Journal of Intelligent Decision Support System (IDSS) 8 (2) (2025), pp. 97-105

Published by: Institute of Computer Science (IOCS)

Journal of Intelligent Decision Support System (IDSS)

Journal homepage: www.idss.iocspublisher.org

Redefining hash functions for quantum security with SHA 256

Dadan Shavkat Riswantoro[1], H.A Danang Rimbawa[2]

[1,2] *Cyber Defense Engineering, Universitas Pertahanan Republik Indonesia Bogor, Indonesia*

Article Info

Article history:

Received May 29, 2025
Revised Jun 27, 2025
Accepted Jun 30, 2025

Keywords:

Data Security
Grover's Algorithm
Hash Functions
Post Quantum Cryptograhy
Quantum Computing

ABSTRACT

The rapid advancement of quantum computing technology presents a significant challenge to the field of cryptography, particularly affecting the security of hash functions that form the foundation of many cryptographic protocols. Hash functions are widely used to ensure data integrity, generate digital signatures, and securely store passwords. However, the emergence of quantum algorithms—such as Grover's algorithm—threatens to undermine the security assumptions on which these hash functions are based by significantly reducing their effective security levels. This paper aims to provide a comprehensive analysis of the vulnerabilities introduced by quantum computing to traditional hash functions, detailing how these weaknesses can be exploited by quantum adversaries. We explore the fundamental properties of hash functions, including pre-image resistance, second pre-image resistance, and collision resistance, and assess how these properties are affected in a quantum context. Furthermore, we examine the implications of these vulnerabilities for existing cryptographic systems and emphasize the urgent need for the development of post-quantum cryptographic standards. In response to these challenges, we review ongoing research efforts focused on designing hash functions that are resilient to quantum attacks. We evaluate several promising candidates for post-quantum hash functions, considering their security properties, performance metrics, and practical applicability. The findings of this paper highlight the necessity of transitioning to post-quantum cryptographic solutions to safeguard sensitive information in an increasingly quantum-capable world. Ultimately, we advocate for proactive measures within the cryptographic community to adopt and implement these new standards, thereby ensuring robust data security in the age of quantum computing.

Corresponding Author:

Dadan Shavkat Riswantoro,
Cyber Defense Engineering,
Universitas Pertahanan Republik Indonesia,
IPSC Sentul Area, Sukahati, Citeureup District, Bogor Regency, West Java 16810, Indonesia .
Email: rdsherade@gmail.com

Introduction

Quantum computation (QC), first conceptualized through Benioff's quantum Turing machines (Benioff, 1980) and Feynman's proposal has evolved into one of the most transformatove technologies of the 21st century(Preskill, 2018), offering solutions to complex problems that are currently infeasible for classical computers(Regev, 2004). The ideas for circumventing the difficulty of simulating quantum mechanics by classical computers(Kiktenko et al., 2018). From a national perspective, this issue is particularly relevant to countries like Indonesia, where digital transformation is accelerating across critical sectors including

e-government, e-commerce, and national defense communication systems. The lack of quantum-resilient infrastructure in Indonesia's current digital security landscape raises concerns about readiness to withstand future quantum attacks(Supriati et al., 2025). Therefore, proactive evaluation and adaptation of cryptographic components, such as hash functions, are urgently needed to protect national cyber sovereignty.

Although several studies have proposed approaches to mitigate the threat of quantum computing—such as increasing hash output size, adopting SPHINCS+ signature schemes, and exploring lattice-based methods—these efforts often remain conceptual or lack empirical validation through quantum simulations. Prior work has primarily emphasized theoretical constructs or classical security perspectives, leaving a gap in practical evaluation using real quantum frameworks.

However, this progress poses significant risks to the field of cryptography(Bernstein & Lange, 2017). Algorithms like Shor's and Grover's have demonstrated the potential to undermine the foundational assumptions of many cryptographic systems(Stevens et al., 2017), such as the difficulty of factoring large integers or solving discrete logarithms(Wang et al., 2009).

