Soft Comput (2017) 21:7285–7291
DOI 10.1007/s00500-016-2440-3

FOUNDATIONS

A secure key authentication scheme for cryptosystems based on GDLP and IFP

Chandrashekhar Meshram[1,2] · **Cheng-Chi Lee**[3,4] · **Chun-Ta Li**[5] · **Chin-Ling Chen**[6]

Published online: 21 November 2016

Abstract The advancement of public-key cryptography in recent years has offered strong background support for the invention of numerous new system applications vastly employed in electronic business as well as other fields. However, that does not change the fact that the one-and-only Internet still remains open and unprotected. Therefore, for the sake of information security, confirming the legality of an entity's public key is always critical. Typically, a key authentication scheme needs one or more authorities to authenticate keys. To make a difference, in this study, we have developed a new key authentication scheme using generalized discrete logarithm problem and integer factorization problem for cryptosystems. Although the new scheme works pretty much the same way as regular certificate-based techniques, it differs in that it needs no authority. Taking the password/secret key pair as the certificate of public key for an entity, the new key authentication technique is very simple but profoundly secure.

Keywords Public-key cryptosystem · Authentication scheme · Integer factorization problem (IFP) · Certificate-based scheme · Generalized discrete logarithm problem (GDLP)

Communicated by A. Di Nola.

✉ Cheng-Chi Lee
cclee@mail.fju.edu.tw

Chandrashekhar Meshram
cs_meshram@rediffmail.com

Chun-Ta Li
th0040@mail.tut.edu.tw

Chin-Ling Chen
clc@mail.cyut.edu.tw

1 Department of Mathematics and Computer Science, Rani Durgavati University, Jabalpur, M.P., India

2 Department of Mathematics, RTM Nagpur University, Nagpur, M.S., India

3 Department of Library and Information Science, Fu Jen Catholic University, 24205 New Taipei, Taiwan, ROC

4 Department of Photonics and Communication Engineering, Asia University, 413 Wufeng Shiang, Taichung, Taiwan, ROC

5 Department of Information Management, Tainan University of Technology, 529 Jhong Jheng Road, 710 Tainan, Taiwan, ROC

6 Department of Computer Science and Information Engineering, Chaoyang University of Technology, 168, Jifeng E. Rd., Wufeng District, 41349 Taichung, Taiwan, ROC

1 Introduction

The rise of public-key cryptography tackled the issue of secure key agreement in routine symmetric key cryptography (Diffie and Hellman 1976; Wang et al. 2015; Wei et al. 2014). Moreover, it makes possible the creation of more advanced digital signature schemes. With the help of the innovations in public-key cryptography, electronic commerce of various kinds over open systems has become more and more conceivable and workable (Chang et al. 2009; He et al. 2013, 2015, 2016a, b; Hu et al. 2015, 2016; Jing et al. 2014; Liu et al. 2006, 2014a; Yao et al. 2015). In a typical public-key cryptographic scheme, each entity has a key couple, namely a secret key and a public key. Public keys are generally kept collectively in an open document such as a public-key manual, which is huge in size and yet fully exposed. A gatecrasher can simply trace an entity by replacing his/her public key with a fake key, unless there is a secure key authentication scheme to help watch the door. So far, quite a number of key authentication schemes have been developed and presented. For the most part, in the designs of these schemes there is at least

one authority, which is usually referred to as a key authentication center (KAC) or a trusted center (TC). The authority must be solid and strong against any sort of inside or outside attack because it is what the security of whole framework relies on.

Girault (1991) did an extensive survey into a wide range of key authentication schemes and observed that in those schemes the control of the authority over the secret keys fell into one of the three scenarios: (1) KAC has total control over an entity's secrete key; (2) KAC does not have access to an entity's secrete key, but it can create a false certificate undetected; (3) KAC does not have access to an entity's secrete key, and it can be demonstrated that the KAC produced a false certificate if it did. Girault then classified key authentication schemes into three levels of trust: (a) ID-based schemes (Shamir 1985); (b) certificate-based schemes; (c) self-certified public-key schemes. Of course Girault offered his own model of self-certified public-key design. Later, to improve Girault's model, Laih et al. (1994) presented a scheme that was a hybrid of an ID-based scheme and a certificate-based scheme.

