sensors

MDPI

Article

Towards Green Computing Oriented Security: A Lightweight Postquantum Signature for IoE

Rinki Rani [1], Sushil Kumar [1], Omprakash Kaiwartya [2,*], Ahmad M. Khasawneh [3], Jaime Lloret [4,5], Mahmoud Ahmad Al-Khasawneh [6], Marwan Mahmoud [7] and Alaa Abdulsalm Alarood [8]

[1] School of Computer and Systems Sciences, Jawaharlal Nehru University (JNU), New Delhi 110067, India; rinki32_scs@jnu.ac.in (R.R.); skdohare@mail.jnu.ac.in (S.K.)
[2] Department of Computer Science, Clifton Campus, Nottingham Trent University, Nottingham NG11 8NS, UK
[3] Department of Mobile Computing, Amman Arab University, Amman 11953, Jordan; a.khasawneh@aau.edu.jo
[4] Integrated Management Coastal Research Institute, Universitat Politecnica de Valencia, 46022 Valencia, Spain; jlloret@dcom.upv.es
[5] School of Computing and Digital Technologies, Staffordshire University, Stoke ST4 2DE, UK
[6] Faculty of Computer & Information Technology, Al-Madinah International University, Kuala Lumpur 57100, Malaysia; mahmoud@outlook.my
[7] Department of Computer and Information Technology, Faculty of Applied Studies, King Abdulaziz University, Jeddah 21589, Saudi Arabia; mmamahmoud@kau.edu.sa
[8] College of Computer Science and Engineering, University of Jeddah, Jeddah 21959, Saudi Arabia; aasoleman@uj.edu.sa
* Correspondence: omprakash.kaiwartya@ntu.ac.uk

check for updates

Citation: Rani, R.; Kumar, S.; Kaiwartya, O.; Khasawneh, A.M.; Lloret, J.; Al-Khasawneh, M.A.; Mahmoud, M.; Alarood, A.A. Towards Green Computing Oriented Security: A Lightweight Postquantum Signature for IoE. *Sensors* **2021**, *21*, 1883. https://doi.org/10.3390/s21051883

Academic Editor: Yuh-Shyan Chen

Received: 5 January 2021
Accepted: 1 March 2021
Published: 8 March 2021

Abstract: Postquantum cryptography for elevating security against attacks by quantum computers in the Internet of Everything (IoE) is still in its infancy. Most postquantum based cryptosystems have longer keys and signature sizes and require more computations that span several orders of magnitude in energy consumption and computation time, hence the sizes of the keys and signature are considered as another aspect of security by green design. To address these issues, the security solutions should migrate to the advanced and potent methods for protection against quantum attacks and offer energy efficient and faster cryptocomputations. In this context, a novel security framework Lightweight Postquantum ID-based Signature (LPQS) for secure communication in the IoE environment is presented. The proposed LPQS framework incorporates a supersingular isogeny curve to present a digital signature with small key sizes which is quantum-resistant. To reduce the size of the keys, compressed curves are used and the validation of the signature depends on the commutative property of the curves. The unforgeability of LPQS under an adaptively chosen message attack is proved. Security analysis and the experimental validation of LPQS are performed under a realistic software simulation environment to assess its lightweight performance considering embedded nodes. It is evident that the size of keys and the signature of LPQS is smaller than that of existing signature-based postquantum security techniques for IoE. It is robust in the postquantum environment and efficient in terms of energy and computations.

Keywords: energy efficiency; green computing; lightweight security; Internet of Things

1. Introduction

The Internet of Everything (IoE) is an interconnection of smart devices, business processes and data structures without any human intervention [1]. It expands applications from digital sensor tools to smart and self-configuring intelligent nodes in distributed hardware to enrich the lives of people [2]. In such smart networks, information security is of paramount importance as all the decisions and actions depend on the accuracy and credibility of the received data [3]. The public key infrastructure (PKI) plays a critical role in information security. In PKI, however, both the sender and the receiver authenticate each other with the help of certificates obtained from the certificate authority. This process can be time-consuming and complex. Identity-based cryptography (IBC) schemes remove these barriers and use public strings such as email addresses or domain names for data encryption and signature verification, instead of digital certificates [4]. The security of IBC depends on solving some mathematical problems such as integer factorization and discrete logarithms. Major recent signature schemes depend on these two mathematical problems, which are infeasible to solve on any classical computer. However, these problems can easily be solved by quantum computers in polynomial time. For instance, Shor's quantum algorithm can solve the integer factorization in polynomial time [5]. Moreover, it can not only forge a signature but also recover private keys. Thus, such system poses serious threats to the modern cryptography. To effectively block these threads, many cryptographers are developing new quantum-resistant algorithms that are unbreakable in the era of quantum computers. Several postquantum cryptography (PQC) classes have been proposed which are currently believed to be quantum resistant, namely: lattice-based [6–8], hash-based [9], code-based PQC [10] and isogeny-based [11].

Over the past few years, isogeny-based cryptography has been gaining a lot of momentum owing to its small key sizes. Various isogeny-based cryptosystems have been published for public key encryption and key exchange protocols [12,13] but later have been broken by a subexponential quantum attack. Recently, a key exchange scheme based on supersingular isogeny Diffie–Hellman (SIDH) has been proposed, for which there is no known subexponential quantum attack [14] and is much faster than ordinary isogeny. SIDH uses supersingular elliptic curves for key exchange and public key encryption [15,16]. Isogeny-based cryptosystems have also been used for digital signatures such as the strong designated verifier signature [17] and the undeniable signature [18]. However, the feasibility of these schemes on resource-constrained devices is not known. The compressed digital signature scheme reduces the public and private key sizes to 336 and 48 bytes, respectively, for the 128-bit quantum security level. Unfortunately, these primary signature schemes are slower than other quantum signature techniques due to their larger signature sizes.

The prime issues in security by green computing for IoE applications are related to the key size, signature and the encryption computation of the postquantum based cryptosystems, which must be kept compact to reduce energy consumption and computation time [19]. Most postquantum based cryptosystems require higher order of magnitude longer keys to provide current the level of protection, which are substantial enough to impact energy requirements and computation time [20]. The use of isogeny curve based postquantum cryptography is considered to be the most practicable solution to the energy required for the shortest key's computation. To efficiently exploit the resistant capability of postquantum cryptography, we use a supersingular isogeny curve and ID-based signature for postquantum cryptography, which requires much shorter keys to maintain the same level of protection and provides user friendly access to the system. In addition to this, it can also reduce the overall energy and time needed for the crypto operations in comparison to postquantum based cryptosystems and therefore facilitate appropriate replacement in sensors, handheld devices, and IoE applications.

In this context, a lightweight postquantum ID-based signature (LPQS) scheme using a supersingular isogeny curve for secure data transmission in the IoE environment is presented. The design of the LPQS scheme aims to provide a signature scheme for the postquantum cryptography and to reduce the complexity of the system with the consumption of fewer system resources. The LPQS scheme uses the identity of the client for the initialization of the process. Further, this scheme uses two isogeny curves for verification to provide double-fold secure encryption. The main contributions of the scheme can be summarized as:

- Firstly, a system model for post quantum security is presented considering its applicability in IoE environments.
- Secondly, the four phases of the execution of the proposed framework LPQS are detailed, where compressed curves are used to reduce the size of keys and the validation of the signature depends on the commutative property of curves.

- Thirdly, the unforgeability of LPQS under an adaptively chosen message attack is proved and security analysis is performed to show its resistance against various cyberattacks.
- Finally, performance analysis and experimental validation of the proposed framework are performed under software simulation environment to assess its lightweight performance in realistic IoE environments considering the embedded nodes.

The rest of the paper is organized as follows. Section 2 presents the recent review of nonquantum and postquantum cryptographic techniques. Section 3 presents the details of the proposed lightweight security framework LPQS. In Section 4 discusses security analysis and experimental comparative performance evaluation considering range of metrics, followed by conclusions presented in Section 5.

2. Related Work

For security in sensor networks, Jao et al. [14] proposed a cryptosystem based on supersingular isogenies for encryption and key exchange which is much faster in contrast to the ordinary isogenies based schemes. This work was further extended by Plut et al. [15] and gave a public key exchange scheme which includes zero-knowledge proof of identity. This model achieves approximately 0.06 s per key exchange runtime operation as presented in test scenario. Costela et al. [16] proposed more efficient algorithms for computing isogenies. This algorithm have claimed to run 2.9 times faster than the scheme by Plut et al. Earlier, the isogeny based cryptographic functions were available only for key exchange protocol or public key encryption scheme. Thereafter, Galbarith et al. [17] proposed the first signature scheme based on supersingular isogeny problems. This scheme is resistant to chosen message attacks in the random oracle model. To achieve a small signature size a time–space trade-off is used which deteriorates the performance of the scheme. Hence, to improve the performance, a signature scheme based on isogeny-based zero-knowledge proof have been suggested which further reduces signature size with small key sizes [18,19]. However, this scheme suffers from poor performance compared to the other postquantum schemes.

Elliptic Curve Cryptosystem (ECC) based models have been very prominent in IoT. Considering the efficiency of ECC, Malasri et al. [20] gave an authentication scheme for medical sensor networks. As a result, this model could maintain confidentiality and message integrity. In this key management scheme, every step computes the message authentication code, which depletes the resources and delays the packets' processing at the receiver end. Further, Oliveira et al. [21] gave a secure scheme for sensor networks based on IBC and proved it to be practical for resource-constrained nodes. In this scheme, senders broadcast their identities with no security measure and it allows adversaries to broadcast several fake identities and helps them to launch denial-of-service (DoS) attacks. This attack reduces the power of low computation devices. Tan et al. [22] proposed an identity-based cryptography scheme for the security of body sensor networks. This approach uses a hash function for public key generation and stores the key on the sensor's flash memory. Further, this model uses the public key for the computation of elliptic curve encryption/decryption using the Elliptic Curve Digital Signature Algorithm (ECDSA). For public key computation, this scheme requires more storage, energy and computation time. Sankaran et al. [23] gave an IDKEYMAN which uses IBC for wireless body area networks parties to exchange symmetric keys. The pairwise symmetric keys support the minimization of energy consumption.

