[UCI]I804:24011-200000232722

*2006*年 *8*月

博士學位論文

P2P 환경에서 개선된 콘텐츠 보안 시스템 설계 및 구현

朝 鮮 大 學 校 大 學 院

컴 퓨 터 工 學 科

李 汀 祺

*P2P*환경에서 개선된 콘텐츠 보안 시스템 설계 및 구현

Design and Implementation of the Improved Contents Security System in the P2P Environments

*2006*年 *8*月 日

朝 鮮 大 學 校 大 學 院

컴 퓨 터 工 學 科

李 汀 祺

*P2P*환경에서 개선된 콘텐츠 보안 시스템 설계 및 구현

Design and Implementation of the Improved Contents Security System in the P2P Environments

指導敎授 李 俊

이 論文을 工學博士學位 論文으로 提出함

*2006*年 *4*月

朝 鮮 大 學 校 大 學 院

컴 퓨 터 工 學 科

李 汀 祺

이정기의 工學博士 學位論文을 認准함

委 員 長　朝鮮大學校 敎授_____印

委　　員　朝鮮大學校 敎授_____印

委　　員　朝鮮大學校 敎授_____印

委　　員　朝鮮大學校 敎授_____印

委　　員　朝鮮大學校 敎授_____印

*2006*年　*6*月

朝鮮大學校　大學院

목 차

ABSTRACT

제 *1* 장 서 론 ····· 1

제 *2* 장 연구의 이론적 배경 ····· 4

제 *1* 절 *P2P* 및 *DRM* 기술에 관한 고찰 ····· 4

제 *2* 절 *DRM* 모델 연구 ····· 12

제 *3* 절 *DRM* 핵심 기술 ····· 16

1. 암호화 알고리즘 ····· 16

2. RSA 알고리즘 ····· 16

3. SEED 알고리즘 ····· 23

4. AES 알고리즘 ····· 31

5. SHA 알고리즘 ····· 34

제 *4* 절 키 관리 기술 ····· 41

제 *3* 장 *P2P*기반 개선된 *DRM* 시스템 설계 ····· 49

제 *1* 절 *P2P*기반 *DRM* 기능의 고려사항 ····· 49

제 *2* 절 기존의 *DRM* 솔루션 분석 ····· 51

제 *3* 절 기존 *DRM* 모델의 한계 ····· 56

제 *4* 절 개선된 *DRM*시스템 설계 ····· 57

1. Clearinghouse 설계 ····· 59

2. Client 설계 ····· 61

제 4 장 시스템 구현 및 분석 ····· 68
제 1 절 DRM 시스템 구현 ····· 68
제 2 절 성능 평가 ····· 70
제 3 절 DRM 시스템 분석 ····· 73

제 5 장 결 론 ····· 75

[참 고 문 헌] ····· 77

도 목 차

[그림 1]. 콘텐츠 암호기술의 기본적인 흐름도 ····· 8
[그림 2]. DRM 프로세스 흐름도 ····· 10
[그림 3]. InterTrust사의 DRM 시스템 ····· 13
[그림 4]. Microsoft사의 DRM 라이선스 획득과정 ····· 15
[그림 5]. RSA 암호시스템의 전체 블럭도 ····· 22
[그림 6]. Feistel cipher 구조 ····· 24
[그림 7]. SEED 구조도 ····· 25
[그림 8]. F 함수의 구조 ····· 27
[그림 9]. G 함수의 구조 ····· 28
[그림 10]. SHA-1의 기본동작 ····· 38
[그림 11]. 단일 블록의 SHA-1처리에서 80-단어 입력생성 ····· 39
[그림 12]. DOI 등록 프로세서 ····· 47
[그림 13]. DOI 변환 프로세서 ····· 48
[그림 14]. P2P 기반 DRM 시스템 설계 ····· 59
[그림 15]. Clearinghouse 설계 ····· 61
[그림 16]. Client 설계 ····· 63
[그림 17]. 인증 프로토콜 ····· 65
[그림 18]. 라이선스 프로토콜 ····· 67
[그림 19]. 클라이언트 환경 ····· 69
[그림 20]. 평균 대역폭 점유율 ····· 70
[그림 21]. 암호화 시간과 재생 지연시간 비교 ····· 72

표 목 차

[표 1]. 선형 시스톨릭 어레이의 처리기수와 처리시간 ····· 19

[표 2]. G함수의 라운드 상수 ····· 30

[표 3]. SHA 라운드 상수 ····· 36

[표 4]. 암호화키의 길이에 대한 해독 소요시간 ····· 42

[표 5]. DRM 시스템 기술 분석 비교 ····· 52

[표 6]. 기존 DRM 시스템과 개선된 DRM 시스템 ····· 74

ABSTRACT

Design and Implementation of the Improved Contents Security System in the P2P Environments

Lee, Jeong-Gi
Advisor : Prof. Lee, Joon, Ph.D.
Dept. of Computer Engineering
Graduate School of Chosun University

The word P2P implies significant changes in current business dynamics. The P2P service enables individuals to be connected to the Internet for provision information directly and even downloads from without the conventional method of passing through search engines. This can be utilized to extend the path of retrieving information from limited web sites to personal and enterprise databases. That is, it is now possible for individuals to manage their own information on a national or global scope, share various information with other members, form communities of users interested in sharing homogeneous information, and utilize remote conference and remote education using groupware.

To incorporate protection to digital contents while they are distributed freely via networking, Microsoft, IntelTrust, and other firms have attempted to apply DRM. However, due to stagnation in the digital music and electronic document market, which constitute the initial market, increased complaints for content-owner oriented DRM technology, and incompatible DRM technologies between DRM providers led to the use of DRM only in very limited areas. Not only expression of content, but production,

distribution, and transmission methods of consumption structure are all being developed unilaterally without much consensus. Thus content providers, managers, and users are faced with the task of solving the problem of compatibility.

This thesis aims to design and implement a DRM system that would protect digital contents exchanged in P2P networks while retaining the freedom to exchange desired contents to and from networked computers.

The copyright protection model suggested in the thesis uses relevant DRM technology to protect digital contents that are being distributed through P2P networks. The implemented DRM technology retains the characteristics of the network - free distribution and file sharing - and is structured to be cooperative with P2P technology.
It also provides content owner verification, content authentication structure, content protection structure, indirect source acquisition effect, and indirect fee-charge effect which are the mandatory criterion for fee based models.

To sustain the advantages of P2P technology, the model is structured to have paid services and free services run parallel to each other rather than implementing only paid service. The problems of current paid video and music services lie on the fact that the DRM server must have a constant connection with users. Realtime streaming download format induces increased traffic to DRM servers and content cutoffs. The suggested method acquiesces downloads while enforcing connection to DRM for authentication, decoding, and license downloading for cases of charged contents. This can retain the merits of P2P technology to its full extent while solving the biggest issue addressed: copyright problems.

제 1 장 서 론

컴퓨터와 인터넷의 발전이 만들어 가는 정보화 사회는 우리의 생활 패턴을 하루가 다르게 변화시키고 있으며 정보의 전달 방식의 변화는 이러한 변혁을 주도하고 있다. 과거에는 정보전달이 TV, 라디오, 신문, 잡지 등과 같은 다양한 매체를 통하여 단 방향으로 비교적 느린 속도로 전달되었던 반면, 인터넷 시대에서는 복합적인 정보가 단일화된 매체를 통하여 매우 빠른 속도로 상호 교환되고 있다.

특히 정보 전달 매체 중 인터넷을 이용한 전자상거래의 활성화에 따라 디지털 상품의 유통이 점차 확대되고 있다[1][2]. 그 가운데 가장 활성화된 음악 온라인 서비스를 들 수 있는데 해외에서는 냅스터, 국내에서는 소리 바다가 그 예라고 할 수 있다.

그러나 이와 같이 음악 온라인 서비스를 주도하는 P2P 기술도 저작권 보호가 되지 않아 음악 권리자들로부터 집중적인 법적 공격을 받아 결국 냅스터는 서비스가 중단되었으며, 소리바다는 법적 공방으로 유료화로 진행되었다. 이러한 상황 속에서 유료화 또는 저작권 보호 모델은 P2P 기술의 가장 중요한 이슈가 되고 있으나 적절한 해결 방법을 찾지 못하고 있다[3].

P2P는 현재 활발히 진행되고 있는 비즈니스의 중요한 변화를 표현하는 함축적 의미를 가진다. P2P 서비스는 인터넷상의 정보를 검색 엔진을 거쳐 찾아야 하는 기존 방식과 달리 인터넷에 연결된 모든 개인 컴퓨터로부터 직접 정보를 제공받고 검색은 물론 내려 받기까지 할 수 있는 서비스이다. 이는 웹 사이트에 한정돼 있던 정보 추출 경로를 개인이나 회사가 운영하는 데이터베이스까지 확대할 수 있다. 즉 자신의 정보를 전국적 혹은 세계적으로 관리 운용하며, 회원 상호간의 다양한 정보 공유뿐만 아니라 동일한 정보를 공유하고자 하는 회원 간의 커뮤니티 형성이 가능하고, 그룹웨어로서 역할을 통해 원격회의, 원격교육 등이 가능하다[1][4].

P2P 기술이 인터넷에 연결되어 있는 모든 컴퓨터를 네트워크화함으로써 방대한 콘텐츠 저장 창고 역할을 하는 반면 그것은 역으로 그 만큼의 저작권 침

해의 문제를 안고 있기도 하다. 이에 따라 많은 콘텐츠 저작권자들이 P2P 솔루션을 기반으로 하는 온라인 서비스업체를 상대로 저작권 소송을 벌이고 있다. PC 통신이나 HTTP를 기반으로 하는 서비스 제공자의 경우 중앙에서 서버를 관리하고 통제함으로써 저작권 위반가능성이 있는 파일들에 대해 제재와 통제가 용이하지 않다. 또한 저작권문제 이전에 개개인의 컴퓨터를 통제하는 데서 오는 프라이버시 침해 문제 등의 유발 가능성이 있어 저작권 관리는 쉽지 않을 전망이다[6]. 그러나 P2P가 미래의 비즈니스 모델로서 기대됨에 따라 바이러스와 해킹 등의 보안 문제와 함께 저작권 문제는 필연적으로 해결해야 할 사안이다.

디지털 콘텐츠가 네트워크를 통해 자유롭게 유통되면서 불법 복제로 인한 저작권 침해를 방지하기 위해 Microsoft, IntelTrust 등의 업체가 DRM(Digital Rights Management)을 시도하고 있으나 초기 시장을 형성하던 디지털 음악 및 전자문서 시장의 침체와 저작권자 위주의 DRM 기술에 대한 사용자의 불만 증대, 상이한 DRM 기술들 간의 상호 호환성 결여 등의 이유로 제한된 범위에서만 사용되어 왔다. 콘텐츠의 표현 방식은 물론 제작 및 배포, 소비 구조 전송 방식, 단말 방식 등이 모두 독립적으로 개발되고 있어 콘텐츠 제공자, 관리자, 사용자 등 모두에게 호환성 문제 또는 해결하는 과제로 남고 있다.

본 논문에서는 P2P의 가장 큰 장점인 네트워크가 연결된 모든 컴퓨터에서 원하는 콘텐츠를 자유롭게 다운로드 받을 수 있는 기술을 유지하고 현재 가장 이슈가 되고 있는 콘텐츠 저작권 문제를 해결하는 방안으로 P2P상에서 디지털 콘텐츠 보호를 위한 DRM시스템을 설계 및 구현 하고자 한다.

본 논문에서 제시된 저작권 보호 모델은 P2P를 통해 유통되고 있는 디지털 콘텐츠의 저작권 보호를 위해 적절한 DRM 기술을 이용하고 본 논문에서 적용된 DRM 기술은 자유 배포 및 정보 공유의 특징을 살리며 아울러 P2P 기술과 가장 협력이 용이한 구조를 갖도록 한다. 또한, 유료화의 필수 사항인 콘텐츠 권리자 확인구조, 콘텐츠 정품 인증구조, 콘텐츠보호 구조, 효과 및 간접 유료화 효과를 제공하도록 하고 P2P 기술의 경쟁력 유지를 위해 완전 유료화 모델

이 아닌 유료와 무료 콘텐츠 서비스를 병행하는 형태를 가지도록 한다.

현재 유료 영상이나, 음악 콘텐츠 문제점은 DRM 서버와 사용자 간에 지속적인 연결 구조를 가지며 실시간 스트리밍 방식으로 다운로드받아 사용하기 때문에 DRM 서버의 트래픽 증가와 끊김 현상이 자주 발생하였다. 특히 기존의 동영상 크기, 사용자 증가에 비례하여 재생시간이 지연되어 스트리밍 지원이 어려웠다.따라서 본 논문에서는 다운로드된 동영상 파일을 재생하고자 할 때 라이선스만을 다운로드받은 후처리 기법을 사용하여 한층 강화된 스트리밍 지원을 하도록 구현 하였다. 다운로드는 자유롭게 할 수 있도록 하고 유료일 경우 클라이언트 정보를 DRM 서버에 전달하고 인증받은 후 복호키와 라이선스만 다운로드 받기 때문에 P2P기술이 가지고 있는 장점을 최대한 살리면서 콘텐츠의 가장 큰 문제점인 저작권 문제를 해결 할 수 있도록 하였다.

제 2 장 연구의 이론적 배경

제 1 절 P2P 및 DRM 기술에 관한 고찰

P2P 기술은 디지털 콘텐츠를 저렴한 비용으로 효과적으로 단시간 내에 광범위하게 유통시킬 수 있는 기술이다. 원하는 정보를 실시간으로 검색하여 정보의 소유자와 그 상대방이 직접 해당 정보를 주고 받는 기술은 인터넷을 통한 정보의 전달에 있어 큰 영향을 미치고 있다[4][6]. 특히 다른 어떤 통신수단보다도 익명성과 정보 교환의 자유가 최대한 보장될 수 있다는 점에서 P2P 기술은 인터넷 이용자들로부터 큰 호응을 얻고 있으며, 그 이용자 수도 날로 증가하고 있다.

영업적 측면에서도 이미 P2P 기술을 이용한 비즈니스 모델이 개발되어 서비스를 하고 있다. 그러나 P2P 기술을 통한 디지털 콘텐츠의 유통이 과거 냅스터가 하였던 것처럼 저작권자, 음반 제작자 기타 권리자의 동의 없이 이루어지는 경우에는 이는 장기적으로는 새로운 디지털 콘텐츠의 제작 및 공급 의욕을 저하시킬 수 있다. 뿐만 아니라 궁극적으로는 디지털 콘텐츠 시장의 발전을 저해하는 요인이 될 수도 있다. 따라서 디지털 콘텐츠 제공자는 저작권 및 저작인접권 등의 권리를 보호해 줄 수 있는 합법적인 사업 모델을 개발하고자 하는 노력이 필요하다.

그 방법으로는 첫째, 디지털 콘텐츠 제공자가 저작권자 및 기타 권리자와 라이선스 계약을 체결하여 그 콘텐츠를 P2P 기술로 유통시키는 것이다. 그러나 이 방법의 경우 불성실한 콘텐츠 제공자는 허락받지 않은 저작물을 유통시킬 위험이 있으며, 소비자의 측면에서도 사용 권한이 없는 자가 저작물에 접근하거나 사용할 위험도 있다. 설령 권한이 있다 하더라도 그에게 허락된 범위를 벗어나서 콘텐츠를 이용할 위험이 있다. 둘째, 파일공유만을 가능하게 하였던 기존의 P2P 기술에다 콘텐츠 제공자가 원하는 방식으로 콘텐츠를 제공하고 판매 수익을 결제할 수 있도록하고, 불법복제방지 시스템인 워터마킹 기술 등을 접목시키는 것이다. 즉 P2P와 DRM를 접목시키는 것이다.

DRM 기술의 개발 초기에는 P2P 환경을 통하여 콘텐츠가 유통되는 것을 예상하지 못하였다. 그러나 DRM 기술에 의하여 보호되는 콘텐츠는 P2P 환경 하에서도 안전하게 유통될 수 있다.

저작권자는 자신의 저작물이 유형의 매체로 제 3자에게 전달되든 아니면 e-mail 이나 P2P 서비스를 통하여 전송되든 더 이상 저작권 침해의 위험을 염려할 필요가 없다[5][9].

P2P 네트워크를 통하여 전달된 디지털 콘텐츠에는 여전히 사용규칙이 첨부되어 있으며, 해당 콘텐츠를 수신한 사용자는 그 사용 규칙에 따라서만 그 콘텐츠를 이용할 수 있다. 또한 사용자는 DRM에 의하여 보호받고 있는 콘텐츠를 DRM 기술을 채택한 장치를 통하여만 이용할 수 있다. 이 경우에도 사용자는 Clearinghouse를 통하여 대가를 지급한 경우에만 정해진 사용 규칙에 따라 해당 콘텐츠를 이용할 수 있게 된다. 사용 규칙은 매우 다양하여 콘텐츠의 복제 횟수, 사용 횟수, 배포 조건 등을 자세히 정할 수 있다.

P2P 서비스가 단순한 파일공유 서비스에만 머무는 경우에는 공유되는 파일이 DRM 기술에 의하여 보호되는 한, 앞으로 저작권 침해 문제는 크게 발생하지 않을 것이다. 그러나 P2P 서비스가 지속적인 수익을 창출할 수 있는 하나의 비즈니스 모델로서 성공하기 위해서는 결국 디지털 콘텐츠의 권리자와 각 소비자들이 P2P 기술을 통하여 전자상거래를 하는 방향으로 나아가야 할 것이다. 이 경우 디지털 콘텐츠의 거래는 P2P를 기반으로 한 사이트의 중개로 이루어지고, 실제 거래 자체는 각 판매자와 구매자의 컴퓨터상에서 이루어지게 된다. 이 경우에는 특히 거래대상인 콘텐츠의 불법거래나 불법이용을 방지하기 위하여 보안기술이 필요하며, 여기서 DRM이 중요한 역할을 할 수 있다.

콘텐츠 판매자는 DRM 기술로 보호받는 콘텐츠를 P2P 망을 통하여 직접 판매한다[15]. 그렇게 판매된 콘텐츠의 사용자는 다시 Clearinghouse를 통하여 사용료를 지급하고, 정해진 사용 규칙에 따라 해당 콘텐츠를 사용하는 것이다. 그러나 종래의 DRM 기술은 '일 대 다'와 한 권리자 대 다수의 이용자의 이용 상황을 전제로 한 것이며, P2P 서비스는 '일 대 일'의 상황을 전제로 한 것이다.

따라서 일반적인 저작물 판매의 경우보다 P2P를 통한 전자상거래 환경 하에서 DRM 기술을 접목시키는 것은 쉽지 않다. 현재 업계에서 P2P 환경과 DRM 기술을 접목시키려는 노력을 하고 있으나, 이러한 유형의 비즈니스모델로 상용화된 것은 아직 없다.

가. *DRM* 기술 정의

DRM(Digital Rights Management) 은 저작권 보호기술을 이용하여 허가되지 않은 사용자로부터 디지털 콘텐츠를 안전하게 보호하며 저작자의 권리 및 이익을 지속적으로 보호하고 관리하는 기술이다.

