 doi:10.20944/preprints202103.0716.v2

Non-Commutative Key Exchange Protocol

Luis Adrián Lizama-Pérez[0000−0001−5109−2927] and J. Mauricio López R.[2]

Cinvestav Querétaro, Libramiento Norponiente 2000,
Real de Juriquilla, 76230, Santiago de Querétaro, Querétaro, México
jm.lopez@cinvestav.mx

Abstract. We introduce a novel key exchange protocol based on non-commutative matrix multiplication defined in $\mathbb{F}_p^{n\times n}$. The security of our method does not rely on computational problems as integer factorization or discrete logarithm whose difficulty is conjectured. We show that the public, secret and channel keys become indistinguishable to the eavesdropper under matrix multiplication. Remarkably, for achieving a 512-bit security level, the public key is 1024 bits and the private key is 768 bits, making them the smallest keys among post-quantum key exchange algorithms. Also, we discuss how to achieve key authentication, interdomain certification and Perfect Forward Secrecy (PFS). Therefore, Lizama's algorithm becomes a promising candidate to establish shared keys and secret communication between (IoT) devices in the quantum era.

Keywords: Non-commutative · matrix · cryptography

1 Introduction

In 2017 the National Institute of Standards and Technology (NIST) initiated the process of evaluating the cryptographic algorithms that will be used to support security in the quantum era. Unfortunately, most of the cryptosystems used today will become obsolete in the foreseeable future because they can be broken by quantum computers [1]. Shor's algorithm [2] solves the mathematical problems on which cryptography is supported: integer factorization and discrete logarithm. Although quantum principles have threatened the security of major cryptographic systems, they have raised a new technology known as quantum key distribution (QKD) that allows remote secret key establishment [3,4,5,6].

Post-quantum crypto-systems under evaluation for public-key quantum-resistant [7] include lattice-based cryptography as well as multi-variate-based, hash-based [8,9] and code-based systems [10]. After the third evaluation round, NIST has selected seven algorithms (and eight alternative candidates), four of them are public key encryption (and key-establishment) systems and three correspond to digital signature algorithms. In the first category, CRYSTALS-KYBER, NTRU-HPS, SABER are lattice-based while Classic McEliece is a code-based public key encryption system. Regarding digital signature schemes, CRYSTALS-DILITHIUM and FALCON are lattice-based and Rainbow is a multivariate-based algorithm [11,12,13]. According to the criteria defined by NIST, quantum algorithms must be resistant against classical and quantum adversaries, their security level must be comparable to the security of SHA-385 and AES-256. Issues to be considered are the size of the keys and the required computing resources and facility of implementation (in hardware and software). Versatility of the algorithm will be evaluated based on its ability to encrypt messages, perform digital signatures and/or allow key exchange.

As discussed in "Non-invertible public key certificates" [14], Lizama's certification method is scalable and interoperable and can be exploited in the pre-quantum and quantum era because the protocol exhibits indistinguishability of the integers used in the public key and ciphertexts.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 23 March 2022 doi:10.20944/preprints202103.0716.v2

Moreover, public key size in Lizama's protocol is the smallest yet: 0.256 kilobytes and 0.384 kilobytes for the public key and certified key, respectively [14].

In this work, we will introduce a new key exchange algorithm based in non-commutative matrix multiplication that can be useful for secret communication in the pre-quantum era, as well as in the quantum era.

The article is organized as follows: in Section 2 we discuss some related protocols starting with the Diffie-Hellman algorithm. In Section 3 we introduce our Non-Commutative Key Exchange Protocol (nc-KEP) to later introduce, in Section 4, the generalized non-commutative KEP. Section 5 describes a process to certificate the public keys across interdomain certificates. Finally, Section 6 details our PFS method that guarantees the secrecy of the new session keys.

2 Related protocols

Without wishing to discuss them exhaustively, in this section we will give a brief introduction to the main cryptographic key establishment methods. We will begin with the Diffie-Hellman protocol, which we consider the starting point for subsequent protocols. In addition, We will briefly describe the Quantum Key Distribution (QKD) method.

2.1 Diffie-Hellman

Diffie-Hellman (DH) key exchange [15] works over a ring $\mathbb{Z}_p$ with large order p. The module p and the generator g, which is primitive root in $\mathbb{Z}_p$ are publicly shared. Alice chooses randomly an exponent integer x_a and computes $k_a = g^{x_a} \bmod p$ which she sends to Bob. Similarly, Bob obtains and responds to Alice with $k_b = g^{x_b} \bmod p$. Then each of them performs exponentiation using the received number as incoming, such that Alice's computes $(g^{x_b} \bmod p)^{x_a} \bmod p = g^{x_b x_a} \bmod p$ and Bob's computes $(g^{x_a} \bmod p)^{x_b} \bmod p = g^{x_a x_b} \bmod p$ (see Figure 1). Both numbers are equal because modular exponentiation follows the normal rules of ordinary exponentiation.