In addition, this approach provides security from replay attacks by using ephemeral values. This technique does not provide protection against other attacks like selective forwarding, Sybil, etc. Li et al. [24] proposed a biometric-based scheme where physiology signals like electrocardiogram are used to create keys and transmits them in a safe mode. This biometric-based scheme improves the network security and increases the lifetime of the model by using fuzzy commitment and an arbitrated-based approach. However, this approach is limited to a wireless body area network only. Ma et al. [25] proposed a practical access control technique based on IBC for the Internet of Things (IoT). This signcryption scheme provides a reduction in energy and less computation cost with large area applicability [26].

Public key cryptographic algorithms depend on the hardness of integer factorization and discrete log problems. However, these algorithms will be vulnerable to attacks from quantum computers. Considerable research has been conducted for postquantum cryptography. Among various postquantum techniques, the lattice-based signatures [27] scheme is prominent and based on the hardness of NTRU (Nth degree Truncated polynomial Ring Units) problems with no algebraic structure. The limitation of these techniques is that they have large public and private keys and are not feasible for many practical applications. Another candidate for postquantum cryptography is multivariate-based signatures [28]. These signatures are based on the multivariate quadratic polynomial problem. These models have a smaller signature but large key sizes and are difficult to scale to higher security levels [29]. Furthermore, hash-based techniques have small key sizes but are inefficient in terms of speed. Hence, none of the abovementioned techniques are feasible for the IoE environment [30]. Because of the small key size, isogeny-based cryptography is a suitable candidate for the IoE environment. An isogeny-based cryptosystem depends on the difficulty of computing isogeny between two given curves of the same order.

The first isogeny-based cryptosystem for public key encryption and the key exchange was a traditional model without considering quantum computing. However, Childs et al. [31] proposed a postquantum algorithm that computes ordinary isogenies in subexponential time. Since the algorithm relies on the commutative property of endomorphism rings, it does not apply to the supersingular singular case [32]. Feo et al. [33] gave a signature model using class group actions for the 128-bit security level. This model uses only a 1 KB signature size and maintains adequate security in the random oracle model. Parrilla et al. [34] have suggested a unified coprocessor framework in order to run the ECC on IoT devices. The group key support strategy is also incorporated for reducing the communication overhead in key distribution. Similarly, to deal with malfunctioning of the IoT enabled systems, Hussein et al. [35] investigated a secure protocol to maintain the secrecy rate in IoT environments and to reduce the energy consumption at IoT nodes. However, both these ECC frameworks are vulnerable against quantum attacks as edge centric faster and efficient security enabler nodes have not been considered to support the security operations of resources constrained IoT nodes. Quantum centric security analyses have been also missing in the analytical investigation of these approaches.

3. Lightweight Postquantum Signature Scheme for IoE

3.1. Preliminaries—Basics of Supersingular Iosgency Curve

Initially, we briefly introduce the supersingular isogeny curve that has been used to design the proposed signature scheme and its problems to prove its resistance against cyberattacks. We consider two elliptic curves E_A, E_B over a finite field F_q also used in [36,37]. An isogeny φ: $E_A \to E_B$ is a nonconstant morphism that preserves the group structure [38]. The degree of an isogeny φ is equal to the degree of φ as a morphism. An isogeny of degree ℓ is called a ℓ-isogeny [39,40]. If φ is separable, then deg φ = #ker φ. If isogeny is separable between two curves, we say that they are isogenous [41]. Tate's theorem [42,43] is that two curves E_A, E_B over F_q are isogenous if and only if $\#E_A(F_q) = \#E_B(F_q)$. An isogeny can be identified by its kernel in such a way that for every finite subgroup G of E_A, there is a unique E_B and a separable isogeny φ: $E_A \to E_B$ with kernel G such that φ: $E_B \cong E_A/G$. To obtain subgroup G we can use Vélu's formulae.

Isogenies with the same domain and range are called as endomorphisms. The set of endomorphisms is maximal order either to quaternion algebra or to an imaginary quadratic field. The curve is supersingular for the first case; otherwise, the curve is ordinary. In the case of a supersingular elliptic curve, there is always a curve in the isomorphism class defined over F_{p^2}, thus its j-invariant is over F_{p^2}. One can construct a so-called isogeny graph for any prime $\ell \neq p$, where an edge and vertex are associated with an l-isogeny and j-invariant, respectively. Next, we present a few hard problems related to supersingular elliptic curves over F_{p^2}.

Problem 1 (computational supersingular isogeny ($CSSI_A$) problem): suppose $\Phi_A : E_0 \to E_A$ to be an isogeny with kernel $(P_A + [\alpha]Q_A)$ where α chose at random from $z/l_A^{e_A}z$ and not divisible by l_A. Find a generator G_A of $(P_A + [\alpha]Q_A)$ where $\{ E_A, \Phi_A(P_C), \Phi_A(Q_C)\}$ is given.

Problem 2 (computational supersingular isogeny ($CSSI_C$) problem): suppose $\Phi_C : E_0 \to E_C$ to be an isogeny with kernel $(P_C + [\beta]Q_C)$ where β chose at random from $z/l_C^{e_C}z$ and not divisible by l_C. Find a generator G_C of $(P_C + [\beta]Q_C)$ where $\{ E_C, \Phi_C(P_A), \Phi_C(Q_A)\}$ is given.

Problem 3 (supersingular isogeny Diffie–Hellman (SIDH) problem): let $\Phi_A : E_0 \to E_A$ be an isogeny with kernel $\langle P_A + [\alpha]Q_A\rangle$, and $\Phi_C : E_0 \to E_C$ be an isogeny with kernel $\langle P_C + [\beta]Q_C\rangle$, where α, β are chosen at random from $z/l_A^{e_A}z$ and $z/l_C^{e_C}z$, respectively. $\{E_A, \Phi_A(P_C), \Phi_A(Q_C), E_C, \Phi_C(P_A), \Phi_C(Q_A)\}$ be given, find j-invariant of $E_0/\langle P_A + [\alpha]Q_A, P_C + [\beta]Q_C)\rangle$.

Problem 4 (supersingular isogeny auxiliary point ccomputation ($SIAPC_A$)): suppose $\Phi_A : E_0 \to E_A$ to be an isogeny with kernel $(P_A + [\alpha]Q_A)$ where α chose at random from $z/l_A^{e_A}z$ and is not divisible by l_A. The supersingular isogeny auxiliary point computation problem is to find the auxiliary point $\Phi_A(P_C)$ and $\Phi_A(Q_C)$, where $\{E, E_A, P_A, Q_A, P_C, Q_C\}$ are given.

Problem 5 (supersingular isogeny auxiliary point computation ($SIAPC_C$)): suppose $\Phi_C : E_0 \to E_C$ to be an isogeny with kernel $(P_C + [\beta]Q_C)$ where β is chosen at random from $z/l_C^{e_C}z$ and is not divisible by l_C. The supersingular isogeny auxiliary point computation problem is to find the auxiliary point $\Phi_c(P_A)$ and $\Phi_C(Q_A)$, where $\{E, E_A, P_A, Q_A, P_C, Q_C\}$ are given.

A signature scheme consists of three polynomial time algorithms: key generation, registration, and validation. We prove the security of the scheme using the existential unforgeable under an adaptively chosen message attack (EU-ACMA) (32). A forger and a challenger play a game where the forger uses the public key and signing oracle model. The forger issues signature queries to the sign oracle to generate a signature σ_i of message m_i and the oracle sends σ_i to the forger. The attack is considered successful when the forger produces a valid signature and message pair different from those generated from the query oracle.

Definition 1. *A digital signature scheme is existentially unforgeable under an adaptively chosen message attack (EU-ACMA) if any adversary Ã cannot produce a valid message–signature pair in polynomial time with access to the signing oracle.*

Setup: Suppose we have a function KeyGen to output key pair (*pk*, *sk*), and challenger give the *pk* to the adversary Ã.

Queries: The adversary Ã issues signature queries to sign oracle Ś to generate valid signature $\sigma_1, \ldots, \sigma_i$ corresponding to messages $M_1, \ldots, M_i$.

Output: Finally, adversary Ã generates a valid message signature pair (M^*, σ^*) and wins the game if $M^* \notin M_i$.

The signature scheme is secure if probability to distinguish between simulated signature and real signature is negligible for adversary Ã with access to signing oracle $(Sign_{sk}(.))$ i.e.,

$$\Pr\left[\begin{array}{c} (pk,\ sk) \leftarrow keyGen(1^n) \\ (M_i, \sigma_i) \leftarrow A^{Sign_{sk}(.)}(\mathrm{pk}) \\ Verify_{PK}(M, \sigma) = 1\ and\ M^* \notin M_i \end{array}\right] \le negl\ (\lambda)$$

3.2. System Model

We consider an IoE environment in which several heterogeneous smart nodes such as an individual human, an organization, sensors, vehicles, smart watches, smart phones are deployed as shown in Figure 1. We classify these smart nodes into two main categories: service provider and client. In the IoE environment, a client can be an organization, an individual human or any device that wants to access services such as health reports collection, banking, e-commerce. The client encrypts the data with its signature and sends it to the service provider. The service provider allows authentic clients to access the service. A service provider provides an organization with three servers: the key generation server, the database server, and the validation server. For individual clients, the key generation server generates the global parameters and public–private keys. The database server maintains the data and the validation server helps in authenticating the clients. The service provider generates appropriate rights using a tag machine and performs key generation, encryption/decryption using the supersingular isogeny curves. It issues the rights to clients based on the service such as a client can view only his/her data for a particular period. The Internet of Everything (IoE) is considered as superset of Internet of Things (IoT). IoE covers the wider concept of connectivity where network intelligence at the edge devices makes it a more complex network tha then IoT. So, basically, it can be considered as an extension of the IoT in terms of network management and network intelligence.

