Journal of Discrete Mathematical Sciences & Cryptography
ISSN 0972-0529 (Print), ISSN 2169-0065 (Online)
DOI : 10.47974/JDMSC-2832

A new hybrid digital signature scheme

Jay Mehta *
Pushparajsinh Parmar †
Department of Mathematics
Sardar Patel University
Vallabh Vidyanagar 388120
Gujarat
India

Abstract

This paper introduces a new hybrid digital signature scheme, which is constructed from two widely recognized computationally difficult cryptographic problems, namely, the discrete logarithm problem (DLP) and the integer factorization problem (IFP). The hybrid scheme proposed here turns out to be superior in comparison to both the RSA and Elgamal signature schemes.

Subject Classification: *94A60, 94A62, 11T71.*

Keywords: *Elgamal digital signature scheme, RSA signature, The discrete logarithm problem, The integer factorization problem, Hybrid scheme.*

1. Introduction

In the contemporary digital age, digital signatures play a vital role in assuring the integrity and authenticity of digital data or documents. Digital signatures are essential in areas such as securing emails, electronic documents, checking the integrity of software, online banking, blockchain systems, and other authentication processes. Generally, digital signature schemes are based on challenging mathematical problems like the *discrete logarithm problem* (DLP), the *elliptic curve discrete logarithm problem* (ECDLP), the *integer factorization problem* (IFP), etc. These problems are computationally very difficult to solve without the knowledge of the secret signing key(s), which ensures the unforgeability of the signature.

The origins of digital signatures are rooted in 1976, when the term `digital signature' first appeared in [1]. However, the authors did not provide a concrete construction of a signature scheme. They first created a secure key exchange

* *E-mail:* jay_mehta@spuvvn.edu (Corresponding Author)
† *E-mail:* pushparaj1507@spuvvn.edu

© Taru Publications

algorithm over an open insecure communication channel whose security is based on the computational difficulty of the well-known *Diffie-Hellman problem*. Rivest, Shamir, and Adleman, in 1978, introduced the first digital signature scheme and a public key cryptosystem, widely popular as the RSA [2]. The RSA digital signature plays a vital role in cryptography. It serves as a foundational component in secure email, software distribution, digital certification, execution of electronic contracts, and among many other applications. The security of RSA depends on the problem of integer factorization. Since the development of the RSA signature, many researchers have shown that it is vulnerable to various attacks. Several attacks on RSA are presented in [3], and an updated survey is available in [4]. Throughout the year, researchers have introduced numerous modifications and improvisations to strengthen the RSA due to its susceptibility to specific attacks (see [5, 6, 7]). In the recent improvisation of the RSA (cryptosystem), Mehta and Rana [8, 9] proposed a generalized multi-moduli RSA.

The first digital signature scheme based on the discrete logarithm problem was proposed by Elgamal in 1985, widely known as the Elgamal digital signature scheme [10]. The Elgamal digital signatures are extensively used in areas like blockchain and cryptocurrency platforms, e-voting systems, cloud computing, etc. Furthermore, the scheme serves as the foundation for several standards and cryptographic protocols, including the *digital signature algorithm* and has been adopted by the U.S. government for digital signature generation and verification. As in the case of RSA, over the years, many researchers have proposed variants and extensions of the Elgamal digital signature scheme (see [11, 12, 13]). In this direction, to address limitations and performance in a digital signature scheme, researchers have also adopted a hybrid approach.

A hybrid signature scheme is formed by integrating multiple signature schemes, each relying on a distinct computationally hard problem. The benefits of such a signature scheme are that, even if one of the hard problems is solved, the hybrid scheme still remains secure. In 1994, Harn [14] proposed a hybrid public-key cryptosystem and a digital signature scheme based on the discrete logarithm and the integer factorization problem. However, Lee and Hwang [15] later demonstrated that if an adversary is successful in solving the discrete logarithm problem, then Harn's signature can be easily forged. To address this vulnerability, they introduced a modified version of the scheme to overcome this limitation. A comprehensive overview of hybrid signature schemes based on these two problems can be found in [16]. The implementation of such a hybrid approach is neither novel nor recent. These hybrid strategies are not only implemented in digital signatures, but also seen to be commonly In one such instance, recently, Mehta and Parmar [17] proposed a hybrid cryptosystem whose security depends on the two aforementioned problems. Applied in practice in various cryptosystems as well as key-exchange algorithms.

Along this idea, we propose a hybrid digital signature scheme which is secured by two well-known hard cryptographic problems: the problem of finding

the discrete logarithm and the factorization of a large interger. The paper is structured as follows. In Section 3, we describe our proposed hybrid scheme, and we analyze its security aspects in Section 4. Before presenting our scheme, a brief overview of the RSA and the Elgamal digital signature is provided in the succeeding section. We conclude the paper, with a conclusion section (Section 5) in which we collate the results about the security analysis of our proposed signature scheme, and elucidate its preferability over the two aforementioned signature schemes.

