Regular paper | Journal of the KIECS. pp. 683-688, vol. 16, no. 4, Aug. 31. 2021, t. 108, pISSN 1975-8170 | eISSN 2288-2189
http://dx.doi.org/10.13067/JKIECS.2021.16.4.683

해시 기반 양자내성 전자서명 기법 연구 동향

이재흥*

Survey on Hash-Based Post-Quantum Digital Signature Schemes

Jae-Heung Lee*

요 약

미래를 이끌 기술로 주목받고 있는 양자 컴퓨터 기술의 발전으로 RSA나 ECDSA와 같은 전자서명 기술들이 위협받고 있다. 대안으로 격자 기반, 다변수 기반, 코드 기반, 해시 기반 등 다양한 양자내성암호가 연구되고 있는데 그 중 해시 기반은 빠르고 정량적 보안 수준을 계산할 수 있으며 안전성도 증명된 상태여서 많은 관심을 받고 있다. 본 논문에서는 그 동안 제안된 다양한 해시 함수 기반 전자서명 기법들을 살펴보고 각각의 특징 및 장단점을 분석한다. 또한 해시 함수 기반 전자서명 기법이 실질적으로 사용되기 위해서는 서명 크기를 줄이는 것이 무엇보다 중요하다는 점을 강조한다.

ABSTRACT

Digital signature algorithms such as RSA and ECDSA are threatened by the development of quantum computer technology, which is attracting attention as a future technology. Alternatively, various post-quantum algorithms such as grid-based, multivariate-based, code-based, and hash-based are being studied. Among them, the hash-based is a fast and quantitative security level that can be calculated and its safety has been proven. So it is receiving a lot of attention. In this paper, we examine various hash-based digital signature algorithms that have been proposed so far, and analyze their features and their strengths and weaknesses. In addition, we emphasize the importance of reducing the size of the signature in order for the hash-based signature algorithm to be practically used.

키워드

Hash-Based Signature, One-Time Signature, Post-Quantum Signature, Pseudo Random Function
해시 기반 서명, 일회용 서명, 양자내성 서명, 의사난수 함수

Ⅰ. 서 론

정보통신기술(ICT)의 발달로 많은 활동이 온라인 기반으로 바뀌면서 보안의 중요성이 강조되고 있다. 온라인 활동에 있어 무결성과 출처 인증을 제공하기 위해 전자서명 기술이 많이 사용되고 있다[1].

전자서명을 위해 현재 가장 많이 사용되는 알고리즘은 RSA[2]와 ECDSA[3]이다. RSA와 ECDSA의 안전성은 각각 소인수분해와 이산대수 문제에 기반하고 있기 때문에 양자 컴퓨터 개발이 완료되면 Shor의 알고리즘[4]으로 실시간 해독이 가능해져 더 이상 사용할 수 없다. 따라서 대안이 필요한 상황이다.

이러한 대안으로 격자 기반[5], 다변수 기반[6], 코드 기반[7], 해시 기반 등 다양한 양자내성암호가 연구되고

* 교신저자: 대전대학교 정보보안학과
• 접 수 일 : 2021. 06. 05
• 수정완료일 : 2021. 07. 11
• 게재확정일 : 2021. 08. 17
• Received : Jun. 05, 2021, Revised : Jul. 11, 2021, Accepted : Aug. 17, 2021
• Corresponding Author : Jae-Heung Lee
Dept. Information Security, Daejeon University,
Email : leejh@dju.kr

있다. 그 중 격자 기반은 상대적으로 빠르고 서명의 크기도 작으나 정량적 보안 수준을 계산하기가 힘들고 안전성도 증명되지 않은 상태라는 점이 단점이다. 다변수 기반도 상대적으로 빠르고 아주 작은 서명 크기를 가지지만 격자 기반과 마찬가지로 정량적 보안 수준이 모호하고 안전성이 증명되지 않은 상태이다. 코드 기반의 경우 서명의 크기가 작고 어느 정도 정량적 보안 수준을 계산하는 것이 가능하나 키 크기가 너무 크다는 단점을 가진다. 해시 기반은 빠르고 정량적 보안 수준을 계산할 수 있으며 안전성도 증명된 상태[8]라는 것이 강점이다.