In 1996, Horng and Yang (1996) presented a key authentication scheme for cryptosystems based on discrete logarithms. Horng and Yang's technique is similar to common certificate-based designs, but it requires no authorities. The certificate of the public key of an entity is created by combining her/his password and secrete key. The hash value of her/his password is handed over to the server and put away in the server's confirmation table. In 1999, pointing out that Horng and Yang's scheme was vulnerable to the password guessing attack, Zhan et al. (1999) presented their enhanced model. In addition, Lee and Wu (2001) also presented another upgraded scheme to guard against the password guessing attack. In 2003, Lee et al. (2003) claimed that the Zhan et al. key authentication scheme had an issue of non-repudiation of an entity's public key. Lee et al. then presented an improved key authentication scheme. Later on, Peinado (2004), Wu and Lin (2004), as well as Zhang and Kim (2005) separately showed that Lee et al.'s work had some security flaws and that anyone could derive the secrete key of an entity from public messages.

Now, since the new key authentication schemes for cryptosystems are still facing security difficulties and confidentiality concerns, in this paper, we shall present a new key authentication scheme for cryptosystems in the hope of mending the safety flaws of the current schemes. Functioning in the absence of any authority, the security of our new scheme is based on generalized discrete logarithm problem (GDLP) and integer factorization problem (IFP) (Meshram et al. 2012a, b). In other words, any attacker that attempts to break our new system is facing the difficulty of solving GDLP and IFP at the same time in the same multiplicative group over finite fields.

The rest of this paper is organized as follows. In Sect. 2, we shall briefly introduce some background materials upon which our new scheme is built. Then, in Sect. 3, we shall present the proposed key authentication scheme based on GDLP and IFP. After that, in Sect. 4, we will offer our security analysis and discussions. Then, Sect. 5 will show how our new scheme compares with some similar authentication schemes. Finally, the conclusion will be in Sect. 6.

2 Background materials

2.1 Password authentication procedure

In a password authentication scheme for multiuser computing systems, an entity needs to not only register with various systems, respectively, but also keep track of the various passwords that are purposely set to be different to achieve high-level security. A password is a series of characters expected to be distinguishable just to the system and the entity. When an entity needs to login to the system, he/she first enters his/her identity (ID) to help the system recognize the person and then provides his/her password (pwd) as a reaction to the system's demand. Then, the system confirms the pair of (ID, pwd) to see whether the entity is authorized. The system keeps all authorized pairs of (ID, pwd) in a table for authorization verification. However, the system would become extremely insecure should the verification table kept in the system be in plaintext, for an adversary could then easily access the table. Therefore, to keep legal passwords safe from the snooping of possible attackers, a cryptographic solution is recommended (Evans et al. 1974). By using a one-way function $h(.)$ (Khan and Kumari 2013, 2014; Yang et al. 2014; Zhou et al. 2015), passwords can be mapped to pictures, and the verification table can then be a table of the mapping results instead of one of the real passwords.

Based on GDLP and IFP, we choose an integer $N = p \times q$ and a primitive component $e \pmod N$, where q and p are two huge primes, and then we can use the capacity $e^{\text{pwd}} \pmod N$ as a picture for an entity's password pwd. Such pictures can then be put away in a generally open password table without leaking out any message about the passwords.

2.2 Public-key cryptosystem using GDLP and IFP

There are many public-key cryptosystems using GDLP and IFP (Meshram and Meshram 2011; Meshram and Powar 2016; Meshram et al. 2012a, b; Meshram and Obaidat 2015). In such schemes, there is an integer $N = p \times q$, where q and p are two huge primes, and there is also a primitive component $e \pmod N$ shared among a group of entities. Each entity arbitrarily chooses an integer $x < N$ and then calculates

$$y \equiv e^x \pmod{N}$$

Here, the public key and the corresponding secrete key of the entity are y and x, respectively.

Let's take an example and see how Meshram's et al.'s (2012b) scheme works. To sign a message M, we can compute

$$r \equiv e^k \pmod{N},$$

where k is an arbitrarily picked integer moderately prime to $\varphi(N)$. Here $\varphi(N) = (p-1)(q-1)$ (Hwang et al. 2013; Yang et al. 2013). Then we can unravel for s in the accompanying mathematical statement:

$$M \equiv xr + ks \bmod \varphi(N)$$

The signature for M is the couple (r, s). To check a signature, we affirm that

$$e^M \pmod{N} = y^r r^s \pmod{N}.$$

To encrypt a message M, we can select a random number b and compute

$$y_1 = e^b \bmod \varphi(N),$$
$$C_0 = M^{y_1} \pmod{N},$$

where k is an arbitrarily picked integer moderately prime to $\varphi(N)$.