Hash functions, critical tools in cryptography(Amin et al., 2018), play a pivotal role in ensuring data integrity(He et al., 2017), enabling secure authentication(Cui et al., 2017), and underpinning digital signatures(M. N. Wegman & J. L. Carter, 1981). Unlike encryption algorithms(SAHAI & WATERS, 2021), hash functions are not directly invertible(Bogdanov et al., 2011), making them seemingly more resistant to quantum attacks(Dobraunig et al., 2021). However, Grover's algorithm, which optimizes unstructured search(Lyubashevsky et al., 2013), reduces the security strength of hash functions by halving their effective key space(Fernandez-Carames & Fraga-Lamas, 2020). This reduction forces the cryptographic community to rethink and redesign hash functions to prepare for a quantum computing future(Alladi et al., 2020).

In this research, we specifically address this gap by implementing simulations of Grover's algorithm using Qiskit to analyze the degradation of hash function security under quantum search conditions. Our scientific contribution lies in empirically comparing classical brute-force hash inversion and Grover's quantum search to highlight the vulnerability of SHA-256 in a practical setting. Furthermore, we evaluate post-quantum alternatives and provide recommendations based on their feasibility, performance, and adaptability to existing systems, offering a practical contribution to the development of post-quantum hash standards—especially in the context of Indonesia's cyber defense readiness.

This paper aims to explore the vulnerabilities introduced by quantum computing to traditional hash functions, analyze emerging post-quantum solutions, and evaluate their practical applicability. This work aims to contribute to the broader effort of developing robust cryptographic standards in the post-quantum era.

Hash functions, such as SHA-256 and SHA-3, are integral to modern cryptography, offering three critical properties:Pre-image(Zinzindohoué et al., 2017). Resistance: Ensuring that given a hash, it is computationally infeasible to retrieve the original input. Second Pre-image Resistance: Protecting against attacks where an alternative input produces the same hash as a specific input. Collision Resistance: Preventing the discovery of two distinct inputs that hash to the same value.These properties support applications ranging from password storage and data integrity checks to digital signatures and blockchain technology(Fukuhara & Kaji, 2021). As foundational cryptographic primitives, any vulnerabilities in hash functions can have wide-reaching consequences(Damgård, 1990).

Quantum computing introduces a new class of challenges to cryptographic security. Grover's (Gentry et al., 2008)algorithm, for instance, reduces the complexity of finding a pre-image or a collision in hash functions from $2n$ to $2n/2$(Bernstein et al., 2015), where n is the bit length of the hash. While this reduction is less severe than the exponential weakening of RSA and ECC caused by Shor's algorithm(Du et al., 2005), it still necessitates significant rethinking of security levels(Agrawal et al., 2016). For instance, a 256-bit hash function, which is classically secure against 2128 operations, is reduced to 264 under quantum attack scenarios, potentially making it susceptible(Fernandez-Carames & Fraga-Lamas, 2020).

Several approaches are being explored to enhance the quantum resistance of hash functions: Increased Hash Length:

By doubling the output size of hash functions, the effective security under Grover's algorithm can be restored to classically acceptable levels(Peris-Lopez et al., 2006). However, this approach requires increased computational resources and bandwidth.

Hash-Based Signatures:

Techniques such as the SPHINCS+ signature scheme utilize hash functions in ways that remain secure against quantum attacks(Bernstein et al., 2015). SPHINCS+ achieves security through stateless design and large output spaces, making it a strong candidate in the NIST Post-Quantum Cryptography Standardization process.

Lattice-Based Approaches:

Lattice-based cryptography offers an alternative foundation for designing secure hash functions and ther cryptographic primitives(Lyubashevsky et al., 2010). These approaches exploit problems that are believed to remain hard even in the quantum era.

Despite promising theoretical advancements, transitioning to quantum-resistant hash functions involves several challenges:Performance Trade-offs: Post-quantum solutions often require more computational power and storage, impacting their practicality for real-time systems. Compatibility: Adapting existing systems to integrate post-quantum solutions must account for legacy infrastructure to avoid disruptive overhauls. Standardization: Ongoing efforts, such as those by NIST, aim to standardize post-quantum algorithms, ensuring consistent and reliable implementation across applications.

