핀테크 보안을 위한 대규모 전자서명 기반구조

이동욱·박승규·최종욱 (마크애니)

목차

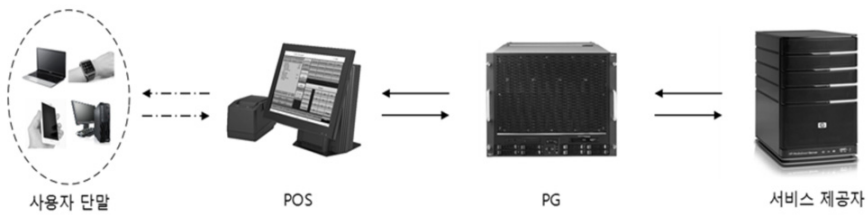
1. 서 론

금융과 기술의 융합인 핀테크는 모바일 결제 및 송금, 개인자산관리, 크라우드 펀딩 등 우리의 생활에 편리함을 주는 기술로 각광받고 있다. 그러나 사용자의 편의성이 중시되는 핀테크는 여러 회사의 협력과 복잡한 네트워크를 형성하게 되고 많은 양의 개인정보와 상용 데이터들이 사용되게 됨에 따라 보안위협에 노출 될 가능성이 높다. 특히 여러 회사들을 통해 구성된 복잡한 네트워크에서 대규모로 송/수신 되는 데이터들에 대한 위/변조 및 악성코드를 이용한 공격이 확연히 증가 될 것이다. 따라서 대규모로 송/수신 되는 데이터들 전부에 대한 무결성 및 인증을 수행하여 해당 위협으로부터의 방어체계를 구축해야 한다.

일반적으로 잘 알려진 전자서명 방법으로는 PKI 기반구조의 전자서명 방법이 있다. PKI 기반 구조의 전자서명 방법은 대규모 정수 연산을 수행하는 공개키 기반의 알고리즘을 사용함으로써 무결성 보장 및 인증을 수행한다. 그러나 PKI를 이용하여 대규모 전자서명을 하는 것은 정수 연산의 특성으로 인해 전자서명 생성 및 검증에 소요되는 시간이 많아 대규모 전자서명 수행에 적합하지 않다. 따라서 대규모 전자서명을 수행하기 위해서는 PKI 기반구조의 전자서명 보다 적합한 전자서명 방법이 필요로 하며 본 고에서는 대규모 전자서명에 적합한 전자서명 기반구조에 대해 설명 및 실제 적용방법에 대해 설명한다.

2. 핀테크에서의 대규모 전자서명의 필요성

핀테크에서는 IT회사, PG사, 통신사, 금융회사 등이 협력하여 서비스를 제공하며, 네트워크를 기

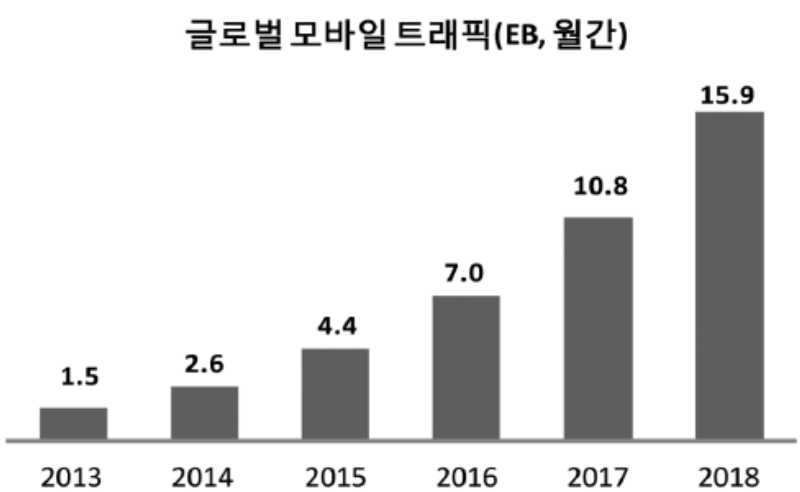


(그림 1) 핀테크 서비스 구성 요소

반으로 각각의 회사들이 연결된다.