2. Preliminaries

As the Elgamal and RSA digital signature schemes serve as the foundation for our proposed hybrid signature scheme. As a prerequisite and to make our paper self-contained, we refer to these popular signature schemes in this section.

2.1 *Elgamal Digital Signature Scheme*

In the Elgamal signature scheme [10], the signing party or a trusted third party or any agency provides a large prime number r and an element $g \in \mathbb{F}_r^*$. The signing party, say the signer, chooses the secret signing key a and computes the public verification key $A \equiv g^a (\mathrm{mod}\, N)$. To sign a digital document D, the signer chooses a random element k co-prime to $p-1$ and computes the pair of signatures $S = (S_1, S_2)$ as follows.

$$S_1 \equiv g^k (\mathrm{mod}\, r),$$
$$S_2 \equiv (D - aS_1)k^{-1} (\mathrm{mod}\, r-1).$$

The signature S on the document D is validated by the verifier by evaluating $A^{S_1} S_1^{S_2} (\mathrm{mod}\, r)$ and confirming it to be equal to $g^D (\mathrm{mod}\, r)$.

Despite its popularity, the Elgamal signature scheme is prone to various attacks, some of which are described in Section 4.

2.2 *RSA Digital Signature*

In the RSA signature scheme [2], the signer begins by generating two large confidential primes p and q, and forms the public modulus N as their product. Furthermore, the signer chooses the verification key e coprime to $\phi(N) = (p-1)(q-1)$. Thereafter, the signer computes the secret signing key d which is e^{-1} modulo $\phi(N)$. A digital document D is then signed by the signer by computing the signature S as follows.

$$S \equiv D^d (\mathrm{mod}\, N).$$

A verifier checks the signature S on the document D is valid by computing $S^e (\mathrm{mod}\, N)$ and ensuring it to be equal to D.

The RSA digital signature scheme is also vulnerable to certain attacks like the Elgamal signature scheme. We describe a few such attacks on the RSA scheme in Section 4.

3. A new proposed hybrid signature scheme

A novel digital signature scheme is introduced in this section. Security-wise, the proposed hybrid scheme proves to be superior than both its preliminary signature schemes.

3.1 *The hybrid signature scheme*

A large prime r and an element $g \in \mathbb{F}_r^*$, preferably a primitive root, are chosen by the signing party or provided by an agency. The signing party, whom we call the signer, chooses a secret signing key, an integer $2 < a < r$ for which finding the discrete logarithm is difficult. The signer computes

$$A \equiv g^a (\mathrm{mod}\ r).$$

Furthermore, the signer selects large primes p and q such that $r < \phi(pq)$ and computes its product $N = pq$, the public modulus. Finally, the signer chooses a number e such that $\gcd(e, \phi(pq)) = 1$ and computes its inverse d modulo $\phi(pq) = (p-1)(q-1)$.

The triplet (A, N, e) is then made public, and is called the verification key triplet, while the numbers a, p, q, and d are called the secret signing keys.

Suppose the signer wants to put a signature on a digital document D modulo r. For this, the signer uses a random value of k with $\gcd(k, r-1) = 1$. The signing process is carried out by evaluating the following three quantities:

$$\begin{aligned} S_1 &\equiv g^k (\mathrm{mod}\ r), \\ S_2 &\equiv (D - aS_1)k^{-1} (\mathrm{mod}\ r-1), \\ S_3 &\equiv S_2^d (\mathrm{mod}\ N). \end{aligned}$$

The ordered triplet $S = (S_1, S_2, S_3)$ is the signer's signature on the document D.

The validity of S on D is ensured by the verifier in a two-step verification process. First, the verifier computes

$$V_1 \equiv S_3^e (\mathrm{mod}\ N),$$

and checks whether $V_1 \equiv S_2 (\mathrm{mod}\ r-1)$. This is the first step of validation of the signature S. Upon its validation, the verifier executes the second step of the verification process by computing

$$V_2 \equiv A^{S_1} S_1^{V_1} (\mathrm{mod}\ r),$$

and checks whether V_2 is $g^D (\mathrm{mod}\ r)$.

Thus, the validity of S on D is verified if

$$(V_1, V_2) = (S_2 (\mathrm{mod}\ r-1), g^D (\mathrm{mod}\ r)).$$

For better comprehension, we give a couple of numerical examples to explain the hybrid signature scheme, one with small values and one with reasonably larger values.