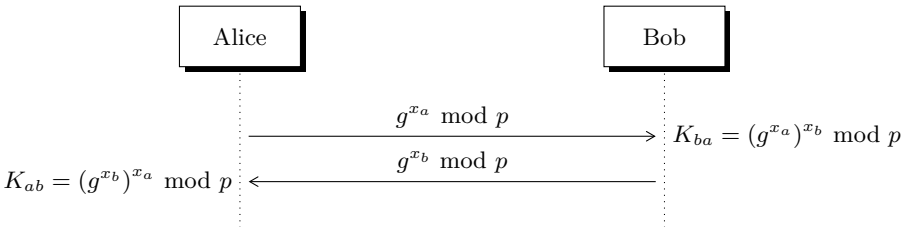

Fig. 1: Diffie–Hellman protocol.

The eavesdropper, Eve, would try to recover g^{ab} from (g, G, g^a, g^b). The Diffie-Hellman algorithm is defined by $F(g, G, g^a, g^b) = g^{ab}$. We say that a group G with large order p satisfies the Computational Diffie-Hellman (CDH) assumption if no efficient algorithm exists to compute $F(g, G, g^a, g^b) = g^{ab}$ [16]. Closely related to the Computational Diffie-Hellman (CDH) assumption is the Discrete Logarithm Problem (DLP) which is defined as recovering x given g and $g^x \bmod p$.

 doi:10.20944/preprints202103.0716.v2

2.2 Stickel

Stickel's key exchange protocol was motivated by the Diffie-Hellman protocol [15]. In the original formulation, a group of invertible matrices over a finite field [17,18] was used in the protocol. Let G be a public non-abelian finite group. Let a, $b \in G$ be public elements such that $ab \neq ba$. Let the orders of a and b be N and M respectively:

1. Alice chooses two random natural numbers $n < N$, $m < M$ and sends $u = a^n b^m$ to Bob.
2. Bob picks two random natural numbers $r < N$, $s < M$ and sends $v = a^r b^s$ to Alice.
3. Alice derives the key as $K_A = a^n v b^m = a^{n+r} b^{m+s}$.
4. Bob computes $K_B = a^r u b^s = a^{n+r} b^{m+s}$.

Unfortunately, a linear algebra attack to this protocol has been published [19,18]. It is sufficient for the adversary to find matrices x and y such that $xa = ax$, $yb = by$, and $xu = y$, because x corresponds to a^{-n}, while y equals b^m [20].

2.3 Anshel-Anshel-Goldfeld

It defines a cryptographic primitive that uses non-commutative subgroups of a given platform group with efficiently computable normal forms. It was implemented in a braid group. This scheme assumes that the Conjugacy Search Problem (CSP) is difficult enough, so it might be implemented in other groups [18]. Let G be a group and elements $a_1, \ldots, a_m, b_1, \ldots, b_n \in G$ be public.

1. Alice picks a private $u \in G$ as a word $a = u(a_1, \ldots, a_m)$ in alphabet $A^{\pm 1}$, encodes (by normal forms), and sends publicly $b_1^a, \ldots, b_n^a$.
2. Bob takes a (secret) word $b = v(b_1, \ldots, b_n)$ in alphabet $B^{\pm 1}$, encodes (by normal forms), and sends publicly $a_1^b, \ldots, a_m^b$.
3. To decode, Alice computes $a^b = u(a_1^b, \ldots, a_m^b)$ and Bob gets $b^a = v(b_1^a, \ldots, b_n^a)$. The common secret key is $a^{-1}a^b = a^{-1}(b^{-1}ab) = (a^{-1}b^{-1}a)b = (b^a)^{-1}b$.

2.4 Jintai Ding

It uses the learning with errors (LWE/RLWE) problem to build a key exchange scheme that is considered post-quantum. The basic idea of the construction can be viewed as an extension of the Diffie-Hellman problem with errors [21] which does the same thing using associativity and commutativity, namely,

$$\mathbf{x}^T\mathbf{M}\mathbf{y} = (\mathbf{x}^T\mathbf{M})\mathbf{y} = \mathbf{x}^T(\mathbf{M}\mathbf{y})$$

where $\mathbf{M}$ is an $n \times n$ matrix in $\mathbb{Z}_q$ and $\mathbf{x}$, $\mathbf{y}$ are vectors in $\mathbb{Z}_q^n$. Introducing small errors is required according to the LWE problem defined as follows: Let $\mathbb{Z}_q$ denote the ring of integers module q and let $\mathbb{Z}_q^n$ denote the set of n-vectors over $\mathbb{Z}_q$. There is a certain unknown linear function $f : \mathbb{Z}_q^n \to \mathbb{Z}_q$ such that, when the input is a sample of pairs $(\mathbf{x}, y)$ where $\mathbf{x} \in \mathbb{Z}_q^n$ and $y \in \mathbb{Z}_q$, we have high probability of $y = f(\mathbf{x})$.