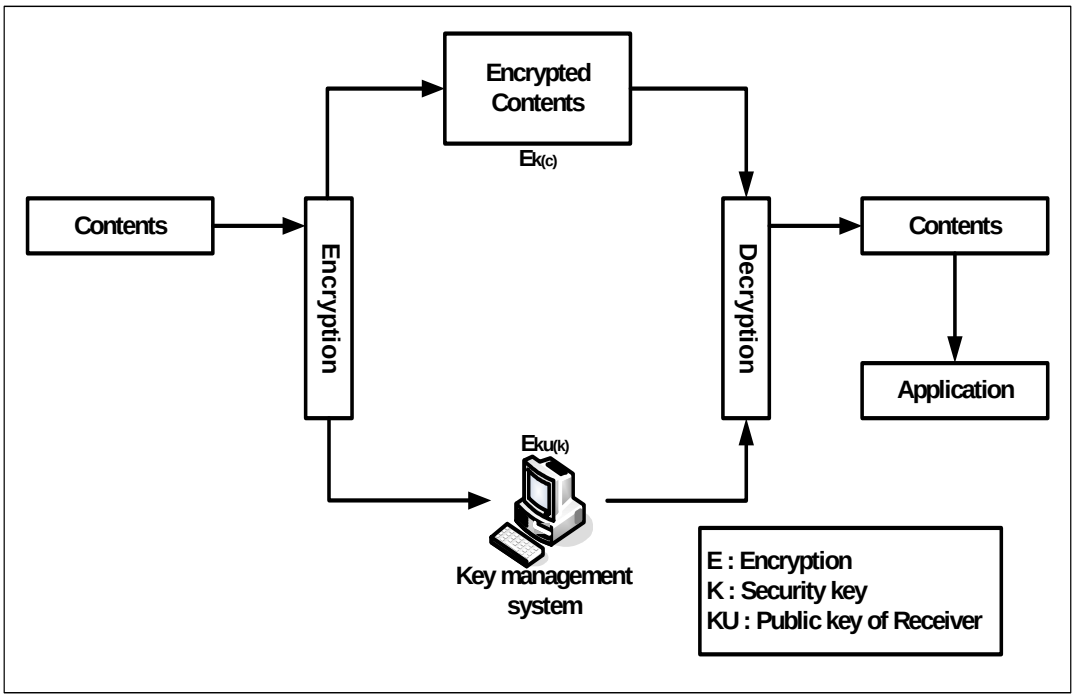
저작권 관리기술을 통해 유통과정을 관리하여 유통 업자에게 안전하게 콘텐츠를 유통 할 수 있도록 한다. 그리고 사용자에게는 법적인 문제없이 안전하고 신뢰할 수 있는 조건하에서 저작물을 사용할 수 있는 환경을 조성함으로써 저작물에 대한 제작, 추적, 이용 등이 일련의 신뢰할 수 있는 환경에서 이루어질 수 있도록 하고 있다. 또한 저작권 보호를 통해서 원 저작자의 저작권을 안전하게 보호하고, 유통업자에게는 콘텐츠를 안정하게 유통할 수 있는 토대를 마련하며, 사용자에게는 안정적으로 저작물을 공급할 수 있는 방법을 제공한다 [8][9][10].

DRM 시스템은 저작물의 기밀성과 무결성의 확보를 위하여 암호기술을 중심으로 발전하여 왔으며 저작권에 대한 내용을 명시하기 위하여 XrML(eXtensible rights Markup Language)을 기반으로 표준화가 진전되고 있고 식별자 부여를 위해서는 DOI(Digital Object Identifier)를 적극 활용하는 추세이다. 전자상거래 시스템과 결합하여 다양한 조건에 따른 디지털 저작물의 유통을 통합된 방식으로 제공할 수 있는 솔루션들이 등장하고 있다. DRM의 용도는 첫째, 디지털 콘텐츠를 유료화하는데 있어 중요한 용도로서 불법 복제를 방지해 디지털 콘텐츠가 가진 상업적인 가치를 보호하는 것이다. 둘째, 기업간 또는 기업 내의 문서 보안에 가까운 개념으로 콘텐츠가 가진 기밀성을 보호하는 것이다. 셋째, 판매 목적이 아닌 학술 논문이나 사진, 소설과 같은 콘텐츠

의 지적 재산권을 보호하기 위한 용도이다. 마지막으로 유통과정에서 디지털 콘텐츠가 변형, 훼손, 수정되지 않고 무결성을 보장하기 위한 용도로 사용할 수 있다.

DRM은 대부분 저작물에 대한 접근을 제한해 불법 복제를 막거나 저작물 내에 각종 저작권 정보, 사용자 ID 등을 삽입해 불법 사용을 추적할 수 있도록 하는 등 불법적인 유통을 원천적으로 막는데 초점을 맞추고 있다[10].

DRM 콘텐츠 보호 기술의 기본적인 흐름도를 보면 [그림 1]과 같으며 전송될 콘텐츠는 암호 기술을 이용하여 암호화된다. 수신자는 암호화된 콘텐츠를 복호화하여 콘텐츠를 이용하는데 첫째, 원본 디지털 콘텐츠가 암호화 되기 전에 과정이며 둘째, 암호화 시킨 후 키 관리시스템에 공용키를 암호화된 콘텐츠에 암호키를 각각 전송시킨다. 셋째, 복호화 할 때 키 관리 시스템에서 공용키를 암호화 콘텐츠에서 암호화키를 추출하여 최종 복호화 하는 과정이다.



[그림 1] 콘텐츠 암호기술의 흐름도

[Fig. 1] A flowchart of contents encryption technique

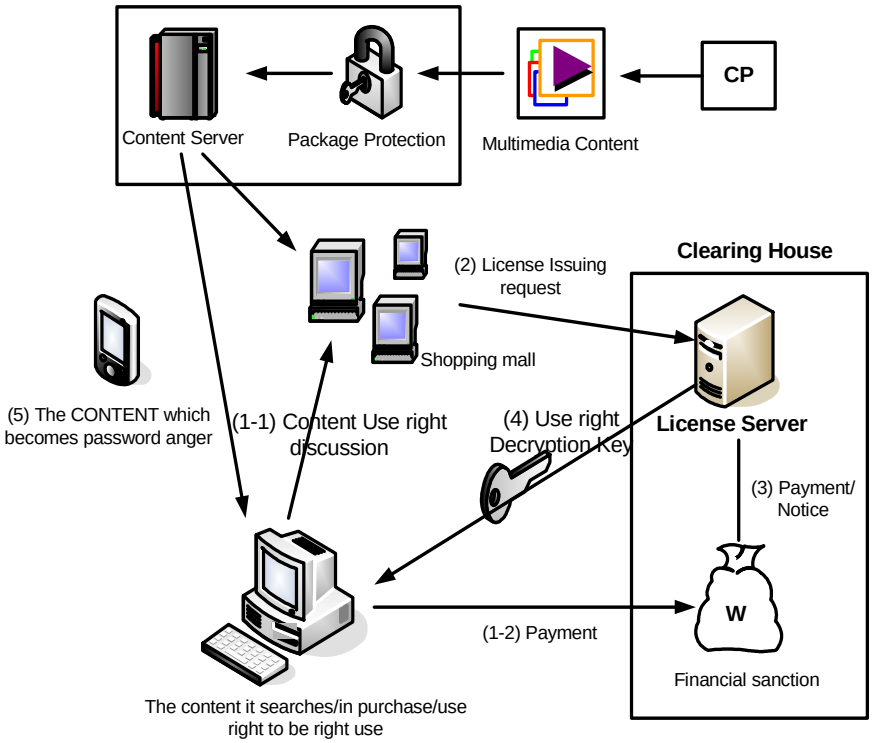
나. 디지털 저작권 관리 시스템의 구성

DRM 구성 요소는 콘텐츠 패키지, 플레이어, 라이선스 발급기의 세 가지 부분으로 나눌 수 있다. 첫째, 콘텐츠 부분은 헤더 사인과 원본 콘텐츠 암호화 부분으로 이루어진다[18]. 헤더 사인은 먼저 콘텐츠 제공업자로부터 제공 받은 원본 콘텐츠에 헤더를 추가하는데 URL(Uniform Resource Locator) 정보와 라이선스 발급기에서 복호키 생성에 필요한 정보 등 여러 정보를 헤더에 추가하고 사인하는 과정이다.

콘텐츠 암호화는 암호화 키를 발생하여 원본 콘텐츠를 암호화 한다. 암호화에 사용된 키가 플레이어 단에서 콘텐츠의 복호화에도 사용되는 대칭키 알고리즘이 DRM 솔루션에 많이 채택되고있다[21]. 헤더를 사인하는 이유는 라이선스 발급기에서 이 헤더에 들어있는 복호키 생성 정보의 변조 유무를 조사 하기 위한 것이다. 둘째, 사용자가 암호화된 콘텐츠를 재생할 때 이용하는 플레이어이다. DRM에 사용되는 일반 플레이어와 모든 기능이 동일하나, 콘텐츠 재생 시 사용자 PC의 라이선스 저장소에 현재 재생하고자 하는 암호화된 콘텐츠 라이선스가 존재하는지를 조사하고 라이선스가 있을 경우에는 라이선스에 들어있는 콘텐츠 사용규칙에 따라 콘텐츠를 재생하여 준다. 라이선스가 없거나 기한 등이 지난 무효 라이선스일 경우에는 콘텐츠의 헤더에 있는 라이선스 구매 URL로 콘텐츠 헤더에 있는 각종 정보와 더불어 자신의 PC에 고유한 정보를 전송한다.

마지막으로 DRM 라이선스 발급기 에서는 플레이어로부터 전달 받은 정보중 콘텐츠 헤더 부문에 들어 있는 키 관련 정보를 이용하여 복호키를 생성하고, CP와 사용자 간에 합의된 콘텐츠 사용 규칙을 라이선스에 담아서 사용자의 PC로 전송한다.

[그림 2]와 같이 Clearinghouse는 라이선스를 발생, 전송하기 전 사용자가 라이선스에 대한 가격을 지불하였음을 Clearinghouse와 연동된 금융기관으로부터 통보받는다.



[그림 2] DRM 프로세스 흐름도

[Fig. 2] A flowchart of DRM process

DRM 프로세스의 세부적인 흐름은 [그림 2]와 같다[12][13][17]. CP로부터 제공 받은 디지털 콘텐츠는 패키지 과정을 통해 보호되며 사용자는 콘텐츠 서비스 사이트로부터 스트리밍이나 다운로드 형태로 콘텐츠를 이용하기로 결정하면

(1-1) 해당 콘텐츠 판매 또는 서비스 사이트는 라이선스 서버로 라이선스 발급요구를 한다.

(1-2) Clearinghouse와 연계된 금융회사로부터 지불 통보하고 통보받은 라이선스 서버는 패키지 된 콘텐츠를 풀 수 있는 키와 사용권리 등을 포함한 라이선스를 사용자에게 발급한다.

(2) 사용자는 서비스 사이트(쇼핑몰)가 부가한 사용요금을 지불한다.

(3) 서비스 사이트에 부가한 사용 요금 정상 처리 확인

(4) 사용자에게 복호키 전달하고 암호화된 콘텐츠가 사용자에게 다운로드 되거나 스트리밍서비스 된다.

(5) 소비자의 사용자 PC에 전송된 라이선스를 이용하여 콘텐츠 플레이어는 암호화된 다운로드 또는 스트리밍 콘텐츠를 복호화, 재생하여 소비자에게 제공한다.

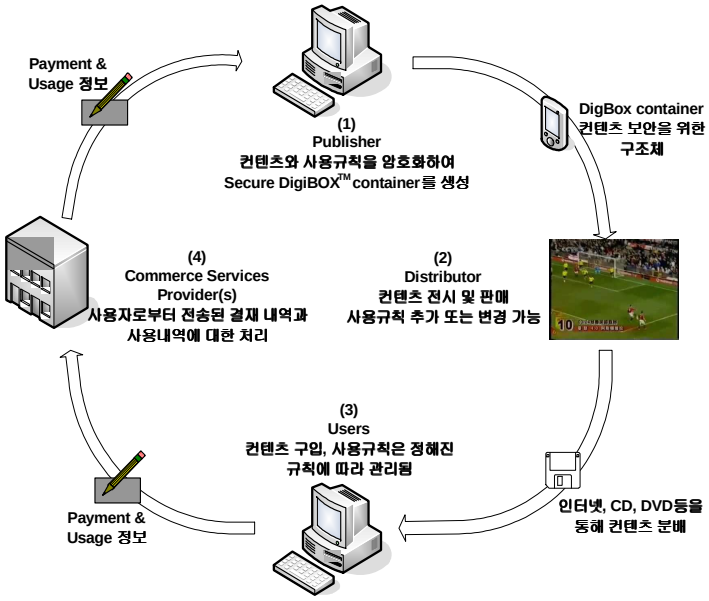
제 2 절 DRM 모델 연구

가. InterTrust의 DRM 솔루션

InterTrust사의 DRM 시스템의 특징은 [그림 3]과 같이 저작물의 보호를 위해서 암호기술과 워터마킹을 사용하며 저작물의 사용 규칙을 지정하여 사용내역의 수집 및 기록, 과금 처리를 수행하는 것이다. 특히, 사용자 컴퓨터에 에이전트를 실행하여 라이선스와 과금처리, 저작물의 실행을 에이전트를 통하여 처리하도록 하였다[20].

저작물은 사전에 암호화되어 배포되므로 사용자의 컴퓨터에서 저작물을 사용하는 시점에서 라이선스 에이전트가 라이선스를 확인하고 지불 정보를 전송하여 거래를 체결하도록 하였다. 그리고 저작물이 암호화되어 보호되고 있으므로 사용자들 사이에 암호화된 저작물을 주고 받을 수 있는 저작물 재분배를 구현하였다.

InterTrust의 DRM 기술은 [그림 3]에서 동작한다. DigiBox Container 는 암호화된 저작물과 사용 규칙 전송을 탑재하게 된다. 사용 규칙은 가격, 결재방법, 재생, 프린트, 복사, 저장, 재분배 등에 관한 규칙으로 DigiBox에 탑재되어 있다. 거래 승인(Transaction Authority) 프레임워크는 저작물의 사용 내역이나 과금 내역 등의 정보를 처리하도록 한다. InterTrust에서 라이선스 획득 방법은 다른 방식들과 다르게 이용자의 컴퓨터에 설치된 라이선스 관리자를 통하여 라이선스를 획득한다. 그러나 이 방식은 오프라인 상태에서는 라이선스를 확인할 수 없으므로 사용 할 수 없는 문제점이 발생한다. 또한 서비스 제공자와 출판업자 사이에 사용내역 전송과정에서 메시지가 평문 상태로 전달되기 때문에 사용자의 지불정보와 사용정보가 노출되어 심각한 보안 위협 사항이 존재한다.

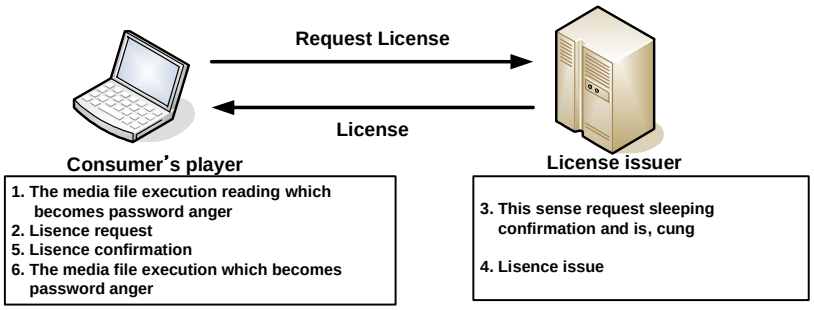


[그림 3] InterTrust사의 DRM 시스템

[Fig. 3] A DRM System of InterTrust company

나. *Microsoft*의 *DRM* 시스템

Microsoft의 DRM 시스템은 저작물 제공자와 소비자들에게 디지털 미디어 파일을 안전하게 분배하는 종단 간(end-to-end) DRM 시스템이다. 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[35][36]. WMRM에서 각각의 서버 또는 클라이언트 인스턴스들은 개인화 과정을 통해 키 쌍을 할당받게 되며, 크래킹 되었거나 안전하지 않다고 판단되는 인스턴스에 대해서는 인증서 취소목록을 이용하여 서비스 대상에서 제외시키게 된다. 인증서 취소목록은 Microsoft의 웹사이트를 통해 배포된다. 키는 라이선스에 포함되고, 라이선스와 저작물은 분리되어 분배된다. [그림 4]는 WMRM에서 라이선스 획득 단계로서 클라이언트가 패키징되어 보호된 저작물을 실행시키면 플레이어는 라이선스 서버에 라이선스를 요청한다. 서버는 라이선스 요청에 대해 사용자의 인증과 지불여부를 확인한 후에 라이선스를 발행한다. 서버는 라이선스 발생 후 라이선스를 클라이언트의 플레이어에 전송한다. 클라이언트 플레이어는 서버에서 전송받은 라이선스를 확인한 후 사용 규칙에 따라 저작물을 실행한다.



[그림 4] Microsoft사의 DRM 라이선스 획득과정

[Fig. 4] A DRM License acquisiting process of microsoft company

WMRM의 경우 Key ID와 Key seed를 결합하여 저작물 암호화키를 생성하는데, Key ID는 저작물 헤더에 포함되며 저작물과 함께 패키징되어 배포되고, Key seed는 Clearinghouse에 저장되어 관리된다.

복호화 키를 생성하기 위해서는 저작물에 포함되어있는 Key ID와 서버가 관리하는 Key seed가 필요하다. WMRM은 윈도우 미디어 플레이어에 탑재되어 널리 사용되지만, 동적 변화 환경에 제한적이고 자사의 동영상 파일 포맷인 asf와 WMA 파일 포맷에만 적용되어 다양한 파일 형식을 지원하지 못한다. 또한 라이선스를 발급 받기 위한[그림 4]의 2단계와 3단계의 인증단계에서 특정한 암호기술을 적용하지 않고 사용자의 정보를 전송하므로 사용자의 정보가 노출되는 보안 위협 요소가 존재한다.

제 3 절 *DRM* 핵심 기술

DRM은 콘텐츠의 정보를 허가되지 않은 사용자로부터 보호하기 위하여 암호화 기술을 이용한다. 암호 기술을 이용한 콘텐츠의 보안 강도는 암호화를 위해 사용된 알고리즘의 강인성과 암호화 키의 길이, 그리고 이러한 암호 알고리즘을 처리하는 프로세스의 공격에 대한 강인성 정도에 따라 달라진다. 일반적으로 암호기술의 강인성은 암호화를 위해 사용된 키의 길이와 비례하여 증가하게 된다.

1. 암호화 알고리즘

정보처리기술 및 정보통신기술의 혁신적인 고도화에 따른 컴퓨터 통신과 인터넷의 발전은 사이버 세계의 출현을 가능하게 하였다. 이러한 사이버 세계에 대한 의존도의 증가는 새로운 형태의 정보 전송 및 저장이 필요하게 되었고, 컴퓨터 시스템의 내부에서 또는 시스템간의 통신에서 새로운 개념의 정보보호 기술의 문제가 대두되게 되었다[22][42]. 암호화 기술은 데이터의 보호와 사용자의 인증이라는 관점에서 사이버 패러다임의 기반을 이루게 되었고 전자상거래나 스마트 카드와 같은 새로운 데이터 통신에서의 데이터 보호를 위한 암호화 알고리즘과 그것의 효율적인 구현에 대한 연구가 활발히 이루어지고 있다.

