A Like ELGAMAL Cryptosystem But Resistant To Post-Quantum Attacks

Ahmed EL-YAHYAOUI[1], Fouzia OMARY[1]

[1] Intelligent Processing & Security of Systems Team (IPSS)
Faculty of Sciences, Mohammed V University in Rabat, Morocco

Abstract: The Modulo 1 Factoring Problem (M1FP) is an elegant mathematical problem which could be exploited to design safe cryptographic protocols and encryption schemes that resist to post quantum attacks. The ELGAMAL encryption scheme is a well-known and efficient public key algorithm designed by Taher ELGAMAL from discrete logarithm problem. It is always highly used in Internet security and many other applications after a large number of years. However, the imminent arrival of quantum computing threatens the security of ELGAMAL cryptosystem and impose to cryptologists to prepare a resilient algorithm to quantum computer-based attacks. In this paper we will present a like-ELGAMAL cryptosystem based on the M1FP NP-hard problem. This encryption scheme is very simple but efficient and supposed to be resistant to post quantum attacks.

Keywords: ELGAMAL, cryptosystem, public key, cryptography, NP-hard, M1FP, post quantum attacks, one-way function**.**

1. Introduction

In 1976, Witfield Diffie and Marten Hellman invented the concept of public key cryptography [1], but without any concrete implementation. One year later, Rivest, Shamir and Adleman proposed a first realization, it was the RSA cryptosystem [2]. Their cryptosystem was protected by a patent until the year 2000 which poses a difficulty for its use. In 1985, Taher ELGAMAL described an ingenious public key cryptosystem [3]. The new algorithm was not patented as RSA, its security depends on the difficulty of solving Discrete Logarithm Problem, i.e. given integers q, g and $g^a \, mod \, q$, guess the smallest positive integer value a. ELGAMAL cryptosystem was improved by introducing its elliptic curve version by Koblitz [4] and its digital signature algorithm [5] by the National Security Agency NSA.

ELGAMAL encryption scheme consists of three algorithms: a key generation algorithm (KeyGen), an encryption algorithm (Enc) and a decryption algorithm (Dec). Suppose that we have two protagonists Alice and Bob wanting to communicate securely using an encryption scheme, they could use the ELGAMAL cryptosystem to secure their communications as follows:

ELGAMAL Key Generation Algorithm:

The first protagonist, Alice, generates a key pair as follows

- Generate an efficient description of a cyclic group G, of order q, with generator g. Let e represent the unit element of G.
- Choose, randomly, an integer x from the set $\{1,2,\dots,q-1\}$.
- Compute $h = g^x$
- The public key of Alice consists of the quadruplet (G, q, g, h). She publishes this public key and retains x as her private key, which must be kept secrete.

ELGAMAL Encryption Algorithm:

A second protagonist, Bob, will encrypt a message M to Alice under her public key (G, q, g, h) as follows:

- Map the message M to an element m of G using a reversible mapping function.
- Choose an integer y randomly from the set $\{1,2,\dots,q-1\}$.
- Compute $s = h^y$. This is called the shared secret.
- Compute $C_1 = g^y$
- Compute $C_2 = m.s$
- The ciphertext C of m consists of pair $C = (C_1, C_2)$.
- Bob sends C to Alice.

Note that if one knows both the ciphertext $C = (C_1, C_2)$ and the plaintext m, one can easily find the shared secret s, since $C_2.m^{-1} = s$. Therefore, a new y and hence a new s is generated for every message to improve security. For this reason, y is also called an ephemeral key.