본 논문에서는 그 동안 제안된 다양한 해시 함수[9, 10] 기반 전자서명 기법들을 살펴보고 각각의 특징 및 장단점을 분석한다. 이를 통해 앞으로 해시 함수 기반 전자서명이 어떻게 발전해 나갈 것인지 살펴볼 것이다.

본 논문은 다음과 같이 구성된다. 2장에서 다양한 해시 함수 기반 양자내성 전자서명 기법들에 대해 살펴본다. 3장에서 해시 함수 기반 전자서명 기법의 발전 방향에 대해 살펴본 뒤, 4장에서 결론을 제시한다.

II. 해시 기반 양자내성 전자서명 기법

2.1 Lamport 일회용 서명 기법

한 비트로 된 메시지를 서명하는 경우를 가정하자. 서명자는 두 개의 난수 x_0와 x_1을 생성하고, 각각에 일방향 함수 f를 적용해 $y_0 = f(x_0)$와 $y_1 = f(x_1)$을 계산한다. 이 때 y_0와 y_1은 외부에 공개하고 x_0와 x_1은 서명자만 알도록 비밀로 한다. 이 후 메시지를 서명할 때, 서명하고자 하는 메시지가 '0' 이면 x_0가 서명이 되고, '1'이면 x_1이 서명이 된다.

이 경우 메시지 값이 '0'이면 검증자는 $y_0 = f(x_0)$임을 확인함으로써 서명자가 '0'이라는 메시지에 대해 서명을 했음을 제3자에게 증명할 수 있다. x_0와 x_1 값은 서명자만이 알고 있고, 이미 공개된 y_0와 y_1 값만 가지고 $y_0 = f(x_0')$이나 $y_1 = f(x_1')$인 x_0'이나 x_1'을 구하는 것은 일방향 함수의 특성에 의해 불가능하므로 검증자가 $y_0 = f(x_0)$인 x_0를 제시할 수 있었다는 것은 서명자가 '0'이라는 메시지에 대해 서명의 의미로 x_0를 검증자에게 공개했음을 의미하기 때문이다. 위의 알고리즘을 여러 비트로 바로 확장시키면 Lamport 일회용 서명 기법[11]이 된다.

Lamport 일회용 서명 기법의 가장 큰 단점은 비밀 키와 공개 키 그리고 서명의 크기가 메시지의 크기에 비례해 너무 크다는 점이다. 이를 보완하기 위해 다양한 기법들이 제안되었다.

2.2 Winternitz 일회용 서명 기법

Winternitz 일회용 서명 기법[12]은 메시지에 따라 서로 다른 횟수의 일방향 함수를 서명에 적용시킴으로써 서명의 크기를 줄인다. 이 기법의 기본 아이디어를 설명하기 위해 두 비트로 된 메시지를 서명하는 경우를 살펴보자. 서명자는 두 개의 난수 x_0와 x_1을 생성하고, 각각에 일방향 함수 f를 3번 적용해 $y_0 = f^3(x_0)$와 $y_1 = f^3(x_1)$을 계산한다. 이 후 Lamport 일회용 서명 기법에서와 마찬가지로 y_0와 y_1은 외부에 공개하고 x_0와 x_1은 서명자만 알도록 비밀로 한다. 메시지에 대한 서명은 다음과 같다.

- 메시지가 '0'일 경우 서명은 $(f^0(x_0), f^3(x_1))$이다.
- 메시지가 '1'일 경우 서명은 $(f^1(x_0), f^2(x_1))$이다.
- 메시지가 '2'일 경우 서명은 $(f^2(x_0), f^1(x_1))$이다.
- 메시지가 '3'일 경우 서명은 $(f^3(x_0), f^0(x_1))$이다.

여기서 x_0와 x_1의 두 개의 값을 사용하는 이유는 x_0만 사용하면 검증자가 실제로는 '1'에 해당하는 서명인 $f^1(x_0)$를 받았는데 여기에 일방향 함수 f를 적용해 '2'에 해당하는 서명인 $f^2(x_0)$를 받았다고 주장할 수도 있기 때문이다. x_1도 같이 사용하면 $f^2(x_1)$을 가지고 $f^1(x_1)$을 구해야 수신자가 위와 같은 주장을 할 수 있는데 이는 일방향 함수의 특성에 의해 불가능하다. 따라서 x_1도 같이 사용함으로써 위와 같은 서명 위조를 막을 수 있다.

