frontiers | Frontiers in Environmental Science

TYPE Original Research
PUBLISHED 23 September 2022
DOI 10.3389/fenvs.2022.996296

OPEN ACCESS

EDITED BY
Muhammad Tayyab Sohail,
Xiangtan University, China

REVIEWED BY
Aamir Akbar,
Abdul Wali Khan University Mardan, Pakistan
Tengyue Hao,
University of Malaya, Malaysia

*CORRESPONDENCE
Shamsa Kanwal,
shams_kanwal@hotmail.com

SPECIALTY SECTION
This article was submitted to Environmental Informatics and Remote Sensing, a section of the journal Frontiers in Environmental Science.

RECEIVED 17 July 2022
ACCEPTED 31 August 2022
PUBLISHED 23 September 2022

CITATION
Kanwal S, Inam S, Ali R, Cheikhrouhou O and Koubaa A (2022), Lightweight noncommutative key exchange protocol for IoT environments. *Front. Environ. Sci.* 10:996296. doi: 10.3389/fenvs.2022.996296

Lightweight noncommutative key exchange protocol for IoT environments

Shamsa Kanwal[1]*, Saba Inam[1], Rashid Ali[2], Omar Cheikhrouhou[3,4] and Anis Koubaa[4]

[1]Department of Mathematical Sciences, Faculty of Science and Technology, Fatima Jinnah Women University, Rawalpindi, Pakistan, [2]Capital University of Science and Technology, Islamabad, Pakistan, [3]CES Laboratory, National School of Engineers of Sfax, University of Sfax, Sfax, Tunisia, [4]Computer Science Department, Prince Sultan University, Riyadh, Saudi Arabia

Network communications are expanding rapidly in many fields, including telecommunications, the Internet of Things, space, consumer electronics, and the military, with different privacy and security issues at stake in each of these areas. The Internet of Things (IoT) has drawn increased attention from academic and industrial researchers over the last few decades. In this environment, keys are routinely exchanged through a public protocol to support the highly secure IoT domain and thwart security threats from unauthorized parties. The environment for IoT devices is subject to numerous limitations, including those related to processing, memory, and energy. These devices need to pass through a gateway or sink to connect to the network. Additionally, the environment must enable secure communication between gateways and IoT devices, even when the devices are disconnected from the rest of the network. In this paper, a lightweight key exchange protocol for IoT environments is presented, allowing the gateway and the IoT device to communicate over an open channel. Our proposed protocol improves security by utilizing noncommutative structures and polynomials over noncommutative rings. The underlying idea is to use the generalized decomposition problem associated with noncommutative rings. Furthermore, how the suggested protocol can achieve key certification and perfect onward secrecy is addressed. Results show this protocol is a strong candidate for key sharing and secure communication between IoT devices. We put our methodology into practice and the results of our experiments demonstrate enhancement of security levels. Finally, the performance analysis of the suggested protocol is compared with some other protocols, in terms of security, communication, and computing overhead.

KEYWORDS

discrete log problem, public key exchange protocol, public key cryptography, noncommutative ring, polynomial, Internet of Things (IoT)

1 Introduction

The Internet-of-Things (IoT) is an environment that enables interconnected devices and human beings to communicate and send one another information. The use of the IoT environment is growing and is increasingly prevalent in our lives. Many applications depend on functionalities that use information collected from IoT devices: monitoring patient health records, for example. Sometimes it is necessary to send large amounts of data over open wireless channels, such as heavy videos or large image files. In all these situations, the communication of data must be secured and authenticated.

In the IoT environment, the gateway/sink is the main object through which the rest of the environment's devices communicate. We thus require secure and authenticated communication between the IoT device and the gateway/sink. However, traditional key exchange protocols cannot be employed for this purpose due to several constraints. These constraints involve dependence on a trusted third party (TTP) and high processing requirements. It is also essential for IoT environments to be capable of operating even in disconnected mode, without access to a TTP.

The authentication and key exchange processes between two entities without a TTP requires a prior shared secret. Additionally, it is important to eradicate the chance of disclosure of that secret in the environment. We therefore look for more than one secret key, each of which is used for a different purpose. The most important requirement for the implementation of a protocol is to have Perfect Forward Secrecy (PFS). PFS is a feature of a key exchange protocol that ensures the secrecy of all previous session communications in the event of any leakage of a long-term private key. This situation can be controlled by using a different key for every session. If the cryptanalyst can somehow extract the session key, that key does not contain any information about further sessions. This is one of the motivations for our proposed key exchange protocol for the IoT environment. The digital certification of IoT devices depicting the authentication is another salient feature of the IoT environment, which our proposal also addresses.

There is a vast literature suggesting various new techniques, as well as case studies of new technologies and solutions (Ko et al., 2000; Sakalauskas and Burba, 2003; Cheikhrouhou et al., 2020; Zhongjun et al., 2022). In 2017, the National Institute of Standards and Technology (NIST) started an evaluation procedure of cryptographic techniques that can resist quantum attacks. Most of today's cryptosystems currently relying on integer factorization (Rivest et al., 1978) and discrete logarithms (ElGamal, 1985) will become obsolete because of the Shor algorithm (Shor, 1997). Given the quantum threats, there is an increasing trend toward developing new technologies known as quantum key distribution (QKD) (Bennett and Brassard, 1984; Center, 2021; Lizama-P´erez et al., 2021). Seven algorithms have been selected by NIST: four are public key cryptosystems and concern key establishment, and three are related to digital signature algorithms. That is why the active area of research is now noncommutative algebraic cryptography (Anshel et al., 1999; Ko et al., 2000; Paeng et al., 2001; Sakalauskas and Burba, 2003; Inam and Ali, 2016; Kanwal and Ali, 2016). The main focus of this area is to develop and analyze cryptographic protocols over noncommutative structures.

The use of noncommutative structures for public key exchange has been examined by several authors. Here we provide a brief overview of these protocols. In their proposals, Anshel et al. (1999) and Ko et al. (2000) suggested using braid groups as the underlying structure for achieving a good level of security. Thomas and Lal (2008) then proposed a public key cryptographic protocol whose security depends on the discrete log problem (DLP) of the inner automorphism. The main strength of this protocol is the difficulty of finding the conjugate element in a noncommutative group.