ELGAMAL Decryption Algorithm:

Alice can decrypt the received ciphertext $C = (C_1, C_2)$using her private key x as follows:

- Compute $s = C_1^x$. Since $C_1 = g^y$, $C_1^x = g^{yx} = h^y$, and thus it is the same shared secret that was used by Bob in encryption.
- Compute s^{-1}, the inverse of s in the group G. This can be computed in one of several ways. If G is a subgroup of a multiplicative group of integers modulo n, where n is prime, the modular multiplicative inverse can be computed using the extended Euclidean algorithm. An alternative is to compute s^{-1} as C_1^{q-x}. This is the inverse of s because of Lagrange's theorem, since $s.C_1^{q-x} = g^{xy}.g^{(q-x)y} = (g^q)^y = e^y = e$.
- Compute $m = C_2.s^{-1}$. This calculation produces the original message m, because $C_2 = m.s$; hence $C_2.s^{-1} = (m.s).s^{-1} = m.e = m$.
- Map m back to the plaintext message M.

Practical use:

As in most asymmetric cryptosystems, the ELGAMAL public key encryption scheme is generally used as part of a hybrid encryption scheme. In such cryptosystems, the confidential message is practically encrypted by a symmetric algorithm, only the symmetric key is encrypted using the ELGAMAL cryptosystem, the ciphertext is the concatenation of the two resulted encrypted messages. We do like this, because of the slowness of asymmetric cryptosystems. The slowness of public key encryption schemes is 10 up to 100 times compared to symmetric encryption schemes. Consequently, it is faster and practical to use a symmetric cipher to encrypt the confidential message, which can be of arbitrary size, and then use ELGAMAL only to encrypt the secrete key of the symmetric cipher (called also session key), which has a small size compared to the size of the arbitrary message.

Security:

What about the security of the ELGAMAL encryption scheme? As well as any padding scheme used on the messages, the security is depending on the properties of the underlying cyclic group G. The encryption algorithm is supposed to be one-way, if the computational Diffie–Hellman assumption (CDH) is holding in the underlying cyclic group G. The ELGAMAL cryptosystem achieves semantic security, if the decisional Diffie–Hellman assumption (DDH) holds in the group G. The computational Diffie–Hellman assumption alone do not imply semantic security. The ELGAMAL encryption scheme is not secure under chosen ciphertext attack (CCA) because it is partially homomorphic (it is a multiplicatively homomorphic encryption scheme) which involves its malleability. For the partial homomorphy we give the example, given an encryption (C_1, C_2) of an unknown message m, we can easily perform some operations on the ciphertext and construct a valid encryption $(C_1, 2C_2)$ of the message $2m$ without any prior decryption. The malleability allows to use the ELGAMAL encryption scheme for the electronic vote for example. The CCA security could be achieved by a modification of the scheme like using a padding process with ELGAMAL scheme. The DDH assumption may or may not be necessary, depending on this modification.

However, there are variants of ELGAMAL encryption scheme that achieve security under chosen cipher attacks, such as the Cramer-Shoup [6] encryption scheme which can be considered as an extension of the ELGAMAL cryptosystem. It was the first cryptosystem [6] to combine the following three properties: it is resistant to chosen ciphertext adaptive attacks (IND-CCA2), it is proven secure in the standard model, and it is efficient.

Efficiency:

The same plaintext could be encrypted to many possible ciphertexts, using the same public key, under the cryptosystem ELGAMAL. Schemes which assure this property are called probabilistic. This property leads to a consequence that a general ELGAMAL encryption scheme produces a two components ciphertext with a double expansion in size from plaintext to ciphertext (1:2). Encrypting messages under ELGAMAL cryptosystem requires two exponentiations; nonetheless, these two exponentiations are independent from the message and can be precomputed in offline mode to speed up the encryption process. In the other hand, decryption requires just one exponentiation, but it requires also one computation of a group inverse, which can, anyhow, be simply combined into just one exponentiation.

2. Related Work

The ELGAMAL encryption scheme was used, without any problem, in the free GNU Privacy Guard software [7] recent versions of PGP [8] and other cryptosystems [9] until the apparition of Shor's algorithm [10]. It [10] is a polynomial-time quantum computer algorithm for integer factorization. Informally, it solves the following problem: Given an integer N, find its prime factors. It was discovered in 1994 by the American mathematician Peter Shor. This algorithm provides a threat against many cryptosystems like RSA, ELGAMAL encryption and its versions. In fact, Shor's algorithm just needs the appearance of a quantum computer to be a real threat for ELGAMAL cryptosystem, which is always possible today at any time. For that reason, many efforts have been provided [11], including the competition launched by the National Institute of Standards and Technologies (NIST) in this direction [11], to develop cryptosystems that resist to post quantum attacks.