2.5 Bennett-Brassard (BB84)

Although quantum principles have threatened the security of major cryptographic systems [2], they have raised a new technology known as Quantum Key Distribution (QKD) that allows remote secret key establishment. QKD protocols exploit the principle of an eavesdropper being unable to alter quantum communication without producing a detectable noise [3]. Let us observe post-processing methods have emerged to accelerate the rate of the secret bits [22,6].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 23 March 2022 doi:10.20944/preprints202103.0716.v2

3 Lizama's Non-Commutative Key Exchange Protocol

Now, we will introduce the non-commutative Key Exchange Protocol (nc-KEP) which is based on classic non-commutative matrix algebra defined in $\mathbb{F}_p^{n\times n}$. The public key of user i is the pair $(\mathbf{P_i}, \mathbf{Q_i})$. The public and private keys of Alice and Bob are written in Table 1. Public keys are computed according to Equation 1 where $\mathbf{u}$ and $\mathbf{w}$ are publicly shared, square, non-diagonalizable matrices defined in $\mathbb{F}_p^{n\times n}$.

Table 1: Key's definition in Lizama's non-commutative algorithm.

User	Public Key	Private Key
Alice	$(\mathbf{P_a}, \mathbf{Q_a})$	$(\mathbf{k_a}, x_a, y_a)$
Bob	$(\mathbf{P_b}, \mathbf{Q_b})$	$(\mathbf{k_b}, x_b, y_b)$

Equation 1 requires exponentiation by squaring in $\mathbb{F}_p^{n\times n}$. The symbol $\cdot$ in the equations represents matrix multiplication. The private key is defined as the triplet $(\mathbf{k_i}, x_i, y_i)$ where $\mathbf{k}$ is a random square matrix in $\mathbb{F}_p^{n\times n}$ and the pair (x_i, y_i) consists of two random private integers module p.

$$\begin{aligned} \mathbf{P_i} &= \mathbf{k_i} \cdot \mathbf{u}^{x_i} \cdot \mathbf{k_i}^{-1} \\ \mathbf{Q_i} &= \mathbf{k_i} \cdot \mathbf{w}^{y_i} \cdot \mathbf{k_i}^{-1} \end{aligned} \tag{1}$$

The public keys of Alice and Bob are shown in Table 1. The protocol behaves according to the following steps:

1. Alice and Bob exchange their keys with each other through a public channel. Then, they compute the so-called channel key $\mathbf{C_a}$ and $\mathbf{C_b}$, as indicated by Equation 2.

$$\begin{aligned} \mathbf{C_a} &= \mathbf{P_b}^{x_a} \cdot \mathbf{Q_b}^{y_a} \\ &= \mathbf{k_b} \cdot \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b} \cdot \mathbf{k_b}^{-1} \\ \mathbf{C_b} &= \mathbf{P_a}^{x_b} \cdot \mathbf{Q_a}^{y_b} \\ &= \mathbf{k_a} \cdot \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b} \cdot \mathbf{k_a}^{-1} \end{aligned} \tag{2}$$

2. The derived channel key $\mathbf{C_a}$ (or $\mathbf{C_b}$) is sent back to the other user, as depicted in Figure 2. Each user applies the left (and right) multiplications indicated by Equation 3. As shown in the equation, both keys are identical, thus the shared secret key between users is $\mathbf{k_s} = \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b}$.

$$\begin{aligned} \mathbf{k_{sa}} &= \mathbf{k_a}^{-1} \cdot \mathbf{k_a} \cdot \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b} \cdot \mathbf{k_a}^{-1} \cdot \mathbf{k_a} \\ &= \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b} \\ \mathbf{k_{sb}} &= \mathbf{k_b}^{-1} \cdot \mathbf{k_b} \cdot \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b} \cdot \mathbf{k_b}^{-1} \cdot \mathbf{k_b} \\ &= \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b} \\ \mathbf{k_{sa}} &= \mathbf{k_{sb}} \end{aligned} \tag{3}$$

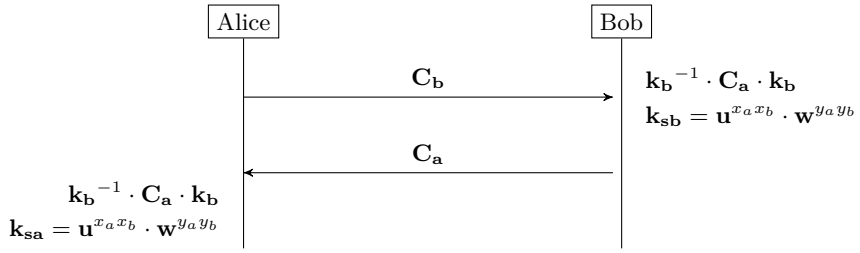

Fig. 2: Lizama's non-commutative Key Exchange Protocol (nc-KEP). The shared secret key is $\mathbf{k_s} = \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b}$.