The use of a public key cryptographic model was highlighted by Shpilrain and Ushakov (2006), who introduced the difficulty of solving the symmetrical decomposition problem. Thomas and Lal (2008) then proposed a cryptosystem based on the symmetric decomposition problem and conjugacy search problems over a noncommutative structure. For their part, Anjaneyulu and Sanyasirao (2014) generated a common key or group key using the polynomial symmetric decomposition problem. Their proposal was based on the polynomial symmetric decomposition problem over noncommutative division semi-rings.

Furthermore, Meshram et al. (2017) proposed a new IND-CCA2 secure public-key cryptographic protocol. They used the integral coefficient ring polynomial concept with the Suzuki 2-group as the underlying work structure. Odoni et al. (1984) previously discussed the DLP for the ring of matrices. Similarly, the Diffie-Hellman protocol for different matrix rings was presented in Stickel (2005) and Alvarez et al. (2009). In 2004, Stickel (2005) proposed a public key exchange scheme using matrices in a particular subgroup. However, Sramka (2022) highlighted some weaknesses in the scheme, and Shpilrain (2008) provided a cryptanalysis of the scheme, suggesting that it would be more secure to work with a semigroup of all matrices over some finite ring as a platform for the scheme. He also provided a modified method for exchanging a shared secret key. However, Mullan (2012) successfully mounted a linear algebra attack against Shpilrain's modifications of Stickel's scheme.

A broad literature is available concerning key exchange protocols for IoT and their weaknesses (Mano et al., 2016; Khan and Salah, 2018; Mutlag et al., 2019; Lizama-Perez and López, 2021). The first public key exchange was proposed by Diffie and Hellman (1976). After that, there followed an extensive

list of public key exchange protocols, which can be found in the literature: for example, Menezes et al. (1996); Schneier (1996); Singh et al. (2017); and the references therein. Abdalla et al. (2005) proposed a three-party password-authenticated key exchange (3PAKE) protocol for wireless mobile communications. Lu and Cao (2007) modified the 3PAKE protocol, and Chung and Ku (2008) consequently indicated that these protocols were vulnerable to attack by impersonation.

Further, Guo et al. (2008) proved that Anjaneyulu and Sanyasirao (2014)'s improvement of the protocol of Chang et al. (2011) had no security against the replay attack. Chang et al. (2011) and Yoon and Yoo (2011) developed a protocol independent of a symmetrical type of algorithm. In this present study, an improved public key exchange protocol is proposed over the noncommutative ring for IoT devices. The main idea of our proposal is to take polynomials over a given noncommutative ring as useful for secure communication in the pre-and post-quantum age. It is also shown that the brute force attack leads to the solution of the generalized decomposition problem, rendering it infeasible. The rest of this manuscript is organized as follows:

Section 2 provides a brief overview of the security challenges and requirements of the IoT environment. Section 3 gives the basic definitions of related cryptographic problems over noncommutative structures. The proposed protocol is presented in Section 4. We also discuss security aspects of the proposed protocols. The guaranteed secrecy of the new session keys achieved by the perfect forward security method is described in Section 5. Section 6 describes a procedure to certify the public keys across inter-domain certificates. The experimental results and discussion of computational cost are covered in Section 7. Finally, the conclusion of the work is drawn in Section 8.

FIGURE 1
A simple hierarchical model for an IoT environment.

2 Security challenges of IoT environments

It is a highly challenging task to achieve proper levels of security in an IoT environment. These environments are vulnerable to potential attacks, such as user privacy and data integrity attacks. The physical failure of IoT devices and malicious invasions are other potential issues involved. The interconnected devices are usually resource-limited, with inadequate storage capability and energy, which is why IoT environments are sensitive to various threats. Thus, critical IoT data may be blocked and changed, with unrecoverable financial and security consequences. To protect the IoT against attacks, while keeping in mind the memory size and computation power of devices (Alohali and Vassialkis, 2015) in IoT environments, advanced protocols and algorithms need to be evolved. For robust security of the IoT environment, data integrity is also necessary, because a large amount of data has to be processed and managed, and therefore the security of data exposure is essential.

2.1 Security risks and secure design requirements for the IoT environment

The environment under consideration consists of a collection of wireless nodes (devices) having sensing elements. These devices, known as things, are structured into groups. There is a node called a gateway in each group responsible for connecting to the rest of the network. The gateway (GW) may connect with other gateways in the network, and all gateways may be connected to the main server. The data generated by different things are deposited on the main server. There are two main related risks. First, there is the risk of someone eavesdropping on the traffic of the data and of traffic analysis, which may result in the disruption of the whole network. To overcome this risk, secure communication is needed between the nodes. Second, there is always the risk of the physical destruction or imprisonment

of nodes. Given these risks, the following security requirements are essential:

- The things should be authenticated when they request to join the network.
- The gateways should forward data only from authenticated things.
- There should be complete privacy of communication between the things and the server.
- If possible, confidential information should not be put on any one individual thing.

Figure 1 shows a simple example of an IoT environment. Note that a thing may connect or leave the network at any time. Our proposal is based on the needs described above. The main aim is to guarantee the authentication of a thing without significant effect on the network and the provided facilities.

3 Background definitions

This section describes different problems involved in the security of noncommutative group-based cryptography. For instance, Diffie and Hellman (1976) and the Birman et al. (1998) used the conjugator search problem in braid groups to develop their approach. A new Diffie-Hellman-like protocol and ElGamal-like cryptosystem were proposed in Cao et al. (2007). These proposals are based on the symmetric decomposition and generalized symmetric decomposition problems over noncommutative groups. The details of these problems are given as follows.

3.1 Definition 1: Conjugator search problem (CSP)

Let G be a noncommutative group. Given two elements $g, h \in G$, the problem of finding an element $k \in G$, where

$$h = k^{-1}gk$$

is known as the conjugator search problem.

3.2 Definition 2: Decomposition problem (DP)

Let G be a noncommutative group and S be the subset of G. Given two elements $g, h \in G$, the problem of finding two elements $k_1, k_2 \in S$, where

$$h = k_1 g k_2$$

is known as the decomposition problem (Cao et al., 2007).

Generally, for a noncommutative group, the two problems CSP and DP are considered difficult enough given the cryptographic assumptions. More specifically, the DP is intractable, meaning that no probabilistic polynomial-time algorithm is used to solve the DP with nonnegligible accuracy.