Taking into consideration possible threats of quantum computing to actual cryptographic schemes, from now on classical mathematical problems like Factoring and Digital Logarithm problems are supposed to be obsoletes. Indeed, the next generation of cryptographic algorithms should be based on new mathematical problems quantum computer resilient. Five classes of candidate problems are known today [11]: code based, isogeny based, hash based, lattice based and multivariate system based cryptographic methods.

Code based candidates: uses the theory of error-coding codes to build public key encryption schemes. The first candidate in this category was early designed in 1978 [12]. We remark that is roughly the same age as RSA [2], but it hasn't take the same importance in deployment. It is due to its large key size which make unfavorable in comparison to RSA. McEliece's cryptosystem [12] was improved a plenty of times [13], [14], the last one is the Classic McEliece [15] candidate to NIST competition.

Isogeny based candidates: Isogeny-based cryptography uses maps between elliptic curves to build public key cryptography. The isogeny problem is to find an isogeny between two elliptic curves that are known to be isogenous. Isogeny-based cryptography is today a fertile branch of cryptography which can provide promising cryptographic candidates post quantum algorithms. The first publically accessible proposals in this domain date back to 2006 [16], [17]. The so called SIKE [18] algorithm is an isogeny based candidate submitted to NIST competition and it is in the third round as an alternate candidate.

Hash based candidates: hash based signatures (HBS) is a promising approach to construct post quantum cryptographic algorithms. The security of hash based signature schemes is well understood because it is dependent to famous security notions corresponding to hash functions, such as collision and pre-image resistance. Due to their importance, many hash based signature schemes [19], [20] has been already standardized by the National Institute of Standards en Technologies NIST [21].

Lattice based candidates: this is the most attractive and prolific area to design post-quantum cryptographic candidates. In fact, lattices provide more parameters than codes, which means that they might offer mathematical problems better adapted to a given situation, but also provide more attack surface. Among the well-known mathematical hard problems in lattice based cryptography, we find the Shortest Vector Problem (SVP), the Closest Vector Problem (CVP), Learning with Error Problem (LWE) and Ring Learning with Error problem (RLWE). These assumptions are widely spread as safe problems to design secure and practical lattice based cryptographic schemes that could resist to future attacks from quantum computing. In the literature we find many candidates, NTRU [22], NewHope [23], Crystals-Kyber

[24], GSW [25] that were designed from lattice underlying problems. Recently, 12 lattice based cryptographic schemes in all of the 26 candidates are in the second round of NIST's competition. Furthermore, lattice based candidates are the only type schemes contained in the three types of primitives required by NIST: public key encryption, digital signature and key exchange.

Multivariate system based candidates: Multivariate cryptography refers to public-key cryptography whose public keys represent a multivariate and quadratic polynomial system. The main computational problem underlying multivariate cryptography is finding a solution to a system of multivariate quadratic equations over finite fields. This problem is known to be NP-hard. Currently, the underlying schemes are not practical, they suffer from space complexity problems of large public keys and time complexity problems of their long decryption times. On the signatures front however, things look a bit better. Rainbow [26] and GeMSS [27] are two of the seven submitted schemes to NIST competition that have been selected to pass to the third round PQC process.

In this paper, we provide a quantum computing resilient public key encryption scheme like ELGAMAL design [3], but its security is not relied on the discrete logarithm assumption. We will design our cryptosystem from a new NP-Hard assumption. The underlying mathematical problem is the fresh M1FP candidate that has been recently introduced by Eric Jarpe [28].

The rest of this paper will be organized as follows: section 3 introduces the proposed encryption scheme. Starting with a mathematical background in the subsection A, passing through the description of our scheme in the subsection B and its correctness in the subsection C, and finally arriving to some properties of the provided cryptosystem in the subsection D. In section 4, we propose a toy example to lead the reader from the theoretical part to a practical case of our scheme that simplifies the comprehension of the suggested algorithm. Finally, the last section (section 5) concludes the paper.

