Applied Data Science and Analysis
Vol.2025, **pp**. 201–220
DOI: https://doi.org/10.58496/ADSA/2025/017 ; ISSN: 3005-317X
https://mesopotamian.press/journals/index.php/ADSA

Research Article

A Quantum Resilient Security System for Smart Power Grid Data: Combining Kyber, FALCON, and Zero-Knowledge Proofs Against Quantum Threats

Mishall Al-Zubaidie[1, *], Tuqa Ghani Tregi[1]

[1]*Department of Computer Sciences and Artificial Intelligence, Education College for Pure Sciences University of Thi-Qar, Nasiriyah 64001, Iraq*

ARTICLE INFO

Article History

Received 14 Jun 2025
Revised 15 Sep 2025
Accepted 30 Oct 2025
Published 07 Nov 2025

Keywords
Citizens' data privacy,
PQC,
Smart cities,
Electricity data security,
Quantum attacks.

ABSTRACT

The rapid progress of quantum computing poses significant challenges to traditional cryptographic mechanisms, necessitating the adoption of post-quantum cryptography (PQC) solutions. This paper proposes a Quantum-Enhanced Security for Smart Meters (QESM) system to protect power plant data in smart cities, integrating Kyber for secure key exchange, FALCON (Fast-Fourier Transform over Lattice-based Cryptography) for quantum-resistant digital signatures, and ZKP (Zero-Knowledge Proof) for effective verification without revealing sensitive data to secure power plant data against quantum attacks. To evaluate the security of the proposed system, we analyze its resistance to various quantum threats, including Shor's algorithm, Grover's algorithm, quantum key analysis, quantum reversal encryption, quantum amplification, quantum switching, and quantum collision attacks. In the current study, accurate measures were used and the average was approximately 7.065 (bits/byte) for randomness, the average execution time was 6.202 milliseconds, the average memory consumption was approximately 4.343 KB, 6.4 Completeness was equal to 1 and unforgeability was 100%. As for the average throughput, it was approximately 485,605 operations per second. That shows the QESM system provides strong security and efficiency, making it a viable solution for protecting the electricity infrastructure in smart cities in the quantum era.

1. INTRODUCTION

A smart city, while lacking a singular definition, is typically characterized as an urban area where services such as healthcare, transportation, agriculture, education, construction, and industrialization, along with policies, governance, infrastructure, and various activities, are predominantly automated. Fig. 1 illustrates a conventional smart city ecosystem. According to McKinsey [1], cities that use digital technologies can enhance the quality of life for residents by as much as 30% relative to residents in conventional urban areas. The meaning of an intelligent city differs in municipalities and on the national scale. In order to measure the intelligence of a certain city, both qualitative and quantitative measures will have to be used. These indicators depend on various aspects, such as the level of population, gross domestic product, literacy, financial and economic stability, and welfare of an individual [2]. The development of a smart-city ecosystem is characterized by numerous internal and external problems and limitations. With the growing number of smart cities all over the world, the security issue of security threats has heightened. Most of the sensors and data-analysis software, among other technologies that are fundamental to smart cities, are susceptible to cyber-attacks and other security lapses. These security problems can significantly influence the safety and well-being of the residents, not to mention the social and economic stability of a city [3].The risk of cyber-attacks is a major security issue in smart cities. Smart cities rely on interconnected networks and technologies that are easily susceptible to malware and hacking, as well as other cyber threats [4]. By affecting the integrity of data and systems, cyber-attacks can badly impact the safety of the population, power allocation, transportation, and other services [5]. Such attacks can also result in identity theft, besides compromising the privacy and personal information of the citizens.

*Corresponding author. Email: mishall_zubaidie@utq.edu.iq

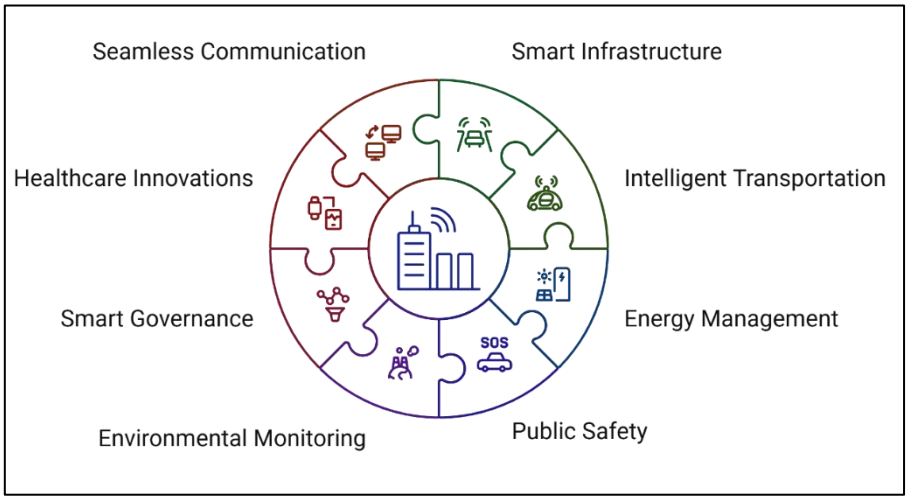

Fig.1. Typical smart city ecosystem.

Among the most essential security issues in smart cities is the risk of cyber-attack. The environments are susceptible to malware and hacking, among other cyber threats, because they are dependent on a network and technologies [6]. The level of threat to data and system integrity can cause severe harm to the security of the people, the power, transport, among other services. In addition to that, privacy and personal data of residents may be violated as a result of such attacks, and it may lead to identity theft [7]. The associated digital structure is complicated, which makes smart cities attractive to attacks and data theft and creates a large attack surface of cyber threats. Securities such as unauthorized data access and violation of privacy are widespread. The problem of data security emerges when artificial intelligence (AI) is used in urban planning. The traditional security systems are usually overwhelmed by the fast-changing environment of cyber threats in smart cities. The variety of interconnected systems and the lack of available security measures complicate the implementation of proper cybersecurity measures [8]. The protection of data within smart cities, therefore, requires a complex security strategy that comprises post-quantum cryptography (PQC) and real-time threat detection. In an attempt to curb cyber-attacks, this paper discusses the possible vulnerabilities and prescribes innovative security measures. The following challenges have to be identified and addressed to ensure that smart cities are safe, resilient, and able to satisfy their promise of a more intelligent and secure future. The main contributions of this paper are as follows:

1. Provides a robust key exchange technology via Kyber, thus reducing the risk of quantum key analysis-based attacks.
2. Based on a lightweight FALCON-based signature, reduces resource usage for low-power devices.
3. Verifies digital signatures without revealing them based on ZKP, significantly reduces processing and storage time, thus enhancing the security of smart energy systems without sacrificing speed.

We give an overview of the research organization here. A comprehensive introduction is provided in Section 1. We review recent studies on related work in Section 2. The history of quantum signatures and pseudonyms is described in Section 3. A research technique for electrical station data protection is presented in Section 4. Section 5's our proposed was examined using a security analysis. In Section 6, the performance analysis is described in full. Section 7 presents the findings of the investigation.

2. RELATED WORK

This section provides an overview of attempts to secure data and smart cities. As Liu et al. [9] stated, an innovative information-exchange system based on blockchains was introduced into a Zero Trust Architecture (ZTA). The offered approach was proven to be effective in terms of performance analysis on Ethereum-based blockchain platforms, which illustrates the security in the context of universal composability. However, this method is hindered by a delay in communication and the cost of the computation protocol. Moya et al. [10] presented ASPMI, an adaptive anti-spam system through which devices are able to offer proof-of-work at a price or delegate it to the network at the cost of providing resiliency and security. Nonetheless, blockchain technology can be used in large networks, but it is expensive to maintain high storage. Panneerselvam and Rajiyakodi [11] used end-to-end encryption to have smart cities ensuring that data privacy is maintained when passing between smart devices. Cyber-physical systems (CPS) convert the data to a digital format and then transmit the data to the computational parts. CPS Data analytics is a method of transmitting data, its statistical analysis, machine learning, and data mining to process data and make predictions, although it may encounter obstacles due to delays in data

transmission. A methodology introduced by Jaganraj and Srinivasan [12] uses deep learning models to improve the identification of cyber-attacks and user privacy protection. The approach uses a novel training technique that makes use of various distributions, thus enhancing the resistance of the model to multiple attacks. Deep attention networks are able to understand detailed patterns in Internet of Things (IoT) traffic information. However, they consume large processing power, which may make them inappropriate for a range of smart devices. Alkhudhayr et al. [13] attempted to mitigate cyber-physical risks in IoT-enabled smart transportation infrastructure by means of a machine learning (ML)-based intrusion detection system (IDS) targeting a cyber-physical system (CPS) of the SDC that effectively identifies and mitigates attacks on the physical components associated with the SDCs. A key feature of the SDC-CPS architecture is the integration of a control area network into the simulator associated with the SDC. But the difficulty lies in extending the scope of security to include all the different intelligent transportation systems. Ahmed et al. [14] proposed a multi-task model that executes distinct person re-identification and forecasts attributes. The model employs a shared base network, either ResNet50 or EfficientNet, in conjunction with generalized mean pooling (GeM) for feature extraction. It further uses feature-specific vertices to forecast several attributes, including gender, age, clothing style, and colour, in conjunction with ReID categorization. Nonetheless, it may encounter the challenge of elevated expenses associated with the operation and analysis of data from many sources in real time. Walunj et al. [15] examined the incorporation of cryptocurrency payment systems into Web 2.0 services, with particular emphasis on their use within the data marketplace, Indian Urban Data Exchange (IUDX), and Agricultural Data Exchange (ADEX) platforms. Primarily, they face regulatory hurdles in financial law, which hinder their implementation. Sahu et al. [16] proposed a secure consumer network model based on blockchain technology to harness green energy and meet the electricity demand of Internet of Vehicles in smart cities. Smart Parking Model (SPM) serves as a platform for customers and electric automobiles. Nonetheless, the scalability of blockchain technology and the increased latency in secure communication networks for automobiles remain challenges. Gulzar et al. [17] introduced a DeepCLG hybrid learning model designed to improve network intrusion detection systems (NIDS). The datasets first undergo preprocessing and normalization. Then, a hybrid learning model called DeepCLG is designed. It integrates a convolutional neural network (CNN), long short-term memory (LSTM), gated recurrent unit (GRU), and a capsule network (CN). However, hybrid models require a lot of computing power. A comparison of earlier research on data privacy and unresolved issues in smart cities is shown in Table I.

