International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 11 (2019) pp. 2608-2615

A Survey on Post-Quantum Cryptography for Constrained Devices

Kumar Sekhar Roy and Hemanta Kumar Kalita

Abstract

The rise of Quantum computers in the recent years have given a major setback to classical and widely used cryptography schemes such as RSA(Rivest-Shamir-Adleman) Algorithm and ECC (Elliptic Curve Cryptography). RSA and ECC depends on integer factorization problem and discrete logarithm problem respectively, which can be easily solved by Quantum Computers of sufficiently large size running the infamous Shor's Algorithm. Therefore cryptography schemes which are difficult to solve in both traditional as well as Quantum Computers need to be evaluated. In our paper we provide a rigorous survey on Post-Quantum Cryptography schemes and emphasize on their applicability to provide security in constrained devices. We provide a detailed insight over the schemes which could possibly replace RSA and ECC for security in constrained deices.

Keywords: PQC(Post Quantum Cryptography), IoT(Internet of Things), RSA(Rivest-Shamir-Adleman)algorithm, ECC (Elliptic Curve Cryptosystem), Ring-LWE(Learning with Error), AES(Advanced Encryption Standard), Constrained devices.

1 INTRODUCTION

The arrival of Quantum computers have raised an immediate need for viable replacements of classical and widely used cryptography schemes dependent on integer factorization problem and discrete logarithm problem such as RSA and ECC. Quantum Computers were theoretical until 2015 when NASA publicly demonstrated their Quantum Computer jointly developed with D-wave and Google. There are several advances in the field of Quantum Computing since then. Quantum Computers differ from traditional binary electronic computers in several aspects. Commonly used digital computations uses data in the form of definitive binary digits which can be in either state i.e. 0 or 1 whereas quantum computation uses quantum bits (qubits), which can be in superposition of states i.e. there is no definitive state. It has several advantages over traditional electronic computers, several sorts of computations which were not possible in electronic computers can be easily solved using Quantum computers. One such algorithm is the Shor's Algorithm, It was proposed by Peter Shor in his paper titled "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer" [1]. Shor's algorithm can solve integer factorization problem as well as discrete logarithm problem used by RSA as well as ECC respectively in polynomial time using a sufficiently large Quantum Computer. Thus making the use of cryptosystems based on integer factorization problem as well as discrete logarithm problem obsolete. This current advances has raised a genuine need for development of cryptosystems which could serve as viable replacement for traditionally used cryptosystems which are vulnerable to quantum computer based attacks. Since the arrival of IoT, the Cyber security scenario has entirely shifted towards security schemes which are lightweight in terms of computational complexity, power consumption, memory consumption etc. This schemes also need to be secure against all known attacks. Most of the recently proposed schemes for constrained devices use RSA or ECC.

Table 1: Post-quantum cryptography

Sl no.	Family	Algorithm
1	Lattice based Cryptography	NTRU
		Ring LWE
		BLISS
2	Multivariate Cryptography	Rainbow
3	Hash based Cryptography	Lamport Signature
		Merkle Signature
4	Code based Cryptography	McEllice
		Niederreiter

As mentioned earlier this schemes are not secure against attacks raised by Quantum Computers. Thus in our research we try to evaluate the cryptosystems which are not vulnerable to Quantum computer based attacks also popularly known as Post-Quantum Cryptography. We evaluate the Post-Quantum cartographic algorithms as per the suggestion made in Report on PostQuantum Cryptography by NIST [3].

2 LITERATURE REVIEW

NIST as well as several authors have suggested several Post-Quantum cryptosystem which could replace RSA and ECC [6] [3] [7]. In this section we explore and critically review these cryptosystems.

Kumar Sekhar Roy and Hemant Kumar Kalita are with the Department of Information Technology, North Eastern Hill University, Shillong, Meghalaya, 793022, India.

2.1 Multivariate cryptography

These schemes are based on multivariate polynomials over a finite field F. These schemes are either defined in ground or expansion field, solving such problem are either NP-hard or NP-complete. Therefore they are strong contenders of Post-Quantum cryptography. Multivariate cryptography has one very important advantage i.e. It uses very short signature [19], which can serve the purpose of authentication in small devices.

2.1.1 Rainbow

J. Ding and D. Schmidt in 2005 proposed a new signature scheme based on multivariate cryptography called Rainbow [2], more specifically the idea of these scheme is based on Oil and Vinegar schemes [4] [5].