이렇게 연결된 네트워크를 통해서 PG사, 금융회사 등으로 개인정보, 거래내역 등이 전송된다. 전송되는 데이터의 양은 해를 거듭할수록 증가하고 있다. CISCO는 글로벌 모바일 트래픽이 2018년에 190EB(Exabyte), 모바일 지급결제시장의 규모가 2017년에 7,210억 달러에 달할 것으로 예상하고 있다[1]. 또한 대표적인 핀테크 서비스인 중국의 알리페이에서 자국 회원 수 3억명을 보유하고 있고, 모바일 결제 대행 건수 4,518만 건을 기록하고 있다[2]. 이러한 수치들은 핀테크에서 매우 대규모의 데이터가 처리되고 있음을 보여준다.

그러나 네트워크가 복잡하게 연결되어있고 회사간 대규모 데이터를 전송하는 점에서 보안 안전성에 대한 우려를 낳고 있다. 공격자는 핀테크 네트워크 구성 요소에 대해 세션 하이재킹, 중간자 공격, APT 공격 등을 이용해 악성코드를 심거나 전송 데이터를 조작하여 연결된 시스템에 대해 침해를 일으킬 수 있다. 실례로 2013년 말 미국 타겟(Target)사의 개인정보유출 사고의 경우, 공격자가 서비스 체인 상에서 가장 취약한 계약업체의 서버를 공격하였고 이후 연결된 POS기기에 악성코드를 전송하였다. 사용자가 POS기기를 통해 카드 결제를 할 때 카드정보가 공격자에게 전송되었고 1억 1천만 건의 카드정보가 유출되었다[1]. 이는 비 인가된 데이터의 전송 및 수신이 허가되어 발생한 사고로 전



(그림 2) 글로벌 모바일 트래픽, CISCO

송 데이터에 대한 위변조를 방지하는 것이 핀테크 서비스에서 매우 중요하다는 것을 보여주는 사례이다.

이러한 위협에 대응하기 위해 핀테크 네트워크 내에서 오고가는 전송 데이터의 무결성을 보장해야하며 무결성 보장을 위해 전자서명 기술을 사용할 수 있다. 전통적으로 전자서명 기술은 메시지 인증을 위해서 사용되어왔다. 이를 핀테크에 적용한다면, 핀테크의 전송 데이터에 대해 위변조된 데이터를 탐지하여 공격자에 의해 조작된 데이터에 의한 위협을 방지할 수 있다.

3. PKI 기반 구조를 이용한 대규모 전자서명의 한계

PKI는 공개키 기반 구조로서 국내외에서 데이터 암/복호화, 전자서명 등 다양하게 활용되고 있으며, 특히 국내에서는 사용자 인증 및 전자서명으로 금융 및 IT 산업 전반에 걸쳐 활용되고 있다. PKI 공개키 기반구조에서 사용되는 전자서명 알고리즘으로는 RSA-PSS, DSA, KCDSA 등의 알고리즘이 있으며 국내 대부분의 전자서명은 RSA-PSS 알고리즘을 활용하는 것으로 알려져 있다. 해당 전자서명은 RSA 공개키 암호 알고리즘을 기반으로 대규모 정수 연산을 필요로 하며 개인키를 사용해 메시지를 암호화 한 것을 전자서명으로 하는 방법이다. RSA 암호 알고리즘의 암호화 연산속도의 경우 Intel 에서 발표한 <표 1>과 같다[3].

<표 1>은 하스웰 Intel core i7 4770 Processor 에서 수행된 1024byte 길이의 입력에 대한 지수 연산의 성능이며 해당 연산을 통해 RSA-PSS 2048bit의 개인키를 사용한 암호화, 즉 전자서명 생성을 수행할 경우 <표 1>의 1024 bit의 2배 만큼의 연산속도가 필요하다고 Intel에서 말하고 있다. 해당 <표 1>의 결과를 인용하면 RSA-PSS 2048bit 전자서명의 서명 생성/검증을 CPU 하스웰 3.5GHz를 사용해 수행 할 경우 단일코어에서 초당 1327개의 전자서명을 생성/검증 할 수 있는 것을 이론적으로 추정 할 수 있다.

