Hindawi
Security and Communication Networks
Volume 2022, Article ID 1232336, 10 pages
https://doi.org/10.1155/2022/1232336

Research Article

A Hyperelliptic Curve Cryptosystem Based Proxy Promised Signcryption Scheme

Farhad Ullah Khan,[1] Fahad Algarni,[2] Insaf Ullah,[3] Hanen Karamti,[4] Muhammad Anwaar Manzar,[3] Ahmed Saeed Alzahrani,[5] Muhammad Adnan Aziz,[6] and Muhammad Asghar Khan[3]

[1]*Department of Computer Science, Allama Iqbal Open University, Islamabad, Pakistan*
[2]*College of Computing and Information Technology, University of Bisha, Bisha, Saudi Arabia*
[3]*HIET, Hamdard University, Islamabad Campus, Islamabad 44000, Pakistan*
[4]*Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O.Box 84428, Riyadh 11671, Saudi Arabia*
[5]*Department of Computer Science, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 21589, Saudi Arabia*
[6]*Department of Electronic Engineering, Isra University, Islamabad 44000, Pakistan*

Correspondence should be addressed to Muhammad Asghar Khan; khayyam2302@gmail.com

Received 3 July 2021; Revised 18 February 2022; Accepted 3 March 2022; Published 27 April 2022

Academic Editor: Shehzad Ashraf Chaudhry

Signcryption is the method of combining a digital signature with encryption in a single logical step to get the cryptographic primitives of a public key. However, in addition to signcryption, we sometimes require anonymity and delegation of rights. In this paper, we propose a scheme called proxy promised signcryption to address all of these objectives simultaneously. In this scheme, the actual sender or signer delegates authority to an agent known as signcrypter for signcryption of the actual ciphertext. The agent entity is facilitated to generate a promised signcrypted text. The signcrypted text is then communicated to the intended recipient. The proposed work aims to reduce computation and communication costs, which has been tested by comparing it to existing schemes. The results obtained from this comparison support the aims of our scheme. The proposed scheme's security has also been evaluated using the Automated Validation of Internet Security Protocols and Applications (AVISPA) tool. The results validate that our scheme resists well-known cyberattacks.

1. Introduction

Transmitting data/information over an unreliable channel needs assurance and security. Data is an essential entity of any business/organization that is why for providing safety to the transmittable data, we need confidentiality, authenticity, and integrity. Confidentiality is provided by encryption algorithms; digital signature algorithms are employed for authenticity to be ensured, while using a one-way hash function, the integrity is assured. Before 1997, the sending data of the sender was to be encrypted first, and then the ciphertext signature of the data would be calculated, where it was a time-wasting process that used more machine cycles. Confiscating this restriction, Zheng [1] devised the word "signcryption." According to this scheme, a single step is used to combine digital signature with encryption to reduce computational cost and communication overhead. Therefore, some applications such as online contract signing and so on need proxy communication to be involved as a man-in-the-middle (MITM), while, in some sort of situation rights, delegation and deniability properties are mutually required. Hence, to provide a proxy communication membo, [2] was the first that coined a proxy signature scheme. A proxy signature or rights delegation is the process

of handing over signing authority by an original sender to an agent known as a proxy signer; there the agent (proxy) makes a signature on the behalf of the original sender. Zhang [3] contributed a complete and partial proxy signature with certificates. Zeng and Park [4] designed a threshold proxy signature scheme. Xiao [5] presented a new scheme called a multiproxy signature. Gamage et al. [6] stretched the scheme to alter proxy signature with proxy signcryption. Chan and Wei [7] projected a new proxy signcryption-based scheme on threshold property, while Ji-Guo et al. [8] demonstrated that the Chan and Wei [9] scheme is unable to satisfy strong unforgeability, nonrepudiation, and identifiability so that an improved scheme was offered. A proxy signcryption scheme [10] was proposed by Zhang that has properties of forward secrecy and message public verifiability, while this scheme suffers from limitations of cost deficiency. A novel research called warrant-based proxy signcryption was a project that is based on IF assumption proposed by Zhou et al. [11]. In this scheme, an attempt was made to define syntax and formalize notions of security. Another scheme named proxy signcryption that is established on discrete logarithm was projected by Elkamchouchi et al. [12]. In this scheme, a demonstration touch by numerical examples was given as well. Elkamchouchi et al. [13] made an attempt again that is known as a "proxy signcryption scheme with forward secrecy and public verifiability." This scheme was proposed for the original and the agent (proxy) signcrypter, while there a situation may occur in the form of a vulnerability known as an MITM attack. A proxy signcryption schemes was projected by Elkamchouchi et al. [14] founded on DLP. The scheme is totally employed using mathematics for realistic parameters (256 bit). Moreover, when the identity of the original signcrypter is verified, then the authorized proxy signcrypter might be able to craft legal proxy signatures. Using a BP in a PKI setting, Lo and Tsai [15] projected a competent signcryption. Thereafter, Lo and Tsai [16] initiated that the Lo and Tsai [15] scheme is vulnerable in terms of the chosen warrant attack (CWA). Hence, using BP, an identity-based proxy signcryption was proposed by Fei-Yu et al. [17] where the forward secrecy public verifiability was the future aspects of the scheme. This scheme [17] was exploited as insecure by Wang et al. [18]. Thereafter, an Id-based proxy signcryption was proposed by Wang and Cao [19]. An Id-based threshold proxy signcryption scheme was attempted by Wang and Liu [20].

