공개키 기반 구조를 이용한 소프트웨어의 저작권 보호

이병천*, 임신영**, 김광조*

* 한국정보통신대학원대학교
** 한국전자통신연구원

Copyright Protection of Software using Public Key Infrastructure

Byoungcheon Lee*, Shinyoung Lim**, Kwangjo Kim*

* School of Information and Computer Engineering,
Information and Communications University
** Electronics and Telecommunications Research Institute

요 약 문

이 논문에서는 소프트웨어의 저작권을 보호하기 위한 방법으로서 공개키 기반 구조(Public Key Infrastructure, PKI)와 사용자 의존적인 소프트웨어(User Dependent Software, UDS)의 개념을 이용하는 새로운 방법을 제안한다. 공개키 인증서(Certificate)는 개인의 ID 와 개인이 사용할 공개키를 묶어서 인증기관이 서명한 문서로서 개인의 사회적 신분을 나타내는 문서라고 볼 수 있다. 사용자 의존적인 소프트웨어란 사용자의 공개키로 암호화한 정보를 포함하여 컴파일되는 사용자 고유의 소프트웨어 컴포넌트이다. 사용자가 소프트웨어를 이용하기 위해서는 개인키를 제시하여 암호화된 정보를 복호화할 수 있음을 증명해야 한다. 공개키 기반 구조를 이용하는 이러한 구조의 소프트웨어를 이용하면 사용자는 자신의 공개키 인증서와 개인키에 대한 사회적 책임하에 소프트웨어를 정당하게 사용하게 됨으로써 불법 복제가 어렵게 되고 소프트웨어의 저작권을 근본적으로 보호할 수 있게 된다. 이러한 조건에서라면 소프트웨어의 온라인 판매에 이은 온라인 배달도 가능해져서 사회적으로 많은 물류 비용을 절감할 수 있을 것으로 기대된다.

1. 서 론

현대의 정보화 사회는 많은 소프트웨어들을 사용하고 있으며 정보화 사회의 발전은 일정 부분 이들 소프트웨어 개발자들이 이끌어 왔다고 볼 수 있다. 소프트웨어의 저작권이 적절히 보호된다면 이들 개발자들은 더 큰 추진력을 가지고 새로운 제품들을 계속 개발해 나갈 수 있겠지만 불법 복제 등으로 인해 보호되지 못한다면 원활한 제품 개발이 어렵고 더 이상의 발전을 기대하기 어렵게 될 것이다. 소프트웨어의 저작권 보호를 위해서 여러 가지 법적 장치들이 마련되고 있지만 기술적인 측면에서는 충분한 기술적 바탕을 제공해 오지 못했던 것이 사실이다.

소프트웨어의 저작권 보호 문제는 크게 두 가지 측면으로 나누어 생각해 볼 수 있는데 하나는 소프트웨어의 법적인 권리를 보호해주는 저작권 보호(copyright protection) 측면이고 또 하나는 불법복제 방지(copy protection) 측면이 있다. 법적인 저작권 보호를 위해서는 개발자에게 소프트웨어에 대한 권리를 법적으로 보호해주고 개발된 소프트웨어를 공공 등록기관에 등록하도록 하는 것이다. 기술적인 측면에서는 데이터, 멀티미디어 저작물 등에 워터마크(watermark)를 추가하는 기술이 많이 연구되고 있다. 즉 워터마크를 추가한 자료를 타인이 불법적으로 복사하여 사용한다면 워터마크를 분석하여 자신의 저작권을 주장할 수 있다. 이를 위해서는 워터마크와 자신의 ID 를 연결할 수 있도록 하는 신뢰 기관에의 등록과정이 필요하다. 최근에는 dummy 변수를 이용하는 방법, 두 가지 동등한 operation 을 선택적으로 이용하는 방법 등 다양한 방법으로 개발자의 ID 정보를 소프트웨어에 넣는 방법도 제안되었다[1,2,3].

한편 소프트웨어의 불법복제를 방지하기 위한 방법으로도 여러 가지 암호학적 연구들이 수행되어져 왔다. J. Gosler 는 소프트웨어를 보호하기 위한 방법론으로서 플로피 디스크에 대한 물리적, 자기적인 서명, 하드웨어 장치를 이용하는 방법, 공격자가 소프트웨어를 분석하고 변조하지 못하도록 하는 Software Analysis Denial(SAD), Technology Denial Concepts(TDC) 등에 대해 설명하였다[4]. A. Herzberg 는 CPU 상에 구현되는 소프트웨어 보호 메커니즘에 대해 제안하였다[5]. R. Mori 등은 Superdistribution[6] 이라는 새로운 개념을 도입하고 이를 이용한 소프트웨어 보호 방식에 대해 연구하였는데 이 또한 하드웨어에 의존하는 메커니즘이다.

실제 소프트웨어 업계에서는 다양한 방법들을 사용하고 있는데 이들은 불법복제를 막기

에는 기술적인 어려움을 가지고 있다고 보여진다. 예를 들면 가장 흔히 사용되는 방법으로 소프트웨어 설치 시 접근 암호를 이용하는 방법이 있을 수 있는데 사용자들은 하나의 라이센스를 가지고 여러 대의 컴퓨터에 무제한으로 설치하여 사용하거나 접근 암호를 타인과 공유함으로써 불법 복제가 가능하였다. 이런 사용자들에게는 사후에 법적인 책임을 물을 수도 있겠지만 기술적인 측면에서 이러한 종류의 불법 사용을 근본적으로 막을 수 있는 방법이 필요한 것이다.

