Quantum-Safe Authentication Protocol for IoT-Enabled Transportation Systems

Rohini Poolat Parameswarath, Chao Wang, and Biplab Sikdar, *Fellow, IEEE*

***Abstract*— The integration of Internet of Things (IoT) technology into transportation systems to enable real-time data collection and analysis helps to achieve increased efficiency and enhanced user experience. However, the IoT network faces significant cybersecurity challenges, especially with the advent of quantum computing, which poses a serious threat to classical security schemes. This paper proposes an authentication protocol for IoT-based transportation systems using CRYSTALS-Kyber, a lattice-based Post-Quantum Key Encapsulation Mechanism (PQKEM), and Quantum Random Number Generators (QRNGs) together with lightweight cryptographic operations. In contrast to the existing literature, the proposed protocol is lightweight and provides protection against conventional and quantum attacks while adhering to standards and ensuring interoperability with future systems. We provide a formal security proof of the proposed protocol using the Real-Or-Random (RoR) model and analyze its security under the Canetti–Krawczyk (CK) adversary model. The performance analysis shows that the protocol offers strong security and scalability with minimal computational overhead compared to existing schemes. Additionally, we assess the performance of the proposed protocol through simulations conducted with the NS3 simulator.**

***Index Terms*—Internet of Things (IoT), key encapsulation mechanism (KEM), post-quantum cryptography (PQC), quantum random number generator (QRNG)**

I. Introduction

The Internet of Things (IoT) refers to smart things/devices with sensory and communication capabilities that collect and transmit data over the Internet to servers for decision-making [1]. The IoT helps to develop a smart environment with real-time data exchange and smart decision-making to improve areas such as transportation, healthcare, agriculture, and more [1].

The adoption of IoT brings significant advantages to the transportation sector. Vehicle monitoring and diagnostics, route optimization, energy management, autonomous driving, and smart charging infrastructure are some of the applications enabled by IoT technology in transportation [2], [3]. By optimizing routes, and improving fuel efficiency, IoT can significantly lower the carbon footprint of transportation systems. Thus, the integration of IoT into transportation systems has the potential to revolutionize mobility by increasing efficiency, improving customer experience, and ensuring sustainability [2], [3].

R.P Parameswarath, C. Wang, and B. Sikdar are with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. (Email: rohini.p@nus.edu.sg, wang.chao@nus.edu.sg, bsikdar@nus.edu.sg)

IoT devices transmit data to the server via the Internet which is an insecure medium. Hence, adversaries can launch various attacks in the IoT network [4], [5]. Adversaries may tamper with the transmitted data which can result in undesired consequences [6]. Since attacks against IoT communication can have an impact on the whole IoT network, attacks against IoT networks and solutions to protect IoT communications are well-investigated topics in the literature. To protect IoT communications from attacks, several existing authentication schemes leverage public-key cryptography. Public-key cryptography derives its security from the hardness of solving mathematical problems such as the discrete logarithm problem and the integer factorization problem which cannot be solved efficiently by classical computers. However, it has been demonstrated that with quantum computers and certain quantum algorithms such as Shor's algorithm [7], these hard problems can be solved efficiently. Hence, with the introduction of quantum computers, it is essential to develop quantum-safe authentication protocols whose security cannot be weakened by quantum-computing-enabled adversaries. The National Institute of Standards and Technology (NIST) recognized the need for Post-Quantum Cryptography (PQC) and initiated a PQC standardization process. As part of this standardization process, CRYSTALS-Kyber [8] has been selected as a quantum-safe Key Encapsulation Mechanism (KEM). CRYSTALS-Kyber obtains its security from the computational difficulty of solving the learning-with-errors (LWE) problem in module lattices [9].

Random numbers are employed in authentication protocols to ensure unpredictability and uniqueness in each session. Pseudo-random number generators (PRNGs) are commonly employed for generating random numbers. However, random numbers generated by PRNGs are predictable if the algorithm and seeds used by PRNGs are known [10]. On the other hand, Quantum Random Number Generators (QRNGs) use properties of quantum physical processes to generate random numbers. As a result, random numbers generated by QRNGs are truly random and secure. By combining CRYSTALS-Kyber and truly random numbers from QRNG, we propose a quantum-safe authentication protocol for IoT-enabled transportation systems that provides security against classical and quantum computer-enabled attacks.

A. Related Work

Xenofontos et al. [6] explored the vulnerabilities in IoT and proposed a taxonomy categorizing the attacks. A lightweight user authentication protocol for generic IoT deployment was proposed in [11]. This scheme employed a symmetric cryptographic algorithm and a fuzzy extractor for biometric verification. Srinivas et al. [12] pointed out that the scheme in [11] is not efficient as the list of devices must be updated during each iteration of the authentication phase. Further, Srinivas et al. [12] developed a mutual authentication protocol for IoT technology-enabled intelligent transportation systems. This protocol is based on Elliptic-Curve Cryptography (ECC). Zhang et al. developed an authentication protocol for IoT-enabled maritime transportation systems in [13]. This scheme was based on blockchain. An authentication protocol for 6G-IoT aided maritime transport systems was proposed in [14]. The scheme in [14] is vulnerable to desynchronization attacks. An authentication protocol tailored for IoT environments that employed hash functions, ECC, and XOR operations was proposed in [15]. The main advantage of this scheme is that it is lightweight. Yang et al. [16] developed an authentication scheme for facilitating information exchange in wireless sensor networks used in IoT applications. Their scheme can resist node-captured attacks. A mutual authentication scheme designed for IoT environments that use cloud technology was introduced in [17]. The scheme in [18] leverages the decentralized and tamper-resistant nature of the blockchain and unique hardware-based identifiers provided by Physical Unclonable Functions (PUFs) to develop an authentication scheme that does not require heavy computational resources for intelligent transportation systems. Another authentication scheme for IoT devices was proposed in [4]. This scheme also used PUFs and fuzzy extractors as the building blocks. Though the schemes in [18] and [4] provide several security features, they need additional hardware, PUFs. Additionally, these schemes require substantial memory space to store the challenge-response pairs of the PUFs. PUF-based solutions are also vulnerable to modelling attacks where an adversary trains a machine learning model with challenge-response pairs to predict the response for a new challenge. Further, the scheme in [4] does not offer resilience against desynchronization and Denial-of-Service (DoS) attacks. Bagga et al. presented a mutual authentication protocol for intelligent transportation systems in [19]. This protocol leveraged ECC and hash functions and is demonstrated to be secure within the Canetti-Krawczyk (CK) adversary model. A mutual authentication protocol for vehicle-to-vehicle communication was presented in [20]. However, this scheme does not meet the session key security under the CK adversary model [19]. Challa et al. [21] proposed a key establishment scheme based on ECC for the IoT environment. Though Challa et al.'s protocol provides several security features, Srinivas et al [12] noted that the scheme in [21] has high computation and communication costs, limiting its applicability in resource-constrained IoT environments. Though the protocols in [11]–[21] addressed some of the security issues in IoT-based transportation systems and IoT environments, none of them are quantum-safe.

There are quantum-resistant authentication schemes in other domains, such as [22]–[24]. Gupta et al. proposed a lattice-based quantum-secure authentication scheme tailored for e-health systems in IoT environments in [22]. The scheme in [22] is vulnerable to desynchronization and impersonation attacks. A lattice-based quantum-secure authentication scheme for the Internet of Drones was proposed in [23]. The scheme in [23] does not maintain strong anonymity and unlinkability. Chaudhary et al. proposed a three-party key agreement scheme in [24]. The scheme in [24] is quantum-secure. However, it is vulnerable to replay attacks and does not offer perfect forward secrecy. Also, it has a very high computation cost, limiting its scalability [25]. Further, the schemes in [22], [23], and [24] do not align with NIST's PQC standards.

B. Comparison with Our Previous Paper

This paper is an extension of our previous paper [26]. In [26], we proposed a quantum-safe authentication protocol using CRYSTALS-Kyber for IoT applications in transportation systems. In this paper, we employ QRNG together with PQKEM to build the authentication protocol. QRNG ensures that the random numbers used are truly unpredictable and secure. We also consider the widely accepted CK adversary model [27] for the adversary model in addition to the Dolev-Yao model [28] considered in [26]. Compared to [26], we have expanded the security proof and demonstrated that the proposed protocol ensures security under the CK model. Further, we evaluate the performance of the proposed protocol using the NS3 simulator. More extensive performance comparisons with similar schemes than given in [26] are also presented in Section VI.

C. Motivation and Contributions

Leveraging IoT applications in transportation systems has the potential to drive sustainability and can revolutionize services and user experience. To realize the potential of IoT-enabled transportation systems, it is important to secure the communication in the IoT network. The existing traditional cryptographic techniques-based authentication protocols for the IoT network do not offer quantum security. Though there are some quantum-safe authentication schemes in other domains in the literature, they have limitations such as being vulnerable to certain attacks or having high computation costs and limited scalability as discussed in Section I-A. Further, those schemes do not align with NIST's PQC standards. Motivated by these limitations in the existing literature, this paper makes the following contributions:

- **Quantum-secure authentication protocol for IoT-enabled transportation systems:** This is the first authentication protocol that employs CRYSTALS-Kyber

and QRNG as building blocks together with lightweight cryptographic operations. Kyber has medium-sized keys and offers the best overall performance compared to other KEM schemes [29], [30]. The proposed protocol is lightweight since Kyber and lightweight cryptographic operations are employed. Further, unlike PRNGs, QRNGs generate random numbers that cannot be predicted, ensuring that the random numbers used are truly random and secure. The proposed protocol leverages QRNG-generated random numbers and Kyber's key encapsulation on these random numbers to derive unique session keys for each session that are quantum-secure. Thus, the proposed protocol's design ensures that it is lightweight and offers protection against quantum attacks.

- **Comprehensive security features:** In addition to post-quantum security, the proposed protocol offers comprehensive security features including session key security, mutual authentication, strong anonymity, message integrity, perfect forward secrecy, unlinkability, known key secrecy as well as resistance to attacks such as eavesdropping and Man-In-The-Middle (MITM), key replicating, desynchronization and DoS, impersonation, replay, and Ephemeral Secret Leakage (ESL). Thus, in contrast to the existing literature, the proposed protocol offers superior security features by protecting against conventional and quantum attacks while adhering to standards and ensuring interoperability with future systems.
- **Security analysis:** We provide a formal security proof using the Real-Or-Random (RoR) model [31] and demonstrate that the proposed protocol offers robust security under the CK adversary model which highlights the effectiveness of the protocol. We also provide a detailed informal security analysis of the proposed protocol.
- **Performance analysis:** We compare the security features and computation cost of the proposed protocol with that of similar schemes to demonstrate its efficiency. We also conduct a scalability analysis of the proposed protocol to assess its practicality.
- **NS3 Simulation:** We also evaluate the performance of the proposed protocol using the NS3 simulator.

