Annales Mathematicae et Informaticae
60 (2024) pp. 75–92
DOI: 10.33039/ami.2024.03.004
URL: https://ami.uni-eszterhazy.hu

A secure key authentication scheme for cryptosystems based on DLP in group ring

Sandeep Kumar, Gaurav Mittal, Sunil Kumar

Defence Research and Development Organization, Near Metcalfe House, New Delhi, 110054, India
sandeepkumar.hqr@gov.in
gaurav.mittaltwins@gmail.com
sunilkumar.hqr@gov.in

Abstract. The public keys in a public key cryptosystem need not to be protected for confidentiality, however, it is important to confirm their legality. In this paper, motivated by Meshram et al. (2017), we develop a simple novel key authentication scheme for public key cryptosystems whose security rely on discrete logarithm problem in group ring. The advantage of our novel scheme is that it requires no authority unlike regular certificate based techniques. In our scheme, we consider a pair of secret key and password as the certificate of public key. We show that the security of our scheme relies on discrete logarithm problem in group ring (DLPGR). The DLPGR is an NP problem for which no known quantum algorithm exists that solves it in polynomial time.

Keywords: Authentication Scheme, Discrete Logarithm Problem, Public-key Cryptosystem, Group Ring, Certificate based scheme

AMS Subject Classification: 94A60, 20C05, 20C07

1. Introduction

The public key cryptography successfully tackled the serious issue of distribution of secret keys in symmetric key cryptography (cf. [4, 41, 42]). Moreover, various advanced digital signature schemes and cryptographic primitives have been created through the aid of public key cryptography (see, for example, [2, 3, 11–14, 17, 18, 28, 29, 31]). Typically, in a public key cryptographic scheme, there are two keys

Submitted: November 5, 2023
Accepted: March 20, 2024
Published online: March 23, 2024

related to an entity, namely a public key which is available in open domain and a private key which is available only with the entity. The public keys are kept open in a repository open to all. But this public availability leads to vulnerability against certain active attacks, such as an adversary can replace the actual public key of an entity with a false public key [7]. Consequently, a secure key authentication scheme is required to verify the legality of public keys. In the available literature, various key authentication schemes have been proposed. But for almost all of these schemes, there is a requirement of atleast one authority known as trusted center (TC) or key authentication center (KAC). Over this authority, the whole trust lies, therefore, it must be strong and safe against any external and internal attacks.

It is worth to mention that in a wide range of key authentication schemes, the hold of KAC over secret keys can be categorized in the following situations: (i) KAC is in total control of secret key of an entity; (ii) KAC can create an undetected false certificate, however, it does not possess the secret key of an entity; (iii) if KAC has produced a false certificate without possessing secret key, then it can be shown that KAC can also produce the false certificate [7]. Accordingly, Girault [7] classified the key authentication schemes into the following levels of trust: (a) schemes depending on identity, i.e., ID-based; (b) schemes depending on certificates; (c) schemes depending on public keys that are self-certified. Moreover, Girault suggested a design of self-certified public keys. The improvement of the model proposed by Girault was discussed in [24] by Laih et al. and their scheme was the combination of ID and certificate based schemes.

For cryptosystems whose security depend on discrete logarithm problem, Horng et al. [15] proposed a key authentication scheme. Their scheme requires no KAC, however, the design is similar to certificate based schemes. Any entity can create a certificate of the public key by combining secret key and password through some known function. The password's hash value is calculated and deposited at the server. Zhan et al. [44] shown that the design of Horng et al. was susceptible to an attack based on guessing the password. In order to protect against the password guessing attack, Lee and Wu [26] proposed another key authentication scheme. Further, in 2003, it is shown by Lee et al. [25] that there is a problem of non-repudiation of public key of an entity in the design of Zhan et al. In addition to this, Lee et al. [25] discussed an upgraded key authentication scheme. However, the scheme of Lee et al. has serious security flaws (see, [37, 43, 45]). Meshram et al. [32] presented another key authentication scheme for cryptosystems whose security depend on the problems such as generalized discrete logarithm and integer factorization. We also refer to the references within Meshram et al. [32] for a nice survey on several other key authentication schemes available in the literature.

Due to the availability of various quantum algorithms (see [2]), the hard problems such as discrete logarithm in a finite field and integer factorization problem are breakable on a sufficiently large quantum computer. Therefore, there is an urgent need to incorporate various other hard problems (possibly NP-hard) in designing new cryptographic primitives, for example, shortest vector problem that arises in lattices, decoding a general linear code etc., (see [2]). In this paper, we utilize the

recently discovered hard problem by Hurley et al. [19] in the algebraic structure of group ring [33]. Precisely, our contribution in this paper is as follows: we incorporate discrete logarithm problem in group ring (DLPGR) to design a novel key authentication scheme for public key cryptosystems whose security relies on the hardness of DLPGR.

To this end, we mention some of the available literature in the direction of group ring based cryptography. Rososhek [38] discussed cryptosystems in automorphism groups of group rings of abelian groups. These cryptosystems implicitly depend on the structure of group ring. Hurley et al. [19] discovered several hard problems in group ring that are important from the perspective of cryptography. Inam et al. [20] designed an ElGamal-like cryptosystem that is based on the matrices over group ring. Goel et al. [8] presented an undeniable signature scheme by utilizing group ring. A key exchange protocol by employing matrices over group ring was proposed by Gupta et al. [9]. Mittal et al. [34–36] constructed few encryption schemes using group ring.

In this paper, our main aim is to present a novel key authentication scheme. This scheme is especially for all the public key cryptosystems whose security relies on solving DLPGR. We show that our scheme works in the absence of any authority and its security relies on deducing the solution of DLPGR. This paper is organized as follows. The Section 2 contains some background material upon which we built our new scheme. Our novel key authentication scheme whose security relies on DLPGR is discussed in Section 3. We discuss the security analysis of our scheme in Section 4. The Section 5 involves a comparison analysis of our scheme with the already available key authentication schemes. In Section 6, we discuss an example to show the practicality of our scheme. Finally, the last section draws some concluding remarks.