TABLE I. COMPARISON OF PREVIOUS STUDIES ABOUT SMART CITY DATA PROTECTION

Author's Name	Techniques and Algorithms Used	Study Proposal	Unsolved Problems
Liu et al. [9] 2022	Blockchain, ZTA and Fair Incentive Model	blockchain-based information exchange system under ZTA and Fair Incentive Model	technique is hindered by communication delay and protocol cost in computation
Moya et al. [10] 2024	Distributed Ledger Technologies (DLTs) for Spam Protection in IoT Networks	Leverages Distributed Ledger Technologies (DLTs) to enhance spam protection in IoT smart environments.	Security vulnerabilities in early implementations of DLT-based spam filtering solutions.
Pannerselvam et al. [11] 2024	End-to-End Encryption for Cyber-Physical Systems in Smart Cities	Utilizes an end-to-end encryption approach to ensure security in cyber-physical systems.	Computational complexity in maintaining encryption across large-scale smart city infrastructures.
Jaganraja et al. [12] 2024	Deep Learning Privacy-Preservation Model for Cybersecurity in IoT Smart Cities	Proposes a privacy-preserving deep learning approach for detecting cybersecurity threats in IoT networks.	Potential trade-offs between privacy preservation and deep learning model performance.
Alkhudhayr et al. [13] 2025	Cyber-Physical Security for IoT-enabled Smart Transport Infrastructure	Develops security solutions for IoT-integrated smart transport networks to mitigate cyber-physical risks.	Scalability issues in securing large-scale IoT networks within smart transport systems.
Ahmed et al. [14] 2025	Multi-task Machine Learning Model for Person Re-identification in Smart Cities	Introduces an AI-driven multi-task learning model for person re-identification in smart city surveillance.	High computational costs associated with AI-driven surveillance models.
Walunj et al. [15] 2025	Crypto-Based Payment Systems for Secure Data Marketplaces in Smart Cities	Investigates the integration of cryptocurrency for secure data transactions in urban infrastructures.	Regulatory concerns and practical adoption barriers in crypto-payment smart city frameworks.
Sahu et al. [16] 2025	Blockchain-based Secure Communication Model for Smart City Electric Vehicles	Develops a blockchain-based model to improve security in vehicle-to-vehicle communication within smart cities.	Blockchain scalability and high latency issues in secure vehicle communication networks.
Gulzar et al. [17] 2025	Hybrid Deep Learning Model for Industrial IoT Security	Implements a hybrid deep learning model for enhanced intrusion detection in Industrial IoT environments.	Optimization of deep learning models for real-time industrial IoT security applications.

3. BACKROUND

In this section, we will review the scientific background of the most important topics that we will address in our current study.

3.1 Post Quantum Cryptography

PQC denotes encryption techniques that are resilient to assaults from quantum computers. Conventional public-key cryptosystems, including those reliant on integer factorization (e.g., RSA) and discrete logarithm problems (e.g., DSA, DH), are susceptible to quantum computing, particularly to Shor's method, which may resolve these issues with efficiency. In 1994, Peter Shor demonstrated that a sufficiently advanced quantum computer might compromise these encryption techniques, underscoring this vulnerability [18]. In reaction to this impending threat, the National Institute of Standards and Technology (NIST) has initiated a process to standardize quantum-resistant encryption methods. In July 2022, NIST designated Crystals-Kyber as the principal key encapsulation mechanism (KEM) for standardization [19]. Crystals_Kyber and four more algorithms (BIKE, traditional McEliece, HQC, and bike) were chosen for further assessment and potential standardization in subsequent rounds [20].

TABLE II .IMPACT OF GROVER'S AND SHOR'S ALGORITHMS ON CRYPTOGRAPHIC SCHEMES.

Cryptographic Schemes	Algorithm (security level)	Type	Impact of Grover's Algorithm	Impact of Shor's Algorithm
Symmetric Cryptography	AES-128 (128-bit) AES-256 (256-bit)	Block Cipher	Halve the security measures	No impact
	Salsa20 (256-bit)	Stream Cipher	Halve the security measures	No impact
	SHA-256 (256-bit) SHA-3 (256-bit)	Hash Function	Halve the security measures	No impact
Public-Key Cryptography	RSA-2048 (112-bit) RSA-3072 (128-bit)	Encryption/Signature	No Impact	Broken
	Diffie-Hellman-2048 (112-bit) Diffie-Hellman-3072 (128-bit)	Key Exchange	No Impact	Broken
	ECDSA-256 (128-bit) ECDH-256 (128-bit) DSA-3072 (128-bit)	Signature	No Impact	Broken
PQC	Kyber-512 (128-bit) Kyber-768 (192-bit) Kyber-1024 (256-bit)	KEM	No Impact	Quantum Resistant
	Dilithium-2 (128-bit) Dilithium-3 (192-bit) Dilithium-5 (256-bit) SPHINCS+ (256-bit) FALCON-512 (128-bit) FALCON-1024 (256-bit)	Digital Signature	No Impact	Quantum Resistant
	Poly1305 (128-bit) GMAC (128-bit)	MAC	No Impact	Quantum Resistant

Table II shows the Impact of Grover's and Shor's Algorithms on Cryptographic Schemes. The security in symmetric encryption relies on the secrecy of the hash functions or keys. Grover's algorithm reduces the number of attempts needed to break the encryption from 2N to 2N/2, thus speeding up the search process. This means that it halves the security but

does not completely destroy the encryption. While it cannot affect public key algorithms [21,22], as Shor's algorithm does with asymmetric encryption [23]. The challenge of factoring or calculating discrete logarithms drives asymmetric encryption [24]. Quantum computing allows Shor's algorithm to destroy these systems completely, that begin to become extremely insecure if sufficiently powerful quantum computers are built.

3.2 CRYSTALS-Kyber

CRYSTALS-Kyber (Kyber) is a crucial encapsulation technology designed to resist quantum attacks. Kyber is built on the intricate challenges of lattice-based encryption, which is currently considered secure against quantum computing threats [25]. Kyber works as a Key Encapsulation Mechanism (KEM) and is used to generate a shared encryption key between two parties securely. It is part of the CRYSTALS (Cryptographic Suite for Algebraic Lattices) project and was proposed within the NIST PQC standard [26]. Kyber has three primary stages:

- **Key generation:** Each party generates encrypted keys to secure the communication and then creates a random matrix A from a finite mathematical field $\boldsymbol{Zq}$. Then a secret vector $\boldsymbol{s}$ and a small noise vector $\boldsymbol{e}$ are chosen. Public key $\boldsymbol{pk}$ is calculated using the equation: $\boldsymbol{t = A * s + e}$.

Algorithm 1 explains Kyber key generation. It is worth noting that there is no external input, as safe random value generators within the method provide keys at random.

***Algorithm 1**: Kyber Key Generation*

Input: None
Output: pk, sk

1. *A ← Sample_Matrix*
2. *s, e ← Sample_Noise*
3. *t ← A * s + e*
4. *pk ← (A, t (*
5. *sk ← s*
6. *Return (pk, sk)*

- **Key encapsulation:** A shared key $\boldsymbol{K'}$ is chosen at random, then a random vector r is chosen and noise $\boldsymbol{e1}$ and $\boldsymbol{e2}$ are added. The ciphertext $\boldsymbol{c = (u, v)}$ is calculated from the equations: $\boldsymbol{u = A^T * r + e1}$ and $\boldsymbol{v = t^T * r + e2 + Encode(K').}$ Then $\boldsymbol{c}$ is sent to the other part, the final Shared key $\boldsymbol{K}$ is extracted using a hash function: $\boldsymbol{K = Hash(K')}$.

Algorithm 2 represents the PK encapsulation process to extract the ciphertext and the shared key.

***Algorithm 2:** Kyber Key encapsulation*

Input: *pk*
Output: *c*, *K*

1. *r, e1, e2 ← Sample_Noise*
2. *u ←AT * r + e1*
3. *v ←tT * r + e2 + Encode(K')*
4. *c ← (u, v)*
5. *K ←Hash(K')*
6. *Return (c, K)*

- **Decapsulate the key:** The receiver uses $\boldsymbol{s}$ to decrypt the key where $\boldsymbol{K' = Decode\ (v - s^T * u)}$, final key $\boldsymbol{K = Hash(K')}$.

Algorithm 3 shows the decryption of Kyber key capsule based on $\boldsymbol{c}$,$\mathbf{s}$ [27, 28]. While Fig. 2 illustrates the Kyber key exchange process from generating keys to terminating the connection.

Algorithm 3: Kyber key Decapsulate

Input*: c, s*
Output*: K*

1. *K' ← Decode (v - s^T * u)*
2. *K ← Hash(K')*
3. *Return K*

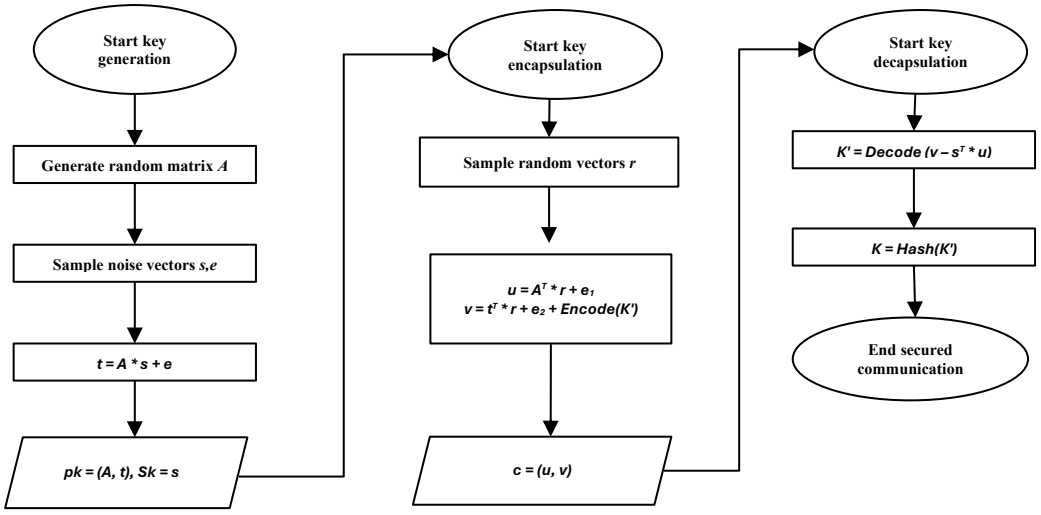

Fig. 2. Kyber KEM process flowchart.

3.3 Falcon

FALCON is a DSA (digital signature algorithm) that utilizes the Gentry-Peikert-Vaikuntanathan (GPV) framework to construct hash-and-sign Lattice-Based Cryptography (LBC) [29]. FALCON depends on the A family of Nth Degree Truncated Polynomial Ring Units (NTRU) lattices, which use a trapdoor sampler that integrates the efficacy of Klein's algorithm with the efficiency of Peikert's method. FALCON demonstrates the minimal aggregate of the public key and signature size among the NIST PQC methods. Sikeridis [30] contended that FALCON is appropriate for online applications if a floating-point hardware module is present at the server, whereas Bindel [31] evidenced its suitability for secure vehicle-to-vehicle (V2V) communication owing to its minimal fundamental safety message packet size. When contemplating the reuse of a key, the frequency of signature creation and verification is comparatively elevated among these three processes. Consequently, prioritizing the acceleration of signature production and verification may prove to be the most efficacious approach. In contrast to signature verification, which possesses a rather straightforward algorithmic framework and has been extensively explored for acceleration by several researchers [32,33], investigations into the acceleration of signature production are quite few. FALCON comprises three procedures: key generation, signature generation, and signature verification [34,35].

