通信情報保護學會誌
第 11 券 第 1 號, 2001. 2

국제 보안 API 표준화 동향

박 수 진*, 신 동 명*, 김 학 범**, 최 용 락*

요 약

다양한 응용프로그램에서 개별적인 보안기능의 개발은 중복된 노력의 투자는 물론 호환성과 이식성이 부족하므로 상업적 상품으로 기대되는 모듈화와 내구성을 제공하지 못한다. 따라서, 각 응용프로그램에서 공통적으로 사용할 수 있는 보안 API 표준화가 관심을 갖고 국제적으로 추진되고 있다. 본 논문에서는 널리 알려진 보안 API 표준으로써 IETF의 GSS-API, X/OPEN의 GCS-API, 마이크로소프트의 CryptoAPI, RSA의 Cryptoki, Intel의 CSSM-API를 중심으로 표준화 동향을 조사 분석하였다.

I. 서 론

21세기를 들어서서 전자상거래, e-Business, 무선 인터넷 결재 등 획기적인 형태의 전자적 업무처리가 실용화되면서 사회의 제반활동 구조에 큰 변화를 가져오고 있다. 이러한 사회활동 변화와 더불어 보안 위협요소 역시 더욱 증가하였기 때문에 그러한 위협 요소를 방어하기 위한 보안 기술의 효과적 정합 방법에 대한 연구개발이 대단히 중요해졌다.

그러나, 동일한 암호 서비스 모듈에 대한 각 응용프로그램별 보안기능의 개발은 중복된 노력의 투자는 물론 호환성, 이식성 등이 없으므로 상업적 상품으로서 기대되는 모듈화와 내구성을 제공하지 못한다. 따라서, 효과적으로 개발비용을 절감하고 각 응용프로그램이 공통적으로 사용할 수 있도록 표준화된 보안API(CAPI: Cryptographic Application Program Interface)를 설계할 필요가 있다[1][2][3].

응용프로그램마다 고유의 특성을 고려한 CAPI가 보다 정교하고 다양하게 개발될수록 해당 응용프로그램에서는 효과적으로 사용할 수 있을 것이다. 그러나, 모든 응용프로그램별로 별도의 CAPI를 개발하는 것이 아니고 공통적으로 사용될 수 있는 CAPI를 개발하는 것이기 때문에 암호학적 지식 측면에서 다양한 계층의 사용자들을 고려하여야 한다. 다양한 암호학적 지식을 갖는 사용자들을 지원하기 위하여 최근까지 여러 가지 형태의 CAPI 표준을 만들기 위해 많은 노력이 진행되어 왔다.

최근까지 널리 알려진 보안 API 관련기술로는 1992년 Linn이 IETF에 제안하여 1993년에 인증된 RFC 1508 및 RFC 1509의 GSS-API (Generic Security Services-Application Program Interface)을 비롯하여 X/OPEN의 GCS-API(Generic Cryptographic Service API), 마이크로 소프트의 CryptoAPI, RSA의 Cryptoki, 인텔에서 제안한 CDSA(Common Data Security Architecture)의 CSSM-API(Common Security Service Manager API) 등이 있다[4~7].

이상적으로는 모든 어플리케이션에 쓰일 수 있는 단 하나의 CAPI 적용이 편리하겠지만, 실제로 복잡한 형태의 응용 분야를 지원해 주기 위해서는 여러 계층의 CAPI가 요구될 수 있다. 그러나, 상업적으로 현재와 미래를 예측한 각종 형태의 시스템 개발환경과 응용분야를 모두 만족시키고, 개발자의 암호학적 지식의 차이를 쉽게 해결해줄 수 있는 이상적인 CAPI의 표준화란 대단히 어려운 일이다.

따라서, 보안 API와 관련된 국제 표준화 및 제품 개발 동향을 조사하여 시스템구조, 함수종류 및 자료 구조를 분석하고, 각 시스템 구조가 갖는 알고리

* 대전대학교 컴퓨터공학과 (sjpark@zeus.taejon.ac.kr)
** 한국정보보호센터
※ 본 연구는 2000년 한국정보보호센터 연구과제로 수행되었음

즘 독립성, 암호 모듈 독립성, 사용자 인증, 키 관리 등의 기능 특성을 비교 분석하여 활용성을 검토해야 한다. 이러한 연구분석을 통하여 국내의 보안 API 요구조건을 분석도출하고 관련 정책 및 설계모델을 정립하기 위한 연구가 필요하다.

보안 관련 표준화 기구로는 IETF, The Open Group(TOG), RSA, IEEE, NSA, NIST, ISO 등이 있으며 대부분은 일반적 보안 기술과 연관하여 공통적인 보안 서비스나 각종 알고리즘 및 메커니즘들을 다루고 있고, 보안 API 표준을 직접 정의하고 있는 곳은 IETF, TOG, RSA 등이다.

본고에서는 보안 API의 국제 표준화 동향분석을 위하여 보안API기술인 GSS-API, GCS-API, Cryptoki, CryptoAPI, CDSA 표준화와 직간접적으로 관련된 동향을 분석하고, 현재 가장 활발한 연구가 진행되는 CDSA에 대해서 국제적 개발 동향을 조사하였다.