D. Organization

The rest of this paper is organized as follows: In Section II, we introduce the building blocks of the proposed protocol. Section III presents the system and adversary models. The proposed authentication protocol is presented in Section IV. In Section V, we provide a formal security analysis and an informal security analysis of the proposed protocol. A performance analysis is presented in Section VI. A discussion on the QRNG system is provided in Section VII. Section VIII presents the results of the performance evaluation using the NS3 simulator. Finally, Section IX concludes the paper.

II. Preliminaries

In this section, we discuss the background required for the proposed authentication protocol.

A. Lattice-Based Cryptography

A lattice L is the collection of all linear combinations of basis vectors $b_1, b_2, \cdots, b_n \in R^m$. L can be written as $L = \{\Sigma a_i b_i \mid a_i \in Z\}$. Cryptographic constructions built on lattices have significant potential for post-quantum cryptography to replace the conventional schemes based on the hardness of integer factorization and discrete logarithms [8]. CRYSTALS–Kyber is a lattice-based key-encapsulation mechanism (KEM). Its security is based on the hardness of solving the learning-with-error (LWE) problem over module lattices [8] which is difficult to solve even for quantum computers [8]. LWE problem is explained next.

B. Learning With Error Problem

The LWE problem was presented in its current form by Oded Regev [32]. The LWE problem involves solving a system of noisy linear equations. Let $\mathbb{Z}_q$, $\mathbb{Z}_q^n$ denote the finite ring of integers modulo q and the vector space of dimension n over $\mathbb{Z}_q$, respectively. Let χ be a probability distribution over $\mathbb{Z}_q$. The parameters a_i and s are chosen uniformly at random from $\mathbb{Z}_q^n$ and e_i is chosen according to the finite probability distribution χ. LWE problem involves finding the secret vector $s \in \mathbb{Z}_q^n$ given $\{(a_i, < a_i, s > +e_i)\}_{i=1}^m$. No probabilistic polynomial time algorithm can recover s with non-negligible probability [32].

C. Key Encapsulation Mechanism

The key encapsulation mechanism enables secure key exchange between two parties. The KEM involves three functions [8]:

Key Generation: We denote the key generation function as $Gen()$. It generates a public key (p_k) and a private key (s_k). This function can be written as $(p_k, s_k) \leftarrow Gen()$.

Encapsulation: The encapsulation algorithm $Enc()$ encapsulates a secret key k in a ciphertext c using p_k. This function can be written as $(c, k) \leftarrow Enc(p_k)$.

Decapsulation: We denote decapsulation as $Dec()$. $Dec()$ takes s_k and c as inputs and outputs the secret key k. Decapsulation can be written as $k \leftarrow Dec(s_k, c)$.

D. Quantum Random Number Generator

In critical applications like cryptography, random numbers are required to be both uniformly distributed and unpredictable. This requirement cannot be met by using conventional solutions such as PRNGs, as they are based on intrinsically deterministic algorithms. QRNGs offer several advantages in providing true random numbers. First, QRNGs harness the intrinsic randomness of quantum mechanical systems. One of the best-known examples is Born's rule:

when a system is prepared in a superposition of measurement basis states, the measurement outcomes are inherently random [33], [34]. Second, true random number generation can be certified by quantum mechanics in a (semi-) device-independent manner, with a minimal number of assumptions about the devices used [35]–[37]. This is important to reduce the impact of practical device imperfections, potential attacks on trusted components, and vulnerabilities caused by device degradation. As a result, quantum technologies provide new methods for certified randomness generation, offering new opportunities for critical applications that require random numbers to be uniform and unpredictable.

The QRNG source can be verified to prevent bias or predictable outputs. QRNG verification process involves several steps. It starts with physical modeling of the quantum process. The next step is device calibration to avoid systematic bias. After that, the generated random numbers are subjected to statistical randomness tests to detect any non-random patterns. Quantum source parameters (e.g., detector rates, noise levels) are also monitored in real-time to detect malfunctions. QRNGs also use cryptographic randomness extractors to convert weak randomness into uniform random bits. For high-assurance scenarios, device-independent QRNGs approaches provide the strongest verification. Device-independent QRNGs use Bell tests to certify randomness without trusting the internal components [35].

QRNGs can be implemented on various platforms, including optical systems and level systems such as atoms. In addition, there are a growing number of on-chip QRNG demonstrations with various features and functionalities that offer promising cost-effective solutions for practical applications. For more details on the working principles, experimental demonstrations, and comparison of commercial QRNG products, we refer the readers to [33]–[35], [38]–[43].

III. System and Adversary Models

A. System Model

The system model is illustrated in Figure 1. We consider a smart transportation application where the vehicles are installed with IoT devices $\{\mathcal{D}_1, \mathcal{D}_2, ...\mathcal{D}_n\}$ which collect data and send the collected data periodically to the cloud server ($\mathcal{CS}$) through the Internet. We also consider a trusted authority ($\mathcal{T}$). The IoT devices and the $\mathcal{CS}$ must first register with the $\mathcal{T}$ before they can begin communicating with each other.

B. Adversary Model

The IoT devices send data to the cloud server through an insecure medium, the Internet. Hence, an adversary can carry out various attacks on the communication channels between them. We consider Dolev-Yao (DY) adversary model [28], CK adversary model [27], and Harvest Now, Decrypt Later (HNDL) attack model [44]. In the DY adversary model, an adversary can eavesdrop on, intercept, or edit the transmitted messages. In the CK adversary model, the adversary

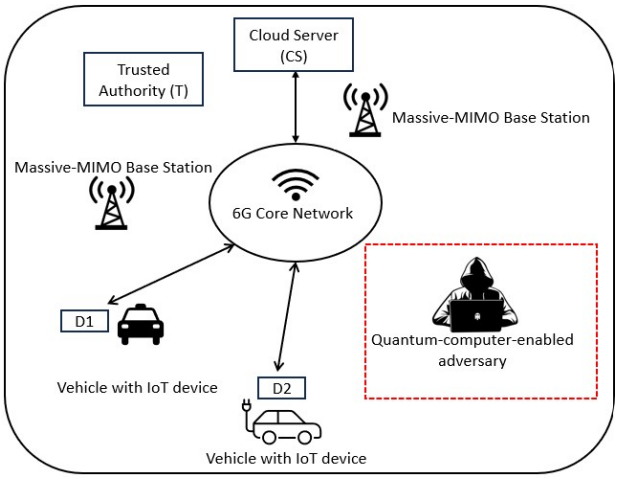


Fig. 1. System model.

can capture long-term secrets (keys used across multiple sessions) and short-term secrets [27], in addition to the DY model capabilities. Impersonation, modification, replay, Man-in-the-Middle, and ESL attacks resistance and perfect forward secrecy come under the CK adversary model [45]. An adversary with quantum-computing capabilities can break the security of asymmetric cryptographic techniques-based systems by using quantum algorithms. Hence, we also consider the HNDL attack model, where the adversary captures and stores messages that leverage asymmetric cryptographic techniques. When the adversary has quantum-computing capabilities, he/she can decrypt those stored messages using quantum algorithms. This will lead to breaking the security of communication based on those classical systems.

IV. Proposed Authentication Protocol

The proposed protocol consists of setup, registration, and authentication and key agreement phases. Figure 2 illustrates the protocol's high-level view.

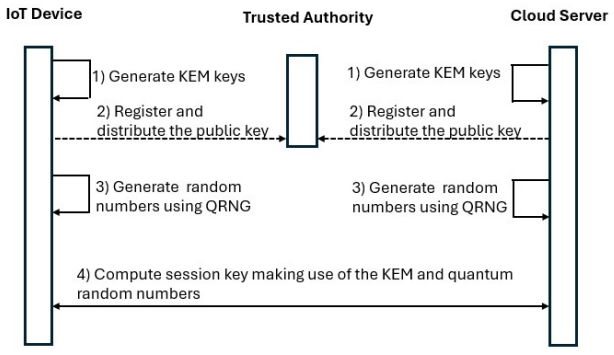


Fig. 2. High-level view of the protocol.

A. Assumptions

- Electromagnetic interference (EMI) occurs when an electrical device emits electromagnetic waves that in-

terfere with the normal functioning of nearby devices [46]. As more and more electronic systems, such as Electronic Control Units (ECUs), sensors, and communication modules, are integrated into modern vehicles, there have been concerns about EMI [46]. The EMI from these electronic systems can have an impact on IoT setups [47]. However, as the impact of EMI on IoT devices in vehicles is beyond the scope of this paper, we assume that vehicles meet Electromagnetic Compatibility (EMC) standards [46] and necessary measures have been taken to mitigate the impact of EMI on the IoT devices installed in them.

- As IoT devices are installed in vehicles, we assume that necessary measures have been taken to ensure the physical security of the vehicles, which will prevent the physical capture of the IoT devices.

B. Setup Phase

In this phase, the participants generate the key pair for Kyber [8] that will be used for encapsulation and decapsulation during the authentication phase.

Step 1: Let R_q denote the ring $\mathbb{Z}_q[X]/(X_n+1)$. The $\mathcal{CS}$ chooses a seed ρ and σ as $\rho, \sigma \leftarrow \{0,1\}^{256}$. Then, the $\mathcal{CS}$ calculates $\mathbf{A} \sim R_q^{k \times k} := Sam(\rho)$, $(s,e) \sim \beta_\eta^k \times \beta_\eta^k := Sam(\sigma)$, $t := Compress_q(\mathbf{A}s + e, d_t)$, where k and d_t are positive integers, Sam is an extendable output function, and β_η^k is the distribution. Finally, the $\mathcal{CS}$ computes the public key ($pk_{\mathcal{CS}}$) and the private key ($sk_{\mathcal{CS}}$) as $pk_{\mathcal{CS}} := (t, \rho)$ and $sk_{\mathcal{CS}} := (pk_{\mathcal{CS}}, z, s)$, where z is a random value.

Step 2: The IoT devices $\{\mathcal{D}_1, \mathcal{D}_2, ...\mathcal{D}_n\}$ also generate private and public keys as mentioned in Step 1. We denote the public key of $\mathcal{D}_i$ as $pk_{\mathcal{D}i}$ and the private key of $\mathcal{D}_i$ as $sk_{\mathcal{D}i}$.

C. Registration Phase

In this phase, the $\mathcal{CS}$ and the IoT devices $\{\mathcal{D}_1, \mathcal{D}_2, ...\mathcal{D}_n\}$ register with the $\mathcal{T}$.