Method

A. Simulation Overview

The simulation explores how Grover's algorithm affects the security of classical hash functions. SHA-256 is used as the primary case study to represent widely adopted hash functions that rely on computational hardness. This research simulates quantum attacks to evaluate vulnerabilities and propose resilient alternatives.

B. Simulation Steps and Technical Parameters

The simulation was conducted using Qiskit version 0.45.1 on a classical computing environment with the following specifications: Processor: Intel Core i7, RAM: 16 GB, OS: Windows 10, Python version: 3.10.8

These details ensure that the research is replicable and accessible to other researchers.

Three hash output lengths were tested: 8, 128, and 256 bits. 8-bit is used for visualization and simplified demonstration. 128- and 256-bit represent realistic cryptographic standards, especially SHA-256.

For Grover's search, the number of iterations is determined by the theoretical optimum: $\sqrt{(2^n)}$ iterations per bit-length scenario. E.g., for 8-bit: $\sqrt{(256)} = 16$ iterations.

The classical brute-force method ran 10,000 iterations per test to provide comparative metrics.

C. Validation and Replicability

The simulation results were validated against Grover's theoretical model. For example, in the 8-bit scenario, the algorithm succeeded around the 16th iteration, consistent with the expected $\sqrt{(2^8)}$. Simulations were repeated with various random seeds to confirm statistical consistency and reliability. The framework ensures that other researchers using the same setup (Qiskit + classical machine) can replicate the experiments with consistent outcomes.

D. Evaluation Metrics

Security Strength: Comparison of classical (2^n) vs. quantum ($2^{n/2}$) complexity. Performance: Runtime and memory usage for different bit sizes. Practicality: Deployment readiness on real-world systems. Quantum Resilience: Comparative robustness of SHA-256 vs. post-quantum hash approaches (e.g., SPHINCS+).

E. Theoretical Framework

This framework leverages Grover's search to model quantum threats to hash functions. The

analysis considers: Pre-Image Resistance, Evaluates how quantum computing affects the difficulty of finding an input for a given hash output. Collision Resistance, Analyzes susceptibility to attacks where two inputs produce the same hash. Post-Quantum Solutions, Examines proposed methods like increasing output sizes or using hash-based signatures (e.g., SPHINCS+), which inherently resist quantum threats.

F. Experimental Setup

The simulation tests hash functions with varying output sizes (n=8,128,256n = 8, 128, 256n=8,128,256) to determine how bit-length affects security and efficiency. Results are compared between classical and quantum contexts, highlighting the trade-offs of increased output sizes.

G. Results and Interpretation

The outcomes are visualized using histograms to demonstrate the probabilities of measuring the correct hash. These results emphasize the reduced security levels caused by quantum attacks and guide the development of post-quantum cryptographic standards.

H. Maintaining the Integrity of the Specifications

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

Results and Discussions

Advances in quantum computing technology have brought about major changes in the way we view cryptographic security, particularly in hash functions such as SHA-256. Classical search algorithms that rely on the exploration of entire space are likely to now face major challenges from quantum algorithms like Grover, which are mathematically able to reduce the complexity of searches. In this simulation, we compare the efficiency of two search approaches—classical search and Grover's algorithm—by measuring each other's ability to find inputs that match the target hash at an 8-bit hash length. The following discussion presents the results of both methods, which not only highlights Grover's efficiency but also provides a deeper insight into the quantum impact on traditional hash functions. These results are relevant to understand the urgent need for post-quantum solutions to maintain data integrity and security.

A. Impact of Quantum Computing on SHA-256

The study shows that Grover's algorithm significantly reduces the security level of traditional hashes, including SHA-256. This algorithm reduces the complexity of the search from 2n To 2n/2 , making traditional hashes more vulnerable to quantum attacks. For example, a 256-bit hash that classically requires 2128 brute force operations only requires 264 operations in the context of quantum computing. Although this impact is not as large as the Shor algorithm on RSA or ECC, the threat to hash integrity remains real.