3. The Proposed Encryption Scheme

3.1. Mathematical Background

- The set of integers is noted $\mathbb{Z}$.
- The set of real numbers is noted $\mathbb{R}$.
- A rational number is a real number that could be expressed as a quotient or fraction $\frac{p}{q}$ of two integers p and q.
- The set of rational numbers is noted $\mathbb{Q}$.
- An irrational number is a real number that is not rational, i.e. it could not be expressed as the ratio of two integers.
- The set of irrational numbers is noted $\mathbb{R}\backslash\mathbb{Q}$.

Definition 1: the integer part of a real number $x \in \mathbb{R}$ is $[x] = \max\{y \in \mathbb{Z} : y \leq x\}$.

Definition 2: For any real number $x \in \mathbb{R}$, the number $x - [x] < 1$ is the decimal part of x. It will be noted $x \bmod 1$, so:

$$x \bmod 1 = x - [x]$$

Theorem: for any real number x and integers a and b: $(ax \bmod 1)b \bmod 1 = (bx \bmod 1)a \bmod 1$.

Proof: interested reader can find the proof at [28] (theorem A3).

3.2. Modulo 1 Factoring Problem

- Given x and c, both in $\mathbb{R}\backslash\mathbb{Q}$

The problem of guessing $a \in \mathbb{Z}$ such that $c = ax \bmod 1$ is called the Modulo 1 Factoring Problem (M1FP).

This problem is proved to be NP-Hard [28].

3.3. The Proposed Scheme

We propose in this part a public key encryption scheme like ELGAMAL cryptosystem, but its security is not based on the discrete logarithm problem. The security of our scheme is based on the hardness of the modulo 1 factoring problem.

Suppose Alice and Bob want to communicate with our public key encryption scheme to exchange confidential message M.

Key generation:

The first step in the encryption scheme is to produce a pair of keys: the public key, and the secret key. The first will be used to encrypt the messages and the second to decrypt them.

- To generate her pair of keys, Alice will start by taking an irrational number $x \in \mathbb{R}\backslash\mathbb{Q}$.
- Alice will then draw an integer $a \in \mathbb{Z}$, which will be his private key sk, and will calculate $h = ax \bmod 1$.
- Finally, Alice will publish $pk = \{x, h\}$ as a public key.

Encryption:

Bob therefore has access to Alice's public key $pk = \{x, h\}$. To encrypt a plaintext M, of n digits, encoded in decimal basis, he proceeds as follows:

- Bob starts by choosing a random integer r
- Then he will compute $C_1 = rx \bmod 1$ and $R = rh \bmod 1$
- Bob takes, R_n, the first n digits of the decimal part of R.
- Then he computes $C_2 = M \oplus R_n$, such that $\oplus$ is a bitwise addition modulo 10.
- Finally, he sends to Alice the encrypted message $C = (C_1, C_2)$

Decryption:

Having access to $C = (C_1, C_2)$ and sk, Alice can decrypt the encrypted message C as follows:

- She computes: $R' = sk.C_1 mod\ 1$
- Then she takes, R'_n, the first n digits of the decimal part of R'.

Finally, Alice can obtain the original plaintext $M = C_2 \ominus R'_n$, such that $\ominus$ is a bitwise subtraction modulo 10.

3.4. Correctness of The Scheme

In this part we will proof that the decryption step is correct and really gives the original message.

i.e we should proof that $M = C_2 \ominus R'_n$

we have $C_2 = M \oplus R_n$

$$\Rightarrow C_2 \ominus R'_n = M \oplus R_n \ominus R'_n$$
$$\Rightarrow C_2 \ominus R'_n = M \oplus (R_n \ominus R'_n)$$

We have also, $R' = sk.C_1 mod\ 1$
$$= sk.r.x \bmod 1$$
$$= r.sk.x \bmod 1$$
$$= rh \bmod 1 = R$$

Consequently, $R_n = R'_n$ by definition.

