Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.33, NO.1, Feb. 2023 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2023.33.1.13

행정전자서명 암호체계 기술 현황 분석 및 고도화 방향*

정 영 훈,[1†‡] 노 동 영,[2] 구 본 욱[2]
[1,2]ETRI 부설연구소 (선임연구원, 책임연구원)

Analysis of the Cryptosystem of the Korean Government Public-Key Infrastructure and Ways to Improve It*

Younghoon Jung,[1†‡] Dongyoung Roh,[2] Bonwook Koo[2]
[1,2]Affiliated Institute of ETRI (Senior Researcher, Principal Researcher)

요 약

행정전자서명(GPKI)은 대한민국 중앙정부, 지방정부, 공공기관 등이 자체행정과 대민서비스를 제공할 때 사용하는 공개키 기반구조로 정보시스템의 인증 및 보안 기능을 수행한다. 현재 행정전자서명에서 사용하는 암호체계는 2000년대 초반에 구축하였으며, 2010년에 한 차례 고도화를 진행한 후 10여 년이 지났다. 지난 10여 년간 암호를 포함하여 정보보안은 많은 변화를 겪어왔으며, 앞으로도 수많은 변화를 맞이하게 될 것이다. 따라서 행정전자서명의 지속 가능한 안전성을 위해서는 현시점에서 암호체계의 안전성을 재검토할 필요가 있다.

이 논문은 행정전자서명 암호체계에 사용된 기술 및 표준의 현황과 안전성을 분석하여, 안전성이 저하된 암호알고리즘을 사용하거나, 폐지 또는 대체된 표준을 참고하는 사례와 안전성 상향을 위해 조정해야 할 파라미터 등을 식별하였다. 그리고 이를 바탕으로 행정전자서명 암호체계 고도화를 위한 암호알고리즘 및 관련 기술의 개편 방향을 제시한다.

ABSTRACT

Korean Government-PKI (GPKI) is a public-key infrastructure which provides authentication and security functions for information system used by central government, local governments, and public institutions of the Republic of Korea to provide their own administrative and public services. The current cryptosystem of GPKI was established in the early 2000s, and more than ten years have passed since the last improvement in 2010. Over the past decade or so, the information security, including cryptography, has undergone many changes and will continue to face many changes. Therefore, for the sustainable security of GPKI, it is necessary to review the security of the cryptosystem at this point.

In this paper, we analyze the current status and the security of technologies and standards used in the system. We identify cryptographic algorithms with degraded security, international standards which are obsoleted or updated, and cryptographic parameters that should be revised for the high security level. And based on this, we make several suggestions on the reorganization of cryptographic algorithms and related technologies for the security enhancement of GPKI.

Keywords: PKI, GPKI, Cryptosystem, Certificate, Digital Signature

Received(10. 20. 2022), Modified(01. 11. 2023), Accepted(01. 11. 2023)

* 이 논문은 2022년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (2021-0-00046, 국가·공공 정보시스템 안전성 및 활용성 제고를 위한 차세대 암호체계 개발)

† 주저자, sky1236@nsr.re.kr

‡ 교신저자, sky1236@nsr.re.kr(Corresponding author)

I. 서 론

1.1 행정전자서명

행정전자서명의 법률적 의미는 '행정기관 등에서 전자문서를 작성한 후 작성자의 신원과 문서의 변경 여부를 확인할 수 있도록 그 문서에 고유하게 생산한 정보'이다[1]. 그리고 법률적 의미의 행정전자서명을 생성하는데 필요한 기술, 체계, 장치 등을 통칭하여 행정전자서명 또는 GPKI(Government Public-Key Infrastructure)라고 부른다.

행정전자서명은 전자정부의 행정환경에서 발생할 수 있는 주요 정보의 노출, 변조, 훼손 등의 문제로부터 전자정부의 신뢰성 및 안정성을 보장하기 위해 구축되었다[2]. 행정전자서명의 관리와 보급은 행정안전부(행안부)가 담당한다. 행안부는 산하의 '행정전자서명 인증관리센터(gpki.go.kr)'를 통해 2000년 4월부터 행정전자서명을 위한 인증업무를 수행하고 있다. 그리고 행정전자서명 인증관리센터의 운영 및 행정전자서명 관련 실무(시스템 운영 및 기술지원 등)는 한국지역정보개발원(KLID)이 수행하고 있다.

행정전자서명은 중앙정부, 지자체 등 행정기관과 그 보조/보좌기관, 그리고 행정기관과 전자문서를 유통하는 기관이나 법인 등이 사용한다. 각 주체는 업무에 필요한 행정전자서명을 위한 기능을 포함하는 정보시스템을 구축하여 사용해야 한다. 전자서명은 전자문서 작성자의 신원과 문서의 변경 여부를 확인하기 위한 정보이다. 따라서 같은 정보시스템을 사용하는 사용자뿐만 아니라 다른 정보시스템을 사용하는 사용자도 전자문서의 서명 및 검증을 수행할 수 있어야 한다. 따라서 호환성을 위해 행정전자서명이 적용된 각 정보시스템은 같은 암호알고리즘, 인증프로토콜 등을 사용하여야 한다.

행정안전부는 각 기관에서 구축하는 정보시스템이 행정전자서명을 이용한 인증기능을 원활하게 사용할 수 있도록 암호체계를 규정하고, 이를 구현하여 제작한 보안 모듈(행정전자서명 표준보안 API)을 배포하고 있다. 행정전자서명 암호체계는 정보시스템에 필요한 암호알고리즘과 인증서 체계, 관리 및 서비스에 관한 기술적 내용을 규정한 것이다. 현재 각 기관은 정보시스템 구축 시 적절한 절차를 통해 행정전자서명 표준보안 API 및 이의 적용에 관한 기술지원을 받고 있다. 따라서 행정전자서명 암호체계의 안전성은 이를 기반으로 구축되는 정보시스템의 안전성에 직접적인 영향을 준다.

2000년에 구축된 정부전자관인 인증시스템은 행정전자서명 암호체계 및 표준보안 API의 시초이며, 안전성 상향을 위한 암호체계의 고도화가 2010년에 한 차례 진행되었다. 주요 고도화 사항은 SHA-1의 안전성 저하에 대비한 SHA-256 적용, 대칭키 암호알고리즘 ARIA 도입, 비대칭키 프리미티브 RSA, KCDSA의 키 길이 확장 등이었다. 고도화된 암호체계를 반영한 인증서 및 API는 2012년부터 사용되었다.

현재 사용 중인 행정전자서명 암호체계는 지난 고도화 이후 10여 년이 지난 상태이다. 암호알고리즘의 안전성은 연산자원의 고도화와 분석기술의 발전으로 인해 세월이 갈수록 저하된다. 그러므로 안전성 파라미터의 상향과 새로운 기술의 적용 등으로 암호체계의 안전성을 유지하는 것이 필요하다. 또한, 통신 및 정보보안 환경의 변화에 따라 행정전자서명에 포함된 다양한 기술도 변화를 거듭하고 있다. 따라서 지난 10여 년간 변화된 환경 및 기술을 고려하여 행정전자서명 암호체계에 사용된 기술의 안전성과 관련 표준의 적절성을 검토해 볼 시점이 되었다고 할 수 있다.

1.2 행정전자서명 암호체계

행정전자서명은 전자문서 송, 수신과정에 필요한 행정기관 및 공무원 신원확인과 위변조 방지 기능을 제공하여 행정 전자문서의 안전한 유통을 돕는다.

행정전자서명 암호체계에 관한 세부적인 사항은 행안부가 관리하는 "행정전자서명 프로파일 및 알고리즘 상세서(이하 상세서)"에 정리되어 있다[3]. 상세서는 '인증서 프로파일 규격', 'OID 및 DN 체계 규격', '인증서 체계 및 저장 규격', '인증서 저장소 규격', '알고리즘 규격', '인증서 관리 규격', '인증서 서비스 규격', '무선 인증서 규격' 등의 내용으로 구성되어 있으며, 이러한 규격을 정의하기 위해 국내·외의 다양한 표준을 준용하고 있다.

1.2.1 행정전자서명 준용 표준 목록

상세서는 행정전자서명 암호체계가 준용하는 표준을 알고리즘, 데이터 형식 및 프로토콜, 무선단말기와 보안토큰 분야로 나누어 그 목록을 수록하였다. 분야별 표준 목록은 다음 각 Table 1., Table 2.,

Table 3.과 같다.

• 알고리즘 표준(기술 규격 포함)

Table 1. Standards on Cryptographic Algorithms for GPKI

Encryption	- SEED: TTAS.KO-12.0004/R1, 128-bit Block Cipher SEED - ARIA:KS X 1213-1, 128bit block encryption algorithm ARIA - KCAC.TS.ENC, Encryption Algorithm Scheme Specification [v1.21]
Digital Signature	- KCDSA: TTAS.KO-12.0001/R2, Digital Signature Mechanism with Appendix - RSA: PKCS #1 v2.2, RSA Cryptography Standard - ECDSA: ANSI,X9.62, Elliptic Curve Digital Signature Algorithm - KCAC.TS.DSIG, Digital Signature Algorithm Specification [v1.30]
Hash Function	- HAS-160: TTAS.KO-12.0011/R2, Hash Function Standard - Part 2 : Hash Function Algorithm Standard(HAS-160) - SHA-1: FIPS 180-4, Secure Hash Standard - SHA256: FIPS 180-4, Secure Hash Standard - KCAC.TS.HASH, Hash Algorithm Specification [v1.20]
Random Generation	- FIPS 186-2, General Purpose RNG - ANSI X9.62 RNG
HMAC	- RFC 2104, HMAC: Keyed hashing for Message Authentication

