2021 13th International Conference on Electronics, Computers and Artificial Intelligence (ECAI) | 978-1-6654-2534-6/21/$31.00 ©2021 IEEE | DOI: 10.1109/ECAI52376.2021.9515012

Towards Quantum Resistant Key Agreement Schemes Using Unpredictability

Alaa Elhao
college of computing and information technology
Arab Academy for Science and Technology and
Maritime Transport,
Cairo, Egypt.
alaamfawzi@gmail.com

Mohamed Helmy Megahed
college of computing and information technology
Arab Academy for Science and Technology and
Maritime Transport,
Cairo, Egypt.
mmhmegahed@gmail.com

Emad A Elsamahy
college of computing and information technology
Arab Academy for Science and Technology and
Maritime Transport,
Cairo, Egypt.
e.elsamahy@aast.edu

***Abstract*—Elliptic curve Diffie Hellman (ECDH) is one of today's most commonly used key agreement schemes, gaining universal usage amongst secure communication protocols such as transport layer security (TLS) and secure shell (SSH). This popularity is attributed to elliptic curve cryptography's (ECC) known benefits including offering high-security levels for lesser key sizes in comparison to other known public key counterparts. With public key schemes being under the threat of becoming obsolete due to quantum computing and associated algorithms, we propose an enhancement to ECDH aiming at exponentially increasing the hardness of exhaustive search methods utilizing quantum computing powers. In order to reach a key agreement scheme that would withstand future and post-quantum processing powers, we introduce an enhancement that will be based on a distinct cryptographic property labeled 'Unpredictability', which offers algorithms the ability to use multiple key pairs (up to 256), in different combinations, all whilst maintaining substantially similar arithmetic operations. The resultant scheme is labelled as unpredictable elliptic curve Diffie Hellman (UP-ECDH)**

***Index Terms*—elliptic curve cryptography, elliptic curve diffie hellman, key agreement, post-quantum cryptography, quantum computers, quantum-resistant, unpredictability**

I. Introduction

Elliptic curve Diffie Hellman (ECDH) is a public key agreement scheme, where two parties can exchange public parameters over an unsecured channel, leading to a joint establishment of a shared secret. It is a modified form of the classical Diffie Hellman protocol [1], which uses elliptic-curve arithmetic, instead of the classical integers modulo p multiplicative groups. The most common usage of this shared secret agreement, is to derive another key of symmetric nature, where the derived symmetric key can then be used to encrypt subsequent communications.

Several classical cryptographic applications were adjusted to be based on the elliptic curve discrete logarithm problem (ECDLP), in order to realize the known gains of elliptic curve cryptography (ECC) [2], [3], amongst them are ECDH and elliptic curve digital signature algorithm (ECDSA) [4]. The security of these ECC based schemes rely on the proven hardness of computing discrete logarithms in finite fields or in groups of points on an elliptic curve.

In Peter Shor's paper "Algorithms for Quantum Computation: Discrete Logarithms and Factoring" [5], it was shown that calculating the unknown "exponent" in the discrete logarithm problem as well as factorizing large integers would change with a quantum computer. On the other side of symmetric algorithms, research has shown that an exponential acceleration of search algorithms is impossible, suggesting that symmetric algorithms and hash functions should be usable in a quantum era [6] by effectively doubling the key length.

With current advances in quantum physics, many organizations are now developing more powerful quantum computing systems, some of which provide public accessibility to their systems, encouraging a community of quantum research. This speed of development comes with a promise of unparalleled processing powers, that would possibly lead to a repertoire of scientific breakthroughs. However, such processing power also poses a major threat to the confidentiality of modern communication systems, if used by the wrong hands. Current public key cryptosystems, including but not limit to ECC, are now considered to be obsolete in a future quantum era, a threat which will inevitably have severe and perilous consequences for the whole gamut of our communication systems and all that they increasingly impact.

According to NIST, quantum computers will bring an end to the current public key encryption schemes [7]. Table I adapted from NIST shows the impact of quantum computing on present cryptographic schemes.