Example 1: The signer chooses the prime $r = 101$ and $g = 38$. The signer chooses $a = 53$ as the secret integer and calculates the public verification key $A \equiv g^a (\mathrm{mod}\ r) \equiv 72 (\mathrm{mod}\ 101)$. Furthermore, the signer selects $p = 113$,

$q = 131$ as two secret primes, and obtains $N = pq = 14803$ as the public modulus. The signer chooses the verification key $e = 999$ such that $\gcd(e, \phi(14803)) = 1$. Finally, the private signing key $d = 8599$ is evaluated as an inverse of the integer $e = 999$ modulo $\phi(14803)$. The signer signs the document $D = 60$, using a random number, say $k = 33$ as follows.

$$S_1 \equiv g^k (\mathrm{mod}\, r) \equiv 2 (\mathrm{mod}\, 101),$$
$$S_2 \equiv (D - aS_1)k^{-1} (\mathrm{mod}\, r - 1) \equiv 38 (\mathrm{mod}\, 100),$$
$$S_3 \equiv S_2^d (\mathrm{mod}\, N) \equiv 7018 (\mathrm{mod}\, 14803).$$

The signer publishes the signature $S = (S_1, S_2, S_3) = (2, 38, 7018)$ on the document $D = 60$. A verifier, first checks $V_1 \equiv S_3^e (\mathrm{mod}\, N) \equiv 38 (\mathrm{mod}\, 14803)$ and verifies $V_1 \equiv S_2 (\mathrm{mod}\, r - 1)$. Thereafter, the verifier computes $V_2 \equiv A^{S_1} S_1^{V_1} (\mathrm{mod}\, r) \equiv 95 (\mathrm{mod}\, 101)$ and verifies that V_2 is $g^D (\mathrm{mod}\, r) \equiv 95 (\mathrm{mod}\, 101)$.

Example 2: The signer chooses the prime $r = 1196089$ and $g = 981024$. The signer chooses $a = 459820$ as the secret integer and calculates the public verification key $A \equiv g^a (\mathrm{mod}\, r) \equiv 680626 (\mathrm{mod}\, 1196089)$. Furthermore, the signer selects $p = 291113$, $q = 297457$ as two secret primes, and obtains $N = pq = 86593599641$ as the public modulus. The signing party chooses the verification key $e = 986532481$ such that $\gcd(e, \phi(86593599641)) = 1$. Finally, the private signing key $d = 1761293185$ is evaluated as the inverse of $e = 986532481$ modulo $\phi(86593599641)$. The signer signs the document $D = 507321$, using a random integer, say $k = 710827$ as follows.

$$S_1 \equiv g^k (\mathrm{mod}\, r) \equiv 86234 (\mathrm{mod}\, 1196089),$$
$$S_2 \equiv (D - aS_1)k^{-1} (\mathrm{mod}\, r - 1) \equiv 179275 (\mathrm{mod}\, 1196088),$$
$$S_3 \equiv S_2^d (\mathrm{mod}\, N) \equiv 80563731559 (\mathrm{mod}\, 86593599641).$$

The signer publishes the signature

$$S = (S_1, S_2, S_3) = (86234, 179275, 80563731559)$$

on the document $D = 507321$. For verification, a verifier, first checks $V_1 \equiv S_3^e (\mathrm{mod}\, N) \equiv 179275 (\mathrm{mod}\, 86593599641)$ and verifies $V_1 \equiv S_2 (\mathrm{mod}\, r - 1)$. Thereafter, the verifier computes $V_2 \equiv A^{S_1} S_1^{V_1} (\mathrm{mod}\, r) \equiv 122491 \pmod{1196089}$ and verifies that V_2 is $g^D (\mathrm{mod}\, r) \equiv 122491 (\mathrm{mod}\, 1196089)$.

Next, we make few significant remarks about the random number k, how the verification process works, and why the prime r is required to be smaller than $\phi(N)$.

Remark 1: Note that the random number k chosen is an ephemeral integer. In other words, it is advisable to choose a new temporary number k each time a signer wants to put a signature on a document.

If this precaution is not taken, then in the case of signature scheme due to Elgamal, an adversary can possibly forge the signature on a given document, as explained below in Section 4 (see 4.1.1). However, in the same subsection, we

prove that it is not possible to mount such an attack in case of our proposed scheme.

Remark 2: In the first step of the verification process, the verifier obtains

$$V_1 \equiv S_3^e \equiv (S_2^d)^e \equiv S_2^{ed} \equiv S_2 (\text{mod } N)$$

since $ed \equiv 1(\text{mod } \phi(N))$. However, its validity is confirmed if

$$V_1 \equiv S_2 (\text{mod } r-1).$$

Observe that, initially S_2 is computed modulo $r-1$ and so $S_2 < r-1$. Since $r < \phi(N) < N$, we have $r-1 < N$, and so $V_1 \equiv S_2 (\text{mod } N)$ has the same value as $S_2 (\text{mod } r-1)$. That is, the integer S_2, being smaller than $r-1 < N$, remains unchanged on taking its value modulo N instead of modulo $r-1$.