2. RSA

W.Diffie와 M.E. Hellman에 의해서 처음 제안된 공개키(Puplic Key) 암호 시스템은 1978년 Rivest, Shamir 및 Adleman 이 제안한 RSA 암호 시스템에서 보다 체계적이고 안전한 시스템으로 구체화 되었다. RSA 암호 시스템은 기존 대칭키(Private Symmetric Key)가 키 관리와 전송을 위한 안전한 채널이 필요한 단점을 극복하였으며 또 사용자 인증 및 디지털 서명이 가능하다는 장점을 가지고 있다[23].

이러한 RSA 시스템의 암호화와 복호화 과정의 핵심연산은 모듈라 지수 연산

(Modular Exponentiation, C : A^E mod M)으로써 암호화 빈도를 증대시키지 위하여 512비트 이상의 큰 수의 모듈라 지수 연산을 사용하여 많은 수의 메시지 블록을 연속적으로 처리하게 됨에 따라 처리 속도에 문제점을 가지게 되었다.

가. *RSA* 공개키 암호

일반적으로 모든 공개키 암호에서 쓰이는 수학적 개념 중에 하나가 '합동'이다. 두 정수 a 와 b의 차이가 1보다 큰 자연수 n의 배수일때 "a 와 b는 n을 법으로 합동(congruence)이다" 라고 정의하고 'a = b mod n' 으로 쓰기도 한다. 한편 두 정수의 덧셈과 곱셈을 정수에서 의 덧셈과 곱셈을 수행한 뒤에 n을 법으로 n보다 작은 정수로 표현하는 것으로 정의하며, 소약법칙을 제외하고 정수에서 성립하는 대개의 성질들이 만족한다. n을 법으로 하는 수들의 집합 Z_n={0,1,.....n−1}은 n을 법으로 하는 연산 아래서 기환을 이룬다.

암호화 과정과 복호화 과정을 설명하기 위하여 필요한 몇 가지 기호를 나열하면 다음과 같다.

n : 암호화 과정과 복호과정에 이용되는 공개된 법(modulus)

p, q : n의 서로 다른 소인수로 공개되지 않은 소수, 즉 n=p · q

e : 공개된 지수(보통 3 또는 216 + 1)

d : 공개되지 않은 지수로 de=1 mod (p-1)(q-1)을 만족하는 양의 정수

M : 암호화하기 전의 평문(양의 정수)

C : 암호화된 암호문(양의 정수)

E : 암호변환 함수, 즉 E(M)=C

D : 복호변환 함수, 즉 D(C)=M

RSA 공개키 암호의 암호변환 함수의 복호변환 함수는

C = E(M) = M^e mod n,

M = D(C) = C^d mod n

으로 정의된다. 이때 공개된 수 e와 n을 이 암호계의 공개키라 하고 d를 개인이 보안을 유지해야 하는 개인키라 한다.

예를 들면, A 는 전문(message)에 자신의 서면(signature) S_A를 자신의 개인키 d를 이용하여 $C_A = S_A^d$ mod n 을 계산하여 B에게 C_A를 보내면, B는 A의 공개키 e를 이용하여 $S_A = C_A^e$mod n을 계산하여 S_A을 구하여 전문을 송신한 사람이 A임을 확인하는데 이용된다. 한편, B는 전문 M을 A의 공개키 e를 이용하여 암호문 $C = M^e$ mod n을 계산하여 A 에게 C를 보내면, A는 자신의 개인키 d를 이용하여 $M = C^e$ mod n을 계산하여 암호문 C를 전문 M으로 복호한다.

이 암호계가 안전한 암호가 되려면 공개된 수 e와 n으로부터 계산량의 기준에서 d를 얻기 어려워야 하고 C = E(M) = M^e mod n = M이 되는 평문이 없어야 한다.

또한 선형 시스톨릭 어레이의 처리기수와 처리시간을 중심으로 비교하면 [표 1]와 같다.

[표 1] 선형 시스톨릭 어레이의 처리기수와 처리시간

[Table 1] The processor possibility and processor time of Linear systolic array

	처리기	계산시간	파이프라인주기	입출력채널수
(1)	n(A) + 1	2n(A)+ n(B, M)+ 2	1	O(n(A))
(2)	n(B, M) + 1	2n(A)+ 3n(B, M)+ 3	2	O(n(B, M))
(3)	n(A)+ n(B, M) + 2	3n(A)+ n(B, M)+ 2	1	O(n(A))
(4)	n(A)+ n(B, M) + 2	4n(A)+ n(B, M)+ 2	3	O(1)

나. 암호화 과정

임의의 자연수 m에 대하여 m보다 작고 m과 서로소인 자연수들의 개수를 함수 값으로 가지는 함수를 Euler~Totient 함수라 하고 기호로 $\Phi(m)$으로 나타낸다. 예를 들면,

$\Phi(2)=1,\ \Phi(3)=2,\ \Phi(4)=2,\ \Phi(6)=2$ 이다.

특히, p와 q가 서로 다른 소수일 때 다음 성질들

(i) $\Phi(p) = p-1$

(ii) $\Phi(p \cdot q) = (p-1)(q-1)$

이 성립한다.

앞 절에서 서술한 RSA 공개키 원리는 Euler 정리를 기본으로 하는 [정리 2]에 의존한다.

[정리 1(Euler 정리)] 임의의 자연수 m과 n이 서로소(relative prime)이며, $m^{\Phi(N)} = 1$ mod n 이다.

[정리 2] n = p · q (p와 q는 서로 다른 홀수인 소수) ed = 1 mod Φ(n)이며, 임의의 M에 대하여 $M^{ed} = M$ mod n 이다.

따라서 RSA 공개키 암호의 복호 과정이 [정리 2]에 의존하는 것을 쉽게 알 수 있다. 이 공개키 암호의 해독은 공개키 e와 n을 이용하여 d를 구하는 문제와 같다. 한편 임의의 자연수 e와 k가 서로소이면, ed = 1 mod k인 d가 존재한다. 이 d를 구하는 데는 Lamed의 정리와 Euclid 호제법을 이용하여 구할 수 있다. 실제로 Lalme의 정리에 의하면 gcd(e, Φ(n)) =1 일때 ed =1 mod Φ(n)을 만족하는 d를 얻기 위하여 $-2+[\log_R \Phi(n)]$단계의 나눗셈을 시행하여 얻을 수 있다. (단, $R = ((1+\sqrt{5})/2)$), 따라서 이 RSA 암호가 해독되지 않으려면 n의 소인수분해가 어려워야 한다.

RSA 암호를 해독하려면 Φ(n)을 알아야 하고 [정리 3]에 의하여 n을 소인수분해하여 p와 q를 얻어야 한다.

[정리 3] n이 소수 p와 q의 곱일 때 p와 q가 주어지는 것은 Φ(n)이 주어지는 것과 동치이다.

결국 빈도가 높은 암호계가 되려면 p와 q로 소인수분해하기 어렵게 되어야 한다. 즉 p, q 및 d가 다음과 같을 때 최적화된다.

(1) p와 q는 길지 않고 거의 같은 크기의 자리 수이어야 한다.

(2) p-1과 q-1은 커다란 소인수를 각각 가져야 한다.

(3) p-1과 q-1의 최대공약수는 작은 수이어야 한다.

(4) max{p, q} + 1<d<n-1이 되도록 d를 선택한다.

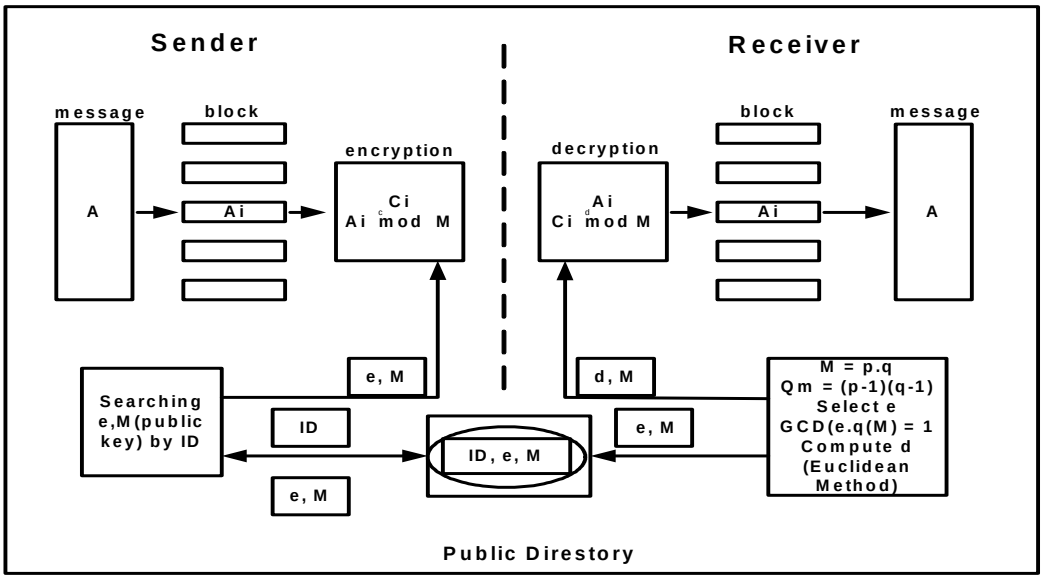
이제 RSA 공개키 암호계에서 공개키 e를 선택하는 방법은 공개키 e를 선택하였을 때, 암호문이 평문과 같으면 문제가 발생한다. 즉, 이 경우는 [정리 4]와 같다.

[정리 4] 공개열쇠의 쌍 (e,n)을 가지는 RSA 알고리즘을 이용하여 암호화 하면 {1+gcd(e-1, p-1)} {1+gcd(e-1, q-1)}개는 평문과 같은 암호문을 가진다.

실제로 두 소수 p와 q가 위의 조건같이 적은 m을 갖도록 만족하면 평문과 암호문이 같은 경우는 몇 가지 되지않고 큰 소수의 곱 n과 비교하면 무시할 만큼만 일어나는 것을 알 수 있다.

일반적으로 RSA 에 대한 공격은 n의 소인수 분해 공격과 암호문의 이산 대수공격의 두 가지를 고려할 수 있다[31].

모듈라 승산 모델을 모듈라 지수 연산으로 확장하여 적정 규모의 VLSI 어레이를 칩으로 제작한 후 필요한 연산의 크기만큼 설계하여 연결함으로써 확장 가능한 모듈로 구성하여 [그림 5]의 RSA 암호시스템의 암호화 및 복호화 과정에서 코어(core)부분에 해당하는 모듈라 지수 연산기를 구성할 수 있다[52][53]. 또한 고속 연산을 위하여 처리요소의 이용률을 높임으로써 하드웨어 요구량과 복잡도를 감소시킬 수 있다.



[그림 5] RSA 암호시스템의 블록도

[Fig. 5] The block diagram of RSA encryption system

3. SEED

C. E. Shannon은 1949년 논문에서 확산(diffusion)과 혼동(confusion)이라 불리는 실용적인 암호 시스템의 기본 개념을 제시하였다. VLSI로 구현한 SEED는 이 개념에 따라 제작된 전자상거래 등 네트워크상의 데이터를 보호하기 위해 만들어진 대칭키 암호알고리즘으로써, 블록 단위로 메시지를 처리하는 블록 암호알고리즘이다[24][42].

대칭키 블록 암호알고리즘은 기밀성을 제공하는 암호시스템의 중요 요소이다. n비트 블록 암호알고리즘이란 고정된 n 비트 평문을 같은 길이의 n 비트 암호문으로 바꾸는 함수를 말한다.(n 비트 : 블록 크기). 이러한 변형 과정에 암호키가 작용하여 암호화와 복호화를 수행한다.

가. SEED 알고리즘 구조

전체적인 구조는 파스텔 암호화(Feistel cipher)구조를 따른다. [그림 6]은 파스텔 암호화 구조로써 특징은 다음과 같다.

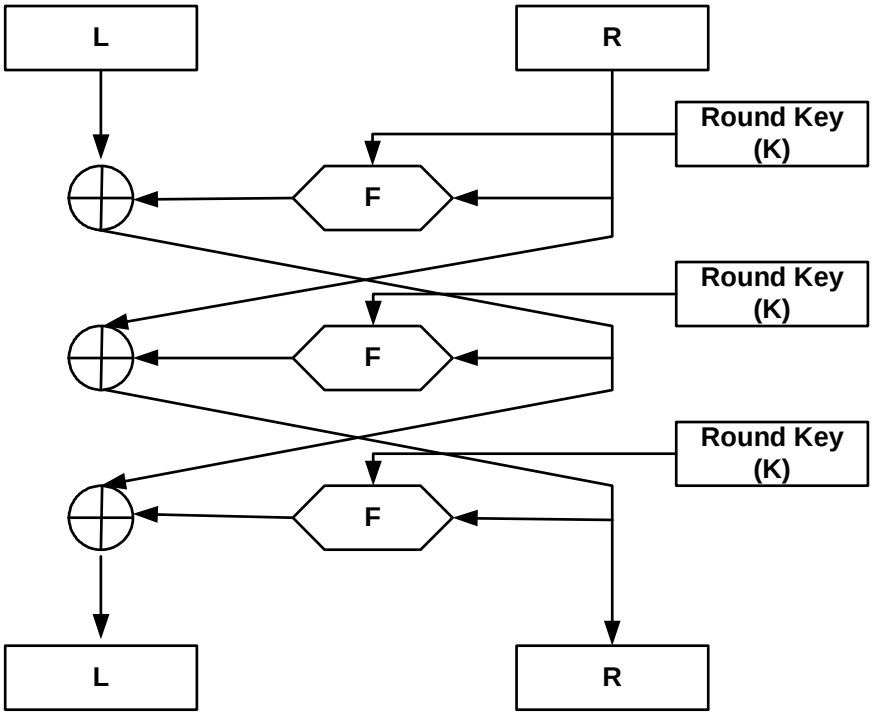
(i) 암호화 방식은 대칭키 블록 암호화(Symmetric Block Cipher)방식을 사용한다.

(ii) 128 bit 블록을 암호화 하는 기본 단위로 사용한다.

(iii) 암호화를 위한 키의 길이는 128 bit를 사용한다.

(iiii) 한 블록을 암호화하는데 사용하는 라운드 횟수는 16번이다.

(iiiii) 하나의 키를 매 라운드 수행 시마다 계산을 통해 새로운 키를 생성하여 사용한다.



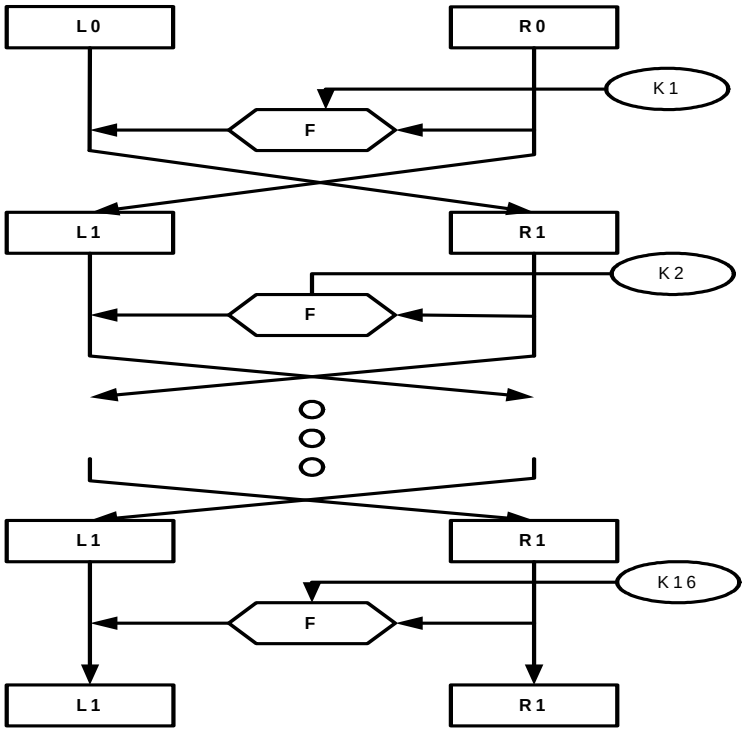
[그림 6] Feistel cipher의 구조

[Fig. 6] The structure of Feistel cipher

나. 블록 암호화 알고리즘 *SEED*

알고리즘의 전체 구조는 파스텔 암호화 구조로 이루어져 있으며, 128 bit의 평문 블록단위당 128 bit 키로부터 생성된 16개의 64 bit 라운드 키를 입력으로 사용하여 총 16라운드를 거쳐 128 bit 암호문 블록을 출력한다.

SEED 알고리즘의 전체 구조는 [그림 7]과 같다.



[그림 7] SEED 구조

[Fig. 7] The structure of SEED

128 bit 입력 평문 블록을 2개의 64 bit 블록 ($L_0(64)$, $R_0(64)$)로 나누어, 16개의 64 bit 라운드를 이용하여 16라운드를 수행 후 최종 128 bit 암호문 블록 ($L_{16}(64)$, $R_{16}(64)$)을 출력한다.