Currently, the security service of deniability got a greater attraction in the area of cryptography because of the personal privacy protections that is necessary for business and real-life scenarios. Suppose a bidder that does not want to disclose the bid's basic contents to a third party or even to anybody. Thus, in such a case, the deniability property enables the bidder to deny self-participation if any conflict occurs. For "deniable authentication," a protocol was proposed by dwork [21] that provided a facility to enable a sender to communicate secrets without revealing self-privacy. To provide an efficient deniable authentication in a protocol, Aumann and Rabin [22] contributed to a scheme based on the factoring problem. However, this protocol was less efficient because of the high computational cost and communication overhead. After this, Deng et al. [23] coined two deniable authentication protocols based on the hardiness of the discrete logarithm problem (DLP). While these schemes also led to high computational costs and communication overhead. Another deniable authentication protocol was suggested by Fan et al. [22]. The protocol was operated on the Diffie–Hellman key exchange fashion to secure communication. However, there was a vulnerability to providing sender anonymity. A ring signature concept was coined by Noar [24] with the inclusion of forward deniability in terms of ring authentication. This kept hiding the original sender from all possible senders in a group. To hide the actual sender with noninteractive deniable authentication (NIdA), Susilo and Mu [25] planned a scheme based on ring authentication protocol. The scheme provided a facility to restrict the anticipated receiver to identify the original sender amongst group/groups. But the high computation was the limitation of the scheme. Using a generalized ElGamal signature, Lee et al. [26] contributed a scheme based on noninteractive deniability. By using the proposed security model, that is, noninteractive deniable authentication protocol, the nominated verifier has the competency to enquire about unforgeability and deniability. A scheme called "fully authentication service" was contributed by Harn and Jian Ren [27] for e-mail applications. To design e-mail authentication supported by Pretty Good Privacy (PGP) and Secure/Multipurpose Internet Mail Extensions (S/MIME), this scheme used cryptographic functions. To provide protection and anonymity to the original sender, Hwang and Sung [28] combined deniable authentication with encryption approaches called promised signcryption. Promised signcryption approach enables the signcrypter to generate a signcryptext with deniability property. However, there are some low-resource devices such as mobile phones and digital assistants that have low computational capabilities and consume more battery power to perform some heavy cryptographic computations. Addressing the aforementioned constraints, Insaf et al. [29] combined the properties of proxy signcryption and promised signcryption in a single approach known as proxy promised signcryption. In this approach, the original signer gives the signing rights to another entity called promised signcrypter (proxy). Later on, the proxy signcrypter generates the promised signcryption on behalf of the original sender/signer. Unfortunately, the approach [29] suffered from high processing cycles and greater bandwidth.

Most of the time, many issues are being faced regarding security in communication and the transition of data from one node to another [30]. In this regard, if some open channels are used for information exchange, vulnerability may arise, that is, intruders can achieve more opportunities to attack the crucial data. Thus, for the sake of secure communication, some techniques are needed that may provide a secure transmission approach to data, that is, confidentiality, authenticity, integrity, and nonrepudiation are the required properties in this regard. Moreover, low resource consumption and less burden on network channels is the main aim of current communications. For the sake of enhancement of both the aforementioned facilities, we have

tried HECC instead of ECC in our research and targeted three former attempts in the field, that is, Insaf [29], Elkamchouchi [31], and Ismail [32]. Due to the small key size, our scheme will result to consume laser computation cost and same as a small amount of communication overhead. The security validations of our scheme are run under a well-known tool of a simulation called AVISPA [33].

2. Preliminaries

Suppose f is the decisive attribute and let us say f^* is the closure of that very f attribute. Hence, the hyperelliptic curve (HEC) Hε of genus $g > 1$ that's over the attribute of f can be narrated as (Hε): $z2 + h\ (g)\ z = f\ (\delta)$, where $(\delta, \mathfrak{Z}) \in f * f$. Furthermore, the degree of at most g and $f\ (\delta) = f\ (\delta)$ is a polynomial, that is, $h\ (\delta) \in f\ (\delta)$ having degree 2g + 1. A pair of polynomials can be the divisor of HEC and can be represented using the Mumford [34]. It is clear that the most vital factor of every cryptographic system (CS) is the discrete logarithm problem (DLP) that is actually in some abelian groups. Let us say a randomly selected number is λ that is picked from the abelian group and that is being computed as λ. $D = d + d + d + \ldots + d$ shows a scalar multiplication of the said divisor. That is known as an HEC discrete logarithm problem, as it picks up a random number, that is, λ from λ. $D + d + d + \ldots\ldots + d$ is some sort of infeasible.

3. Proposed Research Model

The initial idea for HEC was defined by Koblitz [35] that is basically an indiscriminate form of an elliptic curve. Here, the points of HEC cannot be imitated from a group like the points of an elliptic curve, while it is computed of the additive abelian group that is achieved from divisors. Comparing HEC with the EC in terms of small base field size, HEC has more consistency and acceptance. Moreover, HEC has lower parameters size and the same security surveillance as that of RSA and elliptic curve while having a facility to reserve fewer hardware resources [36–38]. The newly proposed model is founded on the hardiness of HECDLP that attempts the proxy promised signcryption, and all the signing authorities are transferred to a proxy signcrypter instead the original signcrypter. A signcrypted cryptogram is calculated by the proxy signcrypter via using the original/sender signcrypter's credentials. This process is consists of the key generation stage (public and private keys), original signcrypter/sender, verification phase (proxy verification), promised signcryption stage (proxy), and unsigncryption stage (receiver). Table 1 shown below.