Cryptosystem. Encryption can be easily achieved because the shared secret key $\mathbf{k_s} = \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b}$ can be properly inverted to decrypt a block message of a size equal to the matrix $\mathbf{k_s}$, as written in Equation 4 where $\mathbf{m}$ and $\mathbf{c}$ are defined in $\mathbb{F}_p^{n\times n}$. Since not every possible matrix is an invertible matrix, users must restart the protocol in the case they derive a non-invertible matrix. The Hill cipher system is vulnerable to a known-plaintext attack, so we will demonstrate in Section 6 how to safely generate a new secret key from the current one.

$$\begin{aligned} \mathbf{c} &= \mathbf{k_s}^{-1} \cdot \mathbf{m} \cdot \mathbf{k_s} \\ \mathbf{m} &= \mathbf{k_s} \cdot \mathbf{c} \cdot \mathbf{k_s}^{-1} \end{aligned} \tag{4}$$

4 Security Analysis

Preliminaries. According to "Group-based cryptography" [18] the Conjugacy Search Problem (CSP) is defined as: given a recursive presentation of a group G and two conjugate elements $u, h \in G$, find out a particular element $k \in G$ such that $k^{-1}uk = h$. It also implies that there should be a way to disguise elements of G so that it would be impossible to recover k from $k^{-1}uk$ just by inspection. Indeed, a derived problem of the Conjugacy Search Problem is the Decomposition Search Problem (DSP) that states: given two elements w and w' of a group G, find two elements x and y that would belong to a given subset (usually a subgroup) $A \subseteq G$ and satisfy $x \cdot w \cdot y = w'$; provided that at least one such pair of elements exists. If we denote kuk^{-1} by u^k, it looks like the DLP [23].

In the nc-KEP, the public key is computed as $\mathbf{k} \cdot \mathbf{u}^x \cdot \mathbf{w}^y \cdot \mathbf{k}^{-1} = \mathbf{h}$ defined in $\mathbb{F}_p^{n\times n}$ where the triplet $(\mathbf{k}, x, y)$ is the private key. Despite $\mathbf{h}$, $\mathbf{u}$ and $\mathbf{w}$ being publicly known, in accordance with the conjugacy problem definition, the eavesdropper is forced to guess $\mathbf{k}$ but also $\mathbf{u}^x$ and $\mathbf{w}^y$ because x and y are unknown.

So, let us rewrite the conjugacy problem as: given a group G defined in $\mathbb{F}_p^{n\times n}$ and one conjugate element $\mathbf{h} \in G$, find out $\mathbf{k}$, $\mathbf{u}^x$ and $\mathbf{w}^y \in {\mathbb{F}_p}^{n\times n}$ such that $\mathbf{k}^{-1} \cdot \mathbf{u}^x \cdot \mathbf{w}^y \cdot \mathbf{k} = \mathbf{h}$. Consequently, this involves complexity other than the conjugacy problem (or the decomposition search problem) alone. Moreover, we will base the security of our method on the property of indistinguishability that the secret, channel and public keys exhibit. By showing that such keys are indistinguishable under multiplication of at least two big integers, we claim that our method must be considered post-quantum.

Secret Key. Suppose a user acts as a malicious Eve, so after they establish a key with Alice, they obtain the shared key $\mathbf{k_s}$. Suppose Eve is equipped with a quantum computer capable of

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 23 March 2022 doi:10.20944/preprints202103.0716.v2

running quantum algorithms that solve DLP. Further on, in a more general case, assume this device is capable to recovering x_a given $\mathbf{u_e}$ and $\mathbf{u_e}^{x_a}$ when $\mathbf{u_e}$ is defined in $\mathbb{F}_p^{n\times n}$. However, the key $\mathbf{k_s}$ is still inaccessible to the eavesdropper because the key $\mathbf{k_s}$ can be separated into pairs of factors $(\mathbf{x_a}, \mathbf{y_a})$ as shown by Equation 5, where $\mathbf{x_a} = \mathbf{u_e}^{x_a}$ and $\mathbf{y_a} = \mathbf{w_e}^{y_a}$, which provides indistinguishability to the key $\mathbf{k_s}$. Let's see how many different pairs of factors can be derived from this equation. Since p is prime, then $\mathbf{k_s}$ generates a group of size p under exponentiation. We can introduce a variation in the exponent of one term and multiply the second term by this variation with the opposite sign.

$$\begin{aligned} \mathbf{u}^{x_a x_e} \cdot \mathbf{w}^{y_a y_e} &= \mathbf{k_s} \\ \mathbf{u_e}^{x_a} \cdot \mathbf{w_e}^{y_a} &= \mathbf{x_a} \cdot \mathbf{y_a} \end{aligned} \tag{5}$$