Cloud Server Registration Phase: The steps involved in this phase are given below:

Step 1: Let $ID_{\mathcal{CS}}$ represent the identity of the $\mathcal{CS}$. The $\mathcal{CS}$ composes a message with a registration request and $ID_{\mathcal{CS}}$ as $MR_1 = \{Reg_{Req}, ID_{\mathcal{CS}}, pk_{\mathcal{CS}}\}$. Then, the $\mathcal{CS}$ sends MR_1 to the $\mathcal{T}$ through a secure channel.

Step 2: The $\mathcal{T}$ registers the $\mathcal{CS}$ and stores $ID_{\mathcal{CS}}$ and $pk_{\mathcal{CS}}$.

IoT Device Registration Phase: The steps involved in this phase are detailed below:

Step 1: The IoT device $\mathcal{D}_i$ with an identity $ID_{\mathcal{D}i}$ composes a message with a registration request as $MR_2 = \{Reg_{Req}, ID_{\mathcal{D}i}, pk_{\mathcal{D}i}\}$. Then, $\mathcal{D}_i$ sends MR_2 to the $\mathcal{T}$.

Step 2: Upon receiving MR_2, the $\mathcal{T}$ registers $\mathcal{D}_i$. Let $T_{\mathcal{D}_i}$ be the time of registration of $\mathcal{D}_i$. The $\mathcal{T}$ computes a temporary identity $TID_{\mathcal{D}i} = h(ID_{\mathcal{D}i} \parallel T_{\mathcal{D}_i})$ for $\mathcal{D}_i$. The $\mathcal{T}$ also generates a set of temporary identities $S_{\mathcal{D}_i}$ for $\mathcal{D}_i$. Then, the $\mathcal{T}$ composes $MR_2' = \{Ack, TID_{\mathcal{D}i}, S_{\mathcal{D}_i}, ID_{\mathcal{CS}}, pk_{\mathcal{CS}}\}$ where Ack is the acknowledgement. Finally, the $\mathcal{T}$ sends MR_2' to $\mathcal{D}_i$ through a secure channel.

Step 3: After receiving MR_2' from the $\mathcal{T}$, $\mathcal{D}_i$ stores $TID_{\mathcal{D}i}$, $S_{\mathcal{D}_i}$, $ID_{\mathcal{CS}}$, and $pk_{\mathcal{CS}}$.

Step 4: The $\mathcal{T}$ also sends $ID_{\mathcal{D}i}$, $TID_{\mathcal{D}i}$, $S_{\mathcal{D}_i}$, and $pk_{\mathcal{D}i}$ to the $\mathcal{CS}$. The $\mathcal{CS}$ stores $ID_{\mathcal{D}i}$, $TID_{\mathcal{D}i}$, $S_{\mathcal{D}_i}$, and $pk_{\mathcal{D}i}$.

D. Authentication and Key Agreement Phase

The authentication and key agreement phase is executed whenever $\mathcal{D}_i$ wants to send data to the $\mathcal{CS}$. Before data transfer, $\mathcal{D}_i$ and the $\mathcal{CS}$ authenticate each other and establish a session key. The steps involved in this phase are listed below:

Step 1: $\mathcal{D}_i$ generates a random number r_1 using the QRNG. $\mathcal{D}_i$ also generates a key k_1. Then, $\mathcal{D}_i$ encapsulates k_1 in a ciphertext c_1 using $p_{\mathcal{CS}}$ as mentioned in Section II-C and computes a validation parameter $V_1 = h(r_1 \parallel k_1 \parallel TID_{\mathcal{D}i})$. Then, $\mathcal{D}_i$ composes a message $MA_1 = \{TID_{\mathcal{D}i}, c_1, r_1, V_1\}$ and sends it to the $\mathcal{CS}$.

Step 2: Upon receiving MA_1, the $\mathcal{CS}$ verifies that the temporary identity received corresponds to the current temporary identity of $\mathcal{D}_i$ stored in its memory. If they do not match, the protocol gets terminated. Otherwise, the $\mathcal{CS}$ decapsulates k_1 from c_1 using $s_{\mathcal{CS}}$ as mentioned in Section II-C. Next, $\mathcal{CS}$ computes $V_1' = h(r_1 \parallel k_1 \parallel TID_{\mathcal{D}i})$. Then, the $\mathcal{CS}$ verifies V_1 against V_1'. Successful verification of the validation parameter indicates that MA_1 has not been tampered with by an adversary during transmission. After that, the $\mathcal{CS}$ generates a random number r_2 using the QRNG. The $\mathcal{CS}$ also generates a key k_2. Then, the $\mathcal{CS}$ encapsulates k_2 in a ciphertext c_2 using $p_{\mathcal{D}i}$ as mentioned in Section II-C. Then, the $\mathcal{CS}$ computes a validation parameter $V_2 = h(r_2 \parallel k_2 \parallel TID_{\mathcal{D}i})$ and the session key $sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$. The temporary identity of $\mathcal{D}_i$ to use in the next round of authentication is computed as $TID_{\mathcal{D}_i}^* = TID_{\mathcal{D}_i} \oplus k_1 \oplus k_2$. Finally, the $\mathcal{CS}$ composes a message MA_2 with an acknowledgment Ack, c_2, r_2, and V_2 as $MA_2 = \{Ack, c_2, r_2, V_2\}$ and sends it to $\mathcal{D}_i$.

Step 3: $\mathcal{D}_i$ decapsulates k_2 from c_2 using $s_{\mathcal{D}i}$ as mentioned in Section II-C. Next, $\mathcal{D}_i$ computes $V_2' = h(r_2 \parallel k_2 \parallel TID_{\mathcal{D}i})$. Then, $\mathcal{D}_i$ verifies V_2 against V_2'. Successful verification indicates that M_2 has not been tampered with by an adversary. Then, $\mathcal{D}_i$ computes the temporary identity to use in the next round of authentication as $TID_{\mathcal{D}_i}^* = TID_{\mathcal{D}_i} \oplus k_1 \oplus k_2$ and the session key $sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$.

The steps in the authentication and key agreement phase are shown in Figure 3.

Remark: An adversary may try to drop the messages between the protocol participants to desynchronize certain parameters, which will result in a DoS attack [45]. In the proposed protocol, if the adversary drops MA_2, the temporary identity of $\mathcal{D}_i$ stored in the $\mathcal{CS}$ for the next authentication session is $TID_{\mathcal{D}_i}^*$ whereas it is $TID_{\mathcal{D}_i}$ in $\mathcal{D}_i$. In such a scenario, when the $\mathcal{CS}$ receives message MA_1

in the next authentication session, the temporary identity received will be different from the temporary identity of $\mathcal{D}_i$ stored in $\mathcal{CS}$'s memory. Since they do not match, the protocol gets terminated. Thus, a one-sided update of the parameter results in desynchronization. If the protocol gets terminated due to such a desynchronization event, $\mathcal{D}_i$ can initiate a protocol session with one of the temporary identities from the set of temporary identities $S_{\mathcal{D}_i}$. Thus, both parties can resynchronize.

$\mathcal{D}_i$	$\mathcal{CS}$
Generate: r_1	
Generate: k_1	
$(c_1, k_1) \leftarrow Enc(p_{\mathcal{CS}})$	
$V_1 = h(r_1 \parallel k_1 \parallel TID_{\mathcal{D}i})$	
$MA_1 = \{TID_{\mathcal{D}i}, c_1, r_1, V_1\}$	
$\xrightarrow{MA_1}$	
	Verify: $TID_{\mathcal{D}i}$
	$k_1 \leftarrow Dec(c_1, s_{\mathcal{CS}})$
	$V_1' = h(r_1 \parallel k_1 \parallel TID_{\mathcal{D}i})$
	Verify: $V_1 = ?V_1'$
	Generate: r_2
	Generate: k_2
	$(c_2, k_2) \leftarrow Enc(p_{\mathcal{D}i})$
	$V_2 = h(r_2 \parallel k_2 \parallel TID_{\mathcal{D}i})$
	$sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$
	$TID^*_{\mathcal{D}_i} = TID_{\mathcal{D}_i} \oplus k_1 \oplus k_2$
	$MA_2 = \{Ack, c_2, r_2, V_2\}$
	$\xleftarrow{MA_2}$
$k_2 \leftarrow Dec(c_2, s_{\mathcal{D}i})$	
$V_2' = h(r_2 \parallel k_2 \parallel TID_{\mathcal{D}i})$	
Verify: $V_2 = ?V_2'$	
$TID^*_{\mathcal{D}_i} = TID_{\mathcal{D}_i} \oplus k_1 \oplus k_2$	
$sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$	

Fig. 3. Authentication and key agreement phase.

V. Security Analysis

In this section, first, we analyze the security of the proposed authentication protocol using the RoR model [31]. Then, we perform an informal security analysis of the proposed protocol.

A. Formal Security Analysis

Security Model: An adversary $\mathcal{A}$ calls Oracle queries and tries to get the established session key in a protocol session between $\mathcal{D}_i$ and the $\mathcal{CS}$. The Oracle queries that $\mathcal{A}$ uses are listed below:

- *Listen($\mathcal{D}_i$, $\mathcal{CS}$):* $\mathcal{A}$ listens to the messages transmitted between $\mathcal{D}_i$ and the $\mathcal{CS}$ using this query. This query models a passive attack.
- *Send(M):* $Send()$ query models an active attack. $\mathcal{A}$ calls *Send(M)* query to send a message M to $\mathcal{D}_i$ or to the $\mathcal{CS}$.
- *Hash(M):* $\mathcal{A}$ calls this query to get the hash of a message M.
- *Reveal(X):* $Reveal(X)$ query models the ephemeral secret leakage attack in the CK adversary model. $\mathcal{A}$ wants to capture the ephemeral secrets of $\mathcal{D}_i$ and the $\mathcal{CS}$ to generate the session key. $\mathcal{A}$ calls *Reveal(X)* query to capture the ephemeral secrets of X, where X represents $\mathcal{D}_i$ or the $\mathcal{CS}$.
- *Corrupt(X):* $\mathcal{A}$ wants to capture the long-term secret credentials of $\mathcal{D}_i$ and the $\mathcal{CS}$ to generate the session key. $\mathcal{A}$ calls the *Corrupt(X)* query to capture the long-term secret credentials of X, where X represents $\mathcal{D}_i$ or the $\mathcal{CS}$.
- *Test():* The $Test()$ query defines the session key's semantic security. $\mathcal{A}$ can call the $Test()$ query only once. When this query is executed, a bit b will be flipped. If $b = 1$, $\mathcal{A}$ receives the actual session key. Otherwise, $\mathcal{A}$ receives a random string instead of the session key.