4. Key Generation Stage

(i) Original sender: Picks a random number. Up is a private key and generates its public key as Vp = Up.d

(ii) Proxy: Picks random number. Pa is a private key and generates its public key as Pb = Pa.d

(iii) Receiver: Picks random number. Ua is a private key and generates its public key as Ub = Ua.d

(iv) Original sender/signcryptext

In this section, the original sender first selects $X \in \{1, 2, 3, \ldots, n\}$ randomly, calculates $Z = X.d \bmod n$, computes $g = h\ (Z, \text{mw})$, calculates $\mathcal{O} = (\text{X} - \text{UP.g}) \bmod Q$ where d is the divisor over HEC and UP is a private key of the original sender, and then sends (Z, $\mathcal{O}$, mw) to proxy.

4.1. Proxy Verification. After receiving (Z, $\mathcal{O}$, mw) proxy signcrypter, compute $Z = \mathcal{O}.d + g.\text{Vp}$ where $g = h\ (Z, \text{mw})$ and accept if $g = g$ holds.

4.2. Proxy Promised Signcryption. After verification, the proxy picks $\in \{1, 2, 3, \ldots, n\}$ randomly, computes $Y = H\ (m// f.d)$, calculates $V = (f + \text{Y.Pa}) \bmod n$, computes the secret K = V.Up, computes the ciphertext by using the secret key $C = \text{EKx (m)}$, computes $S = \text{V.d} \bmod n$ where Up is the public key of the receiver and Pa is the private key of proxy, and then sends (C, Y, S) to receiver.

4.3. Unsigncryption. After receiving the tuple, the receiver first calculates secret key $K = S$.Ua, recovers the plain text by using the secret key $m = \text{dKx (C)}$, computes V= S – Y.Pb, and computes Y' = H (Y//m), where Ua is the private key of receiver and Pb is the public key of proxy.

The receiver only accepts if Y' = Y holds.

5. Security Proof and Comparison

Theorem 1 *The following equality shows the correctness between the original user and proxy:*

$$\begin{aligned} \mathcal{Z}' &= \upsilon \cdot \mathcal{D} + g\mathcal{V}\mathcal{P}, \\ \mathcal{Z}' &= (\mathcal{X} - \mathcal{U}(\mathcal{Z}, m_w))\mathcal{D} + g\mathcal{V}\mathcal{P}, \end{aligned} \tag{1}$$

where $\mathcal{O} = (X - Up.\ g)$.

$$\mathcal{Z}' = \left(\mathcal{X} - \mathcal{U}_p(\mathcal{Z}m_w)\right)\mathcal{D} + (\mathcal{Z}m_w)h \tag{2}$$

where $g = h\ (Z, mw)$.

$$\begin{aligned} \mathcal{Z}' &= \left(\mathcal{X} - \mathcal{U}_p h(\mathcal{Z}, m_w)\right)\mathcal{D} \\ &\quad + h(\mathcal{Z}, m_w)\mathcal{U}_p h \text{ Where } \mathcal{V}\mathcal{P} = \mathcal{U}_p\mathcal{D} \\ \mathcal{Z}' &= \mathcal{D}\left(\mathcal{X} - \mathcal{U}_p h(\mathcal{Z}, m_w)\right)\mathcal{U}_p h(\mathcal{Z}, m_w), \\ \mathcal{Z}' &= \mathcal{D}(\mathcal{X}) \\ &= \mathcal{X} \cdot \mathcal{D} = \mathcal{Z}. \end{aligned} \tag{3}$$

Theorem 2 *In this section, the unsigncrypter/receiver recovers the secret key by using the following equations:*

$$\begin{aligned} \mathcal{K} &= \mathcal{S}\mathcal{U}_a, \\ \mathcal{K} &= \mathcal{V}\mathcal{D}\mathcal{U}_a \text{Where } \mathcal{S} = \mathcal{V}\mathcal{D}, \\ \mathcal{K} &= \mathcal{V}\mathcal{U}_a\mathcal{D}, \\ \mathcal{K} &= \mathcal{V}\mathcal{U}_\text{b}, \text{Where } \mathcal{U}_\text{b} = \mathcal{U}_a\mathcal{D}, \end{aligned} \tag{4}$$

$K = K$, where V. Ub = K in promised signcryption section.

Table 1: Notations for proposed scheme.

Notations of algorithm	Notations for HLPSL code	Description
Kx	V′.Ub	Secret key
UP	Vp	Public key of sender
VP	Vp′	Private key of sender
pb	Pb	Public key of receiver
H, *h*, E	H1, H2, E	Hash function
pa	Pb′	Private key of receiver
Ub	Ub	Public key of proxy sender
Ua	Ub′	Private key of proxy sender
$\mho$	X′.{H1(X′.d′.Mw′)}_inv(Vp)	Signature (warrant message)
mw	Mw	Warrant message
Z	X′.d′	General value
C	E(V′.Ub.M′′)	Encryption
Y	H2(M′.F′.d′)	Hash value
S	F′.{H2(M′.F′.d′)}_inv(Pb).d′	Signature

6. Security Analysis

Our proposed scheme claims the provision of all the security properties that are provided in existing schemes such as confidentiality, warrant integrity, warrant unforgeability, message authenticity, unforgeability, message integrity, deniability, and anonymity. To authenticate the security properties of the proposed scheme, the AVISPA [33] simulation tool is used. AVISPA is mainly a programmed simulation tool that is used to perform a concrete validation, certification, and analysis against the internet safety and sensitive modules, application, and the cryptographic techniques. AVISPA can ensure whether the established protocol is SAFE or UNSAFE, but the security constraints will be essential. An HLPSL language format is necessary for the developed protocol, to find its outcomes. In its initial steps, a code is devised on the basis of HLPSL structures, and furthermore, it is then encoded into machine-understandable structure via the intermediate format (IF). For the said process, an HLPSL2IF translator is used to check the execution in reference to the given initial knowledge and every proxy can create the messages [39, 40].

