블록체인 기반 분산형 키 관리 시스템의 설계[1)]

안형철* 김광조*

*카이스트 정보보호대학원

Design of Blockchain-based Decentralized Key Management System

Hyeongcheol An* Kwangjo Kim*

*Graduate School of Information Security, KAIST.

요 약

블록체인은 2008년 Bitcoin에서 처음으로 제안되었으며, 분산형 데이터베이스 기술이다. 현재 키 관리 시스템 중 하나인 공개키 기반 구조(Public Key Infrastructure, PKI) 기술은 중앙집중형 시스템으로 단일 장애 지점(single point failure)의 가능성이 존재한다. 또한 대표적인 암호화폐인 비트코인과 이더리움에서 사용되는 전자서명 알고리즘인 ECDSA는 Shor 알고리즘에 따라 양자 컴퓨터 공격에 취약하다는 문제점이 존재한다. 이에, 본 논문에서는 양자 내성 암호 기술을 적용한 블록체인 기반 키 관리 시스템에 대하여 제안한다. 분산형 구조인 블록체인을 기반으로 설계하여 single point failure에 안전하다. 또한, 대표적인 양자 내성 암호(Quantum-resistant Cryptography)인 래티스(lattice)기반 전자서명인 GLP 전자서명을 사용하기 때문에 양자컴퓨터의 공격에도 안전하며, 장기간 안전성을 보장한다.

I. 서론

최근 분산형 데이터베이스 시스템인 블록체인이 큰 각광을 받고 있다. 블록체인은 2008년 Nakamoto의 비트코인(Bitcoin) [1]에서 처음으로 제안되었다. 비트코인을 포함한 암호화폐는 기존의 중앙집중형 서버-클라이언트 방식이 아닌 P2P(peer to peer) 방식으로 통신을 하기 때문에 분산형 데이터베이스로 활용할 수 있다. 암호학적 관점에서 비트코인은 전자서명 방식으로 타원곡선(ECDSA)를 사용하고 해시함수로 SHA-256을 사용하고 있다. 또한 이더리움(Ethereum) [2]과 같은 2세대 블록체인에서도 ECDSA를 적용하여 전자서명을 한다.

한편, 비트코인에서 사용하는 전자서명 암호 기술 중 하나인 ECDSA는 이산 대수 문제의 어려움에 기반하기 때문에 Shor 알고리즘 [3]에 따라 양자 컴퓨터 공격에 깨지게 된다. 따라서 양자컴퓨터 공격에도 안전한 전자서명이 필요하다. 양자 내성 암호(Quantum-resistant Cryptography)의 필요성이 제기되고 있으며, 그 중 하나로 래티스(lattice)기반 전자서명 알고리즘이 알려져 있다.

본 논문에서는 양자 내성 기술을 적용한 블록체인 기반 키 관리 시스템을 제안한다. 기존의 암호화폐의 블록체인에서 사용하는 전자서명 알고리즘인 ECDSA를 대체하여 GLP [4] 전자서명 알고리즘을 사용하여 양자컴퓨터의 공격에 안전하게 설계한다. 또한 기존의 공개키 기반 구조(Public Key Infrastructure, PKI)의 문제점인 single point failure를 분산형 구조인 블록체인을 기반으로 설계하였기 때문에 안전하다.

1) 본 연구는 2017년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행되었습니다. (No.2017-0-00555, 양자 컴퓨터 환경에서 래티스 문제를 이용한 다자간 인증키교환 프로토콜 연구)

1.1 논문의 구성

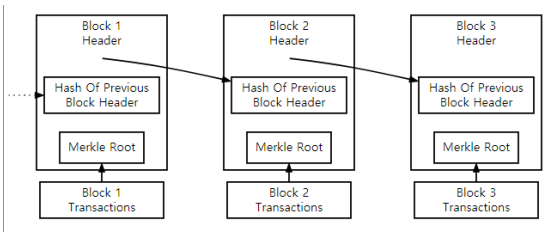
본 논문의 구성으로 Ⅱ장에서는 블록체인과 PKI 시스템에 대하여 간략히 설명하고, Ⅲ장은 블록체인 기반 키 관리 시스템과 프로토콜에 대하여 제안한다. Ⅳ장에서는 기존에 사용하고 있는 PKI 시스템과의 비교에 대하여 논한다. 마지막으로 Ⅴ장에서는 결론과 본 논문 성과를 이용한 추후 연구에 대한 가능성을 제시한다.