In this paper, we propose a practical solution to enhance the security of ECDH by introducing a cryptographic property labeled 'Unpredictability', along with combining the exchange with a symmetric encryption based key distribution center

TABLE I
QUANTUM COMPUTING IMPACT ON CURRENT CRYPTOGRAPHIC SCHEMES.

Cryptographic Algorithm	Type	Purpose	Impact
AES-256	Symmetric key	Encryption	Secure
SHA-256, SHA-3	-	Hash functions	Secure
RSA	Public key	Signatures, key establishment	No longer secure
ECDSA, ECDH (Elliptic Curve Cryptography)	Public key	Signatures, key exchange	No longer secure
DSA (Finite Field Cryptography)	Public key	Signatures, key exchange	No longer secure

(KDC). All of which contribute towards building a secure key exchange scheme candidate with the following properties:

- A mathematically infeasible number of trails for brute force attacks
- A practically secure key exchange scheme with low performance overhead

Even though quantum computers are still being developed and do not pose any immediate threat, several cryptographic families have been introduced to be quantum resistant, with a particular focus on public key cryptosystems. These schemes do not necessarily serve as a replacement for today's current schemes, due to challenges faced, most notably, large key sizes, lacking standardization and proven security.

Examples of most researched post-quantum (PQ) families and their cryptosystems are the following:

1) *Code-based cryptosystems:* First introduced by McEliece in 1978. The McEliece cryptosystem [8] is built on Goppa codes and its mathematical hard problem is based on the syndrome decoding problem [9]. The McEliece scheme is currently considered quantum resistant, while performance is considered quite fast, most code-based primitives suffer from having very large key sizes [10], making them inappropriate for general use in internet protocols. Classical McEliece has been chosen as a finalist in the third round of post-quantum NIST standardization [11].

2) *Lattice-based cryptosystems:* One of most prominent PQ primitives, lattice-based cryptography has seen a huge focus on development. The main mathematical problem shared by most lattice problems is shortest vector path (SVP) which is considered difficult to solve even with current quantum algorithm improvement [12]. However, it is difficult to prove security against known classical cryptanalysis techniques. It is worth mentioning that 5 out of 7 finalist primitives chosen by NIST's standardization are lattice-based [11]

3) *Hash-based cryptosystems:* Hash-based signatures are digital signatures constructed using hash functions. Their security is well-understood and tested. However, up until now, they only offer digital signature schemes. Most notable Hash-based signatures; SPHINCS+ [13] is considered a promising candidate for the post-quantum era. Meanwhile, other lattice-based signatures like; CRYSTALS-DILITHIUM [14] and FALCON [15] have surpassed it as round three finalists in post-quantum NIST standardization [11]

4) *Multivariate cryptosystems:* Multivariate schemes are based on the hard problem of solving systems of multivariate polynomials over finite fields. The main drawback of multivariate schemes is the large size of the (public) keys. The public key size of a multivariate public key cryptosystem is typically about 10 to 100 kB and therefore much larger than that of classical schemes such as RSA and lattice-based cryptosystems. [16]

Most algorithms proposed based on the above schemes are known to be fast. The biggest challenges of these cryptosystems, which prevents their practical adoption is the extremely large key sizes that are required, and difficulty in proving security effectiveness on currently available computing resources.

While the quantum threat is closing in each day, the current state of quantum computers is not capable of bringing down current public key encryption schemes, we argue whether a provable and effective quantum resistant system, will be in widespread use and standardization before the threat is realized.

Seeking immediate answers, we propose our enhancement of current mainstream key exchange scheme, ECDH, to tackle problems that are faced by post-quantum candidates, so as to achieve a practical solution.

A. Paper Outline

The rest of this paper is organized as follows; section II explains the primitives of ECC, ECDH, KDCs and unpredictability. In section III, our proposed model is discussed. Section IV will look at the security proof of our enhanced scheme. Finally, conclusions and suggestions for future work are presented on section V.