2. Preliminaries

2.1. Group ring and units

Definition 2.1. Let R be a ring having unity and let G be a group. Let RG be the set of all R-linear combinations of the form

$$u = \sum_{g \in G} r(g)g, \quad r(g) \in R,$$

where the summation runs over finitely many elements of G. In other words, the set RG contains all finite R-linear combinations of the elements of G. Let $\cdot$ be the operation defined on the group G and

$$u_1 = \sum_{g \in G} r(g)g \quad \text{and} \quad u_2 = \sum_{g \in G} r'(g)g,$$

where $r(g), r'(g) \in R$ and $g \in G$. In order to multiply two elements, we write $u_2 = \sum_{h \in G} r(h)h$ for notational convenience. Then we consider the addition $(+)$

and multiplication $(*)$ operations in RG as follows:

$$u_1 + u_2 = \big(\sum_{g\in G} r(g)g\big) + \big(\sum_{g\in G} r'(g)g\big) = \sum_{g\in G}(r(g) + r'(g))g,$$
$$u_1 * u_2 = \big(\sum_{g\in G} r(g)g\big) * \big(\sum_{g\in G} r'(g)g\big) = \sum_{g,h\in G} r(g)r'(h)(g\cdot h) = \sum_{g'\in G} r''(g')g',$$

where

$$r''(g') = \sum_{g\cdot h=g'} r(g)r(h) = \sum_{g\in G} r(g)r(g'\cdot g^{-1}) = \sum_{h\in G} r(g'\cdot h^{-1})r(h).$$

It is straight-forward to see that both the addition and multiplication operations defined above are well-defined. This is because a ring already has addition and multiplication operations. The set RG along with operations $+$ and $*$ is known as group ring.

Example 2.2. Let $R = \mathbb{Z}_5 = \{0, 1, 2, 3, 4\}$ be a ring of 5 elements and let $G = \mathbb{C}_4 = \{e, a, a^2, a^3\}$ be a cyclic group containing 4 elements. Let $u_1 = 2e + a$ and $u_2 = a + 2a^3$ be the elements of the group ring $\mathbb{Z}_5C_4$. Then we have

$$\begin{aligned} u_1 + u_2 &= (2e + a) + (a + 2a^3) = 2e + 2a + 2a^3, \\ u_1 * u_2 &= (2e + a) * (a + 2a^3) = 2e(a + 2a^3) + a(a + 2a^3) \\ &= 2a + 4a^3 + a^2 + 2a^4 = 2e + 2a + a^2 + 4a^3, \end{aligned}$$

where we have used the fact that $a^4 = e$ for $a \in G$.

Definition 2.3. Units: Let $u_1, u_2 \in RG$ be such that

$$u_1 * u_2 = e = u_2 * u_1.$$

Then the element u_2 is inverse of u_1 (or u_1 is inverse of u_2) and we denote $u_2 = u_1^{-1}$. The elements u_1 and u_2 are known as units of the group ring.

Definition 2.4. Order: The order of an element $u \in RG$ is the smallest positive integer k such that $u^k = e$.

Example 2.5. Let $R = \mathbb{Z}_2 = \{0, 1\}$ be a ring of 2 elements and let G be the quaternion group of 8 elements, i.e.,

$$Q_8 = \langle x, y : x^4 = y^4 = e, x^2 = y^2, yx = x^{-1}y\rangle.$$

Let $w = 1 + x + y$. Using Sharma et al. [39], we know that

$$w^{-1} = (1 + x + y)^3.$$

Thus, $w \in \mathbb{Z}_2Q_8$ is a unit of the group ring.

Definition 2.6. Augmentation map: Let RG be a group ring. Then the following map

$$\mathcal{J}\colon RG \to R \quad \text{defined by} \quad \mathcal{J}\big(\sum_{g\in G} r(g)g\big) = \sum_{g\in G} r(g)$$

is known as the augmentation map. Clearly, $\mathcal{J}$ maps any element $w \in RG$ to the sum of all the coefficients $r(g)$ of the elements g of G appearing in w.

Example 2.7. Let $R = \mathbb{Z}_5$ and $G = Q_8$, where Q_8 is the quaternion group of order 8 (same as discussed in Example 2.5). We take

$$w = \sum_{g\in G} r(g)g = 1 + 2x + y \in RG.$$

Then, we observe that $r(1) = r(y) = 1$, $r(x) = 2$ and rest of the coefficients $r(g)$ are zero for $g \in \{xy, x^2y, x^3y, x^2, x^3\}$. So, we have

$$\mathcal{J}(w) = \mathcal{J}(1 + 2x + y) = r(1) + r(x) + r(y) + 0 = 1 + 2 + 1 = 4.$$

Thus, $\mathcal{J}(w) = 4$ for $w = 1 + 2x + y$.

2.2. Discrete logarithm problem in group ring (DLPGR)

Definition 2.8. Let $u_1, u_2 \in RG$ be given by

$$u_1 = \sum_{g\in G} r(g)g, \quad u_2 = \sum_{g\in G} r'(g)g,$$

where $r(g), r'(g) \in R$. Let k be a positive integer such that $u_1 = u_2^k$, i.e.,

$$\sum_{g\in G} r(g)g = \left(\sum_{g\in G} r'(g)g\right)^k.$$

The DLPGR is the problem of deducing k from the known values of u_1 and u_2.

There are several other versions of DLP in groups, for example, generalized DLP, Elliptic curve DLP (see [31, 32]). DLPGR was discovered by Hurley et al. [19] and used by various researchers to produce secure cryptosystems [34, 36]. To this end, we briefly discuss a public key cryptosystem whose security relies on DLPGR.

2.3. Public key cryptosystem based on DLPGR

Let R be a ring and let G be a finite group. Let u be a unit of the group ring RG with inverse u^{-1}. Both u and u^{-1} are open in public domain. It is worth to mention that unlike groups, computation of inverse of an element in a group ring is also, in general, a hard problem (see [19]). But for some instances, it is easy

(see [35] for a nice overview). Let k be a secret integer and $v = u^k$ be the public key. To encrypt a message M, an ephemeral key r needs to be chosen, where r is a positive integer. The ciphertext is as follows

$$\mathcal{C}_1 = (u^{-1})^r, \quad \mathcal{C}_2 = M * (v)^r.$$

It is worth to mention that the encryption mentioned above is not similar to ElGamal as the ciphertext generated by ElGamal's scheme does not involve the computation of inverse of an element. The decryption is straight-forward by using the private key as follows:

$$M = \mathcal{C}_2 * (\mathcal{C}_1)^k.$$

It is easy to see that the security of above scheme depends on DLPGR. Next, we recall the password authentication scheme for multi-user computing systems.