본 연구에서는 공개키 기반 구조(Public Key Infrastructure, PKI)를 이용하는 새로운 소프트웨어 저작권 보호 방법에 대해 제안하고자 한다. 공개키 암호의 사용은 이미 보편화되고 있으며 공개키의 인증을 위한 공개키 기반 구조에 대해서도 많은 연구 및 개발이 이루어지고 있어서 조만간 이의 적용이 급격히 확대될 것으로 전망된다. 공개키 인증서는 신뢰받는 인증기관이 개인의 ID 와 개인이 사용할 공개키를 결합하여 서명해준 문서로서 개인의 사회적인 신분을 나타내는 문서라고 볼 수 있다. 더구나 공개키 인증서에 포함된 공개키와 쌍을 이루는 개인키는 개인이 비밀히 보관하고 있다는 것을 전제로 하고 있기 때문에 개인키를 이용하는 암호화 동작은 부인이 불가능하고 개인이 책임져야 하는 특징이 있다. 이러한 공개키 기반 구조는 현재 널리 연구 개발되고 있는 전자상거래를 비롯하여 각종 정보보호 시스템 구현에 있어서 필수 불가결한 사회적 기반 구조이다.

본 연구에서 제안하는 소프트웨어 저작권 보호 시스템은, 공급자가 사용자의 공개키로 암호화된 사용자 의존적인 정보를 포함하여 컴파일되는 사용자 의존적인 소프트웨어(User Dependent Software, UDS)를 생성하여 제공하도록 하며, 사용자는 소프트웨어 사용시 사용자의 개인키를 제시하도록 하는 특징을 가지고 있다. 이러한 구조의 소프트웨어를 이용하면 사용자는 자신의 공개키 인증서와 개인키에 대한 사회적 책임하에 소프트웨어를 정당하게 사용하게 됨으로써 불법 복제가 어렵게 되고 소프트웨어의 저작권을 근본적으로 보호할 수 있게 된다. 또한 공급자는 소프트웨어의 온라인 판매에 이은 온라인 배달도 가능해져서 물리적인 패키지를 만들 필요가 없고 판매망을 유지할 필요가 없으며 제품 배달 비용도 필요없게 되는 등 사회적으로 많은 비용을 절감할 수 있으며 결과적으로 제품 가격을 크게 낮출 수 있을 것으로 기대된다. 사용자는 소프트웨어 사용시마다 개인키를 제시해야 하는 불편함이 있겠지만 향후 PKI 가 사회적으로 확산되고 개인마다 개인키가 저장된 IC 카드 형태의 개인 신분증이 발급된다면 큰 불편 없이 정당하게 소프트웨어를 사용할 수 있을 것이다.

2 장에서는 기존의 사용되고 있는 소프트웨어 불법복제 방지를 위한 방법들과 그들의 문제점에 대해 간단히 살펴본다. 3 장에서는 공개키 기반 구조의 개요를 간단히 소개한다. 4 장에서는 본 연구에서 제안하는 시스템과 그 동작 과정, 적용 효과, 실제 소프트웨어의 판매 및 배달 과정에 대해 설명한다. 5 장에서는 결론과 향후 전망에 대해 살펴본다.

2. 기존의 소프트웨어 불법복제 방지 방법

본 장에서는 현재 소프트웨어 업계에서 사용되고 있는 불법복제 방지 기술들에 대하여 조사한 내용을 개략적으로 기술하고자 하는데 실제로는 이보다 훨씬 다양한 방법들이 사용되고 있는 것으로 파악된다.

. 설치 시 접근 암호를 이용하는 방법

소프트웨어의 설치 시 접근 암호 또는 제품 번호를 입력하도록 하는 것으로 현재 판매되고 있는 소프트웨어 패키지들의 많은 수가 이러한 방법을 사용하고 있다. 이것은 접근 암호를 타인과 공유하거나 복수의 컴퓨터에 설치하여 사용함으로써 불법 복제가 가능하다. 디스켓으로 제공되는 경우에는 설치 횟수를 제한하는 방법을 사용하기도 하였는데 CD 로 제공되는 현재의 환경에서는 이것의 적용도 어렵다.

. 하드웨어적인 Key Lock 을 사용하는 방법

컴퓨터 시스템의 포트에 하드웨어적인 Key Lock 을 설치해야만 특정 소프트웨어를 사용할 수 있도록 하는 방법이다. 이런 방법은 하드웨어를 필요로 하므로 추가적인 비용이 필요하며 시스템 포트가 제한되어 있으므로 여러 가지 소프트웨어들이 모두 하드웨어 방식을 이용하기는 어렵다는 단점이 있다.

. 특정 플로피 디스켓을 제시해야 하는 방법

공급자가 제공하는 특정 플로피 디스켓을 제시해야만 소프트웨어를 사용할 수 있도록 하는 방법인데 플로피 디스켓은 휴대가 불편하고 손상되기 쉬운 단점이 있다. 또한 여러 가지 소프트웨어가 이러한 방법으로 보호되어야 한다면 동시 사용이 어렵고 매우 불편한 방법이 될 것이다.