Definition 1: Let Win represent the event in which $\mathcal{A}$ correctly guesses the bit b in the *Test()* query. The advantage of $\mathcal{A}$, Adv^A, in breaking the semantic security of the protocol is the probability of correctly guessing b. If it is negligible, the protocol is secure. Adv^A can be written as:

$$Adv^A = \mid 2 \cdot Pr[Win] - 1 \mid .$$

Definition 2: The advantage of $\mathcal{A}$ in compromising the security of CRYSTALS-Kyber is negligible, i.e., $Adv^{Kyber} \leq \epsilon$, where ϵ is negligible [8].

Theorem 1. Consider the scenario where adversary $\mathcal{A}$ sends n_h, n_s, and n_l $Hash$, $Send$, and $Listen$ queries, respectively. Let $|h|$ be the length of the hash output. Then, the advantage of $\mathcal{A}$ winning against the proposed protocol is $Adv^A \leq \frac{(n_h)^2}{2^{(h)}} + \frac{(n_s+n_e)^2}{l} + \frac{n_s}{2^{(h-1)}} + 2Adv^{Kyber}$, which is negligible.

Proof: Consider a series of games g_i for $i \in \{0, 1, 2, 3, 4\}$.

g_0 : This game models a real attack by the adversary against the proposed protocol. $\mathcal{A}$ guesses the bit b randomly in g_0. Hence, according to Definition 1, the advantage of $\mathcal{A}$ can be written as:

$$Adv^A = \mid 2Pr[Win_{g_0}] - 1 \mid . \tag{1}$$

g_1: All the queries are simulated in this game. Since the queries $Listen$, $Send$, $Hash$, $Reveal$, and $Corrupt$ are simulated as in a real attack, g_0 and g_1 are identical. Hence, we can write:

$$Pr[Win_{g_1}] = Pr[Win_{g_0}]. \tag{2}$$

g_2**:** If there is no collision in hash or transcripts, games g_1 and g_2 are indistinguishable. From the birthday paradox, the collision probability of the hash function is at most $\frac{(n_h)^2}{2^{(h+1)}}$ and the collision probability in transcripts is $\frac{(n_s+n_l)^2}{2l}$ where l is the length of the transcripts. Hence, we can write:

$$Pr[Win_{g_2}] - Pr[Win_{g_1}] \leq \frac{(n_h)^2}{2^{(h+1)}} + \frac{(n_s+n_l)^2}{2l}. \tag{3}$$

g_3**:** The difference between g_2 and g_3 is that in g_3, $\mathcal{A}$ guesses the verifier's value correctly without sending the $Hash()$ query and only by sending $Send()$ queries. If $\mathcal{A}$ can guess this value correctly, $\mathcal{A}$ wins and stops g_3. Hence, we can write:

$$Pr[Win_{g_3}] - Pr[Win_{g_2}] \leq \frac{n_s}{2^{(h)}}. \quad (4)$$

g_4**:** This game considers the session key leakage between $\mathcal{D}_i$ and the $\mathcal{CS}$. $\mathcal{A}$ calls $Corrupt()$ and $Reveal()$ queries in an attempt to capture sk. The session key is calculated as $sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$. The key k_1 is encapsulated as $(c_1, k_1) \leftarrow Enc(p_{\mathcal{CS}})$ and k_2 is encapsulated as $(c_2, k_2) \leftarrow Enc(p_{\mathcal{D}i})$. To decapsulate k_1 and k_2 as $k_1 \leftarrow Dec(c_1, s_{\mathcal{CS}})$ and $k_2 \leftarrow Dec(c_2, s_{\mathcal{D}i})$, $\mathcal{A}$ needs to break the security of Kyber. Hence, the difference between g_4 and g_3 is the advantage of $\mathcal{A}$ in compromising Kyber, i.e., Adv^{Kyber}. By Definition 2, Adv^{Kyber} is negligible. Hence, we can write:

$$Pr[Win_{g_4}] - Pr[Win_{g_3}] \leq Adv^{Kyber}. \quad (5)$$

Finally, $\mathcal{A}$ guesses the bit b and calls the $Test()$ query. Then, we can write:

$$Pr[Win_{g_4}] = \frac{1}{2}. \quad (6)$$

From (1) and (2), we can write the following:

$$\begin{aligned} \frac{1}{2}Adv^A &= \mid Pr[Win_{g_0}] - \frac{1}{2} \mid \\ &= \mid Pr[Win_{g_1}] - \frac{1}{2} \mid . \end{aligned} \quad (7)$$

Using Equations (3) to (7) and by applying the triangle inequality, we have:

$$\begin{aligned} \frac{1}{2}Adv^A &= \mid Pr[Win_{g_1}] - \frac{1}{2} \mid \\ &= \mid Pr[Win_{g_1}] - Pr[Win_{g_4}] \mid \\ &\leq \frac{(n_h)^2}{2^{(h+1)}} + \frac{(n_s+n_l)^2}{2l} + \frac{n_s}{2^{(h)}} + Adv^{Kyber}. \end{aligned} \quad (8)$$

By multiplying both sides of Equation (8) by 2, we have:

$$Adv^A \leq \frac{(n_h)^2}{2^{(h)}} + \frac{(n_s + n_l)^2}{l} + \frac{n_s}{2^{(h-1)}} + 2Adv^{Kyber}. \quad (9)$$

■

B. Informal Security Analysis

- **Resistance to Attacks by Quantum Computers:** Classical public key cryptographic techniques assume that their underlying mathematical problems are hard to solve by classical computers. This assumption will not be valid with the emergence of powerful quantum computers and new algorithms, e.g., Shor's algorithm. On the contrary, CRYSTALS-Kyber is designed to resist attacks by quantum computers as its security arises from the difficulty of solving the LWE problem over module lattices. The proposed protocol is built using CRYSTALS-Kyber. Hence, it can resist attacks by quantum computers and HNDL adversaries.
- **Resistance to Eavesdropping and MITM Attacks:** Two messages $MA_1 = \{TID_{\mathcal{D}i}, c_1, r_1, V_1\}$ and $MA_2 = \{Ack, c_2, r_2, V_2\}$ are exchanged between $\mathcal{D}_i$ and the $\mathcal{CS}$ in the proposed protocol. The session key is calculated as $sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$. The key k_1 is encapsulated as $(c_1, k_1) \leftarrow Enc(p_{\mathcal{CS}})$ and k_2 is encapsulated as $(c_2, k_2) \leftarrow Enc(p_{\mathcal{D}i})$. The adversary cannot decapsulate k_1 and k_2 as $k_1 \leftarrow Dec(c_1, s_{\mathcal{CS}})$ and $k_2 \leftarrow Dec(c_2, s_{\mathcal{D}i})$ since he/she does not have the knowledge of $s_{\mathcal{CS}}$ and $s_{\mathcal{D}i}$. As a result, the adversary cannot eavesdrop on and modify MA_1 and MA_2.
- **Resistance to Key Replicating Attacks:** In the key replicating attack or the key offset attack, the adversary intercepts the messages and edits them so that the participants establish different session keys [45]. In the proposed protocol, the messages exchanged between the participants are $MA_1 = \{TID_{\mathcal{D}i}, c_1, r_1, V_1\}$ and $MA_2 = \{Ack, c_2, r_2, V_2\}$. The session key is calculated as $sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$. When $\mathcal{D}_i$ calculates sk, $\mathcal{D}_i$ knows r_1 and k_1 while r_2 is obtained from MA_2 and k_2 is computed from c_2. If the adversary modifies c_2 and r_2 in MA_2 in an attempt to make $\mathcal{D}_i$ compute the wrong session key, when $\mathcal{D}_i$ computes $V_2' = h(r_2 \parallel k_2 \parallel TID_{\mathcal{D}i})$, it will be different from the received V_2. Similarly, when $\mathcal{CS}$ calculates sk, $\mathcal{CS}$ knows r_2 and k_2 while r_1 is obtained from MA_1 and k_1 is computed from c_1. If the adversary modifies c_1 and r_1 in MA_1 in an attempt to make $\mathcal{CS}$ compute the wrong session key, when $\mathcal{CS}$ computes $V_1' = h(r_1 \parallel k_1 \parallel TID_{\mathcal{D}i})$, it will be different from the received V_1. Hence, the adversary's attempt to edit the messages with an intention of $\mathcal{CS}$ and $\mathcal{D}_i$ establishing different session keys will be detected. Thus, the protocol is resilient to key replicating attacks.
- **Resistance to Impersonation Attacks:** $\mathcal{D}_i$ composes MA_1 as $MA_1 = \{TID_{\mathcal{D}i}, c_1, r_1, V_1\}$. In MA_1, k_1 is encapsulated using $p_{\mathcal{CS}}$ to get the ciphertext c_1. Only the $\mathcal{CS}$ knows $s_{\mathcal{CS}}$, to decapsulate k_1 from c_1. Since the adversary does not know $s_{\mathcal{CS}}$, the adversary cannot impersonate $\mathcal{CS}$ to continue with the rest of the protocol execution. Similarly, the $\mathcal{CS}$ encapsulates k_2 as $(c_2, k_2) \leftarrow Enc(p_{\mathcal{D}i})$ and composes $MA_2 = \{Ack, c_2, r_2, V_2\}$. Only a registered IoT device $\mathcal{D}_i$ knows $s_{\mathcal{D}i}$, to decapsulate k_2 from c_2. Since an adversary does not know $s_{\mathcal{D}i}$, the adversary cannot impersonate $\mathcal{D}_i$ to continue with the rest of the protocol execution. Thus, the proposed protocol prevents impersonation of the participants $\mathcal{D}_i$ and the $\mathcal{CS}$.
- **Strong Anonymity and Unlinkability:** The proposed protocol ensures both anonymity and unlinkability. A protocol is anonymous if pseudonyms are used instead of real identities. To provide strong anonymity, the parameters used should be changed in each protocol

session as well [45]. Instead of using the real identity, $\mathcal{D}_i$ sends $TID_{\mathcal{D}i}$ in MA_1, ensuring anonymity in the proposed protocol. Further, the temporary identity for the next session is computed as $TID^*_{\mathcal{D}_i} = TID_{\mathcal{D}_i} \oplus k_1 \oplus k_2$. Thus, the temporary identity is changed in each session. As a result, the adversary will not be able to link two sessions of the protocol, resulting in the sessions being unlinkable. As a result, the protocol provides strong anonymity as well.

