면서 컨텐츠 제작자의 저작권을 보호하는 사용자 프라이버시 보호 측면이 강조된 라이센스 관리 프로토콜을 제안한다. 제안한 프로토콜은 *KryptoKnight*[7] 인증 방식을 변형하여 만든 임시 ID인 *TID*와 *token*과 같은 사용자를 인증할 수 있는 대체 식별 정보를 이용하여 인증함으로써 사용자의 실제 정보를 노출시키지 않아 사용자의 익명성을 보장한다. 또한 라이센스 발급을 위한 사용자 인증시 최소의 식별 정보만을 공개하여 사용자 정보의 유출을 최소화한다. 그리고 사용 내역 정보의 암호화를 통해 사용자의 라이센스 사용 정보가 지정된 서버이외에는 유출되지 않도록 방지하여 사용자의 프라이버시를 보호하고 기존의 지불 후 다운로딩 방식 외에 후지불 및 종량제 등과 같은 다양한 지불방식을 지원한다. 또한 라이센스의 불법 복제 방지를 통해 기존 DRM에서와 마찬가지로 컨텐츠 제작자의 저작권 보호를 제공한다.

본 논문의 구성은 다음과 같다. 2장에서는 현재의 DRM 기술 현황에 대해 소개하고, 3장에서는 논문에서 제안한 프로토콜이 적용될 시스템 기반 구조에 대해 정의한다. 4장에서는 논문에서 제안한 프라이버시 보호를 제공하는 라이센스 관리 프로토콜을 제안하고 5장에서는 기존의 DRM 기술과 제안한 프로토콜을 비교 분석한다. 마지막으로 6장에서는 결론 및 향후 연구 방향을 제시한다.

2. 관련 연구

본 장에서는 대표적인 DRM 개발업체인 Microsoft와 Interturst의 특징을 기술한다.

2.1 Microsoft

Microsoft의 WMRM(Windows Media Rights Manager)는 컨텐츠 제공자와 소비자들에게 디지털 미디어 파일을 안전하게 분배하는 종단간(end-to-end) DRM 시스템이다[3].

WMRM의 Rights Manager는 컨텐츠 제공자에게 인터넷 상에서 암호화된 파일 형식으로 보호된 음악, 비디오 등의 미디어를 배달한다. WMRM에서 각각의 서버 또는 클라이언트 인스턴스들은 개인화(Indivisualization) 과정을 통해 키 쌍을 할당받게 되며, 크래킹되었거나 안전하지 않다고 판단되는 인스턴스들에 대해서는 인증서 취소목록을 이용하여 서비스 대상에서 제외시키게 된다. 인증서 취소목록은 MS의 웹사이트를 통해 배포된다. 키는 라이센스에 포함되고, 라이센스와 컨텐츠는 분리되어 분배된다. 그림 1은 WMRM에서 라이센스 획득 단계를 나타낸다.

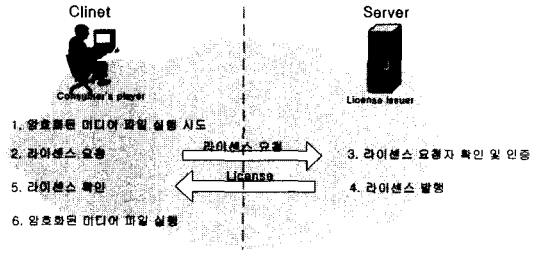

그림 1 라이센스 획득 단계

Client가 패키징되어 보호된 컨텐츠를 실행시키면 player는 License Server에 라이센스를 요청한다. Server는 라이센스 요청에 대해 사용자의 인증과 지불여부를 확인한 후에 라이센스를 발행한다. 서버는 라이센스 발행 후 라이센스를 client의 player에 전송한다. player는 서버에서 전송받은 라이센스를 확인한 후 사용규칙에 따라 컨텐츠를 실행한다.

MS WMRM의 경우 *Key ID*와 *Key seed*를 결합하여 컨텐츠 암호화 키를 생성하는데, *Key ID*는 컨텐츠 헤더에 포함되어 컨텐츠와 함께 패키징되어 배포되고, *Key seed*는 클리어링하우스에 저장되어 관리된다. 복호화키를 생성하기 위해서는 컨텐츠에 포함되어 있는 *Key ID*와 서버가 관리하는 *Key seed*가 필요하다.

MS의 WMRM은 윈도우미디어플레이에 탑재되어 널리 사용되지만, 동적변환변경에 제한적이고 윈도우미디어플레이어에만 적용되어 다양한 파일 형식을 지원하지 못한다. 그리고 라이센스를 발급 받기 위한 그림 1의 2와 3 단계의 인증단계에서 특정한 보호 기술 없이 사용자를 인증하여 사용자 ID나 전자우편 주소와 같은 사용

자정보가 유출된다.

2.2 Intertrust

Intertrust의 DRM은 컨텐츠의 보호를 위한 암호화/복호화, 컨텐츠의 사용 규칙, 사용내역 기록 및 수집, 그리고 과금 체계에 대한 지원이 이루어지고 있고, Superdistribution[8]을 실현하였으며, 사용자의 컴퓨터에서 컨텐츠를 사용하는 시점에서 거래를 체결하도록 하여 신용카드나 전자 화폐등의 결제 방식을 이용하도록 하였다[4]. Intertrust DRM의 서비스 흐름도는 그림 2에서 나타낸다.