II. PRELIMINARIES

This section details the cryptography notations and fundamentals required to grasp ECC. As section II.A introduces elliptic curve cryptography, followed by section II.B presenting ECDH and in section II.C KDC services are briefly discussed. Finally, in section II.D, the concept on unpredictability is explained.

A. Elliptic Curve Cryptography

Elliptic curve cryptosystems offer the potential to provide relatively small block size, high security public key schemes that can be efficiently implemented [17]. ECC is based mainly on finite fields, the following equation describes its mathematical primitives. Let E/F_p denote an elliptic curve, E, over a prime finite field F_p, in (1)

$$y^2 = x^3 + ax + b, \forall a, b \in F_p, \quad (1)$$

and with the discriminant defined by (2), required to avoid singular points

$$\Delta = 4a^3 + 27b^2 \neq 0 \tag{2}$$

The points on the curve E/F_p along with an identity element O, called the 'point at infinity', form a group $G = (x, y) : x, y \in F_p, E(x, y) = 0 \cup O$. G is a cyclic additive group in the point addition "+" defined as follows: Let $P, Q \in G$, draw a line containing P and Q (tangent line to E/F_p if $P = Q$), at third point of intersection with E/F_p, mirror the point along the x-axis, resulting in R as a result of point addition of P, Q in both cases. Scalar multiplication is defined by point addition such that $nP = P + P + \cdot\cdot\cdot + P(n\ times)$, while EC subtraction is another form of EC addition, where the inverse of a point to be subtracted is added in the usual formula for point addition.

B. Elliptic Curve Diffie Hellman

In the same fashion as normal Diffie Hellman key exchange, ECDH follows the same structure albeit using elliptic curve public/private key pairs. Let q denote either a large prime integer or an integer of the form 2^m, and elliptic curve parameters a and b for (1). This defines the elliptic group of points $E_p(a, b)$, along with base point $G = (x_1, y_1)$ in $E_p(a, b)$ with order n of a point G on an elliptic curve such that $nG = 0$. With all parameters of the cryptosystem known to all participants. A key exchange between users Alice and Bob can be accomplished as follows:

1. Alice selects private key = $n_A < n$, $n_A \subset \mathbb{Z}^+$. Alice then generates public key $P_A = n_A * G \in E_p(a, b)$
2. Bob selects private key n_B and computes public key P_B
3. Alice generates secret key $k = n_A * P_B$ and Bob generates secret key $k = n_B * P_A$.

The two calculations result in the same value as shown in (3)

$$n_A * P_B = n_A * (n_B * G) = n_B * (n_A * G) = n_B * P_A \tag{3}$$

Breaking this scheme would require an attacker to compute k given G and kG, which is assumed to be hard.

C. KDC services

Key distribution centers offer the ability to distribute keys across multiple users in a centralized fashion, mostly using symmetric encryption. They are aimed at reducing the native risk of exchanging keys amongst several parties, as well as making the process more manageable from an administrative standpoint.

Before public key schemes were discovered, centralized symmetric key distribution was the method of choice in multi-node networks. Till this day, key distribution is still widely used in environments with a large scale of nodes (users/servers). As KDCs rely on symmetric encryption, this makes them an interesting candidate of a post-quantum cryptosystem.

The most widely used key distribution service (Kerberos) originated from MIT labs, it provides a central authentication server that acts somewhat as a single logon source for user to server, and server to user authentication. The discussion of Kerberos and key distribution involves several paradigms beyond this brief overview, further information can be found in Neuman and Ts'o, 1994 [18]

D. Unpredictability

The concept of unpredictability will serve as the base on which our modification is built upon. Part of its properties will be applied for use with UP-ECDH.

The concept of unpredictability was first introduced by Helmy *et al* [19], it incorporates three main primitives aiming to increase the hardness of cryptographic operations:

1- The usage of multiple keys instead of one key - in order to increase the number of brute force attempts by adversaries
2- Randomness of used encryption algorithm
3- XOR operations on output.