. 비밀키가 저장된 IC 카드를 제시해야 하는 방법

특정 소프트웨어를 사용하기 위한 비밀키를 IC 카드에 저장하여 사용하는 방법은 플로

피 디스켓을 사용하는 방법보다 진보된 방법으로서 하드웨어적으로 높은 수준의 불법 복제 방지가 가능하겠지만 IC 카드 리더를 장착하고 IC 카드를 사용해야 하므로 많은 비용이 소요된다는 단점이 있다. 또한 여러 개의 소프트웨어가 각기 다른 IC 카드를 이용한다면 동시 사용이 어렵고 매우 불편한 방법이 될 것이다.

시스템 ID 정보를 이용하여 특정 시스템용 제품을 제작하여 공급하는 방법
이 방법은 소프트웨어 구매시 사용할 시스템의 ID 정보를 제공하고 공급자가 이를 이용하여 특정 시스템에서만 사용할 수 있는 제품을 만들어 제공하는 방법이다. 이 방법은 다른 시스템에 설치해야 하거나 시스템 정보가 바뀌는 경우에는 공급자에게 조치를 요구해야 하는 번거로움이 있다.

설치 시 네트웍으로 온라인 등록하는 방법
네트웍 어플리케이션의 경우 시스템에 설치 시 공급사의 서버에 자동으로 접속하여 등록하도록 하는 방법이다. 매우 우수한 방법이지만 네트웍 어플리케이션이 아닌 경우에는 적용하기 어렵다.

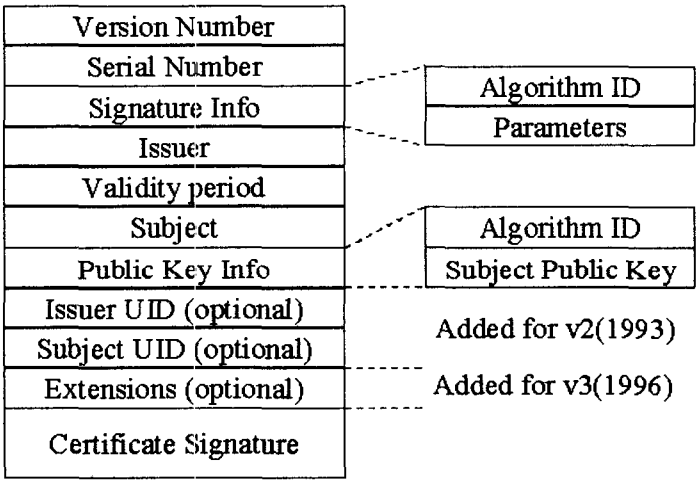
지금까지 다양한 방법들이 연구되고 적용되어 왔지만 소프트웨어의 저작권을 근본적으로 보호해 주기에는 기술적으로 많은 문제가 있었다. 이러한 방법들이 저작권 보호에 미흡했던 이유중의 하나는 사용자의 신분을 확인해 줄 수 있는 방법이 마땅치 않았기 때문이다. 이러한 면에서 공개키 인증서 및 공개키 기반 구조는 개인의 신분을 사회적으로 증명해 줄 수 있는 암호학적 방법으로서 소프트웨어 저작권 보호의 목적으로도 유용하게 사용될 수 있을 것이다. 본 연구에서는 공개키 암호와 공개키 기반 구조를 이용함으로써 사용자의 신분을 명확히 확인할 수 있고 소프트웨어의 저작권을 근본적으로 보장할 수 있는 새로운 방법을 제안하고자 한다.

3. 공개키 기반 구조의 개요

공개키 암호 기술을 적용함에 있어서 가장 큰 이슈는 공개키에 대한 인증 문제이다. 즉 어떤 개인이 사용하고자 하는 공개키가 실제로 그 개인의 소유인지 확인해 줄 수 있는 사회적인 신뢰구조가 형성되어야만 공개키 암호 기술이 본격적으로 적용될 수 있다는 것이다. 개인 수준의 전자우편 보안 서비스인 PGP(Pretty Good Privacy)[7]에서는 "Web of Trust" 라는 개념도 사용하고 있지만 제 3 의 신뢰하는 인증기관이 서명하여 발급해주는

인증서(Certificate)를 이용하는 계층적 인증구조에 대하여 많은 연구가 이루어지고 있다. 현재 널리 사용되고 있는 X.509[8] 인증서는 <그림 1>에 보인 바와 같이 사용자 ID, 사용기간, 기타 옵션 등의 개인 정보와 개인이 사용할 공개키를 묶어서 인증기관이 서명한 문서이다. 이러한 X.509 인증 구조에 근거하여 S/MIME 전자우편 보안기술, SET(Secure Electronic Transaction)[9] 전자상거래 기술 등이 개발되고 있으며 PKIX(Public Key Infrastructure based on X.509)[10]에서는 공개키 기반 구조 기술에 대한 종합적인 표준화를 지향하고 있다. 이와 같은 기술 발전 추세를 살펴볼 때 향후 공개키 기반 구조(PKI)가 개인을 인증해 주는 중요한 사회적 기간 구조의 역할을 할 것이며 개인들은 자신만의 공개키와 이에 해당하는 개인키를 가지고 전자상거래, 전자계약, 이동통신의 이용, 의료진료 등 다방면의 사회 생활을 영위하게 될 것으로 전망된다. 개인의 공개키는 디렉토리서비스 등의 수단으로 상호 공유하게 될 것이며 개인키는 IC 카드 등의 하드웨어 형태로 발급되어 편리하게 사용될 것이다.