6.1. Confidentiality. In the proposed scheme, it is assured to obey the confidentiality property. According to the scheme, if an attacker "A" makes an attempt to get access to the main subjects of a message, the attacker "A" needs to know about the secret session key "K," that is:

$$K = v \cdot ub, \tag{5}$$

while, in (5),

$$v = (f + Y \cdot pa). \tag{6}$$

Now, if the eavesdropper tries to calculate the value of "K," then essentially the eavesdropper will need two effortless efforts to be performed as follows.

Case 1. The value of "K" is essentially needed to be calculated from (5). Moreover, to complete this operation, the eavesdropper will have to know about "V" from (6). That is another computational hard for the said eavesdropper that equals to solving ECDLP.

Case 2. Some calculations may be performed by the attacker to find "V" using (6). But, for this, it is required to find out again the arbitrarily created number "*f*" in (6) and the key Pa (private) on the proxy side.

6.2. Warrant Integrity. Our scheme ensures also warrant integrity. The one-way hash is used to calculate the sender warrant message prior to send, that is, $g = h\,(Z, mw)$ and then send to proxy. Let us say to calculate $\mho$ in (7) where it is also essential for the attacker to find out the sender's private key "Up" from (8), generating up is a complex job for the attacker and will need to solve another HECDLP.

$$\mho = \left(x - u_p \cdot \mathcal{G}\right) \bmod q, \tag{7}$$

$$v = u_p \cdot D. \tag{8}$$

6.3. Warrant Unforgeability. In the proposed scheme, warrant unforgeability meets. If the attacker attempts to calculate a valid signature using (9), warrant likes "$\mho$" where

$$\mho = \left(x - u_p \cdot \mathcal{G}\right). \tag{9}$$

Hence, in (9), then "A" first needs *X* from HECDLP to be found out, and in some possibilities, let us say if *X* is achieved, then:

$$u_p \cdot \mathcal{G} = v_p, \tag{10}$$

while $v_p = u_p \cdot D$.

Here again, it is needed to face a challenge for HECDLP. So facing two times the hard problem cannot forge the sign.

6.4. Message Authenticity. In our scheme, the authentication property also meets. If an attacker "A" wants to get access to calculate a forge signature as shown in equation (5), *d* is publicly available in the network, which is possibly

Table 2: Computational comparisons in terms of major operations.

Participants	Scheme [31]	Scheme [29]	Scheme [41]	Proposed
Sender	1-ECPM	1-ECPM	3-ECPM	1-HECdM
Proxy	4-ECPM	4-ECPM	1-ECPM	4-HECdM
Receiver	3-ECPM	2-ECPM	1-ECPM	1-HECdM
Total	8-ECPM	7-ECPM	5-ECPM	6-HECdM

Table 3: Computational cost reduction in terms of milliseconds.

Participants	Scheme [31]	Scheme (ms) [29]	Scheme (ms) [41]	Proposed (ms)
Sender	4.24	4.24	12.72	2.2
Proxy	16.96	16.96	4.24	8.8
Receiver	12.72	8.48	4.24	2.2
Total	33.92	29.68	21.2	13.2

Table 4: Communication cost comparisons in terms of ciphertext size.

Ciphertext size (bits)	Scheme [31]	Scheme [29]	Scheme [41]	Proposed	Reduction from scheme [31]	Reduction from [29, 41]
128	128 + 128 + 480 + 160	128 + 160 + 160	128 + 160 + 160	128 + 80+80	(896 – 288)/ 896*100 = 67.85%	(448 – 288)/ 448*100 = 35.71%
256	256 + 128 + 480 + 160	256 + 160 + 160	256 + 160 + 160	256 + 80+80	(1,024 – 416)/ 1024*100 = 59.37%	(576 – 416)/ 576*100 = 27.77%
512	512 + 128 + 480 + 160	512 + 160 + 160	512 + 160 + 160	512 + 80+80	(1 280 – 672)/ 1280*100 = 47.5%	(832 – 672)/ 832*100 = 19.23%
1024	1,024 + 128 + 480 + 160	1,024 + 160 + 160	1,024 + 160 + 160	1,024 + 80+80	(1,792 – 1,184)/ 1, 792*100 = 33.92%	(1,344 – 1,184)/ 1, 344*100 = 11.90%

approachable by anyone in the network; then still the calculation of V would be really a complex job as follows:

$$S = v \cdot D, \tag{11}$$

$$v = (f + Y \cdot p_a). \tag{12}$$

Hence, in the situation, f and Pa in (12) are two unknown terms in a single equation to be accessed that results the signature cannot be disclosed; hence, no disclosure to sign leads to the authenticity in actual data.

6.5. Message Integrity. In the proposed scheme, prior to sending the message, proxy signcrypter first computes collision-resistant hash function of the message as in (12) and then forwards it to the addressee. If the invader wishes to translate the code/message C in (13) into C, then it is needed to change the m also, into m'. But we are going to use a fender-bender resisting one-way hash function that is computationally insufficient for the invader.

$$Y = H(f \| D), \tag{13}$$

$$C = kE_X(m), \tag{14}$$

while at receiver

$$Y' = H(Y \| m). \tag{15}$$

So, if the value of $Y' = Y$, then accept; otherwise, the message is fake and can be rejected.

6.6. Deniability and Anonymity. Our proposed scheme is composed of a private/secret network, where no third parties are considered within the promise between sender and receiver. It is just for the sake of information secrecy. Hence, if the receiver gets involved in violation of the promise, then the sender can deny via changing its source credentials, to keep self-anonymous and not to be proven by the tired one.