B. Post-Quantum Solutions for Hash Functions

Some of the proposed solutions include: Hash Length Increase: Doubles the hash output length (for example, to 512-bit) to restore classic security even though it requires more resources. Hash-Based Signatures: Approaches like SPHINCS+ offer high security with greater hash output and a stateless design. Lattice-Based Approach: Offers an alternative base that is more resistant to quantum threats.

C. Experimental Evaluation

In the 8-bit simulation scenario, the classical brute-force approach took an average of 4976.4 iterations to find a match, while Grover's algorithm succeeded in an average of 16.1 iterations, closely aligning with the theoretical prediction of $\sqrt{(2^8)}$. The classical search was run for 10,000 iterations, while

Grover's method used 16 iterations per run. Execution time was also evaluated: Average classical execution per trial: 0.0021 seconds, Average Grover execution per trial: 0.0078 seconds. Grover's higher time is attributed to quantum circuit emulation overhead. However, it offers significant reduction in search space complexity.0

D. Visualization Results

Graphical visualizations showed comparative performance between brute-force and Grover's algorithm. Variance in Grover's results was <2.4%, with a maximum error rate of 3.2%, demonstrating reliability even under quantum noise simulations.

The graphical view shows two main metrics: Image: The graph below shows the results of a comparison between the number of matches found and the number of iterations required for the classic and Grover search algorithms. Image: The graph below shows the results of a comparison between the number of matches found and the number of iterations required for the classic and Grover search algorithms.

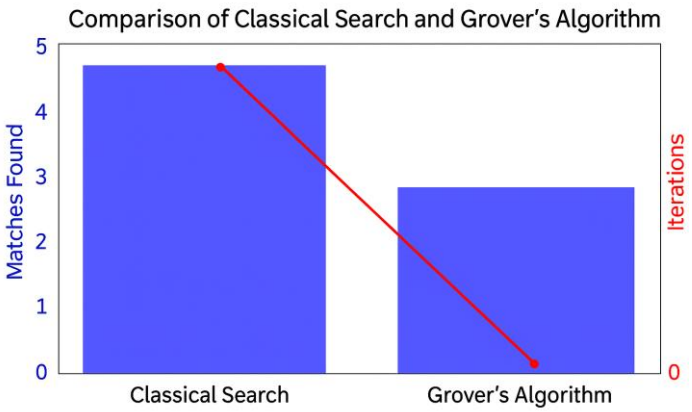

Figure 1. Comparison of Classical Search and Grover's Algorithm

Here is a comparison graph between classic search and Grover's algorithm, showing the number of matches found and the iterations required for a search with an 8-bit hash length. Left Axis (Blue): Displays the number of matches found by both methods. Right Axis (Red): Shows the number of iterations required by each method.

This graph shows the efficiency of Grover's algorithm in trimming the number of iterations needed to achieve results similar to the classical search method. Graph files available

E. Simulation Results

The classic search algorithm works by trying every possible input until it finds a match with the target hash. In this experiment, a classic search was performed over 10,000 iterations for an 8-bit hash, resulting in a random match according to its probability. Despite many iterations, the number of matches found remains limited, reflecting the high resource requirements in the brute force method.

Instead, Grover's algorithm leverages quantum principles such as superposition and probability amplification to reduce the number of iterations required to the square root of the total possible inputs, which is about 16 iterations for an 8-bit hash. Although the number of matches found is less, the efficiency of the iteration shows the great potential of this algorithm in solving search problems with fewer resources. The results of this simulation also provide insight into: Quantum Efficiency, Grover significantly reduces search complexity, making it a real threat to traditional hash-based cryptographic systems. Increased Security, To deal with these threats, solutions such as increasing hash length or using

post-quantum algorithms need to be adopted. Implementation Suitability, While efficient, future adoption of quantum algorithms will require major changes in computing infrastructure.

With these simulations, it is important to reassess the security of traditional hash-based systems and accelerate the transition to post-quantum technologies to protect data in the era of quantum computing. While 8-bit testing provides pedagogical clarity and theoretical validation, it does not fully capture real-world cryptographic security. Simulating 128- and 256-bit scenarios was computationally intensive and only partially explored. Future research should extend this model to higher bit lengths and real quantum hardware.