이러한 공개키 기반 구조가 널리 보급되어 있는 환경에서라면 소프트웨어의 생산, 판매, 사용에서도 이러한 기반 구조를 이용할 수 있을 것이며 본 연구에서는 공개키 기반 구조를 소프트웨어 저작권 보호 및 불법복제 방지의 용도에 사용할 수 있는 새로운 방법을 제안하고자 한다.



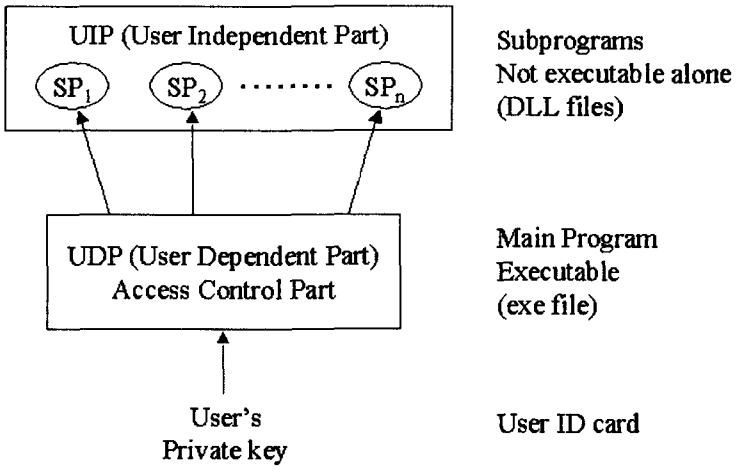
<그림 1> X.509 v3 의 인증서 양식

4. 공개키 기반 구조를 이용하는 불법복제 방지 시스템

4.1 본 연구에서의 접근 방법

본 논문에서 제안하는 불법복제 방지 시스템의 가장 기본적인 특징은 첫째, 소프트웨어 공급자가 사용자의 공개키로 암호화된 정보를 포함하는 사용자 의존적인 소프트웨어(User Dependent Software, UDS)를 제작하여 제공한다는 것이며, 둘째, 사용자가 소프트웨어 사용 시 자신의 개인키를 제시하여 자신의 공개키로 암호화된 정보를 복호화할 수 있다는 것을 증명해야만 소프트웨어에의 접근 권한을 제공한다는 것이다.

이러한 기능을 제공하기 위하여 이 논문에서는 <그림 2>에서와 같이 소프트웨어를 크게 UIP(User Independent Part)와 UDP(User Dependent Part)의 두 부분으로 나누었다. UIP 는 소프트웨어의 대부분의 기능들이 구현되어 있는 부 프로그램(subprogram, 이하 SP_i로 표기)들의 집합이며 독립적으로 실행될 수 없고 외부에서 호출되어 사용된다. 예를 들면 Windows 의 DLL(Dynamic Link Library)이 대표적인 예이다. UDP 는 부 프로그램에의 접근 제어 기능을 가지며 사용자의 공개키로 암호화된 정보를 포함한 사용자 의존적인 정보를 포함하여 컴파일된 실행 프로그램이다. 사용자가 소프트웨어를 사용하기 위해서는 개인키를 제시해야 하며 UDP 는 제시된 개인키가 공개키로 암호화된 정보를 복구할 수 있는지 확인 후 UIP 에의 접근을 허용한다. 사용자의 공개키 인증서에 해당하는 개인키를 소유한 사람은 사용자 자신뿐이며 소프트웨어는 개인키를 제시할 수 있는 단 한명의 사용자만이 사용할 수 있는 것이다.



<그림 2> 저작권 보호를 위한 소프트웨어의 개략적인 구조

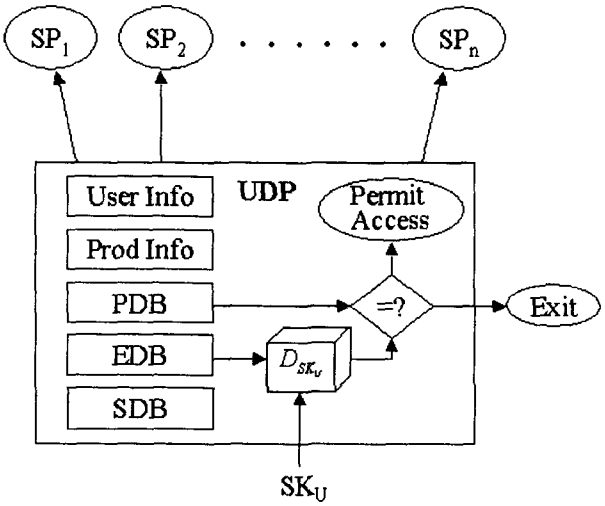
소프트웨어를 UIP와 UDP의 두 부분으로 나누는 것은 사용자 의존적인 정보를 포함하여 사용자별로 생성해야 하는 UDP의 컴파일 시간을 줄이고 UIP는 미리 배포될 수 있도록 하기 위한 것이다. 이 논문에서는 UDP의 구조 및 동작방법에 대하여 접근제어 방식과 실행화일 암호화 방식의 두 가지 방법을 제안한다.