2.4. Password authentication procedure

A password is a series of characters that is anticipated to distinguish between an entity and the system. In a password authentication scheme for multi-user, entity must (i) register with various systems; (ii) must save purposely created various passwords to attain security of high level. While login to the system with his/her identity (ID), an entity needs to enter his/her password (Pw) to help the system in his/her recognition. The system authorize the entity by verifying the pair (ID, Pw). Basically, the system checks whether or not the pair (ID, Pw) belongs to the list of authorized pairs available with the system. Suppose that the list of authorized pairs available with the system is not encrypted. This situation would be extremely insecure. Since any adversary may get access to the system and can easily forge. Consequently, Evans et al. [6] recommended a cryptographic solution of the same that keeps authorized passwords safe from snooping. Furthermore, it was suggested that passwords can be mapped through some cryptographic one-way function to pictures. Therefore, the list of authorized pairs available with the system can then be a list of mapping results (see [21, 22, 46]).

Based on DLPGR, we select a unit u of the group ring RG of large order. Let Pw be the password of the entity. For this password Pw, we use the capacity u^{Pw} as a picture. The benefit of such pictures is that they can be placed openly in the table of passwords since they can only leak the information about the password if DLPGR is solvable. But DLPGR is a hard problem.

3. The novel key authentication scheme

Let R be a ring and let G be a finite group. For an entity j, let his/her password be represented by Pw_j. Let sk_j be the private key of j and the corresponding public key pk_j be

$$pk_j = u^{sk_j},$$

where u is an element of the group ring RG (it may be taken as a unit of large order). We define the notations to be used in our key authentication scheme in Table 1.

Table 1. Notations incorporated in the novel scheme.

Notations	Descriptions
Pw_j	Password of an entity j
sk_j, pk_j	Private and public keys of entity j
$\oplus$	XOR operation
RG	Group ring
u	Unit of group ring
C_j	Certificate of public key pk_j
$\mathrm{E}(\cdot)$	Exponentiation (one-way) function
$\mathcal{J}$	Augmentation map
$\mathcal{H}$	Public hash function

3.1. Registration phase

Our scheme bank on the following assumptions:

(1) Let the one-way exponential function

$$E\colon \mathbb{Z} \to RG \quad \text{where} \quad z \mapsto u^z,$$

where $R = \mathbb{Z}$ or $\mathbb{F}_p$ for some prime p. The function $E(\cdot)$ is open in public domain. It is important to see that by using repeated square and multiply algorithm [31], one can easily compute $E(z)$ for any given z. This one-way exponential function is nothing but same as the function utilized in the cryptosystems that are based on DLPGR.

(2) In order to guard against the password guessing attacks, for the password Pw_j of an entity j, we apply the one-way exponential function E and obtain the encrypted password as $E(\mathrm{Pw}_j \oplus sk_j)$. This encrypted password is saved in the password table.

(3) For saving the storage space, augmentation map $\mathcal{J}$ (see Definition 2.6) and a public hash function $\mathcal{H}$ can be utilized. For this, first apply $\mathcal{J}$ on the multiplication of $E(\mathrm{Pw}_j \oplus sk_j)$, with $pk_j^{\mathcal{J}(pk_j)}$ and then apply $\mathcal{H}$ on the result, i.e., store the picture

$$\mathcal{H}\big(\mathcal{J}(E(\mathrm{Pw}_j \oplus sk_j)\big(pk_j\big)^{\mathcal{J}(pk_j)}))\big)$$

of password Pw_j. We note that $\mathcal{J}(E(\mathrm{Pw}_j \oplus sk_j)\big(pk_j\big)^{\mathcal{J}(pk_j)}))$ is an integer as $R = \mathbb{Z}$ or $\mathbb{Z}_p$. By writing its binary representation, we can compute the above-mentioned hash value by using any of the recently developed state-of-the-art hash functions

such as hash functions based on Gluškov product of automata [5, 10], quantum hash function [27], hash functions based on chaotic maps [40] etc., or conventional hash function such as SHA-3.

(4) The password table can now be openly placed in public domain, since we have stored the picture of passwords or encrypted passwords.

3.2. Certificate generation phase

Since we are working on the self-certified keys, the certificate is generated itself by the entity. There is no need of KAC or TC. The entity j can pair his/her secret key sk_j along with password Pw_j to obtain the certificate

$$\mathcal{C}_j = \mathrm{Pw}_j \oplus sk_j + sk_j \mathcal{J}(pk_j).$$

For verification purpose, both the public key pk_j and certificate $\mathcal{C}_j$ are placed in the network that is open to public.

3.3. Authentication and verification phase

Each entity needs to present the certificate, public key and encrypted password for authentication. The password image can be written as

$$\begin{aligned} E(\mathcal{C}_j) &= u^{\mathrm{Pw}_j \oplus sk_j + sk_j \mathcal{J}(pk_j)} \\ &= u^{\mathrm{Pw}_j \oplus sk_j} u^{sk_j \mathcal{J}(pk_j)} \\ &= E(\mathrm{Pw}_j \oplus sk_j)\big(pk_j\big)^{\mathcal{J}(pk_j)}. \end{aligned} \tag{3.1}$$

If hash function is used on the encrypted password, then we must have

$$\mathcal{H}(\mathcal{J}(E(\mathcal{C}_j))) = \mathcal{H}\big(\mathcal{J}(E(\mathrm{Pw}_j \oplus sk_j)\big(pk_j\big)^{\mathcal{J}(pk_j)}))\big). \tag{3.2}$$

So, whenever an entity t needs to utilize the public key of an entity j, he/she obtains j's certificate $\mathcal{C}_j$ and public key pk_j from the network. Also, t obtains $E(\mathrm{Pw}_j \oplus sk_j)$ or the hash value $\mathcal{H}\big(\mathcal{J}(E(\mathrm{Pw}_j \oplus sk_j)\big(pk_j\big)^{\mathcal{J}(pk_j)}))\big)$ from the password table. Following this, the entity t can ensure the validity of pk_j through equation (3.2) or the following equation:

$$E(\mathcal{C}_j) = E(\mathrm{Pw}_j \oplus sk_j)\big(pk_j\big)^{\mathcal{J}(pk_j)}.$$

If any of these holds, then the entity t gets assurance about the legality of public key pk_j and can use it for the encryption purpose.