이러한 주요 5가지의 보안 API 표준화 문서의 분량이 방대하고 지원하는 서비스 방법과 환경이 모두 다르므로 일괄적인 분석 설명이 대단히 어렵다. 따라서, 보안 API와 관련된 기관들의 동향과 주요 표준기술에 대한 표준화 활동동향을 포괄적으로 조사하였다.

II. GSS-API

IETF(Internet Engineering Task Force)는 인터넷 정보기반을 위한 표준 및 주소문제 등에 관해 선도적 역할을 담당하고 있는 ISOC(Internet SOCiety) 산하 기구로써, 특별히 인터넷 기술에 대한 연구 및 표준을 개발하고 있는 기구이다. IETF는 ISO나 ITU와 같은 공식적인 국제표준기구는 아니지만 사실상의 표준을 제정하는 기구의 역할을 수행하고 있다.

현재, IETF 산하에는 기술별로 8개의 분과(Area)가 조직되어 있는데 이중 하나가 정보보호 분과(Security Area)이며, 이 정보보호 분과 산하에는 18개의 작업반(Working Group)이 조직되어 관련 정보보호 기술에 대한 인터넷 표준을 개발하고 있다.

보안 API와 관련된 표준화 문서는 cat(Common Authentication Technology) 워킹그룹의 GSS-API와 Kerberos를 중심으로 다양한 RFC들이 있으며, 계속하여 초안들이 연구 개발되고 있다. cat는 하부 보안 메커니즘과는 무관하게 분산환경에서 사용자들에게 인증, 무결성 및 비밀성을 포함하는 보안 서비스를 제공하기 위한 표준개발 작업을 수행하고 있다. 특별히 MIT에서 개발된 네트워크 인증 시스템인 Kerberos 기반의 인터넷 표준과, 정보보호 서비스에 대한 범용 인터페이스를 정의하는 GSS-API 관련 표준들을 개발하고 있다[10].

최근까지 보안 API와 직간접적으로 연관하여 진행된 표준 목록은 표 2.1, 표 2.2와 같다.

표 2.1 CAPI 관련 표준(RFC)

표준번호	제 목	년도
RFC 2623	"NFS Version 2 and Version 3 Security Issues and the NFS Protocol's Use of RPCSEC GSS and Kerberos V5."	1999
RFC 2078	"Generic Security Service Application Program Interface, Version2"	1997
RFC 1508	"Generic Security Service Application Program Interface."	1993
RFC 2743	"Generic Security Service Application Program Interface Version 2, Update 1."	2000
RFC 2203	"RPCSEC_GSS Protocol Specification. "	1997
RFC 2853	"Generic Security Service API Version 2 : Java Bindings."	2000
RFC 2771	"An Abstract API for Multicast Address Allocation."	2000
RFC 2744	"Generic Security Service API Version 2 : C-bindings."	2000
RFC 2628	"Simple Cryptographic Program Interface (Crypto API)."	1999
RFC 2712	"Addition of Kerberos Cipher Suites to Transport Layer Security (TLS)."	1999
RFC 2614	"An API for Service Location."	1999
RFC 2479	"Independent Data Unit Protection Generic Security Service Application Program Interface (IDUP-GSS-API)."	1998
RFC 2478	"The Simple and Protected GSS-API Negotiation Mechanism."	1998
RFC 2367	"PF_KEY Key Management API, Version 2."	1998
RFC 2292	"Advanced Sockets API for IPv6."	1998
RFC 2025	"The Simple Public-Key GSS-API Mechanism (SPKM)."	1996
RFC 1964	"The Kerberos Version 5 GSS-API Mechanism."	1996
RFC 1961	"GSS-API Authentication Method for SOCKS Version 5."	1996
RFC 1509	"Generic Security Service API : C-bindings."	1993
RFC 1510	"The Kerberos Network Authentication Service (V5)."	1993
RFC 1411	"Telnet Authentication: Kerberos Version 4."	1993

GSS-API는 개방형 분산 네트워크 환경에서 응용프로그램의 소스레벨 이식성을 지원하기 위하여 정보보호 서비스에 대한 범용 인터페이스 규정을 위하여 제안된 표준이다. 1990년대 초반까지 괄목할 만한 정보보호 API가 없던 때에 1992년 Linn이 IETF에 범용 정보보호 API를 제안하였다.

GSS-API는 1993년에 IETF의 CAT 워킹그룹에 의해 RFC(Request For Comments) 1507-1511로 표준화되었다. RFC 1508은 1997년에 RFC 2078인 GSS-API 버전2로 개정되었다. 이 문서들은 초기의 C언어 버전의 GSS-API를 설명하고 있으며, Kerberos 인증 프로토콜과 CAT를 개략적으로 기술하였다. 지금까지 GSS-API는 시스템 내부의 보안 서비스에 접근하는 인터페이스를 이용하기 위해 시도되는 다수의 개발 활동으로 폭 넓은 주목을 받고 있으며 독립적인 데이터 장치 보호를 위한 GSS-API로써 현재 인터넷 표준으로 작성되어 있다.