• 데이터 형식 및 프로토콜 표준

Table 2. Standards on Data Format and Protocols for GPKI

Entity Authentication	- ISO/IEC 9798-3, Entity authentication - Part 3: Mechanisms using digital signature techniques
ASN & DER Encoding	- ITU-T X.680, Information Technology Abstract Syntax Notation One (ASN.1): Specification of basic notation - ITU-T X.690, Information Technology ASN.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)
Certificate and Certificate Revocation List Profile	- RFC 3280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile - RFC 5280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile - ITU-T X.509, Information Technology Open Systems Interconnection The Directory: Authentication Framework (ISO/IEC 9594-8) - KCAC.TS.CERTPROF, Digital Signature Certificate Profile [v1.70] - KCAC.TS.CRLPROF, Accredited Digital Signature Certificate Revocation List Profile [v.1.50]
OCSP	- RFC 5019, Internet X.509 Public Key Infrastructure Online Certificate Status Protocol - OCSP
Certificate Management	- RFC 4210, Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)
CRMF	- RFC 4211, Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF) - KCAC.TS.CRMF, Accredited Certificate Request Message Format Specification [v1.21]
CMS	- PKCS #7 v1.5, Cryptographic Message Syntax Standard
Base64 Encoding	- RFC 2045, Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies
Storing key	- Private Key Information Syntax: PKCS #8, Private-key Information Syntax Standard - Password-based Encryption: PKCS #5 v2.1, Password-based Cryptography Standard - PKCS #11 v2.2, Cryptographic Token Interface Standard - KCAC.TS.UI, User Interface Specification for the Interoperability between Accredited Certification Authorities [v1.90]
Directory System	- LDAP v3, Lightweight Directory Access Protocol (v3): Technical Specification - KCAC.TS.LDAP, Lightweight Directory Access Protocol Specification [v1.11]

Subscriber Identification	- KCAC.TS.SIVID, Subscriber Identification Based on Virtual ID [v1.21]
Time Stamp	- RFC 3161, Internet X.509 Public Key Infrastructure Time-Stamp Protocol - KCAC.TS.TSP, Time-Stamp Protocol Specification [v1.11]

• 무선단말기와 보안토큰 인증

Table 3. Standards on Wireless Devices and Secure Token Authentication for GPKI

Storage Format and Usage Specification	- KCAC.TS.CM, Certificate Management in Mobile Device [v1.30] - KCAC.TS.UI, User Interface Specification for the Interoperability between Accredited Certification Authorities [v1.90] - KCAC.TS.HSMU, Accredited Certificate Usage Specification for Hardware Security Module [v2.10] - KCAC.TS.HSMS, HSM Storage Format Specification for Accredited Certificate [v1.11]
Transmission Format	- KCAC.TS.CT, Certificate Transmission between PC to Mobile Device [v2.10]

목록의 KCAC 문서들은 (구)공인전자서명인증체계(NPKI, 구 공인인증서) 기술 규격을 규정하고 있다. 행정전자서명은 KCAC를 다수 준용하고 있으므로 논문에서는 KCAC 문서도 함께 검토하였다.

1.3 안전성 수준

국내외 여러 암호 관련 기관은 안전한 암호 사용을 위하여 연도별 적정 안전성 수준(security level)을 권고하고 있다. 행정전자서명 고도화가 예상되는 2020년대 후반부터 2030년 사이의 기관별 권고 내용을 정리하면 Table 4.와 같다.

우리나라 KISA를 비롯한 대부분 기관은 2030년 이전에 128비트 안전성을 제공하는 암호체계로 전환하는 것을 권고하고 있다. BSI는 2023년부터 120비트 이상의 안전성 수준을 가지는 암호체계 사용을 권고하고 있다. 단, BSI는 128비트 이상의 안전성을 제공할 수 있는 파라미터만 권고하고 있다. 현재

Table 4. Recommendations on Security Levels

Organization (Country)	Security Level	Date	ref.
NIST(US)	128	2030~	[4]
ANSSI(FR)	128	2026~	[5]
ECRYPT-CSA(EU)	128	2018~2028	[6]
BSI(DE)	120(128)	2023~	[7]
KISA(KR)	128	2030~	[8]

행정전자서명 암호체계는 112비트 안전성을 기준으로 설계되어 있으므로, 이른 시일 안에 128비트 안전성을 제공하도록 고도화할 필요가 있다.

II. 행정전자서명 암호체계 검토 방법

2.1 검토의 목적과 방향

이 논문은 상세서에 수록된 기술 규격 및 이를 위해 참고한 표준을 검토한 내용을 제시한다. 행정전자서명 암호체계 안전성 검토의 목적은 준용 표준 현행화, 안전성 상향을 위한 파라미터 조정, 노후 및 취약 기술 식별 등이다. 단, 암호체계가 준용하는 기술의 종류와 표준의 형태에 따라 검토의 목적과 방향을 다르게 설정하였다.

상세서에 수록된 모든 기술을 기술의 종류와 표준의 형태에 따라 암호알고리즘과 인증서 체계, 인증서 관리 및 서비스 분야로 분류하고 분야별로 검토 방향을 다음과 같이 설정하였다.

- 암호알고리즘
  - 준용 암호알고리즘 표준 현황 분석
  - 준용 암호알고리즘 최신 안전성 분석 결과를 기반으로 사용 제한 또는 안전성 파라미터 조정 필요성 검토
  - 신규 암호알고리즘 추가 검토
- 인증서 체계, 인증서 관리 및 서비스
  - 준용 표준 현황 검토
  - 해당 표준과 암호 안전성 연관성 검토
  - 신규 암호체계 적용 시 고려사항 검토

2.2 검토의 대상과 방법

행정전자서명 암호체계 관련 표준의 현재 상태를 확인하기 위하여 먼저 상세서가 명시하고 있는 모든

표준문서 및 관련 내용을 검토하였다. 상세서는 1장에 나열한 목록 이외에도 본문에서 별도로 관련 표준들을 명시하고 있다. 또 일부 국내 표준은 다른 국제 표준을 준용하기도 한다. 그래서 모든 기술의 현황을 파악하기 위해서는 관련된 모든 표준을 수집·분석해야 한다.

분석 대상 표준문서는 대략 75편이었다. 국제 표준은 주로 IETF, ISO/IEC, ITU-T가 발간한 문서였다. 국내 표준은 TTA, KS 문서가 있으며, 기술규격으로는 공동인증서가 준용하는 KCAC.TS 문서가 있다.

먼저, 각 표준의 대체(obsolete)·갱신(update) 이력을 통해 현재도 효력이 있는지 확인하였다. 표준의 유효성 확인을 통해 세부 검토가 필요한 표준을 선별하고, 이후 해당 표준에 안전성과 관련한 내용이 있는지를 검토하였다. 암호체계의 안전성과 관련한 표준은 암호알고리즘 규격 등 명세를 다루는 표준과 프로토콜 등 암호알고리즘의 사용에 관한 표준으로 나눌 수 있다.

암호알고리즘의 명세를 다루는 표준은 일반적으로 신규 암호알고리즘의 추가 등으로 인한 갱신이 대부분이므로, 표준의 현황보다는 암호알고리즘의 안전성 분석 현황을 파악하여 암호알고리즘의 현재 안전성을 판단할 필요가 있다. 따라서 암호알고리즘은 최신 공격 결과와 안전성 분석현황 조사를 통해 안전성을 파악하고, 128비트 안전성을 제공할 수 있도록 고도화하는 것을 기준으로 검토하였다. 또한, 양자 컴퓨터를 이용한 공격을 고려하여 알고리즘 및 파라미터의 변경, 추가 등이 필요한지 검토하였다.

암호알고리즘을 사용하는 기술의 표준은 최신 표준 현황 조사를 통해 상세서가 참고하는 표준과 비교하고, 128비트 안전성을 가지도록 상향하는 것을 기준으로 파라미터 조정 등이 필요한지 검토하였다. 암호알고리즘을 사용하는 표준도 양자 컴퓨터를 이용한 공격을 고려한 검토를 수행하였다.

안전성과 관련이 없지만 폐지·갱신 이력이 있는 표준들은 상세서가 참고하는 표준과 비교하고, 참고표준을 최신 표준으로 교체할 필요가 있는지를 검토하였다.

III. 암호알고리즘 검토내용

이 장에서는 행정전자서명 암호체계에서 현재 사용 중인 암호알고리즘의 안전성과 표준 현황을 검토하여 해당 알고리즘이 충분한 안전성을 제공하는지, 새로운 알고리즘의 도입이 필요한지를 분석한다.

3.1 비대칭키 암호

3.1.1 전자서명

행정전자서명은 전자서명 알고리즘으로 KCDSA, RSA, ECDSA를 사용한다. 각 알고리즘에 관한 검토내용은 다음과 같다.

- KCDSA

KCDSA는 이산 대수 문제(DLP, Discrete Logarithm Problem)의 어려움을 이용하여 설계한 ElGamal 서명 방식의 변형 알고리즘으로, 암호모듈검증제도(KCMVP) 검증대상 암호알고리즘이다. KCDSA는 1998년에 한국정보통신기술협회(TTA) 표준으로 제정되었으며, ISO/IEC 14888-3에 수록된 국제표준 알고리즘이다. 상세서는 TTAK.KO-12.0001/R2를 인용하고 있는데, 이 표준의 최신 버전은 2016년 개정된 TTAK.KO-12.0001/R4이다. 행정전자서명 암호체계에서 인증 서비스 제공자와 사용자(무선 제외)는 공개키와 개인키가 각각 2,048비트와 256비트(112비트 안전성 제공)인 KCDSA를 사용한다. KCDSA에 대한 최신 표준은 128비트 안전성을 제공하는 파라미터(공개키 3,072비트, 개인키 256비트)를 제시하고 있다. 따라서 행정전자서명의 새로운 암호체계에서는 TTAK.KO-12.0001/R4를 참고하여 128비트 안전성을 제공하는 파라미터를 사용하는 것을 고려할 수 있다.

- RSA

행정전자서명 암호체계에서 인용하는 PKCS #1은 RSASSA-PSS와 RSASSA-PKCS1-v1_5 두 가지 전자서명 알고리즘을 명시한다. RSASSA-PSS(Probabilistic Signature Scheme)는 랜덤화 요소(salt)를 사용하는 EMSA-PSS 인코딩이 적용된 알고리즘이고, RSASSA-PKCS1-v1_5는 PKCS #1 v1.5에서 제시한 인코딩이 적용된 알고리즘이다. RSASSA-PSS는 RSA 기반의 다른 서명과 달리 랜덤화 요소인 salt를 사용하도록 하여 더 tight한 안전성 증명이 가능한 장점이 있다. 이런 이유로 PKCS #1의 현행 표준인 RFC 8017은 신규 응용을 설계할 때 RSASSA-PSS를 사용하고, RSASS

A-PKCS1-v1_5는 하위 호환을 위해서만 사용하는 것을 요구하고 있다. 더불어 KCMVP 검증대상 암호알고리즘 목록에도 RSASSA-PSS만 포함되어 있다.