〈표 1〉 Modular Exponentiation Performance(cycles)1[3]

Algorithm	Sandy Bridge	Haswell
512bit	231,750	173,348
1024bit	1752,092	1,318,895

앞서 2장에서는 핀테크에서의 대규모 전자서명의 필요성에 대해 기술 하였다. 그러나 PKI 공개키 기반 구조의 대표적 전자서명인 RSA-PSS 2048bit의 서명생성/검증의 이론적 성능 추정치에 따르면 해당 전자서명 방법을 대규모 전자서명에 적용하기에는 한계성이 뚜렷함을 알 수 있다. 그리고 RSA-PSS 전자서명 뿐만 아니라 PKI 기반구조에서 사용되는 DSA, KCDSA 등 또한 공개키 기반을 사용하는 알고리즘으로써 대규모 정수연산이 발생함에 따라 전자서명의 생성/검증 속도 또한 한계성이 뚜렷할 것이다.

따라서 대규모 전자서명을 수행하기 위해서는 기존의 PKI 공개키 기반 구조보다 대규모 전자서명에 더 적합한 새로운 전자서명 기반 구조가 필요하다.

4. 대규모 전자서명을 위한 전자서명 기반 구조

전자서명 방법은 최근까지 꾸준히 연구 되고 있으며, 대규모 전자서명을 위한 방법으로 키를 사용하지 않는 전자 서명 생성 및 검증을 통한 성능 향상과 키관리에 따른 자원소모를 없애는 효율적인 전자서명의 역할에 중점을 맞춰 연구 되고 있다. 최근 언론 및 포럼을 통해 알려진 KIDS 전자서명 기반

구조는 키를 사용하지 않는 기술로 대규모 전자서명에 적합한 전자서명으로 소개되고 있다.

4.1 KIDS 전자서명 기반 구조

KIDS 전자서명 기반 구조는 전자서명/검증을 서버기반을 통해 수행하는 전자서명 기반기술로 해시 및 Merkle tree 알고리즘[4]을 이용한 시간 증명 방법[5]과 서버기반의 전자적 대리서명 방법으로 구성된다.

KIDS 구조는 클라이언트가 전자서명 시스템에 자신을 인증하고 서명하고자 하는 데이터의 메시지(축약값)를 서명서버로 전송, 서버에서 해시만을 이용한 전자서명 생성해 클라이언트로 전송하고 해당 클라이언트는 전자서명을 제3자에게 보내는 형식으로 KIDS의 전자서명 기반 구조의 전체 구조도는 다음 (그림 3)과 같이 나타낼 수 있다.

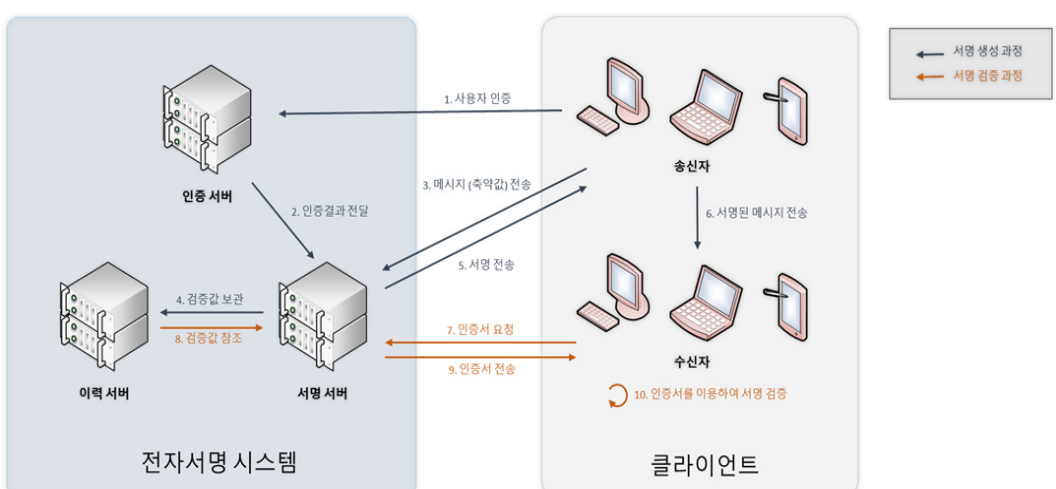
4.2 KIDS 기반 구조의 전자서명 방법

KIDS 전자서명 기반 구조는 해시 및 Merkle tree 알고리즘을 사용하는 전자서명 방법이다. 해당 전자서명 방법은 해시 함수를 통해 데이터에 대한 무결성을 보장하고, Merkle tree를 구성해 최종 값을 생성한다[4]. Merkle tree에서 각 최하위 값에서부터 최종 값을 생성해 내기 위해 필요로 하는 값들은 특정 되며 유일성이 보장된다[4]. KIDS 전자서명 기반구조에서의 전자서명 값은 특정되는 해당 값을 체인으로 구성한 것이다.