These primitives were included on Spread Spectrum Encryption Architecture (SSEA) encryption architecture in order to increase the security level exponentially. Our model takes a slightly different approach, using the same randomness operation for keys, without adding algorithm randomness, Where the unpredictability function is designed into the already existing elliptic curve key exchange scheme, ECDH.

III. Proposed model

UP-ECDH is designed with the aims of increasing ECDH hardness so as to be resistant to the near quantum computer threat, while maintaining familiarity and performance. This enhancement seeks to make the scheme exponentially harder for current attack methods as well as future quantum computing attacks using Shor's algorithm.

The unpredictability functions used include two objectives:

1- Generating a pool of 256 key pairs per communicating party, thus increasing the number of adversarial brute force trials
2- Utilizing a subset from generated key pool, by randomly choosing keys, in a random sequence set for session key derivation. sequence set example shown in (4-5)

$$Seq = \{1, 99, 7, 204\} = \{P_{A1}, P_{A99}, P_{A7}, P_{A204}\} \tag{4}$$

$$Seq = \{5, 84, 36, 250, 18\} = \{P_{A5}, P_{A84}, P_{A36}, P_{A250}, P_{A18}\} \tag{5}$$

With the generation of a key pool containing 256 public/private key pairs, and then using a random sequences set from the pool, an adversary will not only have to compute 256 keys instead of one, but will also be unable to predict which key pair(s) are being used in the sequence set. This would effectively increase the hardness of brute force attempts significantly.

Classical mathematical operations of ECDH, as well as the negotiation flow between sender and recipient will be affected by introducing unpredictability. In order to maintain correctness, classical ECDH (section III.B) functions were adjusted to account for this new enhancement.

A. Proposed UP-ECDH

Having the unpredictability concept serving as the base for our new scheme, adjustments are applied to incorporate the mathematical structure requirements to reach the same session key via an exchange of public parameters i.e. correctness of the algorithm.

First, each communicating party will generate 256 public/private key pairs, to serve the goal of unpredictability. Second, a random sequence set of numbers will be generated and exchange by a KDC (between 2-256) e.g. $Seq_{AB} = \{4, 76, 11, 85, 232\}$ utilizing symmetric encryption (not prone to quantum attacks) with all parties, this sequence set will determine which keys will be used by both parties from their corresponding private/public key pool on subsequent computations.

Our new protocol flow will be the following:

Algorithm 1: UP-ECDH

1: Public parameters : $\{p, G, Certificate_{KDC,A,B}\}$
A trusted party chooses and publishes a (large) prime p and an integer G having large prime order in F_p, along with public certificates for Alice, Bob and KDC
2: Private parameters : $\{n_{oa,ob}\}$
Each sender and recipient has an offline key n_{ox} shared with KDC to establish symmetric encryption
3: A → KDC : {Send ($Certificate_A, ID_B$), Request Sequence set (Seq_{AB})}
Utilizing the symmetrically encrypted channel, Alice requests key sequence set from KDC, and signs the request with her own certificate. By sending Bob's ID, KDC will be able to align the same sequence set sent to Bob later in the flow
4: KDC → A : {Response (Seq_{AB})}
KDC sends a random sequence set Seq_{AB} to Alice
5: B → KDC : {Send ($Certificate_B, ID_A$), Request sequence set (Seq_{AB})}
Bob requests sequence set from KDC, and signs the request with his own certificate. Sending Alice's ID, to receive same sequence set Seq_{AB} sent to Alice by KDC
6: KDC → B : {Response (Seq_{AB})}
KDC shares the same random sequence set Seq_{AB} with Bob
7: A: {Selects ($n_{a1} \dots n_{a256}$), Computes ($N_{a1} \dots N_{a256}$)}
Alice chooses 256 private keys and then computes 256 public keys by multiplying each private key n_{ax} by G
8: B: {Selects ($n_{b1} \dots n_{b256}$), Computes ($N_{b1} \dots N_{b256}$)}
Bob chooses 256 private keys and then computes 256 public keys by multiplying each private key n_{bx} by G
9: A → B: $\{N_{a1} \dots N_{a256}\}$
Alice will exchange public keys with Bob
10: B → A: $\{N_{b1} \dots N_{b256}\}$
Bob will exchange public keys with Alice
11: A, B: Compute $\{S_{ab}\}$
Alice will compute a session key (S_{ab}) by adding all private keys according to sequence set Seq_{AB} e.g. $Seq_{AB} = \{7, 222, 5, 11, 32, 157\}$ by Bob's first public key (P_{B1}) (6)

