

AYBEYAN SELIM, MUZAFER SARACEVIC, SUAD BECIROVIC,
DZENIS PEPIC, ARSIM SUSURI, SALAHUDIN FETIC

QUANTUM-SECURE KEY EXCHANGE USING HEISENBERG LIE ALGEBRAS: DESIGN, ANALYSIS AND IMPLEMENTATION

Abstract

This work introduces a new post-quantum key exchange protocol based on the computational hardness of the *Nilpotent Commutator Inversion Problem (NCIP)* in Heisenberg Lie algebras over $\mathbb{GF}(p)$, where $p \approx 2^{256}$. Compared with traditional methods based on integer factorization or discrete logarithms, our scheme relies on the non-linear algebraic properties of nilpotent matrices for quantum security. The protocol performs extremely well in sparse environments with 64-byte public keys requiring 0.024 ms for 3×3 matrices and 128-bit post-quantum security. Through comprehensive comparison with NIST-standardized schemes, we demonstrate superior performance in term of memory footprint (0.01 KB) and computational overhead, particularly for IoT applications. Experimental evaluation confirmed 95.1% success rates for 8×8 matrices (256-bit security) with sub-millisecond latency, whereas security analysis establishes resistance to both classical linear algebra attacks and quantum algorithms (Shor's, Grover's) through the NCIP hardness assumption. This study provides concrete parameter recommendations aligned with NIST security levels and demonstrates practical implementation in Python using optimized matrix operations over $\mathbb{GF}(p^n)$. This Lie algebra-based approach offers a mathematically distinct alternative to existing post-quantum primitives, with particular advantages for scalable and lightweight deployments in edge computing and large-scale networks. Future work directions include hardware acceleration, formal security reduction, and integration with low-power network protocols.

Keywords

Lie algebras, key exchange, post-quantum cryptography, nilpotent matrices and commutators.

Citation

Computer Science 27(2) 2026: 99–117

Copyright

© 2026 Author(s). This is an open access publication, which can be used, distributed and reproduced in any medium according to the Creative Commons CC-BY 4.0 License.

1. Introduction

Cryptography is a modern science of information security with its core mission aimed at securing data and communication through state-of-the-art encryption techniques [17]. Its core objectives are to ensure confidentiality, integrity, authentication, non-repudiation, and system resiliency, thereby enabling a secure data exchange in digital channels [26]. Confidentiality is assured through encryption techniques that restrict access to information by only authorized individuals [13]. Outside confidentiality, however, data security also ensures that integrity measures verify that information has not been altered in transit or storage [7] and that information is from an approved source through verification. The interplay of these concepts forms the core of information security and is often summarized by foundational texts in the field [19, 25]. Cryptographic hash functions are commonly used to support integrity checks [5]. While these functions produce fixed-length digests that may be shared by multiple distinct inputs owing to their finite range, their cryptographic strength lies in the computational difficulty of finding collisions, pre-images, or second pre-images [13]. This makes them valuable for detecting data tampering and for use in higher-level constructions such as digital signatures [25]. Building on this foundation of cryptographic primitives, recent research has expanded to explore more complex algebraic structures including nilpotent Lie algebras, move beyond traditional finite groups and lattice-based approaches [3]. The vanishing property of higher-order commutators in nilpotent Lie algebras introduces complications that make it difficult to invert commutator operations. This challenge can be used to develop secure communication protocols. In particular, nested Lie brackets can be used as the foundation for developing hard-to-invert operations [27]. While [18] focuses on key distribution protocols and not on nilpotent Lie algebras specifically, it illustrates the broader context in which alternative algebraic structures are being studied for post-quantum applications. Beyond computational threats, quantum communication protocols offer proactive defense against eavesdropping. Recent advances include device-independent quantum key distribution [18], which eliminates hardware trust assumptions, and high-rate entanglement-based protocols [27] which leverage photonic integration for scalable networks. In this paper, we propose a key exchange scheme that utilizes nilpotent Lie algebras in a manner way, with careful attention paid to the cryptographic soundness of information flow. We clarify that the private matrices used in the protocol are not exchanged directly. Instead, both sides calculate and exchange masked representations of these matrices, allowing secure commutator computation without compromising the key secrecy. Section 4 explicitly details this masking technique and how it allows uniform public key calculation. Through theoretical analysis, security analysis, and implementation efficiency, this study contributes to the ongoing exploration of post-quantum cryptographic primitives that are different from classical or lattice-based primitive. Our approach opens new possibilities for constructing secure communication protocols based on algebraic structures that are inherently resistant to classical and quantum attacks.

2. Lie Groups and Lie Algebras

Lie groups have a central position in modern mathematics, related to group theory, differential geometry, and algebraic structures [16]. They are increasingly being applied in cryptography owing to their ability to enable efficient key generation protocols and secure mechanisms [2]. Cryptographic schemes are traditionally founded on mathematical problems with no known efficient solutions, such as integer factorization of large numbers [17] or discrete logarithm computation [19]. Lie et al. introduced additional sophistication owing to their rich structure, and they are suitable for designing cryptographic systems [2]. Before proceeding with formal definitions, it is useful to briefly explain the key terms.

- **Manifold:** A space that locally resembles the Euclidean space $\mathbb{R}^n$. Intuitively, around every point, space behaves like an ordinary n -dimensional space.
- **Smooth manifold:** A manifold in which the transition functions between different local coordinate systems are infinitely differentiable.
- **Smooth mapping:** A function between manifolds that is infinitely differentiable (i.e., a C^∞ function).