4.2 접근제어 방식의 UDP

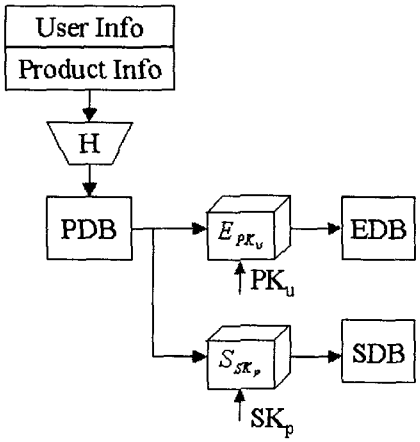
접근제어 방식의 UDP는 <그림 3>에 보인 것과 같이 5개의 사용자 의존적인 정보와 이 정보를 이용하는 접근제어 루틴이 함께 컴파일된 실행화일이다. 여기에서 사용되는 5개의 사용자 의존적인 정보는 사용자의 공개키 인증서를 포함하는 사용자 정보를 이용하여 <그림 4> 및 <표 1>에 나타낸 방법으로 생성된다. User Info는 사용자 이름, 사용자 주소, 사용자의 공개키 인증서, 구매일자 등 사용자 정보를 나타내며 Product Info는 공급자 이름, 공급자 주소, 공급자의 공개키 인증서, 소프트웨어의 사용기간 등의 제품 정보를 나타낸다. PDB(Plain Data Block)는 User Info와 Product Info를 해쉬한 값으로서 공개키 암호화를 위한 평문으로 사용된다. EDB(Encrypted Data Block)는 PDB를 사용자의 공개키 PK_U로 암호화(E_{PK_U})한 암호문이다. SDB(Signed Data Block)는 PDB를 공급자의 개인키 SK_P로 서명(S_{SK_P})한 서명문이다. UDP는 사용자의 개인키를 입력받는 루틴, 암호화 작업을 수행하는 접근제어 루틴, UIP의 부 프로그램들을 호출할 수 있는 루틴을 가진다.

구매자가 소프트웨어 공급자에게 온라인 접속하여 구매의사를 표시하면 공급자는 구매자의 공개키 인증서를 포함하는 구매자 정보를 요구한다. 공급자는 제시된 구매자의 인증서에 서명된 인증기관의 서명을 확인하여 구매자의 신분을 분명하게 확인할 수 있다. 공급자는 구매자 정보를 이용하여 <표 1>의 다섯가지 정보를 생성하고 이들을 접근제어 루틴과 함께 컴파일하여 UDP 부분을 생성한다. 공급자는 새롭게 생성한 UDP와 이미 만들어 놓은 UIP를 함께 구매자에게 송신한다.

이렇게 생성된 소프트웨어를 사용하기 위해서는 사용자의 개인키 SK_U를 제시하여야 한다. 소프트웨어의 기동시 UDP의 접근제어부는 제시된 사용자의 개인키를 이용하여 EDB를 복호화(D_{SK_U})하고 이것이 PDB와 일치하는지 확인하여 맞는 경우에만 UIP에 있는 부 프로그램들에의 접근을 허용한다. 소프트웨어에 대한 사용자의 사용권한을 확인하기 위해서는 SDB를 공급자의 공개키 PK_P로 검증(V_{PK_P})하여 PDB와 일치하는지 확인한다.



<그림 3> 접근제어 방식의 UDP 구조 및 동작



<그림 4> UDP 에 사용되는 사용자 의존적인 정보의 생성

<표 1> UDP에 포함되는 사용자 의존적인 정보

정 보 명	내 용
User Info	사용자 이름, 주소, 사용자의 인증서, 구매일자 등
Product Info	공급자 이름, 주소, 공급자의 인증서, SW 사용기간, 제품 번호 등
PDB(Plain Data Block)	User Info 와 Product Info를 해쉬한 값 $PDB = H(\text{UserInfo} \| \text{ProductInfo})$
EDB(Encrypted Data Block)	PDB를 사용자의 공개키로 암호화한 데이터 블록 $EDB = E_{PK_U}(PDB)$
SDB(Signed Data Block)	PDB를 공급자의 개인키로 서명한 데이터 블록 $SDB = S_{SK_P}(PDB)$

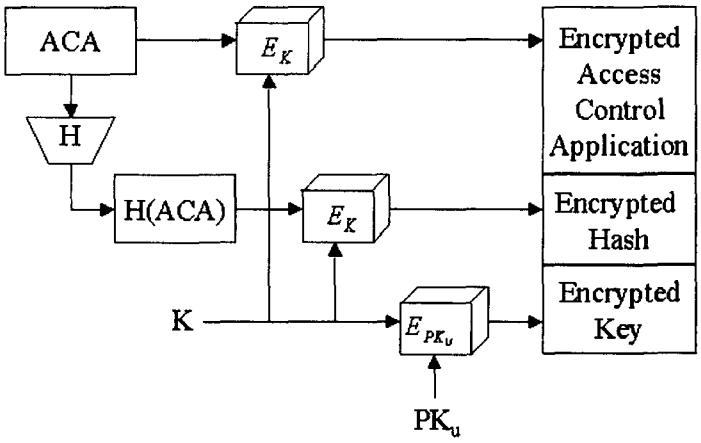
4.3 실행화일 암호화 방식의 UDP

앞에서 언급한 접근제어 방식의 UDP는 UDP 내부에 접근제어 루틴이 가용한 형태로 존재하고 PDB, EDB, SDB 등의 사용자 의존적인 정보가 가용한 형태로 포함되어 있다. 이러한 소프트웨어는 강력한 공격자가 소프트웨어 디버거 등의 수단을 이용하여 UDP를 상세히 분석하고 변조, 바이패스 할 수 있을지도 모른다. 이러한 잠재적인 위험성을 방지하고자 접근제어 실행화일 자체를 암호화하는 방식을 제안한다.