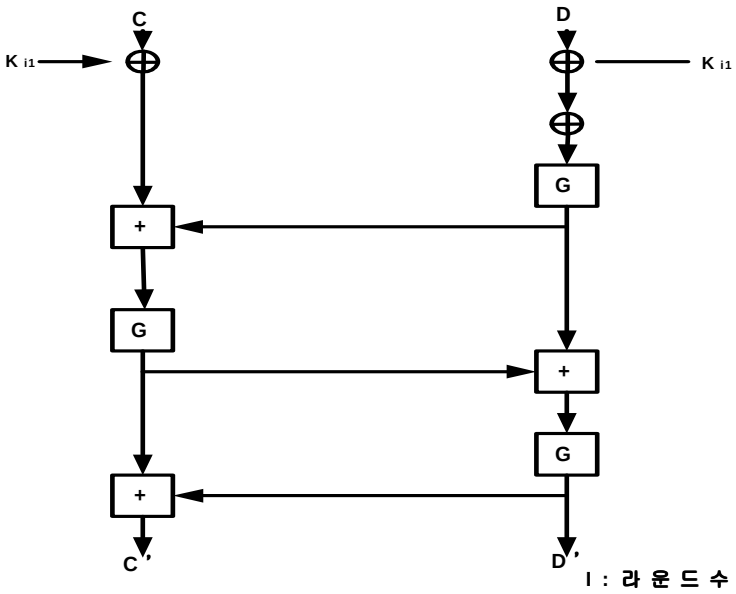
다. *F* 함수

파스텔 구조를 갖는 블록 암호알고리즘은 F 함수의 특성에 따라 구분될 수 있다. SEED의 F 함수는 수정된 64 bit 파스텔 형태로 구성된다. F 함수는 각 32 bit 블록 2개(C, D)를 입력으로 받아, 32 bit 블록 2개(C′, D′)를 출력한다. 즉 암호화 과정에서 64 bit 블록 (C, D)과 64 bit 라운드 키 $K_i = (K_{i,o}\ ;\ K_{i,1})$을 F 함수의 입력으로 처리하여 64 bit 블록 (C′, D′)을 출력한다. [그림 8]은 SEED에서 사용하고 있는 F함수를 보인 것이다.

$$C' = G[\,G[\,G(C \oplus K_{i0}) \oplus (D \oplus K_{i1})^{\boxplus}\,[(C \oplus K_{i0})]^{\boxplus}\ G(C \oplus K_{i0}) \oplus (D \oplus K_{i1})]^{\boxplus}$$

$$G[\,G(C \oplus K_{i0}) \oplus (D \oplus K_{i1})^{\boxplus}\ (C \oplus K_{i0})]$$

$$D' = G[\,G[\,G(C \oplus K_{i0}) \oplus (D \oplus K_{i1})^{\boxplus}\,[(C \oplus K_{i0})]^{\boxplus}\ (C \oplus K_{i0})]^{\boxplus\{}\ G(C \oplus K_{i0}) \oplus (D \oplus K_{i1})]$$

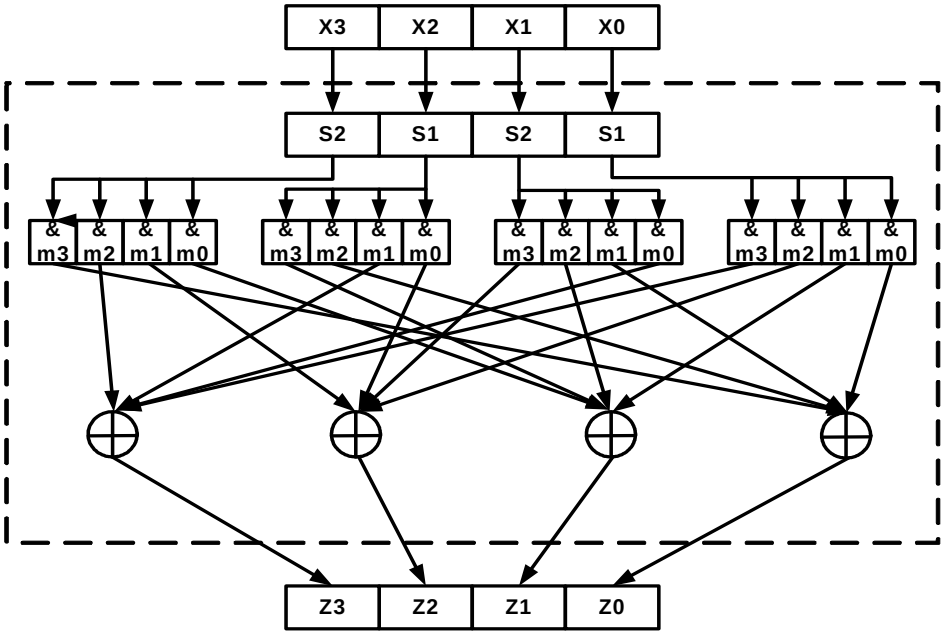


[그림 8] F 함수의 구조

[Fig. 8] The structure of F-function

라. G 함수

SEED의 S-Box는 8 bit 입력(즉, 0~255)을 받아 8 bit 출력(즉, 0~255)을 내는 함수로써 예를 들면, $S_1 - Box$의 경우, 입력이 '21'이면 출력은 '226'이 된다. [그림 9]는 F 함수 내에서 S-Box 4개를 결합하여 이 결과를 XOR 연산을 수행하는 G 함수의 구조를 보인 것이다[43].



[그림 9] G 함수의 구조

[Fig. 9] The structure of G-function

또한, SEED의 키 생성 알고리즘은 128 bit의 암호키를 64 bit씩 좌우로 나누어 이들을 교대로 8 bit 좌/우로 회전이동 후, 결과의 4 word 들에 대한 간단한 산술연산과 G함수를 적용하여 라운드 키를 생성한다. 키 생성 알고리즘은 기본적으로 하드웨어나 제한된 자원을 갖는 스마트카드와 같은 응용에서의 효율성을 위하여, 암호화나 복호화시 암호키로부터 필요한 라운드 키를 간단히 계산할 수 있도록 설계한다.

각 라운드에 사용되는 라운드 키는 다음과 같은 방식으로 생성하며, [표 2]은 G함수의 라운드 상수를 나타낸 것이다.

(i) 128 bit 입력키를 32 bit 4개의 조각으로 쪼갠 후 (A, B, C, D),

(ii) $K_{1,0} = G(A + C - KC_{0})$; $K_{1,1} = G(B + KC_0 - D)$

(단, KC_0 : 라운드상수)로 1라운드 키를 생성하고,

(iii) B||A = (B||A)>>8

(iiii) $K_{2,0} = G(A + C - KC_1)$; $K_{2,1} = G(B + KC_1 - D)$

(단, KC_1 : 라운드상수)로 2라운드 키를 생성하고,

(iiiii) D||C = (D||C)<<8

(iiiiii) $K_{3,0} = G(A + C - KC_2)$; $K_{2,1} = G(B + KC_2 - D)$

(단, KC_2 : 라운드상수)로 3라운드 키를 생성하고,

(iiiiiii) 이후 16라운드 키 생성까지 반복한다.

for(i=1; 1≤16; 1++) {

$K_{i,0} \leftarrow G(A + C - KC_{i-1})$;

$K_{i,1} \leftarrow G(B + D - KC_{i-1})$;

if(1%2==1) A||I ← $(A||B)^{\gg 8}$;

else C||I ← $(C||D)^{\ll 8}$;

}

라운드 상수 KC_i는 황금비의 소수 부분으로부터 다음과 같이 생성한다.

KC_0 = int($\frac{\sqrt{5}-1}{2} X\ 2^{32}$) = 0x9e3779b9,

KC_i = $KC_{i-1} \ll 1$ for $1 \le i \le 15$.

[표 2] G함수의 라운드 상수

[Table 2] The round constant of G-function

상 수	
KC_0 = 0x9e3779b9	KC_8 = 0x3779b99e
KC_1 = 0x3c6ef373	KC_9 = 0x6ef3733c
KC_2 = 0x78dde6e6	KC_{10} = 0xdde6e678
KC_3 = 0xf1bbcdcc	KC_{11} = 0xbbcdccf1
KC_4 = 0xe3779b99	KC_{12} = 0x779b99e3
KC_5 = 0xc6ef3733	KC_{13} = 0xef3733c6
KC_6 = 0x8dde6e67	KC_{14} = 0xde6e678d
KC_7 = 0x1bbcdccf	KC_{15} = 0xbcdccf1b

4. *AES* 알고리즘

AES는 Advanced Encryption Standard의 약어로 현재 미국의 표준 암호 알고리즘에 대한 공식 명칭이다. AES 이전에 미국의 표준 알고리즘으로는 DES와 3DES가 있었다. 1997년 1월 2일 미국의 NIST(National Institute of Standards and Technology)에서 AES 선정 계획을 공고했고 이를 본 전 세계의 많은 암호학자와 연구소에서 암호알고리즘을 개발하여 제출했다. 이때 제출된 알고리즘들은 CAST-256, CRYPTION, DEAL, DFC, E2, FROG, HPC, LOKI97, MAGENTA, MARS, RC6, Rijndael, SAFER+, Serpent, Twofish 등이다. 이 중 일부는 암호 분석학자들에 의해 문제점이 노출되었고 일부는 성능상의 문제로 제외되어 1999년 4월경 AES 후보 알고리즘은 모두 5개(MARS, RC6, Rijndael, Serpent, and Twofish)로 좁혀졌다[39]. Rijndael은 암호알고리즘 분류 상 대칭형암호알고리즘으로 분류된다. 대칭형 암호알고리즘에도 두 가지 종류가 있는데 Rijndael은 이 중에서 Block 암호알고리즘 방식이며 128 bit 블럭 단위로 암호화를 하고, 사용되는 Key의 사이즈는 128 bit, 192 bit, 256 bit 등이 있다. 라운드 수는 각각 10, 12, 14라운드를 사용하고 각 라운드 마다 SubBytes, ShiftRows, MixColumns, AddRoudKey의 4단계를 거치게 된다.

- Block의 길이: 128
- Key의 길이 : 128, 192, 256
- 라운드 수= Nr

Key length 128 $\Rightarrow$ Nr = 10

Key length 192 $\Rightarrow$ Nr = 12

Key length 256 $\Rightarrow$ Nr = 14

AES의 모든 연산은 byte에 기초를 둔 연산이며 1 byte은 16진법 두 자리 수로 나타낼 수 있다. 128 bit 평문이 주어지면 16개의 byte으로 각 행렬의 원

소가 1 byte인 4×4 행렬 모양으로 표현하며 이를 'state' 라고 부른다.

1byte은 16진법 두 자리수로 표현한다. 예를 들어

$x=$ 32 43 f6 a8 88 5a 30 8d 31 31 98 a2 e0 37 07 34

이면 state은 다음과 같다.

32	43	f6	a8
88	5a	30	8d
31	31	98	a2
e0	37	07	34

일반적으로 즉. $x=x_0x_1x_2\cdots x_{127}$ 이면 $s_{0,0}=x_0x_1\cdots x_7$, $s_{1,0}=x_8x_9\cdots x_{15}$ 등이며 평문 x 를 다음과 같은 모양으로 초기화한다.

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

AES에서는 이 state 모양이 초기 input이며 output도 state이다. 위 state 1 행, 혹은 1 열은 32 bit(=4 byte)이며 이를 word 라고 부른다.

1. 평문 x, State를 x로 초기화한 후 Add-RoundKey 연산 RoundKey $\oplus$ State을 실행한다.

2. 첫 번째 라운드에서 Nr-1 라운드까지

- S-box를 이용하여 State에 치환 연산 SubBytes을 실행한다
- State 에 permutation 연산 ShiftRows를 실행
- State 에 MixColumns를 실행
- State 에 AddRoundKey를 실행한다.

3. 마지막 라운드에서는 SubBytes, ShiftRows, AddRoundKey를 실행한다.

4. State를 암호문(ciphertext)으로 정의한다.

SubBytes 연산은 byte 단위의 연산이며 State 의 $s_{i,j}$를 S-box를 이용하여

새로운 $s_{i,j}$로 대체한다. $s_{i,j} = XY$이면

$\pi_S(s_{i,j}) = \pi_S(XY) = S-box$에서 X행 Y열 원소.

사실, 1byte과 F_{2^8}의 원소를 앞에서 설명한대로 자연스럽게 대응시키면 SubBytes 연산은 다음과 같다.

$$\text{SubBytes}(z) = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \end{pmatrix} z^{-1} + \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

정리하면 Subbytes 알고리즘은 다음과 같다.

SubBytes 알고리즘

Input{1 byte $(a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0)$}

Output{1 byte $(b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0)$}

$z \leftarrow (a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0)$

$z \leftarrow$ BinaryToField(z)

if $z \neq 0$

then $z \leftarrow$FieldInv(z)

$(a_7 a_6 a_5 a_4 a_3 a_2 a_1 a_0) \leftarrow$ FieldToBinary(z)

$(c_7 c_6 c_5 c_4 c_3 c_2 c_1 c_0) \leftarrow (01100011)$

comment: 다음 loop에서 아래첨수는 법 8로 계산한 수임

for $i \leftarrow 0$ to 7

do $b_i \leftarrow (a_i + a_{i+4} + a_{i+5} + a_{i+6} + a_{i+4} + c_i) \bmod 2$

return $(b_7 b_6 b_5 b_4 b_3 b_2 b_1 b_0)$

5. SHA 알고리즘

다른 암호화 알고리즘과 달리 복호화 과정이 없으며 출력 값인 해쉬 값을 직접 이용하기 때문에 강한 충돌 저항성을 요구 한다.

SHA은 미국 NIST(National Institute of Standards and Technology)에 의해 개발된 알고리즘으로 512 bit 단위의 메시지 블록을 처리한다. 512 bit 단위의 메시지 블록은 32 bit의 소블록으로 나누어 각 단계마다 처리된다.

전체적인 메시지의 처리과정은 그림[10]에서 블록 길이가 512 bit 이고 헤더 길이와 160 bit 길이의 연쇄 변수를 가질 때 MD5에 대한 구조를 따른다. 처리 과정은 다음 단계로 구성된다.

· Step 1 : Append padding bit.

메시지는 패드 된다. 따라서 메시지의 길이는 448모듈로 512와 합동이다. 패딩은 항상 더한다. 심지어 메시지가 이미 그 길이에 적합할 때도 패딩이 일어난다. 따라서 패딩 비트의 수는 1에서 512의 범위 안에 있다. 패딩은 단일 1 bit 다음에 필요한 0 bit의 숫자로 구성된다.

· Step 2 : Appent length.

64 bit의 블록은 메시지로 추가된다. 이 블록은 부호 없는 64 bit 정수 (most signifiant bytes first)로 원본메시지의 길이를 포함한다.

· Step 3 : Initialize MD buffer.

160 bit의 버퍼의 해시 함수의 중간과 마지막 결과를 가지는데 이용된다. 버퍼는 다섯 개의 32 bit 레지스터(A, B, C, D, E)로 표현 가능하다. 이 레지스터들은 다음 32 bit 정수(16진수)로 초기화 된다.

A = 67452301

B = FGCDAB89

C = 98BADCFE

D = 10325476

E = C3D2E1F0

첫 번째 4가지 값(A,B,C,D)은 MD5에 사용했던 것과 같다. 그러나 SHA-1의 경우에는 이 값들은 가장 큰 바이트가 낮은 주소 바이트에 저장되는 big-endian 형식으로 저장된다.

32 bit 스트링처럼 초기화 값(16진수)은 다음과 같이 나타난다.

word A : 67 45 23 01

word B : EF CD AB 89

word C : 98 BA DC FE

word D : 10 32 54 76

word E : C3 D2 E1 F0

· Step 4 : Process messane in 512- bit(16-word) blocks.

이 알고리즘의 핵심은 [그림 10]과 같이 각각 20단계의 프로세싱의 4가지 라운드로 구성된 모듈로 4개의 라운드는 비슷한 구조를 가지지만 각각은 다른 f1, f2, f3, f4로 부르는 기초 논리 함수를 사용한다. 각 라운드는 (Yq)로 처리되는 현재 512 bit 블록을 입력으로 가지며 160 bit 버퍼값 ABCDE와 버퍼의 내용을 갱신한다. 각 라운드는 또한 덧셈에 대한 상수 Kt($0 \le t \le 79$는 다섯 라운드를 교차하는 80단계중의 하나를 가리킨다.)를 사용한다. 그러나 오직 4개의 별개의 상수만이 사용된다. 그 값들은 16진수와 10진수에서 [표 3]와 같다.

[표 3] SHA 라운드 상수

[Table 3] The round constant of SHA

Step Number	Hexadecimal	Take Integer Part of
$0 \le t \le 19$	K_t = 5A827999	$[2^{30} \times \sqrt{2}]$
$20 \le t \le 39$	K_t = 6ED9EBA1	$[2^{30} \times \sqrt{3}]$
$40 \le t \le 59$	K_t =8F1BBCDC	$[2^{30} \times \sqrt{5}]$
$60 \le t \le 79$	K_t = CA62C1D6	$[2^{30} \times \sqrt{10}]$

4번째 라운드(80번째 단계)의 출력은 CVq+1을 생성하기 위한 첫 번째 라운드 (CVq)에 대한 입력에 더해진다. 덧셈은 덧셈 모듈러 232를 사용한 CVq에서 각각 대응되는 word의 각 다섯 word에 대해 독립적으로 행해진다.

· Step 5 : Output.

모든 L 512 bit 블록이 처리된 뒤에 L 번째 단계로부터의 출력은 160 bit 메시지 다이제스트이다.

SHA-1의 동작은 다음과 같이 요약할 수 있다.

CV_0 = IV

$CV_{q+1} = \sum_{32}(CVq,\ ABCDEq)$

MD = CV_L

여기서,

· IV = 단계3에서 정의된 ABCDE 버퍼의 초기값

ABCDEq = q 번째 메시지 블록의 마지막 라운드 처리의 출력

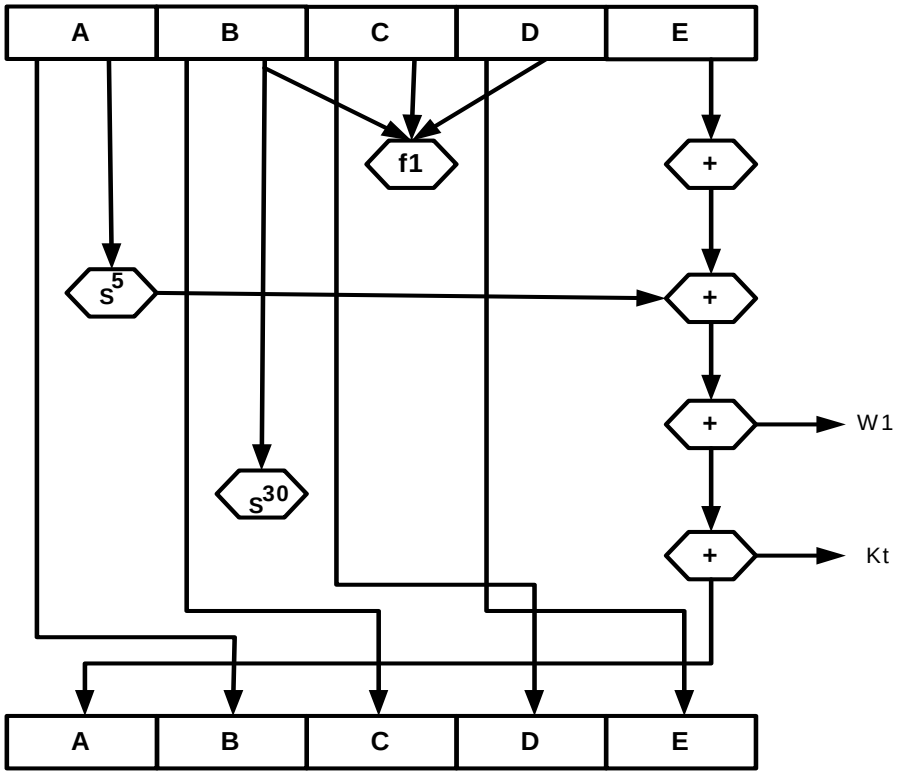
· L = 메시지의 블록의 수 (패딩과 length field포함)

SUM32 = 덧셈 모듈러 232는 각 원드의 입력 쌍을 개별적으로 수행한다.

· MD = 마지막 메시지 다이제스트 값

각 기초 함수는 입력으로 세 가지 32 bit word를 가지고 32 bit word를 출력으로 생성한다.

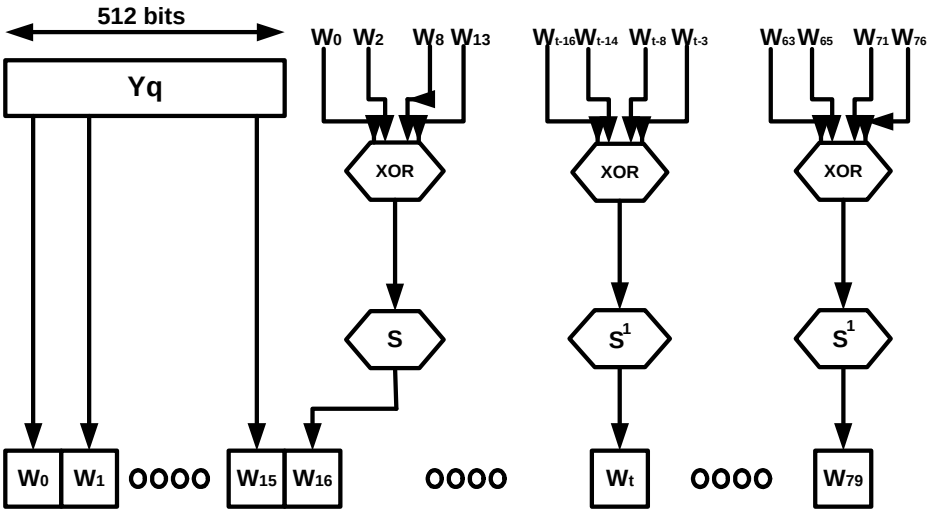
각 함수는 논리 연산 비트 쌍의 집합을 수행한다. 그것은 출력의 n 번째 bit가 3가지 입력의 n 번째 bit의 함수이란 것을 말한다.