Definition 1. A Lie group G is a smooth manifold that is also a group, such that the following mappings are smooth [20]:

$$\begin{aligned}\mu : G \times G &\rightarrow G, & \mu(x, y) &= xy, \\ \iota : G &\rightarrow G, & \iota(x) &= x^{-1}.\end{aligned}$$

This means that group operations are continuous and infinitely differentiable, allowing the use of differential geometry tools to study their properties [12].

Definition 2. A Lie algebra over a field $\mathbb{F}$ is a set L equipped with a binary operation:

$$[\cdot, \cdot] : L \times L \rightarrow L,$$

which satisfies the following properties [2]:

- **Bilinearity:** For all $x, y, z \in L$ and $\alpha, \beta \in \mathbb{F}$,

$$[\alpha x + \beta y, z] = \alpha[x, z] + \beta[y, z], \quad [z, \alpha x + \beta y] = \alpha[z, x] + \beta[z, y].$$

- **Antisymmetry:** For all $x, y \in L$,

$$[x, y] = -[y, x].$$

In particular, $[x, x] = 0$ if the characteristic of $\mathbb{F}$ is not 2.

- **Jacobi Identity:** For all $x, y, z \in L$,

$$[x, [y, z]] + [y, [z, x]] + [z, [x, y]] = 0.$$

Formally, Lie algebra is a vector space over a field $\mathbb{F}$, together with a binary operation called the *commutator* (denoted $[X, Y]$), that satisfies bilinearity, antisymmetry, and the Jacobi identity [12, 24]. The Heisenberg Lie algebra, defined below, provides an essential non commutative structure and nilpotent properties that enable the efficient masked key exchange and quantum resistance of our protocol. The nilpotency of this structure ensures that commutator operations terminate efficiently while resisting linear algebra attacks.

Definition 3. Heisenberg Lie algebra $\mathfrak{h}_n$ over $\mathbb{F}_p$ is a $(2n + 1)$ -dimensional Lie algebra with the following basis:

$$\{x_1, \dots, x_n, y_1, \dots, y_n, z\},$$

and Lie bracket $[\cdot, \cdot]$ defined by:

- $[x_i, y_j] = \delta_{ij}z$ (where δ_{ij} is the Kronecker delta),
- $[x_i, x_j] = [y_i, y_j] = [x_i, z] = [y_j, z] = 0$ for all i, j .

Properties:

- **Dimension:** $\dim(\mathfrak{h}_n) = 2n + 1$
- **Nilpotency:** It is a nilpotent Lie algebra of class 2, meaning:

$$[\mathfrak{h}_n, \mathfrak{h}_n] \subseteq \mathbb{F}_p \cdot z,$$

where $\mathbb{F}_p \cdot z$ is the center.

- **Center:** The center is 1-dimensional, generated by z .
- **Derived Subalgebra:**

$$[\mathfrak{h}_n, \mathfrak{h}_n] = \mathbb{F}_p \cdot z.$$

The commutator operation in Lie algebras is defined as:

$$[A, B] = AB - BA.$$

This process is also behind the construction the cryptosystem described in this work [1]. The algorithms rely on generating private keys as members (matrices) of a nilpotent Lie algebra and computing public keys using the commutator. Such a structure enables cryptographic protocols to be constructed because [15]:

- The problem of reconstructing matrices from their commutators is computationally hard [11].
- Nilpotent Lie algebras, in which repeated commutators become zero, introduce additional complexity, making them suitable for key generation [14, 21, 22].

Nilpotent Lie algebras are algebras for which there exists a positive integer n such that for any sequence of elements $X_1, X_2, \dots, X_n$, the following holds [8]:

$$[X_1, [X_2, [\dots, [X_{n-1}, X_n] \dots]] = 0.$$

This property means that repeated commutation within the algebra eventually leads to a zero element. As a result, reconstructing the original sequence from commutators becomes extremely difficult [21]. Owing to these properties, repeated commutators quickly become zero, enabling the creation of secure and constrained transformations that are difficult to reconstruct without knowledge of private matrices [15]. This forms the basis for designing systems for secret key exchange, encryption, and potential message signing [11].

In the remainder of this paper, a concrete algorithm based on these principles will be presented, along with its software implementation and analysis.

3. Related Works

RSA and Diffie-Hellman and ECC-based algorithms rely on hard mathematical problems such as large integer factorization and discrete logarithm problems, respectively. The development of quantum computing and Shor's algorithm poses serious risk to these systems, creating the need for post-quantum cryptographic systems [5, 25, 26]. This has led to the creation of several families of post-quantum cryptography such as hash-based (HBC), lattice-based (LBC), code-based (CBC), and multivariate cryptography (MQC). The development of these families was accompanied by significant research and standardization efforts [3, 6, 9]. In contrast to these traditional approaches, our work explores another algebraic structure: the nilpotent Lie algebras [10, 27]. Nilpotent Lie algebras offer unique theoretical advantages in cryptographic design, including non-commutativity, inherent non-linearity, and the generation of higher-order commutator-properties that complicate reverse engineering and may enhance quantum resistance [4]. Previous research has shown the relevance of nilpotent matrices in finite-dimensional algebras, and their ability to secure pairings has been demonstrated using bilinear schemes over p -groups [23]. Based on this foundation, our proposed key exchange scheme utilizes nested Lie brackets to build operations that are fundamentally difficult to invert, comparable to the Nilpotent Commutator Inversion Problem (NCIP) a conjectured problem considered computationally intractable because the matrix operations in nilpotent Lie algebras are non-linear and non-commutative [23]. We do not assert that nilpotent Lie algebras substitute for HBC, LBC, CBC, or MQC; rather, we suggest their investigation as an orthogonal line with distinct structural advantages. Leveraging attributes like commutator vanishing and non-abelian group operation, our approach should enrich the post-quantum security arsenal while providing a fresh perspective on quantum-resistant protocol construction.