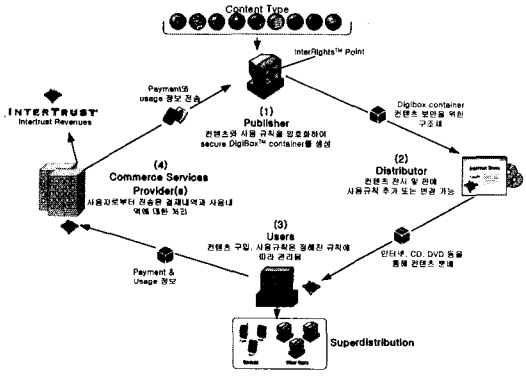

그림 2 Intertrust DRM 흐름도

Interturst의 DRM 기술은 다음과 같은 요소로 구성되었다. 첫째, InterRight Point는 Intertrust 구조의 핵심 구조로 사용자의 컴퓨터와 서버의 MetaUtility 내에서 동작하도록 한다. 둘째, DigiBox® Container는 암호화된 컨텐츠와 사용 규칙 전송을 탑재하게 된다. 셋째, 사용 규칙은 가격, 결재 방법, 재생, 프린트, 복사, 저장, superdistribution 등에 관한 규칙으로 DigiBox에 탑재되어 있다. 넷째, 거래 승인(Transaction Authority) 프레임워크는 컨텐츠의 사용 내역이나 과금 내역 등의 정보를 처리하도록 한다.

Interturst에서 라이센스 획득 방법은 다른 방식들과 다르게 이용자의 컴퓨터에 설치된 라이센스 관리자를 통하여 라이센스를 획득한다. 이 방식이 가능하기 위해서는 오프라인 상태에서도 가능한 지불수단이 존재하여야 한다.

Intertrust DRM은 그림 2의 (4)와 (1) 사이의 사용내역 전송과정에서 평문 상태로 Payment와 Usage 정보를 전달하기 때문에 사용자의 Payment와 Usage 정보가 노출되어 이로 인한 사용자 정보 유출로 사용자의 프라이버시 침해 문제가 발생할 수 있다.

3. DRM 시스템의 구성

본 장에서는 논문에서 라이센스 분배를 위해 적용할 DRM 시스템과 라이센스 에이전트(*LA*, License Agent), 그리고 라이센스의 구조에 대해 기술한다.

3.1 시스템 모델

DRM 시스템은 디지털 컨텐츠 보호와 사용 규칙 관리 및 과금 체계 관리 구조로 구성된다. 본 논문에서 제안하는 라이센스 인증 및 분배를 위한 DRM 시스템 모델은 그림 3에서 나타낸다.

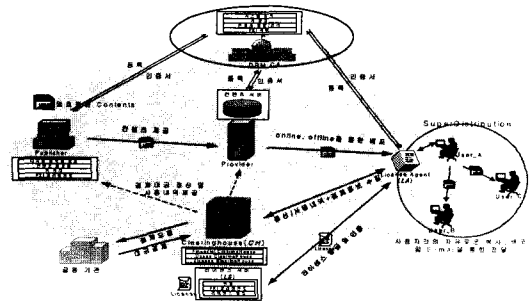

그림 3 DRM 시스템 모델

각각의 참여자는 DRM CA에 공개키를 등록하고 인증서를 발급 받는다. 컨텐츠 출판업자는 컨텐츠 제공자에게 컨텐츠를 암호화하여 전송한다. 사용자는 컨텐츠를 제공자의 웹 서버에서 제공받으며, 제공받은 컨텐츠는 암호화되어 있기 때문에 라이센스 없이는 사용할 수 없다. 사용자는 *LA*를 이용하여 지불 또는 사용 계약 체결 후에 라이센스 서버로부터 라이센스를 발급 받아 컨텐츠를 사용할 수 있다. 클리어링하우스(*CH*, Clearing-house)는 *LA*로부터 컨텐츠 사용 정보와 결제 정보 등을 수집하고 라이센스를 관리하며 사용자의 라이센스를 인증하는 역할을 한다. 암호화된 컨텐츠는 라이센스와 따로 제공되기 때문에 누구에게나 배포할 수 있으나, 라이센스 없이는 사용이 불가능하여 컨텐츠를 다른 사용자로부터 전달받은 사용자는 반드시 지불 또는 사용 계약 체결 후에 라이센스 서버로부터 라이센스를 전송 받아야만 컨텐츠를 사용할 수 있다.

3.2 LA의 구조

*LA*는 사용자 PC에 상주하는 DRM Client로써 라이센스 서버로부터 라이센스를 발급 받고, 소유하고 있는 라이센스를 인증 받는 일을 대행하며, 컨텐츠 사용현황

을 모니터링 하여 클리어링하우스에 보고한다. *LA*는 사용자 인증과 컨텐츠 접근 제어, 컨텐츠 복호화, 정산, 리포팅의 모듈로 이루어져 있다. *LA*에 대한 흐름도는 그림 4에서 나타낸다.

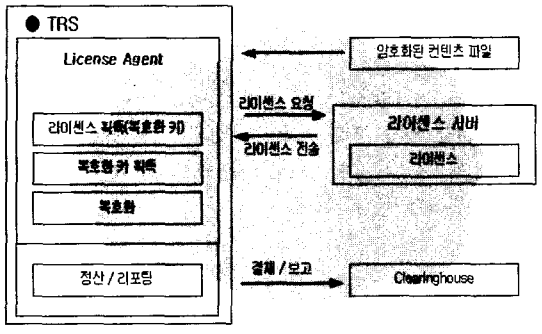

그림 4 LA의 흐름도

*LA*는 컨텐츠 제공자로부터 암호화된 컨텐츠를 다운로드한 다음 지불 또는 사용 계약 체결 후에 라이센스를 획득한다. 라이센스 획득 후 컨텐츠를 실행할 경우 *LA*는 라이센스 서버에서 획득한 라이센스를 라이센스 클리어링하우스로부터 인증 받은 후에 컨텐츠 복호화 키를 획득하여 컨텐츠를 실행한다. 또한 사용자의 사용 현황을 기록하여 클리어링하우스에 보고한다.