Then we calculate the ciphertext

$$C = C_0^e \pmod{N}.$$

To decrypt the ciphertext C, we can compute

$$\gamma = C^d \pmod{N}$$

and recuperate the plaintext M by processing

$$\gamma^{d^b} \pmod{N}.$$

DSA is proposed by the US National Institute of Standards and Technology. It is a variant of the signature scheme based on GDLP and IFP and is depicted in Meshram et al. (2012b).

3 The proposed scheme

In this section, we present the first public-key authentication scheme for cryptosystems based on GDLP and IFP. Suppose an entity i, whose password is pwd_i, produced an integer sk_i as his/her secrete key and computed his/her public key pk_i as

$$\text{pk}_i \equiv h(\text{sk}_i) \equiv e^{\text{sk}_i} \pmod{N}.$$

Table 1 Notations used in the proposed scheme

Notations	Descriptions
pwd_i	The password of an entity i
pk_i, sk_i	The public key and private key of an entity i
p, q	Two large primes
$h(.)$	A one-way exponentiation function
C_i	The certificate of the public key pk_i
$\oplus$	XOR operation

The notations used in the proposed scheme are defined in Table 1.

3.1 Registration phase

This scheme depends on the accompanying suppositions:

1. The password authentication procedure of the system uses a one-way exponentiation function $h : Z_N^* \to Z_N^*$ defined by $h(x) = e^x \pmod{N}$ with respect to basis e and modulus N. By continual squaring, it is easy to calculate $h(b)$ for any given $b \in Z_N^*$, where $Z_N^* = (0, 1, \ldots, N-1)$ is a multiple cyclic group of order N. Let N be an integer that satisfies $= p \times q$, where q and p are huge primes. Its Euler function is given by $\varphi(N) = (p-1)(q-1)$. Choose an arbitrary integer $e, 1 \le e \le \varphi(N)$ s.t. $gcd(e, \varphi(N)) = 1$. It is also a primitive element $(\bmod N)$. Then, pick a distinct integer $d, 1 \le d \le \varphi(N)$ s.t. $ed \equiv 1 (\bmod \varphi(N))$. Then $h(.)$ goes public to all entities.
2. By successfully applying the one-way exponentiation function, the picture of i's password pwd_i is kept in the password table as $h(\text{pwd}_i \oplus \text{sk}_i)$, where $\oplus$ is an excusive-or operation and sk_i is also used against password guessing attacks.
3. Furthermore, to save on the consumption of the storage space for the system password table, a public hash function $H_N(.)$ can be utilized to hash the aftereffect of $h(.)$ of a password. Then, the picture $H_N(h(\text{pwd}_i \oplus \text{sk}_i))$ of the password pwd_i is stored in the password table.
4. The password table can now be directly open to all entities, as the passwords are already under proper protection.
5. The one-way function $h(.)$ is the same exponentiation function used by the cryptosystem based on GDLP and IFP.

3.1.1 Certificate generation phase

The certificate of the public key is produced by the entity, not by means of a TC or a KAC. Now entity i can process the certificate C_i of his/her public key by pairing up his/her password pwd_i and his/her secrete key sk_i such that

$$C_i \equiv \left(\mathrm{pwd}_i \oplus \mathrm{sk}_i + \mathrm{sk}_i\, \mathrm{pk}_i\right) (\mathrm{mod}\, \varphi\,(N))$$

The certificate C_i and pk_i are open to public in the network.

3.2 Authentication phase

Each entity presents three items for authentication: the public key, the encrypted password, and a certificate.