- **Resilience Against Desynchronization and DoS Attacks:** An adversary may try to drop the messages between the protocol participants to desynchronize certain parameters, resulting in a DoS attack [45]. In the proposed protocol, if the adversary drops MA_2, the temporary identity of $\mathcal{D}_i$ stored in $\mathcal{CS}$ for the next session is $TID^*_{\mathcal{D}_i}$ whereas it is $TID_{\mathcal{D}_i}$ in $\mathcal{D}_i$. When the $\mathcal{CS}$ receives the message MA_1, it verifies that the temporary identity received corresponds to the current temporary identity of $\mathcal{D}_i$ stored in the $\mathcal{CS}$. If they do not match, the protocol gets terminated. Thus, a one-sided update results in desynchronization. In such a scenario, $\mathcal{D}_i$ can initiate a protocol session with one of the temporary identities from the set of temporary identities $S_{\mathcal{D}_i}$. This helps the $\mathcal{CS}$ to resynchronize with $\mathcal{D}_i$ on the temporary identity of $\mathcal{D}_i$, and the protocol execution follows the remaining steps. Hence, even if the attacker drops the message MA_2 so that the temporary identity will not be updated at the device, it will not affect the protocol execution. Thus, the proposed protocol is secure against desynchronization and corresponding DoS attacks.
- **Resistance to Replay Attacks:** The parameters r_1 and c_1 used in MA_1 are generated in every iteration of the protocol. Similarly, the parameters r_2 and c_2 used in MA_2 are also generated in every authentication iteration. Thus, the adversary cannot replay MA_1 and MA_2. Thus, the proposed protocol offers resistance to replay attacks.
- **Mutual Authentication:** $\mathcal{D}_i$ composes MA_1 as $MA_1 = \{TID_{\mathcal{D}i}, c_1, r_1, V_1\}$. In MA_1, k_1 is encapsulated using $p_{\mathcal{CS}}$ to get the ciphertext c_1. Only the $\mathcal{CS}$ knows $s_{\mathcal{CS}}$, to decapsulate k_1 from c_1. Similarly, the $\mathcal{CS}$ encapsulates k_2 as $(c_2, k_2) \leftarrow Enc(p_{\mathcal{D}i})$ and composes $MA_2 = \{Ack, c_2, r_2, V_2\}$. Only a registered IoT device $\mathcal{D}_i$ knows $s_{\mathcal{D}i}$, to decapsulate k_2 from c_2. Thus, the proposed protocol ensures mutual authentication between $\mathcal{D}_i$ and the $\mathcal{CS}$.
- **Session Key Agreement:** Both $\mathcal{D}_i$ and the $\mathcal{CS}$ decapsulate k_1 and k_2, respectively, using their private keys and verify V_1 and V_2. After that, a session key $sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$ is established between $\mathcal{D}_i$ and the $\mathcal{CS}$. Thus, the proposed protocol achieves session key agreement between $\mathcal{D}_i$ and the $\mathcal{CS}$.
- **Message Integrity:** If an adversary modifies any parameter in $MA_1 = \{TID_{\mathcal{D}i}, c_1, r_1, V_1\}$ or $MA_2 = \{Ack, c_2, r_2, V_2\}$, the verification of the authentication parameter $V_1 = h(r_1 \parallel k_1 \parallel TID_{\mathcal{D}i})$ and $V_2 = h(r_2 \parallel k_2 \parallel TID_{\mathcal{D}i})$ will fail, respectively. Thus, any modification in MA_1 and MA_2 will be known and the proposed protocol ensures message integrity.
- **Perfect Forward Secrecy:** In the proposed protocol, the session key is computed as $sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$. Ephemeral random numbers generated by QRNG, r_1 and r_2, are used in the computation of the session key sk. Hence, even if the long-term credential, $ID_{\mathcal{D}i}$, is known to $\mathcal{A}$, $\mathcal{A}$ cannot compute the previous session keys. This is because $\mathcal{A}$ does not know the ephemeral random numbers of previous sessions to compute previous session keys.
- **ESL Attack Resistance:** In the proposed protocol, the session key is computed as $sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$. Hence, even if the ephemeral secrets r_1 and r_2 are disclosed, the adversary cannot calculate the session keys because he/she must also know the long-term credential, $ID_{\mathcal{D}i}$, to compute sk. Thus, even if the ephemeral secrets are disclosed to an adversary, the session key is secure in the proposed protocol. Thus, the protocol offers ESL attack resistance.
- **Known Key Secrecy:** Even if an adversary captures the session key of a particular session of the protocol, he/she should not be able to derive session keys of other sessions from the leaked session key. This is called known key secrecy. The session key is computed as $sk = h(r_1 \parallel r_2 \parallel k_1 \parallel k_2 \parallel ID_{\mathcal{D}i})$. To compute the session key for another session, the adversary must know the QRNG-generated random numbers (r_1 and r_2) and the keys k_1 and k_2 for the new session. Further, the adversary must also know the long-term credential ($TID_{\mathcal{D}i}$). Since they are not known to the adversary, the adversary cannot derive the session key for the new session even if he/she knows the session key of a particular session. As a result, the proposed protocol ensures known key secrecy.

VI. Performance Analysis

In this section, we first compare the security features provided by the proposed protocol with that of similar protocols. Then, we analyze and compare the computation cost, the storage cost, and energy consumption cost of the proposed protocol.

A. Security and Functionality Features

The key features of the proposed protocol are its resistance to attacks by quantum computers and the use of QRNG. The protocols in [4], [11], [12], [14], [15], [21], and [24] offer several security features as shown in Table I. However, the schemes in [4], [11], [12], [14], [15], [21] do not provide resistance to attacks by quantum computers. Further, QRNG is used in the proposed protocol which ensures that secure, true random numbers are used in the proposed protocol. This feature is not provided by other protocols.

TABLE I
COMPARISON BASED ON SECURITY AND FUNCTIONALITY FEATURES

Scheme	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	S11	S12	S13	S14	S15	S16
Gope and Sikdar [4]	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	N
Banerjee et al. [11]	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
Srinivas et al. [12]	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
Chaudhry et al. [14]	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	N
Thakare and Kim [15]	N	Y	Y	Y	Y	Y	N	Y	Y	Y	N	Y	N	Y	Y	N
Challa et al. [21]	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
Chaudhary et al. [24]	Y	Y	Y	N	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	N
Proposed Protocol	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

S1: Protection against quantum computer-enabled attacks; **S2:** Mutual authentication; **S3:** Session key; **S4:** Replay attack protection;
S5: Protection against impersonation attacks; **S6:** Eavesdropping and MITM attack protection; **S7:** Strong anonymity;
S8: Resistance to Key Replicating Attacks; **S9:** Perfect forward secrecy;
S10: Known key secrecy; **S11:** Unlinkability; **S12:** ESL resistance; **S13:** Formal Security Proof;
S14: Message Integrity; **S15:** Resilience Against Desynchronization and DoS Attacks; **S16:** Use of QRNG

Additionally, the proposed protocol offers mutual authentication, session key agreement, resistance to replay attacks, protection against impersonation attacks, resistance to eavesdropping and MITM attacks, strong anonymity, resistance to key replicating attacks, perfect forward secrecy, known key secrecy, unlinkability, ESL attack resistance, message integrity, and resilience against desynchronization and DoS attacks, as discussed in Section V-B. Suppose an adversary modifies the last message between the device and the server in [4]. In that case, the parameters will be updated only at one end, resulting in the desynchronization of the parameters. Hence, the scheme in [4] is vulnerable to desynchronization attacks. Similarly, the scheme in [14] is also vulnerable to desynchronization attacks. The scheme in [15] always uses the same identity for the embedded device. Hence, it does not provide strong anonymity and unlinkability. Formal security proof of the protocol is also not provided in [15]. Though the scheme in [24] is quantum-secure, it is vulnerable to replay attacks and does not offer perfect forward secrecy. From Table I, it can be noted that the proposed protocol offers all conventional security features and resistance to attacks by quantum computers. Thus, the proposed protocol provides superior security and functionality features compared to other similar schemes.

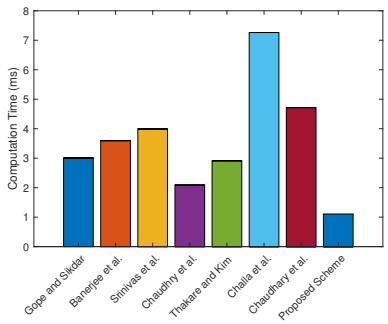
B. Computation Cost

Next, we estimate the proposed protocol's computation cost by computing the execution time of various cryptographic operations. The IoT devices and the $\mathcal{CS}$ must register with the $\mathcal{TA}$ only once. Hence, the registration phase is executed only once for each $\mathcal{D}_i$ and the $\mathcal{CS}$. However, the authentication and key agreement phase is executed whenever $\mathcal{D}_i$ sends data to the $\mathcal{CS}$. Hence, we evaluate the computation cost of the authentication and key agreement phase. In this phase, $\mathcal{D}_i$ executes one decapsulation, one encapsulation, eight concatenation, and three hash operations. The $\mathcal{CS}$ also executes one decapsulation, one encapsulation, eight concatenation, and three hash operations in this phase before session key agreement. Let t_e, t_d, t_c, and t_h denote the time taken by encapsulation, decapsulation, concatenation, and hash operations, respectively. For the calculation of Kyber encapsulation and decapsulation execution time, we used liboqs [48], which is an open-source C library for quantum-safe KEM under the MIT license. In liboqs, the algorithms are implemented based on the resources from the NIST post-quantum cryptography standardization project [48]. For Kyber 512, the sizes of public key, secret key, cipher text, and shared secret are 800, 1632, 768, and 32 bytes, respectively [49]. The simulations were conducted on a personal computer with Intel (R) Core (TM) i5-11320H @3.20 GHz and 8 GB of RAM for the $\mathcal{CS}$. Raspberry Pi 4 Model B, 1.5GHz, 4 cores, 2GB RAM was used for the analysis of $\mathcal{D}_i$. From our analysis, t_e is 0.2610 and 0.5100 ms, t_d is 0.2721 ms and 0.5311, t_h is 0.0023 ms and 0.0170 ms, and t_c is 0.0011 ms and 0.0018 ms for the $\mathcal{CS}$ and $\mathcal{D}_i$, respectively. Hence, the total computation cost at $\mathcal{D}_i$ is 1.1065 ms and at the $\mathcal{CS}$ is 0.5488 ms during one iteration of authentication and key agreement.