3.3 라이센스 구조

DRM 시스템에서 라이센스는 컨텐츠를 패키징할 때 라이센스를 컨텐츠에 포함하여 전달하는 방식과, 컨텐츠와 라이센스를 독립적으로 분리하여 컨텐츠 사용요청이 있을 시에 지불 과정을 거쳐 따로 전달하는 방식으로 나누어진다. 전자의 경우 컨텐츠가 컨텐츠 사용자에게 다운로드될 때 동적으로 라이센스를 포함하여 전달하게 되어 이후 DRM 서버와 더 이상의 처리 없이 컨텐츠를 사용하게 되며, 후자의 경우 패키징된 컨텐츠는 누구에게나 똑같은 형태로 제공되고, 컨텐츠 사용요청 시 라이센스 발급을 위한 별도의 처리가 요구된다. 본 논문에서는 후자의 방법을 사용한다.

라이센스는 라이센스 일련 번호 *sn*, 라이센스의 하드웨어 바인딩을 위한 정보인 $KID=H(DID \| LSID)$, 라이센스가 발행된 시간을 나타내는 라이센스 서버의 일자 *date*, 사용 규칙인 *Usage rule*, 그리고 기타 정보인 *other_data*를 가지고 있다. 이러한 파라미터들은 해수함수 *H*로 처리되고 라이센스 서버의 서명으로 이루어진다. *KID*에서 *DID*는 사용자의 하드웨어 장치 ID이고, *LSID*는 라이센스 서버의 ID이다.

License = {sn, KID, date, Usage rule, other_data, SigLS(H(sn, KID, date, Usage rule, other_data))}

4. 사용자의 프라이버시 보호를 제공하는 라이센스 관리 프로토콜

본 장에서는 DRM에서 사용자의 프라이버시 보호를 제공하는 라이센스 관리 프로토콜을 제안한다. 라이센스 관리 프로토콜은 라이센스 분배 단계와 사용내역 보고 단계로 구분된다.

4.1 라이센스 분배 단계

본 논문에서 라이센스 분배 단계는 라이센스 획득과 인증 프로토콜로 나누어진다. 사용자는 라이센스 서버와 라이센스 획득 프로토콜을 수행시킴으로써 라이센스를 획득한다. 라이센스를 획득한 다음에 사용자는 라이센스를 사용하기 위해 라이센스 인증 단계에서 소유하고 있는 라이센스를 라이센스 클리어링하우스에 전송한다. 라이센스 클리어링하우스는 라이센스의 불법적인 수정과 위조 여부를 검증하고 사용자가 라이센스의 정당한 소유주임을 확인하기 위한 라이센스 인증 프로토콜을 수행한다. 만약 라이센스의 검증이 성공적이면, 컨텐츠 복호화 키를 생성할 수 있는 키 값을 전달하고 라이센스의 사용 규칙에 따라 서비스를 제공한다.

제안하는 프로토콜에서 *U*는 사용자(User), *LA*는 라이센스 에이전트(Licese Agent), *LS*는 라이센스 서버(License Server), *CH*는 클리어링하우스(Clearing-house), *L/C*는 라이센스 클리어링하우스(License Clearinghouse)를 의미한다. 각 참여자들 간의 세션키 설정은 *DH* 알고리즘을 타원곡선으로 변환한 *ECDH(Elliptic Curve Diffie-Hellman)*[9] 키 설정 방식을 사용한다[8].

본 논문은 익명성을 제공하기 위한 프로토콜을 위해 다음과 같은 가정을 한다.

- 각 참여자는 프로토콜에서 사용되는 알고리즘을 알고 있다.
- 각 참여자는 PKI(Public Key Infrastructure)[10][11]를 통해 인증된 공개키-개인키 쌍과 공개키 인증서를 소지하고 있다.
- *LA*는 TRS(Tamper Resistance Software)[12]로 보호된다.
- 사용자는 컨텐츠에 대한 지불을 마친 상태이고, *LS*는 지불 과정을 통해 사용자 *U*의 식별 정보 및 장치 ID인 *DID*를 소유하고 있다.
- 라이센스 인증 프로토콜에서 *LA*와 *L/C*는 공유하는 세션키를 가지고 있다.

4.1.1 라이센스 획득 프로토콜

그림 5는 사용자 U의 장치에 LA가 설치된 다음 사용자가 다운로드한 컨텐츠에 대해 사용료 지불 또는 사용 계약 체결 후에 LS에 라이센스를 요청하여 획득하는 프로토콜이다.

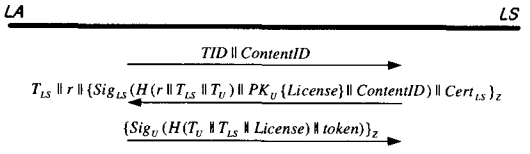

그림 5 라이센스 획득 프로토콜

프로토콜이 시작되면 사용자는 임시 비밀키 r_U를 선택하고 $ECDH$ 매개변수 G, n과 함께 키 설정용 임시 공개키 $T_U = r_U G$를 계산하고 임시 ID인 TID를 만들어 라이센스를 받기 원하는 해당 컨텐츠에 대한 $ContentID$와 전송한다. TID는 $KryptoKnight$[7]의 인증기법을 변형한 $PK_{LS}(T_U, T_U \oplus DID)$로 사용자가 키 설정용 임시 공개키와 사용자의 장치 ID인 DID를 배타적논리합(XOR)한 결과를 LS의 공개키로 암호화한 값이다. TID는 임시 ID로 사용자의 실제 정보를 노출시키지 않기 때문에 사용자의 익명성을 보장하고 LS의 공개키로 암호화하였기 때문에 LS만이 복호화하여 사용자의 DID를 확인하여 인증할 수 있다. 따라서 사용자 정보가 노출되지 않고 만약 정보가 노출되어도 사용자에 대한 정보를 정확히 알 수 없기 때문에 누구의 정보인지 알 수 없다.