[그림 10] SHA-1의 기본 동작

[Fig. 10] The basic movement of SHA-1

[그림 11]은 사상을 나타낸다. 첫 번째 처리의 16단계에서 Wt의 값은 메시지 박스의 대응되는 word와 같다. 남아 있는 64 단계에 대해 Wt의 값은 Wt의 4 가지 이전 값들의 XOR의 하나의 비트에 의해 순환 left shift 를 구성한다. 이것은 양쪽 모두 메시지 블록의 16워드 중 하나를 직접적으로 각 단계함수에 대한 입력으로 사용하는 MD5와 RIP등-160 다른 주목할 만한 점이다.



[그림 11] 단일 블록의 SHA-1 처리에서 80-단어 입력생성

[Fig. 11] The 80-word input creations from SHA-1 control of single unit block

SHA-1은 16블록 워드를 압축 함수에서 사용하기 위해 80워드로 확장한다. 이것은 다량의 감소와 압축된 메시지 블록으로의 상호의존을 소개한다. 이것은 다음과 같은 압축 함수 출력으로 사상 되는 다른 메시지 블록을 찾는 작업을 복잡하게 한다.

가. Unconditionally secure cryptosystem : 아무리 많은 시간과 분석 자원이 암호분석을 위해 제공되더라도 암호시스템의 분석이 불가능한 경우

나. Computationally secure scyptosystem : 실용적인 측면을 고려하여 암호시스템을 분석하는데 소요되는 비용이 그 시스템에 의해서 암호화된 정보의 가치를 훨씬 초과하거나 또는 그 시스템 분석에 소요되는 시간이 암호화된 정보가 유용함을 상실 할 때까지의 시간을 초과하는 조건을 만족하는 경우 DRM은 암호 기술을 기본으로 콘텐츠를 보호하기 때문에 무엇보다도 키의 배포 및 관리 그리고 이러한 키를 어떻게 활용하느냐 하는 것이 매우 중요하다.

많은 사람들은 키의 관리를 PKI에서 사용하는 방식으로 사용해야 한다고 생각하고 있다. 그러나 키의 배포 및 관리에 있어서 PKI와 DRM은 서로 다른 기술적 특성을 가지고 있기 때문에 이것을 제대로 이해하지 않고는 DRM의 효율성을 제대로 보장하기 어려운 문제가 있다.

특히 PKI 기반의 암호화 응용 시스템에서는 사용자를 어떻게 인증할 것인가의 문제점을 해결하기 위해 출발한 기술이고 PKI에서는 공개키의 신뢰성을 보장하기 위해 공인인증기관을 통해 신뢰성을 보장하고 있다.

그러나 DRM은 특정 콘텐츠를 허가된 사용자에 한해서 허가된 용도로만 이용될 수 있도록 통제하는 것을 목적으로 하고 있기 때문에 PKI 기반의 암호화 키 관리 방식과는 매우 다르게 접근하여야 한다[39][45][46].

DRM에서의 키 관리는 콘텐츠에 대한 사용 권한을 통제할 수 있는 수단으로 사용한다. 즉 DRM에서의 키 관리는 콘텐츠의 접근 제어 및 권한 제어를 위해 사용되는 유일한 통제 수단이 되며 이것은 전적으로 DRM 공급업체의 기술적 특성에 따라 고유하게 관리된다. 이런 관점에서 DRM의 키 관리는 PKI 기반의 암호화 시스템과는 다른 차이점을 보이고 있다.

제 4 절 키 관리 기술 (*Key Management*)

DRM은 허가되지 않은 사용자로부터 콘텐츠의 접근을 통제하기 위하여 암호화된 콘텐츠를 복호화 할 수 있는 비밀 키의 배포 통제를 통해 구현한다. 이때 비밀 키는 콘텐츠의 비밀성을 유지하기 위해서 매우 안전하게 관리되고 배포되지 않으면 안 된다[50][54]. [표 4]와 같이 암호화된 콘텐츠의 안전도는 암호화에 사용된 키의 길이와 비례한다. 그러나 암호화에 사용된 키의 길이가 크다고 해서 콘텐츠의 보안이 완벽하게 이루어지는 것은 아니다. 콘텐츠의 완벽한 보안을 위해 가장 중요한 것은 이러한 키를 얼마나 안전하게 관리하느냐에 달려 있다고 할 수 있다.

[표 4] 암호화키의 길이에 대한 해독 소요시간

[Table 4] The decoding requirement time against the length of the encryption-key

키길이(bit)	개인	소그룹	대학수준의 Network	대기업	국가규모의 그룹
40 bit	1주 이상	하루이상	1시간 이상	1/1000초	1/100000초이상
56 bit	100년 이상	10년 이상	1년 이상	1시간 이상	1초 이상
64 bit	1000년 이상	100년 이상	10년 이상	1일 이상	1분 이상
80 bit	불가능	불가능	불가능	100년 이상	100년 이상
128 bit	불가능	불가능	불가능	불가능	1000년 이상

DRM 업체들은 저마다 고유한 키 관리방식을 개발하여 사용하고 있는데 보안상 가장 중요한 부분으로 간주되기 때문에 자세한 키 관리 방식을 밝히는 것은 금기 시 하고 있다. 그러나 키 관리의 방식은 크게 중앙 집중 키 관리 방식과 동봉(Enveloping) 키 관리 방식 등으로 구분할 수 있다.

가. 중앙 집중 키 관리 방식은 콘텐츠를 암호화할 때 사용하였던 비밀키를 중앙의 키 관리시스템에 보관하고 라이선스를 발급할 때 이를 이용하는 방식으로 초기의 DRM 기술들이 사용했던 방식이다.

나. Enveloping 키 관리 방식은 콘텐츠를 암호화 할 때 사용했던 암호화 키를 암호화된 콘텐츠와 함께 동봉해서 관리하는 방식이다. 암호화 키가 콘텐츠와 함께 저장되어 있기 때문에 보안성이 취약할 것이라고 생각할 수 있으나 동봉되는 암호화 키는 특정한 사용자 또는 시스템 비대칭 공개키를 이용하여 암호화되기 때문에 충분한 안전성이 보장된다. 이렇게 동봉 키 관리 방식을 이용하는 주요 이유는 암호화 키의 저장 및 배포에 대한 안전성과 관리의 용이성을 최대화하기 위한 것이다[47][48].

1. 패키저 (Packager)

패키지는 지적 자산의 보호 대상인 콘텐츠를 허가된 사용자만이 이용할 수 있도록 암호화해서 Secure Container의 파일 포맷으로 패키징하는 작업을 수행한다. 패키지는 콘텐츠를 어느 시점에 패키징을 하느냐에 따라 다음과 같이 구분될 수 있다.

가. 사전 패키징(*Pre-packing*) 방식

사전 패키징 방식은 사용자의 요청이 있기 전에 미리 콘텐츠를 Secure Container로 패키징 하는 방식을 말한다. 이렇게 미리 패키징 된 Secure Container는 특정 사용자와 무관하게 패키징 되기 때문에 동시에 많은 사용자가 콘텐츠를 이용하더라도 패키징의 부하 부담을 피할 수 있게 된다. 만약 사

용자의 요청에 따라 패키징을 하게 되면 동시 사용자의 규모에 따라 시스템의 성능이 큰 영향을 받게 된다. 특히 패키징 시에 필수적으로 요구되는 암호화 과정은 CPU의 부담을 크게 주기 때문에 전체적인 시스템의 성능을 떨어뜨리는 결정적인 원인으로 작용한다. 따라서 DRM의 패키징 작업은 사용자의 이용정도에 무관하게 독립적으로 생성할 수 있는 기술적 메커니즘을 제공해야 한다.

나. 주문형 패키징(*ON-THE-fly packaging*)방식

주문형 패키징 방식은 사용자의 요청에 의해 다이내믹하게 콘텐츠를 생성해서 배포해야 되는 콘텐츠를 패키징 할 때 사용하는 방식이다. 이 방식은 동시 사용자가 많을 경우에 암호화로 인해 시스템의 부하가 급속하게 증가하고 전체적인 시스템의 성능을 현격하게 떨어뜨리는 현상을 야기하게 된다. 따라서 이 방식을 이용해서 콘텐츠를 패키징하는 DRM 제품은 동시 사용자의 증가에 따른 시스템의 병렬적 확장이 용이하고 경제적인 비용으로 제공될 수 있도록 구조적 특징을 제공해야 한다.

2. Secure container

Secure container는 저작권 보호 대상인 원본 콘텐츠를 암호화해서 유통하기 위해 사용되는 전자적 장치로 다음과 같은 기능적 요소들을 만족하여야 한다.

가. 기능성

다양한 콘텐츠 타입을 지원하고 단수 또는 복수 개의 콘텐츠를 패키징할 수 있어야 하며 메타데이터 관리가 용이해야 한다.

나. 보안성

인터넷을 통해 유통되는 과정에서 위 · 변조되지 않았음을 증명할 수 있도록 무결성(integrity)과 유통되는 과정에서 허가되지 않은 사용자로부터 콘텐츠의 내용이 차단될 수 있도록 비밀성(confidentiality)를 보장해야 한다.

다. 배포 용이성

온라인을 통해 콘텐츠를 다운로드 또는 스트리밍으로 배포 및 이용할 수 있어야 하며, CP, DVD, diskette 등과 같은 오프라인 전송매체를 통해 콘텐츠 배포와 소비자 간의 콘텐츠 재배포 및 이용 활성화를 위해 superdistribution을 지원할 수 있어야 한다.

지금까지 언급한 항목들을 만족하기 위해서 Secure Container는 콘텐츠의 비밀성을 보장하면서 충분히 유연하고 확장성 있는 구조를 갖지 않으면 안 된다.

3. 콘텐츠 식별 체계(Identification)

디지털 콘텐츠의 식별체계는 디지털 콘텐츠로 하여금 유일한 식별체계를 갖도록 하며 콘텐츠의 유통과정에서 권리 소유자의 결정 및 권리 표현을 연계하는 등 많은 애플리케이션에서 중요한 역할을 담당하게 된다.

현존하는 많은 DRM 제품들도 이러한 용도로 사용되는 각자 고유의 식별관리 체계를 가지고 있으나 DRM 제품들 간의 식별체계에 대한 호환성이 없다. 이러한 이유는 현재의 DRM 제품들이 콘텐츠 식별체계를 콘텐츠의 식별을 위한 용도로만 사용할 뿐만 아니라 콘텐츠의 보호 범위가 특정 DRM 제품의 적용범위로만 국한되기 때문이다.

그러나 디지털 콘텐츠의 유통 및 관리가 지엽적인 범위에서만 국한되지 않고 범용적인 도메인에서 적용 가능키 위해서는 국제적으로 통용되는 표준 식별계의 구축이 필요하다. 이러한 목적을 달성하기 위해 여러 개의 국제 표준기구에서 디지털 콘텐츠의 식별체계를 제시하고 있다.

4. DOI(Digital Object Identifier)

DOI 는 1997년 미국 출판 협회(AAP : Association of American Publisher)가 주축이 되어 디지털 콘텐츠의 식별체계 및 저작권 정보를 관리하기 위해 개발된 시스템이다.

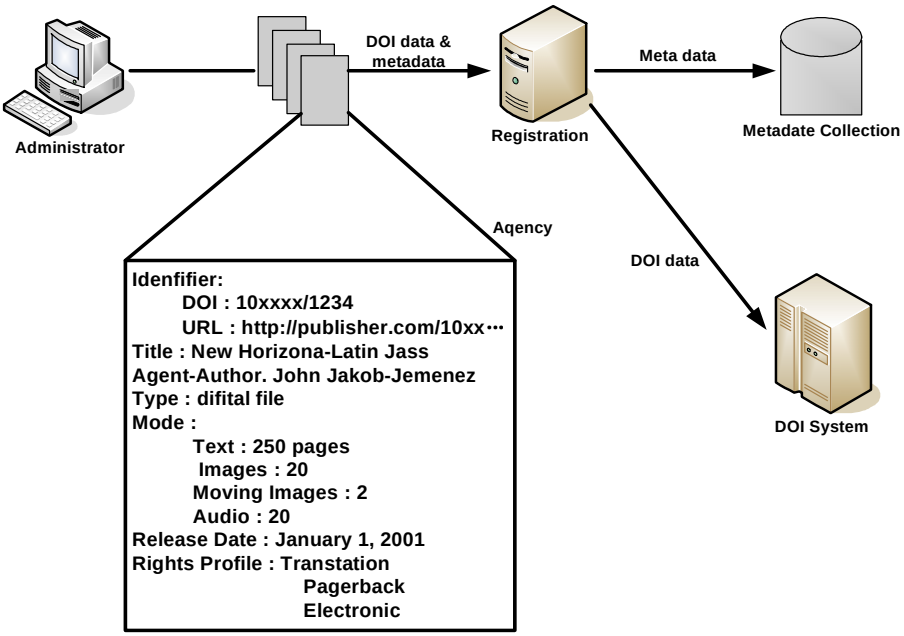
디지털 객체 식별자DOI(Digital Object Identifier)는 디지털 정보자원의 고유한 식별기호로서 기존의 정보자원에 대한 식별기호체계를 포괄하는 총괄적인 체계이다. 또한 INDECS(INteroperability of Data in E-commerce Systems)와 같은 메타데이터와 연계함으로써 지적소유권이 인정되는 콘텐츠의 거래내역 확인이 가능하며, 거래내역의 확인을 통한 거래의 투명성을 확보할 수 있다. DOI는 시스템 운영의 효율을 높이기 위해 다음과 같은 디지털 콘텐츠식별체계를 이용한다[37][59].

- URI(Uniform Resource Identifier): 인터넷상의 정보자원을 식별하기 위한 문자열로서 HTTP, FTP와 같은 다양한 환경에서 이들 정보자원을 이용 할 수 있도록 하기 위한 구조로 URL 과 URN 으로 그 유형을 구분한다.

-URL(Uniform Resource Location) : 실제 프로토콜을 사용하여 탐색할 수 있는 물리적 주소, 네트워크 자원에 대한 참조를 융통성 있고 쉽게 이행 및 확장이 가능한 방법이나 논리적인 내용물이 아닌 객체 접근방법에 대한 단순한 지침으로서 자원이 이동할 경우에는 문제가 발생한다.

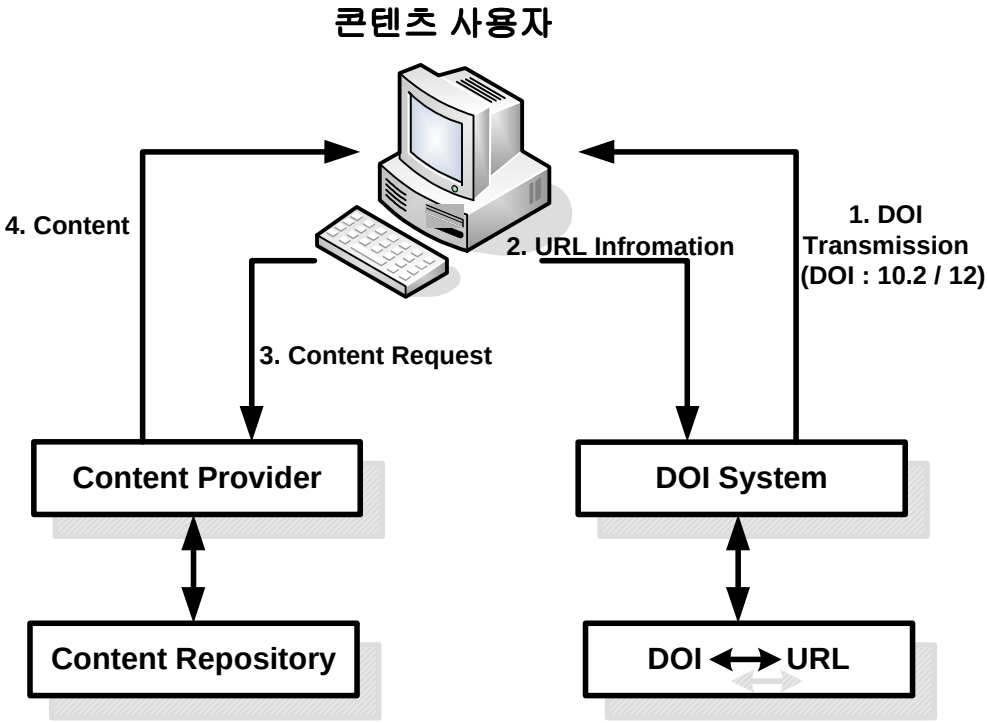
- URN(Uniform Resource Name) : URL의 단점을 보완하며 정보자원의 수명이나 관련된 모든 기구의 존폐와 관계없이 특정 정보자원에 대한 참조를 가능하게 한다.

- URC(Uniform Resource Characteristics) : URN 과 URL 을 연결해 주는 데이터 구조로서 URN 을 이용하여 인터넷 정보자원의 소재를 파악하고 URL 을 이용하여 탐색할 수 있는 시스템이다. [그림 12]은 콘텐츠의 권리자가 RA 시스템을 통해 등록된 DOI 식별번호와 URL 정보는 DOI 시스템에 저장 관리되며 콘텐츠 사용자는 [그림 13]과 같이 DOI에 대한 URL 정보 변환 요청을 통해 실제 콘텐츠의 접근 정보를 얻어서 콘텐츠를 이용할 수 있게 된다.



[그림 12] DOI 등록 프로세서

[Fig. 12] The processor of DOI registration



[그림 13] DOI 변환 프로세서

[Fig. 13] The processor of DOI conversion

제 3 장 P2P기반 개선된 DRM 시스템 설계

제 1절 P2P 기반 DRM 기능의 고려 사항

그누텔라(Gnutella)와 같은 P2P 구조는 피어(peer)들 간의 파일 공유를 직접 하기 때문에 서버를 사용하지 않고 피어들끼리 정보를 바로 바로 교환할 수 있다. 따라서 서버를 거치는 경우보다 훨씬 빠른 최근의 정보를 찾을 수 있으며, 파일을 인덱스 할 필요도 없다. 그러나 P2P를 통한 공유는 저작권 침해의 온상으로 저작권 관리 기술 및 여러 요구사항들이 반드시 필요하다. 라이선스 기법, 높은 분산 컴퓨팅 지원 그리고 콘텐츠를 발생하는데 있어서의 절차의 간소화 등 P2P 콘텐츠 전송에 적합한 DRM 기능을 고려해야 한다.

P2P 기반 DRM을 적용하는데 있어 가장 중요한 사항은 다음과 같다. 특히 주요 기능들을 저작권 보호를 고려하여 설계해야 한다.