3.3 Definition 3: Symmetric decomposition problem (SDP)

Let G be a noncommutative group and $m, n \in Z$. Given two elements $g, h \in G$, the problem of finding the element $k \in G$, where (Cao et al., 2007)

$$h = k^m g k^n$$

is known as the symmetric decomposition problem.

3.4 Definition 4: Generalized symmetric decomposition problem (GSDP)

Let G be a noncommutative group, a subset S of G and $m, n \in Z$. Given two elements $g, h \in G$, the problem of finding the element $k \in S$, where (Cao et al., 2007)

$$h = k^m g k^n$$

is known as the generalized symmetric decomposition problem.

Given these problems, we now define the following cryptographic problem over a noncommutative group G.

3.5 Definition 5: Generalized decomposition problem (GDP)

Let G be a noncommutative group, two subsets S_1 and S_2 of G and $m, n \in Z$. Given two elements $g, h \in G$, the problem of finding two elements $k_1 \in S_1$ and $k_2 \in S_2$, where

$$h = k_1^m g k_2^n$$

is called the generalized decomposition problem.

Note that the GDP can be considered a special form of a constrained DP. If the size of sets S_1 and S_2 is taken to be sufficiently large, and assuming that extracting k_1 and k_2 from $k_1^m g k_2^n$ is impossible from the membership information of sets S_1 and S_2, then it is believed that the GDP is at least as hard as the DP. It follows that the GD assumption states that the GDP is intractable, which means there is no probabilistic polynomial-time algorithm that can solve the GDP with nonnegligible accuracy.

We now give a variant of the GDP over a noncommutative ring R and name it as the polynomial generalized decomposition problem (PGDP).

3.6 Definition 6: Polynomial generalized decomposition problem (PGDP)

Let R be a noncommutative ring, $Z(R)$ be the center of R and $Z(R)[X]$ be the polynomial ring over $Z(R)$. For any random elements $a_1, a_2 \in R$, consider the sets $S_{a_1} \in R$ and $S_{a_2} \in R$ defined as

$$S_{a_1} = \{P(a_1): P(X) \in Z(R)[X]\},$$
$$S_{a_2} = \{P(a_2): P(X) \in Z(R)[X]\}.$$

Let $m, n \in Z$. Given two elements $g, h \in R$, the problem of finding two elements $k_1 \in S_1$ and $k_2 \in S_2$, where

$$h = k_1^m g k_2^n$$

is known as the polynomial generalized decomposition problem.

So, the PGD (polynomial generalized decomposition) cryptographic assumption states that the PGDP over R is intractable, which means there is no probabilistic polynomial-time algorithm that can solve the PGDP with nonnegligible accuracy.

We are going to use the PGDP in our proposed key exchange protocols as described in the following section.

4 Proposed protocol for generation and distribution of keys

In order to increase security, we present a protocol that offers a novel authentication mechanism. It is more efficient and cost effective. The performance analysis of the suggested work is validated and compared with the current protocols in terms of security, communication, and computing overhead.

We now demonstrate the main structure of the proposed protocol for a group g of IoT devices under a gateway GW_g. The nodes/devices are represented by $N_{i,g}$, $i = 1, 2, \cdots, n$. The gateway GW_g manages key generation and distribution for the nodes by performing the following steps.

It selects a noncommutative ring R. Let $Z(R)$ be the center of R and $Z(R)[X]$ be the polynomial ring over $Z(R)$. The elements $c \in R \backslash Z(R)$ and $a_1, a_2 \in R$ are the global parameters. For the ith node/device, the gateway executes the following steps:

- A random polynomial $P_{i,g}(X) \in Z(R)[X]$ such that $P_{i,g}(a_1) \neq 0$, $P_{i,g}(a_2) \neq 0$.
- Small numbers (for instance, less than 10 (Cao et al., 2007)) $r_{i,g}, s_{i,g} \in N$ are chosen.
- The gateway generates the key for each device as follows:

$$KN_{i,g} = \left(P_{i,g}(a_1)\right)^{r_{i,g}} c\left(P_{i,g}(a_2)\right)^{s_{i,g}} \quad (1)$$

.

- The $KN_{i,g}$ is sent to the ith device $N_{i,g}$.

The following steps would be executed for sharing a secret key between ith and jth node:

•ith node computes the shared secret key as follows:

$$W_i = \left(P_{i,g}(a_1)\right)^{r_{i,g}} KN_{j,g}\left(P_{i,g}(a_2)\right)^{s_{i,g}} = K_s. \quad (2)$$

- jth node finds the shared secret key as follows:

$$W_j = \left(P_{j,g}(a_1)\right)^{r_{j,g}} KN_{i,g}\left(P_{j,g}(a_2)\right)^{s_{j,g}} = K_s. \quad (3)$$

The correctness of the proposed protocol is shown in Theorem 1.

Theorem 1: Keeping in mind the specified notation, it follows that the shared secret keys obtained by both entities are the same, that is $W_i = W_j$.

Proof

First, consider the expression

$$W_i = \left(P_{i,g}(a_1)\right)^{r_{i,g}} KN_{j,g}\left(P_{i,g}(a_2)\right)^{s_{i,g}}$$

that becomes by using (1),

$$W_i = \left(P_{i,g}(a_1)\right)^{r_{i,g}} \left(P_{j,g}(a_1)\right)^{r_{j,g}} c\left(P_{j,g}(a_2)\right)^{s_{j,g}} \left(P_{i,g}(a_2)\right)^{s_{i,g}}. \quad (4)$$

Expression (3) gives

$$W_j = \left(P_{j,g}(a_1)\right)^{r_{j,g}} KN_{i,g}\left(P_{j,g}(a_2)\right)^{s_{j,g}}$$
$$= \left(P_{j,g}(a_1)\right)^{r_{j,g}} \left(P_{i,g}(a_1)\right)^{r_{i,g}} c\left(P_{i,g}(a_2)\right)^{s_{i,g}} \left(P_{i,g}(a_2)\right)^{s_{i,g}}. \quad (5)$$

Since the coefficients of the polynomials are from the center $Z(R)[X]$ of the ring R, they commute with every element g of the ring. That is why, for any two polynomials $P(X), Q(X) \in Z(R)[X]$ and $\forall g \in R; \forall l, m \in N$, the following holds:

$$P(g)Q(g) = Q(g)P(g).$$

Using this property successively, we have

$$(P(g))^l (Q(g))^m = (Q(g))^m (P(g))^l, \quad (6)$$
$$\forall g \in R; \forall l, m \in N; \forall P(X), Q(X) \in Z(R)[X].$$

Given property (6), expressions (5) and (6) are the same.