KIDS 전자서명 기반구조에서 전자서명을 생성하기 위해서는 해시 함수의 연산만 수행이 되며 시간 증명을 위한 방법[5] 또한 해시 함수만을 사용함에 따라 해당 기반구조에서는 키가 필요치 않다. 따라서 키 관리 및 키 생성 등에 소요되는 자원이 불필요 하며 복잡한 연산이 아닌 단순 해시만을 함으로써 효율성을 보장 할 수 있다.

4.3 KIDS 기반구조의 전자서명 생성/검증 방법

KIDS 기반구조의 전자서명 생성/검증 과정은 (그림 3)에 따라 다음과 같이 설명 할 수 있다.



(그림 3) KIDS 전체 구조도

· 서명생성 검증/과정

Ⅰ. 전자서명을 생성하고자 하는 사용자는 전자서명 시스템의 인증서버를 통해 고유 사용자로서 사용자 인증을 요청한다.
Ⅱ. 인증서버에서 고유 사용자로 인증해 서명서버로 사용자 인증을 완료 한다.
Ⅲ. 사용자는 전자서명을 생성하고자 하는 전자적 데이터에 대한 메시지(축약값)를 서명 서버로 전송한다.
Ⅳ. 서명서버에서는 해시와 Merkle tree를 이용해 서명을 생성하며 Merkle tree의 최종 값을 검증 값으로 보관하기 위해 이력 서버로 전송한다
Ⅴ. 생성된 서명 값은 사용자에게 전송한다.
Ⅵ. 사용자는 서명된 메시지를 수신자에게 보낸다.
Ⅶ. 수신자는 서명의 검증을 위해 서명서버에 검증에 필요로 하는 확장된 전자서명 값(인증서)를 요청한다.
Ⅷ. 서명서버는 보관된 검증 값을 참조해 확장된 전자서명 값(인증서)을 생성한다.
Ⅸ. 서명서버는 생성된 확장된 전자서명 값(인증서)를 수신자에게 전송한다.
Ⅹ. 수신자는 확장된 전자서명 값을 이용해 서명을 검증한다.

4.4 대규모 전자서명을 위한 효율성

대규모 전자서명을 수행하기 위한 전자서명에서 PKI 기반구조는 3장에서 설명 한 것과 같이 대규모 전자서명을 수행하기 위한 한계성이 나타난다. 이는 공개키 기반 구조가 대규모 정수 연산을 함으로써 전자서명 생성/검증 시에 소요되는 시간이 많다는 것이다. 이에 반해 KIDS는 해시 값과 Merkle Tree를 이용해 전자서명을 생성/검증함으로서 복잡한 정수 연산을 배제 하였고 키를 사용하지 않는 방법으로 키 관리에 필요한 자원 소요가 없다. KIDS 기반구조의 전자서명 생성/검증 속도는 해시 값 생성과 Merkle Tree의 구성을 통한 해시 값 계산의 속도에 따라 결정이 되며 n개의 전자서명 생성하는데 걸리는 시간은 $n-1$ 개의 해시를 수행 하는 것과 동일하다. 해시함수의 속도는 2장의 Intel에서 발표한 RSA 암호 알고리즘 수행에 대한 측정과 동일한 CPU 환경에서 수행한 것으로 다음 <표 2>와 같다.

해당 해시 함수의 속도 cycles/byte로 2장 RSA 암호 알고리즘의 입력 값인 1024byte를 해시 할 경우 <표 3>과 같이 나타 낼 수 있다.