Ⅱ. 배경지식

2.1 블록체인

비트코인은 최초의 분산 암호 및 가상화폐이며 2008 년 Nakamoto에 의해 P2P 네트워크 기반으로 설계되었다. 중앙집중형 방식인 서버-클라이언트 방식을 사용하지 않기 때문에 합의 프로토콜(consensus protocol)이 필요하며, 비트코인에서는 작업증명(Proof-of-Work, PoW) 방식을 사용한다. 블록체인 네트워크의 모든 사용자는 공개키 암호화를 사용하여 트랜잭션을 생성하고 전파할 수 있다. [그림 1]은 비트코인 블록체인의 단순화한 구조이다. 모든 블록의 헤더에는 이전 블록 헤더의 해시 값이 존재한다. 또한 각 블록의 트랜잭션을 사용하여 머클 해시 트리(Merkle hash tree)를 만들 수 있다. 이전 블록의 해시값을 새로운 블록에 추가하는 방법으로 이전 블록의 변조에 대한 안전성을 확보할 수 있다.

비트코인이 제안 된 이후, 2014년 이더리움 등 많은 암호화폐가 제안되었다. 따라서 블록체인은 TTP(Trusted Third Party)가 없어도 무결성을 지원하는 데이터베이스를 만들 수 있다.



[그림 1] 블록체인의 구조

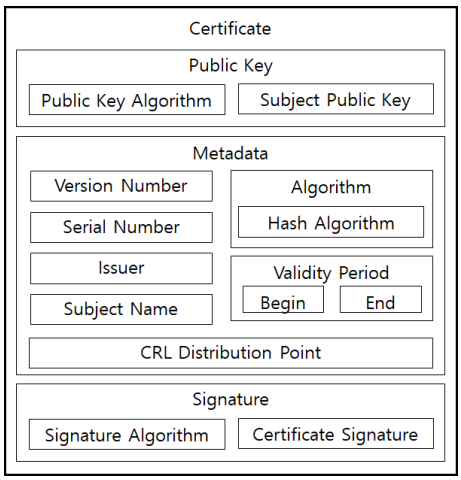
2.2 공개키 기반 구조

PKI 시스템의 인증서는 공개키 및 메타데이터(metadata)가 포함된 디지털 문서이다. 합법적인 CA (Certificated Authority)만이 유효한 인증서를 발급할 수 있다. 현재 PKI 시스템은 X.509 [5] 표준으로 정의되어 있다. [그림 2]는 X.509 v3 인증서의 구조에 대하여 설명한다.