Thus, as shown by Equation 6 the factors $(\mathbf{x_a}, \mathbf{y_a})$ can be defined as $\mathbf{x_a} = \mathbf{u_e}^{x_a \pm i}$ and $\mathbf{y_a} = \mathbf{u_e}^{\mp i} \cdot \mathbf{w_e}^{y_a}$ in the first relation and $\mathbf{x_a} = \mathbf{u_e}^{x_a} \cdot \mathbf{w_e}^{\mp i}$ and $\mathbf{y_a} = \mathbf{w_e}^{y_a \pm i}$ in the second, where $i = 0 \ldots p$. Then, the total number of pairs $(\mathbf{x_a}, \mathbf{y_a})$ is $2^{4\,|p|}$ which yields an exponential order in the length of p.

$$\begin{aligned} \mathbf{u_e}^{x_a \pm i} \cdot \mathbf{u_e}^{\mp i} \cdot \mathbf{w_e}^{y_a} &= \mathbf{k_s} \\ \mathbf{u_e}^{x_a} \cdot \mathbf{w_e}^{\mp i} \cdot \mathbf{w_e}^{y_a \pm i} &= \mathbf{k_s} \end{aligned} \tag{6}$$

Channel Key. Alice's channel key $\mathbf{C_a}$ is defined according to the Equation 7 where $\mathbf{x_{ab}} = \mathbf{u}^{x_a x_b}$ and $\mathbf{y_{ab}} = \mathbf{w}^{y_a y_b}$. However, as it was discussed for the secret key, $\mathbf{C_a}$ can be separated into factors that give indistinguishability to the key, so that the eavesdropper is forced to mount an exhaustive search among the factors.

$$\begin{aligned} \mathbf{P_b}^{x_a} \cdot \mathbf{Q_b}^{y_a} &= \mathbf{C_a} \\ \mathbf{k_b} \cdot \mathbf{u}^{x_a x_b} \cdot \mathbf{w}^{y_a y_b} \cdot \mathbf{k_b}^{-1} &= \mathbf{k_b} \cdot \mathbf{x_{ab}} \cdot \mathbf{y_{ab}} \cdot \mathbf{k_b}^{-1} \end{aligned} \tag{7}$$

Public Key. The relations $\mathbf{P_a} = \mathbf{k_a} \cdot \mathbf{u}^{x_a} \cdot \mathbf{k_a}^{-1}$ and $\mathbf{Q_a} = \mathbf{k_a} \cdot \mathbf{w}^{y_a} \cdot \mathbf{k_a}^{-1}$ define Alice's public key. Yet, using the previous reasoning as shown by Equation 8, the public key $(\mathbf{P_a}, \mathbf{Q_a})$ is conditioned to multiplication for indistinguishability.

$$\begin{aligned} \mathbf{P_a} &= \mathbf{k_a} \cdot \mathbf{x_a} \cdot \mathbf{k_a}^{-1} \\ \mathbf{Q_a} &= \mathbf{k_a} \cdot \mathbf{y_a} \cdot \mathbf{k_a}^{-1} \end{aligned} \tag{8}$$

Furthermore, if n is the size of the square matrices $(\mathbf{P_a}, \mathbf{Q_a})$, we found that $\mathbf{P_a}$ (and also $\mathbf{Q_a}$) defines a system of n^2 equations in $3 \cdot n^2$ variables. For example, if $n = 2$ and we rewrite $\mathbf{P_a} = \mathbf{k_a} \cdot \mathbf{u}^{x_a} \cdot \mathbf{k_a}^{-1}$ as the matrix multiplication $\mathbf{P} = \mathbf{k} \cdot \mu \cdot \mathbf{t}$, then we can expand it as represented in Equation 9, because we have defined $\mathbf{u}$ as a non-diagonalizable matrix; thus $\mathbf{u}^{x_a}$ is also non-diagonalizable.

$$\mathbf{P} = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \cdot \begin{bmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \end{bmatrix} \cdot \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \mod p \tag{9}$$

In this example $(n = 2)$ we arrived to the following 4 equations in 12 variables:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 23 March 2022 doi:10.20944/preprints202103.0716.v2

$$k_{11}\cdot\mu_{11}\cdot t_{11}+k_{12}\cdot\mu_{21}\cdot t_{11}+k_{11}\cdot\mu_{12}\cdot t_{21}+k_{12}\cdot\mu_{22}\cdot t_{21}=\mathrm{P}_{11}$$
$$k_{11}\cdot\mu_{11}\cdot t_{12}+k_{12}\cdot\mu_{21}\cdot t_{12}+k_{11}\cdot\mu_{12}\cdot t_{22}+k_{12}\cdot\mu_{22}\cdot t_{22}=\mathrm{P}_{12}$$
$$k_{21}\cdot\mu_{11}\cdot t_{11}+k_{22}\cdot\mu_{21}\cdot t_{11}+k_{21}\cdot\mu_{12}\cdot t_{21}+k_{22}\cdot\mu_{22}\cdot t_{21}=\mathrm{P}_{21}$$
$$k_{21}\cdot\mu_{11}\cdot t_{12}+k_{22}\cdot\mu_{21}\cdot t_{12}+k_{21}\cdot\mu_{12}\cdot t_{22}+k_{22}\cdot\mu_{22}\cdot t_{22}=\mathrm{P}_{22}$$