RFC 2078은 하부의 메커니즘 및 기술들에 지원 가능한 일반화된 형태로 호출자에게 보안 서비스를 제공하므로 다른 환경의 어플리케이션들에 소스레벨의 이식성을 제공한다. 이 표준은 하부 메커니즘 및 프로그래밍 언어 환경 레벨의 독립적인 GSS-API 서비스와 프리미티브들을 정의하며 다른 관련된 표준들에 의하여 보충 지원된다. 즉, 특정의 보안 메커니즘 위에 GSS-API 서비스를 구현하기 위하여 토큰형식, 프로토콜 및 절차들을 정의하는 문서, 그리고 특정 언어 환경을 위하여 명시된 파라메터를 연결하는 문서들이 지원된다. 2000년 1월 RFC 2743은 RFC 2078을 폐기하고 구현상의 경험과 다른 요청들의 응답으로 많은 변경을 추가하여 표준 명세를 만들었다. 이 표준에서는 기존의 20개 함수에서 14개가 더 추가된 34개의 인터페이스 호출함수를 정의하고 있다. RFC 2743 또는 연속적인 버전이 표준트랙에서 GSS-API의 계속적인 진행에 대한 기초가 될 것이다.

표준으로 인증이 된 이래 널리 수용이 되어 GSS-API는 인터넷 관련 기관, X/Open 및 ECMA 등에서 다양한 메커니즘을 기반으로 구현되고 있다.

GSS-API는 통신 프로토콜의 독립성을 제공하고 정보보호 서비스가 어떻게 구현되는지 상세한 지식을 요구하지 않는 장점이 있다.

GSS-API을 적용한 제품들의 동향을 요약하면 표 2.3과 같다.

표 2.2 CAPI 관련 표준(draft)

표준번호	제 목	년도
draft-ietf-ipsec-isakmp-gss-auth-06	A GSS-API Authentication Method for IKE	2000
draft-ietf-dnsext-gss-tsig-00	GSS Algorithm for TSIG (GSS-TSIG)	2000
draft-ietf-cat-sasl-gssapi-01	SASL GSSAPI mechanisms	2000
draft-ietf-cat-krb5gss-mech2-03	The Kerberos Version 5 GSSAPI Mechanism, Version 2	2000
draft-mameli-issll-cops-api-00	The CCAPI (COPS Client Application Programming Interface)	2000
draft-raeburn-cat-gssapi-krb5-3des-00	Triple-DES Support for the Kerberos 5 GSSAPI Mechanism	2000
draft-ietf-cat-srpgm-02	The Secure Remote Password GSS-API Mechanism (SRPGM)	2000
draft-trostle-win2k-cat-kerberos-set-passwd-00	Extending Change Password for Setting Kerberos Passwords	2000
draft-tso-telnet-krb5-04	Telnet Authentication: Kerberos Version 5	2000
draft-ietf-cat-iakerb-04	Initial Authentication and Pass Through Authentication Using Kerberos V5 and the GSS-API (IAKERB)	2000
draft-hornstein-dhc-kerbauth-02	DHCP Authentication Via Kerberos V	2000
draft-ietf-rap-rsvp-newidentity-00	Identity Representation for RSVP	2000
draft-thomas-snmpv3-kerbusm-00	Kerberized USM Keying	2000
draft-smedvinsky-dhc-kerbauth-01	Kerberos V Authentication Mode for Uninitialized Clients	2000
draft-yu-asn1-pitfalls-00	Potential Pitfalls of the Use of ASN.1 in IETF Protocols	2000
draft-hornstein-snmpv3-ksm-01	A Kerberos Security Model for SNMPv3	2000
draft-aboba-pppext-eapgss-01	PPP EAP GSS Authentication Protocol	2000
draft-ietf-cat-kerberos-set-passwd-03	Kerberos Set/Change Password: Version 2	2000
draft-kaushik-radius-sec-ext-00	Radius Security Extensions using Kerberos v5	2000
draft-ietf-cat-kerberos-revisions-06	The Kerberos Network Authentication Service (V5)	2000

표 2.3 GSS-API 적용 제품

제품명	회사	기능
Application Security Toolkit	CyberSafe	인증, 상호인증, 메시지 기밀성, 메시지 무결성, 디지털서명.
NetSP	IBM	해쉬 함수에 기반한 암호구조를 사용.
Simple Public Key Mechanism (SPKM)	Bell Northern Research	GSS-APIdkfo 암호를 지원하기위해 단지 공개키 기술만 사용.
ISM	Bull	패스워드나 스마트카드에 의해 사용자 인증(중앙집중 사용자 인증 서버). GSS-API라이브러리가 클라이언트/서버 어플리케이션을 보호하기위해 OSF DCE 보안 메커니즘을 지원.
Secure Network Communications (SNC)	Common Authenticati on Technologies (CAT)	암호 메커니즘과 사용자 인증을 위해 스마트 카드를 지원.
Entera 4.2	Inprise	3계층 구조 모델로 중간 계층의 3가지 서비스 타입(기능적 서버, 데이터 억세스 서버, 처리서버)을 사용.
TrustBroker Developer Pack	CyberSafe	3가지 SDK(Software Development Kits)중 Application Security SDK가 인증, 데이터 무결성, 데이터 기밀성을 가짐.
AXXiON-Authe ntication	OpenVision	ftp 클라이언트와 서버에 향상된 보안. 인증, 무결성, 암호 제공.