행정전자서명 암호체계는 RSA 전자서명을 최상위 인증기관, 행정 인증기관 및 공공/민간 인증기관의 유선 인증과 사용자 웹서비스용 전자서명에 사용한다고 하면서도, PSS와 PKCS1-v1_5 중 어느 알고리즘을 사용해야 하는지 명시하지 않고 있다. 따라서 행정전자서명의 새로운 암호체계에서는 RSASSA-PSS 알고리즘의 사용을 고려하고 RSASSA-PSS의 사용을 상세서에 명시할 필요가 있다.

행정전자서명은 현재 키가 2,048비트인 RSA를 사용한다. 2,048비트 RSA는 대략 112비트 안전성을 제공하는 것으로 알려져 있다. 따라서 행정전자서명의 새로운 암호체계가 128비트 안전성을 제공하기 위해서는 RSA의 키 길이를 3,072비트 이상으로 상향할 필요가 있다.

- ECDSA

ECDSA는 타원곡선을 이용하는 전자서명 알고리즘으로 KCMVP 검증대상 암호알고리즘이다. 상세서는 ECDSA의 참고 표준으로 ANSI X9.62와 FIPS 186-3을 제시하고 있다. FIPS 186-3은 2013년에 FIPS 186-4로 대체되었고 현재 초안 상태인 FIPS 186-5로 개정될 예정이다. 상세서가 준용하고 있는 FIPS 186-3은 소수체 상의 곡선, 이진체 상의 곡선, Koblitz 곡선을 규정하고, 곡선별로 최대 112비트 안전성을 제공할 수 있는 권고 파라미터를 제시하였다. 현재 개정 중인 FIPS 186-5는 타원곡선 파라미터를 제공하지 않으며, 권고 타원곡선 파라미터만 제시하는 NIST SP 800-186이 제정될 예정이다. NIST SP 800-186은 128비트 이상의 안전성을 제공하는 곡선도 포함하므로, 최신 표준을 참고한 타원곡선의 선택을 고려할 수 있다.

상세서의 인증서 체계에 따르면 ECDSA는 무선 인증서에만 사용되는 것으로 파악된다. 이 무선 인증서는 WML이라는 XML 기반의 언어로 작성된 웹 페이지만을 접속할 수 있는 환경에서 사용하는 인증서를 말한다. 이는 저사양의 이동통신 장비에 웹 브라우저와 같은 서비스를 제공하기 위해 설계되었으며, 우리나라의 경우 해당 서비스는 2016년도 4분기를 기점으로 모두 종료되었다. 그러므로 실질적으로 행정전자서명은 현재 ECDSA를 사용하지 않는다.

3.1.2 공개키 암호화

현재 행정전자서명 암호체계에서는 공개키 암호화로 RSA만 사용한다. RSA 암호화에 관한 검토내용은 다음과 같다.

- RSAES

RSA를 단독으로 사용하여 데이터를 암호화하면, 선택 암호문 공격(chosen ciphertext attack)에 취약하다. RSAES(RSA Encryption Scheme)는 이러한 문제를 해결하고자 개발된 RSA 기반 암호화 알고리즘이다.

행정전자서명은 현재 키 길이가 2,048비트인 RSAES-PKCS1-v1_5를 사용하는데, 새로운 암호체계에서는 다음과 같은 이유로 키 길이가 3,072비트 이상인 RSAES-OAEP의 사용을 고려할 필요가 있다. 첫째, RSA 2,048비트의 키는 약 112비트의 안전성을 제공한다고 알려져 있으므로, 2,048비트의 키를 사용하면 128비트 안전성을 제공할 수 없다. 둘째, 인코딩 기법 PKCS1-v1_5에 대한 다수의 공격이 존재하며, 행정전자서명 암호체계가 준용하는 RFC 8017도 해당 인코딩 기법을 규정하고는 있지만 하위 호환을 위해서만 사용하는 것으로 제한하고 있다. 대신 RFC 8017은 인코딩 기법으로 OAEP를 사용하여야 하는 것으로 규정하고 있다.

3.2 대칭키 암호

3.2.1 블록암호

행정전자서명 암호체계에 포함된 블록암호는 TDEA, SEED, ARIA이다. 블록암호의 안전성은 차분 특성, 선형 특성, 대수적 성질 등을 기반으로 한 다양한 공격을 이용하여 분석된다. 각 알고리즘에 관한 분석내용은 다음과 같다.

- TDEA

DES(Data Encryption Standard)는 1976년에 IBM이 개발한 Lucifer 알고리즘을 1977년에 NBS(National Bureau of Standards, 현 NIST, National Institute of Standards and Technology)가 FIPS 46으로 표준화 하여 발표한 블록암호이다[9]. DES는 2004년에 미연방 표준에서 제외될 때까지 27년간 미국의 표준암호로 사용되었

다. DES의 블록 길이는 64비트이고, 키 길이는 56비트이다.

DES에 대해서는 이론적 안전성 분석 연구와 함께 키 전수 조사를 위한 시도도 꾸준히 진행되었다. 1997년에는 DESCHALL Project를 통해 96일 만에 키를 찾을 수 있음이 알려졌고, 1998년에는 Distributed.net에서 41일 만에, 1998년에는 EFF(Electronic Frontier Foundation)에서 DES 해독 전용 장비인 Deep Crack을 이용하여 25만 달러의 비용으로 56시간에 키를 찾았다. 1999년에는 Distributed.net과 Deep Crack을 동시에 이용하여 22시간 15분 만에 키를 찾기에 이르렀다.

DES의 키 길이 문제를 보완하기 위해 Triple DES(TDEA)가 개발되었으며(1981년), 이후 RFC 1851로 제정되었다(1995년). TDEA는 DES를 세 번 연속 적용하여 키 길이를 늘이는 방식으로 두 개 또는 세 개의 키를 사용하며(각, 2TDEA, 3TDEA), 112비트 안전성을 제공하도록 개발되었다.

2016년에 TLS나 OpenVPN에서 블록길이가 64비트인 암호알고리즘을 사용할 경우, 인증 토큰을 탈취할 수 있는 공격이 발표되었다[10]. 이 공격은 암호화에 n비트 블록암호를 특정 블록암호 운영 모드와 함께 사용할 때, $2^{n/2}$개 블록의 암호문 내에서 충돌 쌍을 찾을 수 있다는 생일 공격을 이용한다. 128비트 블록암호에 해당 공격을 적용하면 공격에 필요한 데이터가 256EB(Exabyte)로 실현 불가능하지만, 64비트 블록암호인 경우 약 32GB로 실현 가능하다. 이에 따라 NIST SP 800-67은 TDEA를 이용할 때 같은 키로 암호화할 수 있는 블록의 최대 개수를 2^{20}개로 제한하고 있다(특히, 2TDEA는 하위 호환성만을 위해 사용해야 한다). NIST는 현재 TDEA의 사용을 자제하는 것을 권고하고 있으며, 2024년부터는 하위 호환이 필요한 경우를 제외하고는 TDEA를 사용하지 않아야 한다고 명시하고 있다. 실제로 TLS, Windows 등 상당수의 활용처에서 TDEA의 사용을 이미 중단하였다.

그러므로 행정전자서명의 새로운 암호체계에서는 TDEA를 하위 호환을 위해서만 사용하도록 제한하는 것이 바람직하다고 판단된다.

- SEED

SEED는 1999년에 우리나라가 개발한 블록암호이다. SEED는 국내 표준(TTAS.OK-12.0004/R1)이자 국제 표준(ISO/IEC 18033-3, RFC 4269)이다. SEED도 ARIA와 마찬가지로 KCMVP 검증대상 암호알고리즘이다.

SEED는 Feistel 구조를 사용한다. SEED는 128비트 블록과 128비트 키를 가지며, 총 16라운드로 구성되어 있다.

SEED의 라운드 함수 F는 수정된 Feistel 구조이며, 모듈러 덧셈(modular addition)을 사용한다. F에 사용되는 내부 함수 G는 SP(Substitution-Permutation) 구조이다. 즉, 라운드 함수 F는 비선형 요소로 S-box와 모듈러 덧셈을 사용한다.

SEED에 대한 최신 안전성 분석 결과로는 2014년에 발표된 7라운드 차분 특성을 이용한 9라운드 차분 공격이 있다[11]. 그리고 현재까지 전체 라운드의 SEED에 대해 알려진 공격은 없다. 그러나 SEED는 개발된 지 20여 년이 지났고, 국제 표준으로 제정되었음에도 불구하고 다른 암호알고리즘에 비해 안전성 분석 결과가 적은 편이다. 더구나 SEED는 128비트 키만 지원한다. 그러므로 후술할 블록암호의 양자 컴퓨터 공격에 대한 안전성 분석에 따르면 SEED는 64비트 이상의 안전성은 제공하지 못할 것으로 예상한다.

- ARIA

ARIA(Academy, Research Institute, and Agency)는 2003년에 우리나라 학계, 연구소, 정부기관이 공동으로 개발한 블록암호이다[12]. ARIA는 국내 표준(KS X 1213-1)이자 국제 표준(RFC 5794)이다. 또한, KCMVP 검증대상 암호알고리즘이다.

ARIA는 SPN(Substitution-Permutation Network) 구조를 사용하였고, 암, 복호화 라운드 함수가 서로 같은 involution 형태로 설계되었다. ARIA는 블록길이가 128비트이고 키 길이가 각각 128, 192, 256비트인 ARIA-128, ARIA-192, ARIA-256으로 구성되어 있다. 이들의 라운드 수는 각각 12, 14, 16이다.

ARIA는 발표된 지 15년이 넘은 블록암호로 설계자들의 분석 결과의 개선뿐 아니라 알고리즘 발표 이후 고안된 신규 공격의 적용 등 다양한 연구가 진행되었다. ARIA에 대한 안전성 분석 결과를 정리하면 Table 5.와 같다.