3.2.1 Key verification phase

The password image is given by

$$\begin{aligned} h\,(C_i) &\equiv e^{(\mathrm{pwd}_i \oplus \mathrm{sk}_i + \mathrm{sk}_i \mathrm{pk}_i)(\mathrm{mod}\, \varphi(N))} \,(\mathrm{mod}\, N) \\ &\equiv e^{(\mathrm{pwd}_i \oplus \mathrm{sk}_i)} e^{(\mathrm{sk}_i \mathrm{pk}_i)} \,(\mathrm{mod}\, N) \\ &\equiv \left(e^{(\mathrm{pwd}_i \oplus \mathrm{sk}_i)} \,(\mathrm{mod}\, N)\right) \left(e^{(\mathrm{sk}_i \mathrm{pk}_i)} \,(\mathrm{mod}\, N)\right) (\mathrm{mod}\, N) \\ &\equiv h(\mathrm{pwd}_i \oplus \mathrm{sk}_i) \mathrm{pk}_i^{\mathrm{pk}_i} \,(\mathrm{mod}\, N) \end{aligned}$$

Should the password picture get hashed, then we have

$$H_N(h(\mathrm{pwd}_i \oplus \mathrm{sk}_i)) \equiv H_N\left(h\,(C_i)\,/\mathrm{pk}_i^{\mathrm{pk}_i}\right)$$

when another entity j somehow needs to use entity i's public key, he/she first asks entity i for pk_i and C_i or gets them from the network, and then he/she gets i's password picture $h(\mathrm{pwd}_i \oplus \mathrm{sk}_i)$ or $H_N(h(\mathrm{pwd}_i \oplus sk_i))$. directly from the password table. Now, entity j can check the validity of entity i's public key by means of either of the following equations

$$h(\mathrm{pwd}_i \oplus \mathrm{sk}_i) \equiv h\,(C_i)\,/\mathrm{pk}_i^{\mathrm{pk}_i}$$

or

$$H_N(h(\mathrm{pwd}_i \oplus \mathrm{sk}_i)) \equiv H_N\left(h\,(C_i)\,/\mathrm{pk}_i^{\mathrm{pk}_i}\right)$$

If the equation holds, entity j acknowledges the public key and goes on to encrypt the messages.

4 Security analysis and discussions

When we say that the image of entity i's password $h(\mathrm{pwd}_i \oplus \mathrm{sk}_i)$ is protected by the system, we mean that it cannot be altered illicitly. What an intruder can do, then, is try to fashion the public key, speculate the password, or deduce the secrete key.

Theorem 4.1 *The presented key authentication scheme can withstand the public-key forgery attack.*

Proof Assume that an attacker tries to replace entity i's public key with a wrong key k_f. To make k_f certified as an actual public key, the attacker has to also come by a fake certificate C_f so that

$$h\,(C_\mathrm{f}) \equiv h\left(\mathrm{pwd}_i \oplus \mathrm{sk}_i\right) k_f^{k_f}$$

or

$$H_N(h(\mathrm{pwd}_i \oplus \mathrm{sk}_i)) \equiv H_N\left(h\,(C_\mathrm{f})\,/k_f^{k_f}\right)$$

To recover C_f, the intruder needs to calculate

$$C_\mathrm{f} \equiv h^{-1}\left(h(\mathrm{pwd}_i \oplus \mathrm{sk}_i)\right) + h^{-1}\left(k_f^{k_f}\right) (\mathrm{mod}\, \varphi\,(N))\,, \quad (1)$$

$$C_\mathrm{f} \equiv h^{-1}\left(h(\mathrm{pwd}_i \oplus \mathrm{sk}_i)\, k_f^{k_f}\right) (\mathrm{mod}\, \varphi\,(N))\,, \quad \text{or} \quad (2)$$

$$C_\mathrm{f} \equiv h^{-1}\left(H_N^{-1}(H_N\left(h(\mathrm{pwd}_i \oplus \mathrm{sk}_i)\right)) k_f^{k_f}\right) (\mathrm{mod}\, \varphi(N)), \quad (3)$$

where only $h^{-1}(k_f^{k_f})$ is controllable by the attacker. Since the attacker cannot change $h(\mathrm{pwd}_i \oplus \mathrm{sk}_i)$ or $H_N\left(h(\mathrm{pwd}_i \oplus \mathrm{sk}_i\right)$ in the password table, without knowing the entity i's password in both Eqs. (1) and (2), the attacker needs to solve both GDLP and IFP simultaneously in multiple cyclic groups. Meanwhile, in Eq. (3), the attacker will get trapped by the trouble of having to turn the hash function around. Hence it is clear that forging somebody's public key is not what is going to work for the attacker.