4. Security analysis

In the password table, in place of the entity j's password Pw_j, we have stored $E(\mathrm{Pw}_j \oplus sk_j)$. This means that one cannot alter or modify it illegally. However,

an adversary may attempts to illegally create the public key or deduce the private key or speculate the password in our scheme. We show that this would not be possible in our scheme via the following theorem.

Theorem 4.1. *The novel key authentication scheme of this paper can resist the public key based forgery attack.*

Proof. Suppose that an attacker attempts to forge the public key pk_j of an entity j with a wrong key wk_j. In order to certify wk_j as an actual key, the attacker must produce a false certificate $\mathcal{C}'_j$ such that any of the following holds:

$$\begin{aligned} E(\mathcal{C}'_j) &= E(\mathrm{Pw}_j \oplus sk_j)\big(wk_j\big)^{\mathcal{J}(wk_j)}, \\ \mathcal{H}(\mathcal{J}(E(\mathcal{C}'_j))) &= \mathcal{H}\big(\mathcal{J}(E(\mathrm{Pw}_j \oplus sk_j)\big(wk_j\big)^{\mathcal{J}(wk_j)}))\big). \end{aligned} \tag{4.1}$$

From these equations, the adversary can deduce $\mathcal{C}'_j$ by solving any of the following:

$$\mathcal{C}'_j = E^{-1}\Big(E(\mathrm{Pw}_j \oplus sk_j)\big(wk_j\big)^{\mathcal{J}(wk_j)}\Big), \tag{4.2}$$

$$\mathcal{C}'_j = E^{-1}\Big(\mathcal{J}^{-1}\Big(\mathcal{H}^{-1}\Big(\mathcal{H}\big(\mathcal{J}(E(\mathrm{Pw}_j \oplus sk_j)\big(wk_j\big)^{\mathcal{J}(wk_j)}))\big)\Big)\Big). \tag{4.3}$$

Since the attacker cannot modify $E(\mathrm{Pw}_j \oplus sk_j)$ in the table of passwords without possessing the knowledge of entity j's password, the attacker cannot deduce the certificate $\mathcal{C}'_j$ from equation (4.2) without solving DLPGR. Meanwhile, as the cryptographic hash functions, by definition, are pre-image resistant, the attacker cannot deduce the certificate $\mathcal{C}'_j$ from equation (4.3) without solving DLPGR, without inverting $\mathcal{J}$ (which is a many-to-one function) and without inverting the hash function $\mathcal{H}$. Consequently, for the attacker it is not feasible to forge the public-key illegally. □

Next, we talk about the public-key forgery attack model discussed in Lee et al. [25].

Theorem 4.2. *The novel key authentication scheme is secure against the ingenious attack model of Lee et al. [25].*

Proof. Let t be a malicious legal entity and let sk_t be his/her private key and let wk_j be his/her wrong public key. The entity t utilizes his/her private key to sign any record and the signature $\mathcal{C}_t$ must be certified by entity t by using his/her public key pk_t. However, at a later stage, in place of the actual certificate $\mathcal{C}_t$, t may provide a false certificate $\mathcal{C}'_t$ and deny signing the record to infer that wrong public key was utilized in the first place in the following manner:
(1) uses equation (4.1) to compute

$$wk_j^{\mathcal{J}(wk_j)} = E(\mathcal{C}'_j)\big(E(\mathrm{Pw}_j \oplus sk_j)\big)^{-1}.$$

(2) tries to obtain wk_j from $wk_j^{\mathcal{J}(wk_j)}$.
We claim that deducing wk_j from $wk_j^{\mathcal{J}(wk_j)}$ is harder than solving DLPGR. To see this, suppose that $wk_j \in \mathbb{F}_p$. This means $\mathcal{J}(wk_j) = wk_j$. That is to say that we need to deduce wk_j from $wk_j^{wk_j}$. Due to Agnew et al. [1], we know that deducing wk_j from $wk_j^{wk_j}$ is harder than DLP. Further, it is easy to see that DLPGR is hard problem than DLP as the group ring contains the group. Consequently, it follows that deducing wk_j from $wk_j^{\mathcal{J}(wk_j)}$ is harder than solving DLPGR. Thus, the result holds. □

Next, we show that our scheme is secure against the password guessing attack.

Theorem 4.3. *The presented key authentication scheme can withstand the password guessing attack launched by a malicious server.*

Proof. If there is a closed network environment, then it is believed that the servers are trusted. Therefore, it is highly unlikely that any server will initiate a password guessing attack. Consequently, one can assume that password table will not be illegally amended. An example of such an environment is any closed environment in which all the servers are controlled by a single admin. However, in order to strike at the server end, an attacker must have to guess the password Pw_j as well as sk_j in order to deduce $E(\mathrm{Pw}_j \oplus sk_j)$. But this is computationally infeasible as both are randomly chosen and are known only to the entity. Further, even if an attacker guesses the password by some means, e.g., derives it from the certificate

$$\mathcal{C}_j = \mathrm{Pw}_j \oplus sk_j + sk_j \mathcal{J}(pk_j),$$

it is still computationally infeasible to guess the secret key. Therefore, our scheme is safe against any such attack. □

Next, we show that our scheme is secure even if certificate gets intercepted.

Theorem 4.4. *Suppose that the password used in the certificate of public key gets intercepted in the presented scheme. It is still not possible for an adversary to find the private key from the certificate.*

Proof. We suppose that password Pw_j of entity j gets compromised by any means. Then, in order to deduce the secret key, an adversary may try to utilize (i) $E(\mathrm{Pw}_j \oplus sk_j)$, which is available on the server; (ii) certificate

$$\mathcal{C}_j = \mathrm{Pw}_j \oplus sk_j + sk_j \mathcal{J}(pk_j).$$

It is clear that if the adversary somehow deduces the secret key from the known value of $E(\mathrm{Pw}_j \oplus sk_j)$, then he/she has solved DLPGR, i.e., adversary deduced sk_j from the values of $u^{\mathrm{Pw}_j \oplus sk_j}$, where Pw_j is known. But we know that it is computationally infeasible to solve DLPGR.