If $\mathbf{u}$ were diagonalizable, we could write it as $\mathbf{u}=\mathbf{g}\cdot\mathbf{d_u}\cdot\mathbf{g}^{-1}$ then $\mathbf{u}^{x_a}=\mathbf{g}\cdot{\mathbf{d_u}}^{x_a}\cdot\mathbf{g}^{-1}$ and the following eavesdropping strategy could be applied (as provided to us by a reviewer): let $\mathbf{X}$ and $\mathbf{Y}=\mathbf{X}^{-1}$ be two matrices where each entry is represented as a variable. Moreover, $\mathbf{D}={\mathbf{d_u}}^{x_i}$ contains the two additional variables $\mathbf{D}[1,1]=(\mathbf{d_u}[1,1])^{x_i}$ and $\mathbf{D}[2,2]=(\mathbf{d_u}[2,2])^{x_i}$. Then, the matrix equations $\mathbf{Y}\cdot\mathbf{P_a}\cdot\mathbf{X}=\mathbf{D}$ and $\mathbf{Y}\cdot\mathbf{X}=\mathbf{I}$ give a system of 8 quadratic equations in 10 variables (and, at least, one valid solution corresponding to $\mathbf{X}=\mathbf{k_a}\cdot\mathbf{g}$).

For p prime, compute a Gröbner basis for this system, and recover the entries in $\mathbf{D}$ in the following way: utilize lex order, ordering the variables from $\mathbf{D}$ last. This yields a univariate polynomial of degree two in the Gröbner basis whose roots are the entries of $\mathbf{D}$. Note that this does not recover the order in which the entries of $\mathbf{D}$ appear (e.g., which is entry[1,1] and which is entry [2,2]), but since n is typically small, the possible combinations can be brute–forced. There are many possible solutions for the pair $\mathbf{X}$ and $\mathbf{Y}$, but a unique solution seems to be found after e.g., fixing entry [1,1] and [2,1] in $\mathbf{X}$ (to any value).

5 Performance Analysis

Let $|p|$ be the length of the prime public integer p. The public key $(\mathbf{P_a},\mathbf{Q_a})$ is computed as the multiplication $\mathbf{k}\cdot\mu\cdot\mathbf{t}$ where $\mu=\mathbf{u}^{x_a}$ and $\mathbf{t}=\mathbf{k}^{-1}$. Since $\mathbf{k}$, μ and $\mathbf{t}$ are square matrices of size n, the size of $\mathbf{P_a}$ is $n^2\cdot|p|$ and the size of the public key $(\mathbf{P_a},\mathbf{Q_a})$ is $2n^2\cdot|p|$. If $n=2$ and $|p|=128$ the size of the public key achieves 1024 bits.

The private key is defined as $(\mathbf{k_a},x_a,y_a)$. Then we leave $|x_a|=|y_a|=|p|$ because the integer x_a (or y_a) only matters in module p, that is x_a mod p. The size of the private key is computed as $n^2\cdot|p|+2\cdot|p|$. In a case where $n=2$ and $|p|=128$, the size of the private key is $512+256=768$ bits.

The size of the secret key which is computed as $\mathbf{k_s}=\mathbf{u}^{x_ax_b}\cdot\mathbf{w}^{y_ay_b}$ is $n^2\cdot|p|$. In the example, the secret key achieves 512 bits. Other parameter sizes are written in Table 2.

Table 2: Some key sizes when $|p|=128$ and 256 as a function of n. Sizes are written in bits.

$\|p\|$	128			256		
n/key	2	3	4	2	3	4
Public	1,024	2,304	4,096	2,048	4,608	8,192
Private	768	1,408	2,304	1,536	2,816	4,608
Secret	512	1,152	2,048	1,024	2,304	4,096

6 Certificated Keys

An indispensable property of public keys is authentication by a Certification Authority (CA). The keys of the non-commutative Key Exchange Protocol (nc-KEP) can be certified if the CA

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 23 March 2022 doi:10.20944/preprints202103.0716.v2

raises the keys to their private key number x_{ca} and y_{ca} as indicated in Table 3. Alice and Bob exchange their public certified keys from the CA's web service. Then, they perform the usual exponentiation $(\mathbf{u}^{x_i x_{ca}})^{x_j}$ and $(\mathbf{w}^{y_i y_{ca}})^{y_j}$. The secret shared key is $\mathbf{u}^{x_i x_{ca} x_j} \cdot \mathbf{w}^{y_i y_{ca} y_j}$.