위의 아이디어를 일반화시키고 x_1 대신 체크섬을 사용하면 Winternitz 일회용 서명 기법이 된다. Winternitz 일회용 서명 기법의 키 생성, 서명 생성, 서명 검증 과정은 아래 그림 1, 2, 3과 같다.

Input: Security parameters l, w, message length $|m|$
Output: Secret key $(x_1, x_2, \ldots, x_L)$, public key $(y_1, y_2, \ldots, y_L)$
1: Choose $(x_1, x_2, \ldots, x_L) \xleftarrow{\$} \{0,1\}^{(L,l)}$
2: Compute $(y_1, y_2, \ldots, y_L) = (f^{w-1}(x_1), f^{w-1}(x_2), \ldots, f^{w-1}(x_L))$
$(L = L_1 + L_2, L_1 = \lceil \frac{|m|}{\log_2 w} \rceil, L_2 = \lfloor \frac{\log_2 (L_1 (w-1))}{\log_2 w} \rfloor + 1)$

그림 1. Winternitz 일회용 서명 키 생성
Fig. 1 Winternitz one-time signature key generation

Input: Message m, secret key $(x_1, x_2, \ldots, x_L)$
Output: Signature $(sig_1, sig_2, \ldots, sig_L)$
1: Split m into L_1 bit strings of length $\log_2 w$. $(m = m_1 m_2 \ldots m_{L_1})$
2: Compute the checksum $c = \sum_{i=1}^{L_1} (w - 1 - m_i)$
3: Split c into L_2 bit strings of length $\log_2 w$. $(c = c_1 c_2 \ldots c_{L_2})$
4: Compute $sig_i = f^{m_i}(x_i)$ for all $i \in \{1, 2, \ldots, L_1\}$
5: Compute $sig_{(L_1+i)} = f^{c_i}(x_{(L_1+i)})$ for all $i \in \{1, 2, \ldots, L_2\}$

그림 2. Winternitz 일회용 서명 서명 생성
Fig. 2 Winternitz one-time signature signing

Input: Message m, signature $(sig_1, sig_2, \ldots, sig_L)$, public key $(y_1, y_2, \ldots, y_L)$
Output: "accept" or "reject"
1: Split m into L_1 bit strings of length $\log_2 w$. $(m = m_1 m_2 \ldots m_{L_1})$
2: Compute the checksum $c = \sum_{i=1}^{L_1} (w - 1 - m_i)$
3: Split c into L_2 bit strings of length $\log_2 w$. $(c = c_1 c_2 \ldots c_{L_2})$
4: **if** there exists $i \in \{1, 2, \ldots, L_1\}$ such that $f^{w-1-m_i}(sig_i) \neq y_i$ **then**
5: **return** "reject"
6: **else if** there exists $i \in \{1, 2, \ldots, L_2\}$ such that $f^{w-1-c_i}(sig_{(L_1+i)}) \neq y_{(L_1+i)}$ **then**
7: **return** "reject"
8: **return** "accept"

그림 3. Winternitz 일회용 서명 서명 검증
Fig. 3 Winternitz one-time signature verification

2.3 개선 Winternitz 일회용 서명 기법

기존 Winternitz 일회용 서명 기법에서는 키 생성, 서명 생성, 서명 검증 과정에서 매번 같은 일방향 함수 f를 사용한다. 이 경우 생일 공격을 통한 공격이 가능하기 때문에 b 비트의 보안성을 제공하기 위해서는 $2b$ 비트 이상의 보안 매개변수 값이 필요하다.

개선 Winternitz 일회용 서명 기법[13]은 식 (1)과 같은 의사 난수 함수들의 패밀리를 사용해 같은 입력에 대해 매번 다른 의사 난수 함수를 적용함으로써 이러한 제한을 피한다.

$$F(n) = \{ \mathrm{f_k} : \{0,1\}^{\mathrm{n}} \rightarrow \{0,1\}^{\mathrm{n}} \,|\, \mathrm{k} \in \{0,1\}^{\mathrm{n}} \} \qquad (1)$$

이를 통해 더 작은 보안 매개변수 값으로도 기존과 같은 보안성을 제공할 수 있으며 서명의 크기가 줄어들게 된다. 해시 기반 전자서명의 가장 큰 단점이 서명 크기이기 때문에 이는 중요하다. 또한 적응적 선택 메시지 공격에 대해 존재적 위조불가임이 증명되었다.