Table 5.의 왼쪽 열은 ARIA에 적용한 공격 방법이며, 오른쪽 열은 각 공격을 ARIA-128, ARIA-192, ARIA-256에 적용하여 공격할 수 있는 최대

Table 5. Cryptanalysis results for ARIA

Cryptanalysis	Attacked round (128/192/256)
Differential	7/7/7
Truncated differential	7/7/7
Impossible differential	6/6/7
Higher order differential	2/2/2
Boomerang/Rectangle	6/6/7
Linear	8/9/11
Differential-Linear	6/6/7
Integral	6/6/7
Interpolation	2/2/2
Biclique	-/-/16

라운드를 의미한다. ARIA-128, ARIA-192, ARIA-256에 대해 각각 8, 9, 16라운드 공격이 지금까지 알려진 가장 긴 라운드의 공격이다. ARIA-256의 경우 전체라운드인 16라운드에 대한 바이클릭 공격이 존재하는데, 공격 복잡도가 $2^{255.2}$이다. 이는 키 전수 조사 연산량의 평균인 2^{255}보다 큰 연산량을 요구하므로, 바이클릭 공격이 ARIA-256에 유의미한 공격이라고 보기 어렵다. 따라서 바이클릭 공격을 제외하면, ARIA-256에 대한 가장 긴 라운드의 공격은 11라운드의 선형공격이다. 이를 토대로 계산하면 ARIA-128, ARIA-192, ARIA-256의 안전성 마진(security margin)은 각각 4, 5, 5라운드이며 전체라운드의 30% 이상이 되므로 위협이 될만한 공격은 없다고 볼 수 있다.

현재 행정전자서명 암호체계에서는 블록암호로 ARIA-128만 사용하고 있다. 행정전자서명의 새로운 암호체계에서는 더 높은 수준의 안전성을 제공할 수 있도록 ARIA-192, ARIA-256의 사용도 고려할 필요가 있다.

3.2.2 메시지 인증 코드(MAC)

행정전자서명 암호체계는 메시지 인증 코드(MAC, Message Authentication Code)로 해시함수를 사용하는 HMAC 알고리즘만 사용하고 있다. HMAC은 1996년에 개발되었지만, 현재까지 안전성에 문제가 발견되지 않았으며, HMAC을 규정하는 RFC 2104도 1997년에 발간된 이후 현재까지 대체되지 않고 있다. 다만, HMAC의 기반 해시함수로 MD5를 사용하는 것을 제한하기 위한 RFC 6151이 2011년에 발간되었다.

2012년 이후에 HMAC의 안전성에 관해 발표된 결과들은 HAIFA, Whirlpool 등 특정 해시함수를 사용하였을 때의 공격 및 해시함수의 내부 상태 크기에 따른 위조 공격 복잡도 개선 등이 있다[13][14]. 하지만 이러한 공격은 HMAC의 기반 해시함수가 안전하여야 한다는 가정을 만족하지 않는 경우에만 적용할 수 있다. 따라서 안전한 해시함수를 이용하면 HMAC의 안전성에 문제가 없으므로, 행정전자서명에서 HMAC을 사용하는 것은 문제가 없을 것으로 판단된다. 다만, 해시함수의 선택에 있어 일부 고려하여야 할 부분은 있으며, 다양성을 위해 블록암호 등 다른 프리미티브를 이용한 메시지 인증 코드의 추가를 고려할 수 있다. KCMVP 검증대상 암호알고리즘 목록에는 HMAC과 함께 블록암호 기반 MAC인 CMAC과 GMAC이 포함되어 있다.

3.3 해시함수 및 난수 생성

3.3.1 해시함수

행정전자서명은 현재 해시함수 HAS-160, SHA-1, SHA-256을 사용한다. 해시함수에 대한 안전성 분석은 충돌쌍 공격, (제2)역상 공격 등을 통해 이루어진다. 각 알고리즘에 관한 검토 내용은 다음과 같다.

- HAS-160

HAS-160은 국내 표준 해시함수 알고리즘이다(TTAS.KO-12.0011/R2). HAS-160은 160비트 해시값을 출력하며, 총 80 step으로 구성되어 있다.

HAS-160에 대한 충돌쌍 공격은 2005년과 2006년에 각각 45 step과 53 step으로 축소된 압축함수에 적용되었고[15][16], 2007년에는 53 step에 대한 공격의 복잡도를 대폭 줄이는 결과와 함께 59 step에 대한 공격이 발표되었다[17]. 이후 2011년에는 semi-free-start 조건에서 65 step HAS-160의 충돌쌍을 일반적인 PC로 한 시간 정도에 찾을 수 있는 공격이 발표되었다[18].

역상 공격은 2008년에 52 step으로 축소된 압축함수에 대한 공격이 있었으며[19], 최신 연구 결과로는 2018년에 발표된 71 step에 대한 공격이 있다[20].

HAS-160은 160비트 해시값을 출력하므로, 충돌

쌍 공격에 대해 최대 80비트의 안전성만 제공할 수 있다. 더욱이 안전성 마진도 9 step으로, 전체가 80 step인 것을 고려하면 충분하지 않은 것으로 판단된다.

따라서 행정전자서명의 새로운 암호체계에서는 HAS-160을 하위 호환을 위해서만 사용하도록 제한하는 것이 바람직하다고 판단된다.

- SHA-1

SHA-1은 1995년에 개발되어 현재까지도 NIST 표준(FIPS 180-4)에 포함된 해시함수 알고리즘이다. SHA-1은 160비트 해시값을 출력하며, 총 80 step으로 구성되어 있다.

2005년 2월에 SHA-1에 대해 충돌쌍 공격이 가능하다는 것이 발표되었다[21]. 이 공격의 복잡도는 2^{69}이며, 발표 당시에는 실제 충돌쌍을 제시하지는 못하였다.

2013년에는 2^{61}의 연산량으로 SHA-1의 충돌쌍을 찾는 공격이 발표되었는데[22], 이 공격은 연산량이 $2^{57.5}$인 freestart 충돌쌍 공격으로 이어졌다[23]. 이는 2017년에 발표된 Shattered로 발전하였다[24]. Shattered는 대규모의 GPU를 이용하여 최초로 SHA-1의 충돌쌍을 찾고(공격 복잡도 $2^{63.1}$), 찾은 충돌쌍을 이용하여 pdf 문서를 위조할 수 있음을 입증하였다.

2019년에는 chosen-prefix 충돌쌍 공격을 2^{70} 이하의 연산량으로 개선한 결과가 발표되었다[25]. Chosen-prefix 충돌쌍 공격은 공격자가 미리 정한 메시지를 포함하는 충돌쌍을 생성할 수 있어서 전자서명이나 TLS, SSH 등의 보안프로토콜을 직접적으로 위협할 수 있다. 2020년에는 이 chosen-prefix 충돌쌍 공격을 실제로 구현하여 PGP/GnuPG에서 위장 공격(impersonation attack)을 수행한 결과가 제시되었다[26]. 이로써 SHA-1은 실질적으로 안전하지 않은 알고리즘이 되었다.

NIST는 2012년 말부터 전자서명에 SHA-1을 사용하는 것을 허용하지 않고 있으며, 2015년 개정된 SP 800-131A에 해당 내용을 명시하였다. 상세서도 HAS-160과 함께 SHA-1을 전자서명용으로 사용하지 않도록 규정하고 있다.

단, 전자서명 외에 메시지 인증 코드, 키 유도 함수, 난수 생성 등 전자서명을 제외한 알고리즘에 사용되는 기반 해시함수는 충돌쌍 공격에 대한 안전성이 필요하지는 않아, NIST와 상세서 모두 전자서명을 제외한 응용에서는 SHA-1의 사용을 허용하고 있다. 하지만 결과적으로는 설계 당시 수립한 안전성 목표를 만족하지 못하므로 행정전자서명의 새로운 암호체계에 포함하지 않는 것을 고려해 볼 수 있다.

- SHA-256

SHA-256은 2001년에 NSA가 개발하여 2002년에 NIST FIPS 180-2로 제정된 해시함수로 KCMVP 검증대상 암호알고리즘이다. SHA-256은 SHA-2에 포함된 6개의 해시함수 중 하나로, 해시값이 256비트이며 총 64 step으로 구성되어 있다.

SHA-2는 개발 이후 안전성에 관한 연구가 매우 많이 수행되었는데, 아직 전체 step의 SHA-256에 대한 공격결과는 없다. 현재 가장 긴 step을 공격한 사례는 2012년에 발표된 52 step의 의사-역상(pseudo-preimage) 공격이다[27]. 바이클릭 특성을 이용하는 이 공격은 계산복잡도가 2^{255}으로 generic 공격에 근접하며, 역상 공격으로 변환하면 2^{256}을 넘어선다. 2009년과 2010년에는 42 step에 대하여 복잡도가 $2^{251.7}$과 $2^{248.4}$인 meet-in-the-middle 방식의 역상 공격이 발표되었다[28][29].

SHA-256의 충돌쌍 공격은 2008년에 24 step에 대한 공격[30]이 처음 제시된 이후, 2013년에 31 step까지 공격 되었으나, 그 공격 복잡도가 $2^{65.5}$이다[31]. 현실적인 공격량으로 가능한 수준은 2015년에 발표된 28 step의 공격이다[32].

현재 SHA-256은 128비트 안전성을 제공하며, 안전성 마진도 충분하다. 하지만 행정전자서명에 사용된 기존 해시함수 중 충분한 안전성을 제공하는 것은 SHA-256뿐이므로, 신규 해시함수의 추가를 고려해 볼 수 있다.

3.3.2 난수 생성 알고리즘

난수 생성 알고리즘을 사용하기 위해 상세서가 참고하고 있는 표준은 FIPS PUB 186-2와 ANSI X9.62이다. 이 두 표준은 전자서명 알고리즘 표준이지만, 부록에서 난수 생성 알고리즘을 규정하고 있다. 두 표준에서 규정하고 있는 난수 생성 알고리즘은 SHA-1과 DES를 사용한다. SHA-1과 DES는 더 이상 안전하지 않다고 여겨지는 알고리즘이므로, 이를 이용한 난수 생성 알고리즘의 사용도 자제해야 한다.