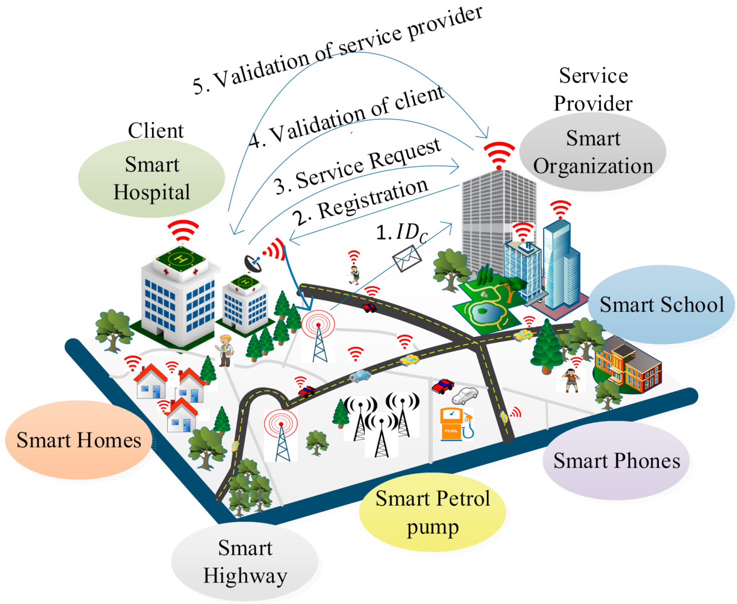


Figure 1. A system model for the lightweight postquantum ID-based signature (LPQS) framework.

To ensure secure data transmission between a service provider and clients, and to reduce the complexity of the system with less consumption of the system resources, we present a LPQS scheme for secure data transmission for an IoE environment. The scheme uses supersingular isogeny curves for the postquantum cryptography signature. The proposed scheme consists of four phases: initialization, registration, signature, and validation. In the first phase, the service provider initializes all the parameters for global access. In the second phase, the service provider calculates the basis points for the clients using the ID of an individual client. The client performs the signature on the data with the help of the service provider in the signature phase. In the validation phase, the clients and service providers validate each other using the two isogeny curves. We want to clarify that "green" means a reduction in the computing requirement for providing security in the IoE environment. The proposed framework LPQS reduces the size of keys and signature for enabling security in the IoE. It also uses keys which can be used for longer period and are flexible in use, further reducing computation at the IoE nodes. Thus, green design means it is energy-efficient for the IoE nodes, as well as computing power efficient for the coordinator nodes at the edge.

3.3. *Lightweight Post Quantum Signature*

Firstly, in the initialization phase, the service provider initializes the system by setting all the global parameters as a set $\{p,\ E,\ P_A,\ Q_A,\ I_A(2),\ I_B(3),\ n,\ m\}$, where the description and use of every parameter is given in Table 1. Isogeny-based cryptosystem uses supersingular elliptic curves over characteristic p, where p is a prime of the form $2^n \times 3^m \times f\ \pm 1$. Here, $n,\ m$ are positive integers such that $2^n\ \simeq\ 3^m$ and f is a small cofactor to ensure p as a prime. This special form of p allows us to efficiently compute isogenies, as given in the next sections. The global parameters generated by service provider include $\{p,\ E,\ P_A,\ Q_A,\ I_A(2),\ I_B(3),\ n,\ m\}$ over the curve E of finite field F_{p^2} of characteristics p with p^2 element. The service provider selects a random integer α, such that $0\ \leq\ \alpha\ \leq 2^n$. The random number α is kept secret as the service provider's secret key. The service provider uses an ephemeral secret key, which changes in every session to support nontraceability. Fix points $P_A,\ Q_A \epsilon E[2^n]$ such that group $\langle P_A,\ Q_A\rangle$ generated by P_A and Q_A in the whole group $E[2^n]$. The elliptic curve points $(P_A,\ Q_A)$ are the global parameters of the supersingular isogeny-based cryptosystem. $GT_A\ = P_A + [\alpha]Q_A$, where $\langle GT_A\rangle$ is the generator of a kernel of service provider which creates a secret subgroup of $E[2^n]$. $E_A = E/\langle GT_A\rangle$ is the elliptic curve that is the image curve under the isogeny $\{\Phi_A\}$.

Table 1. Nomenclature.

Symbol	Description
p	Prime number
E	Elliptic curve over finite field F
$P_A,\ Q_A$	Elliptic curve basis points
$\Phi_A,\ \Phi_C$	Isogeny for supersingular curve $E_A,\ E_C$
$n,\ m$	Positive integers such that $2^n\ \simeq\ 3^m$
GT_A	Generator of a kernel of service provider
f	Small cofactor to ensure p as a prime
r_B	Seed value
ID_C	Identity of client
$\vert\vert$	Concatenation operator
$\oplus$	Xor operator

Secondly, the in registration phase, service provider performs the registration with the help of the client (C) to provide access to the facility/services of the service provider in the IoE environment as shown in Figure 2 and the steps are:

Step 1. The client sends its identity ID_C generated randomly to the service provider through a public channel.

Step 2. After receiving the ID_C, the service provider calculates basis points of client i.e., Q_C and P_C using the ID_C and *right*, which are assigned by service provider as expressed by Equations (1) and (2).

$$Q_C = H(ID_C \ ||\ f) \tag{1}$$

$$P_C = H(right\ ||\ ID_C) \oplus p \tag{2}$$

where, H is a fixed hash function, and rights are the authority assigned to the client. The notation $\oplus$ is the xor function, and $||$ is a concatenation operation.

Step 3. The service provider generates the public key of client as $\{\Phi_A(P_C),\ \Phi_A(Q_C),\ P_C,\ Q_C,\ right\}$ and sends it to the *client*.

Step 4. Upon receiving $\{\Phi_A(P_C),\ \Phi_A(Q_C),\ P_C,\ Q_C, right\}$, the client selects a random number as a secret key from $0 \ \le\ \beta\ \le\ 3^m$. The generator G_C for the kernel of the client is expressed as given by Equation (3).

$$G_C = P_C + [\boldsymbol{\beta}]Q_C \tag{3}$$

where P_C and Q_C are the basis for E_C and $E_C = E/\langle G_C\rangle$.

Step 5. The client computes the image curve E_{AC} and also computes the shared secret value $j(E_{AC})$, where $j(E_{AC})$ is the j-invariant of the image curve E_{AC}.

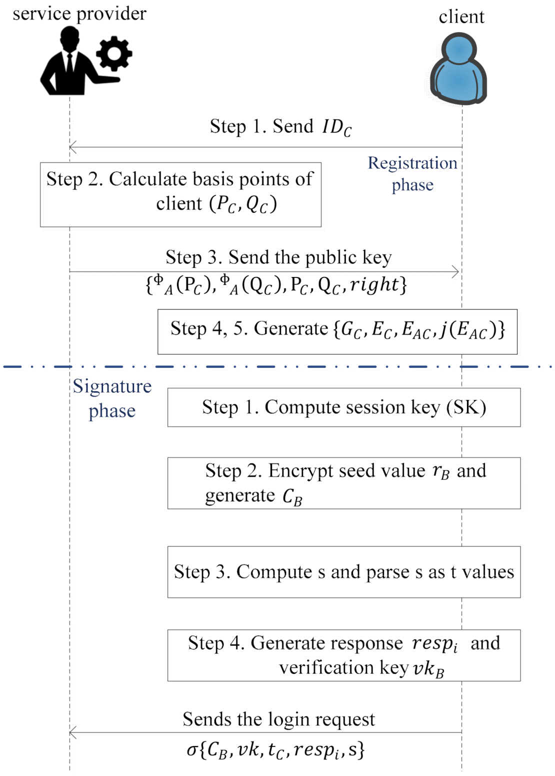

service provider
client
Step 1. Send ID_C
Registration phase
Step 2. Calculate basis points of client (P_C, Q_C)
Step 3. Send the public key $\{\Phi_A(P_C), \Phi_A(Q_C), P_C, Q_C, right\}$
Step 4, 5. Generate $\{G_C, E_C, E_{AC}, j(E_{AC})\}$
Signature phase
Step 1. Compute session key (SK)
Step 2. Encrypt seed value r_B and generate C_B
Step 3. Compute s and parse s as t values
Step 4. Generate response $resp_i$ and verification key vk_B
Sends the login request $\sigma\{C_B, vk, t_C, resp_i, s\}$

Figure 2. Flow diagram of registration and signature.

Thirdly, in the signature phase, the client does the following four steps to sign message m which is shown in Figure 2.

Step 1. The client calculates the $sessionkey\ (sk) = H(t_C, j(E_{AC}), ID_C,\ U,\ V)$, where $U = \Phi_C(P_A),\ V = \Phi_C(Q_A)$ and t_C is the timestamp.

Step 2. Further, encrypt the seed value r_B as expressed by Equation (4).

$$C_B = Enc_{ID_C}(r_B\ \oplus sk),\ \text{for } 1\ \le\ B\ \le\ t \tag{4}$$

Step 3. Compute $s = \ H(m,\ C_1, \ldots, C_t)$. Parse s as t values $CH_B \in \{0,1\}^C$.

Step 4. If $CH_i = 1$ then response $resp_i = (G_C, \Phi_A(G_C))$ else $resp_i = (\Phi_c(G_A))$. $\Phi_A(G_C)$ is only calculated by the service provider and verification key $(vk_B) = h(t_C,\ j(E_{AC}),\ ID_C,\ r_B, CH_B, \text{s})$ for $1 \le B \le t$. The client sends the login request $\sigma\{C_B,\ vk, t_C, resp_i, \text{s}\}$ to the service provider. In this last validation phase the service provider and the client validate each other, which is shown in Figure 3 with stepwise description as follows.