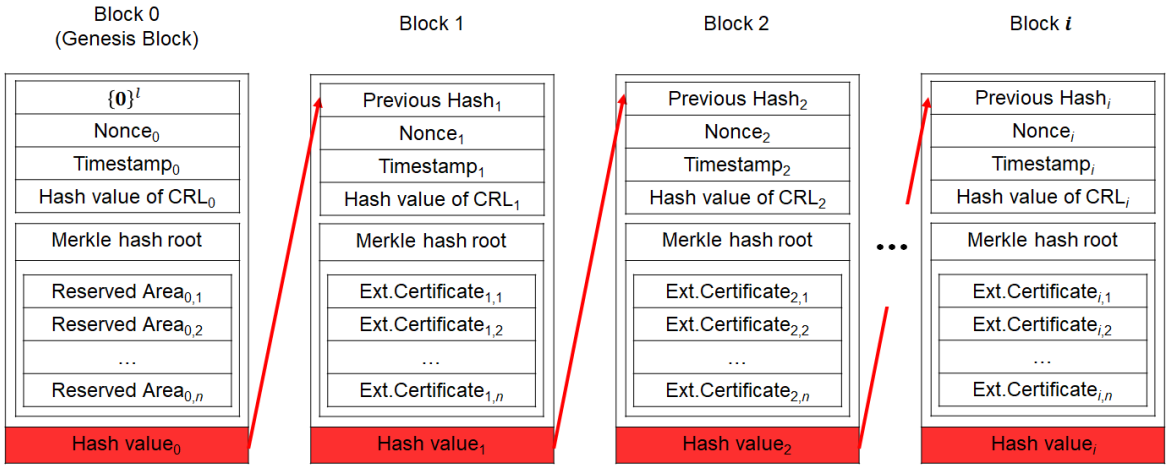
현재 PKI 시스템은 중앙 집중식 데이터베이스를 기반이 특징이다. 그러나 이로 인하여 단일 장애 지점(single point failure)의 취약점이 있다. 블록체인은 분산된 네트워크에서 변경되지 않는 원장을 제공하는 것을 목표로 하기 때문에 공개키의 저장 및 관리에 적합하다. 블록체인 기반 키 관리 시스템에 대한 최근 연구결과는 다음과 같다.

Emercoin (EMC) [6]은 블록체인 기반 PKI 시스템에 사용되는 암호화폐로, OpenSSH와 EMC 블록체인을 통합하여 분산 PKI를 구성한다. 합의 프로토콜로 PoW 및 PoS (Proof-of-Stake)를 사용한다. 또한 SHA-256을 해시 함수로 사용한다. 암호학적 특징 때문에 EMC는 양자 컴퓨팅 공격으로부터 안전하지 않다는 단점이 있다.

Matsumoto 등은 2016년에 IKP [7]라는 이더리움 기반 PKI 시스템을 제안하였다. 그러나 IKP는 이더리움 플랫폼에 기반하여 설계하였기 때문에 장기간 안전성을 보장하지 못한다. 이더리움은 ECDSA 서명 알고리즘을 사용하기 때문에 양자컴퓨터의 공격에 취약하다는 단점이 존재한다.



[그림 2] X.509 v3 인증서의 구조



[그림 3] 양자 내성 블록체인 키 관리 시스템의 구조

블록체인 기반 키 관리 시스템

3.1 블록체인 기반 키 관리 시스템의 설계

- **Setup**(1^λ): 파라미터 λ를 선택하고 $n, q, \sigma = \sqrt{16/2} \approx 2.828$을 출력한다.

- **KeyGen**(n, q): 다항식 곱셈과 덧셈에서 NTT (Number Theoretic Transformation) 연산을 사용하여 가우시안 분포로부터 샘플링 된 다항식 r_1, r_2를 출력한다. 각각의 사용자 i에 대한 공개키는 $(\hat{a_i}, \hat{p_i}, \hat{t_i}) \in pk_i$이고, 비밀키는 $(\hat{r_{1,i}}, \hat{r_{2,i}}, \hat{y_{1,i}}, \hat{y_{2,i}}) \in sk_i$이다.

- **GenesisBlock()**: [그림 3]과 같이 Setup, Merkle, Final로 구성되어 있으며, 초기에 첫 번째 블록을 생성한다. 해시값은 다음과 같이 계산한다.

$$H_{Block0} = H((\{0\})^n || nonce || timestamp || H_{root})$$

- **User.Setup**(pk_i, H_{root}): 사용자의 공개키를 추가하기 위하여 setup을 한다.