It is obvious that for the proposed public key exchange protocol, the passive attack can be resisted with the PGD assumption over the noncommutative ring.

4.1 Device authentication

After getting the key by the gateway, the $i-th$ device $N_{i,g}$ announces its public key by the following:

- A random polynomial $P_i(X) \in Z(R)[X]$, such that $P_i(a_1) \neq 0$, $P_i(a_2) \neq 0$.

TABLE 1 The size of the set of polynomials of different degrees α and prime p with the order of matrices $n = 2$.

	α (degree of the polynomial)	p (prime number)					
		2	3	5	7	11	13
Proposed Protocol	2	4	18	100	294	1210	2028
Climent et al., 2012		12	27	75	147	363	507
Proposed Protocol	3	8	500	2058	13310	26364	78608
Climent et al., 2012		16	36	100	196	484	676
Proposed Protocol	13	8192	3188646	4882812500	581334062442	345227121439310	3634501279107040
Climent et al., 2012		56	126	350	686	1694	2366
Proposed Protocol	20	1048576	6973568802	381469726562500	478753597785672000	6727499949325600000000	2280595652985700000000000
Climent et al., 2012		84	189	525	1029	2541	3549

- Small numbers (for instance, less than 10 [9]) $r_i, s_i \in N$ are chosen.
- The device generates the key $PN_{i,g}$ as follows:

$$PN_{i,g} = (P_i(a_1))^{r_i} KN_{i,g} (P_i(a_2))^{s_i} \tag{7}$$

If jth, the IoT device whose public key is

$$PN_{j,g} = \left(P_j(a_1)\right)^{r_j} KN_{j,g} \left(P_j(a_2)\right)^{s_j} \tag{8}$$

wants to communicate with the ith device, and before communication, $N_{i,g}$ wants to authenticate the device $N_{j,g}(j \neq i)$ of the same group, the device $N_{i,g}$ will have to validate the $N_{j,g}$ device, which is done by executing the following steps:

- The sender device $N_{j,g}$ sends

$$MN_{j,g} = \left(P_j(a_1)\right)^{r_j} PN_{i,g} \left(P_j(a_2)\right)^{s_j} \tag{9}$$

to the device $N_{i,g}$ for validation, where $P_j(a_1) \neq 0$, $P_j(a_2) \neq 0$, and $r_j, s_j \in N$.

- The device $N_{i,g}$ computes the following:

$$VN_{i,g} = (P_i(a_1))^{r_i} PN_{j,g} (P_i(a_2))^{s_i} \tag{10}$$

where $PN_{j,g}$ is the public key of device $N_{j,g}$.

If

$$VN_{i,g} = MN_{j,g}, \tag{11}$$

then device $N_{i,g}$ will validate the device $N_{j,g}$; otherwise, it rejects the request.

4.2 Choice of parameters

Suppose we take the ring of matrices of order 2 over $\mathbb{Z}_p$ where p is any large prime as the noncommutative ring. Care must be taken in the choice of a large value of prime, approximately of the order of 60 decimal digits.

For a brute force attack, one has to check all the polynomials whose coefficients come from the set $Z(M_2(Z_p))$. The cardinality of the set of polynomials having degree α and coefficients from $Z(M_2(Z_p))$ is $(p-1)p^{\alpha}$. The feasibility of a brute force attack can be denied by taking α or prime p which is sufficiently large to be good enough for security. For example, with the choice of $\alpha = 20$ and p of about 60 decimal digits prime, the set of polynomials to be considered is of the order of 10^{660}. Although these parameters are not so high, the space for a brute force search can be made sufficiently large.

The values of $(p-1)p^{\alpha}$ for different values of α and p are shown in Table 1 and compared with the existing literature (Climent et al., 2012). We note that the number of possibilities of our proposal exceeds these drastically. The proposed protocol exhibits some kind of symmetry in the sense that the computation of public keys involves the same polynomial, which is multiplied with element c from both sides. This symmetry can be avoided by introducing two different polynomials for each user.

If the degrees of the two polynomials are α and β, respectively, then the total number of possible polynomials for one user is $(p-1)^2 p^{\alpha+\beta}$. The feasibility of a brute force attack could be avoided by taking α, β, or a prime p that is sufficiently large. For a prime p of about 60 decimal digits (as in the case of the previous example), and $\alpha = 20$ and $\beta = 20$, the number of polynomials that an adversary has to consider is of the order of 10^{1320}.

4.3 Security aspects of the proposed protocols

This section discusses the security analysis of the protocols proposed in Section 3. The security of the protocols depends on the solution of the generalized decomposition problem. For solving such a problem in a noncommutative ring, no polynomial-time algorithm is known. An adversary has to find the solution to the decomposition problems, which are expressed as the following system of equations:

$$M_A M_B = M_B M_A, \tag{12}$$
$$N_A N_B = N_B N_A, \tag{13}$$
$$M_A C N_A = K_A, \tag{14}$$
$$M_B C N_B = K_B. \tag{15}$$

The adversary also knows the elements $a_1, a_2 \in R$ and $c \in R \setminus Z(R)$. To break the protocol, the adversary has to find the elements M_A, M_B, N_A, and N_B. For this, the adversary tries to find out two polynomials $H_1(X), H_2(X) \in$ Z(R)[X] and numbers $l_1, l_2, m_1, m_2 \in N$ such that

$$(H_1(a_1))^{l_1} = M_A,$$
$$(H_1(a_2))^{l_2} = N_A,$$
$$(H_2(a_1))^{m_1} = M_B, \text{and}$$
$$(H_2(a_2))^{m_2} = N_B.$$

One can then guarantee conditions (12) and (13). We note that the size of the space of the polynomials over $Z(R)$ is a set of all possible random choices. Also, the adversary has to verify conditions (14) and (15). By ensuring the space of the polynomials over $Z(R)$ is large enough, the brute force attack becomes infeasible. To make brute force infeasible, it is suggested choosing a prime p of the order of 60 decimal digits and polynomials of degree 20. As discussed earlier, the space for brute force attacks can be made large enough with these choices. The order of the matrices n can be chosen so that $2^n - 1$ is a Mersenne prime. The choice of a Mersenne prime $n = 31$ is recommended (Stickel, 2005).