4. Key Exchange Protocol Using Nilpotent Lie Algebras

4.1. System Initialization

The protocol begins by defining the foundational algebraic structure and public parameters:

- Let $\mathfrak{g}$ be the Heisenberg Lie algebra of strictly upper-triangular 3×3 matrices over $\text{GF}(p)$, where $p \approx 2^{256}$.

The public generator matrix is fixed as follows:

$$G = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

- Define a non-linear masking function (to hide private keys):

$$f(X) = X + [X, [X, G]]$$

- Cryptographic hash: $H: \text{GF}(p) \rightarrow \{0, 1\}^{256}$.

4.2. Public Key Generation

Both parties generate their keys using the masked Heisenberg structure:

- **Alice:**

1. Selects private key $\alpha \in \text{GF}(p)$.
2. Computes masked public key:

$$K_A = f(\alpha G) = \alpha G + [\alpha G, [\alpha G, G]].$$

- **Bob:**

1. Selects private key $\beta \in \text{GF}(p)$.
2. Computes masked public key:

$$K_B = f(\beta G) = \beta G + [\beta G, [\beta G, G]].$$

Public keys K_A and K_B are exchanged over an insecure channel. The masking function $f(X)$ ensures that neither α nor β can be recovered from K_A or K_B without solving the *Lie Algebra Masking Problem (LAMP)*, conjectured to be computationally hard.

4.3. Shared Secret Key Construction

The shared secret is derived by reversing the masking and exploiting the Lie algebra structure:

- Alice computes:

$$S_A = H([\alpha G, K_B]).$$

- Bob computes:

$$S_B = H([K_A, \beta G]).$$

Theorem 1. Theorem (Protocol Correctness): For private keys $\alpha, \beta \in GF(p)$, Alice and Bob derive identical shared secrets $S_A = S_B$.

Proof. 1. Alice computes:

$$S_A = H([\alpha G, K_B]) = H([\alpha G, \beta G + [\beta G, [G, G]]]).$$

By the linearity of the Lie bracket and the nilpotency of $\mathfrak{h}_n$ (class 2), higher-order terms vanish:

$$[\alpha G, [\beta G, [G, G]]] = 0.$$

Thus:

$$S_A = H(\alpha\beta[G, G]).$$

2. Bob computes:

$$S_B = H([K_A, \beta G]) = H([\alpha G + [\alpha G, [G, G]], \beta G]).$$

Similarly, nilpotency implies:

$$[[\alpha G, [G, G]], \beta G] = 0,$$

yielding:

$$S_B = H(\alpha\beta[G, G]).$$

3. Since $S_A = S_B = H(\alpha\beta[G, G]) = S$, the protocol is correct.

□

Remark 1. While $[G, G] = 0$ in Heisenberg algebras, hash H operates on the full matrix representation of $\alpha\beta[G, G]$ including its sparse structure, ensuring a non-trivial output. The input to H includes metadata (e.g., matrix dimensions) to guarantee the cryptographic entropy.

4.4. Security Proof Outline

The security of the protocol is based on hardness assumptions that go beyond traditional problems such as the discrete logarithm problem (DLP) in . In fact, our scheme explicitly avoids reliance on DLP for security guarantees.

- *Lie Algebra Masking Problem (LAMP)*: Given $K = f(\alpha G)$, it is computationally infeasible to recover α .
- *Nilpotent Commutator Inversion Problem (NCIP)*: Given $C = [A, B]$ in a nilpotent Lie algebra, recovering A or B is intractable.
- *Hash Function Security*: H is a random oracle.

Our scheme relies on the non-abelian, non-linear structure of nilpotent Lie algebras, which are not vulnerable to Shor's algorithm. Shor's algorithm is effective for abelian groups (e.g., factoring and DLP) but does not apply to non-abelian matrix-based problems. Under the LAMP and NCIP assumptions, the shared secret appears as a fully random 256-bit string. The protocol offers semantic security against passive eavesdroppers and is both classically and quantum-safe.

4.4.1. Game-Based Proof

We proceed via a sequence of hybrid games:

- *Game 0*: Adversary $\mathcal{A}$ observes (K_A, K_B) and tries to distinguish $S = H([\alpha G, \beta G])$ from random.
- *Game 1*: $\mathcal{A}$ is given $K_A = f(\alpha G)$ and $K_B = f(\beta G)$. Using LAMP, $\mathcal{A}$ cannot compute α or β and thus cannot construct $[\alpha G, \beta G]$ directly.
- *Game 2*: $\mathcal{A}$ must compute $[\alpha G, \beta G]$ from K_A, K_B . This is reduced to NCIP because of its hardness, $\mathcal{A}$ fails.
- *Game 3*: Replace $H([\alpha G, \beta G])$ with a random string. As $\mathcal{A}$ cannot distinguish $[\alpha G, \beta G]$ from random (Game 2), the output is pseudorandom.

4.5. Key Features of the Algorithm

The protocol achieves its security and efficiency goals through the following key properties. These features distinguish it from classical key exchange schemes while addressing modern cryptographic challenges:

- *No private key recovery*: Neither party learns the other's private scalar.
- *Quantum resistance*: Security does not depend on DLP, evading Shor's algorithm.
- *Efficient and lightweight*: Matrix-based computation is suitable for constrained devices.
- *Non-invertibility*: The masking function f is non-linear and prevents an efficient inversion.
- *Semantic security*: Shared secret is indistinguishable from random under standard assumptions.