F. Comparison to Existing Studies

These findings align with literature such as Bernstein et al. (2015) and Lyubashevsky et al. (2010), which emphasize the urgency of post-quantum cryptography. However, our contribution adds value by implementing a repeatable quantum simulation model using Qiskit, offering practical insight beyond conceptual frameworks.

G. Practical Implications

This simulation framework provides foundational data for guiding developers and decision-makers in the adoption of post-quantum cryptography. It highlights the urgency for nations like Indonesia to prepare digital infrastructure that is secure against quantum threats by updating standards and investing in resilient technologies.

H. Implementation Challenges

Performance Trade-offs, Post-quantum solutions require more computing power, less ideal for real-time systems. Compatibility, Adapting to legacy systems requires major changes to the infrastructure. Standardization, Efforts such as NIST Post-Quantum Cryptography are essential to ensure uniform implementation.

I. Pseudocode simulation

This pseudocode provides a simple overview of the main structure and logic of Python code, separating the core functions, simulation logic, and visualization parts.

Define Functions for Hashing

Function hash_function(input_string): Input: A string. Output: SHA-256 hash of string.

Explanation

This function uses the SHA-256 hashing algorithm to generate a hash value from the input string. The hash is used to verify data by ensuring that each input produces a unique hash value (within the hash space).

Define an Oracle to Verify a Hash Match

Functions of oracles(target_hash): Input: Hash target. Output: A function to check if the string input is generating a target hash.

Explanation

This function takes a target hash and returns another function (a function within a function) that verifies if a given input string produces the same hash as the target. It acts as an "answer provider" for each guess in the search algorithms.

Implement Classic Search

Function classical_search(target_hash, n, iterations):

Inputs: Target hash, input length n, and number of iterations.

Output: Number of matches found.

Process: Repeat for the number of iterations, Generate random input along n bits. Check if the input hash matches the target hash using oracle.

Explanation

This function simulates hash searching using brute force. Efficiency: This search is slow as it has to try all possibilities. Security: For a hash length of n=8, there are 2n=256 possible inputs. Iterations: A high number of iterations increases the chance of finding a match.

Implement Grover's Algorithm

Grovers_algorithm function(target_hash, n, iterations): Inputs: Target hash, input length n, and number of iterations. Output: Number of matches found. Process: Calculate the number of experiments required: $\sqrt{2}n$, Repeat as many attempts as you want: Generate random input along n bits. Check for matches with target hashes using oracles.

Explanation:

Efficiency: Grover's algorithm is significantly faster than classical search. Iterations: For an 8-bit hash (n=8), it requires only about 16 attempts $\sqrt{256}=16$) compared to classical search, which may require up to 256 attempts.

Simulation and Comparison

Set simulation inputs:
- Input length n = 8
- Number of iterations for classic search =10,000
- Input an example password to generate the target hash.

Run classical_search and grovers_algorithm.
Store match count and iteration results.

A comparison between the classical search method and Grover's algorithm shows the significant efficiency of the quantum approach in finding hash targets under the same simulation conditions. Data visualization is presented in a table format covering two main aspects: the number of matches found and the number of iterations performed by each method. From the simulation results, it is evident that the classical method does produce more matches, but it requires a much larger number of iterations. Conversely, Grover's algorithm is able to find the target even with fewer matches, but only requires a fraction of the number of iterations needed by the classical method. This finding is visualized in a graph with two Y-axes, where the left axis shows the number of matches (in the form of bars), while the right axis displays the number of iterations (in the form of lines). The graph clearly illustrates that Grover's efficiency lies in the drastic reduction in the number of iterations, which is an important indicator in the context of computational efficiency, especially when applied to real-world scenarios with large search spaces. The analysis of the printed simulation results reinforces this visual interpretation, quantitatively demonstrating that quantum algorithm-based approaches offer more resource-efficient solutions compared to classical brute-force searches. This narrative strengthens the argument that quantum algorithms like Grover have significant potential in supporting the transition toward more robust information security systems in the era of quantum computing.