Ⅲ. GCS-API

X/Open은 안전하고 신뢰성 있는 정보기술 하부구조를 개발하고 구현하는데 있어서 발생하는 여러가지 문제점들에 대한 해답을 얻고자 설립되었다. X/Open은 범용 암호화 서비스들을 필요로 하는 아래와 같은 산업체들을 주요 파트너로 협력하여 실용적인 보안 API를 개발하고자 하였다.

- National institute of Standards and Technology (NIST)
- Trusted Information Systems, Inc.
- VSA National Security Agency (NSA)
- RSA Data Security, Inc.
- Fisher international

이들 파트너들은 BULL, Hewlett Packard, IBM, ICL, Olibetti, Open Vision, 그리고 Siemems Nixdorf와 같은 회사들과 함께 작업하였다. 범용 암호 서비스 프로젝트는 아주 큰 효과가 있음이 증명되었고 전세계로부터 회사들이나 정부의 관심을 끌었다.

GCS-API는 X/OPEN 컨소시엄의 보안 활동그룹 (SWG, Security Working Group)이 어플리케이션에 대해 보안 서비스를 제공하기 위하여 만든 API 메커니즘의 집합이다. GCS-API는 암호학적인 지식이 있는 어플리케이션과 지식이 없는 어플리케이션 모두에 대해 설계되었다. 이 어플리케이션은 최소한 암호 알고리즘과 키 관리 운용에 관한 기본적인 사항을 이해해야 한다.

GCS-API는 1997년 이후 지속적인 연구 진행사항이 발표되지 않고 있으며, 관련된 설계 기본개념이 인텔의 CDSA로 발전되었다고 판단된다.

Ⅳ. Cryptoki

RSA사는 표준의 상호 동작성을 보장하기 위하여 산학연 및 정부의 대표들과 협력하여 PKCS(Public Key Cryptography Standards)이라고 부르는 표준 패밀리를 개발하여왔다. PKCS는 공개키와 관련된 기술에 종사하고 있는 컴퓨터 시스템의 개발자들에게 RSA 실험실에 의하여 제공되었다.

RSA 실험실의 목적은 컴퓨터 시스템 개발자들과 함께 연합하여 모든 개발자는 아니라도 대부분의 개발자들이 채택할 수 있는 표준들을 만들어 보겠다는 목표를 갖고 대상 표준들을 개선 및 정비하기 위한 것이었다. 표준을 만들어 가는 과정의 RSA실험실 역할은 다음의 4가지 측면의 입장을 갖고 있다.

- 표준을 설명하는 문서들을 주의 깊게 작성하여 발행
- 변경과 확장에 유용하거나 필요한 개발자와 사용자들로부터의 주장과 조언 수용
- 적절한 시기에 개정된 표준들을 발행
- 구현을 위한 지침이나 참조를 위한 안내 제공

PKCS 개발과정 중에 RSA 실험실이 검토자들로부터의 내용을 반영하고 각 문서상의 최종적인 권한을 가지고 있다. 그러나, RSA 실험실의 목표는 그러한 작업들과 연관하여 경쟁하는 것이 아니고 정규적인 표준의 개발을 촉진하는데 있다. 따라서, PKCS 문서가 정규적 표준의 기본 문서로 접수될 때, RSA 실험실은 그 문서의 소유권을 포기하고 개방된 표준개발 과정에 제공하려는 것이다. RSA 실험실은 관련된 문서화 작업을 계속할 것이고 관련된 자료들을 온라인 홈페이지에 공개하고 있다. PKCS 문서에 관련된 표준에 확장 등록이나 추가적 표준의

제안과 관련된 어떠한 코멘트들을 환영한다고 접촉 창구를 열어놓고 있다.

Cryptoki는 RSA사의 공개키 암호 표준인 PKCS#11이다. Cryptoki는 RSA사에 의해서 표준화되고 있으며, 아무 제약 없이 배포되고 있다.

Cryptoki는 개인의 암호 토큰에 일차적으로 접근하는 하위 수준의 CAPI의 표준을 제공한다. RSA사는 그들이 갖고 있는 라이브러리가 어플리케이션이 암호 토큰과 함께 동작하는 요구를 지원해주기 위해 융통성이 보편적이지 못하다는 것을 인식했기 때문에 Cryptoki를 개발하게 되었다. Cryptoki의 부가적인 목표는 휴대성, 확장성, 일반성, 자원 고유 지원 그리고 알고리즘 독립성이다.

보안 API와 관련된 표준화 활동으로는 암호학적 정보를 가지고 있고 암호 함수들을 수행하는 장치에 Cryptoki이라고 부르는 암호 토큰 인터페이스를 명시한 표준문서가 PKCS#11이다. PKCS#11은 일반적으로 "Cryptoki"라는 API로 불리는데, "Cryptoki"는 "Cryptographic token Interface"를 줄여 쓴 것이고, "crypto-key"의 발음과 같다.

Cryptoki는 응용프로그램을 상세한 암호장치에서 분리시키며, 응용프로그램이 다른 장치나 환경에서 사용될 때 인터페이스를 변경할 필요가 없게 함으로써 이식성을 제공한다. 즉, 개발자들은 "Cryptoki"에 명세된 데이터나 함수들만을 사용함으로써 암호학적 장치의 사용 방법이나, 장치에 대한 세부적인 지식 없이도 암호 응용 서비스를 쉽게 구현 할 수 있다. 이것은, 바꾸어 말하면, "Cryptoki"에 의해 개발된 암호 응용프로그램은 시스템들간에 쉽게 호환 가능하다는 것이다.