- **FALCON Key Generation:** First, generates short polynomials ***f, g***. Then computes the NTRU basis ***(F, G)***. Creates and transforms the basis matrix B to the Fast Fourier Transform (FFT) space $\hat{\boldsymbol{B}}$ and stores ***pk = g/f*** (compressed) and ***sk = (f, g, F, G,*** $\hat{\boldsymbol{B}}$***)***.

Algorithm 4 explained FALCON Key Generation Based on Secret Parameter and NTRU.

Algorithm 4: FALCON Key Generation

Input*: Security parameter n*
Output*: pk, sk*

1. *(f, g) ← Sample_Short_Polynomials(n)*
2. *(F, G) ← Compute_NTRU_Basis(f, g)*
3. *B ← Construct_Basis_Matrix(F, G)*
4. $\hat{B}$ *← FFT(B)*
5. *pk ← Compress (g / f)*
6. *sk ← (f, g, F, G,* $\hat{B}$*)*
7. *Return (pk, sk)*

- **FALCON Signature**: This stage begins with hashing the message ***m*** using a random salt ***r*** to obtain challenge ***c***. Then computes the intermediate vector ***t*** in Fourier space by using Fast Fourier Sampling (ffSampling) to generate a valid signature ***s***.

Algorithm 5 clarifies the signing process. It begins by generating r to prevent signature reuse, then ***m*** is hashed and mapped to a lattice point to obtain ***c***. Next, ***t*** is computed in the Fourier space using FFT optimizing signature operations. The ***ffSampling*** technique is applied to choose a short vector *z*, and this is modified as ***s=(t−z)*** $* \hat{\boldsymbol{B}}$ and at the same time, *s*

satisfies the necessary security threshold. After validation, the signature is converted from the time domain back to the frequency domain with the help of the Inverse FFT, and then the signature is compressed to minimize its size. The last step is the signature *sig = (r, s)* is returned.

Algorithm 5: FALCON Signature

Input*: m, sk*
Output*: sig*

1. *r ← Generate_Random_Salt ()*
2. *c ← HashToPoint(r || m)*
3. *t ← (-FFT(c) * FFT(F) / q, FFT(c) * FFT(f) / q)*
4. *Repeat:*
   a. *z ← ffSampling_DynTree(t, G)*
   b. *s ← (t - z) * B̂*
5. *Until $||s||^2 \leq \beta^2$*
   a. *(s1, s2) ← IFFT(s)*
   b. *s ← Compress(s2)*
6. *Return sig = (r, s)*

- **FALCON verification**: During this step, *c* is rebuilt with the help of the hashed message and *r*, and the resultant calculated vector *v* is verified against the base threshold required by this number, and the result is valid or invalid. The algorithm that will verify a digital signature or a digital signature of authenticity verification is algorithm 6, which uses lattice-based cryptography. It starts by removing the *r* and the sign in the signature given. The *c* is then reconstructed by hashing the message along with the *r* together. The signature has now been deconstructed. Assuming the norm condition is met, the signature is taken to be a valid signature; in the opposite case, it is an invalid signature.

Algorithm 6: FALCON Verification

Input*: m, sig, pk*
Output*: "Valid" or "Invalid "*

1. *(r, s) ← sig*
2. *c ← HashToPoint(r || m)*
3. *s ← Decompress(s)*
4. *v ← FFT(s) * FFT(g/f) - FFT(c)*
5. *If $||v||^2 \leq \beta^2$: Return "Valid "*
6. *Else : Return "Invalid"*

3.4 Zero-Knowledge Proof

One of the most significant developments in cryptographic verification is Zero-Knowledge Scalable Transparent Argument of Knowledge (zk-STARK), where a prover is able to prove the correctness of a statement by getting the cryptographically verifiable without disclosing any additional information [36]. Zero-Knowledge Proofs (ZKPs) had originally been published by Goldwasser, Micali, and Rackoff in 1985 [Goldwasser et al., 1985] and are extended here [37]. Being more resistant to quantum attacks and allowing greater visibility in the initial setup, zk-STARKs have emerged as a more preferable option over zk-SNARKs.

Algorithm7: NIZK-STARKs Prover(x)

Input*: $x \in F$*
Output*: π, C*

1. *f(x) ← Arithmetize(x)*
2. *g(x) ← FRI(f(x))*
3. *C ← Commitment(g(x))*
4. *r ← Hash(C)*
5. *π ← Generate_Proof(C, r, f(x))*
6. *Return (π, C)*

zk-STARKs Prover is a verifier that generates a non-interactive proof that does not contain any knowledge of the verification procedure [38]. In algorithm 7, a finite field *F* is first given, which transforms the input data *x* into a polyn poisson form; *f (x)*. This transformation allows using algebraic means of validation rather than simple arithmetic operations. *f(x)* is computed using Fast Reed-Solomon Interactive (FRI) Oracle Proof technique, thus making it efficient, and it gives a low-degree Poisson *g(x)*. This reduces the size of the proof and the cost of computation, although it does not violate the integrity of the original function. Subsequently, the dedication to *g(x)* is produced by cryptographic hash functions, typically organized in a Merkle tree to ensure the integrity of data and prevent undesirable modifications [39]. The Fiat-Shamir rule eliminates the need for a challenge created by the verifier by zk-STARKs. Rather, hash the commitment ***C*** to produce a

random challenge value r deterministically, hence rendering the proof non-interactive. The verifier constructs a cryptographic proof π using ***C, r***, and ***f(x)***, thereby encapsulating the data required for verification. Then the proof ***π*** and the commitment ***C***, which the verifier may utilize to independently confirm the validity of the proof devoid of direct contact with the verifier. This method guarantees scalability, post-quantum security, and efficiency, which qualify ZK-STARKs for use in smart cities aiming at maintaining anonymity [21,22]. Figs. 3, and 4 illustrate the Prover and Verification process, while Fig. 5 illustrates the complete non interactive ZK-STARKs (NIZK-STARKs) flowchart.

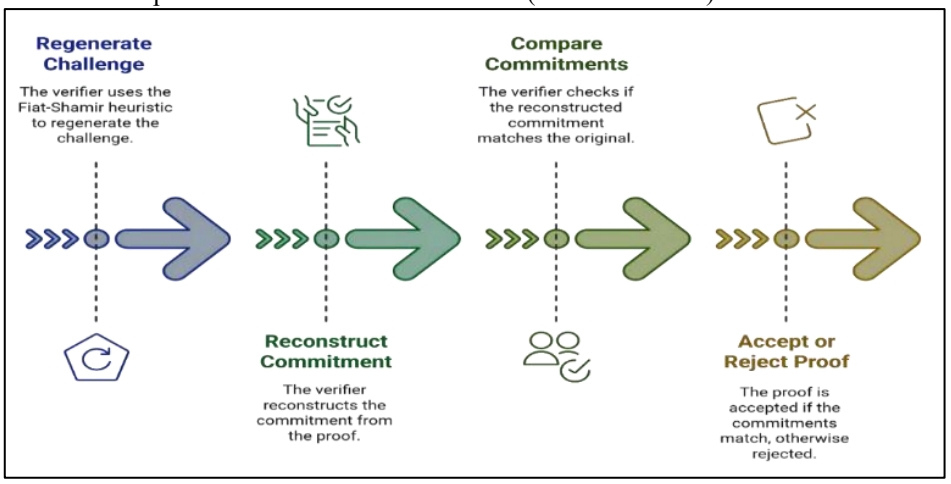

Fig. 3. NIZK-STARKs prover process.

Algorithm 8 independently validates the proof ***π*** by regenerating the challenge ***r*** using the Fiat-Shamir heuristic. It then reconstructs the commitment ***C'*** from the proof and checks if it matches ***C***. If they are equal, the proof is accepted as valid; otherwise, it is rejected to ensure non interactive verification.

Algorithm 8: NIZK-STARKs Verifier (π, C, Parameters)

Input*: π, C, params*
Output*: "Valid" or "Invalid "*
1. *r ← Hash(C)*
2. *C' ← Verify_Proof (π, r, Parameters)*
3. *Return "Valid" if C' = C, else "Invalid"*

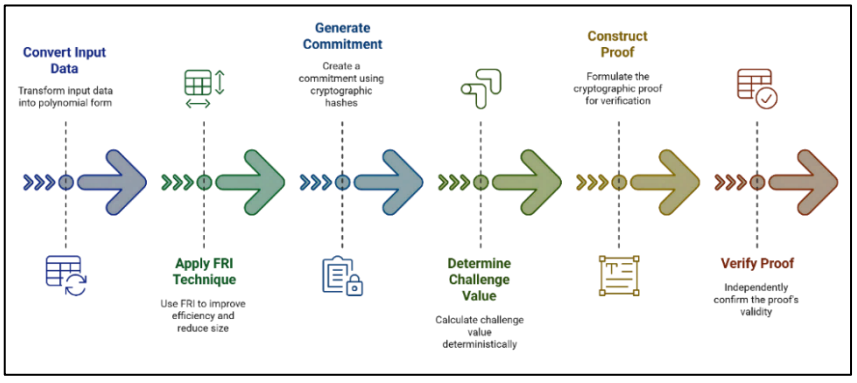

Fig. 4. NIZK-STARKs verification process.

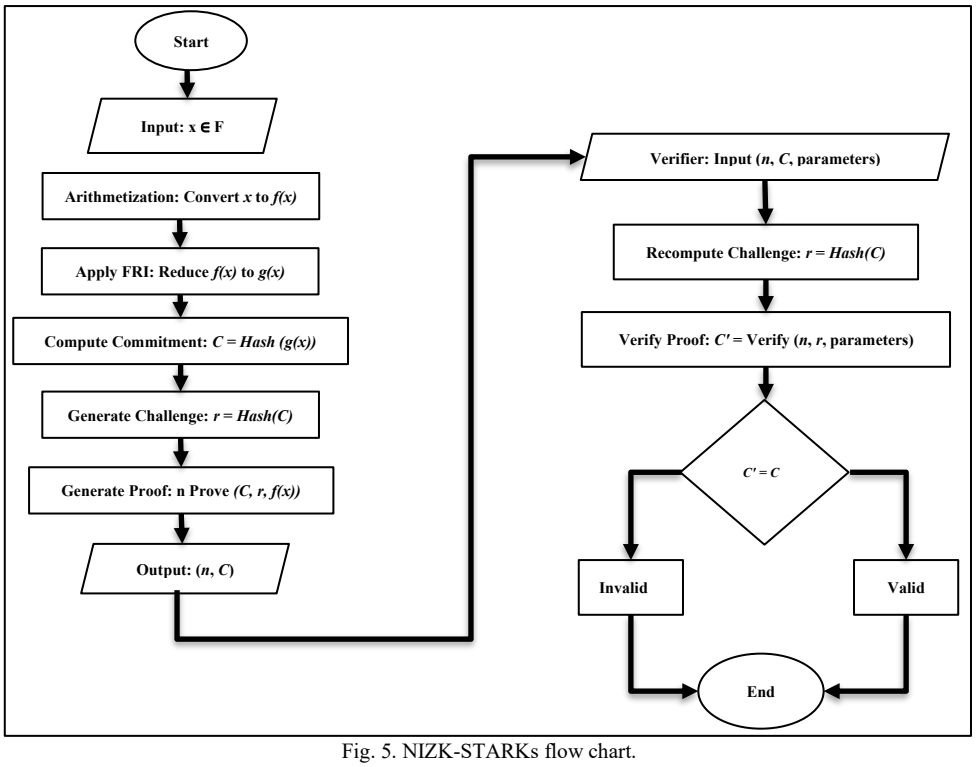

Fig. 5. NIZK-STARKs flow chart.