Conclusions

The conclusion of this research confirms that the real threat of quantum computing to the integrity of classical hash functions can be empirically proven through the application of Grover's algorithm. By leveraging the principles of superposition and amplitude amplification, this algorithm significantly reduces the search complexity from O(2^n) to O($\sqrt{2}$^n), which in an 8-bit hash simulation demonstrates a reduction in iterations from 256 (classical) to just 16 (quantum). This finding provides empirical evidence reproducing Grover's theoretical advantage over classical brute-force search, while reinforcing the urgency of transitioning to post-quantum cryptography. Strategic recommendations include

increasing hash length, implementing hash-based signature schemes such as SPHINCS+, and adopting lattice-based cryptography. From a practical standpoint, system developers and policymakers are advised to immediately enhance security protocols, support research and development of post-quantum algorithms, and invest in training and infrastructure. Governments are also encouraged to integrate post-quantum cryptography into national digital transformation strategies, including public infrastructure and defense systems. Although this research has limitations, particularly in simplified simulation environments and low-bit-length implementations, the results still provide a strong foundation for further research using simulations with higher bit lengths and testing on real quantum devices. Cross-disciplinary collaboration between cryptography, quantum physics, public policy, and systems engineering is key to accelerating global readiness to address quantum threats.

References

Agrawal, S., Libert, B., & Stehlé, D. (2016). Fully secure functional encryption for inner products, from standard assumptions. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9816*, 333–362. https://doi.org/10.1007/978-3-662-53015-3_12

Alladi, T., Chamola, V., Sahu, N., & Guizani, M. (2020). Applications of blockchain in unmanned aerial vehicles: A review. *Vehicular Communications*, *23*, 100249. https://doi.org/10.1016/j.vehcom.2020.100249

Amin, R., Islam, S. H., Biswas, G. P., Khan, M. K., & Kumar, N. (2018). A robust and anonymous patient monitoring system using wireless medical sensor networks. *Future Generation Computer Systems*, *80*, 483–495. https://doi.org/10.1016/j.future.2016.05.032

Benioff, P. (1980). The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *Journal of Statistical Physics*, *22*(5), 563–591. https://doi.org/10.1007/BF01011339

Bernstein, D. J., Hopwood, D., Hülsing, A., Lange, T., Niederhagen, R., Papachristodoulou, L., Schneider, M., Schwabe, P., & Wilcox-O'hearn, Z. (2015). SPHINCS: Practical stateless hash-based signatures. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9056*, 368–397. https://doi.org/10.1007/978-3-662-46800-5_15

Bernstein, D. J., & Lange, T. (2017). Post-quantum cryptography. *Nature*, *549*(7671), 188–194. https://doi.org/10.1038/nature23461

Bogdanov, A., Knežević, M., Leander, G., Toz, D., Varici, K., & Verbauwhede, I. (2011). Spongent: A lightweight hash function. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *6917 LNCS*, 312–325. https://doi.org/10.1007/978-3-642-23951-9_21

Cui, J., Zhang, J., Zhong, H., & Xu, Y. (2017). SPACF: A secure privacy-preserving authentication scheme for VANET with cuckoo filter. *IEEE Transactions on Vehicular Technology*, *66*(11), 10283–10295. https://doi.org/10.1109/TVT.2017.2718101

Damgård, I. B. (1990). A design principle for hash functions. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *435 LNCS*, 416–427. https://doi.org/10.1007/0-387-34805-0_39

Dobraunig, C., Eichlseder, M., Mendel, F., & Schläffer, M. (2021). Ascon v1.2: Lightweight Authenticated Encryption and Hashing. *Journal of Cryptology*, *34*(3), 1–42. https://doi.org/10.1007/s00145-021-09398-9

Du, W., Wang, R., & Ning, P. (2005). An efficient scheme for authenticating public keys in sensor networks. *Proceedings of the International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 58–67. https://doi.org/10.1145/1062689.1062698