<표 3>은 2장 RSA 암호 알고리즘과 동일한 입력에 대해 해시를 수행한 속도의 값으로 해당 SHA-256와 RSA 암호 알고리즘의 속도를 비교할 경우 Haswell에서 약 300배 빠른 것을 알 수 있다. 이를 바탕으로 SHA-256을 적용한 KIDS의 전자서명은 Haswell 에서 n개의 서명을 생성하는데 $(n-1)\times 8{,}796.16$이며, Haswell 3.5GHz

〈표 2〉 SHA Single Buffer Performnace (cycles/byte)[3]

Algorithm	Sandy Bridge	Haswell
SHA-1	5.44	3.80
SHA-256	12.82	8.59
SHA-512	8.60	6.27

〈표 3〉 SHA Single Buffer Performnace : 1024byte input (cycles)

Algorithm	Sandy Bridge	Haswell
SHA-1	5,570.56	3,891.2
SHA-256	13,127.68	8,796.16
SHA-512	8,860.4	6,420.48

의 싱글 코어에서 초당 397,900개로 추정할 수 있다.

KIDS의 전자서명과 PKI기반의 RSA-PSS 전자서명과 비교는 다음 <표 4>와 같다.

<표 4>를 통해 알 수 있듯이 동일한 환경에서 전자서명 생성/검증 측정시 KIDS 기반구조의 전자서명이 PKI기반 구조의 일반적인 RSA-PSS 전자서명 보다 압도적으로 빠르며, 대규모 전자서명에서 PKI 기반구조의 전자서명 보다 KIDS의 전자서명이 더 효율적일 것이다.

〈표 4〉 KIDS전자서명과 RSA–PSS 전자서명 속도 비교

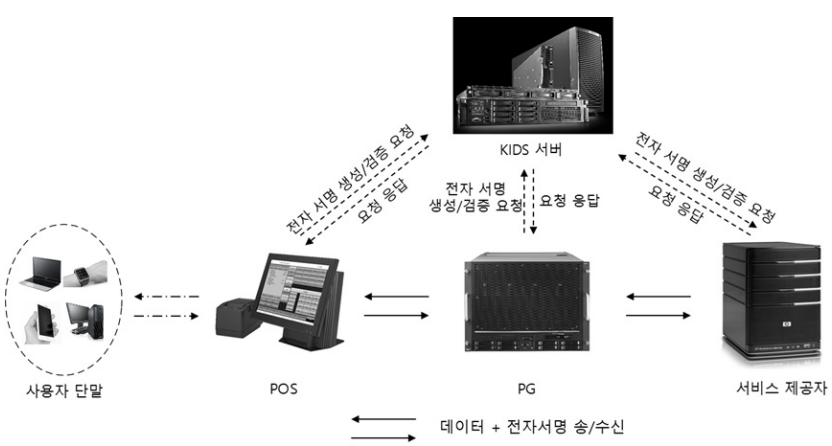
전자서명 방법	초당 전자서명 생성 속도
PKI(RSA-PSS 2048bit)	1,327 개
KIDS(SHA-256)	397,900 개

5. 핀테크에서의 KIDS 전자서명 기반구조 적용

금융회사, PG사, IT회사, 통신사, 창업기업등의 회사에서 제휴 또는 협력을 통해 제공하는 핀테크 서비스에서 각각의 회사 간의 교류되는 데이터들의 대한 KIDS 전자서명 기반구조 적용은 다음 그림과 같이 나타 낼 수 있다.

(그림 4)는 각각의 POS, PG, 서비스 제공자는 전송해야 되는 데이터에 대해 KIDS 서버를 통해 전자서명을 생성 후 각 필요에 따른 데이터를 전송하는 적용 방법이다.

최근 집계된 POS 단말기는 300만대[8]이며 삼성페이, 애플페이 등 간편결제 및 모바일 결제 등이 활발하게 일어나게 됨에 따라 POS, PG, 서비스 제공자로 이어지는 네트워크에서의 데이터의 교류들이 대규모로 늘어나고 있다. KIDS 전자서명 구조에서는 CPU 4 Core 이상의 단일 컴퓨팅 환경에서 이론적 추정치로 초당 최소 160만개 이상의 전자서명 생성 검증을 할 수 있으며, 또한 KIDS의 서버 수가 늘어남에 따라 초당 생성 할 수 있는 전자서명 개수는 해당서버에서 생성되는 전자서명의 개수만큼 증가한다. 즉 사용자 수가 증가하고 사용되는 데이터가 대규모화 되어도 KIDS 전자서명 기반구조를 적용할 경우 사용되는 모든 데이터에 대해 전자서명을 수행하고 데이터에 대한 무결성 및 인증을 보장 할



(그림 4) 핀테크에서의 KIDS 전자서명 기반구조 적용

수 있다.