두 번째 메시지에서 LS는 TID에서 DID를 추출한 다음 일치하는 DID를 찾아서 U가 등록된 사용자인지 확인하고, 계약사항을 확인한 후 해당하는 라이센스를 발급한다. LS는 임시 비밀키 r_{LS}를 선택하고 키 설정용 임시공개키 $T_{LS} = r_{LS}G$를 계산한 후 난수 r을 생성하고 TID에서 추출한 U의 임시 공개키 T_U를 이용하여 LA와 공유하는 세션키 $Z = H(hr_{LS}T_U \| r)$를 생성한다. 그리고 $H(\text{개}\|T_{LS}\|T_U)$, 사용자의 공개키 PK_U로 암호화한 $License$, 사용자가 구매한 컨텐츠에 대한 $ContentID$를 자신의 서명용 개인키로 서명하여 자신의 인증서와 같이 세션키로 암호화해서 키생성용 임시 공개키 T_{LS}, 난수 r과 함께 LA에 전송한다. LA는 전송받은 T_{LS}를 이용하여 세션키 $Z = H(hr_U T_{LS} \| r)$을 계산한 후 메시지를 복호하고 인증서에 있는 서명 검증용 공개키를 이용하여 서명검증을 한 다음 자신이 보낸 T_U와 전송받은 T_{LS}, r을 해쉬처리하여 $H(\text{개}\|T_{LS}\|T_U)$ 값과 비교해서 전송받은 정보가 일치하는지를 확인한다.

라이센스를 획득한 사용자의 LA는 $Sig_U((H(T_U \| T_{LS} \| License) \| token)$을 세션키 Z로 암호화하여 LS에 보낸다. $token=(H(DID \| License))$은 후에 라이센스 인증에 대한 확인을 위해 LS에 저장된다.

4.1.2 라이센스 인증 프로토콜

라이센스를 획득한 사용자가 컨텐츠를 복호하기 위해서는 라이센스를 인증받고 컨텐츠 복호화키를 생성할 수 있는 KI를 L/C로부터 전달받아야 한다.

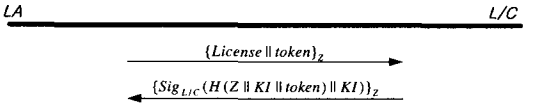

그림 6 라이센스 인증 프로토콜

프로토콜(그림 6)이 시작되면, 사용자의 장치에서 구동되는 LA는 $License$와 $token$을 세션키 Z로 암호화하여 L/C에 전송한다.

L/C는 LA로부터 전송받은 메시지를 복호하여 $License$를 확인하고 전송받은 $token$이 저장되어 있는 $token$과 일치하는지 확인한다. $token$은 사용자 장치에 대한 정보인 DID가 포함되어 있으므로, 악의적인 사용자는 라이센스를 불법적으로 획득하였더라도, 등록되어 있는 사용자의 장치 ID인 DID를 추출할 수 없기 때문에 특정 사용자 장치 외에서는 컨텐츠를 서비스 받을 수 없게 된다. $License$와 $token$을 확인하여 합법적인 사용자임이 검증되면 L/C는 $H(Z \| KI \| token)$과, 컨텐츠 복호화 키에 사용될 키 정보 KI를 서명한 후 세션키 Z로 암호화해서 LA에 전송한다.

LA는 L/C의 서명을 검증하고 자신이 가지고 있던 정보와 $H(Z \| KI \| token)$를 확인하여 L/C의 실체를 인정하고, 복호화 키 생성을 위해 필요한 키 정보 KI를 얻어낸다. LA에서 KI는 컨텐츠 복호화 키 $Key=H(ContentID \| LSID \| KI)$를 생성하여 컨텐츠를 복호화하여 사용하게 된다. KI는 컨텐츠 사용 후 삭제되어 다음 컨텐츠 사용 시 라이센스 인증 후 재 전송 받아야 한다.

라이센스 분배 단계에서 각 프로토콜의 메시지는 세션키로 암호화되어 전송되고, 임시 ID인 TID와 $token$과 같은 대체 식별 정보를 전송한다. 따라서 사용자를 직접적으로 알 수 있는 정보를 전송하지 않으므로 제

삼자에게 사용자 정보를 숨길 수 있게 된다. 또한 악의의 공격자가 메시지를 가로채더라도 사용자의 실제 정보를 노출시키지 않으므로 공격자가 사용할 수 있는 사용자 정보를 추출할 수 없게 되므로 익명성을 제공한다.

4.2 사용내역 보고 단계

*CH*는 *LA*로부터 사용자의 컨텐츠 사용 내역을 보고 받는다. 이 자료는 선지불 외에 후지불 및 종량제 등과 같은 다양한 지불 방식을 지원하기 위해 수집되어 진다. 이때 수집되는 자료는 사용자 정보가 유출될 수 있으므로 암호화하여 악의적인 참여자로부터 보호되어야 한다. 그림 7은 사용정보 보고 프로토콜을 나타낸다.