Remark 3: The necessity of $r < \phi(N)$ ensures that $r-1 < N$ which is required due to the reason mentioned in the above remark. If the condition $r < \phi(N)$ is not satisfied, then the verification may fail, an example of which is given below.

Example 3: Suppose $r = 37$ and $g = 2$. The signer selects $a = 5$ and calculates $A \equiv g^a \equiv 32(\text{mod } 37)$. Furthermore, the signer selects primes $p = 3$ and $q = 5$, and calculates $N = pq = 15$. The signer then selects $e = 7$ coprime with $\phi(15)$ and determines $d = 7$ which is the inverse of $e = 7$ modulo $\phi(15)$. The signer signs the document $D = 8$, using $k = 35$ as follows.

$$\begin{aligned} S_1 &\equiv g^k (\text{mod } r) \equiv 19(\text{mod } 37), \\ S_2 &\equiv (D - aS_1)k^{-1} (\text{mod } r-1) \equiv 15(\text{mod } 36), \\ S_3 &\equiv S_2^d (\text{mod } N) \equiv 0(\text{mod } 15). \end{aligned}$$

The signer publishes the signature $S = (S_1, S_2, S_3) = (19,15,0)$ on the document $D = 8$. First, a verifier computes

$$V_1 \equiv S_3^e (\text{mod } N) \equiv 0(\text{mod } 15).$$

Note that $V_1 \not\equiv S_2 (\text{mod } r-1)$. Hence, the signature is not verified.

4. Security Analysis

The Elgamal and the RSA signature schemes are two widely popular and efficient signature schemes. However, they are prone to certain attacks and also bear a risk of forgery in some specific cases. In the current section, we prove that our proposed scheme is secure and unforgeable against various threats which are applicable to the Elgamal and the RSA signature schemes. The security and efficiency of our hybrid scheme is analyzed in the following two aspects.

(i) Security of the scheme against some attacks on the Elgamal signature scheme.

(ii) Security of the scheme against some attacks on the RSA signature scheme.

Some already known and efficient attacks applicable on the Elgamal signature include the same k-attack given by Elgamal (see [10, Attack 1, Note 2]), the existential attack (see [10, Attack 6]), the Bleichenbacher's attack (see [18, Theorem 1]), Khadir's attack (see [19, Theorem 3]), etc., whereas some of widely known and attacks applicable on the RSA digital signature scheme include the chosen message attack, the partial key exposure attack [20], the multiplicative attack, the Wiener's attack [21], the factoring attack, etc. In what follows, we establish that our scheme withstands the above listed attacks, and thereby proving our novel scheme to be an efficient improvisation of the above two aforementioned signature schemes.

4.1 *Attacks on the Elgamal signature*

In this subsection, we describe a few attacks that are successfully mounted on the Elgamal signature but fail to affect the scheme proposed by us.

4.1.1 Secure against the same-k attack

In the case of Elgamal signature, if the signer uses the same value of the randomly chosen integer k to sign multiple documents, then an adversary can possibly counterfeit a signature by recovering the value of the signer's secret signing key a. This process is given below by which is described in Elgamal [10], and we call it the *same k-attack*. Suppose, the signer utilizes the same value of k in signing two different documents, say D_1 and D_2, then, as shown in [10], the adversary can recover the value of a by solving the following simultaneous system of linear equations for a.

$$D_1 \equiv aS_1^1 + kS_2^1 (\mathrm{mod}\, r-1),$$
$$D_2 \equiv aS_1^2 + kS_2^2 (\mathrm{mod}\, r-1).$$

Hence, on possessing a, the forger can forge their signature on any document.

Unlike the Elgamal scheme, the hybrid signature scheme remains secure despite the value of the secret signing key a being compromised. The reason is as follows. Suppose, while signing, the utilization of the same k by the signer has allowed an adversary to determine the secret signing key a by solving the system of linear equations given above. Then, with the knowledge of a, it is possible for the adversary to choose a random k' and produce the two components S_1' and S_2' of his forged signature $S' = (S_1', S_2', S_3')$ on a document D as follows.

$$S_1' \equiv g^{k'} (\mathrm{mod}\, r),$$

and

$$S_2' \equiv (D - aS_1')(k')^{-1} (\mathrm{mod} r-1).$$

However, to forge the entire signature S', the adversary must be able to forge the third component S_3' of S' by computing

$$S_3' \equiv (S_2')^d (\mathrm{mod}\, N).$$

It is evident that, to be able to execute this, the adversary must possess the knowledge of d. However, as already known, obtaining d is essentially same as

factorizing $N = pq$, which is practically very difficult. This proves that, although it is desirable to use a different value of the ephemeral key k each time while signing a document, as stated in Remark 1, the proposed scheme is not vulnerable even if the same value of k is used.