The Scheme uses the following principle:

Let K be a finite field such that $K = GF(2^8)$ and S be the set 1, . . . , n. Let $v_1,...,v_{u+1}, u \geq 1$ be integers arranged as $0 < v_1 < v_2 < .... < v_u < v_{u+1} = n$. Let the sets of integers be defined as $S_i = 1,..,v_i$ for i = 1, . . . , u. Let the initialization be set to $o_i = v_{i+1} - v_i$ and $O_i = \{v_i + 1,...,v_{i+1}\}(i = 1,...,u)$. The total of elements in S_i be v_i and there is $| O_i | = o_i$. For $k = v_1 + 1,...,n$ multivariate quadratic polynomials be defined in the n variables $x_1,...,x_n$ by

$$f_k(x) = \sum_{i \in O_i, j \in S_i} \alpha_{i,j}^{(k)} x_i x_j + \sum_{i,j \in S_i, i \leq j} \beta_{i,j}^{(k)} x_i x_j + \sum_{i \in s_i \cup o_i} \gamma^{(k) x_{(i)}} + \eta^{(k)},$$

where l be the solitary integer such that $k \in O_l$. These are Oil and Vinegar polynomials with $x_i, i \in S_l$ as the Vinegar variables and $x_j, j \in O_l$ as the Oil variables. The map $F(x) = (f_{v1+1}(x),...,f_n(x))$ can be inverted as follows: Firstly, $x_1,...,x_{v1}$ are chosen randomly. Therefore, a system of o_1 linear equations (given by the polynomials $f_k(k \in O_1)$) in the o_1 unknowns $x_{v1+1},...,x_{v2}$ can be obtained, which is solvable by Gaussian Elimination method. Then the calculated values of $x_i(i \in O_1)$ are put in the polynomials $f_k(x)(k > v_2)$ and a system of o_2 linear equations is derived(given by the polynomials $f_k(k \in O_2)$) in the o_2 with unknowns $x_i(i \in O_2)$. By Reiterating the process values for all the variables $x_i(i = 1,...,n)$[3] can be obtained. .

Key Generation:

1) The private key consists of two invertible affine maps $L_1 : K^m \rightarrow K^m$ and $L_2 : K^n \rightarrow K^n$ and the map $F = (f_{v1+1}(x),...,f_n(x))$.
2) Here, $m = nv_1$ is the number of components of F.
3) The public key consists of the field K and the composed map $P(x) = L_1 \circ F \circ L_2(x) : K^n \rightarrow K^m$.

Signature:

1) To sign a document d, we use a hash function $h : K^* \rightarrow K^m$ to compute the value $h = h(d) \in K^m$. 2) Then we compute recursively $x = L^{-1}{}_1{}^1(h), y = F^{-1}(x)$ *and* $z = L^{-1}{}_2{}^1(y)$.
3) The signature of the document is $z \in K^n$.
4) Here, $F^{-1}(x)$ means finding one (of the possibly many) pre-image of x.

Signature verification:

1) To verify the authenticity of a signature, one simply computes h = P(z)
2) Compute hash value h = h(d) of the document.
3) If h = h holds, the signature is accepted, otherwise rejected.

2.2 Lattice based cryptography

Lattices are geometric objects that have evolved into a major player in cryptography. Latticebased schemes have come to be proven as highly resistant to sub-exponential and quantum attacks. Hard mathematical problems related to lattices were first suggested as the basis for cryptography almost two decades ago. Lattice were first studied by mathematicians such as Joseph Louis Lagrange and Carl Friedrich Gauss. Although Mikls Ajtai first showed in a seminal result the use of lattices as a cryptosystem [8]. A lattice L is a set of points in the n-dimensional Euclidean space Rn in real analysis, It has a strong periodicity property.

Any basis of L can be defined as a set of vectors arranged in such a way that any element of L is uniquely represented as their linear grouping with integer coefficients. Each lattice has infinitely many different bases when the value of n is at least 2. All lattices over Rn have infinitely many elements, whereas in cryptography entities such as the cipher-text, public key, and private key must be chosen from a finite space (bit strings of some fixed length).

2.2.1 NTRU

The first version of the system was developed by mathematicians Jeffrey Hoffstein (de), Jill Pipher, and Joseph H. Silverman [20] in 1996, which was called NTRU . In our survey we come across the latest variant of NTRU i.e. NTRU Prime [23], proposed by Bernstein et al. in 2016, in their paper they prove that their algorithm is stronger than the original NTRU by creating stronger algebraic structure. The algorithm is as follows.