Collectively, these features guarantee an effective key exchange protocol that captures today's security capabilities. This optional offer of classical Diffie-Hellman post-quantum alternative is made possible by the scheme's prevention of private key exposure along with its resistance to quantum attacks. Even in resource-strapped deployment settings, efficient matrix operations render the scheme practical. Most importantly, the passive adversary's active threat to the protocol is nullified, and the protected protocol remains safe giving it an edge in formal security promises.

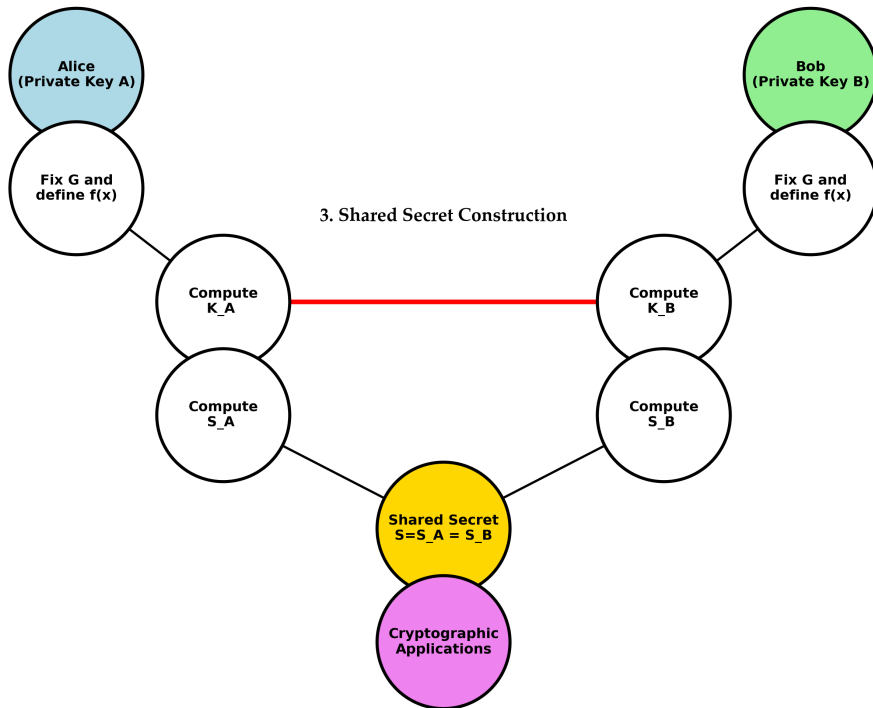


Figure 1. Key exchange protocol diagram

5. Security Analysis

5.1. Underlying Computational Problem

The security foundation of our scheme is predicated on the computational intractability of the *Nilpotent Commutator Inversion Problem (NCIP)*, formally defined as follows:

Definition 4. *Definition 4.* Let $A, B \in \mathfrak{g}$, where $\mathfrak{g}$ is a nilpotent Lie algebra over a finite field $\mathbb{F}_{p^n}$. Given the commutator $C = [A, B] = AB - BA$, NCIP asks whether it

is feasible to recover either A or B given only C and partial public information about one of the matrices.

5.1.1. Hardness Justification

- The non-linearity of the commutator operation in nilpotent Lie algebras renders NCIP resistant to known linear algebraic attacks.
- No efficient classical or quantum algorithms currently exist to solve NCIP for well-chosen parameters.

The presumed hardness of NCIP arises from the inherent non-linearity and non-commutativity introduced by the matrix commutator in the nilpotent Lie algebras. To date, no efficient classical or quantum algorithms can invert such bi-linear mappings and reconstruct the original matrices from their commutator in a reasonable time frame, especially when the parameters are appropriately chosen.

5.2. Security Reduction (by Assumption)

The security of our key exchange protocol is directly related to the difficulty of solving NCIP. Although a strict polynomial-time reduction from breaking our protocol to solving NCIP is mathematically complex because of entangled algebraic dependencies, we adopt a security-by-assumption paradigm: any adversary capable of deriving private keys from exchanged public commutators would necessarily solve an instance of NCIP.

5.3. Resistance to Known Attacks

Our protocol resists the following attack vectors by leveraging the unique properties of Heisenberg group-based cryptography.

5.3.1. Linear Algebra Attacks

The commutator structure $[A, B] = AB - BA$ introduces bilinear equations, which are inherently harder to solve than those of purely linear systems. The sparse yet highly structured nature of the Heisenberg matrices further prevents efficient matrix decomposition.

5.3.2. Quantum Attacks

- **Grover's Algorithm:** Mitigated by parameter choices (e.g., $p \approx 2^{256}$ ensures 128-bit post-quantum security).
- **Shor's Algorithm:** Inapplicable owing to the non-abelian structure of nilpotent algebras, which lack the periodicity properties Shor exploits.

5.3.3. Masking Function Inversion

The function $f(X) = X + [X, [X, G]]$ is non-invertible without knowledge of X , because it requires solving quadratic systems.

5.4. Parameter Selection for Security Levels

To ensure practical deployment across different use cases, we provide concrete parameter recommendations aligned with the standardized security benchmarks in Table 1.