- **User.Add**(ID_i, $Username$, sk_i, pk_i): 사용자의 공개키 추가는 다음과 같이 계산한다.

$$H(ID_i), \quad ID_i \leftarrow \text{User ID};$$

$$H(Username_i), \quad Username_i \leftarrow \text{Username};$$

$$(\hat{r}_{1,i}, \hat{r}_{2,i}, \hat{y}_{1,i}, \hat{y}_{2,i}) \leftarrow sk_i;$$

$$y_{1,i} \leftarrow \text{NTT}^{-1}(\hat{y}_{1,i}); \quad y_{2,i} \leftarrow \text{NTT}^{-1}(\hat{y}_{2,i});$$

$$(\hat{a}_i, \hat{p}_i) \leftarrow pk_i; \quad a_i \leftarrow \text{NTT}^{-1}(\hat{a}_i);$$

$$c_i \leftarrow H(a_i y_{1,i} + y_{2,i}, ID_i); \quad \hat{c}_i \leftarrow \text{NTT}(c)$$

$$r_{1,i} \leftarrow \text{NTT}^{-1}(\hat{r}_{1,i}); \quad r_{2,i} \leftarrow \text{NTT}^{-1}(\hat{r}_{2_i});$$

$$Sign(ID_i, a_i, r_{1,i}, r_{2,i});$$

- **User.Verify**(ID_i, pk_i, **Sign**(ID_i)): 사용자의 공개키 및 서명을 검증하기 위하여 다음과 같이 계산한다.

$$\hat{a}_i, \hat{t}_i \leftarrow pk_i;$$

$$a_i \leftarrow \text{NTT}^{-1}(\hat{a}_i); \quad t_i \leftarrow \text{NTT}^{-1}(\hat{t}_i);$$

$$z_{1,i}, z_{2,i}, \hat{c}_i \leftarrow Sign(ID_i);$$

$$c_i \leftarrow \text{NTT}^{-1}(\hat{c}_i);$$

$$Verify(ID_i, z_{1,i}, z_{2,i}, c_i, a_i, t_i);$$

- **User.Enc**(pk_i, m): 메시지 m을 암호화하기 위하여 다음과 같이 계산한다.

$$(\hat{a}_i, \hat{p}_i, \hat{t}_i) \leftarrow pk_i;$$

$$(a_i, p_i, t_i) \leftarrow (\text{NTT}^{-1}(\hat{a}_i), \text{NTT}^{-1}(\hat{p}_i), \text{NTT}^{-1}(\hat{t}_i));$$

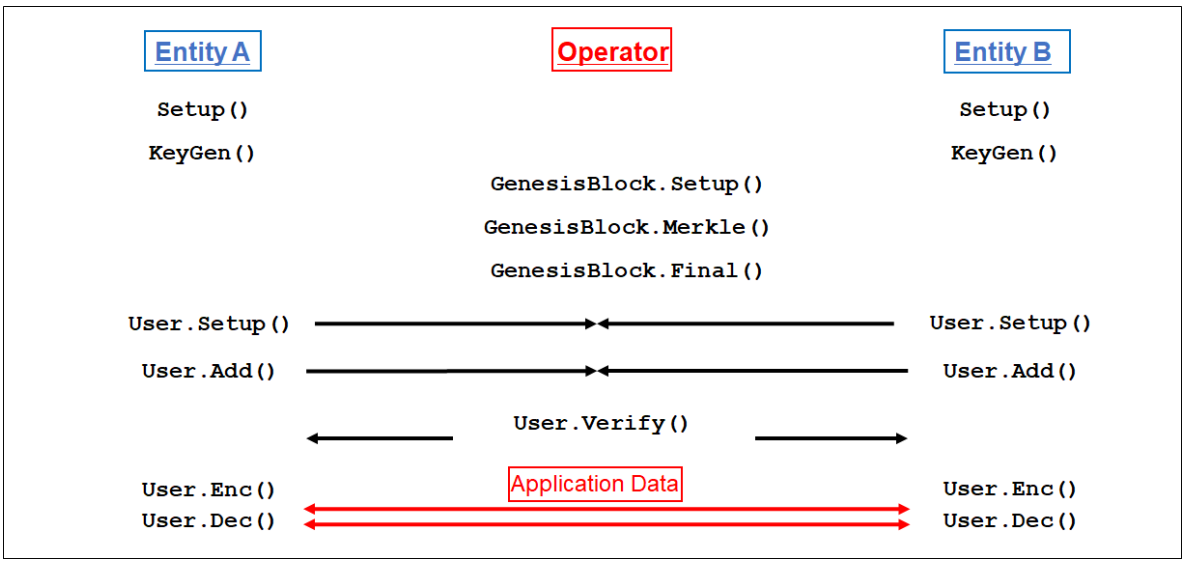
$$e_1, e_2, e_3 \leftarrow \chi_\sigma;$$