Table 3: CA's public database. CA performs exponentiation over the public keys. The secret shared key is $k_s = \mathbf{u}^{x_a x_{ca} x_b} \cdot \mathbf{w}^{y_a y_{ca} y_b}$ defined in $\mathbb{F}_p^{n\times n}$.

User	Public key	Certified key
CA	$\left(\mathbf{k_{ca}} \cdot \mathbf{u}^{x_{ca}} \cdot \mathbf{k_{ca}}^{-1},\ \mathbf{k_{ca}} \cdot \mathbf{w}^{y_{ca}} \cdot \mathbf{k_{ca}}^{-1}\right)$	-
Alice	$\left(\mathbf{k_a} \cdot \mathbf{u}^{x_a} \cdot \mathbf{k_a}^{-1},\ \mathbf{k_a} \cdot \mathbf{w}^{y_a} \cdot \mathbf{k_a}^{-1}\right)$	$\left(\mathbf{k_a} \cdot \mathbf{u}^{x_a x_{ca}} \cdot \mathbf{k_a}^{-1},\ \mathbf{k_a} \cdot \mathbf{w}^{y_a y_{ca}} \cdot \mathbf{k_a}^{-1}\right)$
Bob	$\left(\mathbf{k_b} \cdot \mathbf{u}^{x_b} \cdot \mathbf{k_b}^{-1},\ \mathbf{k_b} \cdot \mathbf{w}^{y_b} \cdot \mathbf{k_b}^{-1}\right)$	$\left(\mathbf{k_b} \cdot \mathbf{u}^{x_b x_{ca}} \cdot \mathbf{k_b}^{-1},\ \mathbf{k_b} \cdot \mathbf{w}^{y_b y_{ca}} \cdot \mathbf{k_b}^{-1}\right)$

6.1 Interdomain certificates

Users that have been certified with different Certification Authorities, say CA_1 and CA_2 can establish a secret key, provided each CA has certified their keys with the other CA. It means that, after the second certification, the public key for Alice is $\left(\mathbf{k_a} \cdot \mathbf{u}^{x_a x_{ca_1} x_{ca_2}} \cdot \mathbf{k_a}^{-1},\ \mathbf{k_a} \cdot \mathbf{w}^{y_a y_{ca_1} y_{ca_2}} \cdot \mathbf{k_a}^{-1}\right)$ and Bob's public key is $\left(\mathbf{k_b} \cdot \mathbf{u}^{x_b x_{ca_1} x_{ca_2}} \cdot \mathbf{k_b}^{-1},\ \mathbf{k_b} \cdot \mathbf{w}^{y_b y_{ca_1} y_{ca_2}} \cdot \mathbf{k_b}^{-1}\right)$. The shared secret key with Bob will be $\mathbf{u}^{x_a x_{ca_1} x_{ca_2} x_b} \cdot \mathbf{w}^{y_a y_{ca_1} y_{ca_2} y_b}$ in $\mathbb{F}_p^{n\times n}$.

7 Perfect Forward Secrecy (PFS)

Remote users Alice and Bob may want to establish a new secret key $\mathbf{k_t}$ based on the one they already have, $\mathbf{k_s}$. However, if for some reason $\mathbf{k_t}$ is compromised by an opponent, Perfect Forward Secrecy (PFS) is a property of key agreement protocols which guarantees that such leakage does not compromise the security of previously used keys. In Figure. 3 we depict our PFS protocol producing new session secret keys. Provided that private keys $\mathbf{k_a}$ and $\mathbf{k_b}$ remain secret, the eavesdropper might be able to capture $\mathbf{k_t}$ but they do not know $t_a t_b$, thus they cannot derive $\mathbf{k_s}$.

8 Conclusions

We introduced here the non-commutative key exchange protocol (nc-KEP) which allows secret key establishment between two remote parties in order to enable private communication. Lizama's nc-KEP does not rely on computational problems as integer factorization or discrete logarithm whose complexity is conjectured. We have evaluated by contrast the security of this method based on the indistinguishability of the public, secret and channel keys. Further on, we have discussed the computational complexity that arises with the involved matrix multiplication.

Lizama's nc-KEP achieves 512-bit security level when the public key is 1024 bits and the private key reaches 768 bits while $n = 2$ and $|p| = 128$ bits, reaching the smallest size when compared to the post-quantum systems currently evaluated by NIST. Moreover, we have demonstrated that our method exhibits Certification-Authority scalability and Perfect Forward Secrecy (PFS).