난수 생성 알고리즘의 국내 표준으로는 TTAK.KO-12.0189, TTAK.KO-12.0331, TTAK.KO-1

2.0332가 있다. 이 표준들은 각각 블록암호, 해시함수, HMAC 기반 결정론적 난수발생기를 규정하는 NIST SP 800-90A를 참고하여 만들어진 표준이다, 국제 표준 ISO/IEC 18031은 국내 표준에 포함된 난수 생성 알고리즘을 모두 규정하고 있다.

난수 생성은 일반적으로 하위 호환성이 필요하지 않으므로 신규 행정전자서명 암호체계에서는 기존 난수 생성 알고리즘의 사용은 자제하고, 기반 프리미티브의 다양성을 위해 2종 이상의 국내·외 표준 난수 생성 알고리즘의 도입을 고려할 필요가 있다.

3.4 양자 컴퓨터 위협과 양자내성암호

암호학계에서는 양자 컴퓨터의 개발에 대비하여 암호알고리즘의 양자 안전성에 관한 연구가 활발히 진행되고 있다. 현재까지 알려진 바로는 양자 컴퓨터를 이용한 공격에 대칭키 암호알고리즘보다 비대칭키 암호알고리즘이 더 취약하다. 이런 이유로 기존 암호알고리즘의 양자 안전성과 함께 양자내성을 가지는 비대칭키 암호알고리즘 개발에 관한 연구가 활발히 진행되고 있다.

3.4.1 전자서명 및 공개키 암호화

소인수분해 문제와 이산 대수 문제를 다항식 시간 내에 해결할 수 있다면 기존 행정전자서명 암호체계의 전자서명 알고리즘(KCDSA, RSA, ECDSA)과 공개키 암호화 알고리즘(RSA)은 모두 안전하지 않다. 이는 소인수분해 문제나 이산 대수 문제를 현존 컴퓨터를 이용하여 다항식 시간 내에 해결하는 알고리즘은 알려지지 않았지만, 양자 컴퓨터를 이용하여 다항식 시간 내에 해결하는 알고리즘은 알려져 있기 때문이다[33]. 따라서 충분한 크기의 양자 컴퓨터가 개발되면 현재 행정전자서명 암호체계에서 사용하는 전자서명 알고리즘과 공개키 암호화 알고리즘의 안전성을 보장할 수 없다. 그러므로 신규 행정전자서명 암호체계는 양자 컴퓨팅 시대에도 안전한 양자내성 전자서명 및 공개키 암호화 알고리즘의 도입을 준비할 필요가 있다.

3.4.2 블록암호

양자 컴퓨터를 이용한 블록암호 공격 방법으로는 Grover 알고리즘을 이용한 키 전수조사 방법이 대표적이다[34]. 일반적으로 n비트 키를 가지는 블록암호는 Grover 알고리즘에 대해 n/2비트 안전성을 제공한다고 알려져 있다. 따라서 양자 컴퓨팅 환경에서 블록암호가 128비트의 안전성을 제공하기 위해서는 256비트 이상의 키 사용을 고려할 필요가 있다.

3.4.3 해시함수

해시함수에도 Grover 알고리즘을 적용할 수 있다. 하지만 메모리 사용량 등을 고려하면 현존 컴퓨터를 이용하는 parallel rho 방법과 공격 복잡도가 비슷할 것이라는 견해가 있다[35]. NIST 역시 양자내성암호 공모 과정에서 블록암호와 해시함수를 이용하여 안전성 수준을 정의하였는데, 블록암호 관점의 안전성 수준은 양자 컴퓨터 복잡도(quantum gate)와 기존 복잡도(classical gate)를 이용하여 명시하였지만, 해시함수 관점에서는 기존 복잡도만을 이용하여 명시하였다[36].

3.4.4 메시지 인증 코드(MAC)

다양한 메시지 인증 코드 알고리즘의 양자 안전성에 관한 연구도 활발히 진행되고 있다. 연구 결과에 따르면, HMAC은 양자 컴퓨터 공격에 안전하며[37], CBC-MAC, PMAC, GCM, OCB 등 일부 블록암호 기반 메시지 인증 코드와 인증 암호화 알고리즘은 양자 컴퓨터를 이용한 다항식 시간 공격이 존재한다[38]. 하지만, 해당 공격 모델은 공격자에게 중첩상태의 질의(superposition query)를 허용하며, 오라클로부터 중첩상태의 응답을 받을 수 있는 환경을 가정한다. 이는 공격 대상 알고리즘이 양자 컴퓨터에서 구현되어 동작하고 있다는 가정이므로 현실과는 다소 괴리가 있는 모델로 여겨지고 있다.

IV. 인증서 체계 검토내용

이 장은 행정전자서명 암호체계에 사용되는 인증서의 규격과 체계에 관련한 표준 현황 검토 결과를 제시한다. 검토 결과는 현행 표준의 준용 필요성, 국내 기술 규격(KCAC)의 개정 필요성 등을 포함한다.

4.1 프로파일 규격

4.1.1 인증서 프로파일

상세서는 인증서 프로파일을 정의하기 위해 국제 표준 3편(RFC 3280, RFC 5280, ITU-T X.509)과 기술규격 1편(KCAC.TS.CERTPROF)을 참고한다.

- 국제 표준

RFC 3280은 RFC 5280에 의해 대체된 노후 표준이며, RFC 5280도 이후 발표된 표준으로 일부 갱신된 상태이다. 또한, 상세서는 1997년에 발간된 ITU-T X.509를 참고하고 있는데, 2019년에 갱신 표준이 발간되었다.

- KCAC.TS.CERTPROF

KCAC.TS.CERTPROF는 RFC 3280과 1997년에 발간된 ITU-T X.509를 참고하고 있으므로 현행화가 필요하다. 또한, 인증서 확장 필드에서 충분한 안전성을 제공하지 못하는 SHA-1을 사용하도록 명시한 부분도 개정이 필요하다.

4.1.2 인증서 폐지/신뢰 목록 프로파일

인증서 폐지/신뢰 목록 프로파일도 인증서 프로파일과 마찬가지로 RFC 3280과 RFC 5280을 참고하고 있으며, 국내 기술규격으로는 KCAC.TS.CRLPROF와 KCAC.TS.CTL를 참고한다. KCAC.TS.CRLPROF의 검토 내용은 앞선 KCAC.TS.CERTPROF와 같다. KCAC.TS.CTL도 인증서에 해시함수 SHA-1과 HAS-160의 적용을 허용하고 있다. 따라서 이러한 해시함수의 사용을 제한하는 내용을 포함하는 개정을 고려해야 한다.

4.2 OID와 DN

4.2.1 OID(Object Identifier) 체계

상세서는 행정전자서명 암호체계에서 사용하는 OID를 명시하고 있다. 대부분 행정전자서명 인증관리센터(기관 코드 100001)가 발급한 OID이며, 그 외는 KISA(200004)와 RSA(113549)가 발급한 것이다. 상세서는 OID 명시를 위해 KCAC.TG.OID v1.4와 RFC 5794를 참고한다.

KCAC.TG.OID는 v1.7이 출판되었으므로 현행화를 고려해 볼 필요가 있다. RFC 5794는 ARIA 알고리즘을 정의하는 표준이며, 여기에 ARIA 알고리즘을 사용하는 운영 모드의 OID가 정의되어 있다. 그런데, 행정전자서명 암호체계는 ARIA-128과 그 운영 모드에 대해 RFC 5794에 정의된 OID 외에도 별도의 OID를 사용하고 있다.

행정전자서명 암호체계에 ARIA 알고리즘을 도입하는 고도화 사업('행정전자서명 인증 암호체계 고도화 사업')이 2010년 3월부터 추진되었고, RFC 5794가 2010년 3월에 출판된 것으로 미루어 볼 때, 고도화 당시 RFC 문서를 참고하지 않고 자체적으로 OID를 정의하여 사용한 것으로 추정된다. 따라서 행정전자서명의 새로운 암호체계에서는 OID의 단일화가 필요하다.

4.2.2 DN(Distinguished Name) 체계

DN은 인증체계에 참여하는 각 객체와 객체의 인증서 등을 식별하기 위해 사용하는 식별 정보 체계이다. DN에 관해서는 ITU-T X.520, ISO/IEC 9594-6, RFC 2256, RFC 2253, PKCS #9 등의 국제 표준을 참고한다. 참고로 DN은 암호체계의 암호학적 안전성과 직접적인 연관성은 없다.

DN 체계를 정의하는 ISO/IEC 9594-6은 2020년에 최신 버전이 출판되었다. RFC 2256는 경량 디렉터리 접근 프로토콜(Lightweight Directory Access Protocol, 이하 LDAP)의 속성(attribute) 구조를 규정하는 표준인데, DN 체계와 관련된 내용은 RFC 4512, 4517, 4519로 대체되었다. RFC 2253은 DN의 인코딩을 규정하는 표준으로, RFC 4514로 대체되었다. PKCS #9(RFC 2985)는 속성 구조체를 정의한 현행 표준이다.

4.3 인증서 체계 및 저장

4.3.1 인증서 체계

상세서의 인증서 체계에는 행정전자서명을 사용하는 각 주체 및 인증서 종류에 따른 인증서의 유효기간과 암호알고리즘 및 키 길이가 명시되어 있다. 암호체계 고도화를 통해 새로운 암호알고리즘을 도입하거나, 안전성 파라미터를 조정하게 되면 변경된 내용

을 반영하여 인증서 체계를 정비할 필요가 있다.

4.3.2 인증서 저장 규격

상세서는 행정전자서명을 사용하는 단말기와 인증서의 종류에 따라 인증서의 저장 규격을 명시하고 있다. 특히 무선 단말기와 보안토큰의 인증서 이용에 관해서는 KCAC.TS.CM, KCAC.TS.CT, KCAC.TS.HSMU를 준용한다.

KCAC.TS.CM이 인증서 저장 규격에 관해 참고한 RFC 2630은 RFC 5652, RFC 3370으로 대체되었고, 이후 다시 일부가 갱신되었다. 또한, KCAC.TS.CM과 KCAC.TS.CT는 전자서명 전송채널 암호화에 RSA-1024를 사용할 수 있도록 명시되어 있으므로 행정전자서명의 RSA 안전성 상향조정과 더불어 두 기술규격의 개정도 필요해 보인다.