A model of public-key forgery attack has been posted on Horng and Yang (1996), Lee and Wu (2001), Zhan et al. (1999) in Lee et al. (2003). Although it is an ingenious model, it still will not work on cracking our key authentication scheme. Suppose a malicious yet legal entity l uses his/her secrete key sk_l to sign a record. Generally, the signature will be certified by utilizing l's public key pk_l. However, later the signer, namely l may deny signing the record and provide a fake certificate C_f instead of his/her genuine certificate C_l to infer that the public key was wrong in the first place as follows:

1. Calculate $k_f^{k_f} \equiv h\,(C_\mathrm{f})\, h\left(\mathrm{pwd}_l \oplus \mathrm{sk}_l\right) (\mathrm{mod}\, N)$; and
2. Attempt to find k_f from $k_f^{k_f}\ (\mathrm{mod}\, N)$.

However, solving the overhead K_f is actually harder than solving GDLP and IFP (Agnew et al. 1990). □

Theorem 4.2 *The proposed key authentication scheme is strong against the password guessing attack by a malicious server.*

Proof The servers tend to be highly trustworthy in some certain closed network environments, for example, inside of an enterprise where all servers are run by the same administrator(s). In such cases, it is not very likely that any server will launch a guessing attack, meaning that the public password table can be considered secure from illegal amendment. However, if an attacker should strike at the server end, for finding $h(\text{pwd}_i \oplus \text{sk}_i)$, the attacker would have to guess the password pwd_i. Clearly, it is computationally infeasible to guess pwd_i because the attacker must simultaneously guess pwd_i and the secrete key sk_i.

On the other hand, an attacker could derive the password from the certificate $C_i \equiv \left(\text{pwd}_i \oplus \text{sk}_i + \text{sk}_i \text{pk}_i\right) (\text{mod}\, \varphi(N))$ at the entity end, but then he/she would still fail to guess the password and the secrete key. □

Theorem 4.3 *The proposed key authentication scheme can keep the secret key from leaking out through an intercepted certificate.*

Proof In case an entity i's password is somehow compromised, the adversary may attempt to recover i's secrete key from $h(\text{pwd}_i \oplus \text{sk}_i)$, which is stored in the server. This means the adversary will have to solve GDLP and IFP simultaneously in multiple cyclic groups.

On the other hand, the adversary may decide to take another route and try to deduce the secret key from the certificate $C_i \equiv \left(\text{pwd}_i \oplus \text{sk}_i + \text{sk}_i\, \text{pk}_i\right) (\text{mod}\, \varphi(N))$. The adversary first assumes $C_i \equiv a + c \text{mod}\, \varphi(N)$, where $a \equiv \text{pwd}_i \oplus \text{sk}_i$ and $c \equiv \text{sk}_i\, \text{pk}_i$, and then tries to find all possible couples (a, c) that satisfy $C_i \equiv a + c \text{mod}\, \varphi(N)$.

For each couple, the adversary takes the following steps to find the secrete key.

1. XOR the compromised pwd_i with a to find sk_i'. On the off chance $\text{sk}_i' \text{pk}_i'$ should be identical to c, then he/she goes to the next step; otherwise, the adversary will have to repeat this first step and try the next couple, and this goes on and on until there is a hit.
2. Confirm if $h\left(\text{sk}_i'\right)$ is identical to pk_i. If yes, the adversary succeeds in finding the secrete key. Please notice that possible candidates for a include all the numbers from 1 to $\varphi(N)$; in other words, there can be a total of $\varphi(N)$ trial-and-error tests required. Given that the integer N is sufficiently extensive, it is computational infeasible to deduce the secrete key from C_i.

In regular key authentication schemes based on certificates or self-certified public keys, if all authorities work as a unity in a bad way, it is still possible for the system to become insecure. In the design of our new key authentication scheme, since there are no such things as authorities, there certainly is zero chance for malicious collaboration among certification authorities to happen. In other words, our new scheme achieves the third and top security level defined by Girault (1991).

In an identity-based authentication scheme, the public key is right the entity's ID. Distinctively, in our new key authentication scheme, an entity is free to change his/her password and secret key. When an entity changes his/her password and(or) public/secrete key, the three associated pieces of data can be effortlessly simplified: the entity's password picture in the system password table is to be simplified by the system, while the public key and its certificate can be simplified by the entity himself/herself. In addition, an entity can easily perform the authentication phase by himself/herself. Moreover, our new scheme can be executed using the self-certified public keys if the password image cannot be developed by using the hash function $H_N(.)$.