식 (2)는 기존 Winternitz 일회용 서명 기법에서 일방향 함수 f를 적용하는 과정을 나타내고, 식 (3)은 개선 Winternitz 일회용 서명 기법에서 의사 난수 함수 f_k를 적용하는 과정을 나타낸다.

$$x \rightarrow f(x) \rightarrow f(f(x)) \qquad (2)$$

$$k \rightarrow f_k(x) \rightarrow f_{f_k(x)}(x) \qquad (3)$$

2.4 W-OTS+

W-OTS+[14]는 개선 Winternitz 일회용 서명 기법에서 의사 난수 함수 f_k를 적용하는 과정에 식 (4)와 같이 무작위화 요소 $\vec{r}$을 추가하여 보안성을 높인다.

$$\vec{r} = (r_1, r_2, r_3, \cdots, r_{w-1}) \qquad (4)$$

식 (5)와 식 (6)은 W-OTS+에서 의사 난수 함수 f_k를 적용하는 과정에 무작위화 요소 $\vec{r}$이 추가되는 과정을 나타낸다.

$$k \rightarrow c_k^1(x, \vec{r}) \rightarrow c_k^2(x, \vec{r}) \rightarrow \ldots \rightarrow c_k^{w-1}(x, \vec{r}) \qquad (5)$$

$$\begin{aligned} c_k^i(x, \vec{r}) &= f_k(c_k^{i-1}(x, \vec{r}) \oplus r_i) \\ c_k^0(x, \vec{r}) &= k \end{aligned} \qquad (6)$$

W-OTS+는 기존 기법들보다 더 작은 보안 매개변수 값으로도 기존과 같은 보안성을 제공할 수 있다. W-OTS+를 뒤에 나오는 XMSS에 적용할 경우 기존 기법들을 적용할 때보다 같은 보안성을 유지하면서 서명 크기를 50% 이상 줄이는 것이 가능하다.

2.5 HORS

HORS[15, 16]는 t개의 원소를 가지는 비밀 키 집합에서 암호학적 해시 함수 H를 통해 서명하고자 하는 메시지에 따라 서로 다른 k개의 원소를 선택하여 이들로 서명을 구성한다. HORS의 키 생성, 서명 생성, 서명 검증 과정은 아래 그림 4, 5, 6과 같다.

Input: Security parameters l, t
Output: Secret key $(s_1, s_2, \ldots, s_t)$, public key $(v_1, v_2, \ldots, v_t)$
1: Choose $(s_1, s_2, \ldots, s_t) \xleftarrow{\$} \{0,1\}^{(t,l)}$
2: Compute $(v_1, v_2, \ldots, v_t) = (f(s_1), f(s_2), \ldots, f(s_t))$

그림 4. HORS 키 생성
Fig. 4 HORS key generation

Input: Message m, secret key $(s_1, s_2, \ldots, s_t)$
Output: Signature $(sig_1, sig_2, \ldots, sig_k)$
1: Compute $h = H(m)$
2: Split h into k pieces $(h_1, h_2, \ldots, h_k)$ of length $\log_2 t$ bits each
3: Interpret each h_j as an integer i_j for all $j \in \{1, 2, \ldots, k\}$
4: Compute $sig_j = s_{i_j}$ for all $j \in \{1, 2, \ldots, k\}$

그림 5. HORS 서명 생성
Fig. 5 HORS signing

Input: Message m, signature $(sig_1, sig_2, \ldots, sig_k)$, public key $(v_1, v_2, \ldots, v_t)$
Output: "accept" or "reject"
1: Compute $h = H(m)$
2: Split h into k pieces $(h_1, h_2, \ldots, h_k)$ of length $\log_2 t$ bits each
3: Interpret each h_j as an integer i_j for all $j \in \{1, 2, \ldots, k\}$
4: **if** there exists $j \in \{1, 2, \ldots, k\}$ such that $f(sig_j) \neq v_{i_j}$ **then**
5: **return** "reject"
6: **return** "accept"

그림 6. HORS 서명 검증
Fig. 6 HORS verification

2.6 XMSS

앞에서 살펴본 Lamport 일회용 서명 기법이나 Winternitz 일회용 서명 기법은 이름이 의미하는 바대로 하나의 메시지만 서명 가능하다. XMSS(eXtended Merkle Signature Scheme)[17]는 XMSS 트리라 불리는 변형된 머클 트리 구조를 통해 일회용 서명 기법을 여러 번 사용할 수 있게 해준다.