$$(n_{A7} + n_{A222} + n_{A5} + n_{A11} + n_{A32} + n_{A157}) * P_{B1} \quad (6)$$

Bob will compute a session key by adding all of Alice's public keys according to $Seq_{AB} = \{7, 222, 5, 11, 32, 157\}$, multiplied by his first private key (n_{B1}) (7)

$$(P_{A7} + P_{A222} + P_{A5} + P_{A11} + P_{A32} + P_{A157}) * n_{B1} \quad (7)$$

Both equations (6 and 7) will be reduced to:

$$\begin{aligned} S_{ab} = {} & n_{a7} * n_{b1} * G + n_{a222} * n_{b1} * G + n_{a5} * n_{b1} * G \\ & + n_{a11} * n_{b1} * G + n_{a32} * n_{b1} * G + n_{a157} * n_{b1} * G \quad (8) \end{aligned}$$

Fig. 1 shows UP-ECDH exchange.

In our example, a sequence set of order 6 is chosen for the sake of simplicity. Our model is designed to have up to 256 keys (or more), thus being flexible to increase its security effectiveness exponentially - by increasing the order per sequence set - depending on the security needs, and the level of advancements of quantum computers.

IV. Security analysis

The proposed key management scheme UP-ECDH achieves the correctness and secrecy requirements necessary to provide a resistant key exchange scheme based on what is referred to as ECDLP. In this section, a security analysis of brute force attacks for the proposed scheme is presented.

A. Correctness

Correctness of the algorithm has been proven on section III.B. Equations 6, 7 and 8 show how both parties are able to reach the same session key (S_{ab}) after exchanging public parameters and securely exchanging the sequence set (Seq_{ab}) with the KDC.

B. Secrecy

Security features for the UP-ECDH are inherited from classical ECDLP, where in order to break this scheme, an attacker would need to compute private key k given G and kG, which is assumed to be hard according to ECDLP. This hardness increases proportionally with the sequence set chosen by both parties, which fulfil the same notion where k is an integer chosen to be smaller than order n of point G where nG = 0.

Definition IV.1 (Brute Force attack)**.** The adversary systematically makes trials on all possible keys until the correct one is found.

- An attacker need to compute 256 k given G and 256 kG resulting in a number of trials dependent on the key size used for each key. With a minimum key length of 256-bits in our scheme, this results in 2^{256} trials for adversary to be able to compute each key, resulting in a total $\sum_1^T 2^{256}$ where T is the order of sequence set Seq
- An adversary will need to compute all possible combinations used on the sequence set. The formula is generally