TABLE II
COMPUTATION COST DURING AUTHENTICATION AND KEY AGREEMENT

Scheme	IoT Device	Server//Service Provider
Gope and Sikdar [4]	$5t_h + 2t_{PUF}$ $+t_{FEG}$ ≈ 3.005 ms	$5t_h + t_{FER}$ ≈ 4.645 ms
Banerjee et al. [11]	$12t_h + 3t_{ed}$ $+t_{FEG}$ ≈ 3.5974 ms	$7t_h + 7t_{ed}$ ms ≈ 1.3405 ms
Srinivas et al. [12]	$15t_h + 5t_{ecm}$ $+t_{FEG} + t_{eca}$ ≈ 3.9883 ms	$20t_h + 6t_{ecm}$ $+t_{eca}$ ≈ 2.8287 ms
Chaudhry et al. [14]	$10t_h + 8t_{ed}$ $+8t_c$ ≈ 2.0868 ms	$3t_h + 4t_{ed}$ $+16t_c$ ≈ 0.7813
Thakare and Kim [15]	$4t_h + 4t_{ecm}$ ≈ 2.912 ms	$3t_h + 3t_{ecm}$ ≈ 1.3305 ms
Challa et al. [21]	$9t_h + 10t_{ecm}$ ≈ 7.263 ms	$3t_h + 4t_{ecm}$ ≈ 4.0069 ms
Chaudhary et al. [24]	$14t_h + 4t_{om} + 2t_{am}+$ 4 $t_{ch} = 4.714$ ms	$10t_h + 2t_{om}+$ $t_g = 1.24$ ms
Proposed scheme	$t_e + t_d + 8t_c$ +3 t_h ≈ 1.1065 ms	$t_e + t_d + 8t_c$ +3 t_h ≈ 0.5488 ms

Next, we compare the computation cost of the proposed protocol with that of similar schemes: Gope and Sikdar [4], Banerjee et al. [11], Srinivas et al. [12], Chaudhry et al. [14], Thakare and Kim [15], Challa et al. [21], and Chaudhary et al. [24]. Let t_{PUF}, t_{FEG}, t_{FER}, t_{ed}, t_{ecm}, t_{eca}, t_{om}, t_{am}, t_{ch}, and t_g represent the time taken for PUF response generation, fuzzy extractor generation, fuzzy extractor reconstruction, symmetric encryption/decryption,



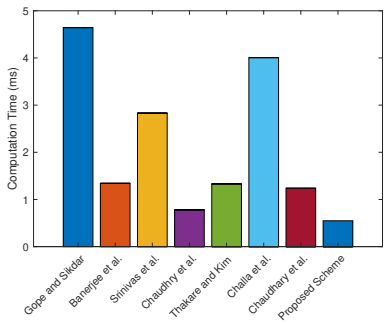


Fig. 4. Computation cost at an IoT device (in ms). Fig. 5. Computation cost at server (in ms).

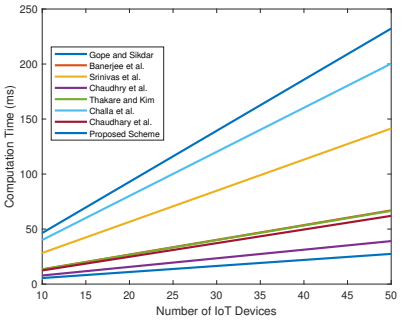


Fig. 6. Computation cost at server as a function of number of IoT devices.

ECC point multiplication, ECC point addition, scalar multiplication in ring Q_q, multiplication and addition in Q_q, characteristics function, and sampling from the Gaussian distribution, respectively. We obtained t_{PUF}, t_{FEG}, and t_{FER} from the scheme in [4]. The computation costs of various schemes are summarised in Table II. We have also plotted the computation costs incurred during the authentication and the key agreement phase at the IoT device and the server in Figures 4 and 5, respectively. Figures 4 and 5 demonstrate that our scheme has the lowest computation cost compared to other similar schemes.

Next, we perform a scalability analysis to estimate the computation cost at the server when the number of IoT devices increases. Figure 6 illustrates the computation cost at the server as a function of the number of IoT devices. The cost increases linearly with the number of IoT devices. The scalability analysis shows that the proposed protocol is highly scalable.

C. Storage Cost

Next, we compute the amount of memory required on the IoT device to store the data required for protocol execution. In the proposed protocol, $\mathcal{D}_i$ stores $TID_{\mathcal{D}i}$, $S_{\mathcal{D}_i}$, $ID_{\mathcal{CS}}$, and $pk_{\mathcal{CS}}$. Let the length of $TID_{\mathcal{D}i}$ and $ID_{\mathcal{CS}}$ be 128 bits each and $S_{\mathcal{D}_i}$ stores four temporary identities. The length of the public key $pk_{\mathcal{CS}}$ for Kyber 512 is 800 bytes, thus making the total storage cost 896 bytes. The keys used in PQC are generally larger than the classic public cryptography keys [45], which increases the storage cost in PQC schemes compared to traditional schemes. Hence, we only compare the storage cost with the PQC scheme in [24]. The scheme in [24] stores two elements from ring Q_q, with each element having a size of 4094 bits. Together with these elements, ID and hash values stored, the total storage cost of the scheme in [24] is 1152 bytes. We have plotted the storage costs in Figure 7, which illustrates that the proposed protocol has a lower storage cost than the scheme in [24].

D. Energy Consumption Cost

We adopt the method used in [50] for energy consumption cost analysis. Data transmission for vehicular networks uses

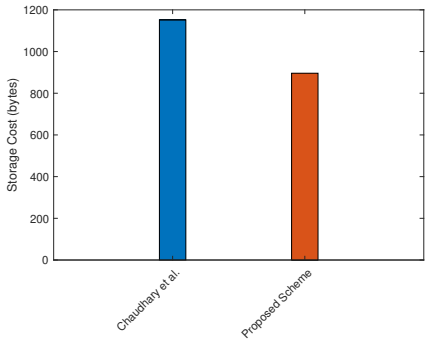


Fig. 7. Storage cost at IoT device (in bytes).

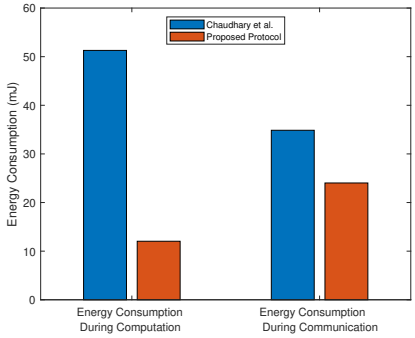


Fig. 8. Energy consumption (in mJ).

Dedicated Short Range Communication (DSRC) defined in IEEE 802.11p [50]. From IEEE standards, data transmission is at 5.8 GHz frequency, 25 dBm transmit power, and 6 Mbps data rate [51]. The energy consumption of the proposed protocol has two components: E_{comp} corresponding to the computation and E_{commu} corresponding to the communication cost during the authentication phase. $E_{comp} = T_{comp} \times P_{cpu}$ where T_{comp} is the computation cost and P_{cpu} is the maximum power consumed by the CPU. For wireless communication networks, P_{cpu} is 10.88 W [50]. $E_{commu} = \frac{S_{msg} \times P_{cpu}}{Rate_{data}}$ where S_{msg} is the the size

of message and $Rate_{data}$ is the data rate for vehicular communications (6000 Kbps) [50]. Two messages are exchanged in the proposed protocol: $MA_1 = \{TID_{\mathcal{D}i}, c_1, r_1, V_1\}$ and $MA_2 = \{Ack, c_2, r_2, V_2\}$. Since the length of the ciphertext is 768 bytes, MA_1 and MA_2 require (128 + 6144 + 128 + 256) = 6656 bits and (64 + 6144 + 128 + 256) = 6592 bits, respectively. Thus, E_{comp} = 12.039 mJ and E_{commu} is 24.023 mJ. Since the keys used in PQC are generally larger than the classic public cryptography keys, the energy consumption cost in PQC schemes is more compared to traditional schemes. Hence, we only compare the energy consumption of the proposed protocol with the PQC scheme in [24]. E_{comp} = 51.288 mJ for the scheme in [24]. Five messages are exchanged in [24] involving 19226 bits. Hence, E_{commu} is 34.86 mJ for [24]. We have plotted the energy consumption costs in Figure 8, which illustrates that the proposed protocol has lower energy consumption cost than the scheme in [24].

VII. QRNG System

A. QRNG Experimental Setup

We employ the experimental setup for the QRNG system given in [52] to generate the random numbers. The QRNG system is illustrated in Figure 9.

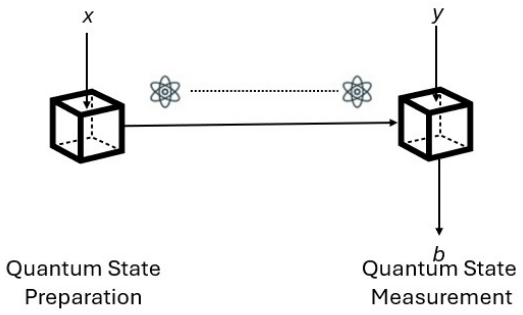


Fig. 9. QRNG experimental setup.

Random number generation using QRNG involves quantum state preparation and quantum state measurement. In the quantum state preparation phase, based on the random variable x, coherent states undergo modulation in the Quadrature Phase Shift Keying (QPSK) format. They are measured by an uncharacterized balanced homodyne detector with the choice of basis determined by y during quantum state measurement. The outcome is a random number, denoted by $b \in \{0, 1\}$.

We use an external cavity semiconductor laser with a central wavelength of 1550 nm and a linewidth of 50 kHz. It splits into two paths: one for quantum state preparation and the other for the Local Oscillator (LO) used in balanced homodyne detection. In the signal path, an Intensity Modulator (IM) initially shapes the continuous-wave laser into pulses, each with a width of 4 ns. A Phase Modulator (PM) is employed to modulate the phase of the quantum states. Finally, the signals are attenuated to single-photon energy level to generate the QPSK quantum states $\{\alpha e^{ixpi/2}\}$ where $x \in \{0, 1, 2, 3\}$.

For quantum state measurement, a fiber-coupled homodyne detector with high efficiency and low noise is used. The overall efficiency of the photodiodes, including the coupling loss, is 98.3% and 98.8%, respectively. The 3 dB bandwidth of the homodyne detector is 72 MHz, and the clearance (shot noise to electronic noise ratio) is 16.94 dB with a 10 mW LO [52]. With this setup, the random numbers were generated.

B. Discussion on QRNG Hardware and QRNG Integration in IoT devices

Integrating QRNGs into IoT ecosystems enhances security, especially in authentication applications. With technological advancements, the accessibility and cost efficiency of QRNGs have been significantly improved. Research has produced integrated QRNGs on photonic chips [41]–[43], and commercially available QRNG chips can now be readily integrated into application systems such as IoT devices. As quantum hardware matures, costs are expected to drop further. Therefore, there are minimal technological obstacles regarding cost, hardware availability, and scalability in IoT deployments when using QRNGs for practical applications.