4. QUANTUM ATTACK RESISTANT METHODOLOGY

In this section, we introduce our proposed QESM system, which uses post-quantum technologies to safeguard smart power plant data by means of a secure and adaptable design. Three main components make it up: ZK-STARKs to validate user identity and electronic payment information without disclosing any private information, safe key exchange between smart meters and central servers using Kyber, and data signing using FALCON to guarantee data integrity before transmission. Starting from its collection in smart meters, this design offers thorough data protection by securing it during transmission to verify its authenticity level using ZK-STARKs before processing, enabling the system to effectively address conventional and quantum threats. Fig. 6 illustrates the general approach of our proposed system.

4.1 Post-Quantum Cryptographic Key Exchange for Secure Data Communication

In smart cities, it is vital to ensure the safety of information between smart meters and central servers of power equipment. Smart meters continuously send sensitive electricity usage information to grid control centers, and conventional key swap protocols (RSA and Diffie-Hellman) have been known to be susceptible to quantum attack, thus unsuitable in the long term infrastructure security. To deal with this issue, we suggest implementing Kyber to create a common encryption key between the smart meters and the grid servers in the electric station. FALCON is used to encrypt the signature with this shared key and zk-STARK is used to validate zero-knowledge proofs. Kyber facilitates establishing a secure shared key between the smart meter and the server without necessarily transferring the secret key between the smart meter and the server. The main steps of exchange occur when there is a key exchange:

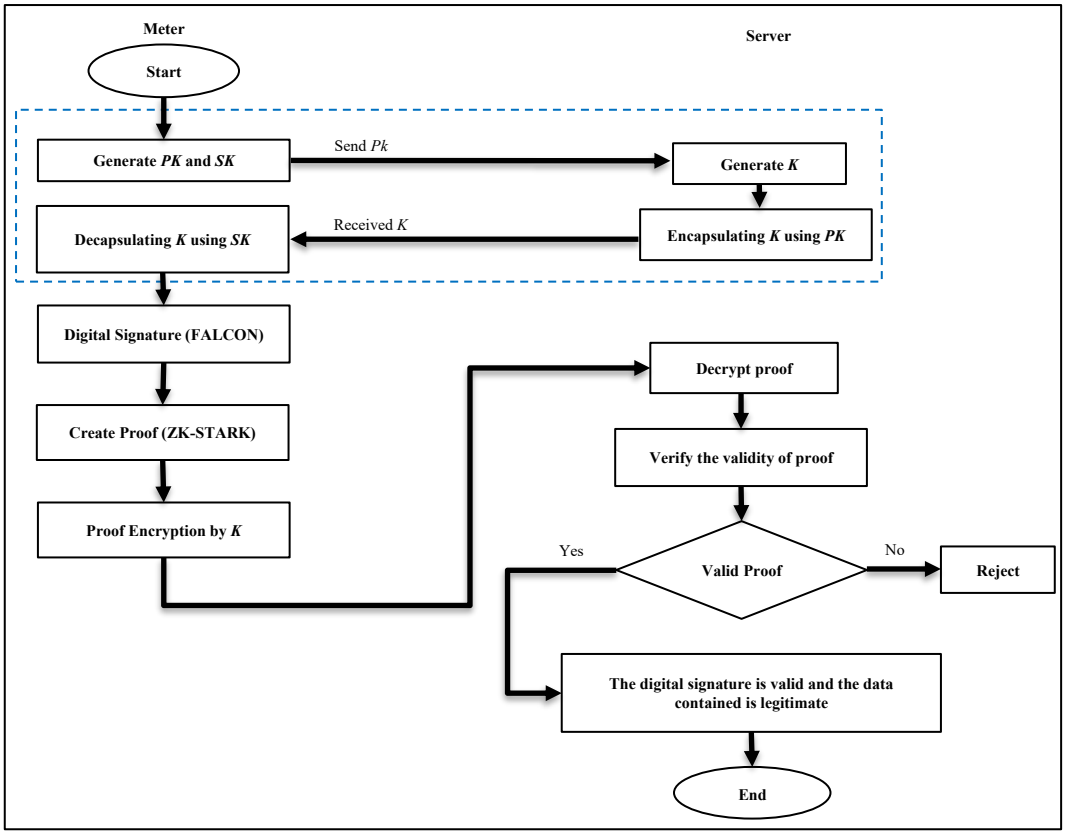

Fig.6. The proposed QESM system.

- The smart meter generates public and secret keys *PK, SK* sends the public key to the server and keeps the private key.
- The server generates and encapsulates a shared key *K* using the smart meter's public key and sends the encapsulated key back to the meter.
- The smart meter decapsulates the *K* using the *SK*.

Algorithm 9 illustrates the key exchange process between the smart meter and the server. Fig. 7 represents the Key Exchange Process.

Algorithim 9: *Kyber Keys Exchange*

Input*: []*
Output*: K*

1. *Began*
2. *(PK, SK) ← Kyber.KeyGen()*
3. *Store SK securely in SmartMeter*
4. *Send PK to Server*
5. *(ciphertext, K_server) ← Kyber.Encapsulate(PK)*
6. *Send (ciphertext) to SmartMeter*
7. *K_meter ← Kyber.Decapsulate(ciphertext, SK)*
8. *If K_meter == K_server Then Key Exchange Successful and Secure Communication Established*
9. *Else Key Exchange Failed*
10. *End*

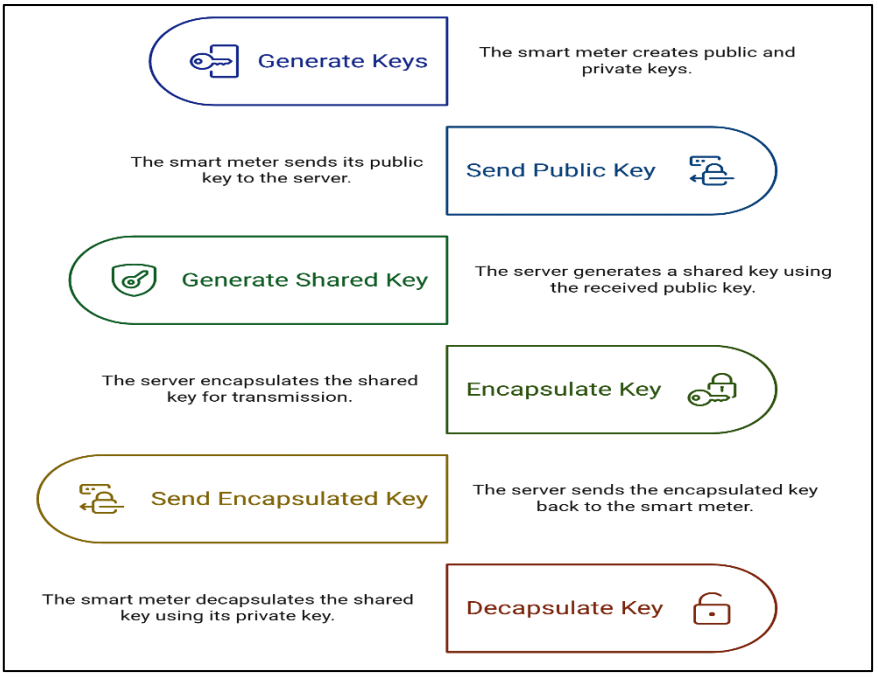

Fig.7. Key exchange process.

4.2 Signing Data Using FALCON

The integrity of data in smart grids is needed when relaying the electricity consumption data between smart meters and servers at the power plant. The data should be made sure that the information is reliable and has not been distorted. This enhances billing, energy management, and load balancing reliability in smart cities. In order to bring this degree of security, FALCON has been incorporated into the system to offer a lightweight, quantum-resistant digital signature mechanism to secure smart city infrastructure in the long term. FALCON ensures that only authentic smart meters can sign energy data and that power grid servers can verify the authenticity of the received data. Since FALCON signatures must remain confidential, a ZKP technique is used to generate a proof of FALCON signature and not send the signature. This prevents unauthorized eavesdropping and replay attacks. The FALCON mechanism of our proposed system is as follows: Before forwarding digital energy consumption reports to the server, whose job is to verify the signature to ensure that the data comes from a valid smart meter, each smart meter digitally signs its energy consumption reports. The server then verifies the validity of these signatures before running any action without seeing the signature or being exposed by ZK-STARK. Algorithm 10 illustrates the process of signing data using FALCON.

Algorithm 10: FALCON Signature

Input*: FALCON_SK, data*
Output*: σ \\ signature*

1. *Began*
2. *σ← FALCON.Sign(FALCON_Sk, data)*
3. *Return σ*
4. *End*

4.3 Non Intreactive ZK-STARK

ZK-STARKs allow digital signatures to be verified without revealing them, therefore securing smart power plant data. By use of a mathematical proof derived from the conversion of the verification process into a polynomial representation, the signature σ is proved legitimate rather than shared with other parties. ZK-STARKs enable the verification procedure. This is accomplished by creating evidence enabling the verification of the signature without direct view of it. The evidence is encrypted using the created shared key from Kyber KEM, therefore providing even another degree of security. The person having the shared key alone can confirm the signature. Smart meters may have limited processing capability; hence, the ZK-STARK-Prove procedure is only carried only when data has to be transmitted. Central servers can be used for verification, therefore lessening the computational demand on smart meters.

Algorithm 11: ZK_STARK_Generate Proof

Input*: K, m, σ, FALCON_PK*
Output*: π \\ proof*

1. *Began*
2. *Verify FALCON(m, σ, FALCON_PK) ⇔ A * FALCON_Sk ≡ u (mod q)*

3. *P(x) = 1 if Verify_FALCON(m, σ, pk) holds*
4. *P(x) = 0 otherwise*
5. *r = H(K || m)*
6. *FRI_Proof = FRI_Test(P(x))*
7. *π = (H, FRI_Proof, Challenge_Values)*
8. *π′ = Encrypt(π, K)*
9. *RETURN π′*
10. *End*

Algorithm 11 demonstrates generating a zero-knowledge proof without the need to transmit sensitive data over communication channels. It starts with inputs *K* (shared key generated from Kyber KEM), *m*, *σ*, *FALCON_PK*, and then defines the FALCON signature verification equation as $A * s \equiv u \pmod q$, where A is a public matrix of the FALCON lattice structure and *u* represents the generated signature value, *q* prime modulus for modular arithmetic. Next, we enter the second stage, which converts the verification into polynomial form *P(x)* so that the proof can be constructed and verified quickly, where *P(x)* is a polynomial over a finite field *Fq*. Then, Fiat-Shamir challenges by computing a random challenge: $r = H(K \| m)$. By applying the FRI protocol for low-degree verification, generate proof for the low-degree check: *FRI_Proof = FRI_Test(P(x))*. After creating the ZK-STARK proof: $\pi = (H, FRI\_Proof, Challenge\_Values)$, the proof is encrypted using $K : \pi' = Encrypt(\pi, K)$.