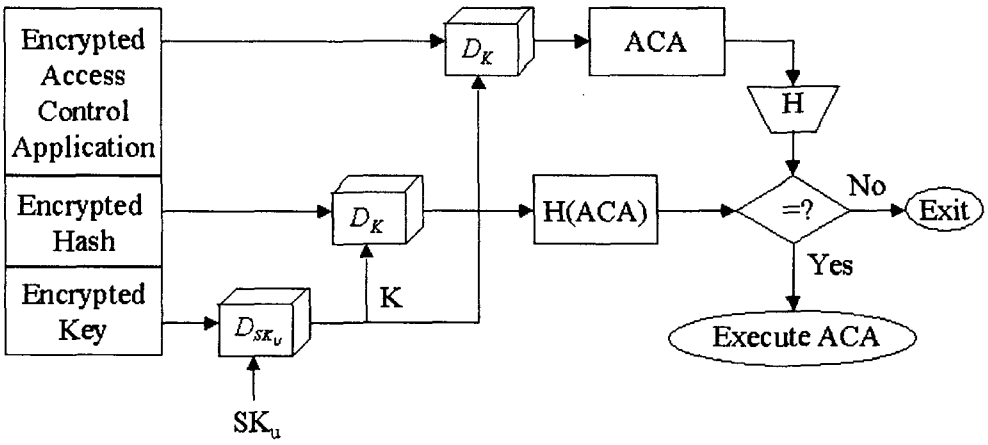
<그림 5>에서 (a)는 실행화일 암호화 방식에서 UDP를 생성하는 과정을 나타낸다. UIP를 접근할 수 있는 접근제어 실행화일(Access Control Application, ACA)을 생성한 후 이것의 해쉬값을 계산한다. 난수적인 대칭형 세션키 K를 생성한 후 이것으로 접근제어 실행화일과 해쉬값을 대칭키 방식으로 암호화(E_K)하여 저장하고 세션키는 사용자의 공개키(PK_U)로 암호화(E_{PK_U})하여 저장한다.

<그림 5>의 (b)는 실행화일 암호화 방식에서 UDP로부터 접근제어 실행화일(ACA)을 복호화하고 이를 실행하는 과정을 나타낸다. 사용자가 개인키(SK_U)를 제시하면 먼저 암호화된 세션키를 복호화(D_{SK_U})하여 세션키 K를 얻고 이것을 이용하여 접근제어 실행화일과 해쉬값을 복호화(D_K) 한다. 복호화된 접근제어 실행화일(ACA)로부터 해쉬값을 계산하여 이것이 복호화된 해쉬값과 일치하는 경우에만 접근제어 실행화일을 실행시키고 일치하지 않으면 프로그램을 종료한다. 이 모델에서는 UIP에 접근 가능한 실행화일이 암호화되어 있어서 먼저 이를 성공적으로 복호화하지 않으면 소프트웨어를 사용할 수 없다. 소프트웨어

에 대한 공격자도 접근제어 실행화일을 복구하기 전에는 분석을 시도하지 못한다.



(a) UDP 의 생성 과정



(b) UDP 로부터 접근제어 실행화일의 복호화 및 실행

<그림 5> 실행화일 암호화 방식의 UDP 를 이용한 UIP 의 접근 제어

ACA : Access Control Application, K : Random Session Key

PK_U : User's Public Key, SK_U : User's Private Key

4.4 제안 시스템의 특징 및 효과 분석

본 논문에서 제안된 공개키 기반 구조를 이용하는 소프트웨어 저작권 보호 방법은 제작자의 저작권을 보호하고 사용자의 불법 복제를 방지할 수 있는 근본적인 방법이라고 생각된다. 공개키 인증서는 인증기관이 사용자의 공개키와 사용자의 ID를 묶어서 서명한 문서로서 개인의 사회적 신분을 나타낸다. 이런 환경에서 공개키 인증서에 해당하는 개인키는 사용자가 비밀리에 보관하고 있다고 가정되며 개인키를 이용한 암호화 동작에 대해서는 부인할 수 없고 개인이 책임을 져야 한다. 만일 같은 공개키 인증서가 전자상거래, 전자문서교환, 이동 통신 등 다른 용도에도 사용되고 있다면 이를 복제하여 남들과 공유하기가 어려울 것이다. 만일 개인키가 IC 카드 등 하드웨어 시스템으로 발급되었다면 복제가 어려워서 더 높은 보안성을 제공할 수 있다.

이러한 저작권 보호 시스템을 이용함에 있어서 추가적인 비용은 전혀 필요치 않다. 공개키 기반 구조가 전자상거래 등의 응용 분야에 이미 널리 사용되고 있으며 개인들은 모두 공개키 인증서를 발급받고 IC 카드에 개인키를 저장하고 있다면 즉시 적용 가능한 것이다.