$$\hat{e}_1 \leftarrow \text{NTT}(e_1); \quad \hat{e}_2 \leftarrow \text{NTT}(e_2);$$

$$\hat{m} \leftarrow m \cdot \left\lfloor \frac{q}{2} \right\rfloor;$$

$$(\hat{c}_1, \hat{c}_2) \leftarrow (\hat{a}_i * \hat{e}_1 + \hat{e}_2, \quad \hat{p}_i * \hat{e}_1 + \text{NTT}(e_3 + \hat{m}));$$

암호문 $c = (\hat{c_1}, \hat{c_2})$를 출력한다.



[그림 4] 양자 내성 블록체인 기반 키 관리 프로토콜의 예시

- **User.Dec(** sk_i, c**)**: 암호화된 메시지 $c=(\hat{c_1},\hat{c_2})$를 복호화 하기 위하여, 비밀키 sk_i를 사용하여 다음과 같이 계산한다.

$$\hat{r}_{2,i} \leftarrow sk_i;$$

$$(\hat{c}_1, \hat{c}_2) \leftarrow c;$$

$$m' \leftarrow \mathsf{NTT}^{-1}(\hat{c}_1 * \hat{r}_2 + \hat{c}_2);$$

$$m \leftarrow \mathsf{Decode}(m');$$

Decode() 함수는 다음과 같이 정의한다.

$$\mathsf{Decode}(m) := \left\lfloor \frac{2}{q} \cdot m \cdot \lfloor q/2 \rfloor \right\rceil \cdot \lfloor \frac{q}{2} \rfloor$$

[그림 4]는 본 논문에서 제안한 양자 내성 기반 키 관리 프로토콜 예시에 대한 그림이다.

3.2 GLP 서명 알고리즘

래티스 기반 전자서명 알고리즘인 GLP[8]는 Ring-LWE에 기반하여 양자컴퓨터 공격에 안전하다. 본 논문에서 사용하는 수정 GLP 전자서명은 다항식 덧셈과 곱셈에서 NTT 연산을 추가하여 연산의 효율성을 증가하였다.

· 해시함수: $H: \{0,1\}^* \rightarrow D_{32}^n$

· 서명키: $r_1, r_2 \xleftarrow{\$} \chi_\sigma$

· 검증키: $a \xleftarrow{\$} R_q,\ \hat{a} \leftarrow NTT(a),\ \hat{r_1} \leftarrow NTT(r_1),$

$\hat{r_2} \leftarrow NTT(r_2),\ \hat{t} \leftarrow \hat{a} \bullet \hat{r_1} + \hat{r_2}$

Sign(μ, a, r_1, r_2**)**

```
2  begin
3  |  y1, y2 ←$ R_q^k;
4  |  c ← H(a y1 + y2, μ);
5  |  ĉ = NTT(c);
6  |  ẑ1 ← r̂1 * ĉ + ŷ1;
7  |  ẑ2 ← r̂2 * ĉ + ŷ2;
8  |  z1 ← NTT^-1(ẑ1);
9  |  z2 ← NTT^-1(ẑ2);
10 |  if z1 ∉ R_q^(k-32) or z2 ∉ R_q^(k-32) then
11 |  |  go to line 3;
12 |  else
13 |  |  return (z1, z2, c);
14 |  end
15 end
```