Algorithm 12: ZK_STARK Verify Signature

Input*: K, m, π′, FALCON_PK*
Output*: TRUE if the signature is valid, FALSE otherwise*

1. *Began*
2. *π = Decrypt(π′, K)*
3. *Extract (H, FRI_Proof, Challenge_Values) from π*
4. *H′ = MerkleRoot(H(P(x_i)))*
5. *IF H′ ≠ H THEN RETURN FALSE*
6. *IF FRI_Verify(FRI_Proof) = FALSE Then Return FALSE*
7. *r′ = H(K || m)*
8. *IF r′ ≠ Challenge_Values Then Return FALSE*
9. *End*

Algorithm 12 shows the process of verifying this proof after decrypting it without revealing sensitive data.

Algorithm 13: QESM_Smart_Meter

Input*: Sensor data*
Output*: π′, data*

1. *Began*
2. *(PK, SK) ← Kyber.KeyGen()*
3. *(FALCON_PK, FALCON_SK) ← FALCON.KeyGen()*
4. *data ← Read_Sensor()*
5. *σ ← FALCON.Sign(data, FALCON_SK)*
6. *π ← ZKSTARK.Generate(σ, data, FALCON_SK)*
7. *π′ ← Kyber.Encrypt(π , PK)*
8. *Transmit (data, π′) to Server*
9. *End*

Algorithm 14: QESM_Server

Input*: π′, data*
Output*: Accept, Reject*

1. *Began*
2. *Receive (data, π′)*
3. *π ← Kyber.Decrypt(π′, SK)*
4. *Valid ← ZKSTARK.Verify(π, data, FALCON_PK)*
5. *If Valid: Accept(data)*
6. *Else: Reject(data)End*

Algorithms 13 and 14 illustrate how QESM takes a robust approach to ensuring the security of data sent from smart meters to the central server in a smart city power plant.

5. SECURITY ANALYSIS

In this section, we demonstrate security analysis through attack analysis and testing of the Scyther verification tool.

5.1 Attack Analysis

- **Shor's algorithm**: It is a quantum algorithm that poses a major threat to modern cryptography because it is able to efficiently solve two mathematical problems: integer factorization and discrete logarithms, which underpin many asymmetric (public-key) encryption schemes. Our proposed QESM system is able to repel this type of attack because it uses a quantum-resistant encryption and digital signature algorithm.

- **Grover's Algorithm**: A quantum algorithm provides a significant speedup for solving problems involving brute-force searches. It threatens symmetric encryption schemes and hash functions in cryptography by reducing the time required to guess secret keys. Kyber and FALCON are not affected, as they use asymmetric network-based encryption.
- **Quantum Key Analysis:** This is the process of cryptographic key analysis by quantum computing methods. It is directly connected to the wider area of quantum cryptanalysis, which examines the way quantum computers can subvert conventional cryptographic systems. In the case of the QESM system, the keys are generated within the smart meters by Kyber, which makes them extremely hard to extract.
- **Quantum Reverse Encryption Attack**: An attack that reconstructs the message by using quantum computers to attempt to reverse the encryption. The ZK-STARKs technology used in QESM does not rely on reversible encryption, but on non-interactive proofs, making it resistant to this attack.
- **Quantum Amplification Attack**: It works by amplifying the probability of finding secret keys using quantum effects. FALCON relies on lattice cryptography and does not rely on simple linear structures that can be amplified quantumly.
- **Quantum Switch Attack**: Exploiting quantum superposition to manipulate data during transmission without detection. Using ZKP prevents undetected data manipulation.
- **Quantum Collision Attack**: This attack exploits quantum computing to find hash function collisions much faster than classical methods. FALCON does not rely on hash functions to generate signatures, but rather uses difficult problems in lattice mathematics, such as NTRU Lattices, which are not affected by this attack.

Table III provides a comparison between the proposed QESM and various previous systems in repelling quantum attacks.

TABLE III. COMPARISON OF ATTACK PREVENTION BETWEEN THE QESM SYSTEM AND SIMILAR SYSTEMS.

Attacks	[40]	[41]	[42]	[43]	[44]	[45]	[46]	QESM
Shor's algorithm	✓				✓	✓	✓	✓
Grover's Algorithm				✓			✓	✓
Quantum Key Analysis	✓	✓	✓		✓	✓		✓
Quantum Reverse	✓	✓	✓	✓		✓		✓
Quantum Amplification		✓		✓	✓			✓
Quantum Switch		✓	✓		✓	✓	✓	✓
Quantum Collision	✓		✓		✓			✓

5.2 Theoretical Performance Analysis

With a security mechanism aimed at protecting the electrical data of the smart grid from quantum risks, the QESM system works on post-quantum cryptography. Even with the increasing obstacles of quantum computing, the proposed system integrates three basic techniques (Kyber, FALCON, and ZKP). Kyber and FALCON rely on mathematical problems in the field of lattice problems, such as the Learning with Noise (LWE) problem and the NTRU problem, which are difficult to solve even using known quantum algorithms to ensure data integrity and confidentiality because they are designed to be unbreakable by the Shor algorithm, as there is no known quantum algorithm that solves lattice problems efficiently. The impact of attacks such as Grover's Algorithm is reduced by choosing parameters that provide a high level of security equivalent to or exceeding 256 bits in classical cryptography. ZKP techniques enable the verification of data to be performed without disclosing the information content, which ensures a high degree of privacy and the computation efficiency that is required in the smart grid environment. The capability of the system to counter a strong quantum attack, including the use of the Shor algorithm, Grover algorithm, and quantum collision attacks, is assessed by conducting a thorough security analysis. The findings demonstrate that the system can resist the tremendous strength of quantum computers. This not only enhances privacy but also assists in enhancing the efficiency of computation in smart grid systems, as one will be able to verify the electrical data safely without exposing the individual information. The suggested solution will be a significant step to safeguard smart grids against quantum risks in the present and future. With the algorithms employed, the system can achieve a good trade-off between the resistance of quantum attacks and the efficiency of execution because the algorithms are supported by solid mathematical proofs that the algorithms are hard to crack and therefore the system is applicable in smart grid application where the security is needed to be very high at the same time maintaining the latency to be very low.

5.3 Scyther as an Analysis Tool

Scyther is an automated formal analysis tool that has evaluated and verified several traditional authentication mechanisms [47]. Scyther uses SPDL to clarify the responsibilities within the protocol and define the associated operations. In modeling the authentication protocol using Scyther, we first use the role sequence as parameters, which includes three roles: Meter, user, and server. Next, we define the sending and receiving events, as shown in Figs. 8 and 9. By defining these roles and events, we can formalize the authentication process in Scyther. All roles in the protocol use Alive, Weakagree, Commit, and Secret claims to declare and verify the security of the created and received variables. Scyther will validate the added associated properties.

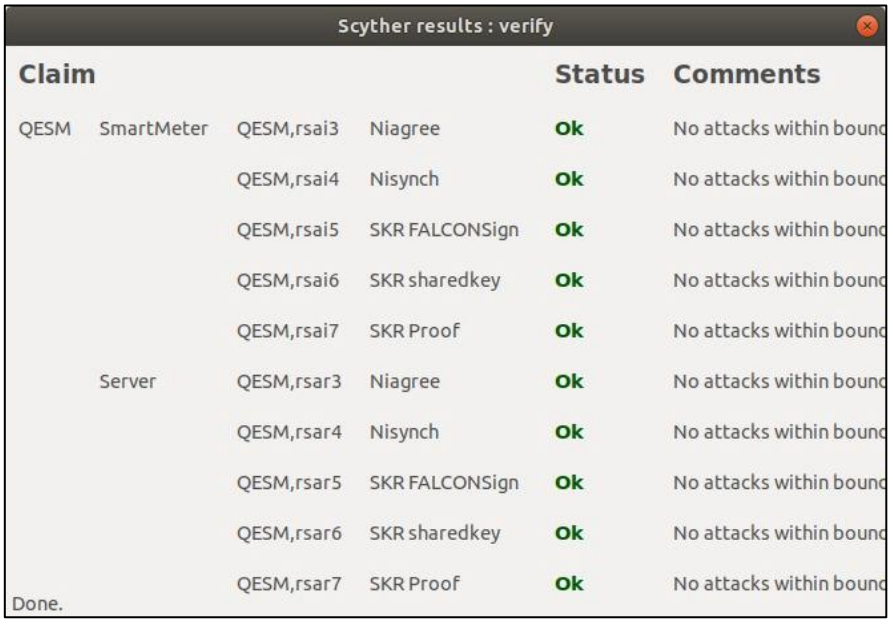

Fig.8. Depiction of the Scyther results.

Fig.9. Proposed QESM system summary of the characterized roles.

6. PERFORMANCE EXPERIMENT RESULTS

In this section, we present the experimental results for evaluating the performance of Kyber, FALCON, and zk-STARK based on key security and efficiency metrics, including randomness, determinism, stability, scalability, complexity, completeness, and traceability. Using the configuration items shown in Table IV, the mentioned algorithms were executed up to 50 times.

TABLE IV. EXPERIMENTAL ENVIRONMENT CONFIGURATION TABLE.

Configuration Items	Configuration Details
OS	Ubuntu 18.04.6 LTS
CPU	Intel(R) Core (TM) i5-1135G7 CPU, 9th generation
RAM	and 8.00 MB
Disk capacity	512GB, SSD

Development environment	Eclipse IDE 2023-06 (4.28)
Programming Language Version	Java, OpenJDK 11.0.19

6.1 Randomness (Entropy)

A mathematical measure of randomness in a given data set to evaluate the complexity and strength of cryptographic keys, randomness in generating digital signatures, and proof for each of Kyber, FALCON, ZKP in the proposed system, we calculated entropy using the Shannon equation:

$$H(X) = -\sum_{i=0}^{n} P(x_i) log_2 P(x_i) \qquad (1)$$

The results were shown in Table V, which shows the amount of entropy for the techniques used in the QESM system. As for Figs. 10, 11 and 12, they show the entropy values for repeating 50 times.

TABLE V. AVERAGE ENTROPY OF THE TECHNIQUE USED IN THE QESM SYSTEM.