Cryptoki에서는 수많은 암호 메커니즘들이 지원되고있으며, 추가적으로 새로운 메커니즘들이 일반적 인터페이스의 변경없이 추가될 수 있다. 추가되는 메커니즘들은 별도의 문서에 발행하는 것이 가능하며, 또한 토큰 업체들이 자신의 메커니즘을 정의할 수 있다. 그러나 상호동작성의 보장을 위하여 PKCS 표준과정을 통하여 등록하는 것이 바람직하다. 표 4.1는 RSA실험실에서 추진한 전체적 PKCS 표준 목록이다[7].

Cryptoki 버전 2.01은 단독 사용자와 교신하는 암호 장치에 사용되는 것을 목적으로 하고 있다. 그래서 범용적 인터페이스에 포함될 수 있는 어떤 특성들이 생략되어 있다. 예를 들면, Cryptoki 버전 2.01은 다중 사용자에 대하여 구별하는 수단을 가지고 있지 않다. 여기서는 하나의 단일 사용자 키와 그와 관련된 적은 수효의 인증서, 그리고 암호를 강조하고 있으며, 비 암호학적 함수를 다루는 장치는 다른 인터페이스의 문제로 인식하고 있다.

Cryptoki는 CDSA와 GSS-API 등의 하부구조로 정합하여 운용하는 방안이 다양한 형태로 연구되고 있으며 CDSA와 인터페이스 함수, 서비스 구조 등을 비교 연구하는 동향들이 있다.

RSA에서 추진되는 보안토큰 관련 표준활동의 문서는 다음의 표 4.2와 같다.

표 4.1 PKCS 표준

구분	제목	버전	일시
PKCS#1	RSA Cryptography Standard, DRAFT 1	2.1	1999
PKCS#3	Diffie-Hellman Key Agreement Standard	1.4	1993
PKCS#5	Password-Based Cryptography Standard	2.0	1999
PKCS#6	Extended-Certificate Syntax Standard	1.5	1993
PKCS#7	Cryptographic Message Standard	1.5	1993
PKCS#8	Private-Key Information Syntax Standard	1.2	1993
PKCS#9	Selected Attribute Types	2.0	2000
PKCS#10	Certification Request Syntax Standard	1.7	2000
PKCS#11	Cryptographic Token Interface Standard	2.1	1999
PKCS#12	Personal Information Exchange Syntax Standard	1.0	1999
PKCS#13	Elliptic Curve Cryptography Standard		1998
PKCS#14	Pseudo-Random Number Generation		1998
PKCS#15	Cryptographic Token Information Format Standard	1.1	2000

표 4.2 보안토큰 관련 표준활동 문서

구분	제목	버전	일시
PKCS#11	Cryptographic Token Interface Standard		1997.12
PKCS#11	Cryptographic Token Interface Standard	2.10	1999.12
PKCS#15	Cryptographic Token Information Format Standard	1.0	1999. 4
PKCS#15	Amendment 1	1.0	1999.12
PKCS#15	Cryptographic Token Information Syntax standard(draft)	1.1	2000. 3

Cryptoki 표준을 적용한 몇 가지 제품의 동향을 요약하면 표 4.3과 같다.

표 4.3 Cryptoki 적용 제품

제품명	회사	기능
CPROV	ERACOM	물리적 보안과 하드웨어 암호 서비스를 제공하는 하드웨어버전 CSA 7000과 소프트웨어 구현된 암호메커니즘을 제공하는 소프트웨어 버전 Software only 가 있음.
SimnaSure CIP	Maithean	공개키 스마트카드 토큰 시스템 기반으로 사용자 식별과 인증, 안전한 정보 저장과 교환, 정보 유효성 검증.
PKCS11 API for Java 1.1	IBM	Abstract 클래스의 실행을 지원하고 JAVA API와 호출되는 PKCS11 라이브러리를 연결
Cyberflex Access SDK V3C	Cyberflex	JAVA 프로그램 가능한 다중 함수와 PC어플리케이션을 위한 암호 스마트카드를 위한 포괄적인 SDK
SPAZIO Data Secure	Primeur	X.509 인증서. RSA 공개키와 개인키. DES, 3DES, RCx 비밀키들을 처리.
PKI-Plus	Baltimore	스마트 카드는 토큰을 위한 Cryptoki 인터페이스를 통해 제공됨.
Entrust Session Toolkit R4.0	Entrust	Crytoki 토큰 유틸리티(win32) : PKCS#11 토큰 리더를 돕는 여러 유틸리티 함수들을 제공.

V. CryptoAPI

마이크로소프트사에서는 윈도우즈 환경하에서 보호할 수 있는 프로그램 정합 모델로써 CryptoAPI를 개발하였다. CryptoAPI는 1996년 7월 버전 1.0이 나왔으며, 1998년 7월 버전2.0이 발표되었다. 버전 1.0에서는 안전한 통신을 위하여 기본적 암호기능으로 기밀성과 무결성을 제공하는 함수들을 포함하고 있다. 버전2.0은 버전1.0의 기본적인 암호 함수 외에 인증서에 관한 함수, 관리 함수, PKCS#7 (ASN.1) 인코딩/디코딩 함수 등이 추가되어 있다. 또한 인증서 관리 함수는 certificate store 이라고 하는 인증서의 유지와 관리기능을 포함하며, 나가는 인증서를 통합하고 메시지에 수신되는 인증서를 검증하는 기능을 갖는다.