Verify(μ, z_1, z_2, c, a, t**)**

```
17 begin
18 |  if z1, z2 ∈ R_q^(k-32) then
19 |  |  c ≠ H(a z1 + z2 − t c, μ);
20 |  |  return reject;
21 |  else
22 |  |  return success;
23 |  end
24 end
```

[표 1] 양자 내성 블록체인 기반 키 관리 시스템과 X.509 v3와의 비교

키 관리 시스템	양자 내성 블록체인 기반 키 관리 시스템	X.509 v3
Connection	Off-line	On-line
Non-repudiation	O	O
Revocation	O	O
Quantum-resistance	O	X
Scalability	$O(n)$	$O(n)$
Trust Model	분산형	중앙집중형

III. 블록체인 기반 키 관리 시스템의 안전성 분석

4.1 안전성 및 공격 복잡도 분석

Grover 알고리즘 [9]은 양자컴퓨터를 이용한 효율적인 데이터베이스 검색 알고리즘이다. 본 논문에서 설계한 블록체인 기반 키 관리 시스템에서 n_1비트의 해시함수의 경우, 전수조사(brute-force) 공격의 복잡도는 $O(2^{n_1/2})$이다. 또한, n_2비트의 비밀키를 가지는 Ring-LWE 기반 전자서명인 GLP에서 SVP(Shortest Vector Problem)을 찾는 경우 $2^{1.799n_2+O(n_2)}$이다 [10]. 따라서 양자컴퓨터를 이용한 공격의 경우 전체 공격복잡도는 $\min(O(2^{n_1/2}), 2^{1.779n_2, O(n_2)})$인 것을 알 수 있다.

한편 현재 사용하는 슈퍼컴퓨터를 활용한 공격에서는 Grover 알고리즘은 적용되지 않는다. 따라서 해시함수의 공격 복잡도는 $O(2^{n_1})$이다. 또한 GLP 전자서명에서 사용하는 Ring-LWE의 SVP를 찾는 경우 효율적인 다항식 시간 내에 풀 수 있는 효율적인 알고리즘은 존재하지 않는다. 현재까지 알려진 가장 효율적인 BKZ-2.0 [11]을 사용할 경우 공격복잡도는 $2^{2.465n_2+O(n_2)}$이다. 따라서 슈퍼컴퓨터를 사용한 공격자를 가정할 경우 전체 공격복잡도는 $\min(O(n_1), 2^{2.465n_2+O(n_2)})$이다.

결론적으로 양자컴퓨터를 포함한 모든 공격자에 대하여 안전하므로 본 논문에서 제시한 블록체인 기반 키 관리 시스템은 장기간 안전성을 달성할 수 있다.

4.2 기존 키 관리 시스템과의 비교

[표 1]에서는 양자 내성 블록체인 기반 키 관리 시스템과 X.509 v3과의 비교를 보여준다.

- Connection: 시작 블록을 제외한 오프라인 상태를 유지할 수 있다. 반면에 X.509는 TTP가 항상 온라인 상태를 유지해야 한다. TTP가 오프라인인 경우 사용자는 공개키가 인증되었는지 확인할 수 없다.

- Non-repudiation: 서명이 있는 사용자의 공개키로 구성된 블록을 가지고 있다. 사용자는 공개키 및 사용자 ID와 같은 공개 정보를 거부 할 수 없다. X.509는 공개키, 사용자 이름 및 서명으로 구성된 인증서를 가지고 있다. 따라서 사용자는 자신의 유효한 인증서를 거부 할 수 없다.

- Revocation: 각 블록 및 사용자 정보에 타임스탬프를 사용한다. 초기에 각 블록에 타임스탬프를 사용하여 만료 시간을 설정할 수 있다. 또한 revocation에 대한 복잡도는 $O(\log_2 n)$이다. X.509는 사용자 인증서에 해지 기록을 저장한다. 그리고 CRL에 대한 블록체인을 만들어 해지 목록을 관리 할 수 있다.