Technique	**Average Entropy (bits/byte)**
Kyber	~ 6.645
FALCON	~ 6.005
ZKP	~ 7.065

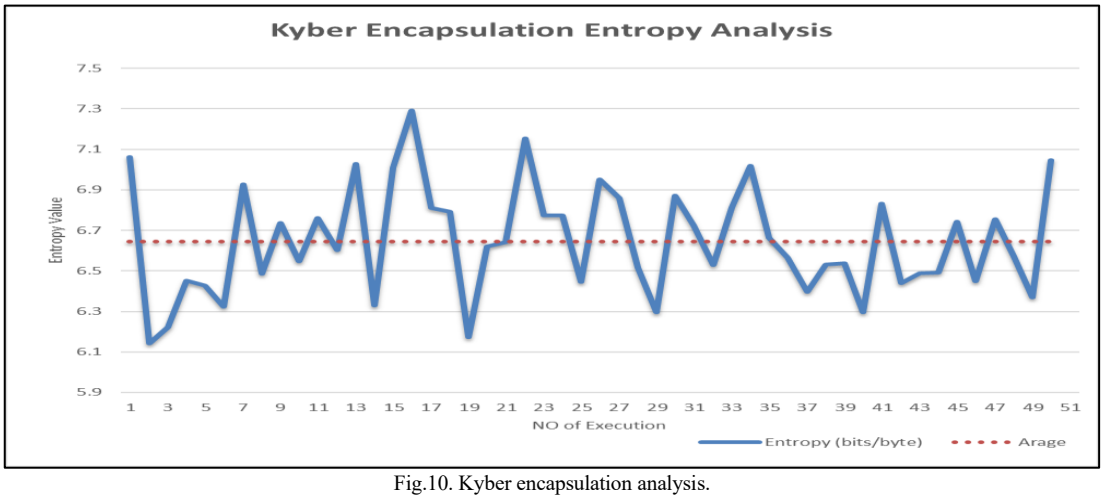

Fig.10. Kyber encapsulation analysis.

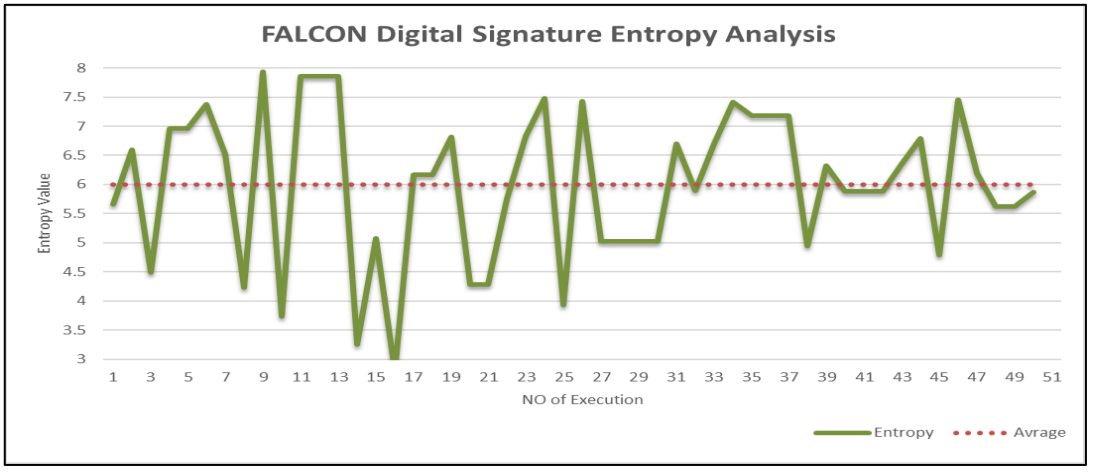

Fig.11. FALCON digital signature encapsulation analysis.

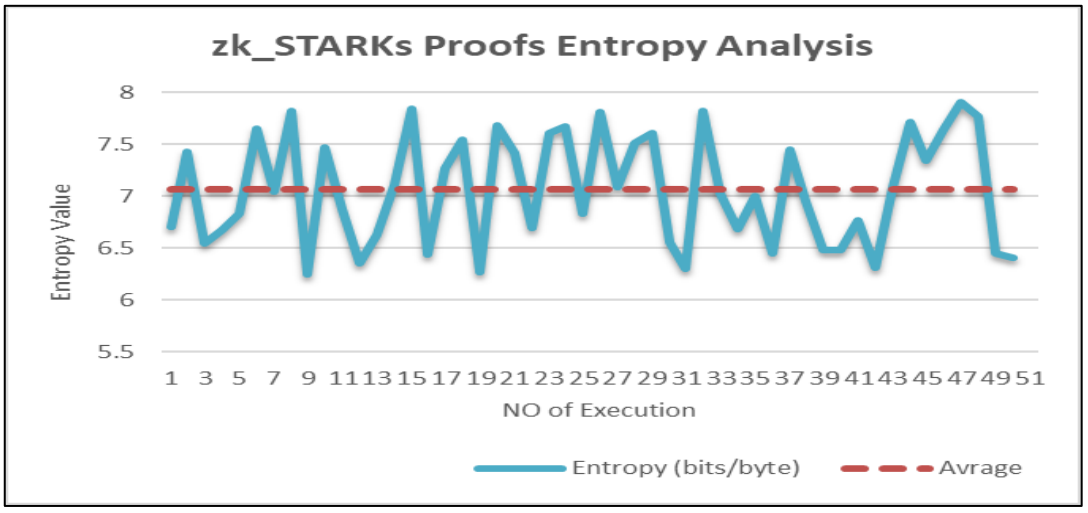

Fig.12. ZK_STARKs proofs entropy analysis.

6.2 Execution Time

It is the time taken to complete each process. We achieved good results in calculating the execution time of each of the following processes, as in Table VI, which shows the average time required by the processes from the beginning of execution until the completion of the work assigned to it. Figs. 13, 14 and 15 show the execution times of the 50 operations.

TABLE VI. THE AVERAGE TIME REQUIRED BY THE ACTIVITY OF THE QESM SYSTEM.

Process	Average Execution Time (ms)
Kyber key generation	2.144
Kyber encapsulation	1.103
Kyber decapsulation	1.135
FALCON signing	0.152
FALCON verification	0. 161
ZK_STARKs verification	0.077026
QESM	6.202

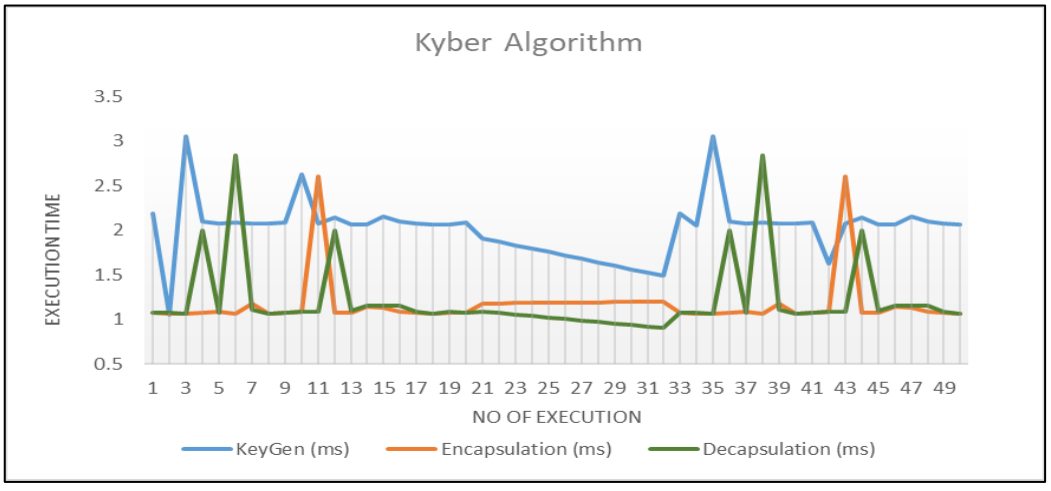

Fig.13. Execution time of Kyber algorithm.

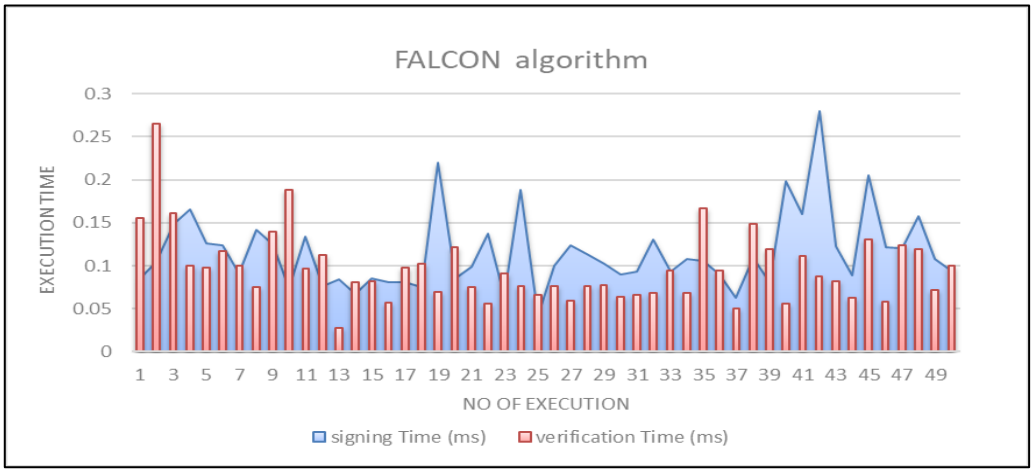

Fig.14. Execution time of FALCON algorithm.

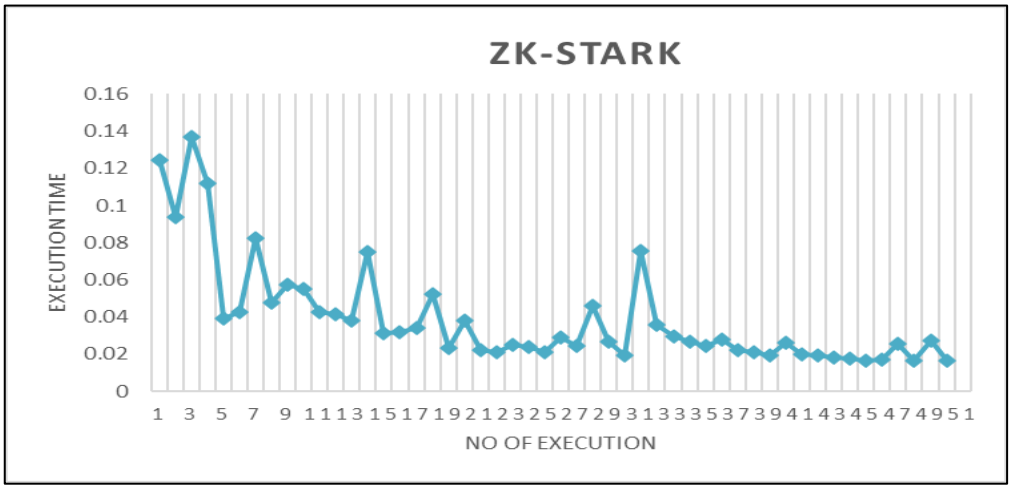

Fig.15. Execution time of Proof ZK_STARKs.

6.3 Memory Usage

The amount of RAM used by processes during the execution of a specific task is distinguished by our proposed system in its low memory consumption, which makes it light during its execution and does not hinder the network performance, thanks to the FALCON algorithm, which is considered one of the effective algorithms with low memory consumption, as well as the Kyber algorithm. As for ZK_STARK, it is non-interactive, which means that the proof is verified only once when exchanging data, and the connection does not need to resend the proof several times, which leads to reducing the load on the RAM. The memory consumption rate when executing FALCON completely is approximately 0.019 KB, while when executing Kyber, the rate is approximately 2 KB, and when executing ZK_STARK, the average size of the proofs consumed is approximately 64 bytes. As for the total consumption rate of the QESM system, it is approximately 4.343 KB. The System.gc() function in Java is used to measure memory consumption.