4.3.1 Man-in-the-middle attack

In this attack scenario, the adversary has a man-in-the-middle position. He can breach the security of the key exchange protocol by intercepting the communication between Alice (*i*th device) and Bob (*j*th). The attacker manipulates the public keys of both parties and blocks the transmission of actual messages on either side. The proposed protocol can be immunized against this type of attack in the following way.

The gateway can use hashes and encrypt the private keys $P_{i,g}$ of each device using the admin key, which are then saved into a devices hash table.

When the devices calculate their secret shared keys, the gateway calculates the same. The gateway then hashes the shared key and keeps it in the device hash table. Alice (*i*th device) and Bob (*j*th) then hash their shared keys and ask the gateway for verification. The gateway checks the hashed shared keys with the hash table. If the hash value of the shared key matches, then communication may be allowed. Otherwise, the shared key is assumed to have been intercepted and manipulated by the attacker.

For authentication of the gateway, the device authentication mechanism described in Section 4.1 can also be implemented between an IoT device and gateway. Each IoT device and gateway pair can have a unique pair of keys specifically for authentication. Further, IoT devices and the gateway are authenticated using encryption and a hash of the keys during a session.

4.3.2 Privileged insider attack

To prevent this type of attack, the passwords can be managed at the time of registration of the users in the following way.

Each user chooses a username and password and provides this information to the web browser. The passwords can be encrypted by the public key, and their hashes can be stored in a password management table. The users are authenticated by their usernames and passwords. It is hard to get passwords because they are encrypted by public keys. The protocol's security depends on the solution of Eqs 12–15 describing the generalized decomposition problem in a noncommutative ring. The insider finds it hard to guess a password.

4.3.3 Impersonation attack

It is difficult to detect an impersonation attack. In our case, if the attacker impersonates the authenticated user and launches a login request, it is not easy to extract the user id and password because they are encrypted, as discussed in the previous section. Inverting the hash function and decrypting it without knowing the key that generalizes the decomposition problem of the polynomials over the noncommutative ring is computationally hard. The user-impersonation attack is thus resisted by our proposed protocol.

5 Perfect forward secrecy (PFS)

Based on the already shared secret key K_s, Alice (*i*th device) and Bob (*j*th) may want to have a new secret key K_t. However, if there arises a situation in which K_t is compromised by an adversary, then perfect forward secrecy (PFS) is a property of key exchange protocols that assure the secrecy of previously used keys in such leakage. Figure 2 depicts our PFS key exchange protocols that yield new session secret keys. As the private keys of Alice and Bob remain secret, the adversary may get access to K_t, but K_s will not be found.

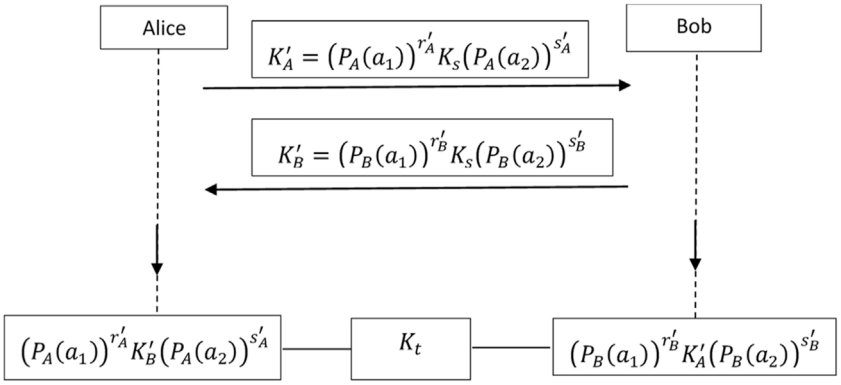

FIGURE 2
Perfect forward secrecy (PFS) in proposed noncommutative key exchange protocol.

Stickel (2005) proposed that the order of matrices $n = 31$ is a secured parameter, and the parameter q (the order of finite field F_q) was not specified. But Shpilrain (2008) remarked in his paper that $q = 2q' \, for \, q' \in [2, 31]$. Shpilrain's attack revealed the shared key without knowledge of any private exponents, whereas Sramka's (2022) attack concentrated on computing the private exponents $l, m, r, and \, s$ of Stickel's scheme.

Shpilrain modified Stickel's scheme to prevent his linear algebra attack, suggesting that the publicly announced elements M, N, W need not be invertible matrices in $M_n(R)$, for some finite ring R. But no specification about ring R was made. Further, he suggested the use of polynomials in the form $\sum c_i X^i, \; for \, c_i \in R$, where R is a finite commutative ring, and then used the fact that all powers Mi of a matrix M commute in the expression of the form $\sum c_i X^i$. This is a generalization of Stickel's scheme, in which Alice and Bob choose polynomials instead of exponents of the public noninvertible matrices M and N.

Mullan (2012) called Shpilrain's modification the polynomial version of Stickel's scheme. He offered a cryptanalysis of Shpilrain's polynomial variant of Stickel's scheme to discover the shared key. Cao et al. (2007) proposed a new scheme for devising a public key cryptosystem based on noncommutative rings. The main idea of their proposal was that for a noncommutative ring, the set of polynomials can be considered the underlying work structure. The Diffie-Helman-like key exchange protocol and consequently ElGamal-like cryptosystems were constructed using polynomials over the noncommutative ring. The authors also showed how to extend their method to noncommutative groups (or semi-groups). The main difference between Shpilrain's polynomial version of the Stickel scheme and the key exchange protocol proposed by Cao et al. (2007) was in the use of the underlying structures. Shpilrain's proposal was for commutative structures, whereas Cao et al. (2007) proposed the use of noncommutative structures.