Key generation:

The receiver generates a public key as follows:

1) Generate a uniform random small element $g \in R$. Repeat this step until g is invertible in R=3.
2) Generate a uniform random t-small element $f \in R$. (Note that f is nonzero and hence invertible in R/q, since t 1.)
3) Compute $h = g/(3f)$ in R/q. (By assumption q is a prime larger than 3, so 3 is invertible in R/q, so $3f$ is invertible in R/q.)
4) Encode h as a string h'. The public key is h'.
5) Save the following secrets: f in R; and $1/g$ in $R/3$.

Encryption

The sender generates a ciphertext as follows:

1) Decode the public key h', obtaining $h \in R=q$.
2) Generate a uniform random t-small element $r \in R$.
3) Compute $hr \in R=q$.

4) Round each coefficient of *hr*, viewed as an integer between -(*q* - 1)/2 and (q - 1) /2, to the nearest multiple of 3, producing $c \in R$. (If $q \in 1 + 3Z$, as in the case study $q = 9829$, then each coefficient of *c* is in {-(*q* - 1)/2, . . . , -6, -3, 0, 3, 6, . . . , (*q* 1)/2}. If $q \in 2 + 3Z$ then each coefficient of *c* is in {-(*q* + 1)/2, . . . , -6, -3, 0, 3, 6, . . . , (*q* + 1)/2}.)
5) Encode *c* as a string *c'*.
6) Hash *r*, obtaining a left half *C* ("key confirmation") and a right half *K*.
7) The cipher-text is the concatenation *Cc'*.

   The session key is *K*.

Decryption:

The receiver decapsulates a cipher-text *Cc*' as follows:

1) Decode *c'*, obtaining $c \in R$.
2) Multiply by 3*f* in *R*/*q*.
3) View each coefficient of 3*fc* in *R*/*q* as an integer between -(q - 1)=2 and (*q* - 1) /2, and then reduce modulo 3, obtaining a polynomial *e* in *R*/3.
4) Multiply by 1/*g* in *R*/3.
5) Lift *e*/*g* in *R*/3 to a small polynomial $r' \in R$.
6) Compute *c', C', K'* from *r'* as in encryption.
7) If *r'* is *t*-small, *c'* = *c*, and *C'* = *C*, then output *K'*. Otherwise output False.

If *Cc'* is a legitimate cipher-text then *c* is obtained by rounding the coefficients of *hr* to the nearest multiples of 3; i.e., $c = m + hr$ in $R=q$, where *m* is small.

In 2007 Nick Howgrave-Graham in his research titled "A Hybrid Lattice-Reduction and Meetin-the-Middle Attack Against NTRU" performed an attack which included lattice reduction at first and then performed meet in the middle attack [11]. The attack methodology performed faster than odlyzko's attack [27]. The author assumed this attack as an improved result of attack performed by Coppersmith et al. which used only lattice reduction [21] presented in 1996. The author suggests that for NTRU to be secure the private vector needed to be thickened or use a trinary vector which would make meet in the middle attack substantially harder to perform without increasing the parameter *N* by much. The NTRU prime algorithm although was published much later (2016) than these proposed attacks, not enough evidence exist that the proposed attacks would work on NTRU prime as well.

2.2.2 Ring-LWE

Ring-LWE is more properly called as Learning with Errors over Rings and is merely a bigger learning with errors (LWE) problem dedicated to polynomial rings over finite fields [24]. It is built over the arithmetic of polynomials with coefficients chosen from a finite field. The solution to the RLWE problem may be reducible to the NP-Hard Shortest Vector Problem (SVP) in a Lattice, which is an important feature of basing cryptography on the ring learning with errors problem. Ring LWE can be used for several purpose such as key-exchange, Digital signature as well as homomorphic encryption. We evaluate an instance of Ring LWE Digital signature scheme by Lyubashevsky et al. [13]

Public key generation:

An entity wishing to sign messages generates its public key through the following steps:

1) Create two small polynomials $p_0(x)$ and $p_1(x)$ with coefficients selected uniformly from the set -1, 0, 1
2) Calculate $t(x) = a(x).p_0(x) + p_1(x)$
3) Hand out t(x) as the entity's public key

The polynomials $p_0(x)$ and $p_1(x)$ oblige as the private key and t(x) is the concerned public key. The security of this signature scheme is based on the following problem. Given a polynomial t(x) find small polynomials $f_1(x)$ and $f_2(x)$ such that: $a(x).f_1(x) + f_2(x) = t(x)$

Signature generation:

1) Create two small polynomials $d_0(x)$ and $d_1(x)$ with coefficients chosen from the set -b, ..., 0, ..., b
2) Calculate $w(x) = a(x).d_0(x) + d_1(x)$
3) Transform w(x) into a bit string ω
4) Calculate c(x) = POLYHASH(ω | m) (This is a polynomial with k non-zero coefficients. The "|" denotes concatenation of strings)
5) Calculate $s_0(x) = p_0(x).c(x) + d_0(x)$
6) Calculate $s_1(x) = p_1(x).c(x) + d_1(x)$
7) Unless the infinity norms of $s_0(x)$ and $s_1(x) \leq \beta$ is satisfied go to step 1. (This is the denial sampling step noted above)
8) The signature is the tripartite of polynomials c(x), $s_0(x)$ and $s_1(x)$
9) Transmit the message along with c(x), $s_0(x)$ and $s_1(x)$ to the verifier.

Signature verification:

To verify a message m articulated as a bit string, the verifying entity must possess the signer's public key (t(x)), the signature (c(x), $s_0(x)$, $s_1(x)$), and the message m. The verifier does the following:

1) Verify that the infinity norms of $s_0(x)$ and $s_1(x)$, if not reject the signature.
2) Calculate $w'(x) = a(x).s_0(x) + s_1(x)$ t(x)c(x)
3) Transform w'(x) into a bit string ω'
4) Calculate $c'(x) = \text{HASH}(\omega^0 | m)$
5) If c'(x) 6= c(x) discard the signature, otherwise agree to take the signature as valid.

In Our survey we also come across an efficient implementation of the ring-lwe problem by R de Clercq et al. [15]. The algorithm is as follows.

Key generation:

1) Two polynomials r_1 and r_2 are sampled from $X\sigma$ using a discrete Gaussian sampler.
2) $r_1^0 \leftarrow \text{NTT}(r_1)$
3) $r_2' \leftarrow \text{NTT}(\quad r_2)$

4) $p_1 \leftarrow r_1 - a * r_2$ The private key is r'_2 and the public key is (a^0, p^0).

Encryption:

1) The input message m is encoded to a polynomial $m \in R_q$.
2) Error polynomials e_1, e_2, $e_3 \in R_q$ are generated from $X\sigma$ using a discrete Gaussian sampler.
3) $e'_1 \in$ NTT(e_1).
4) $e^0{}_2 \in$ NTT(e_2)
5) ($c01$,$c02$) $\leftarrow$ ($a0 * e01 + e02$;$p0 * e01 + NTT(e3 + m0)$

Decryption:

1) The inverse NTT is performed to compute m = INTT($c'_1 * r'_2 + c'_2) \in Rq$. The original message m is recovered from m by using a decoder. They use the parameter sets (n, q,) from [9].

Park et al. [12] in 2016 proposed SPA(Simple Power Analysis) attack on unprotected Ring LWE scheme proposed by Roy et al. [14]

This particular Ring LWE variant also utilized NTT operation and 8-bit implementation. They proved that their attack could deduce the secret by using [$log_2 q$] executions. Although their attack was not performed on the algorithm proposed by R de Clercq et al., it has quite few similarities. Further research is required to analyze Ring LWE.

2.2.3 BLISS

Lo Ducas, Alain Durmus, Tancrde Lepoint and Vadim Lyubashevsky in their 2013 paper "Lattice Signature and Bimodal Gaussians" proposed a Bimodal Lattice Signature Scheme(BLISS) [10]. BLISS became very popular as it claimed to have better computational efficiency, smaller signature size, and higher security. It attracted the attention of several research groups such as NIST which proposed further refinement of the algorithm.

The algorithm is as follows:

Signature:

The user would provide the input as Message m, public key $\mathbf{A} \in Z_{2q}^{n \times m}$, secret key $\mathbf{S} \in Z_2^{m \times n} q$, stand. dev. $\sigma \in R$

1) $\mathbf{y} \leftarrow D_\sigma^m$
2) $\mathbf{c} \leftarrow H(\mathbf{A}\mathbf{y} mod 2q, m)$
3) Select a random bit $b \in \{0,1\}$
4) $\mathbf{z} \leftarrow \mathbf{y} + (-1)^b \mathbf{Sc}$
5) **Output(z,c)** with probability $1/(Mexp(-||Sc||^2)/2\sigma^2) cosh(\mathrm{h}z, Sci/\sigma^2))$ otherwise restart.

Verification:

The User would input message m, public key

$\mathbf{A} \in Z_{2q}^n$

1) if $\|\mathbf{Z}\| > B_2$ then reject.
2) if $\|\mathbf{Z}\|_\sigma \geq q/4$ then reject.
3) Accept iff $\mathbf{c} = H(\mathbf{Az} + q\mathbf{c} mod 2q, m)$.