6.4 Completeness

It measures the system's ability to handle all possible situations without losing data. When testing the QESM system, all results were correct and no data loss occurred during key exchange or in data signing and information validation processes without accessing sensitive data, and the Completeness metric achieved 1 every time. Fig. 16 depicts the Completeness measure for the proposed QESM operations.

$$Completeness = \frac{Total\ number\ of\ cases\ that\ should\ be\ processed}{Number\ of\ successfully\ processed\ cases} = \frac{50}{50} = 1 \qquad (2)$$

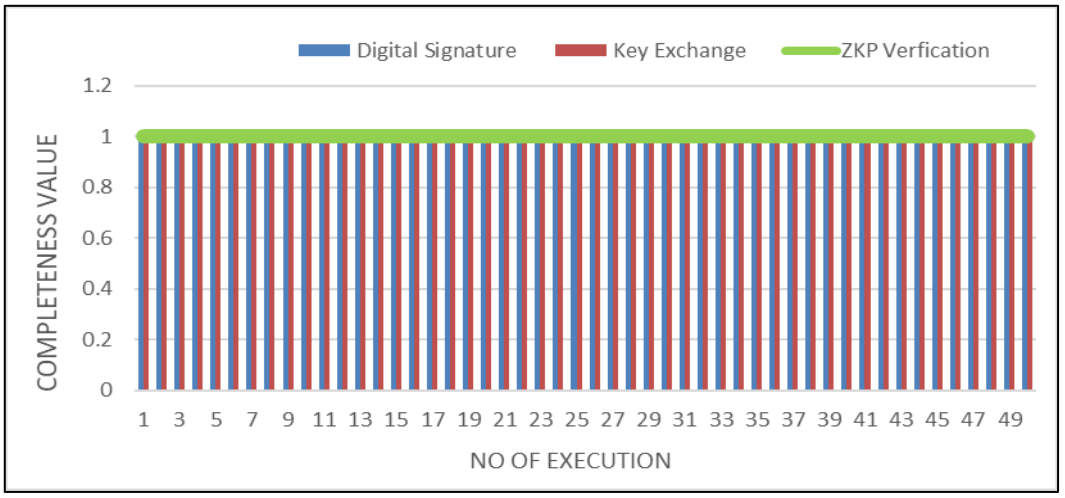

Fig.16. Completeness measure for QESM system operations.

6.5 Scalability

It measures the ability of a system to handle an increase in cryptographic operations without negatively impacting performance. A scalable cryptographic system should be able to support an increasing number of transactions with a reasonable response time. In the QESM system, we calculated the time taken to execute operations from 10 to 5000 operations. Where the time taken for 10 operations (T10) is 9.742061 milliseconds, while for 5000 operations (T5000) it is 4.676429 milliseconds.

$$S = \frac{T(10)}{T(5000)} = \frac{9.742061}{4.676429} = 2.08 \qquad (3)$$

6.6 Unforgeability

A fundamental security feature of digital signature systems. It ensures that without access to the legitimate secret key, an adversary cannot produce a valid signature. Only authorized entities may generate valid signatures under a tamper-resistant cryptographic system; therefore, an attacker cannot forge them even if he has access to many signatures. In a QESM system, the result is that the system is 100% tamper-proof.

$$Failure\ Rate = \frac{Failed\ Operations}{Total\ Operations} \times 100\% = \frac{0}{50} \times 100\% = 0\% \qquad (4)$$

$$Unforgeability = 1 - \left(\frac{Successful\ Forgeries}{Total\ Attempts}\right) \times 100\% = 1 - \left(\frac{0}{50}\right) \times 100\% = 100\% \qquad (5)$$

6.7 Throughput

It is a key performance measure that expresses the number of operations a system can process in a given unit of time. The QESM system is capable of processing approximately 485,605 operations per second, which is a very high performance compared to many traditional systems. Fig. 17 depicts the proposed QESM throughput. Table VII shows that Kyber outperforms traditional key exchange protocols in its resistance to quantum attacks. Table VIII highlights the advantages of FALCON with its relatively small signature size as well as its robustness in resisting quantum network attacks when compared to other signature algorithms.

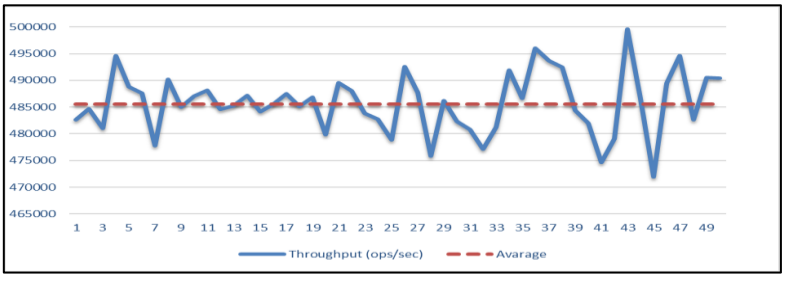

Fig.17. QESM system throughput.

TABLE VII. COMPARING KYBER TO OTHER KEY EXCHANGE PROTOCOLS.

Criteria	[48]	[48,49]	
	Kyber	**RSA-2048**	**Diffie-Hellman (DH-2048)**
Encryption Type	Lattice-Based	Public-Key Encryption	Modular Arithmetic-based Key Exchange
Quantum Resistance	Yes	No	No
Encryption Speed	faster than RSA and DH	Slower	Slower
PK	≈ 800–1200 bytes	≈ 256 bytes	≈ 256 bytes
SK	≈ 2400 bytes	≈ 256 bytes	≈ 256 bytes
Data Exchange Size	≈ 1 KB	≈ 2 KB	≈ 2 KB

TABLE VIII. COMPARING FALCON TO OTHER DIGITAL SIGNATURE ALGORITHMS.

Criteria	[50,51]		
	FALCON 512	**Rainbow (a1)**	**Dilithium2**
Type	Lattice-based	Multivariate	Lattice-based
PK Size	897 Bytes	161600 Bytes	1312 Bytes
SK key	1281 Bytes	103648 Bytes	2528 Bytes
Signature Size	Very Small	Small to Medium	Medium to Large
Quantum Resistance	Yes	Vulnerable to Some Attacks	Yes
Mathematical Complexity	NTRU	Multivariate Polynomials	LWE (Learning with Errors)
NIST Status	Approved	Not Approved	Approved

7. CONCLUSIONS

Smart city data security is a global necessity, as smart cities exchange data from various applications such as smart power grid data. Some of these applications transmit sensitive data. Hackers who get access to such data may make smart city services useless. Our proposed QESM system combines Kyber to facilitate key exchange and FALCON to perform digital signatures and zk-STARKs to carry out zero-knowledge verification to protect sensitive data of smart power plants. We achieve the goals of quantum risk insecurity and computational efficiency. With the help of Kyber, the key exchange between smart meters and central servers can be made secure and quantum-resistant, so there is no longer any risk of key distribution systems of the past. On the other hand, FALCON offers light and strong digital signatures to secure data integrity. Besides, zk-STARKs reduces the risk of repeated attacks, as it allows checking signatures without revealing the real signature to improve privacy. Since the theoretical analysis and practical evaluation have demonstrated high security to quantum attacks, our method is suitable in the real-time usage on smart grid setting. In further research, the concept will be expanded to other infrastructure in a smart city.

8. FUTURE DIRECTIONS

We plan to develop our proposed QESM system by:

1. Supporting the QESM by taking advantage of a blockchain-based key management protocol (BlkKM) to provide stability and transparency in Kyber.
2. Supporting the QESM for finding the optimality through the entropy-based sunflower optimization algorithm.
3. Supporting the QESM system to define the access of users and smart power grid employees by relying on fine-grained access control, such as attribute-based access control, to ensure that legitimate employees have access to databases stored within attribute-based levels.

Conflicts of Interest

The authors declare no conflicts of interest.

Funding

There is no funding.

Acknowledgment

Non.

References

[1] W. A. Jebbar and M. Al-Zubaidie, “Transaction-based blockchain systems security improvement employing micro-segmentation controlled by smart contracts and detection of saddle Goatfish,” *SN Comput. Sci.*, vol. 5, no. 7, pp. 1–23 (2024). doi: 10.1007/s42979-024-03239-9.

[2] D. Cook and B. D. Sdottir, “An appraisal of interlinkages between macro-economic indicators of economic well-being and the sustainable development goals,” *Ecol. Econ.*, vol. 184, p. 106996 (2021). doi: 10.1016/j.ecolecon.2021.106996.

[3] M. Al-Zubaidie, "A critical reliable model for promoting patient medical wireless sensors information security," *Procedia Comput. Sci.*, vol. 263, pp. 17–24 (2025). doi: 10.1016/j.procs.2025.07.003.

[4] R. Sharma and R. Arya, “Security threats and measures in the Internet of Things for smart city infrastructure: A state of art,” *Trans. Emerg. Telecommun. Technol.*, vol. 34, no. 11, p. e4571 (2023). doi: 10.1016/j.iot.2022.100584.

[5] M. Al-Zubaidie and W. A. Jebbar, "Blockchain-powered dynamic segmentation in personal health record," *Mesopotamian J. Cybersec.*, vol. 5, no. 3, pp. 953–976 (2025).

[6] Ö. Aslan, S. S. Aktuğ, M. Ozkan-Okay, A. A. Yilmaz, and E. Akin, “A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions,” *Electronics*, vol. 12, no. 6, p. 1333 (2023). doi: 10.3390/electronics12061333.

[7] R. H. Razzaq, M. Al-Zubaidie, and R. G. Atiyah, “Intermediary decentralized computing and private blockchain mechanisms for privacy preservation in the internet of medical things,” *Mesopotamian J. Cybersec.*, vol. 4, no. 3, pp. 152–165 (2024). [Online]. Available: https://doi.org/10.58496/MJCS/2024/020.

[8] M. K. Hasan, A. A. Habib, Z. Shukur, F. Ibrahim, S. Islam, and M. A. Razzaque, “Review on cyber-physical and cyber-security system in smart grid: Standards, protocols, constraints, and recommendations,” *J. Netw. Comput. Appl.*, vol. 209, p. 103540 (2023). doi: 10.1016/j.jnca.2022.103540.

[9] Y. Liu *et al.*, “A blockchain-based decentralized, fair and authenticated information sharing scheme in zero trust Internet-of-Things,” *IEEE Trans. Comput.*, vol. 72, no. 2, pp. 501–512 (2022). doi: 10.1109/TC.2022.3157996.

[10] F. Moya, F. J. Quesada, L. Martínez, and F. J. Estrella, “ASPMi: An adaptable SPAM protection mechanism for IoT scenarios,” *in Int. Conf. Ubiquitous Comput. Ambient Intell.* Springer, pp. 909–919 (2024). doi: 10.1007/978-3-031-77571-0_87.