이러한 소프트웨어 시스템은 저작권 보호 뿐만 아니라 기존의 소프트웨어가 제공하기 어려운 다음과 같은 특징적인 기능들을 제공할 수 있다.

- 소프트웨어를 원하는 만큼 여러대의 컴퓨터에 설치하여 사용할 수 있다. 기존의 방법을 사용한다면 같은 소프트웨어를 여러대의 컴퓨터에 설치하는 것이 불법적인 행동이 되겠지만 이러한 소프트웨어는 개인키를 가진 적법한 사용자만이 사용할 수 있으므로 저작권을 침해하는 것이라고 생각되지 않는다.
- 같은 개인키를 여러가지 소프트웨어에 사용할 수 있다. 다수의 소프트웨어를 구입함에 있어서 같은 공개키 인증서를 제시할 수 있으며 사용시에는 같은 개인키를 이용할 수 있으므로 매우 편리하다.
- 여러 사용자를 위한 소프트웨어를 제공할 수도 있다. 소프트웨어 판매시 다수 사용자의 공개키 인증서를 제시받고 이들의 사용자 정보를 모두 포함하는 UDP를 생성하여 제공할 수 있다. 또는 각 개인별로 다른 UDP를 제공하고 UIP는 다수 사용자가 공유할 수도 있다.
- 소프트웨어의 설치가 매우 간단하다. 기존의 소프트웨어들은 패키지가 매우 크고 제품 번호를 확인하는 등 설치과정이 복잡하지만 이러한 소프트웨어에서는 UIP를 인

터넷등을 통하여 무료로 사전 배포하거나 컴퓨터 시스템 구입시 무료로 제공할 수 있다. 사용자가 소프트웨어를 구입하면 사용자 의존적인 UDP 만을 제공받고 설치하면 된다.

- 소프트웨어를 이용하여 생성되는 데이터에 사용자의 서명을 추가함으로써 데이터에 대한 저작권을 보호할 수도 있다.
- 회사와 같은 단체를 위해서는 그룹키의 개념을 이용하여 특정 그룹 내에서 사용할 수 있는 소프트웨어를 생성하여 제공할 수도 있다.

4.5 온라인상의 소프트웨어 구매, 배달 과정

현재의 전자상거래 연구는 온라인 지불에 초점을 맞추고 있으며 온라인 지불이 이루어지더라도 디지털 상품의 배달은 오프라인 배달을 이용하고 있는데 그 이유는 디지털 상품의 저작권 보호 문제 때문이라고 생각된다. 소프트웨어 저작권의 문제가 이와 같이 분명하게 해결된다면 소프트웨어의 제작, 판매, 배달 등의 모든 과정이 온라인 상에서 이루어질 수 있을 것이다. 이러한 온라인 배달 과정은 사용자와 공급자 모두에게 편리할 뿐만 아니라 많은 사회적 물류 비용을 절감할 수 있다. 온라인 상에서 이루어지는 소프트웨어의 구매, 배달 과정의 일례를 <그림 6>에 나타내었으며 다음과 같이 정리할 수 있다.

* 사전 준비

(1) 소프트웨어 개발

소프트웨어 개발자는 소프트웨어를 UIP 와 UDP 의 두 부분으로 나누어 개발한다.

(2) UIP 의 사전 배포

UIP 는 인터넷 매체를 통해 사전 배포하거나 하드웨어 생산자들에게 무료 배포한다.

(3) UDP 생성 준비

소프트웨어 공급자는 서버 시스템을 갖추고 UDP 를 생성할 준비를 갖춘다.

(4) UIP 의 사전 획득

구매자는 UIP 를 네트웍을 통해 다운받거나 하드웨어 생산자들로부터 무료로 획득한다.

(5) 공개키 인증서 획득

공급자와 구매자는 인증기관으로부터 공개키 인증서를 발급받는다.

* 온라인 구매 및 배달

(6) 구매 요청

구매자는 공급자의 서버에 접속하여 구매 의사를 표시하고 공개키 인증서를 포함한 사용자 정보를 제공한다. 이때 구매자와 공급자 사이에는 상호 인증이 이루어진다.

(7) 온라인 지불

공급자와 구매자 사이에 소프트웨어 구매에 대한 온라인 지불이 이루어진다.

(8) UDP의 생성

공급자는 제공된 사용자 정보를 이용하여 UDP를 생성한다.

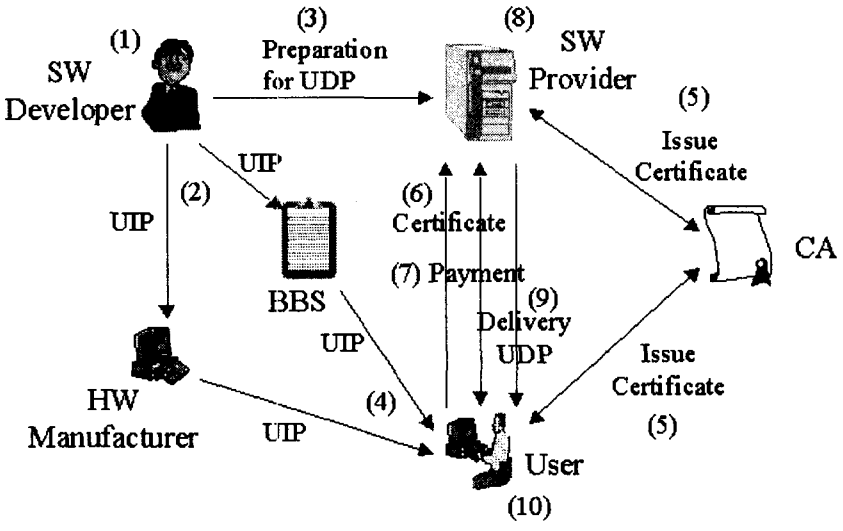
(9) UDP의 배달

공급자는 구매자용 UDP를 온라인 배달한다.