7. Computational Costs Analysis

Computational cost means the amount of machine cycles to be spent by the entire system, that is, original sender, proxy, and the message recipient. Usually, this cost is predictable by counting the number of principal operations involved in processing cycles. Typically, these operations include hyperelliptic curve divisor multiplications (HECdM). In Table 2, we illustrate the computational cost comparison of the proposed scheme with all the three schemes [31, 41] and [29] in terms of operations. We inspected, in the proposed scheme and the schemes [31, 41] and [29], that the costliest operations (major operations) are the hyperelliptic curve divisor multiplication (HECdM) and elliptic curve point's scalar multiplications (ECPM).

Table 3 demonstrates the comparison of the proposed with the other three schemes [31, 41] and [33] with respect to milliseconds. It is observed that the single scalar

Table 5: HLPSL code for proxy promised signcryption.

```
Role
role_ProxySender(ProxySender.agent, sender.agent, Receiver.agent, Receiver.agent,Vp:public_Key,SNd,RCV; channel(dy))
played_by Proxy sender
def =
        Local
        state:nat,Ns; text,X:text,H1:hash_func,Mw:text,h2:hash_func:text,d:text
        init
                state: = 0
        Transition
                1.state = 0∧RCV(Sender Receiver) = |> state: = 1∧Ns': = new()∧SNd(proxySender.{Ns'}_Pb)
                3.state = 1∧RCV(Sender.{X'{h1(X'.d'.Mw')}_inv(Vp).Mw'.X'.d'}_inv(Vp)) = |>state': = 2∧
request(Proxy sender, Sender, auth_1,Mw')∧secret(Mw',sec_2,{sender})∧F: = new()∧secret(M'sec_4,{Receiver})∧
witness(Proxysender, Receiver, auth_3.M′)∧V': = new()∧
SNd(Proxysender.{E(V'.Ub.M′).H2(M'.F'd')}_inv(pb).d'}_inv(pb))
Endrole
Role
role_Sender(ProxySender.agent, sender.agent, Receiver.agent, Receiver.agent,Vp:public_Key,SNd,RCV; channel(dy))
played_by sender
def =
        Local
                state:nat,Ns; text,X:text,H1:hash_func,Mw:text,d:text
        init
                state: = 0
        Transition
                1.state = 0∧RCV(start) = |> state': = 1∧SNd(Sender.Receiver)
                2.state = 1∧RCV(ProxySender.{X'{H1(X'.d'.Mw')}_inv(Vp).Mw'.X'.d'}_inv(Vp))∧
Endrole
Role
role_Receiver(ProxySender.agent, sender.agent, Receiver.agent, Receiver.agent,Vp:public_Key,SNd,RCV; channel(dy))
played_by receiver
def =
        Local
        state:nat, H2 hash_func,F:text,M:text,E:hash_func; V;text,d:text
        init
                state: = 0
        Transition
                6.state = 1∧RCV(ProxySender.{E(V'.Ub.M′).H2(M'.F'.d').F'{H2(M'.F'.d')}_inv(Pb).d'}
-inv(Pb)) = |>state: = 1
∧Secret(M′,sec_4,{Receiver})
Endrole
Role
role_session1(ProxySender.agent, sender.agent, Receiver.agent, Receiver.agent,Vp:public_Key,Ub:public_Key)
        Local
                SNd3,RCV3,SNd2,RCV2,SNd1,RCV1:channel(dy)
        Composition
                Role_ProxySender(ProxySender, sender,receiver,Vp,Pb,Ub,SNd3,RCV3)∧role_Receiver
(ProxySender, sender, receiver, Vp, Pb, Ub, SNd2, RCV2)∧role_sender(ProxySender, sender, receiver, Vp, Pb, Ub, SNd1, RCV1)
Endrole
Role
role_session2(ProxySender.agent, sender.agent, Receiver.agent, Receiver.agent,Vp:public_Key,Ub:public_Key)
def =
        Local
                SNd1,RCV1:channel(dy)
        Composition
                role_Sender(ProxySender, sender,receiver,Vp,Pb,Ub,SNd31,RCV1)
Endrole
```

multiplication consumes 4.24 ms for elliptic curve point multiplication (ECPM) and 2.2 ms for hyperelliptic curve divisors scalar multiplication (HECdM) on a PC running jdk1.6 having two cores of Intel CPU with processing speed of 2.00 GHz and primary memory capacity of 4 GB operating with Microsoft Windows Vista [42–44].

Furthermore, we use the generalized formula for the reduction of computational cost [40]:

Table 6: OFMC and ATSE simulation results.

	Summary
%OFMC	SAFE
%Version of 2006/02/13	dETAILS
SUMMARY	BOUNdEd_NUMBER_OF_SESSIONS
SAFE	TYPEd_MOdEL
dETAILS	PROTOCOL
BOUNdEd_NUMBER_OF_SESSIONS	/Home/span/span/testsuite/results/ design_and_analysis_of_proxy_promise_signcrypt
PROTOCOL	GOAL
/Home/span/span/testsuite/results/ design_and_analysis_of_proxy_promise_signcrypt	As specified
GOAL	BACKENd
As specified	CL-AtSe
OFMC	STATISTICS
COMMENTS	Analyzed; 8 states
STATISTICS	Reachable: 3 states
Parse time: 0.00 s	Translation: 0.00s
Search time: 0.01 s	computation: 0.00s
visitedNodes: 7 nodes	
depth: 5plies	