첫째, 각 콘텐츠 제공자가 독립적으로 운용될 수 있는 형태

둘째, 제공자가 필요에 따라 결합 또는 분리될 수 있는 형태

셋째, 제공자 간의 상호 인증

넷째, 라이선스 서버로부터 콘텐츠 보호

다섯째, 클라이언트로부터 콘텐츠 보호

여섯째, 콘텐츠 종류, 메타데이터 및 시스템 확장 가능한 형태를 고려하여야 한다. 콘텐츠 제공자들은 Provider, Retailer(유통 저작권자), DRM Server, Client로 구성되어 있다.

1. 콘텐츠 제공자

저작권자와 계약에 따라 유료 전송 할 수 있는 콘텐츠를 생성하고 저작권자에서 콘텐츠를 제공한다. 제공자는 저작권자에게 수익의 로열티를 분배 받는다.

2. DRM Server

DRM Server는 사용자 인증과 사용자의 구매 요청에 따라 결제를 수행하고 사용자에게 라이선스를 발급하며, 유료 전송자에게 정산 정보를 리포팅하는 등, DRM 시스템의 핵심적인 역할을 수행한다. DRM Server는 사용자 인증, 단말기 인증 및 콘텐츠 인증을 위한 인증 모듈과 라이선스 발급, 정산 및 DB 등의 관리를 하는 관리 모듈로 구성된다. 인증 모듈은 공개키 암호 알고리즘인 RSA와 해쉬함수를 이용하여 사용자 인증, 전자 서명 기능을 제공하고 다음과 같은 기능을 갖도록 설계한다.

첫째, 사용자 및 단말기 인증 기능

둘째, 사용자 등록 관리 기능

셋째, 인증서 발급 관리 기능

넷째, 인증서 처리 기능

다섯째, 키 관리 기능

여섯째, 라이선스 생성 관리 기능 등이다

3. 패키지

콘텐츠를 암호알고리즘을 이용하여 패키징 역할을 담당한다. 패키지 모듈과 암호 모듈로 구성되어 있으며 정통 암호 알고리즘의 표준으로 사용되고 있는 AES와 SEED를 사용하여 콘텐츠 보호를 위하여 암호화 패키징을 한다.

4. 클라이언트

유로 콘텐츠를 사용하는 주체이며, 클라이언트의 인증 정보를 이용하여 인증을 받기 위한 인증 모듈과 콘텐츠의 복호화를 위한 보호모듈, 콘텐츠의 실행을 위한 Player, 콘텐츠의 다운로드 및 라이선스 발급 등의 외부와의 모든 기능을 관리하는 관리 모듈로 구성된다. 또한 패키지는 콘텐츠의 정보와 사용 권한 등의 부가 정보를 포함하여 패키징하고 암호화하는 기능을 가지고 있다. 본 논문에서는 각각의 시스템은 독립적으로 동작하도록 설계하였으며 설계 과정은 Clearinghouse 설계와 Client 설계, 인증 프로토콜 설계로 나누어진다.

제 2 절 기존의 DRM 솔루션 분석

InterTrust사와 Microsoft사의 DRM 솔루션들의 저작물의 생성규칙, 암호화와 키의 분배, 인증 및 사용 내역 관리, 유통 및 추적기능 등에 대하여 비교 분석하면 [표 5]와 같다. 그러나 여기서 비교 분석된 기술들도 계속 변화 발전하고 있는 중이며 공개된 기술 내용에는 많은 제약이 있어 완벽한 비교 분석을 하는 데에는 한계가 있다.

저작물 사용 규칙의 정의는 사용 규칙을 얼마나 다양하게 제어할 수 있는가 하는 것이 중요한 문제이다. 현재 개발된 기술들의 대부분은 사용 횟수, 사용 기간, 사용자의 나이 등에 따라 저작물을 실행할 수 있도록 허용하는 저작물의 사용규칙을 정의한다[38][50].

InterTrust사에서는 사용 규칙을 정의하기 위한 언어로 표준 명세 언어인 XrML을 사용하고 있으나 두 회사 모두 저작 권리에 대한 내용을 공개하지 않고 자체 데이터 구조로 저작권 정보를 구축하고 있다. 그러므로 향후 디지털 저작 권리에 대한 명시 규칙을 XrML을 사용하여 표준화한다면 사용규칙을 작성하는 도구 및 저작물 응용 시스템의 상호 호환성이 보장될 수 있을 것이다.

[표 5] DRM 시스템 기술 분석 비교

[Table 5] The technical analysis comparison of DRM system

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
인증 처리	배포자로부터 InterRights Point를 다운받아 처리	WMRM이 클리어링하우스를 통해 라이선스 키 획득
사용내역 저장	사용자 컴퓨터	클리어링하우스
결재 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 9 플리에어에 종속

저작물의 패키징 작업은 InterTrust DRM의 경우는 출판업자를 통해서 패키징 처리를 하며 저작물 및 저작권 정보, 사용 권한 등이 암호화되어 함께 DigiBox Container에 저장된다. Microsoft사의 경우는 WMRM을 통해 암호화되어 저장된다. 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 없어 필요시 재 패키징을 해야 하는 문제점이 있다. 그러므로 저작권 정보를 라이선스에 포함하여 라이선스에 의한 관리를 하는 것과 비교해서 저작권의 정적인 관리가 불가피하다.

DRM 솔루션에 적용된 주요기술의 경우 InterTrust사는 암호화, 워터마킹 기술을 적용하고 있으며 Microsoft사는 암호 기술을 활용하고 있다. 그러나 암호 기술의 경우 적법한 사용자가 플레이어를 통하여 평문을 획득했을 경우, 이를 무단으로 복사하여 배포하는 것을 막을 수가 없는 문제점이 있다.

저작물의 암호화 방식은 전체 시스템의 특성에 따라 많은 영향을 주게 된다. 가장 일반적인 방식이 공개키 암호화 기술을 이용하여 인가된 사용자의 공개키로 암호화하는 방식이다. 그러면 인가된 사용자 외에는 복호화할 수 없으므로 초기 저작권 보호는 잘된다고 할 수 있으나, 이 방법은 저작물을 제공 받을 사람을 미리 알아야 한다는 제약이 있다. 또한 전자상거래에서 통신을 통해서 디지털 저작물을 제공하고자 할 경우 미리 암호화를 할 수 없으므로 불특정 다수에게 제공하려면 공개키 암호화와 복호화에 많은 시간이 소요되는 문제점이 발생하게 된다.

비밀키 암호화 기법을 이용하여 암호화를 수행할 경우 미리 암호화를 해 놓을 수 있는 장점이 있으나 반면 암호화키의 전달, 저장, 변경 등 키 관리가 복잡해지며 사용자가 키를 노출할 경우 추적이 어렵게 된다. InterTrust사에서는 두 방식을 혼용하여 사용하고 있다. 기술적으로는 어렵기는 하지만 이 방식을 선택할 경우 많은 디지털 저작물은 암호화된 방식이 같고 키가 같으므로 사용

자가 다른 사용자에게 암호화된 저작물을 전달할 수 있어 이른바 제 3자에 대한 재분배를 실현할 수 있는 장점이 있다. 그러나 여전히 사용자가 키를 노출했을 경우에는 심각한 보안 위협이 야기될 수 있다. 또 다른 기술적인 관심 사항 중에 하나는 사용 권한 키를 어디에 저장하느냐 하는 문제이다. 서버에 저장할 경우 키 관리의 문제를 단순하게 만들기는 하지만 사용자는 항상 서버와 온라인 상태를 유지해야 하는 단점이 있다.

사용자 컴퓨터에 두는 경우 키가 쉽게 복사되어도 안 되며 쉽게 훼손되어서도 안 된다. 그런데 사용자 컴퓨터에 두었을 때 또 다른 문제점 중에 하나는 사용자가 당초 사용권한을 받은 컴퓨터가 아닌 다른 컴퓨터에서는 저작물을 사용할 수 없다는 점이다. 이 문제를 해결하기 위해서는 스마트 카드(smart card)와 같은 곳에 사용 권한을 저장하는 것이 한 가지 방법이기는 하나 스마트카드 인식기의 보급에 따른 비용, 스마트 카드의 표준화 문제 등으로 인하여 아직은 현실적인 대안이 되지 못하고 있다. InterTrust의 경우는 서버에 키가 저장되어 라이선스의 인증과 함께 키를 다운로드 받으며, Microsoft사의 경우 Clearinghouse를 통해 라이선스 키를 다운로드 받는다.

사용권한 처리를 위한 인증방식의 경우 InterTrust의 경우 배포자로부터 InterRights Point를 다운로드 하여 처리하고 Microsoft사의 경우 Clearinghouse를 통해 라이선스 키를 획득하며 사용자 시스템에 인증 내용을 저장하고 있다.

디지털 저작물의 전자 상거래를 지원하기 위해서는 사용 내역을 관리하는 기능이 필수적이다. 사용 내역을 기록해야 저작물의 사용에 따른 요금 부과와 결재를 수행할 수 있다. 사용 기록은 사용자의 장비에 저장하거나 온라인으로 연결된 서버에 저장할 수 있다. 사용자 장비에 저장할 경우 기록이 변경되거나 삭제되지 않도록 적절한 안전성 보장이 필요하며 저장된 사용 내역의 수집이 자동화 되어야 한다. 서버에 저장할 경우 사용 내역을 처리할 때 마다 네트워크에 연결되어야 하는 문제가 있다.

InterTrust사의 경우는 사용자의 장비에 사용 내역을 남기고 있으며

Microsoft사의 경우 서버의 Clearinghouse에 사용 내역이 저장된다.

디지털 저작물에 대한 전자 상거래를 지원하기 위해서는 결재 기능이 추가되어야만 한다. InterTrust사의 경우 거래 체결은 사용자의 장비에서 저작물을 사용하는 순간에 일어나도록 되어 있어 필요에 따라 신용카드를 이용한 즉시 결재나 전자 화폐를 이용한 다양한 결재 방식을 구현하기 쉽도록 하고 있다. Microsoft사의 경우는 Clearinghouse를 통하여 결재를 수행한 후 사용할 수 있다.

저작물 유통은 사용자가 저작물을 다운로드 받는 방식이다. InterTrust사의 경우 저작물을 사전에 암호화하여 재분배를 하므로 유통이 온라인 및 오프라인을 통하여 가능하나 Microsoft사는 다운로드 시 암호화를 수행하므로 많은 시간이 소요되어 오프라인을 통해서만 다운로드 할 수 있다.

추적 기능 제공 측면에서 기존 DRM 솔루션의 경우 특정 저작물에 대한 법적 분쟁이 발생했을 때 불법 사례 입증에 필요한 기초자료 제공을 위한 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.

제 3 절 기존 *DRM* 모델의 한계

P2P 네트워크에 DRM을 적용하기 위해서는 안전성, 서버에 대한 부담성, 빠르고 쉬운 다운로드 등을 고려하여야 한다. Pure 모델은 모든 DRM 처리가 서버에서 수행되는 반면 암호화된 콘텐츠는 P2P 모드로 전송된다. 사용자 피어들은 DRM 서버에 접근하여 수신된 콘텐츠를 복호화 하기 위해 요청(request) 기능을 장치에 가지게 된다. 그리고 DRM 서버에 라이선스 발행을 요청하는 기능은 반드시 사용자 단말기에 설치되어야 한다. 이 방식에서 서버는 반드시 모든 사용자의 계정을 인증, 지불 및 정산을 통해 관리해야 한다. 암호화와 사용 제어 기능이 모두 DRM 서버에서 관리되며 사용자 인증을 한 서버 역시 센터에 위치하게 된다. 따라서 신뢰성과 안전성은 보장되지만 사용자 수가 증가할수록 서버엔 부담이 커지게 된다. Hybrid 시스템은 본 논문에서 제안하고자 하는 유료든 무료든 다운로드는 쉽고 빠르게 수행하도록 한다. 유료일 경우 안정적이며 DRM 서버에 무리를 주지 않는 가장 이상적인 모델로 Hybrid P2P 기반의 DRM이 적당하다. 그러나 Hybrid 시스템과 DRM 기술을 단순 접목을 시키면 저작권자의 이익과 수익만을 보장하므로 P2P 가장 큰 장점인 정보 공유에 대해 문제점이 발생한다. 또한 DRM 시스템의 목적인 보안 위협에 그대로 노출이 되므로 본 논문에서는 P2P 환경에서 개선된 콘텐츠 보안 시스템 설계를 제안한다.

제 4 절 개선된 DRM 시스템 설계

본 논문에서 제안하는 P2P 시스템에 DRM 기술을 효과적으로 적용하는 것이며 무엇보다 P2P 상에서 피어들 끼리의 수익성을 위해 재분배를 통해 유료 콘텐츠의 수익만을 위한 모델이 아니다.

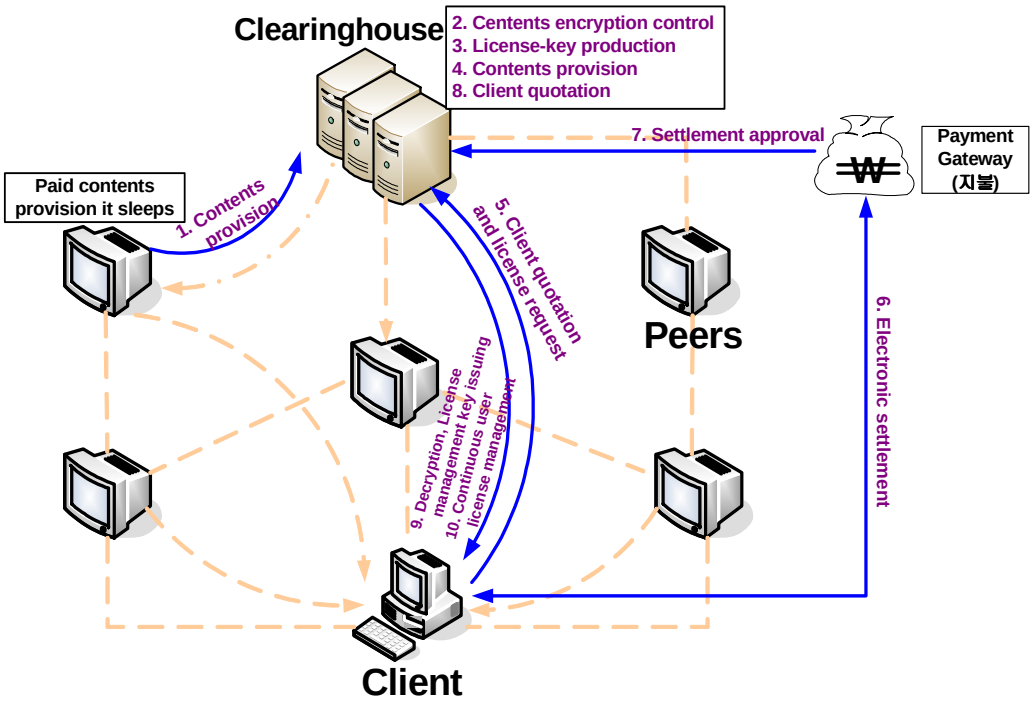
본 논문에서 제안하는 P2P 환경에 적용한 DRM 시스템은 분산컴퓨팅의 성능을 최대한 살리고 사용자 증가에 따른 엄청난 부하는 일어나지 않도록 하며 정보 공유 시 일어날 수 있는 모든 보안 위협 요소를 해결하고자 한다.

기존 DRM 시스템은 Client들을 Server에 접속하는 중앙 집중 방식을 지향하는 시스템 기술이며 중앙 서버가 존재한다. 제안하고자 하는 P2P환경의 DRM 시스템은 분산처리 방식으로 중앙 서버 없이 Peer들 간의 쌍방향 통신이 가능하다. DRM의 경우는 라이선스 발급을 위한 Clearinghouse와 인증 서버 그리고 콘텐츠 암호화를 제공하는 패키저 등이 핵심 기술이다. 이런 요소들은 단순 접목의 P2P 시스템의 피어들 간에 구현하기에는 곤란하다. 현 DRM 시스템이 인증과 과금 정보에 있어 중앙 서버를 따로 두는 경우처럼 P2P 네트워크에 적용하려면 서버/클라이언트 모델의 장점인 신뢰성과 안정성 및 중앙 서버에 의한 시스템 전체 관리 가능 또한 중앙 서버의 다중화를 통해 Peer 장치간 동적 네트워크 구성이 가능하도록 하여 신뢰성을 향상시키도록 해야 한다. 네트워크 관리 측면에서 보면 DRM은 콘텐츠 사용에 대한 라이선스 발행과 관리를 Clearinghouse에서 이행한다.

[그림 14]는 DRM 유료관리 시스템의 P2P기반 개선된 DRM 시스템 설계 내용을 도해한 것이다. DRM Server는 크게 암·복호 모듈, 라이선스 관리 모듈, 키 관리 모듈, 거래 관리 모듈 등으로 구성된다. DRM Client는 라이선스 키 관리 모듈, 복호화 모듈 등으로 구성된다. [그림 14]에서 DRM Server는 콘텐츠 제공자로부터 받은 파일을 암호화 처리하며, 라이선스 키 제작을 하고 온라인으로 유료 전송을 담당하여 Client 로부터 콘텐츠 요청이 있을 경우 결제를 통하여 승인 여부를 체크한다. 다음 콘텐츠는 암호화 모듈을 이용하여 암호화하고 이 정보를 DB에 기록하여 DRM Client가 이 정보를 통하여 사용자 라이선

스 체크 여부를 결정할수 있게 하는 역할을 한다.

DRM Client는 DRM Server 측에서 다운로드 받은 암호화된 파일을 라이선스 획득 여부에 따라 복호화 시켜주며 복호화와 동시에 사용자 하드웨어 정보를 Server로 전송한다. 이 사용자 하드웨어 정보를 이용하여 사용 여부를 결정해 주며 계속적인 사용자 관리를 가능하게 한다.