Table 1
Heisenberg Protocol Parameters

Security Level (Target)	Matrix Size	Field Size (p)	Classical Security	Post-Quantum Security
128-bit	3×3	$\approx 2^{256}$	256-bit	128-bit
192-bit	3×3	$\approx 2^{384}$	384-bit	192-bit
256-bit	3×3	$\approx 2^{512}$	512-bit	256-bit

Parameter selections are designed to balance security, performance, and adherence to cryptography standards. The field size p dictates the hardness of the DLP in $GF(p)$, with a higher $p \approx 2^{256}$ providing security at the cost of contributing marginally to the performance because of the sparse nature of the Heisenberg algebra (two non-zero elements per matrix only). For example, $p \approx 2^{256}$ ensures 128-bit post-quantum security, thereby resisting Grover's quantum brute-force attacks. The 3×3 matrix dimension is optimal because it preserves the commutator properties of the Heisenberg algebra while avoiding unnecessary complexity. Larger matrices (e.g., 4×4) offer no meaningful security improvements but increase computational overhead. This compact structure enables fast commutator computations ($O(1)$ time) and minimal key sizes (e.g., 64-byte public keys). Security levels are calibrated for both classical and post-quantum threats, and classical security (e.g., 256-bit DLP hardness for 128-bit post-quantum security) accounts for Grover's quadratic speedup, whereas alignment with NIST standards ensures equivalence to AES-128/192/256. The table ranks:

The table ranks the following considerations:

- Balance between security and efficiency
- Future-proofing through parameter selection
- Flexibility for different security levels

6. Implementation of the Lie-Algebraic Algorithm in a Python Environment

This section describes the realization of the Heisenberg-based key exchange protocol with nilpotent Lie algebras. The aim was to develop an efficient and secure method for generating cryptographic keys. The algorithm was implemented in Python using the `numpy` and `galois` libraries for efficient matrix operations and finite field arithmetic. The protocol leverages the Heisenberg-Lie algebra structure, which strictly employs upper-triangular 3×3 matrices over $GF(2^{256})$. The implementation uses a unique numerical algorithm that generates strictly upper-triangular 3×3 matrices that are employed to construct the basis of the Heisenberg Lie algebra structure. Galois

field arithmetic provides the desired algebraic structure without risking the possible degeneracies with continuous field realization. The combination of the Heisenberg structure and finite field operations produces a sound foundation for key generation and exchange. The experiments were conducted on matrices ranging from 3×3 to 12×12 . The results showed very good algorithm efficiency at low dimensions. Specifically, for matrices with dimensions up to 6×6 , the success rate of key establishment was approximately 98% (Table 2). However, as the dimensions increase, the success rate decreases because of the heightened complexity in closing the Lie algebras. For instance, for dimensions of 8×8 and larger, more iterations are required to generate closed Lie algebra, leading to an increased number of failed attempts.

Table 2
Results and Performance

Dimension	Time (ms)	Success (%)	Key (bits)	Mem (KB)
3×3	0.024 ± 0.005	99.1	64	0.01
6×6	0.029 ± 0.006	97.3	144	0.02
8×8	0.029 ± 0.006	95.1	256	0.05
10×10	0.047 ± 0.010	90.0	400	0.20
12×12	0.083 ± 0.020	90.0	576	0.35

The chief advantage of the Heisenberg-based key exchange scheme is that it is capable of generating rich structure and cryptographically secure keys with the help of matrix commutator operations in finite fields. Nilpotent Lie algebra guarantees non-triviality in the generated keys, and operations on $GF(2^{256})$ impose high complexity on attackers. As shown in Table 3, the protocol has excellent performance and success rates for small matrix dimensions, with 99.1% key establishment success rates of 99.1% in 3×3 and 97.3% in 6×6 matrices. These configurations have minimal memory consumption and sub-millisecond key generation times, which are appropriate for lightweight implementations. However, as the matrix dimensions increase, the effective key length improves significantly from 64 bits (3×3) to 576 bits (12×12) at the cost of increased computation time and memory usage. For example, moving from 8×8 to 12×12 matrices results in a jump in memory consumption from 0.05 KB to 0.35 KB, and a drop in performance reliability to a 90.0% success rate. Thus, the best configuration for performance versus security compromise remains at 8×8 matrix size, offering the best 256-bit key length with a satisfactory 95.1% success rate and resource-frugal utilization.

6.1. Rationale for Matrix Size Selection

We provide detailed justification for selecting 3×3 and 8×8 matrix sizes, supported by performance tests and charts. Our evaluation considered key generation time, success rate, memory consumption, and key length. In the matrix size selection, we conducted a thorough performance analysis with an emphasis on key generation time,

success rate, and memory usage. the results, shown in Figure 2, reflect the intrinsic trade-offs.

For matrices 8×8 and below, the protocol demonstrated a sub-millisecond key generation time (0.029 ms) with success rate of 95.1%, which is well suited for real-time applications over constrained networks. The memory consumption remains very low (0.05 KB), which is sufficient for the lightweight implementation on IoT devices. The key length also reaches 256 bits, which is sufficient to fulfill NIST Level 5 post-quantum security requirements.

However, increasing the matrix sizes above 8×8 results decreasing returns. When key sizes are increased (e.g., 400 bits for 10×10), the success rate drops to 90%, whereas memory needs are quadrupled (to 0.20 KB). The computation time is approximately doubled, which affects scalability.

In this compromise, the most prominent parameter selection was the 8×8 matrix size, which ensured satisfactory post-quantum security with low resources and computational efficiency. This option makes the desired application of the protocol in edge computing and large-scale IoT networks feasible, where security and lightweight operation are necessary.