The protocol proposed by Cao et al. (2007) deals with the polynomials having coefficients from the set of positive integers. In our proposal, polynomials with coefficients from the center of the respective underlying noncommutative structure are used. The advantage of using the coefficients from the center over integer coefficients is that these coefficients depend on the nature of the noncommutative structure used as a platform. That is why determining the values of coefficients is difficult when choosing a complex platform.

A random choice of the polynomial $P(X) \in Z(R)[X]$ and $a_1, a_2 \in R$, such that $P(a_1) \neq 0 \, and \, P(a_2) \neq 0$ is the essential idea. An attacker has no way of determining a polynomial such that $P(a_1)(\neq 0) \in S_{a_1}$ and $P(a_2)(\neq 0) \in S_{a_2}$, even with unlimited computational power. Keeping in mind the huge size of the set of polynomials, as discussed in previous sections, and consequently the huge number of elements of sets S_{a_1} and S_{a_2}, there is an insignificant probability of tracing the private key by an attacker. That is why, even with infinite computing power, the protocol is sound.

The proposed protocol meets the different kinds of requirements of lightweight mechanisms. As it is based on polynomials, it requires fewer bytes for manipulation over the network. Also, the computations involved in key exchange and authentication are fast and space efficient, which makes the protocol suitable for implementation in a lightweight scenario. The protocol can also resist various attacks. Table 2 summarizes the overall strengths of the proposal.

TABLE 2 Comparison of security in some existing protocols.

	Protocols				
Security aspects	Proposed	Poomagal et al., 2020	Wang et al., 2008	Strangio, 2005	Zhangxiang et al.; Hu et al., 2022
Authentication	Yes	Yes	Yes	Yes	No
Perfect Forward Security	Yes	Yes	Yes	Yes	No
Impersonation Attack	Yes	Yes	Yes	No	No
Man-in-Middle Attack	Yes	No	No	No	Yes

TABLE 3 Certification Authority's public database.

User	Public key	Certified key
Certification Authority	$K_{ca} = (P_{ca}(a_1))^{r_{ca}} c(P_{ca}(a_2))^{s_{ca}}$	—
*i*th device	$K_i = (P_i(a_1))^{r_i} c(P_i(a_2))^{s_i}$	$(CK)_i = (P_i(a_1))^{r_i} K_{ca}(P_i(a_2))^{s_i}$
*j*th device	$K_j = (P_j(a_1))^{r_j} c(P_j(a_2))^{s_j}$	$(CK)_j = (P_j(a_1))^{r_j} K_{ca}(P_j(a_2))^{s_j}$

6 Certified keys

A vital characteristic of public keys is authentication by a certification authority (CA). The keys of the proposed key exchange protocol can be certified using the CA's private key (P_{ca}, r_{ca}, s_{ca}), as described in Table 3.

After getting their public certified keys from the web service, *i*th and *j*th device find the shared secret key as $(P_i(a_1))^{r_i}(CK)_j(P_i(a_2))^{s_i}$ and $(P_j(a_1))^{r_j}(CK)_i(P_j(a_2))^{s_j}$, respectively.

7 Experimental results and performance analysis

For the implementation of our protocol, Python 3.6.9 with cryptography library Pycrpto 2.6.1 is used. The Mininet platform (Hu et al., 2022) is used for creating the networking environment. The communication cost of a protocol is affected by different attributes. By analyzing these attributes, the effects of the implementation of the protocol can be studied. The following are some parameters to be studied in this regard:

7.1 Passes overhead

The number of messages exchanged in the execution of the protocol is known as the number of passes. A key exchange protocol with a significant number of passes is considered more costly. The proposed protocol costs three passes in distributing and sharing keys, with the choice of parameters suggested in Section 4.2. Practically, the *i*th and *j*th nodes can generate their shared key simultaneously and need only one communication pass from either node without any further communications.

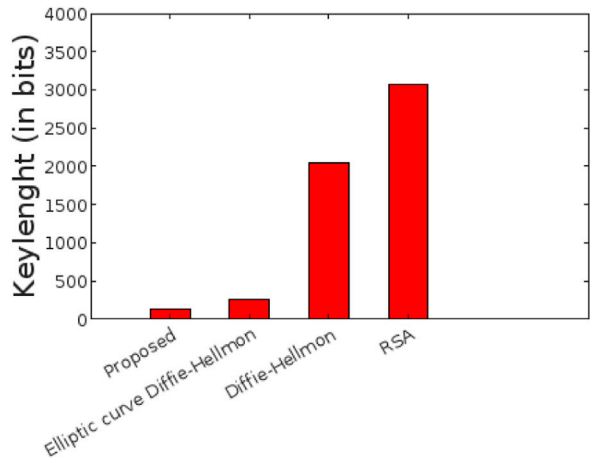

FIGURE 3
Key length comparison.

7.2 Communication cost

The total number of transmitted bits for optimized performance is known as communication overhead or communication cost. A protocol with a low communication overhead is considered more efficient. We take the key length of our key exchange protocol to be

TABLE 4 Comparison of cost.

	Protocols			
Parameters	Proposed	Poomagal et al., 2020	Wang et al., 2008	Strangio, 2005
Total number of operations	$9t_{sm}+10t_e+1t_h$	$3t_{sm}+1t_e$	$3.5t_{sm}+1t_{fi}+2t_h$	$5t_{sm}+2t_h$
computational cost (in seconds)	0.192348	0.20845	0.228967	0.316015

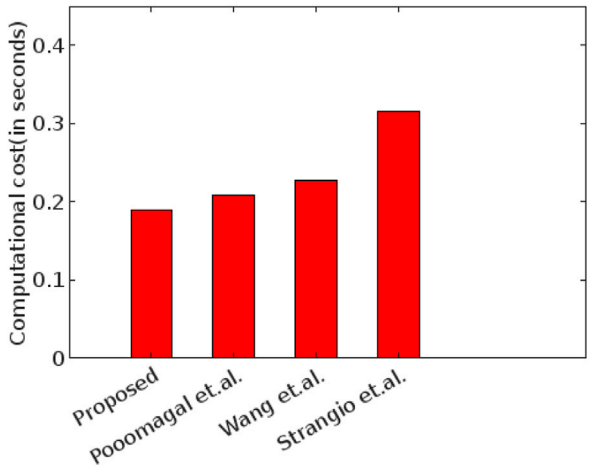

FIGURE 4
Comparison of computational cost.