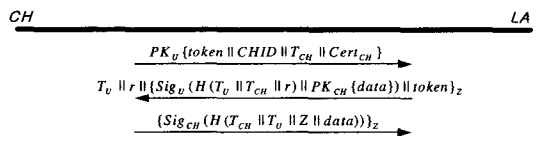

그림 7 사용정보 보고 프로토콜

프로토콜이 시작되면, *CH*는 보고 받기 원하는 라이센스에 해당하는 컨텐츠의 사용정보를 얻기 위하여 라이센스 인증단계에서 사용자로부터 전송받은 *token*과, *CH*의 식별정보 *CHID*, 세션키 설정용 임시 공개키 T_{CH}, *CH*의 인증서 $Cert_{CH}$를 사용자의 공개키 PK_U로 암호화하여 전송한다. 공개키 T_{CH}를 계산하는 것은 4.1.1에서의 방법과 같다.

*LA*는 전송 받은 메시지를 자신의 비밀키로 복호한 후 저장하고 있던 *token*과 전송 받은 *token*을 비교하여 동일할 경우 신뢰된 *CH*로부터 전송된 메시지인 것으로 확인하고 세션키 $Z = H(hr_U T_{CH} \| r)$을 계산한다. 그리고 자신의 공개키로 암호화한 컨텐츠 사용정보 데이타 *data*와 $H(T_U \| T_{CH} \| r)$을 서명용 비밀키로 서명하고 자신이 소유하고 있던 *token*과 함께 세션키 *Z*로 암호화하여 *CH*에 전송한다.

*CH*는 전송받은 T_U를 이용하여 세션키 $Z = H(hr_{CH} T_U \| r)$을 계산한 후 전송받은 메시지를 복호화한다. *CH*는 자신이 가지고 있는 정보의 해쉬값과 $H(T_U \| T_{CH} \| r)$를 확인하여 통신하는 상대방에 대한 실체 인증을 확인하고 사용자의 사용 내역에 대한 정보 *data*를 획득한다. 사용자의 사용 내역 및 사용자 정보는 세션키로 암호화하고 *CH*의 공개키 PK_{CH}로 암호화 된다. 따라서 *CH*만이 이 정보를 확인 할 수 있기 때문에 악의적인 공격자로부터 공격을 방어할 수 있다. 이것은 사용자 정보의 악의적인 유출을 막아 클리어링하우스 자신이 누출하거나 다른 용도로 사용하지 않는 한 사용자의 프라이버시를 보호할 수 있다.

*CH*는 획득한 정보들을 $H(T_{CH} \| T_U \| Z \| data)$로 처리하여 자신의 서명 비밀키로 서명하고 세션키 *Z*로 암호화하여 *LA*에 전송한다. 이것은 *CH*가 정상적으로 *data*를 전송받았다는 것을 확인하고, 후에 사용자의 정보가 누출되어 악용되었을 경우, 정보유출의 책임에 대한 부인 방지를 위해 필요하다.

5. 성능 평가 및 분석

본 장에서는 기존에 개발된 DRM 시스템 및 프로토콜과 제안한 프로토콜과의 성능을 비교 평가하고, 제안한 프로토콜의 안전성을 분석한다.

5.1 성능 평가

DRM 시스템의 궁극적인 목적은 불법적인 유통과 사용을 막아 저작자 및 모든 참여자에게 이익을 보장하는 것이다[1]. 기존 DRM 기법에서 문제가 되는 것은 라이센스를 획득하는 과정에서의 사용자 인증과, 라이센스 사용에 대한 사용내역 보고 과정에서 사용자의 정보가 유출되어 사용자의 프라이버시가 침해된다는 것이다[5]. DRM 서버가 사용자를 인증하기 위해서는 사용자 정보를 전달받아야 하고, 사용내역에 대해 보고를 받을 때도 사용자의 정보를 수집할 수 있다. 이 과정에서 기존의 DRM은 사용자 프라이버시 문제에 대해 고려하지 않았다. 따라서 별다른 조치 없이 사용자 정보를 수집하고 관리함으로 인해 사용자 정보가 유출되었다. 그러나 제안한 프로토콜은 *ECDH* 세션키와 공개키 암호화로 악의적인 공격자로부터 메시지를 보호하고, 만약 메시지가 유출되더라도 사용자 식별을 위해 임시 ID인 *TID*를 사용하였으므로 익명성이 제공되기 때문에 사용자의 프라이버시가 보호된다. 표 1은 MS-DRM의 WMRM, Intertrust의 DRM, 그리고 [13]에서 제안한 ID 기반 DRM 기법과 제안한 라이센스 분배 프로토콜의 특성을 비교하고 있다.

본 논문에서 보인 모든 기법들은 대칭키 방식을 통한 컨텐츠 암호화로 superdistribution을 지원한다. 대칭키 방식은 암호화 및 복호화키가 동일하기 때문에 컨텐츠의 superdistribution을 가능하게 한다[8]. 허가된 사용자의 공개키로 컨텐츠를 암호화하던 초기의 보호방식은 공개키 암호화 방식을 사용하기 때문에 근본적으로 superdistribution 모델을 지원할 수 없다. 키 분배 방식은 Intertrust만이 대칭키/공개키 둘 다 사용하고 나머

표 1 기존 DRM 프로토콜과 제안한 프로토콜의 특성 비교

	MS-DRM (WMRM)	Intertrust	ID 기반	제안한 기법
Superdistribution	○	○	○	○
키 분배 방식	공개키(ECC)	공개키/대칭키	공개키(ID 기반)	공개키(ECC)
라이센스 형태	컨텐츠와 분리	컨텐츠에 포함	컨텐츠에 포함	컨텐츠와 분리
키 관리 방식	Distributed key management	Server-based key management	Server-based key management	Distributed key management
하드웨어 바인딩	○	○	×	○
저작권 보호	○	○	○	○
사용자 프라이버시	×	×	×	○

○ : high, △ : middle, × : low

지는 공개키 방식만을 사용한다.