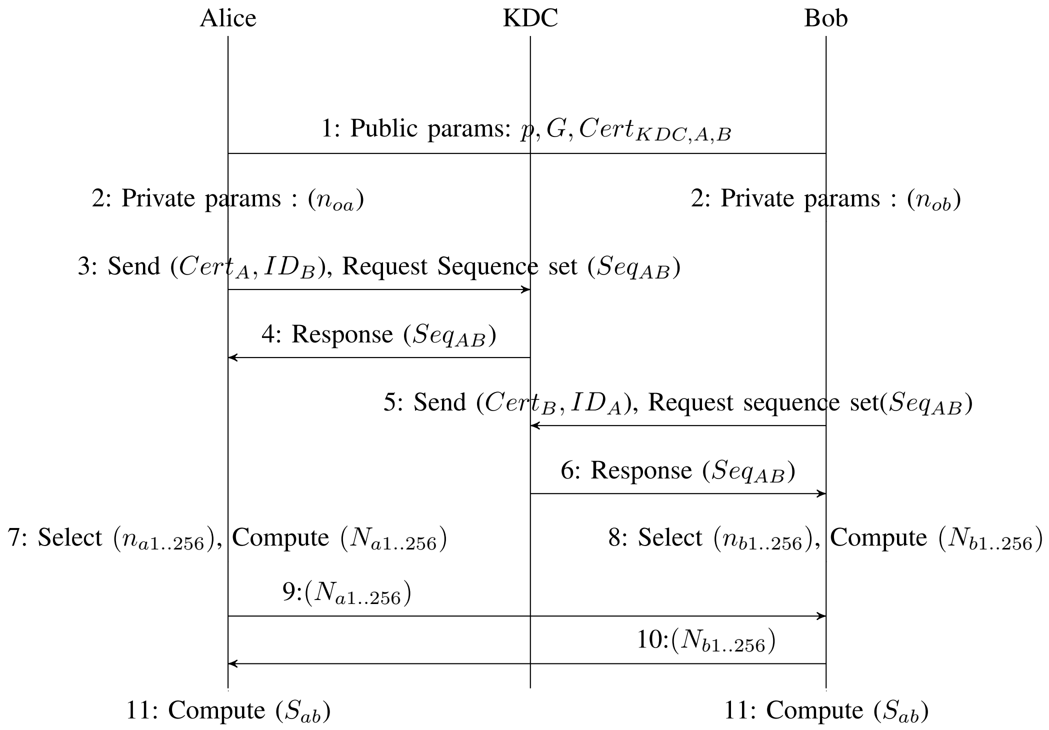
Alice
KDC
Bob
1: Public params: $p, G, Cert_{KDC,A,B}$
2: Private params : (n_{oa})
2: Private params : (n_{ob})
3: Send $(Cert_A, ID_B)$, Request Sequence set (Seq_{AB})
4: Response (Seq_{AB})
5: Send $(Cert_B, ID_A)$, Request sequence set(Seq_{AB})
6: Response (Seq_{AB})
7: Select $(n_{a1..256})$, Compute $(N_{a1..256})$
8: Select $(n_{b1..256})$, Compute $(N_{b1..256})$
9:$(N_{a1..256})$
10:$(N_{b1..256})$
11: Compute (S_{ab})
11: Compute (S_{ab})

Fig. 1. UP-ECDH exchange

n^r, where n is the total number of possibilities and r is the order of sequence set generated via the KDC. In our scheme, 256 keys are chosen; therefore, n = 256. If a sequence set Seq of 256 keys is selected, such that r = 256, this would result to 256^{256} trials to calculate the correct sequence set generated.

- If the number of keys inside the key pool and the sequence set are further increased, it would exponentially increase the number of trials required.
- Benchmarks conducted by Alagic, Gorjan, *et al* in [20] show that a QC requires approximately 2330 logical qubits to successfully break NIST P-256 elliptic curves, and approximately 4.64×10^6 to break the scheme in one day (24 hours). Making UP-ECDH resistant to an attack running on a 2330-qubit quantum computer and arguably beyond.

V. Conclusion and future work

With the speed of development, large scale quantum computers could be in our hands in the coming decades if not the next, with no existing answers for the threats they impose.

By integrating the primitive of unpredictability, UP-ECDH is proposed, aiming at achieving a quantum resistant scheme with a high security level, low performance impact and low storage overhead, that could provide an immediate answer to the current state of affairs.

In future research, unpredictability can be further extended to other cryptosystems such as supersingular elliptic curve isogenies and other, in order to enhance their security effectiveness, while maintaining relatively low performance overhead. The UP-ECDH scheme can also be further enhanced to utilize a quantum-secure KDC for initial sequence set exchange to make it more quantum resistant.

References

[1] W. Diffie and M. E. Hellman. New directions in cryptography. IEEE Transactions on Information Theory, 22(6):644–654, 1976.