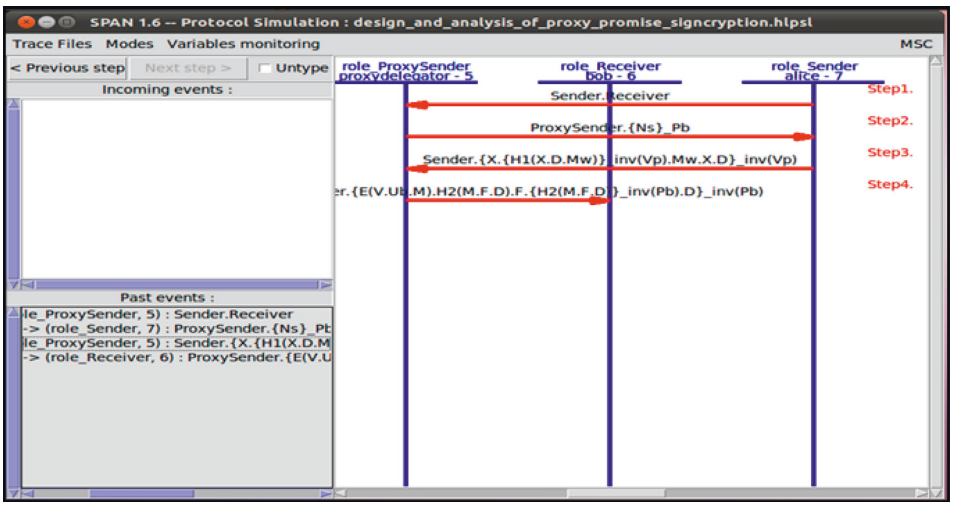

Figure 1: Actual flow of our scheme. [45–51].

$$\frac{\text{Existingscheme} - \text{proposedscheme}}{\text{Existingscheme}}. \tag{16}$$

The reduction of the proposed scheme as compared with the scheme [31] is shown below:

$$\frac{(33.92 - 13.2)}{33.92 * 100} = 61.08. \tag{17}$$

The reduction of the proposed scheme as compared with the scheme [41] is shown below:

$$(29.68 - 13.2) \cdot 29.68 * 55.5\%. \tag{18}$$

The reduction of the proposed scheme as compared with the scheme [29] is shown below:

$$\frac{(21.2 - 13.2)}{21.2 * 100} = 37.7\%. \tag{19}$$

8. Communication Overhead

In Table 4, we show communication cost comparisons of the proposed scheme with existing schemes [31, 41] and [29] with respect to different ciphertext sizes, for example, 128, 256, 512, and 1,024 bits. In the proposed scheme, wireless communication is involved, and there the factor of importance is communication overhead. Due to the bandwidth limits of wireless media, communication overhead for some cryptographic techniques must be kept to a minimum. As a result, a few cryptographic algorithms are required for wireless media to maintain a low communication overhead. Hence, the selection of parameters will mark most of the

communication cost for the proposed system and the amount of information that are to be transmitted. For simplification, we assume that:

(1) $|\mathrm{H}\ (\mathrm{value})| \cong |\mathrm{q}|$ where q is a large prime number $\gg 2\ 160$

(2) $|\mathrm{H}\ (\mathrm{value})| \cong |\mathrm{n}|$ where n is a large prime number $\gg 2\ 80$

For the sake to compare communication cost of proposed scheme $|C| + |n| + |H\ (\mathrm{value})|$ with scheme [31] and scheme [29, 41], we know that the communication cost of scheme [31] is $|C| + |mw| + 3|q| + |\ (\mathrm{value})|$ and schemes [29, 41]are $|C| + |q| + |H\ (\mathrm{value})|$.

Therefore, the generalized formula for the reduction of communication costs is as follows [40]:

$$\frac{\mathrm{Existingscheme} - \mathrm{proposedscheme}}{\mathrm{Existingscheme}}. \tag{20}$$

Communication overhead reduction of the proposed scheme as compared with the scheme [31] is as follows:

$$\frac{|c| + |m_w| + 3|q| + |\mathrm{H}(\mathrm{value})| - |c| + |\mathrm{n}| + |\mathrm{H}(\mathrm{value})|}{|c| + |m_w| + 3|q| + |\mathrm{H}(\mathrm{value})|}. \tag{21}$$

Communication overhead reduction of the proposed scheme as compared with the scheme [29, 41] is as follows:

$$\frac{|c| + |q| + |\mathrm{H}(\mathrm{value})| - |c| + |\mathrm{n}| + |\mathrm{H}(\mathrm{value})|}{|c| + |q| + |\mathrm{H}(\mathrm{value})|}. \tag{22}$$

9. Conclusion

In this article, we proposed proxy promised signcryption, which is based on HECC. The proposed scheme provided all the security requirements of proxy and promised signcryption schemes, where the AVISPA tool is tried to validate the scheme in terms of security analysis and validations. Furthermore, the computation and communication costs bounded 37.7% to 61.08% and 35.71% to 67.85% as compared to existing schemes [31, 41] and [29], respectively. The proposed method is ideal for resource-constrained devices since it can perform rapid implementations, follows rules for a smaller number of public keys, has fewer parameters, uses less power, and has fewer machine processing cycles.

Appendix

The proposed scheme is validated with the help of HLPSL language, and each role parameter is checked on OFMC and CL-Ate. For its proper validation, a well-known tool is used called AVISPA. Basic code is provided in Table 5, for the scheme that consists of three main agents (original sender, proxy sender, and receiver), as all these agents play a vital role in any communication channel. The proxy sender starts the communication by sharing his nonce by using his public key. In the response of the original sender, an encrypted text is sent to proxy, while proxy signs the text on the behalf of the original sender and forwards it to a receiver by using his public and private secret key with a nonce. The actual simulation process of the protocol is shown in Table 6 and Figure 1.