4.1.2 Secure against existential attack

In case of the Elgamal scheme, knowing the signature on some document, a forger can generate a legitimate signature for the document by an attack described below as given in [10]. Assume that the Elgamal digital signature for a document D is $S = (S_1, S_2)$. If the forger can find three integers X, Y, Z such that $XS_1 - ZS_2$ and $r - 1$ are coprime, then the forger computes the following.

$$S_1' \equiv S_1^X g^Y A^Z (\text{mod } r),$$
$$S_2' \equiv \frac{S_2 S_1'}{XS_1 - ZS_2} (\text{mod } r - 1),$$
$$D' \equiv S_1' \frac{XD + YS_2}{XS_1 - ZS_2} (\text{mod } r - 1),$$

and claims the signature (S_1', S_2') on the document D'. It can be seen that (S_1', S_2') is a legitimate signature on the new document D', a proof of which is given in [10]. Therefore, the Elgamal digital signature scheme is insecure against such an existential attack.

Suppose a forger wants to execute this attack on the hybrid signature scheme. Assume that the forger has success in finding the triplet X, Y, Z such that $\gcd(XS_1 - ZS_2, r - 1) = 1$ for the signature $S = (S_1, S_2, S_3)$ on an already known document D, signed by the signer. Suppose that the forger has created his signature pair (S_1', S_2') for the document D' modulo $r - 1$ as above. Now, the third part of the signature S_3' of the hybrid scheme is still needed to be forged. But for forging $S_3' \equiv (S_2')^d (\text{mod } N)$, the knowledge of the secret signing key d is required, which is equivalent to factorizing $N = pq$ as argued in the above attack (see 4.1.1).

Thus, the difficulty of integer factorization refrains any forger in executing the aforementioned existential attack on the hybrid scheme. In other words, finding integers X, Y, Z that fulfill the above condition does not undermine the security of the proposed signature scheme.

4.1.3 Secure against Bleichenbacher's attack

In the Elgamal scheme, the public parameters r and g are provided by an agency or some trustworthy party or chosen by the signer themselves. However, these same public parameters also play a key role to ensure that the scheme is secure. By a special case of the result due to Bleichenbacher [17] in 1996, if the prime r chosen such that $r - 1 = bw$ for some integer w, where b is smooth, then the Elgamal signature scheme can be broken under certain conditions. Specifically, if a generator $\beta = cw$, where c is a positive integer less than b, and an integer t with $\beta^t \equiv g (\text{mod } r)$ are known, then for any document D, the signature pair (S_1, S_2) can be determined without knowing the private signing key a. This result emphasizes the fact that the security of Elgamal signatures

depends not only on the private signing key, but also on the arithmetic properties of the public parameters.

Coming to the case of our scheme, when the prime r and the primitive root g that satisfy the condition given in [17] are provided or chosen, it may be possible for a forger to forge the signature components S_1 and S_2 of the signature $S = (S_1, S_2, S_3)$ on any given document D. However, the computation of $S_3 \equiv S_2^d (\mathrm{mod}\ N)$ still relies on solving the integer factorization problem as mentioned in the two above mentioned attacks. This ensures our scheme remains secure and unforgeable against Bleichenbacher's attack, in contrast to its success for the Elgamal signature scheme.

4.1.4 Secure against Khadir's attack

An extension of the attack in [17] described above is given in [18], which is as follows. Suppose given a prime $r \equiv 1(\mathrm{mod}\ 4)$ and a generator g of $\mathbb{F}_r^*$, a forger finds a positive integer i with $\gcd(i, r-1) = 1$ such that $g^i (\mathrm{mod}\ r)$ is smooth, and divides $r-1$. Then, for any document D, the forger can forge a pair of signatures (S_1, S_2) without possessing the signer's signing key a of the Elgamal signature.

Consider the similar situation in our proposed scheme, where the prime r and the primitive root g fulfilling the aforementioned condition given in [18] are disclosed or exposed. Again, in this case, as mentioned in the above attacks, the hybrid signature scheme remains secure due to the computationally difficult problem of the integer factorization. That is, even if, for a document D, an attacker forges the values of S_1 and S_2 of the signature $S = (S_1, S_2, S_3)$ on D, they would still need to forge the valid value of the remaining one-third portion of the signature S, i.e., $S_3 \equiv S_2^d (\mathrm{mod}\ N)$, requiring a legitimate value of d which is as good as finding the factors of N. Hence, on the contrary to the Elgamal scheme, the Khadir's attack is ineffective on the proposed hybrid signature scheme.