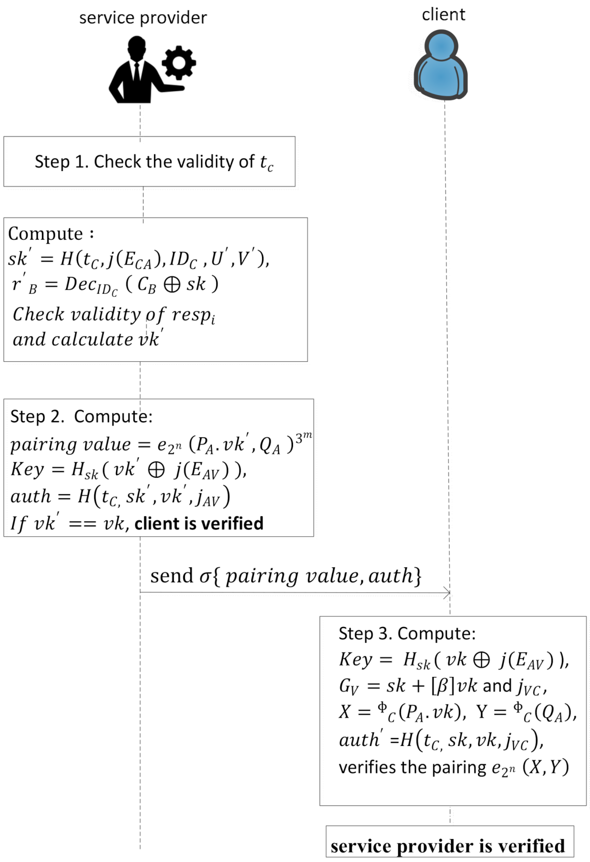

service provider
client
Step 1. Check the validity of t_c
Compute :
$sk' = H(t_C, j(E_{CA}), ID_C, U', V')$,
$r'_B = Dec_{ID_C}(C_B \oplus sk)$
Check validity of $resp_i$
and calculate vk'
Step 2. Compute:
$pairing\ value = e_{2^n}(P_A.vk', Q_A)^{3^m}$
$Key = H_{sk}(vk' \oplus j(E_{AV}))$,
$auth = H(t_{C,} sk', vk', j_{AV})$
$If\ vk' == vk$, client is verified
send $\sigma\{pairing\ value, auth\}$
Step 3. Compute:
$Key = H_{sk}(vk \oplus j(E_{AV}))$,
$G_V = sk + [\beta]vk$ and j_{VC},
$X = \Phi_C(P_A.vk)$, $Y = \Phi_C(Q_A)$,
$auth' = H(t_{C,} sk, vk, j_{VC})$,
verifies the pairing $e_{2^n}(X, Y)$
service provider is verified

Figure 3. The work flow of client and service provider validation.

Step 1. The service provider checks the validity of t_C of received signature σ and if it is valid then proceeds further; otherwise the service provider rejects the request. After checking the t_C validity, the service provider calculates the image of the client with the help of its basis as, $\Phi_A(P_C) = \Phi_C(P_A) = U', \Phi_A(Q_C) = \Phi_C(Q_A) = V'$ and also computes $sk' = H(t_C,\ j(E_{CA}), ID_C, U', V')$ and r'_B as expressed by Equation (5).

$$r'_B = Dec_{ID_C}(C_B \oplus sk) \tag{5}$$

for i = 1 to t, parse s as t values and check if $CH_i = 1$, then parse $resp_i$. Check if $resp_i$ has order 3^m and if G_C generates E_C and $\Phi_A(G_C)$ generates E_{CA}. If $CH_i = 0$, then check if $resp_i$ has order 2^n and generates E_{AC} and $vk' = h(t_C,\ j(E_{CA}), r'_B,\ ID_C,\ CH_{B,}$s). If vk' is equal to vk then *clientC* is authenticated.

Step 2. The service provider computes $pairingvalue = e_{2^n}(P_A.vk',\ Q_A)^{3^m}$ and develops the key and authentication using sk and vk as expressed in Equations (6) and (7), and computes the value of $\Phi_A(sk'),\ \Phi_A(pk'),\ E_V$ and j_{AV} (as shown in Figure 3) and send $\sigma\{\ pairingvalue,\ auth\}$ to the client.

$$Key = H_{sk}\big(vk' \oplus\ j(E_{AV})\big), \tag{6}$$

$$auth = H\big(t_{C,}\ sk',\ vk',\ j_{AV}\big) \tag{7}$$

Step 3. After receiving the signature, the client verifies the authenticity of the service provider and computes $Key =\ \ H_{sk}(vk \oplus\ j(E_{AV}))$ and $G_V\ = sk + [\beta]vk$ and j_{VC} as shown in Figure 3. Further, it calculates $X = \Phi_C(P_A.vk),\ Y = \Phi_C(Q_A),\ auth'$ $=H(t_{C,}\ sk,\ vk,\ j_{VC})$ and also verifies the pairing $e_{2^n}(X, Y)$. Now the service provider is also verified.

4. Security Analysis and Experimental Results

4.1. Mathematical Security Analysis

Theorem 1. *The digital signature LPQS is EU-ACMA in the quantum random oracle model with constraint relation expressed in Equation (8).*

$$\varepsilon\ (1/2^n)\ \left(1-\left(q_q/2^k-4q_h-q_s\right)\right)\left(1-q_q/2^{|F_{P^2}|}\right)\leq \mathbf{Pr}[C] \tag{8}$$

where

$$1/2^n<\frac{1}{4},q_q/2^{|F_{P^2}|}<\frac{1}{3},\ \text{so}\ \frac{\varepsilon}{2}\left(1-\left(q_q/2^k-4q_h-q_s\right)\right)\leq \mathbf{Pr}[C].$$

Proof. Suppose an adversary A exists in the system who can produce valid LPQS signatures. It takes system parameters $\{\ p,\ E,\ P_A,\ Q_A, I_A(2),\ I_B(3),\ n,\ m,\ P_c,\ Q_c\}$, public keys $(E_A, \Phi_A(P_C),\ \Phi_A(Q_C))$ and a verifier $(E_B, \Phi_C(P_A),\ \Phi_C(Q_A))$. The adversary make queries q to the oracle of client C with queries of a signing oracle ($\mathbb{S}$), and a verifying oracle (ν), and a hashing oracle ($\mathcal{H}$). The adversary A aims at producing $\sigma\{C_B,\ vk, t_C, resp_i, \mathrm{s}\}$ for $M^* \notin\ M_i$. To generate a regular LPQS signature, he first calculates the basis point U, V. Then he computes sk and encrypts the seed value. Let $CH_0,\ CH_1$ represent the possible outcome of the challenge $ch = 0, 1$, respectively, with the cardinality of c. If $ch = 0$, then $resp = (\Phi_c(G_A))$ otherwise $resp = (G_C, \Phi_A(G_C))$. The verifier will accept the signature if the $resp$ contains the right order. □

We now calculate the success probability of adversary A. The probability of the secret value of the signing oracle $(0 \leq\ \alpha \leq 2^n)$ is guessed successfully is $1/2^n$. The probability adversary A can produce a valid signature by inquiring q_q queries to the signing oracle are $\left(1-\left(q_q/2^k-4q_h-q_s\right)\right)$ where q_h, q_s denotes the total number of queries for a hashing and signing oracle and k is the output length of the hash function h. The $4q_h$ queries are required to calculate sk', vk', key, and $auth$. Another probability that A solves the SSCDH problem is at least $\left(1-q_q/2^{|F_{P^2}|}\right)$. Therefore, the successful simulation of A happens with a probability constraint relation as expressed in Equation (8). This contradicts with the hardness of the SIDH problem (Poblem 3). Thus, there is no adversary A that could forge a signature under an adaptively chosen message attack.

4.2. Theoretical Security Analysis

In this subsection, we present theoretical analysis of the LPQS scheme to prove its resistance against various cyberattacks and it is described as:

(1) Mutual authentication: the client and the service provider share the messages $\{C_B,\ vk,\ t_C.\}$ and $\{\ pairingvalue,\ auth\ \}$, respectively. vk depends on the $j(E_{AC})$ which is a SIDH problem (Problem 3) and it is hard to find the value of $j(E_{AC})$. Furthermore, C_B is also difficult for the adversary to obtain as it contains sk. Similarly, $auth$ cannot be calculated because of the hardness of SIDH. Therefore, our scheme provides mutual authentication.

(2) Anonymity: in the proposed scheme, the client's identity is hidden in the message $\{C_B,\ vk, t_C\}$, where $vk = h(t_C,\ j(E_{AC}),\ ID_C,\ r_B), C_B = Enc_{ID_C}(r_B \oplus sk), sk = H(t_C, j(E_{AC}), ID_C,\ U,\ V)$. To find the value of the client's identity, the adversary has to calculate the $j(E_{AC})$ which is a SIDH problem (Problem 3). Therefore, our scheme is secure to maintain the anonymity of the client.

(3) Nontraceability: suppose the adversary stores the value of $\{C_B,\ vk, t_C\}$ and the $\{\ pairingvalue,\ auth\ \}$ exchange between client and service provider. As α and β are the ephemeral keys and changing in each session separately, even if the adversary guesses the private key it will not be possible to find the auxiliary point $\{\Phi_c(P_A), \Phi_C(Q_A),\ \Phi_A(P_C), \Phi_A(Q_C)\}$ as given in Problem (4),(5).

(4) No verification table: in the proposed scheme, no verification table has been maintained for the mutual authentication between the client and the service provider.

(5) Session key agreement: the client and the service provider both generate the session key, $key = \mathrm{h}\,(sk,\, vk,\, j(E_{AC}))$, where $sk = H(t_C, j(E_{AC}), ID_C,\, U,\, V), vk = h(t_C,\, j(E_{AC}),\, ID_C,\, r_B),\, U = \Phi_A(P_C),\, V = \Phi_A(Q_C)$. For an adversary it is not possible to create a valid login session because of the Problem (4) and (5). So, our scheme could provide the session key agreement.