Bindel et al. in their research analyzed BLISS, Ring Tesla and GLP signature algorithms based on several fault attacks [22]. The authors claimed to have found that either of the three signature schemes were vulnerable to atleast 9 of the 15 attacks they performed. Although Ring Tesla and GLP are out of our research scope, BLISS when attacked with first ordered fault attacks (randomization, skipping and Zeroing) was found to be vulnerable to 7 different fault attacks. The authors also provided reasonable countermeasures to defend against these attacks.

2.3 Hash based cryptography

In 1979 Ralph Merkle in his paper tited "Secrecy, Authentication and Public Key System" proposed the first hash based digital signature scheme [28]. Since then a lot of research has been done on it. These sort of scheme depends on the security of one way hash functions. Although this scheme has a particular disadvantage of producing a particular amount of signature at once, it still provide long term security against known algorithms in quantum computers [29]. Ralph Merkle in 1990 came up with Merkle signature scheme which could convert any one time signature into a multi-time one. Merkle used Lamport-diffie one-time signature [17]. The algorithms are as follows:

2.3.1 Merkle Signature

Signature generation:

1) Generate public key X_i and private key Y_i of 2^n one-time signatures.
2) For each public key X_i, with $1 \leq i \leq 2^n$ calculate a hash value $h_i = X_i = H(Y_i)$. With these hash values h_i build a hash tree.
3) Each node of the tree is represented as $a_{i,j}$, where i denotes the height of the node and j denotes the left-to-right position of the node.
4) In the Merkle Tree the hash values h_i are the leaves of a binary tree, so that $h_i = a0,i$.
5) Each inner node of the tree is the hash value of the concatenation of its two children.
6) A tree with 2^n leaves and $2^{n+1} - 1$ nodes is built. The root of the tree, $a_{n,0}$, is the public key pub of the Merkle Signature Scheme.

Signature

To sign a message M with the Merkle Signature Scheme:

1) The corresponding leaf of the hash tree to a one-time public key X_i is $a_{0,i} = H(X_i)$.
2) Identify the path in the hash tree from $a_{0,i}$ to the root A.
3) The path A would consist of n+1 nodes,

   $A_0,...A_n$, with $A_0 = a_{0,i}$ being the leaf and $A_n = a_{n,0} = pub$ being the root of the tree.
4) To compute the path A, every child of the nodes $A_1,...,A_n$ is needed.
5) To compute the next node A_{i+1} of the path A, identify both children of A_{i+1}. Therefore the brother node of A_i is required.

6) Identify $auth_i$, so that $A_{i+1} = H(A_i||auth_i)$.
7) Therefore, n nodes $auth_0,...,auth_{n-1}$ are needed, to compute every node of the path A.
8) Compute and save these nodes $auth_0,...,auth_{n-1}$.

These nodes, plus the one-time signature sig^0 of M is the signature $sig = (sig^0||auth_0||auth_1||...||auth_{n-1})$ of the Merkle Signature Scheme.

Verification:

1) The receiver knows the public key pub, the message M, and the signature $sig = (sig^0||auth_0||auth_1||...||auth_{n-1})$.
2) The receiver verifies the one-time signature sig^0 of the message M.
3) If sig^0 is a valid signature of M, the receiver computes $A_0 = H(X_i)$ by hashing the public key of the one-time signature.
4) For $j = 1,..,n-1$, the nodes of A_j of the path A are computed with $A_j = H(Aj-1||authj-1)$.
5) If A_n equals the public key pub of the merkle signature scheme, the signature is valid.

In 2005 Garcia presented a paper titled "On the security and the efficiency of the Merkle signature scheme" made a thorough analysis of Merkle signature [31]. They proved that Merkle signature is unforgeable under adaptive chosen message attack, they also claimed to provide an improved variant with forward security, unlimited keys and low power consumption. Buchmann et al. presented XMSS (eXtended Merkle Signature Signature) which used Winternitz one-time signature scheme's (WOTS) [25] collision-resilient version with the collision-resilient hash tree construction [26] and adds two different kinds of pseudorandom key generation. There are several other variants of merkle Signature scheme [36] [37].