For example, in our authentication scheme, entity i's public key pk_i can be computed by

$$\text{pk}_i \equiv h(C_i) / h(\text{pwd}_i \oplus \text{sk}_i),$$

where C_i can be regarded as the self-certified public key and $h\left(\text{pwd}_i \oplus \text{sk}_i\right)$ is obtained from the password table. In this technique, the size of the system password table will go up. To deal with that, the original public keys can now be removed from the public file because we do not need them there anymore. By doing that, we can save half of the original space, since only the self-certified public keys, or certificates, remain in the file. As Girault has mentioned in his discussion on schemes using self-certified public keys, the public file can be removed if we do not require the cryptographic scheme to be non-interactive, since in our design an entity can ask another entity for his/her self-certified public key, and the key itself is a certificate. As a result, the storage space consumed for this part is equivalent to that of an identity-based scheme. □

5 Comparisons with other schemes

In this section, we shall demonstrate that our new key authentication scheme can run smoothly at a very low computation cost and that it supports all the functions an ideal key authentication scheme is supposed to serve. To see how our new scheme compares with some related schemes (Hsieh and Leu 2012; Kumaraswamy et al. 2015; Lee et al. 2003; Liu et al. 2014b; Peinado 2004; Wu and Lin 2004; Zhang and Kim 2005) in terms of computation cost, please check out Table 2 and Fig. 1.

Table 2 Computation cost in registration phase and authentication phase

Authentication scheme	Registration phase	Authentication phase
Lee et al. (2003)	$1t_{inv} + 4t_{mul} + 3t_{exp} + 2t_{add} + 1t_h$	$2t_{mul} + 2t_{exp}$
Wu and Lin (2004)	$1t_{inv} + 2t_{mul} + 4t_{exp} + 2t_{add} + 1t_h$	$1t_{mul} + 1t_{exp}$
Peinado (2004)	$1t_{inv} + 4t_{mul} + 3t_{exp} + 2t_{add} + 1t_h$	$2t_{mul} + 2t_{exp}$
Zhang and Kim (2005)	$2t_{mul} + 1t_{exp} + 3t_{add} + 1t_h$	$3t_{mul} + 1t_{exp} + 4t_{add} + 2t_h$
Hsieh and Leu (2012)	$9t_h + 7t_{XOR}$	$7t_h + 5t_{XOR}$
Liu et al. (2014b)	$4t_h + 10t_{XOR} + 2t_{mul}$	$3t_h + 6t_{XOR} + 13t_{mul}$
Kumaraswamy et al. (2015)	$3t_{mul} + 2t_{exp} + 3t_{add}$	$2t_{mul} + 3t_{exp} + 1t_{add}$
Our scheme	$2t_{mul} + 1t_{exp} + 1t_{add} + 1t_h + 2t_{XOR}$	$2t_{mul} + 1t_h$