128-bits, whereas the equivalent key lengths of the state-of-the-art protocols of RSA, Diffie-Hellman, and elliptic curve Diffie-Hellman are 3072-bits, 2048-bits, and 256-bits, respectively. Our proposed protocol performs better than these protocols because it uses polynomials over the noncommutative ring. When compared to the exponential operations used in RSA and Deffie-Hellman, and the elliptic curve operations used in elliptic curve based protocols, operations based on polynomials use less processing space and power. Figure 3 shows the comparison of the key lengths of the protocols.

7.3 Computational time

The total time consumed by the protocol is known as computation time. Some of the attributes of the computational cost are as follows:

For the execution of an algorithm, it is preferred that the total number of arithmetical operations is as low as possible, so as to enhance efficiency and reduce the computational cost. With the increase in the number of operations, the energy and running time of the algorithm may be compromised. The calculation of the computational time of the proposed protocol is based on the ring scalar multiplication, hash operation, and exponentiation operation. The calculated result is then compared with some recent protocols.

With the choice of parameters suggested in Section 4.2, the scalar multiplication involved in our proposed protocol requires 0.003432 s, the hash needs 0.00025 s, and the modular exponentiation operation takes 0.016121 s. The gateway performs six scalar multiplications and eight exponentiations, while three scalar multiplications, two exponentiation operations, and a hash function calculation are involved in the calculation of the key at each IoT device end. Therefore, 0.14956 and 0.042788 s are required for calculations on the gateway and each device, respectively. The total time cost is 0.192348.

The time for simple addition and multiplication operations can be neglected because it is negligible compared to other operations. Table 4; Figure 4 show the comprehensive result of the total computation time for all the operations of our protocols and other protocols. Notations t_e, t_{sm}, t_{fi}, and t_h represent the computation time required for exponentiation, scalar multiplication, field inversion, and hash function, respectively. The protocol in Poomagal et al. (2020), Wang et al. (2008), and Strangio (2005) takes more time than the proposed algorithm. It also performs better than some existing protocols, as depicted in Table 4.

8 Conclusion

A lightweight, efficient, and secure key exchange protocol for secret communication in IoT environments is presented. The related features of key exchange protocol, such as PFS and key certification, are addressed in the proposal. The security aspects of the new protocol are discussed in detail. For the scheme's implementation, the values of related parameters are suggested. It is shown that the proposed protocol enables secure communication between IoT devices in the future regime. Further, an ElGamal-like cryptosystem can also be constructed based on the proposed protocol.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material; further inquiries can be directed to the corresponding author.

Author contributions

SK designed the model and the computational framework and analysed the data. SI and RA assisted with the measurements and wrote the paper with input from all authors. OC and AK contributed to the interpretation of the experimental results. AK provided critical feedback and helped shape the overall structure of the revised manuscript.

Acknowledgments

The authors would like to acknowledge the support of Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Abdalla, M., Fouque, P. A., and Pointcheval, D. (2005). "Password-based authenticated key exchange in the three-party setting," in International Workshop on Public Key Cryptography, 2005 Jan 23 (Berlin, Heidelberg: Springer), 65–84.

Alohali, B. A., and Vassialkis, V. G. (2015). "Secure and energy-efficient multicast routing in smart grids," in 2015 IEEE Tenth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 Apr 7 (Singapore: IEEE), 1–6.

Alvarez, R., Tortosa, L., Vicent, J. F., and Zamora, A. (2009). Analysis and design of a secure key exchange scheme. *Inf. Sci.* 179, 2014–2021. doi:10.1016/j.ins.2009.02.008

Anjaneyulu, G. S. G. N., and Sanyasirao, A. (2014). Distributed group key management protocol over non-commutative division semirings. *Indian J. Sci. Technol.* 7 (6), 871–876. doi:10.17485/ijst/2014/v7i6.18

Anshel, I., Anshel, M., and Goldfeld, D. (1999). An algebraic method for public-key cryptography. *Math. Res. Lett.* 6, 287–291. doi:10.4310/mrl.1999.v6.n3.a3

Bennett, H., and Brassard, G. (1984). "Quantum cryptography: Public key distribution and coin tossing," in Int. Conf. on Computers, Systems and Signal Processing, Bangalore, India, Dec. 1984, 175–179.

Birman, J., Ko, K., and Lee, S. J. (1998). A new approach to the word and conjugacy problems in the braid groups. *Adv. Math. (N. Y).* 139, 322–353. doi:10.1006/aima.1998.1761

Cao, Z., Dong, X., and Wang, L. (2007). *New public key cryptosystems using polynomials over non-commutative rings.* Cryptology e-print Archive.

Center, C. S. R. (2021). *Post-quantum cryptography standardization conference.* Online; Accessed May 17, 2021.

Chang, T. Y., Hwang, M. S., and Yang, W. P. (2011). A communication-efficient three-party password authenticated key exchange protocol. *Inf. Sci.* 181 (1), 217–226. doi:10.1016/j.ins.2010.08.032

Cheikhrouhou, O., Koubâa, A., and Zarrad, A. (2020). A cloud based disaster management system. *J. Sens. Actuator Netw.* 9, 6. doi:10.3390/jsan9010006

Chung, H. R., and Ku, W. C. (2008). Three weaknesses in a simple three-party key exchange protocol. *Inf. Sci.* 178 (1), 220–229. doi:10.1016/j.ins.2007.08.004

Climent, J. J., Navarro, P. R., and Tortosa, L. (2012). Key exchange protocols over noncommutative rings. The case of End ($Z_p \times Z_{p2}$). *Int. J. Comput. Math.* 89 (13-14), 1753–1763. doi:10.1080/00207160.2012.696105

Diffie, W. D., and Hellman, M. E. (1976). New directions in cryptography. *IEEE Trans. Inf. Theory* 22 (6), 644–654. doi:10.1109/tit.1976.1055638

ElGamal, T. (1985). A public key cryptosystem and a signature scheme based on discrete logarithms. *IEEE Trans. Inf. Theory* 31 (4), 469–472. doi:10.1109/tit.1985.1057074

Guo, H., Li, Z., Mu, Y., and Zhang, X. (2008). Cryptanalysis of simple three-party key exchange protocol. *Comput. Secur.* 27 (1-2), 16–21. doi:10.1016/j.cose.2008.03.001

Hu, Z., Li, J., Mergendahl, S., and Wilson, C. (2022). "Toward a resilient key exchange protocol for IoT," in Proceedings of the Twelfth ACM Conference on Data and Application Security and Privacy (CODASPY '22), April 2022, 214–225.