Then, $C_2 \ominus R'_n = M \oplus (R_n \ominus R'_n) = M \oplus 0 = M$ □

3.5. Some Properties of the Proposed Scheme

Probabilistic scheme: this scheme is probabilistic. Due to the random integer r used in the encryption algorithm, the same message, when encrypted several times, will yield different ciphertexts.

Security of the scheme: to inverse our cryptosystem, the attacker should know the afore defined R'_n. To find R'_n, he should compute R' given the public parameters of the crytosystem. But to compute $R' = sk.C_1 mod\ 1$ the attacker don't have access to the secret key sk, so he should resolve the Modulo 1 Factoring Problem, which is proved to be NP-hard [28]. This problem seems to be resilient to quantum computing attackers [28]. Consequently, our proposed encryption scheme is secure under the hardness of M1FP and even with the presence of a quantum computer i.e. its security is relied to the difficulty resolving of the Modulo 1 Factoring Problem.

Additive homomorphism: the proposed scheme is additively homomorphic [29]. This is a very useful property in electronic voting. In fact, if we add two ciphertexts, the obtained ciphertext will be the encrypted result of the addition of the two underlying original plaintexts. i.e. $C \oplus C' = E(M) \oplus E(M') = E(M \oplus M')$.

The addition could be performed as follows:

$$\begin{aligned} & C \oplus C' \\ & = (C_1 \oplus C'_1, C_2 \oplus C'_2) \\ & = \left(rx\ mod\ 1 \oplus r'x\ mod\ 1, (M \oplus R_n) \oplus (M' \oplus R'_n)\right) \\ & = ((r \oplus r')x\ mod\ 1, (M \oplus M') \oplus (R_n \oplus R'_n)) \end{aligned}$$

4. A Toy Example

In this section we will provide a practical example to illustrate how to use the new scheme.

Suppose that Alice will generate a pair of keys (sk, pk) and shares the public key pk with Bob, who will encrypt a secret message $M =$ "$I\ am\ Bob$" and sends it to Alice.

Alice's Key Generation:

Alice will choose:

$$\begin{aligned} x &= \ln(5)\ mod\ 1 \\ &= 0.609437912434100374600759333222619\ ... \end{aligned}$$

$a = 5940941723$, and will calculate:

$$\begin{aligned} & h = ax\ mod\ 1 \\ & = 9561576844.557767403435580590245118343884 53..mod\ 1 \\ & = 0.55776740343558059024511834388453\ ...mod\ 1 \end{aligned}$$

the key pair of Alice is $(sk, pk) = (a, \{x, h\})$.

Alice will share with Bob $pk = \{x, h\}$.

Encryption of Bob's message:

Suppose Bob has a message $M =$ "$I'm\ Bob$" and wants to encrypt it, with Alice's public key, before sending it to Alice.

As a first step Bob will encode it to ASCII, so:

$M = 073\ 039\ 109\ 032\ 066\ 111\ 098$. The message M contains 21 digits, so $n = 21$.

Next, he will choose a random number $r = 8710936522$ and compute:

$$\begin{aligned} & C_1 = rx\ mod\ 1 \\ & = 0.31364287132363564473236599596853\ ...mod\ 1 \end{aligned}$$

And

$$\begin{aligned} & R = rh\ mod\ 1 \\ & = 0.36810723784051831395590772780466\ ...mod\ 1 \end{aligned}$$

Consequently, the first 21 digits of the decimal part of R are: $R_n = 368107237840518313955$.

Then, Bob will encrypt the message M as:

$$\begin{aligned} C_2 &= M \oplus R_n \\ &= 073039109032066111098 \\ &\oplus 368107237840518313955 \\ &= 331136336872574424943 \end{aligned}$$

Finally, Bob will send to Alice his encrypted message:

$$C = \begin{pmatrix} 0.31364287132363564473236599596853\ ...mod\ 1 \\ 331136336872574424943 \end{pmatrix}$$

Decryption at Alice's side:

At the reception, Alice will take C_1 from C and compute:

$$\begin{aligned} & R' = sk.C_1 mod\ 1 \\ & = 0{,}36810723784051831395588967197719\ ...mod\ 1 \end{aligned}$$

Next, she will take the first 21 digits of the decimal part of R' are: $R'_n = 368107237840518313955$.