XMSS 트리의 기본 구조는 머클 트리와 같지만 머클 트리가 두 자식 노드의 연결 값을 해시 함수에 넣는 방식으로 부모 노드 값을 구하는데 반해 XMSS 트리는 식 (7)과 같이 두 자식 노드의 연결 값을 해시 함수에 넣을 때 각 레벨마다 고유한 랜덤 비트마스크 값을 XOR하는 방식을 사용한다.

$$N_{i,j} = h_k((N_{2i,j-1} \oplus BM_{l,j}) \| (N_{2i+1,j-1} \oplus BM_{r,j} \quad (7)$$

XMSS 트리의 리프 노드 구성은 다음과 같다. 각 리프 노드마다 하나의 일회용 서명 검증키를 보관하며 따라서 XMSS 트리의 높이가 H이면 2^H개의 메시지까지 서명 가능하다. 리프 노드에서 서명 검증키를 보관하기 위해 L-tree라 불리는 자료 구조를 사용하며 L-tree의 리프 노드 수는 2^k 형태가 아니기 때문에 XMSS 트리와 약간 다른 구조를 가지지만 두 자식 노드의 연결 값을 해시 함수에 넣을 때 각 레벨마다 고유한 랜덤 비트마스크 값을 XOR하는 방식을 사용하는 점은 마찬가지이다.

2.7 SPHINCS

SPHINCS[18]는 크게 세 부분에서 XMSS와 다르다.

첫째, XMSS는 Stateful 하지만 SPHINCS는 Stateless 하다. XMSS를 포함하여 여러 번 서명 가능한 대부분의 기존 해시 기반 전자서명 기법들이 서명을 생성할 때마다 비밀키 업데이트를 수행한다. 이는 서명을 생성하더라도 비밀키를 바꾸지 않는 일반적인 전자 서명 개념과는 다르다.

둘째, 리프 노드에서 사용하는 일회용 서명 기법을 HORST(HORS with trees)라 불리는 다회 서명 기법으로 변경하였다. HORS의 경우 공개키 크기가 크기 때문에 실행 시간이 약간 늘어나더라도 공개키 크기와 서명 크기를 줄이기 위해 트리 구조를 추가하였다.

셋째, 하이퍼 트리 구조를 채택하였다. XMSS에서 2^H개의 메시지까지 서명 가능하도록 하기 위해서는 미리 2^H개의 일회용 서명 키를 만들어 두어야 한다. 이러한 키 생성 시간을 줄이기 위해 SPHINCS에서는 높이 H/d인 트리를 d개의 레이어로 연결하여 서명 크기를 약간 희생하고 키 생성 시간을 줄였다.

2.8 Gravity-SPHINCS

Gravity-SPHINCS[19]는 기존 SPHINCS에 다섯 가지 최적화를 적용하였다.

첫째, PORS(PRNG to Obtain a Random Subset)라 불리는 보안성이 강화된 HORS의 변형을 사용하였다. HORS가 제안된 논문에서 HORS의 보안성은 비적응적 선택 메시지 공격에 대해서만 계산되어 있다. 또한 HORS를 그대로 구현할 경우 서명을 위한 비밀키를 선택할 때 충돌이 일어날 수 있는데 이 경우 보안성이 약해진다. PORS는 서명을 위한 비밀키를 선택할 때 해시 함수가 아닌 PRNG(Pseudo Random Number Generator)를 사용하여 이러한 HORS의 단점을 보완하였다.

둘째, 비밀 키 캐싱을 통해 서명 시간과 서명 크기를 줄였다. 매번 서명을 수행할 때마다 필요한 계산들을 잘 관찰해보면 중복되는 부분들이 상당히 있다. 이러한 부분들을 캐싱을 통해 개선할 수 있었다.

셋째, 배치 서명 기술을 적용하여 여러 메시지를 동시에 서명할 때 서명 시간을 분산시키고 서명 크기를 줄일 수 있었다.