Intertrust와 ID 기반 방식은 키 관리를 라이센스 발급 서버에서 집중관리하는 서버 기반 키 관리(sever-based key management) 방식을 사용하므로 서버에서 키를 집중 관리하여 서버가 키 관리에 부담을 느낄 수 있다. 그러나 WMRM과 제안한 프로토콜은 암호화키 생성을 위해 컨텐츠에 포함되어 있는 컨텐츠 ID와 클리어링하우스가 관리하는 키가 모두 필요하다. 따라서 암호화키를 서버와 분리하여 보관하는 분산 키 관리(distributed key management)방식을 사용하므로 키 관리 서버의 부담을 줄이며 키 관리의 안정화를 가져올 수 있다. ID 기법은 하드웨어 바인딩을 제공하지 않으나 제안한 프로토콜 및 나머지 기법들은 라이센스와 컨텐츠의 키 생성시에 논문의 DID와 같은 사용자의 하드웨어 일련번호를 추출해서 하드웨어 정보에 바인딩하여 라이센스를 특정 하드웨어에서만 유효하도록 한다.

기존의 DRM 기술들은 사용자의 프라이버시 보호가 저작권 보호에 직접적으로 필요하지 않는다는 이유로 저작권자의 저작권 보호만을 중요시하고 사용자의 프라이버시 보호에 대해서는 고려하지 않았다. 따라서 저작권 보호 기능은 제공하였으나, 라이센스 발급시의 사용자 인증과 컨텐츠의 불법 사용 감시를 위한 사용내역 보고 과정에서 사용자의 정보가 유출되는 것으로 인한 사용자의 프라이버시 침해 문제를 가져오게 되었다. 그러나 제안한 프로토콜은 상호 인증을 통하여 신뢰성을 얻고, 사용자의 식별을 위해 TID를 사용하였다. 프로토콜의 TID는 실제 식별 정보를 노출시키지 않기 때문에 익명성을 제공하게 되고 $ECDH$ 세션키와 공개키 암호화로 사용자 정보의 유출을 방지하여 사용자의 프라이버시를 보호한다. 임시 ID인 TID는 사용자의 장치 ID인 DID와 키 설정용 임시 공개키 T_U를 배타적 논리합(XOR)한 값을 라이센스 서버의 공개키로 암호화한 값인 $PK_{LS}(T_U, T_U \oplus DID)$로 이 메시지를 받은 LS는 자신의 비밀키로 메시지를 복호화하여 나온 값을 키 설정용 임시 공개키로 배타적 논리합(XOR)하여 사용자의 DID를 얻어 낼 수 있다. 메시지는 LS의 공개키로 암호화되었기 때문에 LS만이 메시지를 복호화 할 수 있다. 공격자는 LS의 비밀키를 알 수 없기 때문에 사용자의 DID를 유추해 낼 수 없다. 따라서 공격자가 라이센스를 가진 사용자의 정보를 추출할 수 없기 때문에 사용자의 실제 식별 정보를 숨길 수 있게 되므로 익명성을 제공한다. 또한 사용자 정보수집에서 사용정보를 CH의 공개키로 암호화하고, 이것을 다시 세션키 Z의 암호화를 통해 사용자의 정보가 유출되지 않도록 하여 악의적인 제공자가 사용 정보를 가로채어 악용할 수 없게 하므로 사용자의 프라이버시를 보호한다.

표 2는 제안한 프로토콜을 시스템에 적용했을 경우 기존 시스템과의 특성 비교이다.

표 2 기존 DRM 시스템과 제안한 시스템의 특성 비교

	MS-DRM (WMRM)	Intertrust	ID 기반	제안한 기법
TRM	○	○	○	○
사용자설치모듈	×	○	○	○
지원 응용 프로그램	동영상/음악	다양한 응용 지원	다양한 응용 지원	다양한 응용 지원
동적변환변경	×	○	×	△
지불방식	지불후 다운로딩	지불후 다운로딩	지불후 다운로딩	선지불/후지불/종량제
네트워크비 의존성	△	○	△	×

○ : high, △ : middle, × : low

WMRM은 사용자 설치 모듈을 따로 설치하지 않고 기존의 윈도우미디어플래이어에 탑재하여 따로 DRM client 모듈을 설치하지 않아도 되지만, 윈도우미디어플레이어에서만 구동하므로 다양한 지원 응용 프로그램을 지원하지 못한다. 반면 제안한 기법을 포함한 나머지 기법들은 따로 *LA*와 같은 전용 사용자 설치 모듈을 설치하여야 하지만 특정 파일 형식에 제한을 두지 않기 때문에 다양한 응용 프로그램을 지원한다. MS와 ID 기반 기법들은 라이센스 획득시에만 인증을 하고 그 이후로는 사용자 장치에 저장되어 구동되므로, 라이센스 사용 규칙 변경 등에서 제한적이나 Interturst와 제안한 프로토콜은 온-라인 상에서 라이센스 사용할 때마다 라이센스의 인증을 받고 실시간으로 클리어링하우스에 사용자의 사용 내역을 보고하므로 동적권한변경이 가능하다. 또한 제안한 기법은 사용내역에 대한 보고로 기존의 지불후 다운로딩 방식인 선지불 방식 이외에 사용후 요금을 지불하는 후지불 및 일정액의 요금을 지불하고 그 요금만큼 컨텐츠를 사용하는 종량제 방식 등 다양한 지불방식을 지원한다. 제한한 기법은 라이센스 분배와 사용 내역 보고에서 라이센스의 인증을 강화하여 컨텐츠의 불법 사용을 방지하고, 라이센스를 따로 분배하여 사용 규칙 등을 동적으로 변경할 수 있지만 이로 인해 네트워크에 대한 의존도가 다른 기법에 비해 높다.