Then, Alice will obtain the message M in ASCII's encoding format as follows:

$$\begin{aligned} M &= C_2 \ominus R'_n \\ &= 331136336872574424943 \\ &\ominus 368107237840518313955 \\ &= 073039109032066111098 \end{aligned}$$

Finally, Alice will decode the ASCII message to a text and obtain the plaintext originally encrypted by Bob: $M =$ "$I'm\ Bob$".

5. Conclusion

In this paper, we proposed a probabilistic like ELGAMAL public key encryption scheme. The security of our cryptosystem is based on the hardness of resolving the modulo 1 factoring problem (M1FP). The M1FP is proved to be NP-hard. Our cryptosystem is resilient to post quantum attacks unlike the classical ELGAMAL cryptosystem for which the security is based on the discrete logarithm problem.

In the case of discrete logarithm problem on finite fields, the number of secrets in the field is limited by the order of the finite field q. Therefore, using Grover's algorithm, we can easily drive a cryptanalysis of the cryptosystem in the presence of a quantum computer. However, this cryptanalysis is not applied to our cryptosystem due to the infinity of the decimal part of a transcendental number.

This work is a gateway to the use of the promising M1FP problem in the construction of post-quantum encryption schemes. In a future work, we think construct the underling signature scheme to our cryptosystem.

References

[1] W. Diffie and M. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory,* vol. 22, no. 6, p. 644–654, 1976.

[2] R. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM,* vol. 21, no. 2, pp. 120-126, 1976.

[3] T. ELGAMAL, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *IEEE TRANSACTIONS ON INFORMATION THEORY,* vol. 31, no. 04, pp. 469-472, 1985.

[4] N. Koblitz, "Elliptic Curve Cryptosystems," *MATHEMATICS OF COMPUTATION,* vol. 24, no. 177, pp. 20.1-209, 1987.

[5] D. Pointcheval and J. Stern, "Security Arguments for Digital Signatures and Blind Signatures," *J. of Cryptology*, vol. 13, p. 361–396, 2000.

[6] Cramer R., Shoup V. (1998) A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: Krawczyk H. (eds) Advances in Cryptology — CRYPTO '98. CRYPTO 1998. Lecture Notes in Computer Science, vol 1462. Springer, Berlin, Heidelberg. https://doi.org/10.1007/BFb0055717

[7] J. Callas, L. Donnerhacke, H. Finney and R. Thayer, "OpenPGPmessage Format," 11 1998. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc2440. [Accessed 16 09 2021].

[8] J. Callas, L. Donnerhacke, H. Finney, D. Shaw and R. Thayer, "OpenPGP Message Format," 11 2007. [Online]. Available: https://datatracker.ietf.org/doc/html/rfc4880. [Accessed 16 09 2021].

[9] J. Jonathan Katz and Y. Lindell, Introduction to Modern Cryptography, 2nd Edition, Boca Raton, 2014.

[10] P. Shor, "Polynomial-time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer," *SIAM J. Comput*, vol. 25, p. 1484–1509, 1997.

[11] W. Beullens, J. D'Anvers, A. Hülsing, T. Lange, L. Panny, C. de Saint Guilhem and N. Smart, "POST-QUANTUM CRYPTOGRAPHY Current state and quantum mitigation," European Union Agency for Cybersecurity, 2021.

[12] R. McEliece, "public-key cryptosystem based on algebraic coding theory," Technical report, NASA, 1978. https://tmo.jpl.nasa.gov/progress_report2/42-44/44N.PDF, 1978.

[13] H. Niederreiter, "Knapsack-type cryptosystems and algebraic coding theory," *Problems of Control and Information Theory*, vol. 15, no. 2, p. :159–166, 1986.