Data Availability

All data generated or analyzed during this study are included in this published article.

Conflicts of Interest

The authors declare that there are no conflicts of interest with respect to the research, authorship, and/or publication of this article.

Acknowledgments

Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R192), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

References

[1] Y. Zheng, "digital signcryption or how to achieve cost(signature & encryption) ≪ cost(signature) + cost(encryption)," in *Proceedings of the Advances in Cryptology-CRYPTO 97*, pp. 165–179, Springer-Verlag, Santa Barbara, CA, USA, August 1997.

[2] M. Mambo, K. Usuda, and E. Okamoto, "Proxy signatures for delegating signing operation," in *Proceedings of the 3rd ACM conference on Computer and communications security - CCS*, vol. 96, pp. 48–57, New delhi India, March 1996.

[3] K. Zhang, "Threshold proxy signature schemes," *Information Security*, pp. 191–197, Springer, Berlin, Germany, 1997.

[4] S. Kim, S. Park, and d. Won, "Proxy signatures, revisited," in *Proceedings of the Information and Communication Security, 1st International Conference, ICICS'97*, pp. 223–232, Beijing, China, November 1997.

[5] L. Yi, G. Bai, and G. Xiao, "Proxy multi-signature scheme: a new type of proxy signature scheme," *Electronics Letters*, vol. 36, no. 6, pp. 527-528, 2000.

[6] C. Gamage, J. Leiwo, and Y. Zheng, "An effcient scheme for secure message transmission using Proxy-signcryption," *Structure*, pp. 420–431, 1999.

[7] W. Chan and V. Wei, "A threshold proxy signcryption," in *Proceedings of the International Conference on Security and Management*, Washington, dC, USA, June 2002.

[8] L. I. Ji-Guo, L. I. Jian-Zhong, C. A. O. Zhen-Fu, and Z. Yi-chen, "A Nonrepudiable Threshold Proxy signcryption scheme with known Proxy agent," 2003.

[9] W. Chan and V. Wei, "A threshold proxy signcryption," in *Proceedings of the International Conference on Security and Management*, Las Vegas, Nevada, June 2002.

[10] Z. Zhang, Q. Dong, and M. Cai, "A new publicly verifiable proxy signcryption scheme," in *Proceedings of the Progress on Cryptography*, pp. 53–57, Chennai, India, ecember 2004.

[11] Y. Zhou, Z. Cao, and R. Lu, "Constructing secure warrant-based proxy signcryption schemes," in *Proceedings of the Cryptology and Network Security*, pp. 172–185, Xiamen, China, december 2005.

[12] d. H. Elkamshoushy, A. K. AbouAlsoud, and M. Madkour, "New proxy signcryption scheme with dSA verifier," in *Proceedings of the Twenty 3rd National Radio Science*

Conference (NRSC'2006), pp. 1–8, Nrsc, Menouf, Egypt, March 2006.

[13] H. Elkamchouchi, M. Nasr, and R. Ismail, "A new efficient strong proxy signcryption scheme based on a combination of hard problems," in *Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics*, pp. 5123–5127, San Antonio, TX, USA, October 2009.

[14] H. M. Elkamchouchi, E. F. Abu Elkhair, and Y. Abouelseoud, "An efficient proxy signcryption scheme based on the discrete logarithm problem," *International Journal of Information Technology, Modeling and Computing*, vol. 1, no. 2, pp. 7–19, 2013.

[15] N. Lo and J. Tsai, "A provably secure proxy signcryption scheme using bilinear pairings," *Journal of Applied Mathematics*, vol. 2014, Article ID 454393, 10 pages, 2014.

[16] N. W. Lo and J. L. Tsai, "A provably secure proxy signcryption scheme using bilinear pairings," *Journal of Applied Mathematics*, vol. 2014, Article ID 454393, 10 pages, 2014.

[17] L. Fei-yu, C. Wen, C. Ke-fei, and M. Chang-she, "Efficient identity based signcryption scheme with public verifiability and forward security," *Wuhan University Journal of Natural Sciences*, vol. 10, no. 1, pp. 248–250, 2005.

[18] F. Li, X. Xin, and Y. Hu, "Id-based threshold proxy signcryption scheme from bilinear pairings," *International Journal of Security and Networks*, vol. 3, no. 3, pp. 206–215, 2008.

[19] Q. Wang and Z. Cao, "Two proxy signcryption schemes from bilinear pairings," *Cryptology and Network Security*, pp. 161–171, 2005.

[20] Y. Ming and Y. Wang, "Proxy signcryption scheme in the standard model," *Security and Communication Networks*, vol. 8, no. 8, pp. 1431–1446, 2015.

[21] C. dwork and M. Naor, "Concurrent zero-knowledge," in *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pp. 409–418, ACM, dallas TX, USA, May 1998.

[22] Y. Aumann and M. Rabin, "Efficient deniable authentication of long messages," in *Proceedings of the International Conference On Theoretical Computer Science in Honor of Professor Manuel Blum's 60th Birthday*, pp. 20–24, Hualien, Taiwan, April 1998.

[23] X. deng, H. Zhu, and C. H. Lee, "eniable authentication protocols," *IEE Proceedings - Computers and digital Techniques*, vol. 148, no. 2, pp. 101–104, 2001.

[24] M. Naor, "deniable ring authentication," in *Proceedings of the 22nd Annual International Cryptology Conference on Advances in Cryptology*, pp. 481–498, Santa Barbara, CA, USA, August 2002.

[25] W. Susilo and Y. Mu, "Non-interactive deniable ring authentication," in *Proceedings of the Information Security and Cryptology-ICISC*, pp. 386–401, Seoul, Korea, december 2004.