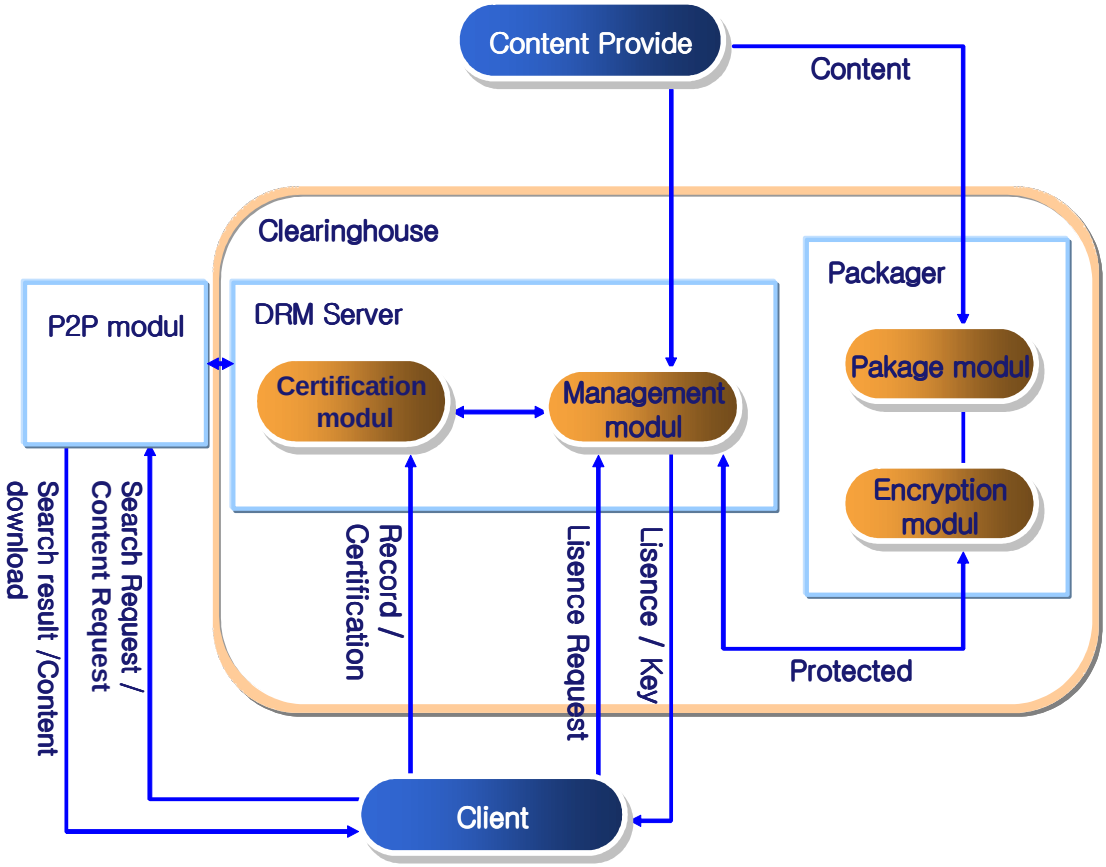
[그림 14] P2P 기반 DRM 시스템 설계

[Fig. 14] The DRM system design of P2P-base

1. Clearinghouse 설계

콘텐츠의 사용 권한을 부여하고 이에 대한 지속적인 관리를 담당하는 시스템이 필요하게 되는데 이러한 역할을 담당하는 것이 Clearinghouse이다. Clearinghouse는 사용자의 콘텐츠 결제 처리를 담당하는 Financial Clearinghouse와 사용권한 정보가 담겨 있는 라이선스를 발급 및 관리하는 License Clearinghouse, 그리고 사용자의 콘텐츠 사용내역을 수집하여 이를 통계 처리하는 Usage Clearinghouse로 크게 구분할 수 있다.

본 논문에서 제안한 개선된 DRM 시스템은 [그림 15]와 같이 하나의 서버에서 콘텐츠 생성 관리, 일괄 처리 할 수 있도록 DRM Server와 Packager, Retailer의 기능을 하나의 Clearinghouse에 포함시켰다. 그러나 DRM Server 와 Packager, Retailer는 독립적인 시스템으로 동작되도록 하였다. 그리고 인증 모듈에서는 콘텐츠의 불법 사용을 방지하기 위하여 사용자의 인증 정보와 하드웨어 바인딩 기법으로 사용자의 하드웨어 정보를 이용하여 인증을 받는다. 하드웨어 바인딩 기법은 사용자 단말기 네트워크 카드의 MAC Address를 해쉬 함수에 적용한 값을 이용하는 방법으로 사용 가능한 단말기를 구분하도록 하였다. 그리고 콘텐츠의 암호화와 패키징을 위하여 128 비트의 SEED 암호알고리즘과 128비트의 AES 암호 알고리즘을 사용 하도록 설계하였다.



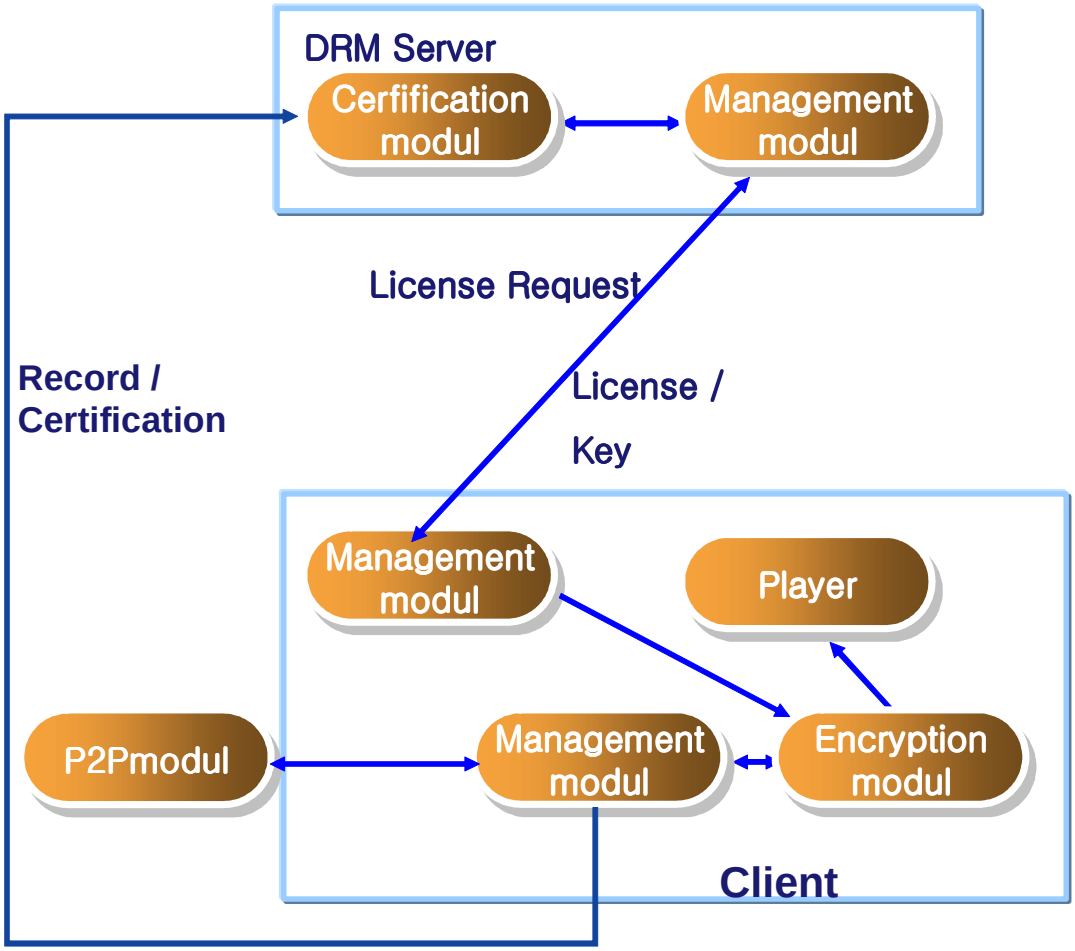
[그림 15] Clearinghouse 설계

[Fig. 15] The design of clearinghouse

2. DRM Client 설계

클라이언트는 다운로드 받은 콘텐츠의 실행을 위하여 어플리케이션 영역에서 복호화기능을 지원함으로써 디지털 콘텐츠에 대한 보안 서비스 강화 및 정당한 절차를 거쳐 라이선스를 구매한 사용자에 대한 인증 기능을 수행하고, Player로 실행하는 기능을 담당한다. DRM 시스템에서 클라이언트는 사용자 단말기에서 동작되므로 DRM 시스템에 손상을 줄 수 있다. 따라서 적절한 보안 레벨을 제공해서 클라이언트 환경을 안전하고, 시스템을 손상시키려는 의도를 어렵게 해야 한다.

본 논문에서 제안한 DRM 시스템의 클라이언트는 [그림 16]과 같이 구성된다. 악의가 있는 불법자로부터 클라이언트 모듈 내에 저장된 값들을 보호하기 위해 클라이언트 모듈에 저장되는 데이터는 모두 암호화된다. 클라이언트는 DRM 서버와 같이 인증 모듈, 관리 모듈, 암호 모듈 및 Player가 포함된다. 사용자는 실제 DRM 서버로부터 콘텐츠를 실행할 때 콘텐츠에 대한 규칙을 확인하고 사용 권한에 대한 라이선스를 획득한다.

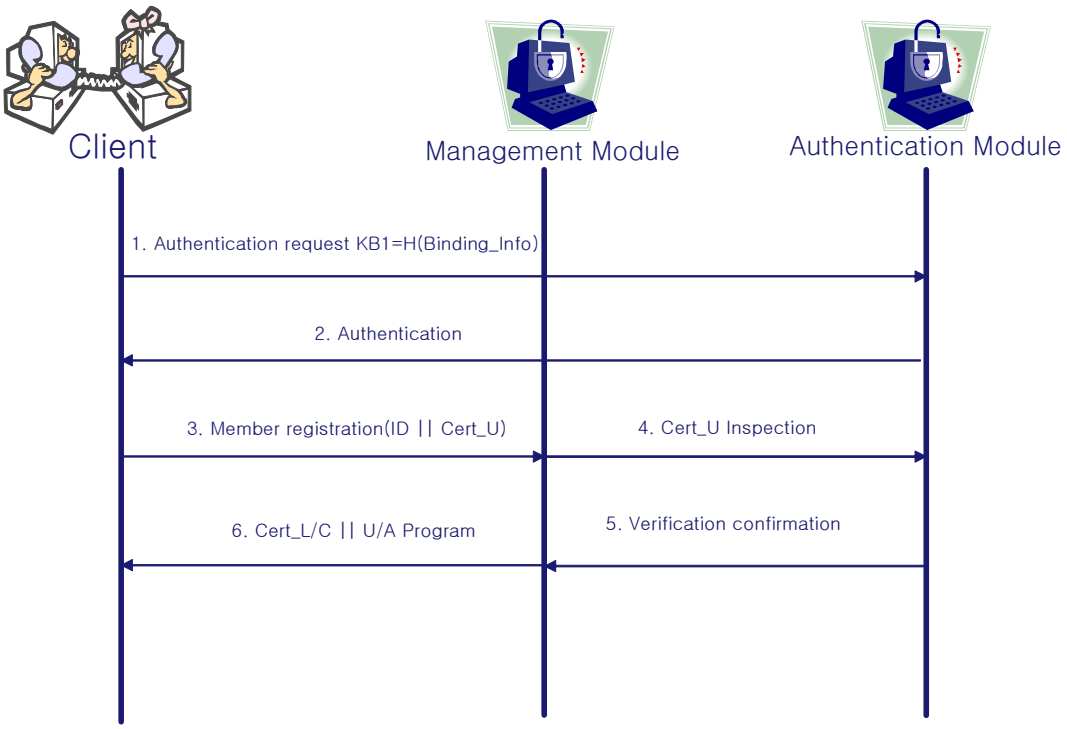


[그림 16] Client 설계

[Fig. 16] The design of client

(1) 인증 프로토콜

콘텐츠를 사용하기 위해서는 [그림 17]과 같이 사용자 인증을 받아야 한다. 사용자 인증은 회원 가입을 통해 이루어지지만 본 논문에서는 파일 다운로드 후 복호화 기법을 사용하였기 때문에 불법 사용자가 정당한 사용자의 정보를 획득하여 손쉽게 복호화 하지 못하도록 한층 더 보안 강화하였다. 특히 정당한 사용자의 컴퓨터의 하드웨어 정보까지 인증하도록 하는 하드웨어 바인딩 기법을 사용하기 때문에 회원 가입과 동시에 클라이언트 네트워크 카드 주소까지 함께 전송을 시켜 인증을 하도록 하였다. 콘텐츠를 다운로드 받은 사용자는 누구든지 회원에 가입할 수 있으며 이를 위해서 PKI 기반의 인증서를 통한 로그인을 지원한다. ID와 암호기반의 로그인이나 인증서 기반의(Kb1=h(Binding_info)) 로그인 과정을 통하여 사용자 인증을 받으면 서버는 인증서 Cert_u를 검증 후 올바른 인증서이면 사용자에게 인증이 성공적이라는 것을 알려주게 된다.

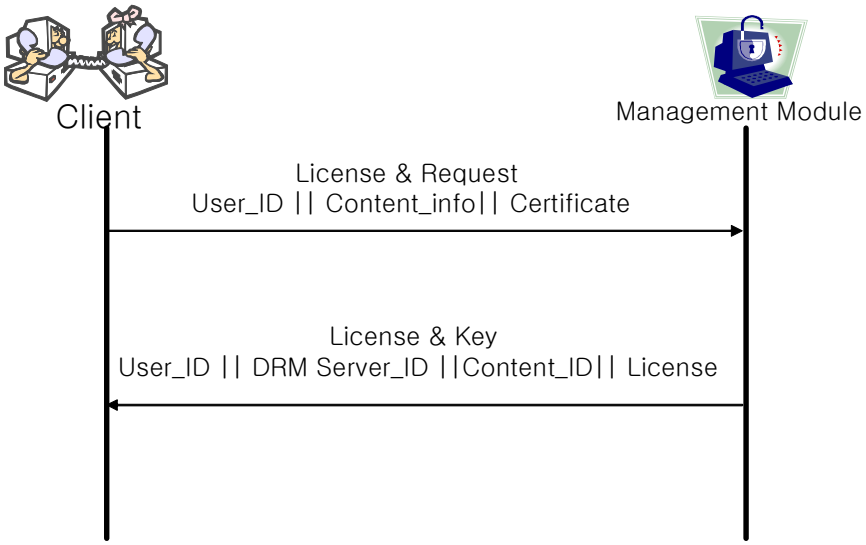


[그림 17] 인증 프로토콜

[Fig. 17] The authentication protocol

(2) 라이선스 프로토콜

사용자는 콘텐츠를 복호화하기 위하여 라이선스를 발급받아야 한다. [그림 18]과 같이 라이선스가 있다면 서버를 통하여 라이선스를 인증하고, 없다면 서버에 접속하여 서버에서 사용자 인증 후 발급한다. 사용자는 콘텐츠의 라이선스를 발급 받기 위해 라이선스 요청 메시지를 서버에 전송한다. 라이선스 요청 메시지에는 사용자 ID, 라이선스 ID, 유효기간, 라이선스 검증주기, 사용자의 공개키 정보, 서버 식별 정보 및 권한 등 인증서를 사용자의 개인키로 서명하고 이를 서버의 공개키로 암호화 값을 전송하여 라이선스 발급 요청을 한다. 서버는 해당 인증서의 정보에 의하여 공유키에서 개인 정보를 추출하고 추출된 개인정보와 암호화된 동영상의 비밀키를 공유키에 저장한 암호화된 공유키에 라이선스와 함께 사용자에게 전송한다. 라이선스 프로토콜 수행 과정은 [그림 18]과 같다.



[그림 18] 라이선스 프로토콜

[Fig. 18] The license protocol

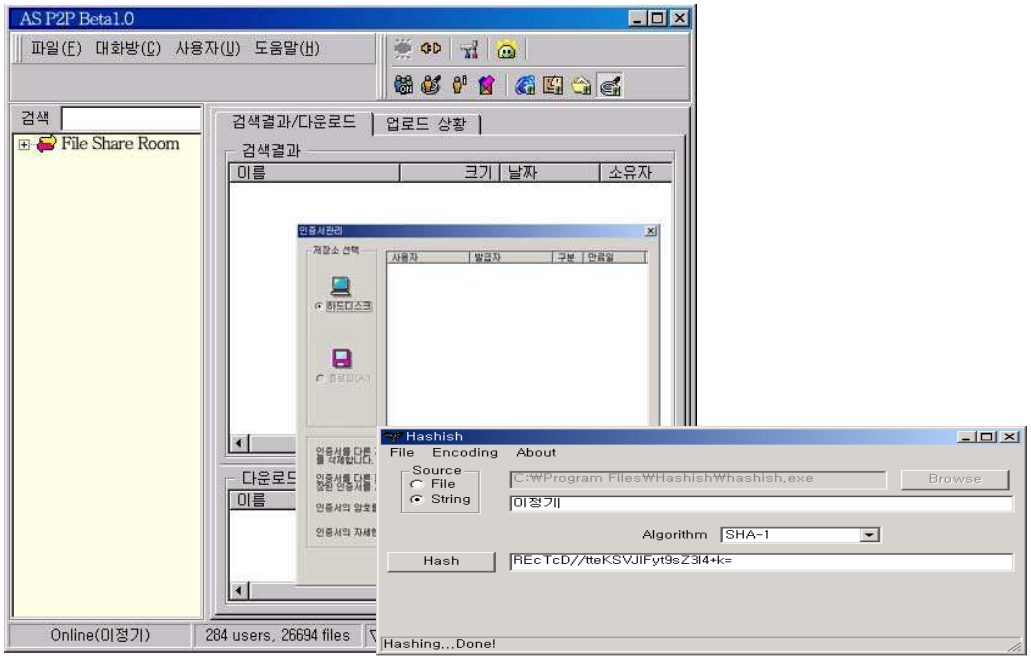
제 4 장 시스템 구현 및 분석

제 1 절 DRM 시스템 구현

제안된 시스템은 P2P 시스템 환경에서 개선된 콘텐츠 보안 시스템을 설계하고 구현 하였다. 제안된 시스템은 피어끼리 파일을 공유할 수 있는 P2P 환경과 인증과 라이선스를 위해 클라이언트-서버 구조로 되어 있다.

제안된 시스템 개발을 위한 환경은 시스템 Inter(R) Pentium-Ⅵ CPU 2.6GHz 와 1GB의 RAM, 그리고 clearinghouse Operation System : Windows 2000 Server를 사용하였다. 또한 프로그래밍을 위한 구현 언어는 Visual C++, Visual Basic, XML을 사용하였고 데이터베이스의 처리를 위하여 MS-SQL 2000 프로그램을 사용하였다.

DRM Client 는 SEED암호 알고리즘과 AES암호 알고리즘을 선택하여 암호화를 수행할 수 있도록 구성하였다. 특히 인증을 받기 위한 인증 모듈, AES, SEED 의 복호화 모듈과 SHA-1의 복호화 해쉬 모듈, 이를 통합적으로 관리하는 관리 모듈로 구현하였다. [그림 19]는 클라이언트에서 파일 다운로드를 받기 위한 P2P 시스템과 서버에 사용자 인증을 받기 위한 인증 시스템을 보인 것이다.

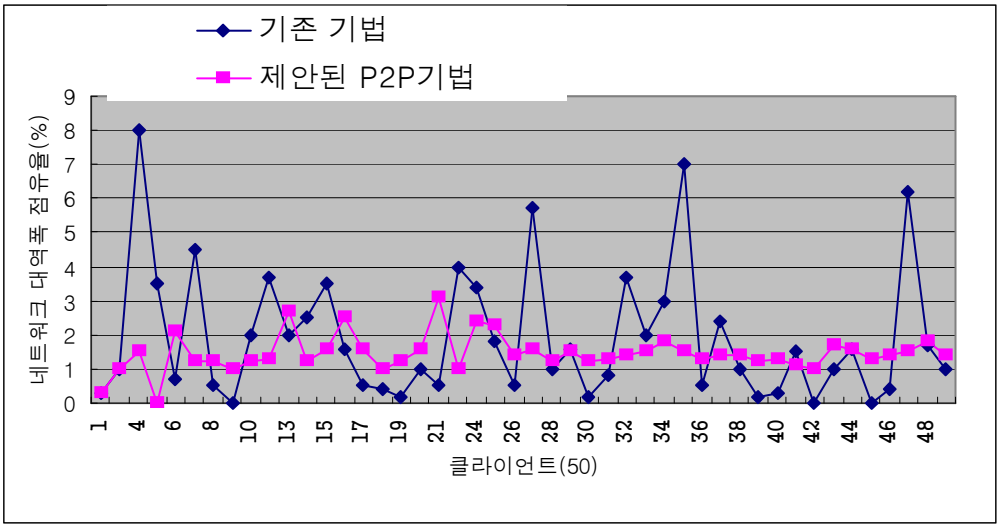


[그림 19] 클라이언트 환경

[Fig. 19] The environment of Client

제 2 절 성능 평가

제안된 시스템의 성능을 측정하기 위하여 네트워크 대역폭 측정을 하였다. [그림 20]은 클라이언트들이 다운로드 시작 후 60초 동안 네트워크 대역폭을 사용한 평균 비율을 측정한 그래프이다. 다운로드는 50대의 클라이언트가 동시에 시작하였다. 제안한 시스템은 평균 1%~ 4%사이의 대역폭을 점유하고 있으며 기존 시스템과 비교하여 일정하게 네트워크 대역폭을 이용하고 있음을 알 수 있다. 이는 스트리밍 서비스를 하기 위한 조건을 만족하며, 스트리밍의 큰 문제점의 끊김 현상이 거의 발생하지 않는다.