[14] D. Bernstein, T. Chou and P. Schwabe, "McBits: Fast constant-time codebased cryptography," in *15th International Workshop*, Santa Barbara, CA, USA, August 20-23, 2013.

[15] M. Albrecht, D. Bernstein, T. Chou, C. Cid, J. Gilcher, T. Lange, V. Maram, I. Maurich, R. Misoczki, R. Niederhagen, K. Paterson, E. Persichetti, C. Peters, P. Schwabe, N. Sendrier, J. Szefer, C. Tjhai, M. Tomlinson and W. Wang, "Classic McEliece," 30 10 2020. [Online]. Available: https://classic.mceliece.org/index.html. [Accessed 16 09 2021].

[16] J. Couveignes, "Hard Homogeneous Spaces," IACR Cryptology ePrint Archive 2006/291., 2006.

[17] A. Rostovtsev and A. Stolbunov, "PUBLIC-KEY CRYPTOSYSTEM BASED ON ISOGENIES," IACR Cryptology ePrint Archive 2006/145., 2006.

[18] D. Jao, R. Azarderakhsh, M. Campagna, C. Costello, L. Feo, B. Hess, A. Hutchinson, A. Jalali, K. Karabina, B. Koziel, B. LaMacchia, P. Longa, M. Naehrig, G. Pereira, J. Renes, V. Soukharev and D. Urbanik, "SIKE – Supersingular Isogeny Key Encapsulation," SIKE.ORG, 09 06 2021. [Online]. Available: https://sike.org/. [Accessed 16 09 2021].

[19] J. Buchmann, E. Dahmen and A. Hulsing, "Xmss-a practical forward secure signature scheme based on minimal security assumptions," in *In International Workshop on Post-Quantum Cryptography*, Taipei, Taiwan, 2011.

[20] F. Leighton and S. Micali, "Large provably fast and secure digital signature schemes based on secure hash functions". USA Patent 5,432,852, 07 1995.

[21] D. Cooper, D. Apon, Q. Dang, M. Davidson, M. Dworkin and C. Miller, "NIST Special Publication 800-208: Recommendation for Stateful Hash-Based Signature Schemes," National Institute of Standards and Technology, 2020.

[22] J. Hoffshtein, J. Pipher and J. Silverman, "NTRU: A ring-based public key cryptosystem," in *Algorithmic Number Theory, Third International Symposium*, ANTS-III, Portland, Oregon, USA, 1998.

[23] E. Alkim, L. Ducas, T. Poppelmann and P. Schwabe, "Newhope without reconciliation," IACR Cryptology ePrint, 2016:1157, 2016.

[24] J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. Schanck, P. Schwabe, G. Seiler and D. Stehlé, "Crystals-kyber: a cca-secure module-lattice-based kem," IACR Cryptology ePrint Archive, 2017:634, 2017.

[25] C. Gentry, A. Sahai and B. Waters, "Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based," IACR Cryptology ePrint Archive, 2013:340, 2013.

[26] J. Ding, M. Chen, A. Petzoldt, D. Schmidt, B. Yang, M. Kannwischer and J. Patarin, "Post-Quantum Cryptography PQC," Technical report, National Institute of Standards and Technology, 2019.

[27] A. Casanova, G. Faugère, G. Macario-Rat, J. Patarin, L. Perret and J. Ryckeghem, "Post-Quantum Cryptography PQC," Technical report, National Institute of Standards and Technology, 2019.

[28] E. Jarpe, "An Alternative Diffie-Hellman Protocol," *Cryptology*, vol. 4, no. 5, 2020.

[29] Baharon, M., Shi, Q., Abdollah, M., S.M.M YASSIN, S., & Idris, A. (2018). An Improved Fully Homomorphic Encryption Scheme for Cloud Computing. *International Journal of Communication Networks and Information Security (IJCNIS)*, *10*(3). doi:https://doi.org/10.54039/ijcnis.v10i3.3573