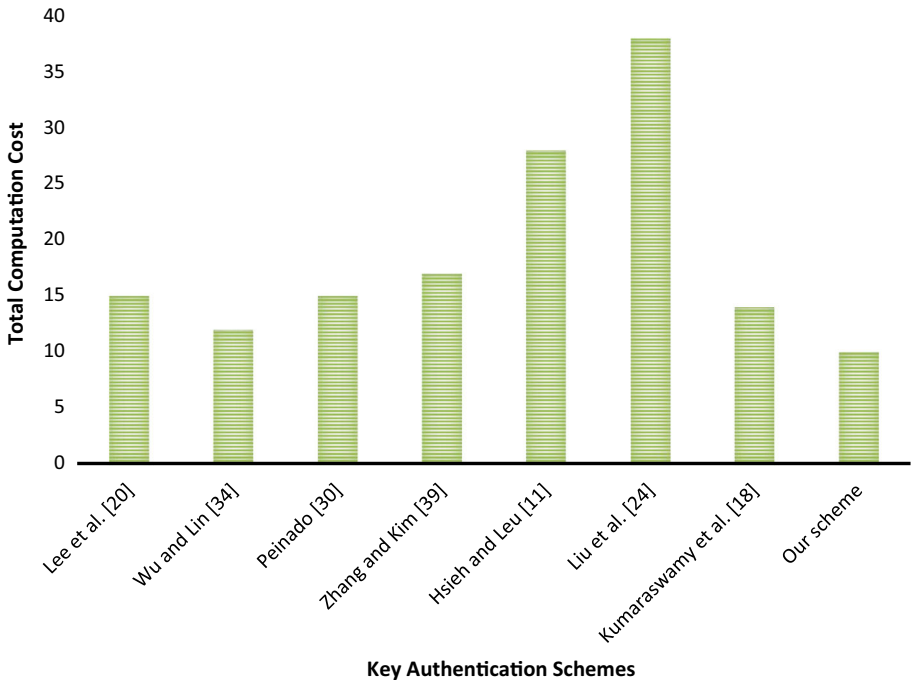

Fig. 1 Total computational cost in both registration phase and authentication phase

Here are the definitions of some notations we use in Table 2:

t_{inv}: Time for executing a modular inverse computation
t_{exp}: Time for executing a modular exponentiation computation
t_{mul}: Time for executing a modular multiplication computation
t_{add}: Time for executing a modular addition computation
t_h: Time for executing a one-way hash function computation
t_{XOR}: Time for executing a XOR function computation.

As Table 2 and Fig. 1 suggest, our new scheme runs at the lowest computation cost of them all. This means our new scheme is at the highest level of efficiency.

6 Conclusion

Key authentication schemes have come a long way, and yet the public key authentication problem has always stayed a major challenge. In this study, we try building up the strength of security on the basis of GDLP and IFP in a design where no authorities are included. In our key authentication scheme for cryptosystems, the certificate is controlled by the entity, while the authentication procedure relies on the password table at the system end. Such a design can indeed lift the level of trust of our new key authentication scheme to be significantly higher than that of self-certified schemes.

Acknowledgements The author would like to thank both anonymous reviewers for their helpful advice. This work was supported by Dr. D.S. Kothari Post-Doctoral fellowship awarded by University Grants Commission, New Delhi, India.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

Agnew GB, Mullin RC, Vanstone SA (1990) Improved digital signature scheme based on discrete exponentiation. Electron Lett 26:1024–1025

Chang CC, Chen YH, Lin CC (2009) A data embedding scheme for color images based on genetic algorithm and absolute moment block truncation coding. Soft Comput 13(4):321–331

Diffie D, Hellman ME (1976) New directions in cryptography. IEEE Trans Inf Theory 22(6):644–654

Evans A, Kantrowitz W, Weiss E (1974) A user authentication system not requiring secrecy in the computer. Commun ACM 17(8):437–441

Girault M (1991) Self-certified public keys. Proceedings of EUROCRYPTO 91:490–497

He D, Kumar N, Khan MK, Lee JH (2013) Anonymous two-factor authentication for consumer roaming service in global mobility networks. IEEE Trans Consum Electron 59(4):811–817

He D, Zeadally S, Wu L (2015) Certificateless public auditing scheme for cloud-assisted wireless body area networks. IEEE Syst J PP (99):1–10

He D, Kumar N, Shen H, Lee JH (2016a) One-to-many authentication for access control in mobile pay-TV systems. Sci China Inf Sci 59(5):1–14

He D, Zeadally S, Kumar N, Lee JH (2016b) Anonymous authentication for wireless body area networks with provable security. IEEE Syst J PP (99):1–12

Horng G, Yang CS (1996) Key authentication scheme for cryptosystems based on discrete logarithms. Comput Commun 19:848–850

Hsieh W, Leu J (2012) Exploiting hash functions to intensify the remote user authentication scheme. Comput Secur 31(6):791–798

Hu C, Liu P, Zhou Y, Guo S, Wang Y, Xu Q (2015) Public-key encryption for protecting data in cloud system with intelligent agents against side-channel attacks. Soft Comput 20(12):4914–4932

Hu C, Liu P, Guo S (2016) Public key encryption secure against related-key attacks and key-leakage attacks from extractable hash proofs. J Ambient Intell Humaniz Comput 7(5):681–692

Hwang MS, Lee CC, Tzeng SF (2013) A new proxy signature scheme for a specified group of verifiers. Inf Sci 227(1):102–115

Jing Q, Vasilakos AV, Wan J, Lu J, Qiu D (2014) Security of the internet of things: perspectives and challenges. Wirel Netw 20(8):2481–2501