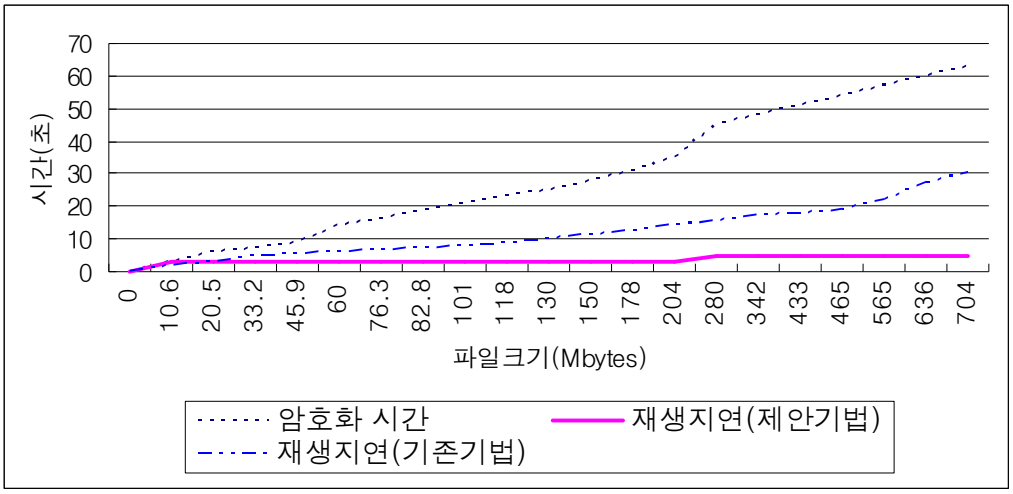


[그림 20] 평균 대역폭 점유율

[Fig. 20] The bandwidth possession ratio of even

본 논문에서 제안한 개선된 DRM 시스템에 대한 성능 평가를 위해 콘텐츠 데이터 중 가장 비중이 높은 동영상 데이터를 기준으로 실험하였다. 기존 시스템인 암호화되어 있는 비디오 데이터 파일의 복호화를 먼저 수행한 후 재생하는 방식과 제안한 시스템인 실시간으로 복호화를 하면서 재생하는 방식을 구현하여 수행시간을 측정하였다. 사용자가 동영상 파일을 실행할 경우 복호화와 실행이 동시에 수행되므로 재생지연시간에서 복호화 시간을 분리하여 정확하게 계산하기 어려우므로 암호화 시간을 측정하고 복호화와 실행시간을 동시에 측정하여 복호화 시간과 동영상 파일의 로딩시간이 포함된 지연시간을 계산하였다. 그러므로 재생 지연시간에서 암호화시간을 빼면 동영상 파일 재생을 위한 로딩 시간을 측정할 수 있다. [그림 21]은 동영상 파일을 이용하여 기존시스템과 제안하는 시스템의 암호화 시간과 재생지연시간을 측정한 후 그래프로 나타내었다.

기존 시스템은 동영상의 크기에 비례하여 암호화 시간이 늘어나며 지연시간 역시 비례하여 늘어나는 것을 알 수 있다. 그러므로 기존의 시스템을 대용량의 동영상데이터에 사용할 경우 많은 지연시간이 발생하여 사용자에 대한 서비스 지연 시간이 늘어나게 된다. 이에 반하여 제안하는 시스템은 동영상의 크기에 비례하여 암호화시간이 늘어나지만 복호화를 수행하면서 동시에 재생을 하므로 로딩 시간이 줄어들어 실제 지연시간은 줄어드는 것을 알 수 있다.



[그림 21] 암호화 시간과 재생 지연시간 비교

[Fig. 21] The comparison of Password anger hour and remaking delay hour

기존의 시스템과 제안된 시스템을 비교하여 보면 비디오 파일 전체를 복호화 한 후 재생하는 기존의 방식에서는 동영상 파일 크기가 커질수록 복호화와 파일의 재생을 위한 로딩 시간이 길어지며, 제안하는 방식에서는 복호화와 동시에 재생을 수행하므로 로딩 시간이 줄어들었다. 그러므로 제안된 시스템의 지연시간은 거의 안정한 시간대를 유지 하여 스트리밍의 끊김 현상이 발생 하지 않는다는 것을 증명 할 수 있다.

제 3 절 *DRM* 시스템 분석

P2P 네트워크 상에서 디지털 콘텐츠를 자유롭게 배포하는 장점을 살리고 문제점인 보호 기능을 제공할 수 있는 DRM 시스템을 설계하였다. 설계된 DRM 시스템은 기존의 DRM 시스템이 가지는 복제 소프트웨어, 화면 캡쳐 등을 이용한 불법 복제 방지의 어려움과 불법 유통 되는 콘텐츠에 대한 지속적인 저작권 관리 문제의 어려움을 해결 할 수 있다. 공개키 암호 알고리즘을 이용한 효율적인 키분배와 대칭키 암호 알고리즘을 이용한 콘텐츠의 기밀성 유지, 해쉬 함수를 이용한 무결성 등을 활용하여 인증 프로토콜 설계 하였다. [표 6]은 제안된P2P DRM 시스템과 기존 DRM 시스템을 비교 분석하였다.

본 논문에서 제안한 DRM 시스템은 라이선스 보안 측면에서 Microsoft DRM과 InterTrustDRM 시스템에 비하여 클라이언트 하드웨어 정보를 암호화함으로써 라이선스 자체의 변경 및 불법 사용자의 도용, 위조를 방지 할 수 있다. 또한 AES, SEED를 이용하여 암호화 시켰기 때문에 스트리밍 서비스 지원도 가능하나, Microsoft DRM 과 InterTrustDRM 시스템은 자사의 콘텐츠 타입을 지원하고자 설계 되어 있기 때문에 호환성에서 문제점이 있지만 제안된 DRM 시스템은 모든 콘텐츠를 패킹할 수 있도록 설계되었다. 기존 시스템과 제안하는 시스템의 차이는 기존의 시스템의 경우 대용량 동영상 파일인 경우에 자주 끊기는 현상으로 실시간으로 서비스를 제공하지 못 한다. 이를 P2P를 이용하여 개선함으로써 실시간적인 서비스를 가능하게 하였다. 사용자가 대용량의 동영상을 실행하면 사용자의 근접한 피어에게 접속하여 파일을 실행 할 수 있도록 하여 사용자에게 서비스를 제공한다. 그러므로 제안하는 DRM 시스템은 대용량의 파일의 스트리밍 지원이 가능하였다.

[표 6] 기존 DRM 시스템과 개선된 DRM 시스템

[Table 6] The Existing DRM system and The improved DRM system

	Microsoft DRM	InterTrust DRM	개선된 DRM
라이선스에대한 보안	다른 PC로 복제 방지 기간/횟수 조작방지 Tampering 방지	다른 PC로 복제 방지 기간/횟수 조작방지 Tampering 방지	다른 PC로 복제 방지 기간/횟수 조작방지 Tampering 방지 라이선스에 하드웨어 바인딩 기법 적용
불법콘텐츠 보안	미지원	미지원	지원
컨텐츠 분산	지원	미 지원	지원
암호화	서버 + 암호화된 콘텐츠에 파일내	서버	서버(콘텐츠파일암호) +클리이언트(하드웨어정보)
복호화	라이선스	라이선스	라이선스 클리이언트 정보
서비스	다운	다운	다운 + 스트리밍 지원
콘텐츠 지원	AUDIO, VIDEO	HTML, PDF, MS-office	모든 콘텐츠, 대용량콘텐츠일때 효율성극대화
재생소요시간	상대적으로 큼 (제안시스템과 비교)	상대적으로 큼 (제안시스템과 비교)	상대적으로 작음 (기존 시스템과 비교)
적용파일	소용량데이터파일	소용량데이터파일	소/대용량데이터파일

제 5 장 결 론

P2P 기술이 인터넷에 연결되어 있는 모든 컴퓨터를 네트워크화 시킴으로써 방대한 콘텐츠 저장 창고 역할을 하는 반면 그것은 역으로 그 만큼의 저작권 침해의 문제를 안고 있다. 이에 따라 많은 콘텐츠 저작권자들이 P2P 솔루션을 기반으로 하는 온라인 서비스업체를 상대로 저작권 소송을 벌이고 있다. PC 통신이나 HTTP를 기반으로 하는 서비스 제공자의 경우 중앙에서 서버를 관리하고 통제함으로써 저작권 위반 가능성이 있는 파일들에 대해 제재와 통제가 용이하지 않다. 또한 저작권문제 이전에 개개인의 컴퓨터를 통제하는 데서 오는 프라이버시 침해 문제 등의 유발 가능성이 있어 저작권 관리는 쉽지 않다. 그러나 P2P가 미래의 비즈니스 모델로서 기대됨에 따라 바이러스와 해킹 등의 보안 문제와 함께 저작권문제는 필연적으로 해결해야 할 사안이라고 할 수 있다.

본 논문에서 제안한 P2P 환경에서 개선된 콘텐츠 보안 시스템이 기존 시스템과의 호환성을 높이기 위하여 표준화되어 있고 범용으로 쓰이는 RSA, SEED, AES, SHA-1 등의 알고리즘을 이용하였다. 그리고 안전하고, 신뢰성 있는 디지털 콘텐츠 유통 및 인증을 위해 공개키 암호 알고리즘을 적용하여 비밀키의 분배와 서명에 사용하였다. 최소한의 정보만을 공개키 암호 알고리즘을 이용하여 암호화 하고, 콘텐츠 및 다른 정보들은 일반적으로 관용 암호 알고리즘을 사용하여 효율성을 고려하였다. 사용 되는 대칭키 암호 알고리즘은 AES와 SEED를 사용할 수 있게 설계하였고 공개키 암호 알고리즘은 기본적으로 RSA를 사용하도록 하였다. 해쉬 알고리즘으로는 SHA-1을 사용하였다. P2P 환경에서 DRM 서버를 구현함으로써 기존의 문제점인 콘텐츠 다운로드 속도를 자유롭게 다운로드 받을수 있게 개선하였다. 기존 시스템은 자사의 콘텐츠 타입을 지원하고자 설계 되어 있기 때문에 호환성에 문제점이 있지만 제안된 DRM 시스템은 모든 콘텐츠를 패킹할 수 있도록 설계되었다. 기존 시스템과 제안한 개선된 DRM 시스템의 차이는 기존의 시스템의 경우 대용량 동영상 파일인 경우에 자주 끊김 현상으로 실시간으로 인해 서비스를 제공하지 못

한다. 이를 P2P를 이용하여 서버에서 라이선스만을 다운로드받는 후처리 시스템을 사용하도록 개선함으로써 실시간적인 서비스를 가능하게 하였다. 사용자가 대용량의 동영상을 실행하면 사용자의 근접한 피어에게 접속하여 파일을 실행 할수 있도록 하여 사용자에게 서비스를 제공한다. 따라서 제안한 개선된 DRM 시스템은 대용량의 파일의 스트리밍 지원이 가능하다.

향후연구 과제는 사용자가 키를 노출 했을 경우를 대비하여 사용자의 하드웨어 정보를 인증하는 방식 시스템을 이용하였기 때문에 같은 사용자가 다른 컴퓨터에서 콘텐츠를 보고자 할 때의 처음부터 다시 인증해야 한다는 문제점을 어떻게 해결 하느냐 이다. 또 다른 문제점 중 하나는 키를 서버에 저장할 경우 키 관리의 문제를 단순하게 만들기는 하지만 사용자는 항상 서버와 온라인 상태를 유지해야 하는 단점이 있다. 사용자 컴퓨터에 두는 경우 키가 쉽게 복사되어도 안 되며 쉽게 훼손되어서도 안 된다. 이 문제를 해결하기 위해서는 플래쉬 메모리에 사용 권한을 저장하는 방법이 한 가지 해결책이나 복제가 가능하다는 문제점이 있다. 플래쉬 메모리에 사용 권한을 저장하고 보안 카드를 발급하는 방법과 핑거 프린팅 기술을 추가하여 핑거 프린팅 기반의 불법 콘텐츠 추적 시스템에 관한 계속적인 연구가 필요하다.

【참고문헌】

[1] Kan. G., "Gnutella", in Andy Oram(ed), Peer-to-Peer: Harnessing the Power of Disruptive Technologies, 2001.

[2] Ganchev, Philip, "Clusters-a Proposed Topology for a Peer-to-Peer Network". University of Pittsburgh, October 2000.

[3] Merrill Lynch, “The Decentralized Web-Peer to Peer Internetworking Joins Browser-Based Computing", June 2000.

[4] William Stalling, "Network and Internetwork Security", Prentice Hall, 1995.

[5] Doraswamy, N., and Harkins, D., IPSec, Upper Saddle River, NJ: Prentice Hall, 1999.

[6] Cheswick, W., and Bellovin, S., “Firewalls and Internet Security: Repelling the Wily Hacker”, Reading, MA: Addison-Wesley, 2000.

[7] Drew, G., Using SET for Secure Electronic Commerce, Upper Saddle River, NJ: Prentice Hall, 1999.

[8] Murhammer, M., et. al., TCP/IP: Tutorial and Technical Overview, Upper Saddle River: NJ: Prentice Hall, 1998.

[9] http://www.ibgen.co.kr/ebnewscontent.asp?board_idx=362&Gotopage=1

[10] http://web-biz.pe.kr/web/p2p3.html

[11] http://www.koscom.co.kr/koscom16.html

[12] http://wwwk.dongguk.ac.kr/~shkim2/chapter3/chapter3_7/tsld016.htm

[13] http://kmh.yeungnam-c.ac.kr/Hacking/1st/node199.html

[14] http://www.openp2p.com

[15] http://www.internet2.org

[16] http://www.openp2p.com/topics/p2p/security/

[17] http://sourceforge.net/

[18] William Stalling, SNMP, SNMPv2, SNMPv3, and RMON 1 and 2, Reading, MA: Addison-Wesley, 1999.

[19] William Stalling, Data and Computer Communications, 6th edition. Upper Saddle River, NJ: prentice Hall, 2000.

[20] CINCO, NetXRay users manual, CINCOLAN analysis products, 1999.

[21] 한국정보통신 기술협회, 128-bit 블록 암호화 알고리즘(SEED), 한국 정보통신 기술 협회, 1999.

[22] 광운대 IDEC 지역센터, “고속 PC 인터페이스 설계기술 및 CASE STUDY", 2004. 4.

[23] 삼성, “MPW 9차 Design Review Sheet", 2000. 3.

[24] 하재철, 문상재, “분할형 CSA를 이용한 Montgomery 곱셈기”, 한국정보보호학회 논문지, Vol.9, No.1, pp 77-84 2000. 9.

[25] 하재철, 문상재, “Montgomery 곱셈기를 이용한 효율적인 모듈라 멱승기 구현”. 한국정보보호학회 논문지, Vol. 11, No. 5, pp 63-74, 2001. 10.

[26] 아마추어를 위한 암호학과 대수곡선, 서광석, 양희선, 김대열, 권승탁, 박증수, (주)북스힐, 2000.

[27] eXtensible right markup Language (XrML) 2.0 Specification, http://www.xrml.org, 2001.

[28] Intertrust, intertrust.com/mail/overview/drm.html, 2001.

[29] James Daziel, "DOI in DRM environment", DOI EPICS Project White Paper, 2001.

[30] Joshua, D., Susan, K., "Understanding DRM System", IDC White Paper IDC, 2001.

[31] Microsoft, microsoft.com/windows/windowsmwdia/drm.asp, 2001.

[32] OMA, "Digital Rights Management Version 1.0" 05-September-2002.

[33] RSA Security, http://www.rsasecurity.com/rsalabs/node.asp?id=2176, 2000.

[34] The Language for Digital Rights, Michael Miron, Thomas DeMartini, Xin Wang, Brad Gandee, ContentGuard, Inc., December 2001.

[35] "Download Architecture Candidate Version 2.0", Open Mobile Alliance DRM W/G, 2004.

[36] "Digital Rights Management Candidate Version 2.0", Open Mobile Alliance DRM W/G, 2004.

[37] "Rights Expression Language Candidate Version 2.0", Open Mobile Alliance DRM W/G, 2004.

[38] "DRM Content Format Candidate Version 2.0", Open Mobile Alliance DRM W/G, 2004.

[39] "Generic Content Download Over The Air Candidate Version 2.0", Open Mobile Alliance DRM W/G, 2004.

[40] Joshua Duhl, Susan Devorkian, "Understanding DRM Systems", IDC White Paper, 2001.

[41] F.Hartung, F.Ramme, "Digital Rights Management and Watermarking of Multimedia Content for M-commerce Applications" IEEE Com. Magazine, Vol.38, pp.78-84, Nov. 2000.

[42] 윤영수, “Mobile DRM", 한국전자통신연구소 DRM 포럼, 2003.

[43] “MP3 DRM 규격”, KTF 플랫폼연구실, 2004년 4월.

[44] “통합 DRM 구축 전략 보고서”, KTF 신사업전략실, 2004년 6월.

[45] “무선 인터넷 플랫폼 기술”, KTF 기술 교육 자료, 2003년 5월.

[46] 한국정보보호학회, “현대 암호화 및 응용”, 한국정보보호진흥원, 2002년.

[47] 한국정보보호학회, “차세대 네트워크 보안기술”, 한국정보보호진흥원, 2002년.

[48] 강석준, 배태면, 노용만, 인소란, 유비쿼터스 환경에서 컨테츠 적응변환을 위한 워터마킹 시스템, 정보과학회 춘계학술대회, 2004.

[49] 삼성 SDS, "디지털 정보서비스 활성화를 위한 DOI의 특징 및 확산 전망“, 삼성 SDS IT Review, 2002.

[50] 전문석외, “정보이론 및 PKI pp158~197", 미래컴, 2003.

[51] Intel, "Advanced Digital Set Top Box Design-White Paper Revision 1.0", September 2003.

[52] 4C Entity, "Content Protection for Prerecorded Media Specification Revision 1.0", January 2003.

[53] ISMA, "ISMACryp Draft Specification, Version 7.0", ISMA, April 2003.

[54] IBM Research Division Almaden Research Center, "xCP: eXtensible Content Protection", 2003.

[55] IBM, "IBM Response to DVB-CPT Call for Proposals for Content Protection & Copy Management: xCP Cluster Protocol", October 2001.

[56] ETSI, "Digital Video Broadcasting (DVB): Support for use of scrambling and Conditional Access (CA) within digital broadcasting systems", ETR 289, ESTI, October 1996.