2.3.2 Lamport Signature Lamport Signature is a one time signature scheme which uses secure cryptographic hash based function to create a digital signature [38]. Although a signature can be used to sign only one message, that range can be extended using merkle tree algorithm presented in the previous section. **key:**

1) Let k be a positive integer and let $P = \{0,1\}^k$ be the messages.
2) Let $f: Y \rightarrow Z$ be a one-way function.
3) For $1 \leq i \leq k$ and $j \in \{0,1\}$ the signer chooses $y_{i,j} \in Y$ randomly and computes $zi,j = f(yi,j)$.
4) The private key, K, consists of 2k values $y_{i,j}$. The public key consists of the 2 k values $z_{i,j}$.

Signature:

1) Let $m = m_1...m_k \in \{0,1\}^k$ be a message.
2) The signature of the message is $sig(m_1...m_k) = (y_{1,m1},...,y_{k,mk}) = (s_1,...,s_k)$

Verification:

The verifier validates a signature by checking that $f(s_i) = z_{i,mi}$ for all $1 \leq i \leq k$.

2.4 Code based cryptography

2.4.1 McEllice

In 1978 Robert McEllice developed an asymmetric encryption system which used randomization while encryption [32]. McEllice cryptosyste has several advantages and is a viable replacement for traditionally used cryptosystems. It is faster than most cryptosystems and uses a large matrices as its public and private keys. It is based on the NP hard problem of decoding linear codes.

Key generation:

1) Generate a binary (n,k)-linear code C having the capability of correcting t errors. This code must possess an efficient decoding algorithm and generates a $k \times n$ generator matrix G for the code C.
2) Choose a random $k \times k$ binary nonsingular matrix S.
3) Choose a random $n \times n$ permutation matrix P.
4) Calculate the $k \times n$ matrix $\hat{G} = SGP$.
5) The public key is $(G,\hat{t})$; private key is (S,G,P)

Message encryption:

1) Encode the message m as a binary string of length k. 2) Calculate the vector $c^0 = m\hat{G}$.
3) Select a random n-bit vector z containing exactly t ones (a vector of length n and weight t)'
4) The cipher-text can be computed as $c = c^0 + z$.

Message decryption:

1) Calculate the inverse of P (i.e. P^{-1}).
2) Calculate $\hat{c} = cP^{-1}$.
3) Decode $\hat{c}$ to $\hat{m}$ using the decoding algorithm for the code C.
4) Generate $m = m\hat{S}^{-1}$.

Bernstein et al. in 2008 extracted a plain-text from a cipher-text by decoding 50 errors in a [1024; 524] code [18]. It aslo provided with viable countermeasures.

2.4.2 Niederreiter cryptosystem:

Niederreiter cryptosystem is quite similar to McEllice cryptosystem, Although the primary difference is Niederreiter uses linear Goppa codes and the cipher text is a syndrome and the message is an error pattern.

Key generation:

1) Generate a binary (n, k)-linear Goppa code, G, with the capability of correcting t errors. This code possesses an efficient decoding algorithm.
2) Select a $(n-k)*n$ parity check matrix, H, for the code, G.
3) Generate a random $(n-k)*(n-k)$ binary non-singular matrix, S.
4) Generate a random $n * n$ permutation matrix, P.
5) Calculate the $(n-k) * n$ matrix, $H^{pub} = SHP$.
6) The public key would be (H^{pub},t); and the private key would be (S,H,P).

Message encryption:

1) Encode the message, m, as a binary string of length n and weight at most t. 2) Generate the cipher-text using $c = HpubmT$.

Message decryption:

1) Calculate $S^1c = HPm^T$.
2) recover Pm^T by applying syndrome decoding algorithm for G. 3) Generate the message, m, via $m^T = P^1Pm^T$.

Roberto et al. provided a survey with several possible side channel attacks on McEllice as well as Niedrreiter schemes along with their countermeasures[33].

3 CONCLUSION

In our survey we have analyzed several postquantum cryptography schemes and provided a comparison in Table 2. as per implemented by several authors. We can see from the comparison table that Lattice based cryptography schemes even when implemented in a constrained micro-controller shows good promise as they take the least amount of time for several operations as well as consume the least memory. Although, to definitively state that Lattice based cryptography is the best among post-quantum Cryptography further rigorous analysis has to be made. Future works would include rigorous implementation and analysis of post-quantum Cryptography algorithms in Software simulation as well as in constrained devices.