For the possibility (ii), we assume that the adversary may try to obtain the secret key from the known value of certificate $\mathcal{C}_j$. More precisely, adversary tries to find the couple (r_1, r_2), where

$$r_1 = \mathrm{Pw}_j \oplus sk_j, \quad r_2 = sk_j \mathcal{J}(pk_j).$$

Suppose that the adversary succeeds in deducing the above-mentioned couple. Then, by XORing the compromised password Pw_j with r_1, the private key sk_j can be computed, i.e.,

$$sk_j = \mathrm{Pw}_j \oplus sk_j \oplus \mathrm{Pw}_j.$$

The same can be obtained from r_2 also as $\mathcal{J}(pk_j)$ is public knowledge. However, if Pw_j and sk_j are sufficiently large, then it is computationally infeasible to obtain the secret key from the certificate via brute force. Thus, result. □

Next, we briefly discuss the hardness of DLPGR.

4.1. Hardness of DLPGR

There is no known classical/quantum algorithm that can find the solution of DLP in group rings. However, one can always apply the brute force attack. So, in order to utilize DLPGR in cryptography, we study its brute force complexity.

4.1.1. Brute force attack

Let G be a finite group with order z and

$$u_1 = \sum_{j=1}^{z} r_{g_j} g_j.$$

Let order of u_1 be s, i.e., s is the least positive integer for which $u_1^s = 1$. It is straight-forward to see that on s multiplications of u_1 with itself, one can solve DLPGR discussed in Definition 2.8. We note that on multiplying u_1 with u_1, $O(2z^2)$ multiplications are required that includes z^2 group and z^2 ring multiplications. That is to say that DLPGR can be solved in $O(2z^2 s)$ multiplications. Let

$$R = \mathbb{Z}_p \quad \text{(finite field of order } p\text{)} \quad \text{and} \quad G = \langle g \rangle \quad \text{(cyclic group of order } z\text{)},$$

where p is a k_1-bit number and z is a k_2-bit number. Then one can solve DLPGR in

$$\mathcal{T} = O(z^2 2^\alpha (k_1^2 + k_2^2)) \quad \text{bit operations},$$

where the size of s is α bits. Clearly, if we consider the input size as the order of u_1, then $\mathcal{T}$ is an exponential time.

4.1.2. Collision algorithms

The general effect of any collision algorithm is that it reduces the number of bit operations by square root times the operations required for brute force. Furthermore, it is easy to see that the collision algorithms available to solve DLP in groups can be extended to solve DLPGR. Consequently, the number of bit operations needed to solve DLPGR through collision algorithms can be reduced to

$$\mathcal{T}' = O(z^2 2^{\alpha/2}(k_1^2 + k_2^2)) \quad \text{bit operations.}$$

To the best of authors' knowledge, currently, there is no algorithm that takes lesser than $\mathcal{T}'$ bit operations to solve DLPGR. Therefore, DLPGR is an extremely hard problem. The security provided by DLPGR with the different parameters is provided in the following Table 2.

Table 2. Size of parameters and the security provided by DLPGR.

Parameters Size	security (atleast)
$\|G\| = z \geq 2^7$, $s \geq 2^{225}$	128 bits
$\|G\| = z \geq 2^8$, $s \geq 2^{485}$	256 bits

5. Advantages of our scheme and comparison analysis

The various advantages of our scheme are discussed as follows:

(1) The size of parameters required for DLPGR is considerably smaller than the related hard problems such as integer factorization problem (IFP), DLP, DLP in a subgroup, Elliptic curve discrete logarithm problem (ECDLP), DLP with conjugacy search problem (DLCSP) (cf. [8]). This is shown in Table 3.

Table 3. Parameters sizes and security.

Hard Problem	Parameters sizes and security (bits)
IFP [31]	((3072, primes of size 1536 bits), 128)
DLP [31]	(3072, 128)
DLP in a subgroup (DSA) [31]	(256, 128)
DLCSP [8]	(48, 128)
ECDLP [2]	(256, 128)
DLPGR (Table 2)	$(10, 256)$

(2) The proposed scheme attains the third and top level of security mentioned by Girault [7]. This is because the regular key authentication schemes are insecure due

to the presence of authorities as they can work together in a bad way. However, in our scheme, there is no involvement of any authority. Consequently, there would not be any malicious collaboration among the certifying authorities.

(3) It is known that a public key in an identity-based (ID-based) scheme is the ID of entity. However, in our presented scheme, entity can effortlessly change his/her private key as well as password. Accordingly, on changing his/her private key and (or) password, the entity can also modify the associated data which involves password's picture, public key and the certificate. In addition, the authentication phase can also be performed by the entity himself/herself.

(4) Our scheme can be executed even if the image of password is not produced by using the Hash function $\mathcal{H}$. This is discussed in Subsection 3.1. However, the use of a hash function can considerably reduce the storage requirement.

(5) Suppose that it is possible to compute inverse of an element in a group ring through an oracle and $\mathcal{J}(pk_j) = 1$. Then equation (3.1) implies that

$$pk_j = E(\mathcal{C}_j)E(\mathrm{Pw}_j \oplus sk_j)^{-1}. \tag{5.1}$$

That is one can compute entity's public key via equation (5.1), where $E(\mathrm{Pw}_j \oplus sk_j)$ can be obtained from the table of passwords and $\mathcal{C}_j$ can be considered as self-certified public key. Therefore, the original public keys may be deleted from the public domain, since they are no longer required to store there. As a result, we only need to store the picture of the password and the self-certified public key. Girault [7] discussed that the public key file can be removed from the public domain in the self-certified schemes, provided the cryptographic scheme is non-interactive. So, our scheme is in line with the discussion of Girault. Thus, the storage required for this portion of the scheme is equal to that of ID-based scheme.