넷째, 마스크를 제거한 해시를 통해 보안성을 약간 희생하여 키 크기를 줄이고 알고리즘을 단순화하였다. 마스크를 적용하는 이유는 충돌 저항 요건을 제 2 역상 저항 요건으로 완화하기 위함인데 최근 연구 결과에 따르면 양자 컴퓨터에 대해서는 충돌 저항 요건과 제 2 역상 저항 요건이 비슷한 보안성을 가진다고 알려져 있다[20].

다섯째, 특정 높이 이상의 트리 내 모든 노드 정보를 미리 다 저장하여 머클 트리 인증 경로의 중복을 줄이는 방법을 적용하여 서명 크기를 줄였다.

이러한 기술들을 통해 기존 SPHINCS보다 공개키, 비밀키, 서명 크기를 모두 줄일 수 있었고, 서명 및 검증에 걸리는 시간도 줄일 수 있었다.

III. 발전 방향

2장에서 살펴본 바와 같이 다양한 해시 기반 양자내성 전자서명 기법들이 개발 중이다. SPHINCS의 개선 알고리즘인 SPHINCS+의 경우 표 1과 같이 미국 연방정부의 표준 기술을 제정하는 NIST(National Institute of Standards and Technology)에서 진행 중인 포스트 양자 암호 알고리즘 표준화 진행 과정에서 전자서명 분야 3라운드 대체 후보 알고리즘으로 채택된 상황이다.

표 1. NIST 포스트 양자 암호 3라운드 제출
Table 1. NIST Post-quantum cryptography submissions

	Round 3 Finalists	Alternate Candidates
PKE / KEA	Classic McEliece	BIKE
	CRYSTALS-KYBER	FrodoKEM
	NTRU	HQC
	SABER	NTRU Prime
		SIKE
DSA	CRYSTALS-DILITHIUM	GeMSS
	FALCON	Picnic
	Rainbow	**SPHINCS+**

해시 기반 양자내성 전자서명의 경우 타원 곡선 암호보다 먼저 소개되었고 상당 기간 동안 안전성이 검증된 방식이라 신뢰성에 큰 강점을 가지고 있다. 하지만 서명 크기가 상대적으로 크다는 단점이 있어 이를 줄이기 위한 추가 연구가 필요해 보인다. 한 예로 128비트 보안을 제공하기 위해 필요한 서명의 크기가 타원 곡선의 경우 65B 정도이지만 SPHINCS는 41KB, SPHINCS+는 8KB로 각각 640배, 120배 정도이다.

IV. 결론

본 논문에서는 그 동안 제안된 다양한 해시 함수 기반 전자서명 기법들을 살펴보고 각각의 특징 및 장단점을 분석하였다. 또한 해시 함수 기반 전자서명 기법이 실질적으로 사용되기 위해서는 서명 크기를 줄이는 것이 무엇보다 중요하다는 점을 강조하였다.

미래를 이끌 기술로 주목받고 있는 양자 컴퓨터 기술의 발전으로 다양한 양자내성 암호들이 주목받고 있다. 격자 기반, 다변수 기반, 코드 기반, 해시 기반 등 어떠한 기술이 차세대 표준이 될지 아직 알 수 없기 때문에 해시 기반 전자서명 기법에 대한 추가 연구도 다방면으로 진행될 것으로 보인다. 본 연구가 앞으로의 해시 기반 양자내성 전자서명 기법 연구에 도움이 될 수 있으리라 기대한다.

감사의 글

이 논문은 2020학년도 대전대학교 교내학술연구비 지원에 의해 연구되었음.

References

[1] Y. Kim, “On a Deterministic Attack Against The RSA Cryptosystem,” *J. of the Korea Institute of Electronic Communication Sciences*, vol. 13, no. 4, 2018, pp. 737-744.

[2] R. L. Rivest, A. Shamir, and L. Adleman, “A Method for Obtaining Digital Signatures and Public-Key Cryptosystems,” *Communications of the ACM*, vol. 21, no. 2, Feb. 1978, pp. 120-126.

[3] D. Johnson, A. Menezes, and S. Vanstone, “The Elliptic Curve Digital Signature Algorithm (ECDSA),” *Int. J. of Information Security*, vol. 1, no. 1, Aug 2001, pp. 36-63.