Table 2: A Theoretic Comparison of several post-quantum Cryptosystem

Properties	BLISS [34] (128-bit security)	Ring LWE[35] (more than 156-bit security)	NTRU(80-bit security)	Lamport	Lamport with Merkle (80- bit security)	Rainbow[39]	McEllice	Neidrreiter [40] 80-bit security
Public Key (KB)	7		2	∼10	0.08	132.7	500	∼74.032
Private Key (KB)	2		2	∼10	∼250	95.4	1000	∼ 4.096
Signature Size	7.680					6.32		
Signature Time	329 ms					257.1 ms		
Verification Time	88 ms					288.0 ms		
Encryption Time		68 ms						1.6 ms/op
Decryption Time		18.8 ms						180 ms/op
Possible attacks								
Platform	Atmel ATxmega -128A1	Atmel ATxmega -128A1	PC (not specified)	PC (not specified)	PC (not specified)	Atmel ATxmega -128A1	PC (not specified)	ATxMega256A1

REFERENCES

[1] Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computSIAM

[2] Ding, Jintai, and Dieter Schmidt. ”Rainbow, a new multivariable polynomial signature scheme.” International Conference on Applied Cryptography and Network Security. Springer Berlin Heidelberg, 2005.

[3] Alkim, Erdem, Lo Ducas, Thomas Pppelmann, and Peter Schwabe. ”Post-quantum Key Exchange-A New Hope.” In USENIX Security Symposium, pp. 327-343. 2016.

[4] Kipnis, Aviad, Jacques Patarin, and Louis Goubin. ”Unbalanced oil and vinegar signature schemes.” International Conference on the Theory and Applications of Cryptographic Techniques. Springer Berlin Heidelberg, 1999.

[5] Patarin, Jacques. ”Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms.” International Conference on the Theory and Applications of Cryptographic Techniques. Springer Berlin Heidelberg, 1996.

[6] Cheng, Chi, Rongxing Lu, Albrecht Petzoldt, and Tsuyoshi Takagi. ”Securing the Internet of Things in

a Quantum World." IEEE Communications Magazine 55, no. 2 (2017): 116-120.

[7] Takagi, Tsuyoshi, ed. Post-Quantum Cryptography: 7 th International Workshop, PQCrypto 2016, Fukuoka, Japan, February 24-26, 2016, Proceedings. Vol. 9606. Springer, 2016.

[8] Ajtai, Mikls. "Generating hard instances of lattice problems." Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. ACM, 1996.

[9] Gttert, Norman, et al. "On the design of hardware building blocks for modern lattice-based encryption schemes." International Workshop on Cryptographic Hardware and Embedded Systems. Springer Berlin Heidelberg, 2012.

[10] Ducas, Lo, et al. "Lattice signatures and bimodal gaussians." Advances in CryptologyCRYPTO 2013. Springer Berlin Heidelberg, 2013. 40-56.

[11] Howgrave-Graham, Nick. "A hybrid lattice-reduction and meet-in-the-middle attack against NTRU." Annual International Cryptology Conference. Springer Berlin Heidelberg, 2007.

[12] Park, Aesun, and Dong-Guk Han. "Chosen ciphertext Simple Power Analysis on software 8-bit implementation of ring-LWE encryption." Hardware-Oriented Security and Trust (AsianHOST), IEEE Asian. IEEE, 2016.

[13] Gneysu, Tim, Vadim Lyubashevsky, and Thomas Pppelmann. "Practical lattice-based cryptography: A signature scheme for embedded systems." International Workshop on Cryptographic Hardware and Embedded Systems. Springer Berlin Heidelberg, 2012.

[14] Roy, Sujoy Sinha, et al. "Compact ring-LWE cryptoprocessor." International Workshop on Cryptographic Hardware and Embedded Systems. Springer Berlin Heidelberg, 2014.

[15] De Clercq, Ruan, et al. "Efficient software implementation of ring-LWE encryption." Design, Automation and Test in Europe Conference and Exhibition (DATE), 2015. IEEE, 2015.

[16] Ralph C. Merkle. "Secrecy, authentication, and public key systems." (1979).

[17] Ralph C. Merkle. A certified digital signature. In G. Brassard, editor, Advances in Cryptology - CRYPTO89 LNCS, volume 435. Springer-Verlag Berlin Heidelberg 1990, 1990.

[18] Bernstein, Daniel J., Tanja Lange, and Christiane Peters. "Attacking and defending the McEliece cryptosystem." International Workshop on Post-Quantum Cryptography. Springer Berlin Heidelberg, 2008.

[19] Mohamed, Mohamed Saied Emam, and Albrecht Petzoldt. "The Shortest Signatures Ever." In Progress in CryptologyINDOCRYPT 2016: 17th International Conference on Cryptology in India, Kolkata, India, December 11-14, 2016 , Proceedings 17, pp. 61-77. Springer International Publishing, 2016.