[2] D. J. Bernstein and T. Lange (editors). eBACS: ECRYPT Benchmarking of Cryptographic Systems. http://bench.cr.yp.to, October 2013

[3] A. K. Lenstra and E. R. Verheul. Selecting cryptographic key sizes. Journal of Cryptology, 14(4):255–293, 2001

[4] Johnson, Don, Alfred Menezes, and Scott Vanstone. ”The elliptic curve digital signature algorithm (ECDSA).” International journal of information security 1.1 (2001): 36-63.

[5] P. W. Shor, “Algorithms for Quantum Computation: Discrete Logarithms and Factoring,” in Proceedings of the 35th Annual Symposium on Foundations of Computer Science, ser. SFCS ’94. Washington, DC, USA: IEEE Computer Society, 1994, pp. 124–134.

[6] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, Strengths and weaknesses of quantum computing, SIAM J. Comput., 26 (5), 1997, pp. 1510–1523. http://dx.doi.org/10.1137/s0097539796300933

[7] L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone, “NIST: Report on Post-Quantum Cryptography,” NIST, Tech. Rep., 2016

[8] McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. In: DSN Prog. Rep., Jet Prop. Lab., California Inst. Technol., Pasadena, CA, pp. 114–116 (January 1978)

[9] Elwyn Berlekamp, Robert McEliece, and Henk van Tilborg. On the inherent intractability of certain coding problems. IEEE Trans. Inform. Theory, 24(3):384 {386, May 1978.}

[10] Téllez, Claudio, Diogo Pereira, and Fábio Borges. ”Trade-off between performance and security for coding and ring learning with errors-based diffie-hellman cryptosystems.” 2019 II Workshop on Metrology for Industry 4.0 and IoT (MetroInd4. 0&IoT). IEEE, 2019.

[11] Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process. Retrieved 2020-07-23.

[12] Ducas L., Plançon M., Wesolowski B. (2019) On the Shortness of Vectors to Be Found by the Ideal-SVP Quantum Algorithm. In: Boldyreva A., Micciancio D. (eds) Advances in Cryptology – CRYPTO 2019. CRYPTO 2019. Lecture Notes in Computer Science, vol 11692. Springer, Cham.

[13] D.J. Bernstein, A. Hülsing, S. Kölbl, R. Niederhagen, J. Rijneveld, P. Schwabe, The sphincs+ signature framework, in Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, 2019, pp. 2129–2146

[14] Ducas, Léo, et al. ”Crystals-dilithium: A lattice-based digital signature scheme.” IACR Transactions on Cryptographic Hardware and Embedded Systems (2018): 238-268.

[15] Fouque, P.A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Prest, T., Ricosset, T., Seiler, G., Whyte, W., Zhang, Z.: Falcon: Fast-Fourier lattice-based compact signatures over NTRU. https://csrc.nist.gov/CSRC/media/Projects/ Post-Quantum-Cryptography/documents/round-1/submissions/Falcon.zip, accessed: 2018-11-26

[16] Ding, Jintai & Petzoldt, Albrecht. (2017). Current State of Multivariate Cryptography. IEEE Security & Privacy. 15. 28-36. 10.1109/MSP.2017.3151328.

[17] Menezes, Alfred & Vanstone, Scott & Okamoto, Tatsuaki. Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field. IEEE Transactions on Information Theory, 1991

[18] Neuman, B. Clifford, and Theodore Ts'o. ”Kerberos: An authentication service for computer networks.” IEEE Communications magazine 32.9 (1994): 33-38.

[19] Mohamed Helmy Mostafa Megahed, “SurvSecSecurity Architecture for Reliable Surveillance WSN Recovery from Base Station Failure”, PhD Thesis, Ottawa University, Ottawa, Canada, 2014

[20] Alagic, Gorjan, et al. ”Status report on the second round of the NIST post-quantum cryptography standardization process.” US Department of Commerce, NIST (2020).