-Scalability: 본 설계의 확장성에 대한 복잡도는 $O(n)$이다. 그러나 사용자 및 공개 정보 수가 증가하더라도 TTP 서버 수를 늘릴 필요가 없다는 점이 장점이다. 그러나 X.509는 사용자의 공개 정보를 추가하기 위해 서버 확장에 대한 문제가 존재한다.

-Trust model: 본 설계의 핵심은 PKI 시스템에 대한 분산 서비스이며, TTP가 필요하지 않다. X.509는 TTP가 필수적이므로 단일 장애 지점(single point failure)의 문제가 존재한다.

IV. 결론

본 논문에서는 양자 내성 기술을 적용한 블록체인 기반 키 관리 시스템에 대하여 제안하였다. 기존의 블록체인 기반 키 관리 시스템은 양자컴퓨터의 공격에 해독이 가능하다는 단점이 존재한다. 그러나 본 논문의 설계에서는 양자 내성 전자서명을 적용하여 양자컴퓨터의 공격에도 안전하다. 또한 양자 및 고전 공격자에 대한 공격복잡도를 제시하여 안전한 것을 검증하였다. 한편 기존의 X.509기반 PKI 시스템과 대비하여 분산형 구조로 설계한 것을 확인할 수 있었으며, 키 관리 시스템에서의 요구조건을 만족하는 것을 알 수 있었다.

향후 연구 과제로는 합의 프로토콜에 대한 고려가 필요하다. 키 관리 시스템의 경우 private, consortium, public 블록체인에 모두 활용될 수 있다. 따라서 public 블록체인의 PoW 또는 PoS, private과 consortium 블록체인에서는 BFT 계열의 합의 프로토콜을 적용할 것 이다.실제 구현하여 본 연구 성과의 성능을 검증할 것이다. 또한 블록체인 기술은 키 관리 시스템 뿐만 아니라 접근통제(access control), 인증과 같은 부분에 사용될 수 있기 때문에 연구를 확장할 예정이다.

[참고문헌]

[1] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” 2008.

[2] G. Wood, “Ethereum: A secure decentralized generalized transaction ledger,” Ethereum Project, Yellow Paper, vol. 151, 2014.

[3] P. W. Shor, “Algorithms for Quantum Computation: Discrete Logarithms and Factoring,” in Proceedings of the Foundations of Computer Science, 35th Annual Symposium on IEEE, 1994, pp. 124-134.

[4] T. Güneysu, V. Lyubashevsky, and T. Pöppelmann, "Practical lattice-based cryptography: A signature scheme for embedded systems," in Conference on Cryptographic Hardware and Embedded Systems, CHES'12, pp. 530-547, Springer, 2012.

[5] P. Yee, “Updates to the internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile," 2013.

[6] “Emercoin", https://emercoin.com

[7] S. Matsumoto and R. M. Reischuk, “IKP: Turning a PKI around with blockchains.," IACR Cryptology ePrint Archive, 2016/1018, 2016.

[8] Z. Brakerski, A. Langlois, C. Peikert, O. Regev, and D. Stehle, “Classical hardness of learning with errors," in Proceedings of the forty-fth annual ACM symposium on Theory of computing, STOC'13, pp. 575-584, ACM, 2013.

[9] T. Güneysu, V. Lyubashevsky, and T. Pöppelmann, ”Practical lattice-based cryptography: A signature scheme for embedded systems.," in Conference on Cryptographic Hardware and Embedded Systems, CHES'12, pp. 530-547, Springer, 2012.

[10] L. K. Grover, ”A fast quantum mechanical algorithm for database search," in Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing, STOC'96, pp. 212-219, ACM, 1996.

[11] Y. Chen and P. Q. Nguyen, “BKZ 2.0: Better lattice security estimates," in International Conference on the Theory and Application of Cryptology and Information Security Asiacrypt'11, pp. 1-20, Springer, 2011.