* 소프트웨어의 사용

(10) 소프트웨어를 설치 및 사용

소프트웨어 설치는 UIP와 UDP를 컴퓨터 시스템에 등록함으로써 간단히 이루어진다. 소프트웨어 사용시 사용자는 UDP의 접근제어를 받아 개인키를 제시한 후 사용한다. 사용자가 소프트웨어의 사용권한을 확인하고자 할 때는 UDP 내의 공급자의 서명을 확인한다.



<그림 6> 온라인상의 소프트웨어 구매, 배달 과정의 일례

위에서 설명한 일련의 과정은 공개키 기반 구조를 이용한 소프트웨어의 구매 및 배달 과

정의 일례이며 제공되는 환경에 따라 다른 세부 절차도 가능할 것이다. 이와 같이 소프트웨어의 판매 및 배달 과정이 온라인으로 처리될 수 있다면 공급자는 물리적인 형태로 제품을 만들 필요가 없고, 판매망을 유지할 필요가 없으며, 상품 배달 비용이 들지 않아서 매우 큰 비용 절감 효과를 얻을 수 있으며 결과적으로 제품 가격을 크게 낮출 수 있을 것이다.

5. 결 론

본 연구에서는 공개키 기반 구조(PKI)를 이용하는 새로운 소프트웨어 저작권 보호 시스템을 제안하였다. 가장 큰 특징으로는 공개키 인증서를 이용하여 사용자 의존적인 정보를 포함하는 소프트웨어를 생성하여 제공하고 소프트웨어의 사용시에는 개인키를 제시하여야 한다는 것이다. 이러한 메커니즘하에서는 개인의 신분이 명확해지고 소프트웨어를 정당하게 사용하게 되며 불법 복제가 근본적으로 어려워지는 장점이 있다. 이러한 시스템을 사용하기 위해서는 공개키 기반 구조하에서 추가적인 비용이 전혀 필요 없이 즉시 이용할 수 있으며 기존의 소프트웨어들이 제공하지 못했던 여러 가지 새로운 기능들을 제공할 수 있다는 장점이 있다. 향후 공개키 기반 구조의 확산과 함께 소프트웨어의 생산, 판매, 배달 등에서 전반적인 구조 변화가 예상되며 온라인 지불과 온라인 배달이 함께 이루어지는 실질적인 전자상거래가 확립될 수 있을 것이다.

이 논문에서는 공개키 기반 구조가 확립된 환경을 전제로 이를 이용한 소프트웨어 저작권 보호에 대해 언급하였는데 이것이 확립되기 전에는 소프트웨어적인 화일 형태의 개인키 관리를 고려해 볼 수 있다. 또는 소프트웨어 공급자가 적법한 구매자에게 공개키 인증서를 발급하고 이를 이용하여 제품을 판매하는 경우를 생각해 볼 수도 있다. 이 논문에서는 제안된 프로토콜들에 대한 안정성 분석은 향후 과제이며 암호 분석자(Cryptanalyst)들의 입장에서는 특히 이 논문에서 제시된 UDP 의 해독에 대해서 많은 관심을 가질 수 있을 것이다.

참고 문헌

[1] N. Hirose, et al., "A proposal for software protection", SCIS'98-9.2.C, The proceedings of the 1998 Symposium on Cryptography and Information Security, Shizuoka, Japan, Jan. 28-31, 1998

[2] A. Monden, et al., "A watermarking method for computer programs", SCIS'98-9.2.A, The proceedings of the 1998 Symposium on Cryptography and Information Security, Shizuoka, Japan, Jan. 28-31, 1998

[3] T. Kitagawa, et al., "Digital watermark for Java programs", SCIS'98-9.2.D, The proceedings of the 1998 Symposium on Cryptography and Information Security, Shizuoka, Japan, Jan. 28-31, 1998

[4] James R. Gosler, "Software protection: Myth or reality?", Advances in Cryptology -- CRYPTO '85, 140-157

[5] A. Herzberg and S. S. Pinter, "Public protection of software", Advances in Cryptology -- CRYPTO '85, 158-179

[6] R. Mori, M. Kawahara, et al., "Superdistribution Architecture", SCIS'90-6A, The proceedings of the 1990 Symposium on Cryptography and Information Security, Nihondaira, Japan, Jan. 31 – Feb. 2, 1990

[7] An Open Specification for Pretty Good Privacy, http://www.ietf.org/html.charters/openpgp-charter.html

[8] ITU-T Recommendation X.509, The Directory: Authentication framework, 1993

[9] Secure Electronic Transaction, http://www.mastercard.com/set/

[10] Public-Key Infrastructure (X.509), http://www.ietf.org/html.charters/pkix-charter.html