(6) Perfect forward secrecy: perfect forward secrecy is provided by $j(E_{AC})$ and is explained in Theorem 1.

(7) Attack resistance: we present that our scheme is resistant to impersonation attacks, replay attacks, modification attacks, stolen verifier attacks and the man-in-the-middle attacks.

(a) Impersonation attack: according to Theorem 1, we can claim that any adversary without any secret key cannot generate a generator as described in problem (1), (2) and without the generator no auxiliary point can be calculated as described in problem (4) and (5). So, only a valid client and service provider can create a login message or response $\{C_B,\, vk, t_C\}$, $\{\, pairingvalue,\, auth\}$. Then the client and the service provider can check the validity of each other by checking the $\{\, pairingvalue,\, auth\}$, and $\{C_B,\, vk, t_C\}$ and can find out if any adversary is present in the system.

(b) Replay attack: in the LPQS scheme, the client access the service by generating the message $\{C_B,\, vk, t_C\}$. After receiving the message, the service provider checks the freshness of t_C, before executing the other steps. If in any case adversary generates t_C and captures the packet $\{C_B,\, vk, t_C\}$, the adversary would not be able to calculate the key without knowing the private key of the client i.e., β. Furthermore, an adversary cannot use the same login message in another session as clients and service providers use a different key {α, β} in each session. So, the client and service provider could find the replay attack by checking the $\{pairingvalue, auth\}$ and $\{C_B,\, vk, t_C\}$.

(c) Modification attack: the service provider can detect the modification attack by checking the validity of the signature $\{C_B,\, vk, t_C\}$. Similarly, the clients can check the validity of $\{pairingvalues, auth\}$.

(d) Stolen verifier table attack: no table is maintained in our scheme by the client or the service provider. So, no such attack is possible.

(e) Man-in-middle attack: due to the mutual authentication, no man-in–the middle attack is possible.

(8) Due to the usage of supersingular isogeny curves, we can effectively compress the keys and signature size. The infinite field F_{p^2} elements used to transmit the points $\Phi_A(P_C)$, $\Phi_A(Q_C)$ are rather large compared to the size of the integer coefficients. However, we have used compressed curves which can be represented by one field element. The key basis calculated by the nodes need not be published as a public parameter, as long as all nodes are able to generate the same basis independently by a predefined algorithm. It also supports perfect forward-secrecy, nontraceability and anonymity as detailed in Section 4.2. In summary, to efficiently exploit the resistant capability of postquantum cryptography, we have used a supersingular isogeny curve and an ID-based signature for postquantum cryptography that requires much shorter keys to maintain the same level of protection and provides user friendly access to the security system.

4.3. Computation Cost Analysis

The computation cost of the LPQS scheme is given in detail for the public key, the private key and the signature. In this computation, we have neglected the lightweight operations like XOR and string concatenation, as we know primes p have the form of $2^n.3^m.f\ \pm 1$, such that $2^n \simeq 3^m$. We compute the cost in terms of λ bits for the λ bits of a quantum computer. We assume p has 6λ bits length. All values are calculated for 128-bit security. Our scheme uses Montgomery curves $E : By^2 = x^3 + Ax^2 + x$, where A–coefficient is sufficient for isogeny computation. The isomorphism classes of the Montgomery form have the same Kummer line. So, both can be represented by one field element, requiring 12 λ-bits. We compare LPQS in the terms of the sizes of public and private keys, and signatures with variants of lattice, multivariate and isogeny, and is shown in Table 2.

Table 2. Postquantum signatures scheme comparison in bytes with 128-bit quantum security.

Scheme	Public Key Size	Private Key Size	Signature Size
Lattice-based (6)	11,653	6769	2444
Lattice-based (33)	7168	2048	5120
Multivariate-based (28)	417,408	14,208	48
Multivariate-based (29)	81,800	8900	337
Multivariate-based (30)	136,100	101,300	79
Hash-based (9)	1000	1000	41,000
Isogeny-based (11)	768	48	141,312
LPQS	336	96	9984

(1) Public Keys

In LPQS, public keys contain $\{\Phi_A(P_C),\ \Phi_A(Q_C),\ P_C,\ Q_C,\ right\}$, where $P_C\ and\ Q_C$, are the points on the elliptic curve E of order 3^m calculated by the service provider using XOR and concatenation operations. So, its cost is negligible and *right* needs no operation. Further, torsion basis $(\Phi_A(P_C),\ \Phi_A(Q_C))$ requires three 3 λ-bits coefficients and 12 λ-bits for the curve. Thus, the public key requires 21 λ-bits. For 128-bit quantum, it needs 336 bytes (21 × 128 = 2688 bits). Other postquantum techniques such as lattice-based (6) and multivariate (28) need 11,653 bytes and 417,408 bytes, respectively.

(2) Private keys

Private keys contain the two generators $GT_{A,}$, $G_{Av,}$ as described in the Section 4. The private key $GT_A(GT_A = P_A + [\alpha]Q_A)$ can be represented as a single coefficient α with respect to the basis point $P_A,\ Q_A$ and it requires 3 λ-bits. So, for two generators we need 6 λ-bits and for 128-bit security level we need 96 bytes (6 × 128=768 bits).

(3) Signature

The signature of the client includes $\{C_B,\ vk, t_C\}$, where C_B is an encrypted representation of the random seed value r_B and $(sk = H(t_C, j(E_{AC}), ID_C,\ U,\ V))$. As we discussed in the previous section, the computation cost of $U,\ V$ is 6 λ-bits and the hash function is 3 λ-bits. The J-invariant $(j(E_{AC}))$ requires 6 λ-bits to store the value in the 128-bit computer. Further, vk $(vk = h(t_C,\ j(E_{AC}),\ ID_C,\ r_B))$ takes 3 λ-bits for the hash function. So, the total cost will be 18 λ-bits. The service provider's signature includes the $\{\ pairingvalue,\ auth\}$, where the mapping cost is negligible and auth = $H_{key}(t_{C,}\ sk',\ vk',\ j_{AV})$. The hash function requires 3λ-bits and similarly the sk', vk' need 15 and 3λ-bits, respectively, and Key = $H_{sk}(vk' \oplus\ j(E_{AV}))$ requires 3 λ-bits. Thus, the total signature cost of the client and service provider is 39 λ-bits. Thus, on average, our scheme requires 21λ-bits (336 bytes) for a public key, 6 λ-bits (96 bytes) for private key and 39 λ^2-bits (39 × 128 × 128 = 79,872 bits) which is equal to 9984 bytes for a signature to achieve 128-bit of quantum security. Comparatively, the signature size is larger than the public and private key because for the signature we use two torsion groups $(E_A,\ E_C)$ to increase the hardness of the isogeny problem, but it requires more storage space.

4.4. Experimental Implementation and Discussion

In this section, we evaluate the performance of the ID-based LPQS scheme for secure data transmission in the IoE environment. The C implementation done in (36) is further extended to include the signature scheme introduced in this paper. For the comparison analysis, we compute the energy consumption, computation time, and CPU cycles taken by the key generation, signing, and verification. We use the C language in the Microsoft Visual Studio 2013 platform on Intel(R) Core(TM) i7-8700 CPU @3.20 GHZ with ×64-based processor, running Windows 10 to implement the proposed scheme. Intel Power Gadget 3.7.0 is used to measure the execution time and energy consumption of LPQS. We also used Raspberry Pi-based IoE nodes to measure the performance of the embedded devices. Our scheme uses Montgomery curves $E : By^2 = x^3 + Ax^2 + x$, where the A–coefficient is sufficient for isogeny computation. The comparative analysis is performed with state-of-the-art nonquantum and postquantum techniques.

4.4.1. Nonquantum Schemes

In this subsection, we compare the energy and time of LPQS with predicate nonquantum signature schemes ASMS (20) and TinyTate (21) for 128-bit nonquantum security level. Nonquantum security 128-bit is approximately equal to 85-bit security level. ASMS and TinyTate use the elliptic curve $y^2 = x^3 + x$. We have considered one ID and one byte of data transmission using AES-128. In terms of energy, ASMS and TinyTate take 110 mJ and 440 mJ, respectively, to perform key generation, signature and verification, while LPQS needs 196.85 mJ to perform the same task, which is 123% more efficient than TinyTate. The total time consumption of LPQS is 8.057 ms. ASMS and TinyTate take 2410 ms and 600 ms, respectively, as is shown in Figure 4. So, LPQS is approximately 300 and 74 times faster than ASMS and TinyTate, respectively. The reason for less computation time is the use of the isogeny curve. It takes less time to perform addition, subtraction and multiplication and hence the overall time reduces effectively. It is noted that 128-bit nonquantum security can be achieved at 85-bit quantum security level with a reasonable tradeoff between energy and time.

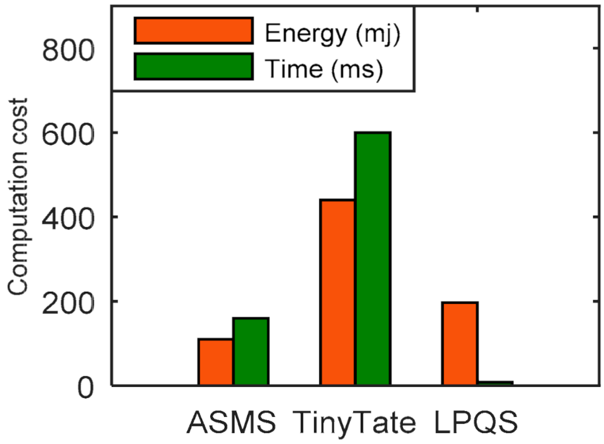


Figure 4. Computation cost of nonqunatum techniques for energy and time consumption.