4.3.3 인증서 저장소 규격

상세서는 인증서 저장소 규격을 정의하기 위해 RFC 4510과 KCAC.TS.LDAP를 참고한다. KCAC.TS.LDAP는 LDAP에 관한 다수의 RFC 문서를 참고하고 있는데, 대부분이 신규 표준으로 대체된 노후 표준들이다. LDAP는 암호체계의 안전성과 직접적인 관련은 없지만, 최신 국제 표준을 참고하여 KCAC.TS.LDAP를 개정할 필요는 있어 보인다.

4.3.4 보안토큰 기반의 인증서 이용 기술

상세서는 보안토큰(HSM, Hardware Secure Module) 기반의 인증서 사용을 위해 KCAC.TS.HSMU를 준용한다. 그리고 전자서명 알고리즘으로 KCDSA를 사용하기 위한 별도의 사항을 정의하고 있다.

상세서에 명시된 KCDSA 파라미터는 112비트 안전성밖에 제공하지 못한다. 따라서 128비트 이상의 안전성을 제공할 수 있는 파라미터의 사용을 고려할 수 있다.

KCAC.TS.HSMU는 PKCS #11 v.2.11을 준용한다. PKCS #11은 2020년에 v3.0이 출판되었으므로 현행 표준의 준용을 검토해야 할 필요가 있다. 또한, 규격서가 준용하는 암호알고리즘과 키 길이를 128비트 안전성 기준에 맞추어 개정할 필요가 있다.

V. 인증서 관리 및 서비스 기술 검토내용

이 장은 행정전자서명용 인증서의 관리와 사용에 필요한 규격과 기능에 관한 표준을 검토한 결과를 제시한다. 주로 표준 및 관련 기술의 현황과 안전성 파라미터 조정에 관한 사항이다.

5.1 인증서 관리

5.1.1 인증서 관리 프로토콜 및 요청형식

- Certificate Management Protocol(CMP)

상세서는 인증서 관리 프로토콜을 규정하기 위해 현행 표준인 RFC 4210을 준용하고 있다. 다만, RFC 4210을 갱신한 RFC 6712가 존재하는데, 이 표준은 인증서 관리에 HTTP 프로토콜의 POST 방식을 지원하는 방안을 규정하고 있다. 행정전자서명의 POST 방식 지원 필요성을 검토하고, 필요하다면 RFC 6712의 추가 준용을 고려할 수 있다.

- Certificate Request Message Format (CRMF)

CRMF는 RFC 4211과 KCAC.TS.CRMF V1.21을 참고한다. 그런데, KCAC.TS.CRMF은 RFC 4211에 의해 폐지된 RFC 2511을 참고한다. 한편, RFC 4211은 RFC 9045에 의해 갱신되었다. RFC 9045는 PBMAC에서 SHA-1을 SHA-256으로 대체하고, DES와 TDEA를 AES로 대체하고, 패스워드 기반 키 유도 과정의 해시함수 최소 반복 횟수를 1,000회에서 10,000회로 상향하였다. SHA-1, DES, TDEA는 충분한 안전성을 제공하지 못하므로 KCAC.TS.CRMF는 현행 표준인 RFC 9045를 참고하도록 개정할 필요가 있다.

5.1.2 인증서 메시지 처리

- Base64 인코딩

Base64 인코딩은 RFC 2045를 준용한다. 하지만 해당 표준은 인터넷 메시지의 형식에 관한 것으로, Base64 인코딩을 규정하는 것은 아니다. 따라서 Base64 인코딩을 규정하는 RFC 4648의 준용이 필요한 것으로 보인다.

• ASN.1과 ASN.1 인코딩

ASN.1과 ASN.1 인코딩은 각각 ITU-T X.680~683과 X.690~697을 참고하고 있으며, 이들 모두 현행 표준이다.

• Cryptographic Message Syntax(CMS)

행정전자서명 암호체계는 RFC 2315(PKCS #7)가 규정한 암호화 메시지 구문(CMS)을 준용하여 데이터를 송수신한다. RFC 5652는 RFC 2315의 최신 버전이다. 특히, RFC 5652는 암호화 메시지 전송 시에 키 교환, 키 유도 등을 이용할 수 있도록 관련 콘텐츠 유형(content type)을 개정하였다. 한편, RFC 5083은 인증 암호화(GCM, CCM 등)를 지원하기 위한 콘텐츠 유형을 규정하고 있다. 따라서 신규 행정전자서명 암호체계가 다양한 키 공유 방식 및 인증 암호화를 지원하기 위해서는 RFC 5652와 RFC 5083을 추가로 준용하는 것을 고려할 필요가 있다.

5.1.3 개인키 저장 및 전달 방식

• PKCS #5

패스워드 기반 키 유도 기법은 PKCS #5의 현행 표준인 RFC 8018을 참고한다. 이 문서는 키 유도 함수 PBKDF1, PBKDF2, 암호화 함수 PBES1, PBES2, 메시지 인증 코드 PBMAC1을 명시한다. PBKDF1과 PBES1은 충분한 안전성을 제공하지 못하는 MD2, MD5, SHA-1, DES, RC2에 기반을 둔 방식이므로 사용을 배제할 필요가 있다. RFC 8018도 PBKDF1과 PBES1은 하위 호환을 위해서만 사용하는 것을 권고하고 있다.

• PKCS #8

개인키 정보의 문법을 정의하기 위해 상세서가 참고하는 PKCS #8은 RFC 5208에 명시되어 있다. 그런데 RFC 5208은 2010년에 RFC 5958로 대체되었다.

• PKCS #12

PKCS #12는 개인키와 인증서를 전송하기 위한 파일 포맷을 정의한다. 상세서는 PKCS #12 v1.0을 준용하고 있으나 2014년에 최신 버전인 v1.1이 RFC 7292로 제정된 상태이다.

• KCAC.TS.UI

KCAC.TS.UI는 앞서 소개한 3편(PKCS #5, PKCS #8, PKCS #12)의 국제 표준을 인용하여 공인인증기관 간 상호연동을 위한 사용자 인터페이스 관련 기술을 규정하는 표준이다. 따라서 KCAC.TS.UI가 참고하는 3편의 표준도 앞에서 기술한 최신 버전을 참고하도록 개정을 고려해야 한다.

5.2 인증서 서비스

인증서 서비스는 실시간 인증서 유효성 검증, 시점 확인, LDAP, 본인 확인, 시각 동기화, 인증서 검증을 포함한다. 항목별 검토 결과는 다음과 같다.

5.2.1 실시간 인증서 유효성 검증(OCSP)

OCSP(Online Certificate Status Protocol)는 RFC 2560과 RFC 5019(G-SSL 용)를 참고하고 있다. RFC 2560은 RFC 6960으로 대체되었고, RFC 5019는 RFC 8996에 의해 내용이 갱신되었다.

5.2.2 시점 확인

시점 확인 프로토콜은 RFC 3161을 참고하는데, 이는 RFC 5816에 의해 일부 내용이 갱신되었다. 또 시점 확인 요청과 응답 메시지의 무결성 검증에 SHA-1을 사용하는 것으로 명시하고 있으므로 충분한 안전성을 제공할 수 있는 해시함수로 변경하는 것을 고려할 필요가 있다.

5.2.3 LDAP

LDAP의 세부 사항 정의는 RFC 3377을 참고하는데, 이는 2006년에 발간된 RFC 4510으로 대체되었다.

5.2.4 본인 확인(VID)

VID(Virtual ID, 가상 식별번호) 관련 표준은 대체되거나 갱신된 내용은 없다. 그러나 가입자 식별번호에 적용하는 해시함수를 HAS-160으로 명시하고 있으므로 충분한 안전성을 제공할 수 있는 해시함수로 변경하는 것을 고려할 필요가 있다.

5.2.5 시각 동기화(NTP)

NTP(Network Timing Protocol)는 RFC 867, 868, 1305, 1361을 참고하는데, 이 중 RFC 1305와 1361은 RFC 5905로 대체되었으며, 보안 업데이트로 RFC 8573이 있다. RFC 5905로 대체되면서 NTP 버전이 3에서 4로 변경되었다.

RFC 5905는 NTP 패킷을 보호하기 위해 해시 함수 MD5를 이용한 인증값을 사용한다. MD5는 충분한 안전성을 제공하지 못하므로, NTP 패킷 보호에 MD5의 사용을 제한하는 RFC 8573이 발간되었다. 행정전자서명에서는 LEA-CMAC, LSH-HMAC 등의 사용을 고려할 수 있다.

5.2.6 인증서 검증

행정전자서명 암호체계의 인증서 검증은 RFC 3280을 따른다. 인증서 프로파일 항목에서 기술한 바와 같이 RFC 5280으로 현행화가 필요하다.

5.3 무선 인증서

명세서에서 참고하는 무선 인증서 관련 표준은 WAP(Wireless Application Protocol)에서 동작하도록 변형된 TLS(WTLS)를 이용하는 표준들이다. 그런데, WAP는 휴대 전화 등에서 무선 통신을 사용하는 응용 프로그램의 국제 표준으로, 피쳐폰과 같은 저사양의 이동통신 장비에서 웹 브라우저와 같은 서비스를 제공하기 위해 설계된 것이다. 따라서 일반적인 웹페이지에 접속하지 못하고, WML이라는 XML 기반의 언어로 작성된 웹페이지만 접속할 수 있다. 이마저도 통신사 포털 애플리케이션을 통해서만 서비스가 제공되었다.

이는 진정한 의미의 웹 브라우징 연결이라고 보기는 힘든 기술이며, 스마트폰의 등장 이후 사용자가 급격히 감소하였다. 우리나라는 2016년도 4분기를 기점으로 통신사가 제공하는 WAP를 이용하는 서비스가 모두 종료되었다. 따라서 현재는 사용되지 않는 기술이므로 신규 행정전자서명 암호체계에서는 제외하기를 권고한다.

VI. 행정전자서명 암호체계 고도화 방향

이 장은 행정전자서명 암호체계가 128비트 안전성을 제공하고, 최신 기술 및 표준을 반영하도록 하는 고도화 방향을 제시한다.