CryptoAPI는 Windows 95, 98, 2000 및 Windows NT4.0의 일부로서 제공되는 어플리케이션 프로그램 인터페이스이다. Windows 2000에서 제공되는 CryptoAPI는 1999년 11월 9일 성공적으로 FIPS 140-1인준을 완료하였다. FIPS 140-1은 암호 제품에 의하여 제공되는 보안성을 인준하기 위한 확인과정을 제공한다. 확인과정에서는 그 제품이 참조 구현에 따라서 암호 알고리즘이 정확히 구현되었고, 키 관리와 버퍼들이 올바르게 다루어지는지 검사한다. DES, DSA, SHA-1, Triple- DES, Skipjack 알고리즘들이 사용 가능하다.

일단 암호 함수들이 평가되면 암호 시스템에 의하여 제공되는 전체 보안성을 나타내기 위하여 1부터 4까지의 보안 등급을 할당한다. 일반적 보안등급의 의미는 다음과 같다.

- 레벨 1: 암호 시스템이 범용 PC에서 동작한다.
- 레벨 2: 암호 시스템이 보호된 환경에서 동작한다. 이것은 TCSEC프로그램의 레벨 C2 또는 그 이상의 평가를 받는 운영체제하에서 동작하는 소프트웨어 구현을 포함한다.
- 레벨 3: 암호 시스템이 상당한 노력을 갖고 만들어진 환경에서 동작한다. 레벨 3 시스템은 일반적으로 물리적 보호와 함께 특별한 목적의 하드웨어 또는 소프트웨어를 요구한다.
- 레벨 4: 암호 시스템이 위협요소에 대하여 특별한 장치가 강구된 환경에서 동작한다. 이것은 위태롭게 하는 어떠한 시도를 자동으로 검출하고 대응할 수 있는 하드웨어를 포함한다.

CryptoAPI 는 DES, DSA, SHA-1의 구현에서 모두가 레벨1시험을 완료했지만 전자적 간섭과 호환성의 저항에서는 특별히 레벨3을 받았다. 최근의 Windows NT4.0을 위한 C2평가의 완료는 레벨2로 상향 될 인준의 가능성을 보이고 있다.

Windows 2000이 유사한 평가를 위하여 제출될 것이고 완료될 때는 상향 된 평가레벨을 받을 것으로 기대하고 있다.

CryptoAPI는 공유된 비밀키에 의한 알고리즘과 공개키 암호방식을 모두 지원하며, 인증서 처리 서비스를 지원하고 IETF와 산업표준 등의 PKIX, S/MIME, PKCS, X.509등의 표준을 적용하고 있다.

CryptoAPI는 Win32-환경에서 작동하는 응용프로그램에 인증서, 인코딩 그리고 암호 기능을 추가하여 개발이 가능토록 하고 있다. 응용 프로그램은 CryptoAPI의 구현에 기초적인 내용을 모르더라도 API함수를 이용하여 프로그램을 사용할 수 있다. 현재 인증서 관리와 사용 기능을 추가한 버전 2까지 발표되어 있으며, windows 환경 및 Visual-C들의 개발 도구에 포함되어 있고 인증서를 기반으로 하는 암호시스템의 구현이 가능하다. CryptoAPI을 적용한 몇 가지 제품동향을 요약하면 표 5.1과 같다.

표 5.1 CryptoAPI 적용 제품

제품명	회사	기능
SignaSURE CIP	Datakey	사용자 인증, 안전한 자료 교환 및 정보검증을 위하여 CryptoAPI에서 인식과 검증되는 Datakey의 스마트카드나 스마트 토큰을 사용.
SignaSURE CSP	CPE(Consulting Program and Equipment)	윈도우 95와 윈도우 NT4.0에서 CryptoAPI를 사용하는 미국과 캐나다의 고객에게 보안 서비스 제공 목적.
LapLink 2000 for Windows 95/98/3.1/Nt 4.0/2000	Travsoft	안전한 인터넷 통신 보호를 위하여 CryptoAPI를 포함(3중 DES 암호)하는 다중 레벨 암호.
CryptoObject 1.0	CryptoObject	CrytpAPI에서 제공되는 기본적 암호 함수들을 사용하는 개체들을 정의.
NetPay Payment System	MAITHEAN SPYRUS	카드상의 암호 함수를 접근하는 SET 지불응용을 위해 CryptoAPI CSP 개발.
SecureFile v1.0	Querisoft	CryptoAPI 상에서 구축하는 인터넷 보안 응용을 위한 데스크 탑 제품.
VerSecure	HP	MS는 CryptoAPI를 갖고 HP의 Versecure 암호 구조를 지원.
TrustMaster CSP	Compaq	MS의 CrytpAPI 인터페이스하에서 디지털 서명, 해쉬, 키 생성 및 데이터 암호화작업을 수행하는 범용 암호 서비스 제공자.