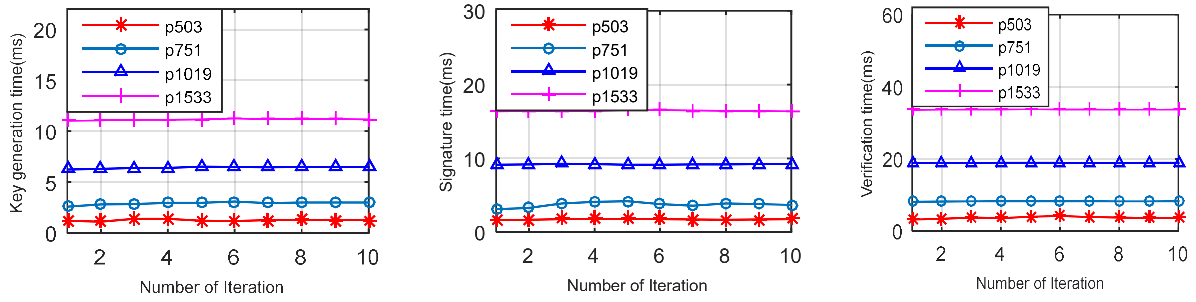
4.4.2. Postquantum Schemes

In this section, we evaluate the performance of the LPQS scheme with state-of-the-art schemes. The performance of the LPQS scheme is evaluated in terms of time for key generation, signature and verification, which are iterated 10 times for prime $p503$, $p751$, $p1019$, and $p1533$. A comparative analysis of the energy with nonisogeny signature schemes SPHINCS (9) and Rainbow (30) are presented. The total number of clock cycles is also analysed and compared with the isogeny based schemes Efficient Algorithms for Supersingular Isogeny (EASI) (16), Microsoft's Supersingular Isogeny Diffie-Hellman (MSIDH) (36), Efficient Post-Quantum Undeniable signature (EPQU) (39), and Key Compression for Isogeny-Based cryptosystems (KCIB) (40). In LPQS, we use supersingular elliptic curves with prime $p = 2^n.3^m.f \pm 1$. For prime $p503$, n is 250, m is 159, f is 1 and it provides 83 bit quantum security, which is approximately equal to 85-bit quantum security, and other prime values are shown in Table 3.

Table 3. Public parameters with comparative nonquantum and quantum security (bits).

$p = 2^n.3^m.f \pm 1$	NonQuantum Security (bit)	Quantum Security (Bit)
$p503 = 2^{250}3^{159} - 1$	125	83
$p751 = 2^{372}3^{239} - 1$	186	124
$p1019 = 2^{508}3^{319}.35 - 1$	253	168
$p1533 = 2^{776}3^{477} - 1$	378	252

The computation time of key generation for different p values is shown in Figure 5a and Table 4. All results are run for 10 iterations. For $p503$, $p751$, $p1019$, and $p1533$ the key generations' average running times are 1.25, 2.96, 6.45 and 11.17 ms, respectively. Further, the average running times of signature generation for $p503$, $p751$, $p1019$, and $p1533$ are 1.75, 3.9, 9.20 and 16.44 ms, respectively. Signature time is more than key generation time because we use two isogeny curves (i.e., Φ_A, Φ_C) and only one isogeny is used for key generation (i.e., Φ_A). In Figure 5c, the computation time of verification is shown and it is clear that average running times for $p503$, $p751$, $p1019$ and $p1533$ are 3.45, 8.17, 18.84 and 33.66 ms, respectively. Verification needs three times more computation time than key generation and two times more computation time than the signature phase. Thus, most of the computation time is spent on verification because the signature size is larger than the public and private keys and in addition, two isogeny operations and one pairing operation are also performed.



(**a**) Key generation time (**b**) Signature time (**c**) Verification time

Figure 5. Various computation time of different phase vs. number of iterations with different p values.

Table 4. Computation time of different phases for different prime values.

Key Generation Time with Number of Iterations										
P	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
$p503$	1.20	1.11	1.39	1.40	1.20	1.18	1.27	1.29	1.27	1.27
$p751$	2.60	2.81	2.85	2.97	2.96	3.10	2.95	2.99	3.02	3.01
$p1019$	6.25	6.30	6.39	6.40	6.53	6.49	6.45	6.49	6.51	6.45
$p1533$	11.02	11.07	11.12	11.13	11.17	11.25	11.20	11.21	11.19	11.17
Signature Time with Number of Iterations										
$p503$	1.65	1.69	1.77	1.79	1.81	1.75	1.73	1.69	1.71	1.76
$p751$	3.10	3.30	3.90	4.10	4.20	3.80	3.60	3.90	3.80	3.70
$p1019$	9.14	9.20	9.25	9.21	9.15	9.13	9.17	9.19	9.22	9.21
$p1533$	16.35	16.39	16.41	16.44	16.51	16.52	16.49	16.44	16.39	16.38
Verification Time with Number of Iterations										
$p503$	3.10	3.30	3.70	3.50	3.90	4.10	3.80	3.70	3.50	3.60
$p751$	8.05	8.11	8.17	8.21	8.23	8.20	8.19	8.16	8.17	8.20
$p1019$	18.76	18.81	18.83	18.86	18.89	18.84	18.81	18.83	18.85	18.86
$p1533$	33.58	33.62	33.65	33.69	33.70	33.68	33.64	33.63	33.63	33.64

In Figure 6, the energy consumption of the LPQS is shown for different message sizes. The message size's impact on the energy consumption and is clear from Figure 6 and Table 5. For a 5 byte message, the maximum and minimum energy consumptions are 848.440 mJ and 8243.409 mJ, respectively. Energy consumption is increasing exponentially with the increase of the message size and security level. Hence, for a security level of 256-bits and a message size of 20 bytes, the energy consumption is 34,733.251 mJ. The total times taken to complete the processes for $p1019$ are 43.82, 49.64, 93.00, 103.00 and 131.21 ms for 1, 2, 5, 10 and 20 bytes of message, respectively. It is clear from Figure 7 and Table 6 that the total time is increasing linearly with increase in the size of the messages.

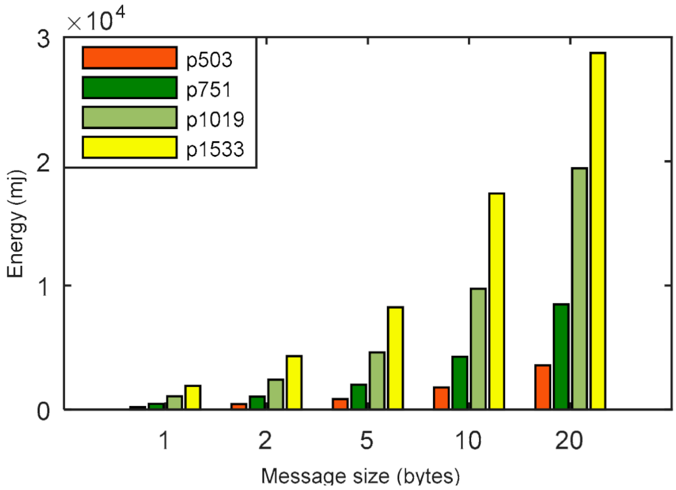


Figure 6. Comparison for energy (in millijoules) with message sizes (in bytes) for various p sizes.

Table 5. Message size vs. energy consumption (mJ) for different p values.

Message Size (Bytes)	1	2	5	10	20
P503	196.854	442.921	848.440	1791.371	3574.868
P751	467.154	1051.096	2013.433	4251.101	8483.516
P1019	1070.640	2408.940	4614.458	9742.824	19,442.822
P1533	1912.624	4303.404	8243.409	17,404.878	34,733.251

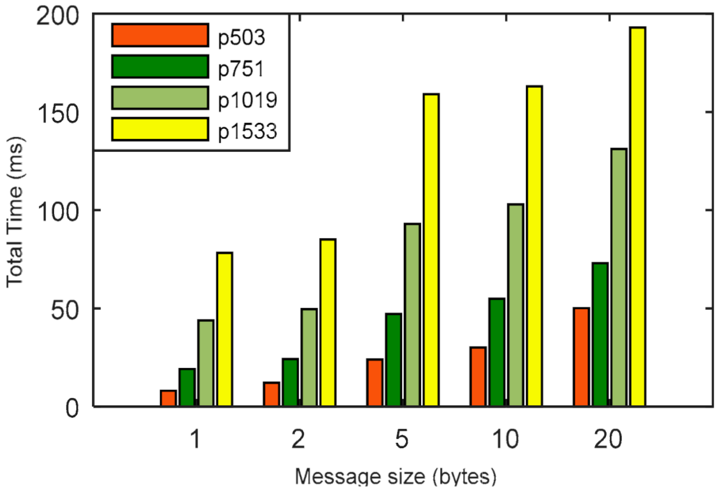
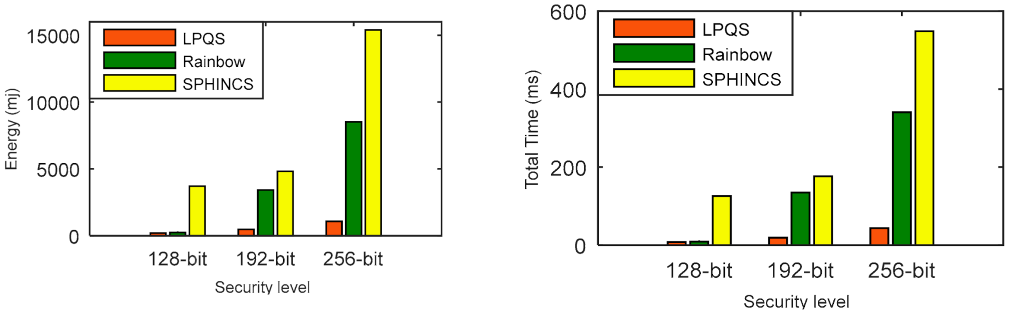


Figure 7. Total time to perform the operations considering different message sizes (in bytes) for various p sizes.

Table 6. Message size vs. time (ms) for different p values.