[11] K. Pannerselvam and S. Rajiakodi, “Towards smarter, interconnected futures: The crucial role of data in cyber-physical systems,” *in Intell. Cyber-Physical Syst. Healthc. Solut.* Springer, pp. 181–194 (2024). doi: 10.1007/978-981-97-8983-2_9.

[12] V. Jaganraja and R. Srinivasan, “An agile solution for enhancing cybersecurity attack detection using deep learning privacy-preservation in IoT-smart city,” *Wireless Netw.*, pp. 1–16 (2024). doi: 10.1007/s11276-024-03876-1.

[13] H. Alkhudhayr and H. Ardah, “Mitigating cyberphysical risks in IoT-enabled smart transport infrastructure,” *J. Supercomput.*, vol. 81, no. 2, p. 446 (2025). doi: 10.1007/s11227-025-06948.

[14] M. F. Ahmed and A. A. N. Oyshee, “Multi-task model with attribute-specific heads for person re-identification,” *Pattern Anal. Appl.*, vol. 28, no. 1, p. 38 (2025). doi: 10.1007/s10044-025-01421-0.

[15] V. Walunj, V. Rajaraman, J. Dutta, and A. Sharma, “Integrating crypto-based payment systems for data marketplaces: Enhancing efficiency, security, and user autonomy,” *in Int. Conf. Inf. Syst. Secur.* Springer, pp. 443–452 (2025). doi: 10.1007/978-3-031-80020-7_25.

[16] B. L. Sahu and P. Chandrakar, “Blockchain-oriented secure communication and smart parking model for internet of electric vehicles in smart cities,” *Peer-to-Peer Netw. Appl.*, vol. 18, no. 1, pp. 1–17 (2025). doi: 10.1007/s12083-024-01872-y.

[17] Q. Gulzar and K. Mustafa, “Enhancing network security in industrial IoT environments: A DeepCLG hybrid learning model for cyberattack detection,” *Int. J. Mach. Learn. Cybern.*, pp. 1–19 (2025). doi: 10.1007/s13042-025-02544-w.

[18] T. G. Tregi and M. Al-Zubaidie, “Enhancing traffic data security in smart cities using optimized quantum-based digital signatures and privacy-preserving techniques,” *Mesopotamian J. Cybersec.*, vol. 5, no. 1, pp. 256–272 (2025). [Online]. Available: https://doi.org/10.58496/MJCS/2025/017.

[19] M. Iavich and T. Kuchukhidze, “Investigating CRYSTALS-Kyber vulnerabilities: Attack analysis and mitigation,” *Cryptography*, vol. 8, no. 2, p. 15 (2024). doi: 10.3390/cryptography8020015.

[20] G. Alagic *et al.*, “Status report on the third round of the NIST post-quantum cryptography standardization process,” National Institute of Standards and Technology (2022). doi: 10.6028/NIST.IR.8413.

[21] B. Khanal, J. Orduz, P. Rivas, and E. Baker, “Supercomputing leverages quantum machine learning and Grover’s algorithm,” *J. Supercomput.*, vol. 79, no. 6, pp. 6918–6940 (2023). doi: 10.1007/s11227-022-04923-4.

[22] R. H. Razzaq *et al.*, "Sturdy blockchain combined with e-apps repositories based on reliable camouflaging and integrating mechanisms," *IJ Comput. Netw. Inf. Secur.*, vol. 17, no. 3, pp. 35–53 (2025). doi: 10.5815/ijcnis.2025.03.03.

[23] S. Sonko, K. I. Ibekwe, V. I. Ilojianya, E. A. Etukudoh, and A. Fabuyide, “Quantum cryptography and us digital security: A comprehensive review: Investigating the potential of quantum technologies in creating unbreakable encryption and their future in national security,” *Comput. Sci. IT Res. J.*, vol. 5, no. 2, pp. 390–414 (2024). doi: 10.51594/csitrj.v5i2.790.

[24] R. H. Razzaq and M. H. Al-Zubaidie, “Maintaining security of patient data by employing private blockchain and fog computing technologies based on internet of medical things,” *Informatica*, vol. 48, no. 12 (2024). doi: 10.31449/inf.v48i12.6047.

[25] M. Al-Zubaidie, *Securing Smart Cities Through Modern Cryptography Technologies*. IGI Global (2025). ISBN: 9798337333267. [Online]. Available: https://books.google.iq/books?id=bCVk0QEACAAJ.

[26] H. Bandara, Y. Herath, T. Weerasundara, and J. Alawatugoda, “On advances of lattice-based cryptographic schemes and their implementations,” *Cryptography*, vol. 6, no. 4, p. 56 (2022). doi: 10.3390/cryptography6040056.

[27] A. H. Al-Tameemi *et al.*, "A Systematic review of metaverse cybersecurity: Frameworks, challenges, and strategic approaches in a quantum-driven era," *Mesopotamian J. Cybersec.*, vol. 5, no. 2, pp. 770–803 (2025). [Online]. Available: https://doi.org/10.58496/MJCS/2025/045.

[28] Y. Xing and S. Li, "A compact hardware implementation of CCA-secure key exchange mechanism CRYSTALS-KYBER on FPGA," *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, pp. 328–356 (2021). doi: 10.46586/tches.v2021.i2.328-356.

[29] J. Wang *et al.*, "Quantum-safe cryptography: crossroads of coding theory and cryptography," *Sci. China Inf. Sci.*, vol. 65, no. 1, p. 111301 (2022). doi: 10.1007/s11432-021-3354-7.

[30] D. Sikeridis, P. Kampanakis, and M. Devetsikiotis, "Post-quantum authentication in TLS 1.3: A performance study," *Cryptology ePrint Arch.* (2020). doi: 10.14722/ndss.2020.24203.

[31] N. Bindel, S. McCarthy, H. Rahbari, and G. Twardokus 3rd, "Suitability of 3rd round signature candidates for vehicle-to-vehicle communication," *in Workshop Rec. Third PQC Stand. Conf.* (2021). [Online]. Available: https://csrc.nist.gov/Presentations/2021/suitability-of-3rd-roundsignature-candidates-for.

[32] L. Beckwith, D. T. Nguyen, and K. Gaj, "Hardware accelerators for digital signature algorithms dilithium and FALCON," *IEEE Des. Test* (2023). doi: 10.1109/MDAT.2023.3305156.

[33] P. Karl, J. Schupp, T. Fritzmann, and G. Sigl, "Post-quantum signatures on RISC-V with hardware acceleration," *ACM Trans. Embed. Comput. Syst.*, vol. 23, no. 2, pp. 1–23 (2024). doi: 10.1145/3579092.

[34] Y. Lee *et al.*, "An efficient hardware/software co-design for FALCON on low-end embedded systems," *IEEE Access* (2024). doi: 10.1109/ACCESS.2024.3387489.

[35] [35] H. Jung and H. Oh, "Designing a scalable and area-efficient hardware accelerator supporting multiple PQC schemes," *Electronics*, vol. 13, no. 17, p. 3360 (2024). doi: 10.20944/preprints202407.0039.v1.

[36] L. Zhou, A. Diro, A. Saini, S. Kaisar, and P. C. Hiep, "Leveraging zero knowledge proofs for blockchain-based identity sharing: A survey of advancements, challenges and opportunities," *J. Inf. Secur. Appl.*, vol. 80, p. 103678 (2024). doi: 10.1016/j.jisa.2023.103678.

[37] R. Lavin, X. Liu, H. Mohanty, L. Norman, G. Zaarour, and B. Krishnamachari, "A survey on the applications of zero-knowledge proofs," (2024). [Online]. Available: https://arxiv.org/abs/2408.00243.

[38] Q. Kniep and R. Wattenhofer, "Tyche: Collateral-free coalition-resistant multiparty lotteries with arbitrary payouts," (2024). [Online]. Available: https://arxiv.org/abs/2409.03464.

[39] L. Horstmeyer, "Lakat: An open and permissionless architecture for continuous integration academic publishing," (2023). [Online]. Available: https://arxiv.org/abs/2306.09298.

[40] E. Jhessim and T. Santigie-Sankoh, "Quantum computing in financial security: A risk management framework for systemically important financial institutions," *World J. Adv. Res. Rev.*, pp. 1963–1967 (2025). doi: 10.30574/wjarr.2025.25.1.0235.

[41] K. Jaggi, "Advancing cybersecurity strategies: Balancing threat detection, compliance, and resilient architectures," (2025). doi: 10.2139/ssrn.5124287.

[42] K. K. Singamaneni *et al.*, "A novel hybrid quantum-crypto standard to enhance security and resilience in 6G enabled IoT networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, pp. 1–19 (2025). doi: 10.1109/JSTARS.2025.3540905.

[43] N. Ul Ain *et al.*, "A novel approach based on quantum key distribution using BB84 and E91 protocol for resilient encryption and eavesdropper detection," *IEEE Access*, vol. 13, pp. 32819–32833 (2025). doi: 10.1109/ACCESS.2025.3539178.

[44] A. K. Singh, N. Sharma, V. P. Singh, and A. Prabhakar, "Backflash attack on coherent one-way quantum key distribution," *IEEE Photonics J.* (2025). doi: 10.48550/arXiv.2502.04081.

[45] I. O. T. Musaddiq, Arslan; Azam, "Machine learning for resource management in industrial Internet of Things," *Front. Comput. Sci.*, p. 1566353 (2025). doi: 10.3389/fdata.2024.1422546.

[46] P. Nahta, "Securing the digital supply chain: Challenges, innovations, and best practices in cybersecurity," *Innovation Manag. Resil. Digital Econ.* IGI Global, pp. 205–230 (2025). doi: 10.4018/979-8-3693-8357-5.ch008.

[47] E. H. Nurkifli, "A biometric and PUF-based authentication with preserving anonymity in smart grid environment," *Ain Shams Eng. J.*, vol. 15, no. 12, p. 103177 (2024). doi: 10.1016/j.asej.2024.103177.

[48] C. Gewehr, L. Luza, and F. G. Moraes, "Hardware acceleration of crystals-Kyber in low-complexity embedded systems with RISC-V instruction set extensions," *IEEE Access*, vol. 12, pp. 94477–94495 (2024). doi: 10.1109/ACCESS.2024.3416812.

[49] S. A. Islam, M. MohanKumar, and U. K. Jannat, "Enhancing data security in mobile traffic networks through reverse engineering," *in 2024 4th Int. Conf. Ubiquitous Comput. Intell. Inf. Syst. (ICUIS)*, pp. 1408–1416 (2024). doi: 10.1109/ICUIS64676.2024.10866267.

[50] D. A. D. C. Alagic, G. and Q. Dang, "Status report on the third round of the NIST post-quantum cryptography standardization process," *in NIST IR 8413*, pp. 1–90 (2022). doi: 10.6028/NIST.IR.8413.

[51] D. Sikeridis, P. Kampanakis, and M. Devetsikiotis, "Post-quantum authentication in TLS 1.3: A performance study," *Cryptology ePrint Arch.*, Paper 2020/071 (2020). doi: 10.14722/ndss.2020.24203.