Next, we compare our scheme with the already available key schemes in the literature. Basically, we show that the computation cost of our scheme is very much comparable to various other schemes. We refer to Table 4 and Figure 1 to study the comparison analysis with various other schemes such as Hsieh et al. [16], Kumaraswamy et al. [23], Lee et al. [25], Liu et al. [30], Peinado [37], Wu et al. [43], Zhang et al. [45], Meshram et al. [32]. The notations used in Table 4 are as follows:
T_{inv}: Time required in a modular inverse computation
T_{mul}: Time required in a modular multiplication computation
T_{exp}: Time required in a modular exponentiation computation
T_{add}: Time required in a modular addition computation
$T_{\mathcal{H}}$: Time required in hash computation
T_{XOR}: Time required in a XOR function computation
T'_{add}: Time required in an addition computation through map $\mathcal{J}$
T'_{exp}: Time required in exponentiation computation in a group ring
T'_{mul}: Time required in a multiplication computation in a group ring.

It is worth to mention that for the parameters sizes mentioned in Table 3, we have that

$$T'_{\text{exp}} \approx T_{\text{exp}}, \quad T'_{\text{add}} \approx T_{\text{add}}, \quad T'_{\text{mul}} \approx T_{\text{mul}}.$$

6. Example

In this scheme, we study a toy example related to our scheme. All the results are calculated using the software GAP (Groups, Algorithm, Programming). We consider

$$R = \mathbb{Z}_5 = \{0, 1, 2, 3, 4\} \quad \text{and} \quad G = Q_8 = \langle x, y : x^4 = y^4 = e, x^2 = y^2, yx = x^{-1}y \rangle.$$

Registration phase: Let $u = 1 + xy$. Let $sk_j = 2$ be the private key. Then the public key is

$$pk_j = u^2 = 1 + 2xy + y^2.$$

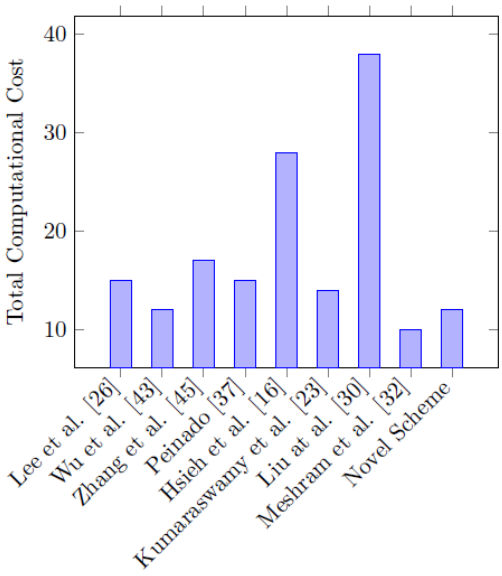

Figure 1. Key Authentication Schemes.

Let $\text{Pw}_j = 12$. Then

$$\text{Pw}_j \oplus sk_j = 14.$$

This means the encrypted password is

$$E(\text{Pw}_j \oplus sk_j) = u^{14}.$$

Also, we have

$$\mathcal{J}(pk_j) = 1 + 2 + 1 = 4 \quad \text{and} \quad (pk_j)^{\mathcal{J}(pk_j)} = (1 + 2xy + y^2)^4.$$

The picture to be stored is

$$\begin{aligned}&\mathcal{H}\big(\mathcal{J}(E(\mathrm{Pw}_j \oplus sk_j)\big(pk_j\big)^{\mathcal{J}(pk_j)}))\big)\\&= \mathcal{H}\big(\mathcal{J}((u^{14} * (1 + 2xy + y^2)^4)\big) = \mathcal{H}\big(\mathcal{J}(1 + y^2 + 2xy)\big) = \mathcal{H}(4). \qquad (6.1)\end{aligned}$$

Table 4. Computation cost in registration and authentication phases.

Authentication scheme	Registration phase	Authentication phase
Lee et al. (2003)	$T_{\text{inv}} + 4T_{\text{mul}} + 3T_{\text{exp}} + 2T_{\text{add}} + T_{\mathcal{H}}$	$2T_{\text{mul}} + 2T_{\text{exp}}$
Wu and Lin (2004)	$T_{\text{inv}} + 2T_{\text{mul}} + 4T_{\text{exp}} + 2T_{\text{add}} + T_{\mathcal{H}}$	$T_{\text{mul}} + T_{\text{exp}}$
Zhang and Kim (2005)	$2T_{\text{mul}} + T_{\text{exp}} + 3T_{\text{add}} + T_{\mathcal{H}}$	$3T_{\text{mul}} + T_{\text{exp}} + 4T_{\text{add}} + 2T_{\mathcal{H}}$
Peinado (2004)	$T_{\text{inv}} + 4T_{\text{mul}} + 3T_{\text{exp}} + 2T_{\text{add}} + T_{\mathcal{H}}$	$2T_{\text{mul}} + 2T_{\text{exp}}$
Hsieh and Leu (2012)	$9T_{\mathcal{H}} + 7T_{\text{XOR}}$	$7T_{\mathcal{H}} + 5T_{\text{XOR}}$
Kumaraswamy et al. (2015)	$3T_{\text{mul}} + 2T_{\text{exp}} + 3T_{\text{add}}$	$2T_{\text{mul}} + 3T_{\text{exp}} + T_{\text{add}}$
Liu et al. (2014b)	$4T_{\mathcal{H}} + 10T_{\text{XOR}} + 2T_{\text{mul}}$	$3T_{\mathcal{H}} + 6T_{\text{XOR}} + 13T_{\text{mul}}$
Meshram et al. (2017)	$2T_{\text{mul}} + T_{\text{exp}} + T_{\text{add}} + T_{\mathcal{H}} + 2T_{\text{XOR}}$	$2T_{\text{mul}} + T_h$
Our Scheme	$2T_{\text{mul}} + 2T_{\text{exp}} + 3T_{\text{add}} + T_{\mathcal{H}} + T_{\text{XOR}}$	$T_{\text{add}} + T_{\text{exp}} + T_h$

Certificate generation phase: The certificate corresponding to key pk_j is

$$\begin{aligned}\mathcal{C}_j &= \mathrm{Pw}_j \oplus sk_j + sk_j\mathcal{J}(pk_j)\\&= (12 + 2) + (2 \times 4) = 22.\end{aligned}$$

Verification phase: For verification, we must have

$$\mathcal{H}(\mathcal{J}(E(\mathcal{C}_j))) = \mathcal{H}\big(\mathcal{J}(E(\mathrm{Pw}_j \oplus sk_j)\big(pk_j\big)^{\mathcal{J}(pk_j)}))\big). \qquad (6.2)$$

We note that

$$\mathcal{H}(\mathcal{J}(E(\mathcal{C}_j))) = \mathcal{H}(\mathcal{J}(u^{22})) = \mathcal{H}(\mathcal{J}(1 + y^2 + 2xy)) = \mathcal{H}(4). \qquad (6.3)$$

Thus, verification is completed because of equations (6.1)–(6.3).