Inam, S., and Ali, R. (2016). A new ElGamal-like cryptosystem based on matrices over groupring. *Neural Comput. Appl.* doi:10.1007/s00521-016-2745-2

Kanwal, S., and Ali, R. (2016). A cryptosystem with noncommutative platform groups. *Neural comput. Appl.* 29, 1273–1278. doi:10.1007/s00521-016-2723-8

Khan, M. A., and Salah, K. (2018). IoT security: Review, blockchain solutions, and open challenges. *Future gener. Comput. Syst.* 82, 395–411. doi:10.1016/j.future.2017.11.022

Ko, K. H. S., Lee, J., Cheon, J. H., Han, J. W., Kang, J. S., and Park, C. (2000). "New public-key cryptosystem using braid groups," in *Advances in cryptology - crypto 2000, 1880. Lecture notes in computer science* (Berlin: Springer-Verlag), 166–183.

Lizama-Perez, L. A., and López, R., J. M. (2021). Non-invertible public key certificates. *Entropy* 23, 226. doi:10.3390/e23020226

Lizama-P´erez, L. A., L´opez, R. J. M., and Samperio, E. H. (2021). Beyond the limits of shannon's information in quantum key distribution. *Entropy* 23, 229. doi:10.3390/e23020229

Lu, R., and Cao, Z. (2007). Simple three-party key exchange protocol. *Comput. Secur.* 26 (1), 94–97. doi:10.1016/j.cose.2006.08.005

Mano, L. Y., Faiçal, B. S., Nakamura, L. H., Gomes, P. H., Libralon, G. L., Meneguete, R. I., et al. (2016). Exploiting IoT technologies for enhancing Health Smart Homes through patient identification and emotion recognition. *Comput. Commun.* 89, 178–190. doi:10.1016/j.comcom.2016.03.010

Menezes, A. J., Van Oorschot, P. C., and Vanstone, S. A. (1996). *Handbook of applied cryptography*. Boca Raton, FL: CRC Press.

Meshram, A., Meshram, C., and Khobragade, N. W. (2017). An IND-CCA2 secure public key cryptographic protocol using Suzuki 2-group. *Indian J. Sci. Technol.* 10 (12), 1–8. doi:10.17485/ijst/2017/v10i12/111588

Mullan, C. (2012). *Some results in group-based cryptography*. Technical report. London: Department of Mathematics, Royal Holloway, University of London.

Mutlag, A. A., Abd Ghani, M. K., Arunkumar, N. A., Mohammed, M. A., and Mohd, O. (2019). Enabling technologies for fog computing in healthcare IoT systems. *Future Gener. Comput. Syst.* 90, 62–78. doi:10.1016/j.future.2018.07.049

Odoni, R. K., Varadharajan, V., and Sanders, P. W. (1984). Public key distribution in matrix rings. *Electron. Lett.* 20, 386–387. doi:10.1049/el:19840267

Paeng, S. H., Ha, K. C., Kim, J. H., Chee, S., and Park, C. (2001). "New public key cryptosystem using finite non abelian groups," in *Advances in cryptology - crypto 2001, 2139. Lecture notes in computer science* (Berlin: Springer-Verlag), 470–485.

Poomagal, C. T., S. Kumar, G. A., and Mehta, D. (2020). Multi level key exchange and encryption protocol for internet of things (iot). *Comput. Syst. Sci. Eng.* 35 (1), 51–63. doi:10.32604/csse.2020.35.051

Rivest, R. L., Shamir, A., and Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM* 21 (2), 120–126. doi:10.1145/359340.359342

Sakalauskas, E., and Burba, T. (2003). Basic semigroup primitive for cryptographic session key exchange protocol. *Inf. Technol. Control* 28 (3), 76–80.

Schneier, B. (1996). *Applied cryptography*. Second edition. New York, NY: John Wiley & Sons.

Shor, P. W. (1997). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* 26 (5), 1484–1509. doi:10.1137/s0097539795293172

Shpilrain, V. (2008). Cryptanalysis of Stickel's key exchange scheme. *Proc. Comput. Sci. Russ.* 5010, 283–288.

Shpilrain, V., and Ushakov, A. (2006). A new key exchange protocol based on the decomposition problem. *Contemp. Math.* 418, 161–167.

Singh, S. R., Khan, A. K., and Singh, T. S. (2017). A new key management scheme for wireless sensor networks using an elliptic curve. *Indian J. Sci. Technol.* 10 (13), 1–7. doi:10.17485/ijst/2017/v10i13/108661

Sramka, M. (2022). On the security of Stickels key exchange scheme. Available at: http://crisesdeim.urv.cat/msramka/pubs/sramka-stickelkesecurity.pdf.

Stickel, E. (2005). "A new method for exchanging secret key," in Proceedings of the Third International Conference on Information Technology and Applications (ICITA'05), Sidney, Australia, 426–430.

Strangio, M. A. (2005). "Efficient Diffie–Hellmann two-party key agreement protocols based on elliptic curves," in Proc. 20th ACM Symposium on Applied Computing (SAC), 324–331.

Thomas, T., and Lal, A. K. (2008). A zero-knowledge undeniable signature scheme in nonabelian group setting. *Int. J. Netw. Secur.* 6 (3), 265–269.

Wang, S., Cao, Z., Strangio, M. A., and Wang, L. (2008). Cryptanalysis and improvement of an elliptic curve diffie-hellman key agreement protocol. *IEEE Commun. Lett.* 12 (2), 149–151. doi:10.1109/lcomm.2008.071307

Yoon, E. J., and Yoo, K. Y. (2011). Cryptanalysis of a simple three-party password-based key exchange protocol. *Int. J. Commun. Syst.* 24 (4), 532–542. doi:10.1002/dac.1168

Zhongjun, T., Shah, S. K., Ahmad, M., and Mustafa, S. (2022). Modeling consumer's switching intentions regarding 5G Technology in China. *Int. J. Innov. Technol. Manag.* 19. doi:10.1142/s0219877022500110