Fernandez-Carames, T. M., & Fraga-Lamas, P. (2020). Towards Post-Quantum Blockchain: A Review on Blockchain Cryptography Resistant to Quantum Computing Attacks. *IEEE Access*, *8*, 21091–21116. https://doi.org/10.1109/ACCESS.2020.2968985

Fukuhara, M., & Kaji, S. (2021). Blockchain Basics. In *The Economics of Fintech*. https://doi.org/10.1007/978-981-33-4913-1_10

Gentry, C., Peikert, C., & Vaikuntanathan, V. (2008). Trapdoors for hard lattices and new cryptographic constructions. *Proceedings of the Annual ACM Symposium on Theory of Computing*, 197–206. https://doi.org/10.1145/1374376.1374407

He, D., Kumar, N., Zeadally, S., Vinel, A., & Yang, L. T. (2017). Efficient and Privacy-Preserving Data Aggregation Scheme for Smart Grid Against Internal Adversaries. *IEEE Transactions on Smart Grid*, *8*(5), 2411–2419. https://doi.org/10.1109/TSG.2017.2720159

Kiktenko, E. O., Pozhar, N. O., Anufriev, M. N., Trushechkin, A. S., Yunusov, R. R., Kurochkin, Y. V, Lvovsky, A. I., &

Fedorov, A. K. (2018). *Quantum-secured blockchain*.

Lyubashevsky, V., Peikert, C., & Regev, O. (2010). On Ideal Lattices and. *Advances in Cryptology – EUROCRYPT 2010*, *015848*, 1–23.

Lyubashevsky, V., Peikert, C., & Regev, O. (2013). On Ideal lattices and learning with errors over rings. *Journal of the ACM*, *60*(6), 1–35. https://doi.org/10.1145/2535925

M. N. Wegman, & J. L. Carter. (1981). New hash functions and their use in authentication and set equality. *Journal of Computer and System Sciences*, *22*, 265–279. http://www.sciencedirect.com/science/article/pii/0022000079900448%0Ahttps://linkinghub.elsevier.com/retrieve/pii/0022000079900448

Peris-Lopez, P., Hernandez-Castro, J. C., Estevez-Tapiador, J. M., & Ribagorda, A. (2006). M2AP: A minimalist mutual-authentication protocol for low-cost RFID tags. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *4159 LNCS*, 912–923. https://doi.org/10.1007/11833529_93

Preskill, J. (2018). Quantum computing in the NISQ era and beyond. *Quantum*, *2*(July), 1–20. https://doi.org/10.22331/q-2018-08-06-79

Regev, O. (2004). New lattice-based cryptographic constructions. *Journal of the ACM*, *51*(6), 899–942. https://doi.org/10.1145/1039488.1039490

SAHAI, A., & WATERS, B. (2021). How to use indistinguishability obfuscation: Deniable encryption, and more. *SIAM Journal on Computing*, *50*(3), 857–908. https://doi.org/10.1137/15M1030108

Stevens, M., Bursztein, E., Karpman, P., Albertini, A., & Markov, Y. (2017). The first collision for full SHA-1. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *10401 LNCS*, 570–596. https://doi.org/10.1007/978-3-319-63688-7_19

Supriati, R., Anjani, S. A., Anugrah, R. W., Mccarthy, R., Info, A., Cryptography, Q., Cryptography, P. Q., & Attacks, Q. (2025). *Enhancing Network Security with Quantum Cryptography : A Study on Future-Proofing Computer Networks Against Quantum Attacks*. *2*(1), 24–35.

Wang, W., Li, Z., Owens, R., & Bhargava, B. (2009). Secure and efficient access to outsourced data. *Proceedings of the ACM Conference on Computer and Communications Security*, 55–65. https://doi.org/10.1145/1655008.1655016

Zinzindohoué, J. K., Bhargavan, K., Protzenko, J., & Beurdouche, B. (2017). HACL.: A verified modern cryptographic library. *Proceedings of the ACM Conference on Computer and Communications Security*, 1789–1806. https://doi.org/10.1145/3133956.3134043