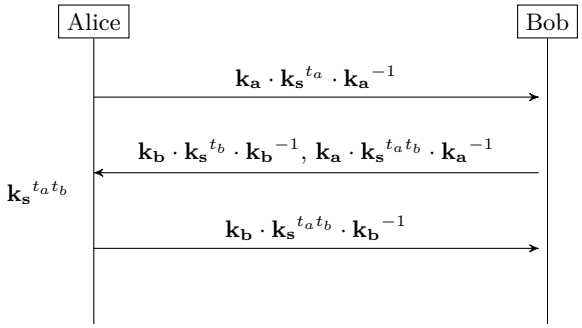

Fig. 3: Perfect Forward Secrecy (PFS) in Lizama's non-commutative Key Exchange Protocol (nc-KEP). The new shared secret key between users is $\mathbf{k_t} = \mathbf{k_s}^{t_a t_b}$ defined in $\mathbb{F}_p^{n \times n}$.

As a result, our method enables secret communication between restricted computational IoT devices in the quantum era. The algorithm would be further optimized in hardware/software, since it basically only requires matrix-multiplication.

References

1. M. A. Barreno, "The future of cryptography under quantum computers," *Dartmouth College Computer Science Technical Reports*, 2002.
2. P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th annual symposium on foundations of computer science*, pp. 124–134, Ieee, 1994.
3. H. Bennett Ch and G. Brassard, "Quantum cryptography: public key distribution and coin tossing int," in *Conf. on Computers, Systems and Signal Processing (Bangalore, India, Dec. 1984)*, pp. 175–9, 1984.
4. L. A. Lizama-Pérez, J. M. López, and E. D. C. López, "Quantum flows for secret key distribution," *Advanced Technologies of Quantum Key Distribution*, p. 37, 2018.
5. L. A. Lizama-Perez and J. M. López, "Quantum key distillation using binary frames," *Symmetry*, vol. 12, p. 1053, Jun 2020.
6. L. A. Lizama-Pérez and J. M. López R., "Beyond the limits of shannon's information in quantum key distribution," *Entropy*, vol. 23, no. 229, 2021.
7. C. S. R. CENTER, "Post-Quantum Cryptography Standardization Conference," 2021. [Online; accessed March 23, 2022].
8. L. A. Lizama-Perez, "Digital signatures over hash-entangled chains," *SN Applied Sciences*, vol. 1, no. 12, p. 1568, 2019.
9. L. A. Lizama-Pérez, L. J. Montiel-Arrieta, F. S. Hernández-Mendoza, L. A. Lizama-Servín, and S.-A. Eric, "Public hash signature for mobile network devices," *Ingeniería, Investigación y Tecnología*, vol. XX, no. 2, pp. 1–10, 2019.
10. D. J. Bernstein, "Introduction to post-quantum cryptography," in *Post-quantum cryptography*, pp. 1–14, Springer, 2009.
11. I. T. Laboratory, "PQC Standardization Process: Third Round Candidate Announcement." `https://csrc.nist.gov/news/2020/pqc-third-round-candidate-announcement`, 2020. [Online; accessed March 23, 2022].
12. L. Chen, L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone, *Report on post-quantum cryptography*, vol. 12. US Department of Commerce, National Institute of Standards and Technology, 2016.
13. E. Persichetti, "NIST Round 3 finalists." `https://pqc-wiki.fau.edu/w/Special:DatabaseHome`, 2020. [Online; accessed March 23, 2022].

14. L. A. Lizama-Pérez and J. M. López R., "Non-invertible public key certificates," *Entropy*, vol. 23, no. 2, 2021.
15. W. Diffie and M. Hellman, "New directions in cryptography," *IEEE transactions on Information Theory*, vol. 22, no. 6, pp. 644–654, 1976.
16. D. Kahrobaei, C. Koupparis, and V. Shpilrain, "Public key exchange using matrices over group rings," *arXiv preprint arXiv:1302.1625*, 2013.
17. E. Stickel, "A new method for exchanging secret keys," in *Third International Conference on Information Technology and Applications (ICITA'05)*, vol. 2, pp. 426–430, IEEE, 2005.
18. A. Myasnikov, V. Shpilrain, and A. Ushakov, *Group-based cryptography*. Springer Science & Business Media, 2008.
19. V. Shpilrain, "Cryptanalysis of stickel's key exchange scheme," in *International Computer Science Symposium in Russia*, pp. 283–288, Springer, 2008.
20. D. Grigoriev and V. Shpilrain, "Tropical cryptography," *Communications in Algebra*, vol. 42, no. 6, pp. 2624–2632, 2014.
21. J. Ding, X. Xie, and X. Lin, "A simple provably secure key exchange scheme based on the learning with errors problem.," *IACR Cryptol. ePrint Arch.*, vol. 2012, p. 688, 2012.
22. L. A. Lizama-Perez and J. M. López, "Quantum key distillation using binary frames," *Symmetry*, vol. 12, no. 6, p. 1053, 2020.
23. J. H. Cheon and B. Jun, "A polynomial time algorithm for the braid diffie-hellman conjugacy problem," in *Annual International Cryptology Conference*, pp. 212–225, Springer, 2003.