Message Size (Bytes)	1	2	5	10	20
P503	8.05	12.16	24.0	30.14	50.14
P751	19.12	24.19	47.10	54.90	73.01
P1019	43.82	49.64	93.00	103.00	131.21
P1533	78.29	85.12	159.00	163.00	192.98

We have compared the energy consumption and time computation of LPQS with the nonisogeny signature scheme for 128-bit, 192-bit and 256-bit security levels. In this comparison, we are considering message size as one byte for one ID. For 128-bit security level, Rainbow and SPHINCS need energy of 234.76 mJ and 3706.66 mJ, respectively. LPQS consumes 196.854 mJ, which is approximately 1.1 times and 19 times more efficient than Rainbow and SPHINCS, respectively, and is shown in Figure 8a and Table 7. For 256-bit security level, LPQS needs 1070.64 mJ while Rainbow and SPHINCS take 8518.95 mJ and 15,394.60 mJ, respectively. Further time taken by Rainbow and SPHINCS for 128-bit security are 9.12 ms and 125.9 ms, respectively. For the same security level LPQS needs 8.057 ms, which is approximately 15 times faster than SPHINCS.



(**a**) (**b**)

Figure 8. (**a**) Energy consumption, (**b**) computation time comparison of LPQS with nonisogeny based methods.

Table 7. Comparison of total energy (mJ) with postquantum techniques at different security level.

Security Level	Energy (mJ)			Total Time (ms)		
	128-bit	192-bit	256-bit	128-bit	192-bit	256-bit
Rainbow	234.76	3421.63	8518.95	9.12	134.93	340.86
SPHINCS	3706.66	4812.19	15,394.60	125.90	176.57	548.30
LPQS	196.58	467.15	1070.64	8.057	19.12	43.82

For 256-bit security level, Rainbow and SPHINCS take 340.86 ms and 548.30 ms. However, LPQS needs 43.821 ms for 256-bit security level, as is shown in Figure 8b. These values may be different for different processors. However, LPQS has smaller public and private key sizes (as shown in Table 2), and it consumes less energy and time, and is clear from Figure 8. As shown in Figure 9, EASI takes 754.102 mJ of energy and 7580 million CPU cycles for SIDH key exchange, while EPQU needs energy of 1637.039 mJ and 16,455 million cycles for an undeniable signature. MSIDH and EASI consume 7836 and 3009 million cycles, respectively, for the complete process, while LPQS takes 1976 million cycles and needs 196.854 mJ of energy for the signature, which is the least among the state-of-the-art schemes. The reason for the lower amount of energy and fewer CPU cycles is the usage of two isogeny curves instead of one, which takes the previously computed values for the second verification.

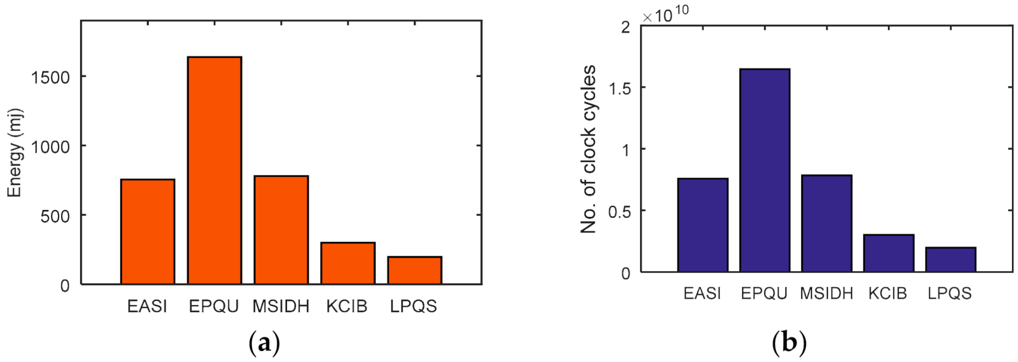


Figure 9. (**a**) Energy consumption; (**b**) clock cycle comparison with isogeny based postquantum schemes.

The energy consumption of the embedded devices implemented in Raspberry Pi for different numbers of nodes is shown in Figure 10a. In this environment, the numbers of clients are increasing from 2 to 10. For two clients the energy consumption is 233.109 mJ and for six clients 497.805 mJ for p503. Further, the energy consumption for p1019 with eight clients is 2612.706 mJ. As we know, the keys are computed once and used for a long period of time. For the signature, the clients need only one pairing and hash operation, which takes less energy for computation. Figure 10b shows the number of clock cycles consumed for a number of nodes ranging from 2 to 10. For p751, the number of clock cycles taken are 1391 and 1640 million cycles for 8 and 10 nodes, respectively. The LPQS consumes fewer CPU cycles because it uses previously computed isogeny values for the next computation.

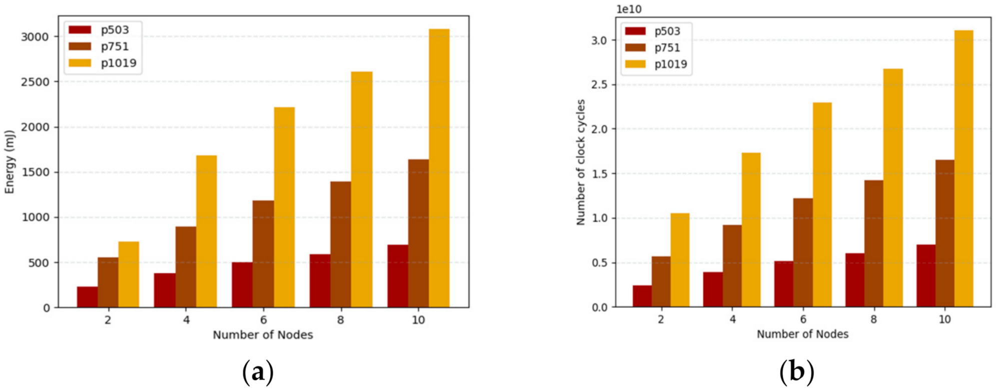


Figure 10. (**a**)Energy consumption, (**b**) number of clock cycles in million cycles with number of nodes.

5. Conclusions and Future Work

In this paper, we presented a lightweight postquantum ID-based signature scheme using the supersingular elliptic curve isogeny for the IoE environments. We use the ID for the calculation of the basis for clients and two isogenies for the verification of service provider and clients. Compressed curves are used to reduce the size of keys and validation of signature depends on the commutative property of curves. In comparison with the nonquantum schemes, LPQS outperforms state-of-the-art techniques in terms of time, CPU cycle and energy. Further, Montgomery curves reduced the public and private keys, and signature sizes. We performed a thorough analysis of postquantum schemes on X86-64 system and Raspberry Pi enabled embedded nodes. The results have clearly shown that the LPQS is feasible for embedded devices. Finally, in comparison with the state-of-the-art techniques, the LPQS scheme is more efficient and secure. In the future, we will extend our scheme to investigate how to represent the elliptic curves efficiently and use the three-party id-based signature scheme based on the supersingular isogeny curve for future networks such data or content focused networking [44] and vehicular communication [45].

Author Contributions: Conceptualization, R.R.; formal analysis, R.R.; investigation, R.R.; methodology, R.R.; supervision, S.K., O.K.; validation, A.M.K., J.L.; writing, R.R.; review and editing, O.K., M.A.A.-K., M.M., A.A.A. All authors have read and agreed to the published version of the manuscript.

Funding: This project was funded by the Deanship of Scientific Research (DSR), King Abdulaziz University. Jeddah. under grant No. (DF-457-156-1441).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The experimental data and associated settings will be made available to researchers and practitioners on individual request to corresponding author, with the restrictions that it will solely be used for further research in literature progress. As the associated research data is being further utilized for development research by the team.

Acknowledgments: This project was funded by the Deanship of Scientific Research (DSR), King Abdulaziz University. Jeddah. under grant No. (DF-457-156-1441). The authors, therefore, gratefully acknowledge DSR for technical and financial support.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Farhan, L.; Kharel, R.; Kaiwartya, O.; Quiroz-Castellanos, M.; Alissa, A.; Abdulsalam, M. A concise review on Internet of Things (IoT)-problems, challenges and opportunities. In Proceedings of the 11th International Symposium on Communication Systems, Networks & Digital Signal Processing (CSNDSP), Budapest, Hungary, 18–20 July 2018; pp. 1–6.
2. Rahman, A.-U.; Afsana, F.; Mahmud, M.; Kaiser, M.S.; Ahmed, M.R.; Kaiwartya, O.; James-Taylor, A. Toward a Heterogeneous Mist, Fog, and Cloud-Based Framework for the Internet of Healthcare Things. *IEEE Internet Things J.* **2019**, *6*, 4049–4062. [CrossRef]
3. Kumar, S.; Singh, K.; Kumar, S.; Kaiwartya, O.; Cao, Y.; Zhou, H. Delimitated anti jammer scheme for Internet of vehicle: Machine learning based security approach. *IEEE Access* **2019**, *7*, 113311–113323. [CrossRef]
4. Verma, G.K.; Singh, B.B.; Kumar, N.; Kaiwartya, O.; Obaidat, M.S. PFCBAS: Pairing Free and Provable Certificate-Based Aggregate Signature Scheme for the e-Healthcare Monitoring System. *IEEE Syst. J.* **2019**, *14*, 1704–1715. [CrossRef]
5. Monz, T.; Nigg, D.; Martinez, E.A.; Brandl, M.F.; Schindler, P.; Rines, R.; Wang, S.X.; Chuang, I.L.; Blatt, R. Realization of a scalable Shor algorithm. *Science* **2016**, *351*, 1068–1070. [CrossRef]
6. Alkim, E.; Bindel, N.; Buchmann, J.; Dagdelen, Ö.; Eaton, E.; Gutoski, G.; Krämer, J.; Pawlega, F. Revisiting TESLA in the Quantum Random Oracle Model. In *Constructive Side-Channel Analysis and Secure Design*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 143–162.
7. Ducas, L.; Kiltz, E.; Lepoint, T.; Lyubashevsky, V.; Schwabe, P.; Seiler, G.; Stehlé, D. CRYSTALS-Dilithium: A Lattice-Based Digital Signature Scheme. *IACR Trans. Cryptogr. Hardw. Embed. Syst.* **2018**, 238–268. [CrossRef]
8. Stehlé, D.; Steinfeld, R. Making NTRU as Secure as Worst-Case Problems over Ideal Lattices. In *Proceedings of the Constructive Side-Channel Analysis and Secure Design*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 27–47.
9. Bernstein, D.J.; Hopwood, D.; Hülsing, A.; Lange, T.; Niederhagen, R.; Papachristodoulou, L.; Schneider, M.; Schwabe, P.; Wilcox-O'Hearn, Z. SPHINCS: Practical stateless hash-based signatures. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, 26–30 April 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 368–397.
10. Hamza, M.; Guenda, K. A New variant of the McEliece cryptosystem based on the Smith form of convolutional codes. *Cryptologia* **2018**, *42*, 227–239.
11. Yoo, Y.; Azarderakhsh, R.; Jalali, A.; Jao, D.; Soukharev, V. A post-quantum digital signature scheme based on supersingular isogenies. In Proceedings of the International Conference on Financial Cryptography and Data Security, Sliema, Malta, 3–7 April 2017; Springer: Cham, Switzerland, 2017; pp. 163–181.
12. Couveignes, J.M. Hard Homogeneous Spaces. Available online: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.61.5396&rep=rep1&type=pdf (accessed on 25 December 2020).
13. Rostovtsev, A.; Stolbunov, A. Public-key cryptosystem based on isogenies. *IACR Cryptol. ePrint Arch.* **2006**, 145.
14. De Feo, L.; Jao, D.; Plût, J. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In *Proceedings of the International Workshop on Post-Quantum Cryptography*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 19–34. [CrossRef]
15. De Feo, L.; Jao, D.; Plût, J. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *J. Math. Cryptol.* **2014**, *8*, 209–247. [CrossRef]