VIII. Simulation of the Proposed Protocol Using NS3

In this section, we present the experimental details and results of the simulation of the protocol using the network simulator NS3 [53]. We measure the performance of the proposed protocol using the metrics: network throughput (in Kbps), end-to-end delay (in seconds), and packet delivery ratio.

A. Simulation Settings

Simulations were conducted using NS 3.38 on Ubuntu 18.04.6 LTS. We consider three different scenarios in the simulation: Scenario A with 20, Scenario B with 40, and Scenario C with 60 IoT devices. In all three scenarios, we consider one cloud server. The simulation parameters are given in Table III.

B. Discussion of Results

Throughput: Throughput refers to the number of bits transmitted in unit time. Throughput is calculated as $\frac{n_p \times n_i}{t}$ where n_p is the number of packets received, n_i is the size of the packet (number of bits) and t is the total time taken. From the simulation of the proposed protocol, throughput values are 795.1 Kbps, 824.2 Kbps, and 844.4 Kbps for scenarios A, B, and C, respectively. Figure 10 illustrates the network throughput (in Kbps) of the proposed protocol under these three scenarios. As the number of IoT devices increases, the number of exchanged messages also increases. Hence, the throughput increases as the number of IoT devices increases.

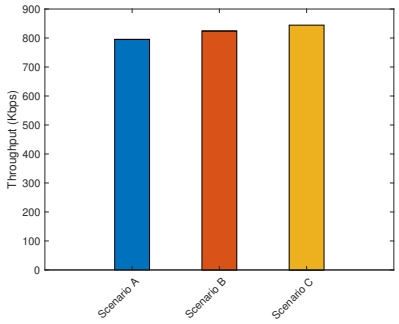


Fig. 10. Comparison of Throughput.

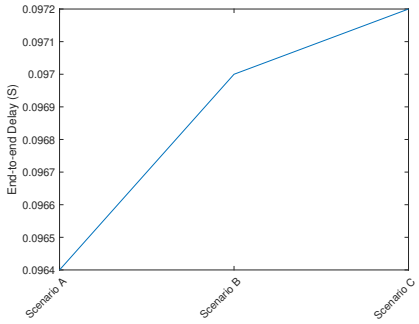


Fig. 11. Comparison of end-to-end delay.

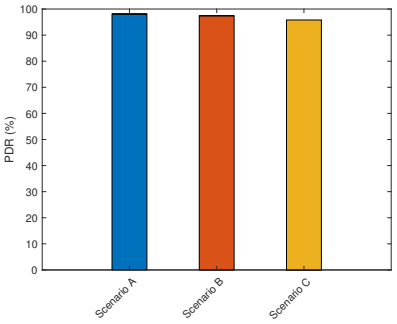


Fig. 12. Comparison of PDR.

TABLE III
PARAMETERS USED IN NS3 SIMULATION

Parameter	Value
Operating System	Ubuntu 18.04.6 LTS
Simulator	NS 3.38
Network Simulation Area	1000 $\times$1000m^2
Simulation Time	1800 s
Scenarios	A, B, C
No. of IoT devices	20,40,60 for Scenarios A,B,C
No. of cloud servers	1 for all scenarios
Routing Protocol	OLSR
Propagation Loss Model	TwoRayGround PropagationLossModel
Mobility Model	RandomDirection2D
Mobility	20 mps
MAC Protocol	IEEE 802.11
Channel Bandwidth	6Mbps

End-to-End Delay: End-to-end delay is the time taken while sending the packets from the source to the destination. From the simulation of the proposed protocol, end-to-end delay values are 0.0964 s, 0.097 s, and 0.0972 s for scenarios A, B, and C, respectively. Figure 11 illustrates the end-to-end delay values for these three scenarios. There are more IoT devices in Scenario B than in Scenario A. With the increase in the number of IoT devices, more messages are exchanged, and hence, the congestion increases from Scenario A to Scenario B. As a result, the end-to-end delay in Scenario B is more than that in Scenario A. Similarly, the end-to-end delay in Scenario C is more than that in Scenario B.

Packet Delivery Ratio: Packet Delivery Ratio (PDR) is the ratio of the number of packets received to the number of packets sent by the sender. Packet delivery ratio is calculated using the following equation:

$$\text{Packet Delivery Ratio} = \frac{n_r}{n_s}$$

where n_r and n_s represent the number of packets received and packets sent by the sender, respectively. From the simulations, the PDRs for scenarios A, B, and C are 98.1%, 97.4%, and 95.8%, respectively. Figure 12 illustrates the PDR for these three scenarios. The PDR results show that as the number of IoT devices increases, the PDR decreases due to congestion.

IX. CONCLUSION

In this paper, we presented a quantum-safe authentication protocol for IoT-enabled transportation systems. With the advent of quantum computing, traditional cryptographic techniques face an imminent threat, affecting the security of critical infrastructure like transportation networks. Our proposed protocol leverages quantum-resistant cryptographic algorithm CRYSTALS-Kyber and QRNG to ensure security against classical and quantum attacks. The performance analysis demonstrated that the proposed protocol offers better security features with lower computation cost compared to other similar schemes. This work provides a solid foundation for future research in quantum-safe protocols within transportation and other IoT-based sectors, contributing to the broader goal of securing critical infrastructure in a post-quantum era. As noted in the performance analysis, the keys used in PQC are generally larger than the classic public cryptography keys, which results in increased storage cost and energy consumption. In future work, we plan to improve the proposed protocol by reducing storage cost and energy consumption. We will also explore providing physical protection to the IoT device so that an adversary cannot extract private keys even after capturing the device.

X. ACKNOWLEDGEMENT

This research is supported by the National Research Foundation, Singapore and Infocomm Media Development Authority under its Future Communications Research Development Programme, under grant FCP-NUS-RG-2022-019. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of National Research Foundation, Singapore and Infocomm Media Development Authority.

REFERENCES

[1] M. Chernyshev, Z. Baig, O. Bello, and S. Zeadally, "Internet of things (iot): Research, simulators, and testbeds," *IEEE Internet of Things Journal*, vol. 5, no. 3, pp. 1637–1647, 2017.

[2] F. Zantalis, G. Koulouras, S. Karabetsos, and D. Kandris, "A review of machine learning and iot in smart transportation," *Future Internet*, vol. 11, no. 4, p. 94, 2019.

[3] G. Bedi, G. K. Venayagamoorthy, R. Singh, R. R. Brooks, and K.-C. Wang, "Review of internet of things (iot) in electric power and energy systems," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 847–870, 2018.

[4] P. Gope and B. Sikdar, "Lightweight and privacy-preserving two-factor authentication scheme for iot devices," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 580–589, 2018.

[5] N. A. Khan, A. Awang, and S. A. A. Karim, "Security in internet of things: A review," *IEEE access*, vol. 10, pp. 104 649–104 670, 2022.

[6] C. Xenofontos, I. Zografopoulos, C. Konstantinou, A. Jolfaei, M. K. Khan, and K.-K. R. Choo, "Consumer, commercial, and industrial iot (in) security: Attack taxonomy and case studies," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 199–221, 2021.

[7] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 1994, pp. 124–134.

[8] J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehlé, "Crystals-kyber: a cca-secure module-lattice-based kem," in *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2018, pp. 353–367.

[9] "CRYSTALS-Kyber Algorithm," Online,https://www.ibm.com/docs/en/zos/2.5.0?topic=cryptography-crystals-kyber-algorithm, [Accessed: Jan 2024].

[10] X. Ma, X. Yuan, Z. Cao, B. Qi, and Z. Zhang, "Quantum random number generation," *npj Quantum Information*, vol. 2, no. 1, pp. 1–9, 2016.

[11] S. Banerjee, V. Odelu, A. K. Das, J. Srinivas, N. Kumar, S. Chattopadhyay, and K.-K. R. Choo, "A provably secure and lightweight anonymous user authenticated session key exchange scheme for internet of things deployment," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8739–8752, 2019.

[12] J. Srinivas, A. K. Das, M. Wazid, and A. V. Vasilakos, "Designing secure user authentication protocol for big data collection in iot-based intelligent transportation system," *IEEE Internet of Things Journal*, vol. 8, no. 9, pp. 7727–7744, 2020.

[13] P. Zhang, Y. Wang, G. S. Aujla, A. Jindal, and Y. D. Al-Otaibi, "A blockchain-based authentication scheme and secure architecture for iot-enabled maritime transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 2322–2331, 2022.

[14] S. A. Chaudhry, A. Irshad, M. A. Khan, S. A. Khan, S. Nosheen, A. A. AlZubi, and Y. B. Zikria, "A lightweight authentication scheme for 6g-iot enabled maritime transport system," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 2, pp. 2401–2410, 2021.

[15] A. Thakare and Y.-G. Kim, "Secure and efficient authentication scheme in iot environments," *Applied Sciences*, vol. 11, no. 3, 2021. [Online]. Available: https://www.mdpi.com/2076-3417/11/3/1260

[16] S.-K. Yang, Y.-M. Shiue, Z.-Y. Su, I.-H. Liu, and C.-G. Liu, "An authentication information exchange scheme in wsn for iot applications," *IEEE access*, vol. 8, pp. 9728–9738, 2020.

[17] S. Ju and Y. Park, "Provably secure lightweight mutual authentication and key agreement scheme for cloud-based iot environments," *Sensors*, vol. 23, no. 24, p. 9766, 2023.

[18] S. Son, D. Kwon, S. Lee, Y. Jeon, A. K. Das, and Y. Park, "Design of secure and lightweight authentication scheme for uav-enabled intelligent transportation systems using blockchain and puf," *IEEE Access*, vol. 11, pp. 60 240–60 253, 2023.

[19] P. Bagga, A. K. Das, M. Wazid, J. J. P. C. Rodrigues, K.-K. R. Choo, and Y. Park, "On the design of mutual authentication and key agreement protocol in internet of vehicles-enabled intelligent transportation system," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 2, pp. 1736–1751, 2021.

[20] H. Vasudev, V. Deshpande, D. Das, and S. K. Das, "A lightweight mutual authentication protocol for v2v communication in internet of vehicles," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 6, pp. 6709–6717, 2020.

[21] S. Challa, M. Wazid, A. K. Das, N. Kumar, A. Goutham Reddy, E.-J. Yoon, and K.-Y. Yoo, "Secure signature-based authenticated key establishment scheme for future iot applications," *IEEE Access*, vol. 5, pp. 3028–3043, 2017.