Khan MK, Kumari S (2013) An authentication scheme for secure access to healthcare services. J Med Syst 37(4):9954

Khan MK, Kumari S (2014) Cryptanalysis and improvement of "An efficient and secure dynamic ID-based authentication scheme for telecare medical information systems". Secur Commun Netw 7(2):399–408

Kumaraswamy P, Rao CVG, Janaki V, Prashanth KVTKN (2015) A new key authentication scheme for cryptosystems based on discrete logarithms. J Innov Comput Sci Eng 5(1):42–47

Laih CS, Chiou WH, Chang CC (1994) Authentication and protection of public keys. Comput Secur 13:581–585

Lee WB, Wu YC (2001) A simple and efficient key authentication scheme. In: Proceedings of The 18th workshop on combinational mathematics and computational theory, pp 70–77

Lee CC, Hwang MS, Li LH (2003) A new key authentication scheme based on discrete logarithms. Appl Math Comput 139:343–349

Liu CL, Xie K, Miao Y, Zha XF, Feng ZJ, Lee J (2006) Study on the communication method for chaotic encryption in remote monitoring systems. Soft Comput 10(3):224–229

Liu B, Bi J, Vasilakos AV (2014a) Toward incentivizing anti-spoofing deployment. IEEE Trans Inf Forensics Secur 9(3):436–450

Liu TH, Wang Q, Zhu HF (2014b) A multi-function password mutual authentication key agreement scheme with privacy preserving. J Inf Hiding Multimedia Signal Process 5(2):165–178

Meshram C, Meshram S (2011) An identity based beta cryptosystem. In: IEEE Proceedings of 7th international conference on information assurance and security (IAS 2011), pp 298–303

Meshram C, Obaidat M (2015) An ID-based quadratic-exponentiation randomized cryptographic scheme. In: IEEE Proceedings of international conference on computer, information, and telecommunication systems (CITS 2015), pp 1–5

Meshram C, Powar PL (2016) An efficient identity-based QER cryptographic scheme. Complex Intell Syst. 1–7: doi:10.1007/s40747-016-0030-8

Meshram C, Meshram S, Gupta D (2012a) An ID-based beta cryptosystem using generalized discrete logarithm problem and integer factorization problem. J Inf Assur Secur 7(4):275–283

Meshram C, Meshram S, Zhang M (2012b) An ID-based cryptographic mechanisms based on GDLP and IFP. Inf Process Lett 112(19):753–758

Peinado A (2004) Cryptanalysis of LHL-key authentication scheme. Appl Math Comput 152:721–724

Shamir A (1985) Identity-based cryptosystems and signature schemes. In: Proceedings of CRYPTG, vol 84, pp 47–53

Wang T, Liu Y, Vasilakos AV (2015) Survey on channel reciprocity based key establishment techniques for wireless systems. Wirel Netw 21(6):1835–1846

Wei L, Zhu H, Cao Z, Dong X, Jia W, Chen Y, Vasilakos AV (2014) Security and privacy for storage and computation in cloud computing. Inf Sci 258:371–386

Wu TS, Lin HY (2004) Robust key authentication scheme resistant to public key substitution attacks. Appl Math Comput 157:825–833

Yang FY, Lo JH, Liao CM (2013) Improving an efficient ID-based RSA multisignature. J Ambient Intell Humaniz Comput 4(2):249–254

Yang H, Zhang Y, Zhou Y, Fu X, Liu H, Vasilakos AV (2014) Provably secure three-party authenticated key agreement protocol using smart cards. Comput Netw 58:29–38

Yao G, Bi J, Vasilakos AV (2015) Passive IP traceback: disclosing the locations of IP spoofers from path backscatter. IEEE Trans Inf Forensics Secur 10(3):471–484

Zhan B, Li Z, Yang Y, Hu Z (1999) On the security of HY-key authentication scheme. Comput Commun 22:739–741

Zhang F, Kim K (2005) Cryptanalysis of Lee-Hwang-Li's key authentication scheme. Appl Math Comput 161:101–107

Zhou J, Cao Z, Dong X, Xiong N, Vasilakos AV (2015) 4S: A secure and privacy-preserving key management scheme for cloud-assisted wireless body area network in m-healthcare social networks. Inf Sci 314:255–276