Ⅵ. CDSA

The Open Group(TOG)은 1996년에 설립되었다. TOG는 개방형 시스템 명세의 채택, 명세서를 준수하는 기술의 발전과 전체 기업 컴퓨팅 시장에서 그들 사용의 촉진을 포함한 "전체 개방형 시스템 처리 강화와 합리화"로 X/Open과 OSF를 합병하여 생성되었다.

보안관련 분야로는 Authorization, Authentication, CDSA and CSSM, Distributed Audit Services (DAS), Distributed Security Framework (XDSF), Firewalls, Information Labelling, Java Security, Public Key Infrastructure (PKI), Secure eCommerce, Single Sign-On, SmartCards, Trust, Wireless and Mobile가 있다.

TOG에서 CAPI관련 표준문서로는 표 6.1과 같다.

표 6.1 CAPI 관련 표준문서

표준번호	제목	년도
G901	CDSA Explained - An indispensable guide	1999
C441	Generic Security Service API (GSS-API) Base	1995
C914	Common Security: CDSA and CSSM, Version 2 (with corrigenda)	2000
C902	Common Security: CDSA and CSSM, Version 2	1999
C707	Common Security: CDSA and CSSM	1997
P442	Generic Cryptographic Service API (GCS-API)	1996
S307	GSS-API Security Attribute and Delegation Extensions	1994

1999년 5월 인텔은 "CDSA-API Specification, CDSA Version 2.0 release 3.0"이라는 검토 버전의 개정을 통하여 1999년 11월에는 TOG의 CDSA 버전 2 기술 표준으로 C902를 발행하였다. 이 개정판은 초기의 C707 표준으로부터 실질적인 변화에 영향을 많이 준 구현상의 경험들을 기반으로 보완되었다.

결과적으로 애플과 인텔의 상세한 구현작업에 의하여 C902(CDSAv2)표준에 교정된 내용들이 포함되었고 CDSA API와 서비스 제공자 인터페이스 함수 호출에 중복된 기술 정보들을 삭제하고 표준을 재 구조화하였다. 이러한 결과들은 2000년 5월 "Common Security: CDSA and CSSM, Version 2(with corrigenda) Technical Standard, C914"로 발행되었다.

인텔의 CDSA 기술은 아무도 사용하지 않는 가

치 없는 소프트웨어가 아니며 수많은 PC 제조 업체들이 지원하고 있다. 즉, IBM, Apple, Compaq, Hewlett-Packard, AT&T, Motorola 및 보안 소프트웨어 제작자 Certicom 등을 포함하여 광범위하게 연구개발이 진행되고 있다.

예를 들면, Hewlette-Packard의 HP Unix 11, IBM의 AIX, OS390과 OS400에 탑재되었고 Apple의 Mac OS에서 지원되고 있으며, Compaq는 True64 유닉스 개정판에서 CDSA를 지원할 것이라고 발표하고 있다[8-10].

표 6.2 CDSA 적용 제품

제품명	회사	기능
HP-UX 11	Hewlett-Packard	CDSA CSP가 사용가능. CSP는 데이터 암호화, 메시지 다이제스트 해싱, 디지털 서명과 같은 암호학적 동작 수행
Apple OS	Apple	보안 구조를 운영체제의 기본 요구사항을 간주하고 Apple OS에 확장성과 융통성을 갖고 쉽게 지원할 수 있는 구조를 요구.
CST(Crypto Systems Toolkit)	Baltimore	강력한 정보 보안 시스템을 구축할수 있도록 개발자들에게 제공하는 암호학적 컴포넌트들의 집합. 개발자들에게 안전한 어플리케이션을 구축할 수 있도록 지원. PKI 시스템과 상호동작 허용.
TrustWay Crypto	Bull	유럽 네트워크 보안과 인터넷 기반 처리를 안전하게 하기 위한 암호학적 서브 시스템을 제공. 소프트웨어 구조에 사용된 PKCS#11 API와 CDSA를 지원하며, 이러한 개방 표준의 사용은 산업 응용과 특정 개발을 쉽게 통합할 수 있도록 허용.
KeyWokr Toolkit	IBM	키 복구 모듈 매니저 범용적 특권 메커니즘 CSP상호 동작을 위한 지원(CSSM 키 형식 탑재) 다중 플랫폼 다중-스레딩 어플리케이션 지원 추적설비(trace-facility) 어플리케이션 개발 툴(서명)

비록 CDSA의 버전 3.0이 리눅스에 대하여 개방된 소스로 추진되지만 윈도우즈의 코드가 먼저 될 것으로 보인다. 리눅스 버전은 32-비트와 64-비트 버전에서 itanium 프로세서의 사용을 위하여 최적화 될 것이다. 인텔의 파트너 Bull은 CDSA 3.0을 리눅스에 접목할 예정이고 CDSA 2.0은 이미 리눅스에 이식하였다. 그러나, 그 기술이 개방된 소스로 사용 가능하지 않다. 개방형 소스로 CDSA를 개정하는 것은 회사의 전체 OS전략에서 작은 부분이며, 리눅스가 인텔의 칩에 대하여 운영체제로써 어떻게 중요한 역할을 하는지 보여 주는 것이다.

이러한 CDSA는 최근에 많은 산업체에서 관심을 갖고 채택하고 있으며 몇가지 사례를 요약하면 표 6.2와 같다.