4.2 *Modulus factorization attack*

One security feature of our new scheme that is common to all four attacks previously described is the forger's inability or difficulty in forging the final third $S_3 \equiv S_2^d (\mathrm{mod}\ N)$ of the signature $S = (S_1, S_2, S_3)$. Restating the rationale, this is due to the fact that forging S_3 would necessitate the determination of the value of d such that $ed \equiv 1(\mathrm{mod}\ \phi(N))$. This is essentially the same as factoring $N = pq$. Therefore, a significant question that arises here is what are the repercussions of factorization of the modulus N, by a forger, into a product of a pair of its prime factors? It turns out that the factorization of N merely doesn't impact on its security. The argument for which is established below.

4.2.1 Secure against the attack of modulus factorization

In the proposed hybrid signature scheme, suppose N is factored into its prime factors p and q. Then the signing key d can be easily obtained from the product $(p-1)(q-1)$ and the fact that d and e are inverses of each other

modulo the product. In order to forge the signature $S = (S_1, S_2, S_3)$ of a document D, a forger can now pass any value of S_2 to create the corresponding third part of the signature S_3 as

$$S_3 \equiv S_2^d (\text{mod } N).$$

As a consequence, there will be no problem in the first phase of the verification algorithm since

$$\begin{aligned} V_1 \equiv S_3^e (\text{mod } N) &\equiv S_2 (\text{mod } N) \\ &\equiv S_2 (\text{mod } r-1). \qquad (\because r < \phi(N)) \end{aligned}$$

However, the forger is not free to choose any value of S_2 as the values of S_1 and S_2 satisfy the following relation.

$$S_2 \equiv (D - aS_1)k^{-1} (\text{mod } r-1).$$

Note that, the forger can choose k' and compute S_1 as,

$$S_1 \equiv g^{k'} (\text{mod } r).$$

Now, based on this value S_1, the value of S_2 must be chosen so that the second phase verification goes through successfully. This would require the forger to find the value of S_2 such that

$$V_2 \equiv A^{S_1} S_1^{V_1} \equiv A^{S_1} S_1^{S_2} \equiv g^D (\text{mod } r).$$

Certainly, finding the value S_2 satisfying the above expression is nothing but solving the DLP modulo r. Consequently, our proposed scheme is resistant against factoring modulus attack.

4.3 *Attacks on the RSA signature*

The proposed scheme includes the idea of the RSA signature algorithm, described in Section 2, in creating the third portion of the signature triplet $S = (S_1, S_2, S_3)$ on a document D, i.e.,

$$S_3 \equiv S_2^d (\text{mod } N).$$

This necessitates consideration of the attacks applicable on the RSA scheme in case of our proposed hybrid scheme. In the remaining part of this section, we show that this is not the case. In other words, the hybrid scheme is secure against various attacks applicable on the RSA signature scheme. One such attack involving factorization of the modulus N is already discussed above, in which we concluded that the hybrid signature scheme is secure due to the difficulty of finding the discrete logarithm modulo the prime r. Other attacks which are effective on the RSA signature scheme but not on the proposed hybrid scheme are described below.

4.3.1 Resiliant against the chosen message attack

In one of the potential attacks on the RSA scheme, a forger chooses a random integer $1 < S < N$ as a pre-determined signature and computes the corresponding document D as $D \equiv S^e (\text{mod } N)$. Then the forger can claim that the document D bears the signature S. As described above, the verification of S

as a legitimate signature on D is easily justified since $S^e \equiv (D^d)^e \equiv D(\text{mod } N)$. This is how the forger is capaable to create a valid pair (D, S) of document-signature, despite not having the signer's private signing key d.

This is impossible for our proposed hybrid scheme contrasting to the RSA signature. Suppose a forger attempts to apply the same attack strategy to forge a signature $S = (S_1, S_2, S_3)$ on a desired document D in case of our scheme. For this purpose, the forger chooses any value of S_3 and computes

$$V_1 \equiv S_3^e(\text{mod } N) \equiv S_2(\text{mod } r - 1).$$

Now, the forger has the values of the signature components S_2 and S_3, for the document D. In order to complete the forgery of the signature S, the forger needs to find the value of $S_1 \equiv g^k(\text{mod } r)$ for some random integer k. It appears that the forger can generate S_1 by choosing a random integer k' and is successful in mounting this attack on our hybrid scheme. However, this is not possible as S_1 is dependent on S_2 and a by the following relation.

$$S_1 \equiv (D - kS_2)a^{-1}(\text{mod } r - 1).$$

Such a value of S_1, that satisfies the above congruence, is dependent on the value of a, and determining a is computationally very difficult, as mentioned earlier. This is one of the scenarios for finding such a value of S_1. In another instance, the forger may try to solve the verification equation

$$A^{S_1}S_1^{S_2} \equiv g^D(\text{mod } r),$$

for S_1 from the given values of A and S_2. As mentioned in [10], it is evident that computing S_1 is equivalent to solving the DLP modulo r. Hence, the signature scheme proposed here remains resiliant against the chosen message attack.