[4] P. W. Shor, “Algorithms for quantum computation: discrete logarithms and factoring,” In *Proc. 35th Annual Symposium on Foundations of Computer Science*, Santa Fe, NM, USA, 1994, pp. 124-134.

[5] J. Hoffstein, N. Howgrave-Graham, J. Pipher, J. H. Silverman, and W. Whyte, “NTRUSign: Digital signatures using the NTRU lattice,” *Lecture Notes in Computer Science*, vol. 2612, 2003, pp. 122-140.

[6] J. Porras, J. Baena, and J. Ding, “ZHFE, A New Multivariate Public Key Encryption Scheme,” *Lecture Notes in Computer Science,* vol. 8772, 2014, pp. 229-245.

[7] R. J. McEliece, “A Public-Key Cryptosystem Based On Algebraic Coding Theory,” *The Deep Space Network Progress Report*, vol. 42, no. 44, 1978, pp. 114-116.

[8] L. K. Grover, “A fast quantum mechanical algorithm for database search,” In *Proc. the*

Annual ACM Symposium on Theory of Computing, Philadelphia, PA, USA, July 1996, pp. 212-219.

[9] C. Lee, "Security Authentication Technique using Hash Code in Wireless RFID Environments," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 14, no. 6, 2019, pp. 1077-1082.

[10] H. Lee and J. Oh, "SHA-256 based Encapsulated Electronic Medical Record Document Storage System," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 15, no. 1, 2020, pp. 199-204.

[11] L. Lamport, "Constructing Digital Signatures from a One Way Function," *Technical Report SRI-CSL-98*, Oct. 1979.

[12] R. C. Merkle, "A Certified Digital Signature," *Lecture Notes in Computer Science*, vol. 435, 1990, pp. 218-238.

[13] J. Buchmann, E. Dahmen, S. Ereth, A. Hülsing, and M. Rückert, "On the security of the Winternitz one-time signature scheme," *Int. J. of Applied Cryptography*, vol. 3, no. 1, 2013, pp. 84-96.

[14] A. Hülsing, "W-OTS+ - Shorter signatures for hash-based signature schemes," *Lecture Notes in Computer Science*, vol. 7918, 2013, pp. 173-188.

[15] L. Reyzin and N. Reyzin, "Better than BiBa: Short one-time signatures with fast signing and verifying," *Lecture Notes in Computer Science*, vol. 2384, 2002, pp. 144-153.

[16] J. Lee, S. Kim, Y. Cho, Y. Chung, and Y. Park, "HORSIC: An efficient one-time signature scheme for wireless sensor networks," *Information Processing Letters*, vol. 112, no. 20, 2012, pp. 783-787.

[17] J. Buchmann, E. Dahmen, and A. Hülsing, "XMSS - A practical forward secure signature scheme based on minimal security assumptions," *Lecture Notes in Computer Science*, vol. 7071, 2011, pp. 117-129.

[18] D. J. Bernstein, D. Hopwood, A. Hülsing, T. Lange, R. Niederhagen, L. Papachristodoulou, M. Schneider, P. Schwabe, and Z. Wilcox-O'hearn, "SPHINCS: Practical stateless hash-based signatures," *Lecture Notes in Computer Science*, vol. 9056, 2015, pp. 368-397.

[19] J. P. Aumasson and G. Endignoux, "Improving stateless hash-based signatures," *Lecture Notes in Computer Science*, vol. 10808, 2018, pp. 219-242.

[20] A. Chailloux, M. Naya-Plasencia, and A. Schrottenloher, "An Efficient Quantum Collision Search Algorithm and Implications on Symmetric Cryptography," *Lecture Notes in Computer Science*, vol. 10625, 2017, pp. 211 - 240.

저자 소개

이재흥(Jae-Heung Lee)

2001년 서울대학교 컴퓨터공학부 졸업(공학사)
2003년 서울대학교 대학원 전기·컴퓨터공학부 졸업(공학석사)
2013년 서울대학교 대학원 전기·컴퓨터공학부 졸업(공학박사)
2016년~현재 대전대학교 정보보안학과 조교수
2013년~현재 한국전자통신학회 종신회원
※ 관심분야 : 정보보안, 암호학, 시스템보안