[20] Hoffstein, Jeffrey, Jill Pipher, and Joseph Silverman. "NTRU: A ring-based public key cryptosystem." Algorithmic number theory (1998): 267-288.

[21] Coppersmith, Don, and Adi Shamir. "Lattice attacks on NTRU." International Conference on the Theory and Applications of Cryptographic Techniques. Springer Berlin Heidelberg, 1997.

[22] Bindel, Nina, Johannes Buchmann, and Juliane Krmer. "Lattice-based signature schemes and their sensitivity to fault attacks." Fault Diagnosis and Tolerance in Cryptography (FDTC), 2016 Workshop on. IEEE, 2016.

[23] Bernstein, Daniel J., et al. "NTRU Prime." IACR Cryptology ePrint Archive 2016 (2016): 461.

[24] Lyubashevsky, Vadim, Chris Peikert, and Oded Regev. "On ideal lattices and learning with errors over rings." Annual International Conference on the Theory and Applications of Cryptographic Techniques. Springer Berlin Heidelberg, 2010.

[25] Johannes Buchmann, Erik Dahmen, Sarah Ereth, Andreas Hlsing, and Markus Rckert. On the security of the Winternitz one-time signature scheme. In A. Nitaj and D. Pointcheval, editors, Africacrypt 2011, volume 6737 of LNCS, pages 363 378. Springer Berlin / Heidelberg, 2011. 2, 16

[26] Erik Dahmen, Katsuyuki Okeya, Tsuyoshi Takagi, and Camille Vuillaume. Digital signatures out of secondpreimage resistant hash functions. In Johannes Buchmann and Jintai Ding, editors, Post-Quantum Cryptography 2008, volume 5299 of LNCS, pages 109123. Springer, 2008. 2, 16, 19

[27] Howgrave-Graham, Nick, Joseph H. Silverman, and William Whyte. A Meet-in-the-Middle Attack on an NTRU Private key. Vol. 4. Technical report, NTRU Cryptosystems, June 2003. Report, 2003.

[28] Merkle, Ralph Charles, and Ralph Charles. "Secrecy, authentication, and public key systems." (1979).

[29] Daniel, A., and B. Lejla. Initial recommendations of longterm secure post-quantum systems. Technical report, 2015. [30] H. Kopka and P. W. Daly, *A Guide to L^{A}TEX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.

[31] Garca, LC Coronado. On the security and the efficiency of the Merkle signature scheme. Technical Report 2005/192, Cryptology ePrint Archive, 2005. Available at http://eprint. iacr. org/2005/192, 2005.

[32] McEliece, Robert J. "A public-key cryptosystem based on algebraic." Coding Thv 4244 (1978): 114-116.

[33] Avanzi, Roberto, et al. ”Side-channel attacks on the McEliece and Niederreiter public-key cryptosystems.” Journal of Cryptographic Engineering 1.4 (2011): 271-281.

[34] Oder, Tobias, Thomas Pppelmann, and Tim Gneysu. ”Beyond ECDSA and RSA: Lattice-based digital signatures on constrained devices.” Proceedings of the 51st Annual Design Automation Conference. ACM, 2014.

[35] Pppelmann, Thomas, Tobias Oder, and Tim Gneysu. ”High-performance ideal lattice-based cryptography on 8bit ATxmega microcontrollers.” International Conference on Cryptology and Information Security in Latin America. Springer International Publishing, 2015.

[36] Johannes Buchmann, Erik Dahmen, and Andreas Hlsing. XMSS - a practical forward secure signature scheme based on minimal security assumptions. In BoYin Yang, editor, Post-Quantum Cryptography 2011, volume 7071 of LNCS, pages 117129. Springer Berlin / Heidelberg, 2011. 2, 3, 16

[37] Hlsing, Andreas, Joost Rijneveld, and Fang Song. ”Mitigating multi-target attacks in hash-based signatures.” Public-Key CryptographyPKC 2016. Springer Berlin Heidelberg, 2016. 387-416.

[38] Lamport, Leslie. Constructing digital signatures from a one-way function. Vol. 238. Palo Alto: Technical Report CSL-98, SRI International, 1979.

[39] Czypek, Peter, Stefan Heyse, and Enrico Thomae. ”Efficient implementations of MQPKS on constrained devices.” International Workshop on Cryptographic Hardware and Embedded Systems. Springer Berlin Heidelberg, 2012.

[40] Heyse, Stefan. ”Low-reiter: Niederreiter encryption scheme for embedded microcontrollers.” International Workshop on Post-Quantum Cryptography. Springer Berlin Heidelberg, 2010.