4.3.2 Secure against multiplicative attack

In an another threat on the RSA digital signature, if the signer signed two documents, say D_x and D_y, with the corresponding digital signatures, say S_x and S_y respectively, then a forger is capable to generate their signature on a new document without knowing the private signing key of the signing party as follows. Using S_x and S_y, the RSA signature S on a new document $D \equiv D_xD_y(\text{mod } N)$ is forged as

$$S \equiv S_xS_y(\text{mod } N).$$

Then the forger claims S to be a legitimate signature on the document D and the claim is valid since

$$S^e \equiv (S_xS_y)^e \equiv (D_x^dD_y^d)^e \equiv D_xD_y \equiv D(\text{mod } N).$$

Assume that the signer uses the hybrid scheme to sign two documents, D_x and D_y with their corresponding signatures $S_x = (S_{1x}, S_{2x}, S_{3x})$ and $S_y = (S_{1y}, S_{2y}, S_{3y})$ respectively. Suppose the document D is given by $D \equiv D_xD_y(\text{mod } r)$, and suppose that the forger wants to forge a signature $S = (S_1, S_2, S_3)$ on D. Then as in the previous case, a forger can compute the third part S_3 of the signature S as follows.

$$S_3 \equiv S_{3x}S_{3y} \equiv S_{2x}^d S_{2y}^d \equiv S_2^d \pmod N,$$

where $S_2 \equiv S_{2x}S_{2y} \pmod N$. To achieve a full forgery of S on D, the forger may choose $1 < k < r$ and compute $S_1 \equiv g^k \pmod r$. As discussed in the above attack, computing S_2 requires solving the discrete logarithm problem of finding a. Therefore, the multiplicative attack is not applicable on the proposed hybrid scheme.

4.3.3 Secure against Wiener's attack

Sometimes to accelerate the signing process, the signer may choose the secret signing key d to be small. In such a scenario, the value of d, in the RSA signature scheme, can be easily obtained by a forger. More precisely, as demonstrated in [20], a forger can recover d if its value is less than one-third of the fourth root of N. As already known in case of the RSA scheme, once the forger has the value of d, the signature S for any document D can be easily generated by computing $S \equiv D^d \pmod N$.

Suppose the forger wants to forge the signature $S = (S_1, S_2, S_3)$ on a given document D using the hybrid scheme, in the case when the signer has chosen $d < \frac{N^{1/4}}{3}$. Then, as in the Wiener's attack on the RSA scheme, the forger is able to determine the value of d as mentioned above. Using the known value of d, for any desired value of S_2, the forger can compute $S_3 \equiv S_2^d \pmod N$. To forge the initial component S_1 of S, the forger picks a value of k and generates $S_1 \equiv g^k \pmod r$. The only thing a forger needs is a suitable value of S_2 such that $S_2 \equiv (D - as_1)k^{-1} \pmod{r-1}$. However, as mentioned in the above cases, in order to determine S_2, the forger needs to solve the DLP for a. Thus, our scheme is not vulnerable to Wiener's attack.

4.3.4 Secure against partial-key exposure attack

Let (N, e) and d be the verification key and the secret signing key for the RSA digital signature, repectively. It is known, from [19], that a partial knowledge of d is enough for an attacker to recover the full signing key d. Furthermore, in the same article cited above, it is shown that, if e is less than $\sqrt{N}$, then access to only a one-fourth of the bits of d allows its complete recovery in polynomial time, thereby compromising the RSA scheme's security.

On the contrary to this, although the forger possesses the signing key d of our hybrid scheme, they cannot produce an authentic signature $S = (S_1, S_2, S_3)$ on the document D without the knowing the value of a as argued earlier. Consequently, the hybrid signature scheme remains secure against this attack.

Apart from the attacks on the RSA signature scheme described above, it is evident that, any attack mounted on it due to the revealed value of the secret signing key d would fail to mount on the hybrid scheme due to the difficulty of solving the underlying discrete logarithm problem.