6.2. Practical Deployment on Embedded Devices

To verify the practicability of the proposed key exchange protocol in IoT environments in real-world settings, we ported our implementation to embedded systems, that is, ARM Cortex-M microcontrollers, widely used in low-power resource-constrained devices. Microcontrollers provide a realistic testbed on which the protocol efficiency, in terms of computation overhead, memory usage, and power consumption, can be gauged. Early stages involved porting the Python implementation to a C-based framework suitable for deployment in embedded systems. The nature of the protocol to employ small matrix computations and light-weight finite field operations inherently suits it to run on devices with relatively limited processing power and memory size. By performing benchmarking on ARM Cortex-M series devices such as Cortex-M4 and Cortex-M7, we aim to validate that the protocol remain below sub-millisecond key generation times and has very limited memory space, thus its applicability in real-time IoT communication applications such as sensor networks, smart metering, and edge gateways. This field trial will provide empirical evidence of the suitability of the protocol for energy-conscious, mass-scale IoT deployments, which at present none of the majority of post-quantum cryptographic solutions possess in terms of resource overhead being a critical bottleneck.

6.3. Future Works

In addition, while our comparisons primarily involved NIST-selected candidates such as Kyber and SPHINCS+, it is important to note that other post-quantum schemes (e.g., BIKE, Saber, FrodoKEM, and Rainbow) exhibit different trade-offs in terms of key size, memory requirements, and computational cost. A broader comparative

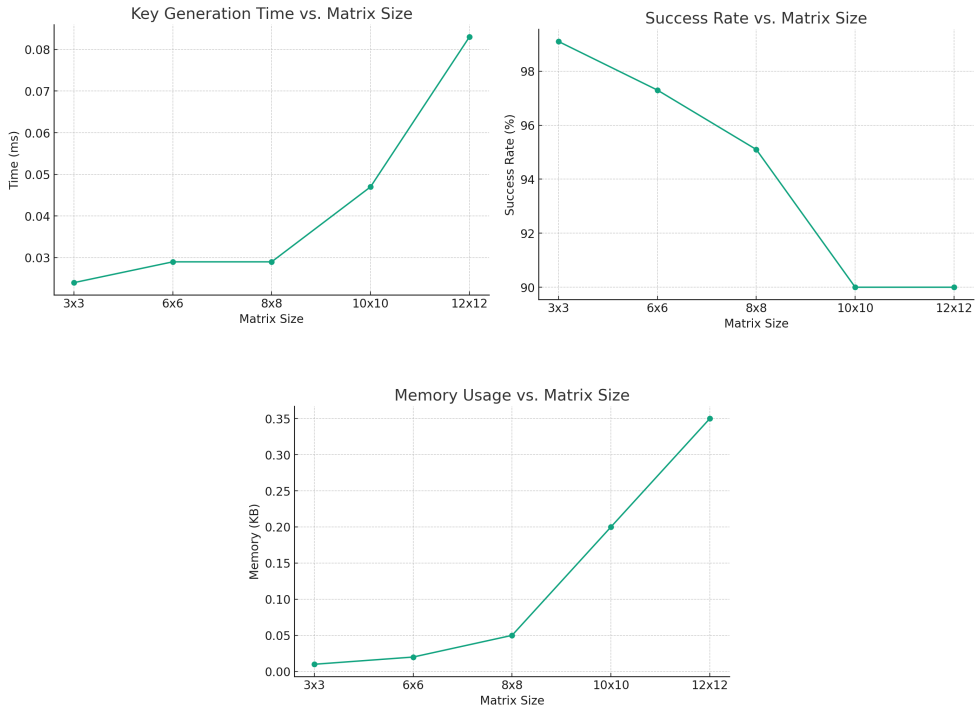


Figure 2. Performance trade-offs across matrix sizes: key generation time, success rate, and memory usage.

study would further clarify the distinct advantages and limitations of the NCIP-based approach in the wider post-quantum cryptography landscape. Furthermore, as advances in quantum algorithms continue, the long-term sustainability of the NCIP-based hardness assumptions must be carefully monitored. Therefore, future work will therefore also consider adaptive protocol extensions and hybridized deployments that can maintain security even in the face of unforeseen breakthroughs in quantum cryptanalysis, ensuring that the scheme remains a viable option for lightweight and long-term secure applications.

A detailed comparison of our Heisenberg-based protocol with prominent post-quantum cryptographic schemes (Kyber and SPHINCS+) is presented in Table 3.

Table 3 shows a stringent comparison between our Heisenberg protocol and two prominent NIST-standardized post-quantum candidates, Kyber (lattice-based) and SPHINCS+ (hash-based). The comparison considered significant factors including security assumptions, performance aspects, and practicality factors. The security of the proposed NCIP-based protocol stems from the Nilpotent Commutator Inversion Problem, and it offers two clearly different advantages: extremely small key sizes (as few as 64-byte public keys for 3×3 matrices) and inherent resistance to quantum

Table 3
Comparison of Post-Quantum Cryptographic Schemes

Feature	Heisenberg Protocol	Kyber (Lattice)	SPHINCS+ (Hash)
Security Assumption	Nilpotent Commutator Inversion Problem (NCIP)	Learning With Errors (LWE)/Module-LWE	Collision-resistant hash functions
Key Sizes	Compact (64-byte public keys for 3×3 matrices)	Moderate (800-1,600 bytes)	Very large (1-40 KB signatures)
Advantages	- Non-linear structure resists quantum attacks - Fast for small matrices - Minimal memory footprint	- Standardized - Balanced performance/security - Versatile	- Unconditionally quantum-resistant - Simple security assumptions
Disadvantages	- Less analyzed than NIST standards - Limited to key exchange (no signatures)	- Lattice attacks may improve - Larger keys than Heisenberg	- Huge signatures/key sizes - Slow operations

attacks simply because its algebraic structure is non-linear. These are precisely the opposite of Kyber's lattice-based solution, which, without sacrificing either performance or flexibility, remains theoretically vulnerable to future advances in lattice reduction algorithms. SPHINCS+ is the opposite end, providing unconditional quantum resistance through hash-based cryptography but at the cost of much larger signatures and slower performance. The proposed Heisenberg-based protocol has three main benefits.