16. Costello, C.; Longa, P.; Naehrig, M. Efficient Algorithms for Supersingular Isogeny Diffie-Hellman. In Proceedings of the Advances in Cryptology | CRYPTO 2016: 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, 14–18 August 2016; Springer: Berlin/Heidelberg, Germany, 2016; pp. 572–601.
17. Galbraith, S.D.; Petit, C.; Silva, J. Identification protocols and signature schemes based on supersingular isogeny problems. In *Proceedings of the International Conference on the Theory and Application of Cryptology and Information Security*; Springer: Berlin/Heidelberg, Germany, 2017; pp. 3–33.
18. Adi, S. Identity-based cryptosystems and signature schemes. In *Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques*; Springer: Berlin/Heidelberg, Germany, 1984.
19. Vélu, J. Isogénies entre courbes elliptiques. In *Comptes Rendus de l'Académie des Sciences de, C.R. Acad. Sci.*; Elsevier of behalf of the French Academy of Sciences (France): Paris, France, 1971; Volume 273, pp. 238–241.
20. Malasri, K.; Wang, L. Addressing Security in Medical Sensor Networks. In *Proceedings of the ACM SIGMOBILE International Workshop on Systems and Networking Support for Healthcare and Assisted Living Environments (HealthNet'07)*; Association for Computing Machinery: New York, NY, USA, 2007; pp. 7–12.
21. Oliveira, L.B.; Aranha, D.; Morais, E.; Daguano, F.; Lopez, J.; Dahab, R. TinyTate. In Proceeding of the Identity-Based Encryption for Sensor Networks, White Plains, NY, USA, 19–23 March 2007.
22. Tan, C.C.; Wang, H.; Zhong, S.; Li, Q. Body Sensor Network Security: An Identity-Based cryptography Approach. In Proceedings of the ACM Conference on Wireless Security, Alexandria, VA, USA, 31 March–2 April 2008; pp. 148–153.
23. Sankaran, S.; Husain, M.I.; Sridhar, R. IDKEYMAN: An identity-based key management scheme for wireless ad hoc body area networks. In Proceedings of the 5th Annual Symposium on Information Assurance (ASIA'09), Buffalo, NY, USA, 3–4 June 2009.
24. Miao, F.; Jiang, L.; Li, Y.; Zhang, Y. AES based biometrics security solution for body area sensor networks. *Bull. Adv. Technol. Res.* **2009**, *3*, 37–41.
25. Ma, C.; Xue, K.; Hong, P. Distributed access control with adaptive privacy preserving property for wireless sensor networks. *Secur. Commun. Netw.* **2014**, *7*, 759–773. [CrossRef]
26. Sun, X.; Tian, H.; Wang, Y. Toward Quantum-Resistant Strong Designated Verifier Signature from Isogenies. *2012 Fourth Int. Conf. Intelligent Netw. Collab. Syst.* **2012**, *5*, 292–296. [CrossRef]
27. Fouque, P.-A.; Hoffstein, J.; Kirchner, P.; Lyubashevsky, V.; Pornin, T.; Prest, T.; Ricosset, T.; Seiler, G.; Whyte, W.; Zhang, Z. Falcon: Fast-Fourier Lattice-Based Compact Signatures over NTRU. Available online: https://www.di.ens.fr/~{}prest/Publications/falcon.pdf (accessed on 10 December 2020).
28. Casanova, A.; Faugere, J.C.; Macario-Rat, G.; Patarin, J.; Perret, L.; Ryckeghem, J. GeMSS: A Great Multivariate Short Signature. Ph.D. Thesis, UPMC-Paris 6. Sorbonne Universités, Paris, France, 2017.
29. Petzoldt, A.; Chen, M.S.; Ding, J.; Yang, B.Y. HMFEv-an efficient multivariate signature scheme. In Proceedings of the International Workshop on Post-Quantum Cryptography, Utrecht, The Netherlands, 26–28 June 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 205–223.
30. Ding, J.; Petzoldt, A. Current State of Multivariate Cryptography. *IEEE Secur. Priv. Mag.* **2017**, *15*, 28–36. [CrossRef]
31. Childs, A.; Jao, D.; Soukharev, V. Constructing elliptic curve isogenies in quantum subexponential time. *J. Math. Cryptol.* **2014**, *8*, 1–29. [CrossRef]
32. Shim, K.A.; Park, C.M.; Koo, N.; Seo, H. A High-Speed Public-Key Signature Scheme for 8-b IoT-Constrained Devices. *IEEE Internet Things J.* **2020**, *7*, 3663–3677. [CrossRef]
33. De Feo, L.; Galbraith, S.D. SeaSign: Compact isogeny signatures from class group actions. In Proceedings of the Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, 19–23 May 2019; Springer: Cham, Switzerland, 2019; pp. 759–789.
34. Parrilla, L.; Castillo, E.; López-Ramos, J.A.; Álvarez-Bermejo, J.A.; García, A.; Morales, D.P. Unified compact ECC-AES co-processor with group-key support for IoT devices in wireless sensor networks. *Sensors* **2018**, *18*, 251. [CrossRef] [PubMed]
35. Hussein, M.S.; Ramos, J.A.L.; Álvarez-Bermejo, J.A. Distributed Key Management to Secure IoT Wireless Sensor Networks in Smart-Agro. *Sensors* **2020**, *20*, 2242. [CrossRef]
36. Microsoft Research. Available online: https://www.microsoft.com/en-us/research/project/sidh-library/ (accessed on 15 December 2020).
37. Li, F.; Zheng, Z.; Jin, C. Secure and efficient data transmission in the Internet of Things. *Telecommun. Syst.* **2015**, *62*, 111–122. [CrossRef]
38. Lee, W.; Kim, Y.S.; No, J.S. A New Signature Scheme Based on Punctured Reed–Muller Code with Random Insertion. *arXiv* **2017**, arXiv:1711.00159.
39. Jalali, A.; Azarderakhsh, R.; Mozaffari-Kermani, M. Efficient post-quantum undeniable signature on 64-bit ARM. In Proceedings of the International Conference on Selected Areas in Cryptography, Ottawa, ON, Canada, 16–18 August 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 281–298.
40. Azarderakhsh, R.; Jao, D.; Kalach, K.; Koziel, B.; Leonardi, C. Key compression for isogeny-based cryptosystems. In Proceedings of the 3rd ACM International Workshop on ASIA Public-Key Cryptography, Xi'an, China, 30 May–3 June 2016; pp. 1–10.
41. Banerjee, U.; Pathak, A.; Chandrakasan, A.P. 2.3 An Energy-Efficient Configurable Lattice Cryptography Processor for the Quantum-Secure Internet of Things. In Proceedings of the 2019 IEEE International Solid- State Circuits Conference—(ISSCC), San Francisco, CA, USA, 17–21 February 2019; pp. 46–48. [CrossRef]

42. Ebrahimi, S.; Bayat-Sarmadi, S.; Mosanaei-Boorani, H. Post-Quantum Cryptoprocessors Optimized for Edge and Resource-Constrained Devices in IoT. *IEEE Internet Things J.* **2019**, *6*, 5500–5507. [CrossRef]
43. John, T. Endomorphisms of abelian varieties over finite fields. *Invent. Math.* **1966**, *2*, 134–144.
44. Prasad, M.; Liu, Y.-T.; Li, D.-L.; Lin, C.-T.; Shah, R.R.; Kaiwartya, O.P. A New Mechanism for Data Visualization with Tsk-Type Preprocessed Collaborative Fuzzy Rule Based System. *J. Artif. Intell. Soft Comput. Res.* **2016**, *7*, 33–46. [CrossRef]
45. Kaiwartya, O.; Kumar, S. Geocasting in vehicular adhoc networks using particle swarm optimization. In Proceedings of the International Conference on Information Systems and Design of Communication, Lisbon, Portugal, 16 May 2014; pp. 62–66.