5.2 안전성 분석

라이센스에 있어서 문제가 되는 점은 라이센스가 불법적으로 유통되거나 라이센스 정보가 무단으로 변경 또는 손상되는 것이다. 따라서 라이센스 정보의 무단 변경을 방지하고 라이센스 정보의 무결성을 보장할 수 있어야한다. 정당하지 못한 사용자는 라이센스를 복제할 수 있다. 그러나 라이센스 획득 프로토콜에서의 서명을 생성해낼 수 없고, 사용자의 하드웨어에서 추출한 특정 장치키인 *DID*를 제시할 수 없으므로 라이센스의 복제 사용은 방지될 수 있다. 라이센스 서버 이외의 개체로부터의 라이센스의 수정과 위조는 가능하지 않다. 왜냐하면 정당하지 못한 라이센스의 사용자는 라이센스 서버의 서명을 생성해낼 수 없기 때문이다.

라이센스 획득 프로토콜에서 사용자의 인증으로 인해 유출되는 프라이버시 문제는 암호화와 익명성을 통해 보호할 수 있다[5][6]. 프로토콜의 *TID*는 사용자의 식별 정보를 대신하는 임시 ID이다. $TID = PK_{LS}(T_U, T_U \oplus DID)$는 사용자가 생성한 키 설정용 임시 공개키와 사용자의 장치 ID인 *DID*를 키 설정용 임시 공개키와 배타적논리합(XOR)한 결과를 라이센스 서버 *LS*의 공개키로 암호화한 임시 ID로 라이센스 서버만이 사용자의 *DID*를 확인하여 인증할 수 있다. *TID*에서 *DID*는 사용자를 식별할 수 있는 식별 정보 이지만, 이 정보는 단지 사용자의 장치 ID일뿐 사용자를 정확하게 나타내는 정보가 아니므로, 사용자가 등록한 신뢰된 서버만이 이 정보를 가지고 이미 저장된 정보와 비교하여 사용자가 누구인지를 식별할 수 있다. 또한 사용자의 인증서와, 서명 등을 세션키 *Z*로 암호화하여 전송하므로, 신뢰된 클리어링하우스만이 사용자가 누구인지를 알 뿐, 다른 제공자들은 알 수 없다. 따라서 제 삼자에게 사용자의 정보가 노출되지 않으므로 사용자의 익명성을 보장하여 사용자의 프라이버시를 보호한다.

본 논문에서 사용하는 *ECDH* 세션키 설정 방식에서 도메인 매개변수는 표수(characteristic)가 p인 유한체 Fq상에서 정의된 타원곡선 E와 위수가 n인 기저점 $P \in E(Fq)$로 구성된다. 실체의 개인키는 $[1, n-1]$에서 임의로 선택한 d이고, 공개키는 타원곡선의 점 $Q=dp$이며, 키 쌍은 (Q, d)가 된다. 기저점 p에 대하여 aP와 bP가 주어졌을 때 abP를 구하는 것을 타원곡선 상의 *Diffie-Hellman* 문제라 하며, 통상 유한체 상에서 정의된 *Diffie-Hellman* 문제보다 어렵다고 알려져 있어 키 비트 당 보다 많은 안정성을 보장하는 장점이 있다[14][15].

세션키 *Z*는 통신 상대자로부터 생성된 난수 r과 같이 만들어진다. 이는 이전에 사용되었던 세션키 *Z*가 재 사용되는 것을 방지하여 replay attack을 막기 위함이다. 난수 r은 세션키가 새로운 키(key freshness)임을 증명한다. 또한 세션키 생성에서 선택 변수 h는 여인자 $h=|E|/n$로 *Diffie-Hellman* 프로토콜에 대한 *small subgroup attack*[16]을 막는다.

그림 7의 사용내역 보고 프로토콜에서의 마지막 메시지인 $H(T_{CH} \| T_U \| Z \| data)$에 클리어링하우스의 서명을 하여 보내는 것은 클리어링하우스가 사용자의 사용정보를 유출하였을 경우 이것에 대한 책임 회피를 방지하기 위함이다. 사용자의 정보가 유출되는 것은 악의 적인 공격자가 가로채는 경우도 있으나 사용자와 연결된 클리어링하우스가 유출하는 경우도 있다. 이러한 것은 암호학적인 방법만으로는 사용자 정보의 유출을 막기에는 부족하다[6]. 사용자에게 라이센스를 제공하고 관리하는 클리어링하우스의 정보 유출을 막는 것은 클리어링하우스 제공자에게 법적 책임을 지우는 방법밖에 없다. 따라서 논문에서는 이러한 책임을 부가하기 위해 클리어링하우스의 서명을 통해 사용자 정보를 받았다는 확인을 해준다.

라이센스 내의 *KID*는 하드웨어 바인딩을 위해 사용자의 하드웨어 장치에서 추출한 고유 키와 라이센스 서버의 식별 정보를 해쉬처리한 키 값으로 라이센스를 하

드웨어에 바인딩하여 해당 라이센스를 특정 장치에서만 실행시킬 수 있다. 따라서 라이센스의 불법 복제를 방지하고 결과적으로 컨텐츠의 불법적인 사용을 방지할 수 있다.