[22] D. S. Gupta, S. H. Islam, M. S. Obaidat, A. Karati, and B. Sadoun, "Laac: Lightweight lattice-based authentication and access control protocol for e-health systems in iot environments," *IEEE Systems Journal*, vol. 15, no. 3, pp. 3620–3627, 2020.

[23] D. Mishra, M. Singh, P. Rewal, K. Pursharthi, N. Kumar, A. Barnawi, and R. S. Rathore, "Quantum-safe secure and authorized communication protocol for internet of drones," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 12, pp. 16 499–16 507, 2023.

[24] D. Chaudhary, U. Kumar, and K. Saleem, "A construction of three party post quantum secure authenticated key exchange using ring learning with errors and ecc cryptography," *IEEE Access*, vol. 11, pp. 136 947–136 957, 2023.

[25] D. Chaudhary, P. K. Dadsena, A. Padmavathi, M. M. Hassan, B. F. Alkhamees, and U. Kumar, "Anonymous quantum safe construction of three party authentication and key agreement protocol for mobile devices," *IEEE Access*, 2024.

[26] R. P. Parameswarath and B. Sikdar, "Quantum-safe authentication protocol using post-quantum key encapsulation mechanism for transportation systems," in *2024 IEEE 4th International Conference on Sustainable Energy and Future Electric Transportation (IEEE SeFeT 2024)*. IEEE, 2024.

[27] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *International conference on the theory and applications of cryptographic techniques*. Springer, 2001, pp. 453–474.

[28] D. Dolev and A. Yao, "On the security of public key protocols," *IEEE Transactions on information theory*, vol. 29, no. 2, pp. 198–208, 1983.

[29] G. Tasopoulos, J. Li, A. P. Fournaris, R. K. Zhao, A. Sakzad, and R. Steinfeld, "Performance evaluation of post-quantum tls 1.3 on resource-constrained embedded systems," in *International Conference on Information Security Practice and Experience*. Springer, 2022, pp. 432–451.

[30] M. Mehic, L. Michalek, E. Dervisevic, P. Burdiak, M. Plakalovic, J. Rozhon, N. Mahovac, F. Richter, E. Kaljic, F. Lauterbach *et al.*, "Quantum cryptography in 5g networks: A comprehensive overview," *IEEE Communications Surveys & Tutorials*, vol. 26, no. 1, pp. 302–346, 2023.

[31] M. Abdalla, P.-A. Fouque, and D. Pointcheval, "Password-based authenticated key exchange in the three-party setting," in *Public Key Cryptography-PKC 2005: 8th International Workshop on Theory and Practice in Public Key Cryptography, Les Diablerets, Switzerland, January 23-26, 2005. Proceedings 8*. Springer, 2005, pp. 65–84.

[32] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," *Journal of the ACM (JACM)*, vol. 56, no. 6, pp. 1–40, 2009.

[33] X. Ma, X. Yuan, Z. Cao, B. Qi, and Z. Zhang, "Quantum random number generation," *npj Quantum Inf.*, vol. 2, p. 16021, Jun. 2016.

[34] M. Herrero-Collantes and J. C. Garcia-Escartin, "Quantum random number generators," *Rev. Mod. Phys.*, vol. 89, no. 1, p. 015004, 2017.

[35] A. Acín and L. Masanes, "Certified randomness in quantum physics," *Nature*, vol. 540, no. 7632, pp. 213–219, Dec. 2016.

[36] W.-Z. Liu, M.-H. Li, S. Ragy, S.-R. Zhao, B. Bai, Y. Liu, P. J. Brown, J. Zhang, R. Colbeck, J. Fan, Q. Zhang, and J.-W. Pan, "Device-independent randomness expansion against quantum side information," *Nat. Phys.*, vol. 17, no. 4, pp. 448–451, 2021.

[37] L. K. Shalm, Y. Zhang, J. C. Bienfang, C. Schlager, M. J. Stevens, M. D. Mazurek, C. Abellán, W. Amaya, M. W. Mitchell, M. A. Alhejji, H. Fu, J. Ornstein, R. P. Mirin, S. W. Nam, and E. Knill, "Device-independent randomness expansion with entangled photons," *Nat. Phys.*, vol. 17, no. 4, pp. 452–456, Apr. 2021.

[38] D. Frauchiger, R. Renner, and M. Troyer, "True randomness from realistic quantum devices," *ArXiv13114547 Quant-Ph*, Nov. 2013.

[39] F. Acerbi, N. Massari, L. Gasparini, A. Tomasi, N. Zorzi, G. Fontana, L. Pavesi, and A. Gola, "Structures and Methods for Fully-Integrated Quantum Random Number Generators," *IEEE J. Sel. Top. Quantum Electron.*, vol. 26, no. 3, pp. 1–8, 2020.

[40] V. Mannalatha, S. Mishra, and A. Pathak, "A comprehensive review of quantum random number generators: Concepts, classification and the origin of randomness," *Quantum Inf Process*, vol. 22, no. 12, p. 439, Dec. 2023.

[41] C. Wang, I. W. Primaatmaja, H. J. Ng, J. Y. Haw, R. Ho, J. Zhang, G. Zhang, and C. Lim, "Provably-secure quantum randomness expansion with uncharacterised homodyne detection," *Nat. Commun.*, vol. 14, no. 1, p. 316, Jan. 2023.

[42] S. Q. Ng, G. Zhang, C. Lim, and C. Wang, "A chip-integrated homodyne detection system with enhanced bandwidth performance

for quantum applications," *Quantum Sci. Technol.*, vol. 9, no. 4, p. 045010, Jul. 2024.

[43] G. Zhang, I. W. Primaatmaja, Y. Chen, S. Q. Ng, H. J. Ng, M. Pistoia, X. Gong, K. T. Goh, C. Wang, and C. Lim, "Self-testing quantum randomness expansion on an integrated photonic chip," 2024.

[44] K. Hashimoto, S. Katsumata, and T. Wiggers, "Bundled authenticated key exchange: A concrete treatment of (post-quantum) signal's handshake protocol," Cryptology ePrint Archive, Paper 2025/040, 2025. [Online]. Available: https://eprint.iacr.org/2025/040

[45] A. Shahidinejad and J. Abawajy, "An all-inclusive taxonomy and critical review of blockchain-assisted authentication and session key generation protocols for iot," *ACM Computing Surveys*, vol. 56, no. 7, pp. 1–38, 2024.

[46] K. Ramya, J. Gopalakrishnan, B. Chokkalingam, R. Verma, and L. Mihet-Popa, "A complete review of electromagnetic interference in electric vehicle," *IEEE Access*, 2025.

[47] J. Van Waes, J. Vankeirsbilck, D. Pissoort, and J. Boydens, "Electromagnetic interference in the internet of things: An automotive insight," in *2017 XXVI International Scientific Conference Electronics (ET)*. IEEE, 2017, pp. 1–4.

[48] "liboqs," Online,https://openquantumsafe.org/liboqs/, [Accessed: Jan 2024].

[49] "Kyber," Online,https://openquantumsafe.org/liboqs/algorithms/kem/kyber, [Accessed: Jan 2024].

[50] J. Lee, G. Kim, A. K. Das, and Y. Park, "Secure and efficient honey list-based authentication protocol for vehicular ad hoc networks," *IEEE Transactions on Network Science and Engineering*, vol. 8, no. 3, pp. 2412–2425, 2021.

[51] Z. Hameed Mir and F. Filali, "Lte and ieee 802.11 p for vehicular networking: a performance evaluation," *EURASIP journal on wireless communications and networking*, vol. 2014, no. 1, pp. 1–15, 2014.

[52] C. Wang, I. W. Primaatmaja, H. J. Ng, J. Y. Haw, R. Ho, J. Zhang, G. Zhang, and C. Lim, "Provably-secure quantum randomness expansion with uncharacterised homodyne detection," *Nature Communications*, vol. 14, no. 1, p. 316, 2023.

[53] "ns-3," Online,https://www.nsnam.org/releases/ns-3-42/, [Accessed: Sep 2024].

Rohini Poolat Parameswarath received her Ph.D. in Electrical and Computer Engineering from the National University of Singapore. She is a cybersecurity researcher at the Department of Electrical and Computer Engineering, National University of Singapore. Her research focus is on protocols for security and privacy in vehicular environments. Before joining the National University of Singapore, she was part of a cybersecurity research team at the Singapore University of Technology and Design, Singapore. Before embarking on her career in cybersecurity research, she worked as a software engineer in multinational companies. She is passionate about finding solutions to the current challenges in the cybersecurity landscape. Her papers have been published in prestigious journals and conferences such as IEEE Transactions on Vehicular Technology, IEEE Transactions on Network Science and Engineering, ACM Transactions on Management Information Systems, ACM Transactions on Cyber-Physical Systems, IEEE GLOBECOM, IEEE Vehicular Technology Conference, IEEE Intelligent Vehicles Symposium, IEEE S&P, and Black Hat USA. She is a regular reviewer for IEEE Internet of Things Journal, IEEE Transactions on Vehicular Technology, IEEE Transactions on Wireless Communications, IEEE Access, IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Network Science and Engineering, and IEEE Vehicular Technology Conference.

Chao Wang received his B.Sc. in Physics from Huazhong University of Science and Technology in 2013, followed by a Ph.D. in Physics from the University of Science and Technology of China in 2018. Currently, he serves as a Senior Research Fellow in the Department of Electrical and Computer Engineering at the National University of Singapore. His primary research interests include quantum communication, quantum cryptography, and quantum networks.

Biplab Sikdar is a Professor in the Department of Electrical and Computer Engineering at the National University of Singapore, where he also serves as the Head of the Department of Electrical and Computer Engineering. He received the B. Tech. degree in electronics and communication engineering from North Eastern Hill University, Shillong, India, in 1996, the M.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, India, in 1998, and the Ph.D. degree in electrical engineering from the Rensselaer Polytechnic Institute, Troy, NY, USA, in 2001. He was an Assistant Professor from 2001-2007 and Associate Professor from 2007-2013 in the Department of Electrical, Computer, and Systems Engineering at Rensselaer Polytechnic Institute from 2001 to 2013. He is a recipient of the NSF CAREER award, the Tan Chin Tuan fellowship from NTU Singapore, the Japan Society for Promotion of Science fellowship, and the Leiv Eiriksson fellowship from the Research Council of Norway. His research interests include IoT and cyber-physical system security, network security, and network performance evaluation. Dr. Sikdar is a member of Eta Kappa Nu and Tau Beta Pi. He has served as an Associate Editor for the IEEE Transactions on Communications, IEEE Transactions on Mobile Computing, and IEEE Internet of Things Journal and is an IEEE COMSOC and VTS Distinguished Lecturer and ACM Distinguished Speaker.