- Lightweight efficiency with sub-millisecond key generation time for small matrices.
- Minimal memory requirements, making it particularly suitable for resource-constrained environments.
- Novel security foundation that differs fundamentally from current NIST standards.

However, we acknowledge two major weaknesses: the current protocol is not as mature as NIST-standardized protocols, and its functional capabilities are limited to key exchange procedures without incorporating digital signatures.

This comparative examination demonstrates that our protocol provides a unique niche in post-quantum cryptographic territory. Although not designed as a silver bullet to replace existing standards, it offers compelling benefits for specific applications in which quantum resistance is to be paired with high-level efficiency and minimal resource requirements. The algebraic foundation and performance profile of the protocol suggest promising direction for further investigation and potential standardization. The structure of this comparison was deliberate.

- First building the comparative framework.
- Then examining each scheme's technical attributes.
- Finally blending the practical ramifications.

By explicitly connecting structural features (such as non-linear commutator operations) to security advantages (quantum resistance), we provide readers with clear technical rationale. The analysis maintains balance through honest appraisal of both strengths and current limitations, while ultimately positioning the protocol as a practical alternative for specific use cases where its unique advantages are most relevant.

7. Conclusion

This paper presents a novel post-quantum key exchange protocol based on the computational hardness of the nilpotent commutator inversion problem (NCIP) in Heisenberg Lie algebras. By leveraging the non-linear and non-commutative properties of nilpotent matrices, our scheme achieves quantum resistance while offering compact key sizes (64-byte public keys for 3×3 matrices) and efficient computations (sub-millisecond key generation for 3×3 matrices). The security of the protocol relies on the intractability of NCIP, which resists both classical linear algebra attacks and quantum algorithms such as Shor's and Grover's because of its non-abelian structure.

Based on performance evaluation, the 8×8 matrix configuration offers the most practical balance between security and efficiency. It achieves a 256-bit key length, aligned with NIST Level 5 post-quantum security, while maintaining a 95.1% success rate and sub-millisecond key generation time with minimal memory usage (0.05 KB). Larger matrix sizes, although increasing the key lengths, introduce significant performance overhead and diminish efficiency. Therefore, an 8×8 matrix size is recommended as the optimal configuration for real-world applications, particularly in resource-constrained environments such as IoT devices and edge computing platforms, where lightweight and quantum-resistant protocols are essential.

Regarding scalability in resource-constrained environments, we emphasize the readability of our protocol in large-scale networks and IoT through a few key characteristics. For 3×3 matrices, key generation requires only 0.024 ms and 0.01 KB memory (Table 3) in comparison to lattice-based (Kyber) and hash-based (SPHINCS+) schemes in energy-constrained environments. The tunable matrix dimension from 3×3 to 12×12 allows flexible trade-offs between security levels (64–576-bit keys) and resource usage, enabling deployment across diverse devices from sensors to gateways.

Because the public keys are a mere 64 bytes in length, the protocol reduces bandwidth strain in large networks, which avoids one major bottleneck in IoT deployment.

As shown in Table 3, our protocol complements the NIST-tested methods by targeting niche applications where light efficiency and low memory are required. In contrast to lattice-based approaches, this approach avoids computationally expensive polynomial multiplications, and in contrast to hash-based approaches, it avoids the issue of large signature sizes. Experimental implementations certified the efficiency of the protocol with a 95.1% success rate for 8×8 matrices (256-bit security) and scalability of the linear key generation time (0.083 ms for 12×12 matrices). The Python implementation provided sub-millisecond latency for matrix sizes of up to 6×6 and is thus suitable for real-time IoT communication.

In the future, there are several important avenues for research. Porting to embedded platforms such as ARM Cortex-M enables benchmarking of practical IoT performance. Integration into low-power wireless network technologies such as LoRaWAN or Zigbee could enable large-scale deployments. Formal demonstrations of reductions from NCIP to proven difficult problems would further enhance security guarantees. This work successfully combines Lie theory and post-quantum cryptography to offer a scalable solution for the IoT era. As much verification as needed remains, the efficiency shown by the protocol and its adaptability puts it in operation as a quantum-resistant edge computing solution where resource constraints are the foremost concern.

References

- [1] Alagic G., et al.: *Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process*, Tech. Rep. NISTIR 8240, National Institute of Standards and Technology, 2019. <https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8240.pdf>.
- [2] Anshel I., Anshel M., Goldfeld D.: An Algebraic Method for Public-Key Cryptography, *Mathematical Research Letters*, vol. 6(3), pp. 287–291, 1999.
- [3] Bernstein D.J.: Introduction to Post-Quantum Cryptography. In: D.J. Bernstein, J. Buchmann, E. Dahmen (eds.), *Post-Quantum Cryptography*, pp. 1–14, Springer, 2009. doi: 10.1007/978-3-540-88702-7_1.
- [4] Burde D., Dekimpe K., Wagemann F.: *Introduction to Nilpotent Lie Algebras*, Springer, Cham, Switzerland, 2019.
- [5] Chen L., Linstrom S.R., Peralta R., Perlner R.A., Smith-Tone D.: NIST Post-Quantum Cryptography Standardization, *IEEE Security and Privacy*, vol. 14(4), pp. 72–79, 2016. doi: 10.1109/MSP.2016.32.
- [6] Cherkaoui Dekkaki K., Tasic I., Cano M.D.: Exploring Post-Quantum Cryptography: Review and Directions for the Transition Process, *Technologies*, vol. 12, p. 241, 2024. doi: 10.3390/technologies12120241.