7. Conclusion

We have proposed a simple novel key authentication scheme for public key cryptosystems based on discrete logarithm problem in group ring. In our scheme, the

entity controls the certificate and authentication procedure is based on the table of passwords and there is no requirement of authorities in our scheme. We have carefully discussed the security of our scheme as well as the size of various parameters required in our scheme. Moreover, in order to show the worth of our scheme, we compared it with the several other related schemes. Finally, in order to show the practicality of our scheme, we have discussed a toy example.

Acknowledgements. The authors are thankful to the editor and anonymous reviewer of the manuscript for their valuable comments and suggestions that improved the paper to a great extent.

References

[1] G. Agnew, R. Mullin, S. Vanstone: *Improved digital signature scheme based on discrete exponentiation*, Electronics Letters 26.14 (1990), pp. 1024–1025, doi: 10.1049/el:19900663.

[2] D. Bernstein, J. Buchmann, E. Dahmen: *Post quantum cryptography*, Berlin, Heidelberg: Springer, 2009, doi: 10.1007/978-3-540-88702-7.

[3] C. Chang, Y. Chen, C. Lin: *A data embedding scheme for color images based on genetic algorithm and absolute moment block truncation coding*, Soft Computing 13.4 (2009), pp. 321–331, doi: 10.1007/s00500-008-0332-x.

[4] W. Diffie, M. Hellman: *New directions in cryptography*, IEEE Transactions on Information Theory 22.6 (1976), pp. 644–654, doi: 10.1109/TIT.1976.1055638.

[5] P. Dömösi, G. Horváth: *Hash functions based on Gluškov product of automata*, in: Eleventh Workshop on Non-Classical Models of Automata and Applications (NCMA 2019), Valencia, Spain, 2019, pp. 1–15.

[6] A. Evans, W. Kantrowitz, E. Weiss: *A user authentication system not requiring secrecy in the computer*, Communications of ACM 17.8 (1974), pp. 437–441, doi: 10.1145/361082.361087.

[7] M. Girault: *Self-certified public keys*, in: Proceddings of Eurocrypt'91, Valencia, Spain, 1991, pp. 490–497, doi: 10.1007/3-540-46416-6_42.

[8] N. Goel, I. Gupta, M. Dubey: *Undeniable signature scheme based over group ring*, Applicable Algebra in Engineering, Communication and Computing 27 (2016), pp. 523–535, doi: 10.1007/s00200-016-0293-8.

[9] I. Gupta, A. Pandey, M. Dubey: *A key exchange protocol using matrices over group ring*, Asian European Journal of Mathematics 12.5 (2019), p. 1950075, doi: 10.1142/S179355711950075X.

[10] C. Hannusch, G. Horváth: *Properties of Hash Functions based on Gluškov Product of Automata*, Journal of Automata, Languages and Combinatorics 26.1-2 (2021), pp. 55–65, url: jalc.de/issues/2021/issue_26_1-2/jalc-2021-055-065.php.

[11] D. He, N. Kumar, M. Khan, J. Lee: *Anonymous two-factor authentication for consumer roaming service in global mobility networks*, IEEE Transactions on Consumer Electronics 59.4 (2013), pp. 811–817, url: ieeexplore.ieee.org/document/6689693.

[12] D. He, N. Kumar, H. Shen, J. Lee: *One-to-many authentication for access control in mobile pay-TV systems*, Science China-Information Sciences 59.5 (2016), pp. 1–14, doi: 10.1007/s11432-015-5469-5.

[13] D. He, S. Zeadally, N. Kumar, J. Lee: *Anonymous authentication for wireless body area networks with provable security*, IEEE Systems Journal 11.4 (2016), pp. 2590–2601, url: ieeexplore.ieee.org/document/7458160.

[14] D. He, S. Zeadally, L. Wu: *Certificateless public auditing scheme for cloud-assisted wireless body area networks*, IEEE Systems Journal 12.1 (2015), pp. 64–73, URL: ieeexplore.ieee.org/document/7111218.

[15] G. Horng, C. Yang: *Key authentication scheme for cryptosystems based on discrete logarithms*, Computer Communications 19.9-10 (1996), pp. 848–850, DOI: 10.1016/S0140-3664(96)01112-7.

[16] W. Hsieh, J. Leu: *Exploiting hash functions to intensify the remote user authentication scheme*, Computers and Security 31.6 (2012), pp. 791–798, DOI: 10.1016/j.cose.2012.06.001.

[17] C. Hu, P. Liu, S. Guo: *Public key encryption secure against related-key attacks and key-leakage attacks from extractable hash proofs*, Journal of Ambient Intelligence and Humanized Computing 7.5 (2016), pp. 681–692, DOI: 10.1007/s12652-015-0329-0.

[18] C. Hu, P. Liu, Y. Zhou, S. Guo, Y. Wang, Q. Xu: *Public-key encryption for protecting data in cloud system with intelligent agents against side-channel attacks*, Soft Computing 20.12 (2016), pp. 4919–4932, DOI: 10.1007/s00500-015-1782-6.

[19] B. Hurley, T. Hurley: *Group ring cryptography*, International Journal of Pure and Applied Mathematics 69.1 (2011), pp. 67–86.

[20] S. Inam, R. Ali: *A new ElGamal-like cryptosystem based on matrices over groupring*, Neural Computing and Applications 29 (2018), pp. 1279–1283, DOI: 10.1007/s00521-016-2745-2.

[21] M. Khan, S. Kumari: *An authentication scheme for secure access to healthcare services*, Journal of medical systems 37.4 (2013), pp. 1–12, DOI: 10.1007/s10916-013-9954-3.