5. Conclusion

This paper gives a new digital signature scheme that is a preferable improvisation of both the Elgamal and the RSA signature schemes using a hybrid approach. As discussed in Section 4, the signature scheme presented in this paper provides a double-layered security since it is based on two classical cryptographic problems, namely, the discrete logarithm problem (DLP) and the integer factorization problem (IFP) due to which it is resilient to all the attacks mounted on the above two signature schemes. As an outcome, the hybrid signature scheme is preferred over both of the standalone schemes previously mentioned.

Acknowledgment

The authors would like to express their sincere appreciation to Mr. Hitarth Rana for his insightful suggestions and valuable feedback, which contributed to the improvement of the manuscript. The second author expresses gratitude to the UGC for the financial support as Junior Research Fellowship.

Declaration

The authors do not have any conflict of interest.

References

[1] W. Diffie and M. E. Hellman, "New Directions in Cryptography," *IEEE Trans. Inf. Theory,* vol. 22, no. 6, pp. 644–654 (1976).

[2] R. L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Commun. ACM,* vol. 21, no. 2, pp. 120–126 (1978).

[3] D. Boneh, "Twenty Years of Attacks on the RSA Cryptosystem," *Notices Amer. Math. Soc*., vol. 46, no. 2, pp. 203–213 (1999).

[4] M. Mumtaz and L. Ping, "Forty years of attacks on the RSA cryptosystem: A brief survey," *J. Discrete Math. Sci. Cryptography*, vol. 22, no. 1, pp. 9–29 (2019).

[5] W. de Jonge and D. Chaum, "Some variations on RSA signatures & their security," in *Conference on the Theory and Application of Cryptographic Techniques*, pp. 49–59, Springer (1986).

[6] S. C. Gupta and M. Sanghi, "Matrix Modification of RSA Digital Signature Scheme," *J. Appl. Secur. Res.,* vol. 16, no. 1, pp. 63–70 (2021).

[7] R. A. Haraty, A. N. El-Kassar, and B. Shibaro, "A Comparative Study of RSA Based Digital Signature Algorithms," *J. Math. Stat.,* vol. 2, pp. 354–359 (Mar 2006).

[8] J. Mehta and H. Rana, "A multi-moduli approach towards enhancing the security of multiprime RSA," *J. Discrete Math. Sci. Cryptography, early access,* (2026).

[9] J. Mehta and H. Rana, "Safest-Value of the Number of Primes in RSA Modulus and an Improvised Generalized Multi-Moduli RSA," *Mathematics*, vol. 13, no. 10 (2025).

[10] T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *IEEE Trans. Inf. Theory,* vol. 31, pp. 469–472 (1985).

[11] I. M. Duursma and S. Park, "ElGamal type signature schemes for n-dimensional vector spaces." Cryptology ePrint Archive, Paper 2006/312 (2006).

[12] J. He and T. Kiesler, "Enhancing the security of ElGamal's signature scheme," *IEE Proc. Comput. Digit. Tech.*, vol. 141, no. 4, pp. 249–252 (1994).

[13] P. Horster, H. Petersen, and M. Michels, "Meta-ElGamal signature schemes," in *Proceedings of the 2nd ACM Conference on Computer and Communications Security*, pp. 96–107 (1994).

[14] L. Harn, "Public-key cryptosystem design based on factoring and discrete logarithms," *IEE Proc. Comput. Digit. Tech.*, vol. 141, pp. 193–195 (1994).

[15] N.-Y. Lee and T. Hwang, "Modified Harn signature scheme based on factorising and discrete logarithms," *IEE Proc. Comput. Digit. Tech.,* vol. 143, pp. 196–198 (1996).

[16] A. B. Nimbalkar, "The Digital Signature Schemes Based on Two Hard Problems: Factorization and Discrete Logarithm," in *Cyber Security,* pp. 493–498, Springer Singapore (2018).

[17] J. Mehta and P. Parmar, "A novel hybrid cryptosystem efficiently enhancing RSA security while ensuring forward secrecy," *Romanian Journal of Mathematics and Computer Science,* to be published.

[18] D. Bleichenbacher, "Generating ElGamal signatures without knowing the secret key," in *International Conference on the Theory and Applications of Cryptographic Techniques,* pp. 10–18, Springer (1996).

[19] O. Khadir, "Insecure primitive elements in an ElGamal signature protocol," *J. Discrete Math. Sci. Cryptography,* vol. 18, no. 3, pp. 237–245 (2015).

[20] D. Boneh, G. Durfee, and Y. Frankel, "An Attack on RSA Given a Small Fraction of the Private Key Bits," in *Advances in Cryptology—ASIACRYPT'98*, pp. 25–34, Springer (1998).

[21] M. J. Wiener, "Cryptanalysis of Short RSA Secret Exponents," *IEEE Trans. Inf. Theory*, vol. 36, no. 3, pp. 553–558 (1990).

Received February, 2026