6.1 암호알고리즘 체계

6.1.1 비대칭키 암호

현재 행정전자서명 암호체계에 포함된 비대칭키 암호알고리즘은 RSA 암호화, RSA 전자서명, KCDSA, ECDSA이다. 이 비대칭키 암호 알고리즘은 모두 양자 컴퓨터의 공격에 취약하다. 따라서 양자 컴퓨터 공격 대비 관점에서는 기존의 비대칭키 암호를 모두 양자내성암호로 대체하는 방향으로 고도화가 진행되어야 한다. 양자내성암호는 현재 NIST의 공모 사업을 계기로 알고리즘 개발 및 표준화 작업이 활발히 진행 중이며, 2024년까지는 표준화가 마무리될 것으로 예상된다. 그러므로 향후 양자내성암호의 개발 및 안전성 연구 동향과 국내외 표준화 진행현황을 고려하여 적절한 도입방안을 마련하는 것이 바람직해 보인다.

하지만 양자내성암호 기술의 안정화, 표준화 시기를 고려하면 현시점에서 특정 양자내성 암호알고리즘을 도입하는 것은 어려워 보인다. 따라서 단기적으로는 기존 비대칭키 암호의 키 길이와 안전성 파라미터 조정을 통해 안전성 수준을 상향하는 것이 필요해 보인다. 128비트 안전성에 부합하기 위해서는 RSA는 3,072비트 이상의 키를 사용하고 암호화에는 EME-OAEP, 전자서명에는 EMSA-PSS 인코딩을 사용하는 것이 적절해 보인다, KCDSA는 공개키와 개인키의 길이를 각각 3,072와 256비트 이상으로 선택하여야 하고, ECDSA는 256비트 이상의 타원곡선을 사용하여야 한다.

한편, 기존 암호체계에서는 ECDSA를 무선 인증서에만 적용하였다. 그러나 현재는 무선 인증서를 사용하지 않으므로(5.3 참고), ECDSA를 암호체계에서 제외하거나, 적절한 활용 방안을 마련할 필요가 있다고 판단된다.

6.1.2 블록암호

현재 행정전자서명 암호체계에 포함된 블록암호 알고리즘은 DES, TDEA, SEED, ARIA-128이다. 이들 중 128비트 안전성을 제공하지 못하는 DES와 TDEA는 하위 호환성이 필요한 경우를 제외하

고는 사용하지 못하도록 하는 것이 바람직해 보인다.

SEED 알고리즘은 사용을 최소화하는 것이 좋을 것으로 판단된다. SEED는 개발된 후 많은 시간이 흘렀으나, 안전성 분석 결과가 거의 발표되지 않아 충분한 분석이 이루어졌다고 보기 어려우며, 이후 개발된 알고리즘에 비하여 효율성 관점에서도 장점을 갖지 못한다. 더구나 128비트 키만으로는 양자 컴퓨터에 대한 충분한 안전성을 제공하지 못할 것으로 예상된다.

ARIA 알고리즘은 192와 256비트 키를 사용할 수 있도록 ARIA-192와 ARIA-256을 추가하는 방향으로 고도화하는 것이 좋다고 판단된다. ARIA 알고리즘은 안전성과 효율성 관점에서 사용에 문제가 없는 알고리즘이며, 사용자의 선택권 및 양자 안전성 확보 관점에서 더 긴 길이의 키를 사용할 수 있도록 할 필요가 있다고 판단된다.

블록암호 고도화 방향 중 하나로 알고리즘의 다양화 관점에서 신규 알고리즘을 암호체계에 추가하는 것이 적절하다고 판단된다. ARIA 알고리즘 개발 이후 여러 우수한 블록암호가 개발되었으며, 그중 LEA 알고리즘[39]은 국내 기술로 개발하여 KS 표준화(KS X 3246:2016) 및 국제표준화(ISO/IEC 29192-2, 2019년)까지 마무리된 상태이다. LEA 알고리즘은 다양한 소프트웨어 환경에서 ARIA와 AES보다 우수한 성능을 제공하며, KCMVP 검증대상 암호알고리즘 목록에 포함되어 널리 사용 중이다.

행정전자서명은 현재 상세서에 명시되지 않았지만, 블록암호 운영모드로 CBC(Cipher Block Chaining) 모드를 사용하고 있다. 고도화의 한 방향으로 효율성 강화를 위하여 병렬화가 가능한 CTR 모드 등의 도입과 기능성 강화를 위한 인증 암호화 운영모드(GCM, CCM 등)의 도입을 검토해 보는 것이 바람직해 보인다. 더욱이 RFC 5083에 명시된 최신 버전의 CMS에는 인증 암호화 운영모드를 지원하는 메시지 형식이 추가되어 있으므로 CMS의 고도화가 병행된다면 인증 암호화 운영모드를 사용할 수 있을 것으로 판단된다.

6.1.3 해시함수

현재 행정전자서명 암호체계에 포함된 해시함수 알고리즘은 HAS-160, SHA-1, SHA-2이다. HAS-160과 SHA-1은 최대 80비트의 안전성을 제공하도록 설계되었을 뿐만 아니라, 앞에서 살펴보았듯이 충분한 안전성을 제공하지 못한다고 알려져 있다. 따라서 HAS-160과 SHA-1은 하위 호환성이 필요한 경우를 제외하고는 사용하지 못하도록 하는 것이 바람직해 보인다.

행정전자서명은 현재 SHA-2 알고리즘의 규격 중 256비트 해시값을 출력하는 규격인 SHA-256만 포함하고 있다. SHA-256 알고리즘이 충분한 안전성과 효율성을 제공하지만, 다양한 수준의 안전성 제공 및 양자 안전성 확보 관점에서 384비트 또는 512비트와 같은 더 긴 길이의 해시값을 출력하는 SHA-2의 규격을 추가할 필요가 있어 보인다.

또한, 사용할 수 있는 안전한 해시함수가 SHA-2 뿐이므로 알고리즘의 다양화 관점에서 신규 알고리즘을 암호체계에 추가하는 방향이 적절하다고 판단된다. 추가를 고려할 수 있는 해시함수 알고리즘으로는 LSH와 SHA-3를 들 수 있다.

LSH 알고리즘[40]은 국내 기술로 개발하여 KS 표준(KS X 3262:2018)으로 제정되어 있으며, KCMVP 검증대상 암호알고리즘이다. LSH 알고리즘은 소프트웨어 고속 구현이 가능하여, PC, 스마트 기기, 클라우드 등 다양한 환경에서 대용량의 데이터를 빠르게 처리할 수 있다.

SHA-3[41]는 미국 NIST가 2008년부터 2012년까지 진행한 신규 해시함수 선정 공모 사업(the SHA-3 cryptographic hash algorithm competition)에서 선정된 알고리즘으로, SHA-1, SHA-2, LSH와는 다른 구조로 설계되었다. SHA-3 알고리즘은 국제표준(ISO/IEC 10118-3:2018)이며, KCMVP 검증대상 암호알고리즘이다.

LSH와 SHA-3는 다양한 규격을 제공하는데, 행정전자서명 암호체계에 LSH와 SHA-3를 추가한다면, 다양한 수준의 안전성 제공 및 양자 안전성 확보 관점에서 256, 384, 512비트의 해시값을 출력하는 규격을 포함하는 것이 적절해 보인다.

6.1.4 메시지 인증 코드(MAC)

메시지 인증 코드는 일반적으로 그 안전성이 해시함수나 블록암호와 같은 기반 프리미티브의 안전성으로 환원됨이 증명되어야 한다. 현재의 행정전자서명 암호체계에 포함된 HMAC은 기반 해시함수의 안전성이 충분하다면 그대로 사용하여도 문제가 없어 보인다. 다만 해시함수 또는 HMAC의 공격 등에 의한 안전성 저하를 대비한 알고리즘의 다양화 관점에

서는 블록암호 기반 MAC의 추가가 필요할 것으로 보인다.

6.1.5 난수 생성

현재의 행정전자서명 암호체계에서 사용하는 난수 생성 알고리즘은 충분한 안전성을 제공하지 못하는 해시함수 SHA-1과 블록암호 DES에 기반한 것일 뿐만 아니라, 폐지된 표준에 포함된 기술이다. 난수 생성 알고리즘은 일반적으로 하위 호환성을 제공할 필요가 없으므로 전면 교체하는 방향으로 고도화가 추진되어도 괜찮을 것으로 판단된다.

미국 NIST가 SP 800-90A에서 제시한 블록암호, 해시함수, HMAC 기반 난수 생성 알고리즘이 ISO/IEC(국제), TTA(국내)에서 모두 표준화되어 있고, 또한 KCMVP 검증대상 암호알고리즘 목록에도 포함되어 국내에서도 널리 사용되고 있다. 그러므로 행정전자서명도 이들 난수 생성 알고리즘을 수용하는 방향으로 고도화하는 것이 적절해 보인다.

6.2 인증서 체계

인증서 규격과 체계는 기존 표준에 대해 대체되거나 갱신된 국제 표준을 식별하여 최신 기술을 적용할 필요가 있다.

인증서 저장 규격 등 인증서 체계는 신규 암호체계에 포함된, 충분한 안전성을 제공하는 암호알고리즘을 사용하고, 관련 기술문서는 이를 명시할 필요가 있다.

6.3 인증서 관리 및 서비스 체계

인증서 관리 및 서비스 체계는 기존 참고 표준 중 갱신 또는 대체된 표준을 식별하여 최신 기술을 적용하는 방향으로 고도화를 진행할 필요가 있다. 특히, CMS와 같이 새 표준이 기존보다 더 다양한 기능을 제공할 수 있는 경우에는 더욱 적극적으로 검토하여 수용할 필요가 있다.

무선 인증서 관련 기능은 2016년도 이후 사용하지 않는 WAP 기술과 관련된 것으로, 신규 행정전자서명에서는 사용하지 않는 방향으로 고도화를 진행할 필요가 있다.

인증서 서비스 중 NTP는 기존 버전 3에서 워크스테이션 및 고속 LAN 환경을 고려하여 설계된 버전 4(RFC 5905 참고)로 변경하는 방향으로 고도화를 진행할 필요가 있다. 또한, NTP 패킷 보호에 MD5 대신에 128비트 안전성을 제공할 수 있는 메시지 인증 코드의 사용이 필요해 보인다(RFC 8573을 참고).