[22] M. Khan, S. Kumari: *Cryptanalysis and improvement of "an efficient and secure dynamic ID-based authentication scheme for telecare medical information systems"*, Security and Communication Networks 7.2 (2014), pp. 399–408, DOI: 10.1002/sec.791.

[23] P. Kumaraswamy, C. Rao, V. Janaki, K. Prashanth: *A new key authentication scheme for cryptosystems based on discrete logarithms*, Journal of Innovation in Computer Science and Engineering 5.1 (2015), pp. 42–47, URL: https://www.indianjournals.com/ijor.aspx?target=ijor:jicse&volume=5&issue=1&article=008.

[24] C. Laih, W. Chiou, C. Chang: *Authentication and protection of public keys*, Computers and Security 13.7 (1994), pp. 581–585, DOI: 10.1016/0167-4048(94)90009-4.

[25] C. Lee, M. Hwang, L. Li: *A new key authentication scheme based on discrete logarithms*, Applied Mathematics and Computation 139.2-3 (2003), pp. 343–349, DOI: 10.1016/S0096-3003(02)00192-3.

[26] W. Lee, Y. Wu: *A simple and efficient key authentication scheme*, in: Proceedings of The 18th workshop on combinational mathematics and computational theory, 2001, pp. 70–77.

[27] D. Li, J. Zhang, F. Guo, W. Huang, Q. Wen, H. Chen: *Discrete-time interacting quantum walks and quantum hash schemes*, Quantum information processing 12 (2013), pp. 1501–1513, DOI: 10.1007/s11128-012-0421-8.

[28] B. Liu, J. Bi, A. Vasilakos: *Toward incentivizing anti-spoofing deployment*, IEEE Transactions on Information Forensics and Security 9.3 (2014), pp. 436–450, DOI: 10.1109/TIFS.2013.2296437.

[29] C. Liu, K. Xie, Y. Miao, X. Zha, Z. Feng, J. Lee: *Study on the communication method for chaotic encryption in remote monitoring systems*, Soft Computing 10.3 (2006), pp. 224–229, DOI: 10.1007/s00500-005-0475-y.

[30] T. Liu, Q. Wang, H. Zhu: *A Multi-function Password Mutual Authentication Key Agreement Scheme with privacy preserving*, Journal of Information Hiding and Multimedia Signal Processing 5.2 (2014), pp. 165–178, URL: https://bit.nkust.edu.tw/~jihmsp/2014/vol5/JIH-MSP-2014-02-005.pdf.

[31] A. Menezes, P. Oorschot, S. Vanstone: *Handbook of applied cryptography*, CRC press, 2018, DOI: 10.1201/9780429466335.

[32] C. Meshram, C. Lee, C. Li, C. Chen: *A secure key authentication scheme for cryptosystems based on GDLP and IFP*, Soft Computing 21 (2017), pp. 7285–7291, doi: 10.1007/s00500-016-2440-3.

[33] C. Milies, S. Sehgal: *An introduction to group rings*, vol. 1, Springer Science and Business Media, 2002, url: https://link.springer.com/book/9781402002380.

[34] G. Mittal, S. Kumar, S. Kumar: *Novel public-key cryptosystems based on NTRU and algebraic structure of group rings*, Journal of Information and Optimization Sciences 42.7 (2021), pp. 1507–1521, doi: 10.1080/02522667.2021.1914811.

[35] G. Mittal, S. Kumar, S. Kumar: *A quantum secure ID-based cryptographic encryption based on group rings*, Sādhanā 47.1 (2022), p. 35, doi: 10.1007/s12046-022-01806-5.

[36] G. Mittal, S. Kumar, S. Narain, S. Kumar: *Group ring based public key cryptosystems*, Journal of discrete mathematical sciences and cryptography 25.6 (2022), pp. 1683–1704, doi: 10.1080/09720529.2020.1796868.

[37] A. Peinado: *Cryptanalysis of LHL-key authentication scheme*, Applied mathematics and computation 152.3 (2004), pp. 721–724, doi: 10.1016/S0096-3003(03)00590-3.

[38] S. Rososhek: *Cryptosystems in automorphism groups of group rings of Abelian groups*, Journal of Mathematical Sciences 154.3 (2008), pp. 386–391, doi: 10.1007/s10958-008-9168-2.

[39] S. Rososhek: *The unit group of Z_pQ_8*, Algebras Groups and Geometries 24 (2008), pp. 425–430.

[40] A. V. Tutueva, A. I. Karimov, L. Moysis, C. Volos, D. N. Butusov: *Construction of one-way hash functions with increased key space using adaptive chaotic maps*, Chaos, Solitons and Fractals 141 (2020), p. 110344, doi: 10.1016/j.chaos.2020.110344.

[41] T. Wang, Y. Liu, A. V. Vasilakos: *Survey on channel reciprocity based key establishment techniques for wireless systems*, Wireless Networks 21 (2015), pp. 1835–1846, doi: 10.1007/s11276-014-0841-8.

[42] L. Wei, H. Zhu, Z. Cao, X. Dong, W. Jia, Y. Chen, A. V. Vasilakos: *Security and privacy for storage and computation in cloud computing*, Information sciences 258 (2014), pp. 371–386, doi: 10.1016/j.ins.2013.04.028.

[43] T.-S. Wu, H.-Y. Lin: *Robust key authentication scheme resistant to public key substitution attacks*, Applied mathematics and computation 157.3 (2004), pp. 825–833, doi: 10.1016/j.amc.2003.08.074.

[44] B. Zhan, Z. Li, Y. Yang, Z. Hu: *On the security of HY-key authentication scheme*, Computer Communications 22.8 (1999), pp. 739–741, doi: 10.1016/S0140-3664(99)00032-8.

[45] F. Zhang, K. Kim: *Cryptanalysis of Lee–Hwang–Li's key authentication scheme*, Applied mathematics and computation 161.1 (2005), pp. 101–107, doi: 10.1016/j.amc.2003.12.012.

[46] J. Zhou, Z. Cao, X. Dong, N. Xiong, A. V. Vasilakos: *4S: A secure and privacy-preserving key management scheme for cloud-assisted wireless body area network in m-healthcare social networks*, Information Sciences 314 (2015), pp. 255–276, doi: 10.1016/j.ins.2014.09.003.