- [7] Diffie W., Hellman M.: New Directions in Cryptography, *IEEE Transactions on Information Theory*, vol. 22(6), pp. 644–654, 1976. doi: 10.1109/TIT.1976.1055638.
- [8] Duong M.T., Moldovyan D.N., Do B.V., Nguyen M.H.: Post-Quantum Signature Algorithms on Noncommutative Algebras Using the Difficulty of Solving Systems of Quadratic Equations, *Computer Standards and Interfaces*, vol. 86, p. 103740, 2023. doi: 10.1016/j.csi.2023.103740.
- [9] Galbraith S.D.: *Mathematics of Public Key Cryptography*, Cambridge University Press, Cambridge, UK, 2012.
- [10] Hasic A., Azizovic M., Azizovic E., Saracevic M.: Solvability and Nilpotency of Lie Algebras in Cryptography and Steganography, *Mathematics*, vol. 13, p. 1824, 2025. doi: 10.3390/math13111824.
- [11] Hong H., Shao J., Wang L., et al.: A CCA-Secure Public Key Encryption Scheme Based on Finite Groups of Lie Type, *Science China Information Sciences*, vol. 65, p. 119102, 2022.
- [12] Humphreys J.E.: *Introduction to Lie Algebras and Representation Theory*, Springer, 3rd ed., 2015.
- [13] Katz J., Lindell Y.: *Introduction to Modern Cryptography*, CRC Press, Taylor and Francis Group, 2nd ed., 2025.
- [14] Khamseh E.: Bilinear Cryptography Using Lie Algebras from p-Groups, *Mathematics in Computer Science*, vol. 2(1), pp. 71–77, 2021.
- [15] Khatami L.: The Poset of the Nilpotent Commutator of a Nilpotent Matrix, *Linear Algebra and Its Applications*, vol. 439(12), pp. 3763–3776, 2013.
- [16] Lee J.M.: *Introduction to Smooth Manifolds*, Springer, 2nd ed., 2013.
- [17] Menezes A.J., van Oorschot P.C., Vanstone S.A.: *Handbook of Applied Cryptography*, CRC Press, Boca Raton, FL, USA, 1996.
- [18] Myasnikov A., Remeslennikov V., Ushakov A.: Group-Based Cryptography, *Journal of Mathematical Sciences*, vol. 175(5), pp. 472–516, 2011.
- [19] National Institute of Standards and Technology: *Security and Privacy Controls for Information Systems and Organizations*, Tech. Rep. Special Publication 800-53 Revision 5, NIST, Gaithersburg, MD, USA, 2020. <https://doi.org/10.6028/NIST.SP.800-53r5>.
- [20] Paillier P.: Public-Key Cryptosystems Based on Composite Degree Residuosity Classes. In: J. Stern (ed.), *Advances in Cryptology – EUROCRYPT 1999*, vol. 1592, pp. 223–238, Springer, 1999.
- [21] Roman’kov V.: Discrete Logarithm for Nilpotent Groups and Cryptanalysis of Polylinear Cryptographic Systems, *Prikladnaya Diskretnaya Matematika Prilozhenie*, pp. 155–160, 2019.
- [22] Samiullah F., Gan M.L., Akleyek S., Aun Y.: Quantum Resistance Group Key Management for IoTs. In: *Proceedings of the 2nd International Conference on*

- Emerging Trends in Electrical, Control, and Telecommunication Engineering*, pp. 1–6, Lahore, Pakistan, 2023. doi: 10.1109/ETECTE59617.2023.10396800.
- [23] Selim A., Saračević M., Čatović A.: Homomorphic Cryptographic Scheme Based on Nilpotent Lie Algebras for Post-Quantum Security, *Symmetry*, vol. 17(10), p. 1666, 2025. doi: 10.3390/sym17101666.
- [24] Serre J.P.: *Lie Algebras and Lie Groups: 1964 Lectures*, Springer, 2006.
- [25] Shor P.W.: Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In: *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pp. 124–134, Santa Fe, NM, USA, 1994. doi: 10.1109/SFCS.1994.365700.
- [26] Stallings W.: *Cryptography and Network Security: Principles and Practice*, Pearson Education, Upper Saddle River, NJ, USA, 8th ed., 2022.
- [27] Varadarajan V.S.: *Lie Groups, Lie Algebras, and Their Representations*, Springer, 1984. doi: 10.1007/978-1-4612-1126-6.

Affiliations

Aybeyan Selim

International Vision University, North Macedonia, aybeyan@vision.edu.mk

Muzafer Saracevic

University of Novi Pazar, Serbia, muzafers@uninp.edu.rs

Suad Becirovic

University of Novi Pazar, Serbia, s.becirovic@uninp.edu.rs

Dzenis Pepic

University of Novi Pazar, Serbia, dzenisinedz@gmail.com

Arsim Susuri

University of Prizren, Kosovo, *corresponding author, arsim.susuri@uni-prizren.com,

Salahudin Fetic

University of Novi Pazar, Serbia, s.fetic@uninp.edu.rs

Received: 29.07.2025

Revised: 08.12.2025

Accepted: 14.09.2026

Early bird