6. 결 론

금융과 기술의 융합인 핀테크 서비스에서는 보안을 위해 서비스에 사용되는 방대한 데이터들에 대한 대규모 전자서명을 함으로써 무결성 보장 및 인증을 필요로 하다. 그러나 현재 일반적인 사용하고 있는 PKI 기반구조의 전자서명 방법으로는 해당 대규모 전자서명을 수행하기에는 한계성을 가진다. 이는 해당 전자서명이 대규모 정수 연산을 수행함에 따라 전자서명의 생성/검증의 속도가 대규모 전자서명을 하기에는 느리기 때문이다. 따라서 대규모 전자서명에 적합한 기반구조가 필요로 하며 본 고에서 대규모 전자서명에 적합한 KIDS 전자서명 기반구조를 설명하였다. 해당 방법은 이론적 수치로 초당 160만개의 전자서명을 생성/검증 할 수 있는 전자서명 기반구조로 기존의 PKI 기반구조보다 압도적으로 빠른 전자서명 생성/검증을 수행할 수 있다. 핀테크 서비스에서 필요로 하는 무결성 보장 및 인증을 위해 는 본 고에서 설명한 KIDS 전자서명과 같은 대규모 전자서명 방법이 필요하며 핀테크 서비스에서의 보안 안정성을 위해서는 대규모 전자서명을 수행 할 수 있는 적절한 전자서명 기반 구조를 적용해야 할 것이다.

참 고 문 헌

[1] 박정국, “핀테크(Fintech)와 정보보안”, 정보과학회지 제 33권 제 5호, pp. 23-32, 2015년 5월

[2] KISA, INTERNET & SECURITY FOCUS, 2015년 2월

[3] Intel Corporation. "Haswell Cryptographic Performance", http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/haswell-cryptographic-performance-paper.pdf, July 2013

[4] Merkle, R. C. (1988). "A Digital Signature Based on a Conventional Encryption Function". Advances in Cryptology — CRYPTO '87. Lecture Notes in Computer Science 293. pp. 369, 1988

[5] D. Bayer, S. Haber and W. Scott, "Improving the Efficiency and Reliability of Digital Time-Stamping", Sequences II: Methods in Communication, Security and Computer Science, pp.329-334, 1993

[6] Wikipedia contributors. "Comparison of SHA functions" https://en.wikipedia.org/w/index.php?title=SHA-2&oldid=681317424

[7] P. Hammarlund "4th Generation intel Core Process codenamed Haswell", http://www.hotchips. org/wp-conten t/uploads/hc_archives/hc25/HC25.80-Processors2-epub/HC25.27.820-Haswell-Hammarlund-Intel.pdf" August 2013

[8] 손정현, "원활한 지급결제를 위한 소액결제인프라 활성 방안", 한국산업협회자료 2012

저 자 약 력

이 동 욱

이메일: dulee@markany.com

- 2000년 아주대학교 컴퓨터공학 (학사)
- 2000년~2002년 MJL Technology 연구원
- 2002년~현재 ㈜마크애니 연구부소장
- 관심분야 : 컴퓨터보안, 정보보안, IoT, 핀테크

박 승 규

이메일: skpark@markany.com

- 1989년 연세대학교 전기공학과 (학사)
- 1991년 연세대학교 전기공학과 (석사)
- 1996년 연세대학교 전기공학과 (박사)
- 1996년~1998년 삼성SDS ERP사업팀 책임
- 1999년~2002년 쌍용정보통신 컨설팅사업팀 차장
- 2002년~2003년 (사)기업정보화지원센터 정보화수준 평가본부 본부장
- 2004년~2005년 연세대학교 정보산업공학과 연구교수
- 2005년~2014년 동부CNI Application Service사업팀 부장
- 2015년 ~현재 ㈜마크애니 기술연구소 연구소장
- 관심분야 : 금융IT, 암호화, 네트워크 보안, 해킹

최 종 욱

이메일: juchoi@markany.com

- 1982년 아주대학교 산업공학과 (학사)
- 1988년 University of South Carolina (MIS 박사)
- 1989년~1991년 KIST 인공지능 연구실장
- 1991년~현재 상명대학교 소프트웨어학부 교수
- 1999년~현재 ㈜마크애니 대표이사
- 관심분야 : 컴퓨터보안, 저작권관리기술, 인공지능, 정보보호 응용기술