[26] W.-B. Lee, C.-C. Wu, and W.-J. Tsaur, "A novel deniable authentication protocol using generalized ElGamal signature scheme," *Information Sciences*, vol. 177, no. 6, pp. 1376–1381, 2007.

[27] L. Harn and J. Jian Ren, "esign of fully deniable authentication service for E-mail applications," *IEEE Communications Letters*, vol. 12, no. 3, pp. 219–221, 2008.

[28] S.-J. Hwang and Y.-H. Sung, "Confidential deniable authentication using promised signcryption," *Journal of Systems and Software*, vol. 84, no. 10, pp. 1652–1659, 2011.

[29] I. Ullah, N. U. Amin, and A. I. Umar, "Proxy promised signcrypion scheme based on elliptic curve crypto system," *International Journal of Computer*, vol. 20, no. 1, pp. 167–173, 2016.

[30] S. Hawkins, d. C. Yen, and d. C. Chou, "Awareness and challenges of Internet security," *Information Management & Computer Security*, vol. 8, 2000.

[31] E. F. A. Elkhair, "An efficient Proxy signcryption scheme," *International journal of information technology modeling and computing(IJITMC)*, vol. 1, no. 2, pp. 7–19, 2013.

[32] L. Fan, C. X. Xu, and J. H. Li, "deniable authentication protocol based on Deffie-Hellman algorithm," *Electronics Letters*, vol. 38, no. 14, pp. 705-706, 2002.

[33] L. Virani`, "Automated security protocol analysis with the arista tool," *Electronic Notes in Theoretical Computer Science*, vol. 155, pp. 61–86, 2006.

[34] R. Ganesan and M. Gobi, "E-commerce channel 4 hyperelliptic curve cryptosystems," *International Journal on Network Security*, vol. 11, no. 3, pp. 121–127, 2010.

[35] N. Koblitz, "Hyperelliptic cryptosystems," *Journal of Cryptology*, vol. 1, no. 3, pp. 139–150, 1989.

[36] T. Wollinger, "Software and hardware implementation of hyperelliptic curve cryptosystem," issertation for the degree of doctor-lngenieur, p. 201, Ruhr-Universit at Bochum, Bochum, Germany, Ruhr-Universit at Bochum, 2004.

[37] J. Pelzl, T. Wollinger, J. Guajardo, and C. Paar, "Hyperelliptic curve cryptosystems: closing the performance gap to elliptic curves," p. 15, 2003, http://eprint.iacr.org/026.pdf.

[38] J. Pelzi, T. Wollinger, and C. Paar, "High performance arithmetic for hyper elliptic curve cryptosystems of genus two," p. 12, 2004, http://eprint.iacr.org/212.pdf.

[39] d. Dolev and A. Yao, "On the security of public-key protocols," *IEEE TransActions on Information Theory*, vol. 2, no. 29, 1983.

[40] d. Basin, S. Mödersheim, and L. Viganò, "An on-the-fly model-checker for security protocol analysis," in *Proceedings of the Computer Security - ESORICS 2003, 8th European Symposium on Research in Computer Security*, pp. 253–270, Gjøvik, Norway, October 2003.

[41] R. Ismail, "A novel proxy signcryption scheme and its elliptic curve variant," *International Journal of Computers and Applications*, vol. 165, no. 2, 2017.

[42] Y. Boichut, P.-C. Heam, O. Kouchnarenko, and F. Oehl, "Improvements on the genet And klay technique to automatically verify security protocols," in *Proceedings of the AVIS '04 3rd International Workshop on Automatic Verification of Infinite-State Systems*, Barcelona, Spain, April 2004.

[43] I. Ullah, A. Alkhalifah, S. Rehman et al., "A provable and privacy-preserving authentication scheme for UAV-Enabled intelligent transportation systems," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 5, pp. 3416–3425, 2022.

[44] M. A. Khan, H. Shah, S. U. Rehman et al., "Securing internet of drones with Identity-Based Proxy signcryption," *IEEE Access*, vol. 9, pp. 89133–89142, 2021.

[45] M. Wang and Z. Liu, "Identity based threshold proxy signcryption scheme," in *Proceedings of the 5th International Conference on Computer and Information Technology (CIT'05)*, vol. 2005, pp. 695–699, Shanghai, China, September 2005.

[46] U. Insaf, I. U. H. Muhammad, A. Noor, U. Arif, and K. Hizbullah, "Proxy signcrypion scheme based on hyper elliptic curves," *International Journal of Computer*, vol. 20, no. 1, pp. 157–166, 2016.

[47] N. Koblitz, "Elliptic curve cryptosystems," *Mathematics of Computation*, vol. 48, no. 177, pp. 203–209, 1987.

[48] d. Mumford, "Tata lectures on theta 1I," in *Modern Birkhäuser Classics,* Vol. 43, Springer, Salmon, NY, USA, 1984.

[49] Y. Chevalier and L. Vigneron, "Automated unbounded verification of SecurityProtocols," in *Proceedings of the Computer Aided Verification 14th International Conference, CAV 2002*, Copenhagen, denmark, July 2002.

[50] A. Armando and L. Compagna, "SATMC: a SAT-based model checker for security protocols," in *Proceedings of the Logics in Artificial Intelligence 9th European Conference JELIA 2004*, Lisbon, Portugal, September 2004.

[51] A. C. Shehzad, "Public verifiable signcryption schemes with forward secrecy based on hyper elliptic curve cryptosystem," *ICISTM 2012, CCIS*, vol. 285, pp. 135–142, 2012.