Ⅶ. 결 론

본 연구에서는 인터넷 등의 개방형 정보통신망을 기반으로 하는 다양한 분산 응용 프로그램에서 공통적으로 이용할 수 있는 보안 API 국제 표준화 동향을 분석하였다.

GSS-API는 보안에 관한 지식이 없는 일반 프로그래머가 쉽게 적용할 수 있으나, Kerberos나 SESAME와 같은 하위 인증 구조를 갖고 있어야 한다는 단점이 있으며, Cryptoki는 암호 서비스 하위구조를 지원하는 API로서, 다른 CAPI에게 보안 알고리즘을 제공하는데 적당하다. GCS-API는 암호학적 지식이 있는 프로그래머는 물론 암호학적 지식이 없는 프로그래머에게도 보안 서비스를 제공할 수 있다. CryptoAPI는 운영체제에서 직접 보안 서비스를 제공하는 장점이 있는 반면, 윈도우 기반 시스템에 종속적으로 동작한다. CDSA는 최근의 인증서 저장 운영 및 신뢰성 검증과 모듈별 장래의 확장성, 모듈의 무결성 검증 등을 고려하여 다양한 조건의 평가기준을 만족시킬 수 있도록 제안하였다.

앞으로 국내외의 동향을 보다 더 조사하여 국내의 실정에 맞는 요구사항을 도출하고 타당성 있는 모델 설정과 함께 정립된 방안들의 연구개발이 필요하다.

참 고 문 헌

[1] W. Caelli, I. Graham, L. O'Connor, "Cryptographic Application Programming Interfaces(APIs)", Computer & Security, 1993.

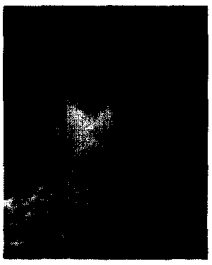
[2] NSA Cross Organization CAPI Team, "Security Service API : Cryptographic API Recommendation Updated and Abridged Edition", International ICE/CAPI Workshop, RSA, July 25, 1997.

[3] NSA Cross Organization CAPI Team, "Security Service API : Cryptographic API Recommendation Second Edition", International ICE/CAPI Workshop,

RSA, July 1, 1996.
[4] J. Linn, "Generic Security Service Application Program Interface, Version 2", RFC2078, January, 1997.
[5] X/Open Company Ltd., "Generic Cryptographic Service API(GCS-API) Base - Draft 8", X/Open Preliminary Specification, April, 1996.
[6] J. Wray, "Generic Security Service API Version 2 : C-bindings", August 7, 1998.
[7] RSA Laboratories, "PKCS #11: Cryptographic Token Interface Standard", RSA, December 22, 1997.
[8] Preparing your LDAP Directory for HP-UX Integration Version1.00, Hewlett Packard, January 19, 2000.
[9] Cylink Cryptokit, CYLINK.
[10] Compaq Trust Master CSP, COMPAQ.

〈著者紹介〉

박 수 진 (Soo-Jin Park)
1997년 : 대전산업대학교 제어계측공학과 졸업(공학사)
1999년 : 아주대학교 대학원 컴퓨터공학과 졸업(공학석사)
1999년~현재 : 대전대학교 대학원 컴퓨터공학과 박사과정
관심분야 : 컴퓨터시스템 및 네트워크보안, 보안API



신 동 명 (Dong-Myung Shin)
1997년 : 대전대학교 컴퓨터 공학과(학사)
1998년~ 2000년 : 대전대학교 대학원 컴퓨터 공학과 (석사)
2000년 1월~2000년 6월 : Nitz(주)위촉연구원(IPsec API 설계 & 보안API 구현)
2000년~현재 : 대전대학교 대학원 컴퓨터 공학과 (박사과정)
관심분야 : 컴퓨터·네트워크 보안, 보안 API, PKI, IPsec

김 학 범 (Hark-Beom Kim)
1990년 : 중앙대학교 대학원 전자계산학과(석사)
1996년~1999년 : 아주대학교 대학원 컴퓨터공학과 박사과정 수료
1991년~1996년 : 한국전산원 주임연구원
1997년~1998년 : TTA/TC10/SG3(시스템보안 연구위원회) 간사
2000년~현재 : TTA/TC10(정보보호 기술위원회) 간사,
TTA/TC10/SG3(시스템보안 연구위원회) 의장
한국통신정보보호학회 학회지 편집위원
1996년~현재 : 한국정보보호센터 선임연구원, 표준과제책임자
관심분야 : 컴퓨터·네트워크 보안, 접근통제, 정보보호 표준화

최 용 락 (Yong-Rack Choi)
1976년 2월 : 중앙대학교 전자계산학과 공학사
1982년 2월 : 중앙대학교 전자계산학과 이학석사
1989년 2월 : 중앙대학교 전자계산학과 이학박사
1979년~1980년 : (주) 금성사 전산실
1982년 3월~1986년 1월: 한국전자통신연구원 선임연구원
1983년 9월~1984년 8월: 스웨덴 스톡홀름 LM ERICSSON 방문연구원
1986년 2월~현재: 대전대학교 컴퓨터공학과 교수
관심분야 : 컴퓨터 시스템 및 네트워크 보안, 전자상거래 보안, 접근통제, 보안 API, PKI, IPsec