프로토콜에서의 *DID*는 사용자의 하드웨어 장치 ID로 하드웨어 바인딩을 위해 추출한다. 이것은 사용자 장치의 특정 정보와 바인딩 되어 저장되므로 사용자 인증키의 물리적인 복사에 의한 사용자 인증은 불가능하다. 또한 사용자 인증키와 *PC* 고유정보의 재암호화로 타 사용자의 불법적인 인증을 원천적으로 봉쇄한다.

복호화키를 생성하기 위해서는 사용자의 장치에 저장되어 있는 *ContentID*와 클리어링하우스에서 관리하는 *KI*가 모두 필요하므로, 암호화키를 분산 관리하여 키 관리의 안정화를 기할 수 있다.

6. 결론 및 향후 연구

DRM은 컨텐츠 저작권 보호 기술에서 그 활용 가치를 인정받고 있고, 그 개발도 활발히 진행되어 컨텐츠 제공자의 저작권을 보호하였다. 그러나 라이센스 인증과 사용 내역 추적 시에 사용자의 정보가 유출되는 문제점을 가지고 있다. 본 논문에서는 암호화와 임시 식별정보를 이용하여 익명성을 제공하는 사용자의 프라이버시 보호 측면이 강조된 라이센스 관리 프로토콜을 제안하였다. 제안한 프로토콜은 사용자 프라이버시 보호뿐만 아니라 기존의 기법에서 제공하던 라이센스의 불법 복제 및 사용 방지에 대해서도 제공한다. 또한 사용 내역 보고로 라이센스의 동적 변환이 가능하도록 하고 다양한 지불 방식을 지원한다.

향후 연구 과제로는 확장된 비즈니스 모델에서도 사용자의 프라이버시를 보호할 수 있고, 사용자의 편의성을 제공하는 컨텐츠 저작권 보호기술의 개발에 대한 연구를 할 것이다.

참 고 문 헌

[1] J. Dubl, "Digital Rights Management : A Definition," IDC, 2001.

[2] J. Dubl, S. Kevorkian, "Understanding DRM System : An IDC White paper", IDC, 2001.

[3] Microsoft : http://www.microsoft.com/windows/windowsmedia/drm.asp

[4] Intertrust : http://www.interturst.com

[5] P. Vora, D. Reynolds, L. Dickinson, J. Erickson, D. Banks, "Privacy and Digital Rights Management," A position paper for the W3C Workshop on Digital Rights Management, January 2001.

[6] J. Feigenbaum, M. J. Freedman, T. Sander, A. Shostack, "Privacy Engineering for Digital Rights Management Systems," Workshop on Security and Privacy in Digital Rights Management, November 2001.

[7] R. Molva, G. Tsudik, E. Van Herreweghen, S. Zatti, "KryptoKnight Authentication and Key Distribution System," Proceeding of ESORICS'92, November 1992.

[8] Brd J. Cox, "Superdistribution: Objects As Property on the Electronic Frontier," Addision-Wesley, May 1996.

[9] ANSI X9.63 : Public key cryptography for the financial services industry : Key agreement and key transport using elliptic curve cryptography, ANSI, X9.63-199x draft, January 1999.

[10] ITU-T Recommendation X.509: Information Technology-Open Systems Inter- connection-The Directory: Authentication Framework.

[11] Stefan Brands, "Rethinking Public Key Infrastructures and Digital Certificates; Building in Privacy," MIT Press, August 2000.

[12] Aucsmith, D., "Tamper Resistant Software: An Implementation," in Anderson, R., ed., Information Hiding, First International Workshop, Cambridge, UK., Springer- Verlag Lecture Notes in Computer Science, Vol. 1174, pp. 317-333, May 1996.

[13] J.M.Jeon, S.J.Park, B.C.Kim, D.H.Won, "DRM Security Framework - ID Base Approach for Content Super-Distributions," IIIS, July 2001.

[14] Certicom Corp., "Remarks on the security of the Elliptic curve cryptosystem," 2000., http://www.certicom.com

[15] Julio Lopez and Ricardo Dahab, "Performance of Elliptic Curve Cryptosystems," Technical report IC-00-08, 2000., http://www.dcc.unicamp.br/ic-main/publications-e.html

[16] C. H. Lim and P. J. Lee., "A Key Recovery Attack on Discrete Log-based Schemes Using a Prime Order Subgroup," In Advances in Cryptology: Crypto '97m B. S. Kaliski, Jr., Ed., Lecture Notes in Computer Science 1294, Springer-Varlag, pp. 249-263, 1997.

[17] Schneier, Bruce., Applied Cryptography, Second Edition, Essential reference for cryptographic engineers by the foremost pundit in the field, Wiley, 1996.

[18] A. Menezes, P. van Oorschot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997.

박 복 녕

2001년 한성대학교 멀티미디어정보처리학과 학사. 2001년~현재 고려대학교 컴퓨터학과 석·박사 통합과정 재학. 관심분야는 DRM, 암호 프로토콜, 이동 통신 보안, Ad-Hoc 네트워크

김 태 윤

1981년 고려대학교 산업공학과 학사 1983년 미국 Wayne State University 전산과학과 석사. 1987년 미국 Auburn University 전산과학과 박사. 1988년~2002년 고려대학교 컴퓨터학과 교수. 관심분야는 전자상거래, 컴퓨터 네트워크 EDI, 이동통신, 멀티미디어 등