VII. 결 론

행정전자서명 암호체계는 2010년에 개정된 상태로 현재 10년 이상의 시간이 지났다. 국내외의 여러 기관이 기준으로 삼는 최소 안전성 수준을 살펴본 결과, 국내 암호체계도 2030년까지 128비트 이상으로 안전성 수준을 상향하는 것이 적절해 보인다. 더구나 양자 컴퓨터 공격이 점점 현실화하여오고, 이에 대비한 양자내성암호의 표준화가 활발히 진행 중이므로 이러한 신기술의 도입을 고려해 볼 필요가 있다. 따라서 현시점에서 행정전자서명 암호체계에서 사용 중인 암호 기술의 안전성 현황을 검토해 보는 것은 의미 있다고 판단된다.

이 논문은 행정전자서명 암호체계에 적용된 다양한 기술을 암호알고리즘, 인증서 체계, 인증서 저장 및 서비스로 구분하여, 충분한 안전성을 제공하지 못하는 암호알고리즘의 사용, 대체 또는 갱신된 노후 표준의 준용 등 다양한 검토 결과를 제시하였다. 그리고 이들 검토 결과를 바탕으로 신규 암호알고리즘의 도입, 안전성 상향을 위한 파라미터 조정, 노후되거나 취약한 기술의 사용 제한, 최신 참고 표준 제시 등의 암호체계 고도화 방향을 제안하였다.

이 논문에서 제안한 바와 같이 암호알고리즘의 안전성 기준을 128비트로 상향하기 위해서는, 기존 알고리즘의 파라미터를 조정하거나 새로운 알고리즘을 도입하여야 한다. 이는 일부 암호 기능 구현에 속도 저하를 초래할 수 있지만, 행정전자서명의 활용 환경, 최근 소프트웨어와 컴퓨팅 환경 등을 고려하면, 전체 시스템에 영향을 거의 주지 않을 것이다.

이 논문이 제시한 검토 결과와 제안사항이 행정전자서명 고도화를 위한 차세대 암호체계의 개발에 활용되기를 기대한다.

References

[1] 전자정부법[시행 2020. 12. 10.] [법률 제17370호, 2020. 6. 9. 타법개정] 제2조 제9항, 2020년 12월.

[2] GPKI, https://www.gpki.go.kr/jsp/centerIntro/center/effect/searchEffect.jsp, Aug. 2022.
[3] Dept. of policy on digital security Ministry of the Interior and Safety, Government, Public Key Infrastructure Profile and Algorithm Specification, Feb. 2021,
[4] NIST, "Recommendation for Key Management," NIST SP 800-57, May. 2020.
[5] ANSSI, "Guide des Mecanismes cryptographques," Jan. 2020.
[6] ECRYPT-CSA, "Algorithms, Key Size and Protocols Report (2018)," ECRYPT-CSA, Feb. 2018.
[7] BSI, "Cryptographic Mechanisms: Recommendation and Key Legnths," Jan. 2022.
[8] KISA, KISA-GD-2018-0034, https://seed.kisa.or.kr/kisa/Board/38/detailView.do, Dec. 2018.
[9] NIST, "DATA ENCRYPTION STANDARD," FIPS 46-3, Jan. 1977.
[10] REDHAT, "SWEET32: Birthday attacks against TLS ciphers with 64bit block size(CVE-2016-2183)," https://access.redhat.com/articles/2548661, Aug. 2022.
[11] J. Lu, W. Yap, M. Henricksen and S. Heng, "Diffrerntial attack on nine rounds of the SEED block cipher," Information Processing Letters, vol. 114, no. 3, pp. 116-123, Mar. 2014.
[12] D. Kwon, J. Kim, S. Park, S. Sung, Y. Shon, J. Song, Y. Yeom, E. Yoon, S. Lee, J. Lee, S. Chee, D. Han and J. Hong, "New block cipher: ARIA," ICISC 2003, LNCS 2971, pp. 432-445, 2003.
[13] I. Dinur and G. Leurent, "Improved Generic Attacks against Hash-Based MACs and HAIFA," CRYPTO 2014, LNCS 8616, pp.149-168, 2014.
[14] J. Guo, Y. Sasaki, L. Wang, M. Wang and L. Wen, "Equivalent Key Recovery Attacks against HMAC and NMAC with Whirlpool Reduced to 7 Rounds," FSE 2014, LNCS 8540, pp. 571-590, 2014.
[15] A. Yun, S. Sung, S. Park, D. Chang, S. Hong and H. Cho, "Finding Collision on 45-Step HAS-160," ICISC 2005, LNCS 3935, pp. 146-155, 2006.
[16] H. Cho, S. Park, S. Sung and A. Yun, "Collision Search Attack for 53-Step HAS-160," ICISC 2006, LNCS 4296, pp. 286-295, 2006.
[17] F. Mendel and V. Rijmen, "Colliding Message Pair for 53-Step HAS-160," ICISC 2007, LNCS 4817, pp. 324-334, 2007.
[18] F. Mendel, T. Nad and M. Schl¨affer, "Cryptanalysis of Round-Reduced HAS-160," ICISC 2011, LNCS 7259, pp. 33-47, 2011.
[19] Y. Sasaki and K. Aoki, "A Preimage Attack for 52-Step HAS-160," ICISC 2008, LNCS 5461, pp. 302-317, 2008.
[20] Y. Shen and G. Wang, "Improved Preimage Attacks on RIPEMD-160 and HAS-160," KSII Transactions in Internet and Information Systems, vol. 12, no. 2, pp. 727-746, Feb. 2018.
[21] X. Wang, Y. Yin and H. Yu, "Finding Collisions in the Full SHA-1," CRYPTO 2005, LNCS 3621, pp. 17-36, 2005.
[22] M. Stevens, "New Collision Attacks on SHA-1 Based on Optimal Joint Local-Collision Analysis," EUROCRYPT 2013, LNCS 7881, pp. 245-261, 2013.
[23] M. Stevens, P. Karpman and T. Peyrin, "Freestart Collision for Full SHA-1," EUROCRYPT 2016, LNCS 9665, pp. 459-483, 2016.
[24] M. Stevens, E. Bursztein, P.

Karpman, A. Albertini and Y. Markov, "The first collision for full SHA-1," CRYPTO 2017, LNCS 10401, pp. 570-596, 2017.
[25] G. Leurent and T. Peyrin, "From collisions to chosen-prefix collisions application to full SHA-1," EUROCRYPT 2019, LNCS 11478, pp. 527-555, 2019.
[26] G. Leurent and T. Peyrin, "SHA-1 is a Shambles: First Chosen-Prefix Collision on SHA-1 and Application to the PGP Web of Trust," USENIX Security 2020, pp. 1839-1856, Aug. 2020.
[27] D. Khovratovich, C. Rechberger and A. Savelieva, "Bicliques for Preimages: Attacks on Skein-512 and the SHA-2 family," FSE 2012, LNCS 7549, pp. 244-263, 2012.
[28] K. Aoki, J. Guo, K. Matusiewicz, Y. Sasaki and L. Wang, "Preimages for step-reduced SHA-2," ASIACRYPT 2009, LNCS 5912, pp. 578-597, 2009.
[29] J. Guo, S. Ling, C. Rechberger and H. Wang, "Advanced meet-in-the-iddle preimage attacks: First results on full Tiger, and improved results on MD4 and SHA-2," ASIACRYPT 2010, LNCS 6477, pp. 56-75, 2010.
[30] S.K. Sanadhya and P. Sarkar, "New Collision Attacks Against Up To 24-step SHA-2," INDOCRYPT 2008, LNCS 5365, pp. 91-103, 2008.
[31] F. Mendel, T. Nad and M. Schläffer, "Improving Local Collisions: New Attacks on Reduced SHA-256," EUROCRYPT 2013, LNCS 7881, pp. 262-278, 2013.
[32] C. Dobraunig, M. Eichlseder and F. Mendel, "Analysis of SHA-512/224 and SHA-512/256," ASIACRYPT 2015, LNCS 9453, pp. 612-630, 2015.
[33] P.W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," FOCS 1994, pp. 124-134, Nov. 1994.
[34] L.K. Grover, "A fast quantum mechanical algorithm for database search," STOC 1996, pp. 212-219, July 1996.
[35] D.J. Bernstein, "Quantum algorithmsto find collisions," https://blog.cr.yp.to/20171017-collisions.html, Aug. 2022.
[36] NIST, "Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process," https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf, Aug. 2022.
[37] F. Song and A. Yun, "Quantum Security of NMAC and Related Constructions," CRYPTO 2017, LNCS 10402, pp. 283-309, 2017.
[38] M. Kaplan, G. Leurent, A. Leverrier and M. Naya-Plasencia, "Breaking Symmetric Cryptosystems using Quantum Period Finding," CRYPTO 2016, LNCS 9815, pp. 207-237, 2016.
[39] D. Hong, J. Lee, D. Kim, D. Kwon, K. Ryu and D. Lee, "LEA: A 128-bit block cipher for fast encryption on common processors," WISA 2013, LNCS 8267, pp. 3-27, 2013.
[40] D. Kim, D. Hong, J. Lee, W. Kim and D. Kwon, "LSH: A New Fast Secure Hash Function Family," ICISC 2014, LNCS 8949, pp. 286-313, 2014.
[41] NIST, "SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions," FIPS 202, Aug. 2015.

〈저자소개〉

정 영 훈 (Younghoon Jung) 정회원
2011년 2월: 한양대학교 수학과 이학사
2016년 8월: 한양대학교 수학과 이학박사
2015년 12월~현재: ETRI 부설연구소 선임연구원
〈관심분야〉 암호학, 정보보호

노 동 영 (Dongyoung Roh) 정회원
2004년 2월: KAIST 수학과 이학사
2011년 2월: KAIST 수리과학과 이학박사
2011년 2월~2012년 4월: 국가수리과학연구소 연구원
2012년 4월~현재: ETRI 부설연구소 책임연구원
〈관심분야〉 암호학, 정보보호

구 본 욱 (Bonwook Koo) 종신회원
2001년 2월: 한양대학교 수학과 이학사
2003년 2월: 한양대학교 수학과 이학석사
2010년 2월: 한양대학교 수학과 이학박사
2006년 12월~현재: ETRI 부설연구소 책임연구원
〈관심분야〉 암호학, 정보보호