On the (im)possibility of ElGamal blind signatures

Liliya Akhmetzyanova, Evgeny Alekseev, Alexandra Babueva and Stanislav Smyshlyaev

CryptoPro LLC, Russia
{lah, alekseev, babueva, svs}@cryptopro.ru

Abstract

In the current paper we investigate the possibility of designing secure blind signature scheme based on ElGamal signature equation. We define the generalized construction and analyze its security. We consider two types of schemes with the proposed construction, that cover all existing schemes. For schemes of the first type we provide generic ROS-style attack that violates unforgeability in the parallel setting. For schemes of the second type we prove that they do not provide either blindness, or unforgeability. As the result, we prove that all known ElGamal blind signature schemes are not secure. Moreover, these results show that the existence of secure ElGamal blind signature scheme is potentially possible only for small set of signature equations and requires the non-standard way of generating the first component of the signature.

Keywords: ElGamal signature scheme, blind signature scheme, ROS attack.

1 Introduction

Blind signature schemes are widely used in many applications that guarantee user anonymity, e.g. e-voting [8] and e-cash [4] systems. They allow the Requester to obtain a signature for an arbitrary message after interacting with the Signer in such a way that the Signer does not receive any information about either the message or the signature value (blindness property) and the Requester can compute only one single signature per interaction with the Signer (unforgeability property).

ElGamal signature scheme [6] is one of the most well-studied and widely-deployed signature schemes. Thus, development of blind signature scheme based on it is a relevant task. And sure enough, there exists a variety of blind signature schemes based on ElGamal signature equation [5, 10, 12, 13, 14, 17, 19, 20, 21, 24, 25]. However, the unforgeability of these schemes was not formally proven under some relevant assumptions.

At the same time, no attacks on these schemes were proposed. So, their security remains an open question. The only exception is the scheme introduced in [24], which additionally uses homomorphic encryption and non-interactive zero-knowledge proof (NIZK) for providing blindness. Its unforgeability was proven in [16] in the so-called algebraic bijective random oracle model. However, this scheme is not nearly as interesting for us since it uses the additional cryptographic mechanisms.

In the current paper we examine the possibility of constructing secure blind signature scheme based only on ElGamal signature equation. We introduce generalized ElGamal blind signature scheme called GenEG-BS. The signing protocol in this scheme is fixed only on the Signer side, where the ElGamal signature generation algorithm is performed for masked hash-value e generated on the Requester side in an arbitrary way. GenEG-BS construction covers all existing blind signature schemes based on ElGamal equation except for the scheme [24], in which the Signer side involves, in particular, verifying the NIZK proof.

We study the security of the GenEG-BS schemes. It turned out that the ROS attack [3], that breaks the security of blind Schnorr signature [18], can be adapted to break several GenEG-BS schemes. We provide the generic ROS-style attack on these schemes violating unforgeability in the parallel setting and the necessary condition for its applicability. Further we consider the schemes that are not vulnerable to the ROS-style attack. More specifically, we study the particular case of these schemes for which the way of generating the first component of the signature on the Requester side is fixed. We prove that such schemes do not provide either unforgeability, or blindness. As the consequence, we show that all existing GenEG-BS schemes [5, 10, 12, 13, 14, 17, 19, 20, 21, 25] are not secure. Moreover, we identify the form of ElGamal signature equation that can potentially lie in the heart of the secure GenEG-BS scheme. However, the construction of such scheme requires the radically new method of generating the first component of the signature.

2 Basic notations and definitions

By $\{0,1\}^*$ we denote the set of all bit strings of finite length including the empty string. If p is a prime number then the set $\mathbb{Z}_p$ is a finite field with characteristic p. We assume the canonic representation of the elements in $\mathbb{Z}_p$ as integers in the interval $[0 \dots p-1]$. Each non-zero element x in $\mathbb{Z}_p$ has an inverse $1/x$. We define $\mathbb{Z}_p^*$ as the set $\mathbb{Z}_p$ without zero element.

We denote the group of points of elliptic curve over the field $\mathbb{Z}_p$ as $\mathbb{G}$, the

order of the prime subgroup of $\mathbb{G}$ as q and elliptic curve point of order q as P. We denote by H the hash function that maps binary strings to elements from $\mathbb{Z}_q$ and assume that all field operations are performed modulo q.

If the value s is chosen from a set S uniformly at random, then we denote $s \xleftarrow{u} S$. If the variable x gets the value val then we denote $x \leftarrow val$. Similarly, if the variable x gets the value of the variable y then we denote $x \leftarrow y$. If the variable x gets the result of an algorithm A we denote $x \leftarrow A$.

The blind signature scheme is determined by three algorithms:

- $(sk, pk) \leftarrow \mathsf{KGen}$: a key generation algorithm that outputs a secret key sk and a public key pk;
- $(b, \sigma) \leftarrow \langle \mathsf{Signer}(sk), \mathsf{Requester}(pk, m) \rangle$: an interactive signing protocol that is run between a Signer with a secret key sk and a Requester with a public key pk and a message m; the Signer outputs $b = 1$ if the interaction completes successfully and $b = 0$ otherwise, while the Requester outputs a signature σ if it terminates correctly, and a fail indicator $\perp$ otherwise.
- $b \leftarrow \mathsf{Vf}(pk, m, \sigma)$: a (deterministic) verification algorithm that takes a public key pk, a message m, and a signature σ, and returns 1 if σ is valid on m under pk and 0 otherwise.

3 ElGamal blind signature scheme

Standard ElGamal signature scheme. The generalised ElGamal type signature scheme was introduced in [11] and further extended in [7]. A key generation algorithm involves picking random d uniformly from $\mathbb{Z}_q^*$ (secret signing key) and defining $Q = dP$ (public verifying key).

A signing algorithm for message m involves computing hash-value $e = H(m)$, picking random k uniformly from $\mathbb{Z}_q^*$ and defining r value as $kP.x \bmod q$. To ensure functionality and security, certain such values need to be excluded. The s value is determined from the ElGamal signature equation. According to [11], ElGamal signature equation is defined as follows:

$$G_d(r, e, s) \cdot d + G_k(r, e, s) \cdot k + G_0(r, e, s) = 0, \tag{1}$$

where G_d, G_k, G_0 are the functions $\mathbb{Z}_q^3 \to \mathbb{Z}_q^*$ that are affine by z or z^{-1} for all $z \in \{r, e, s\}$. If there exists a unique s such that the equation (1) is satisfied, then the signing algorithm returns (r, s) pair as the signature value, otherwise it returns the fail indicator.

For example, GOST [26] signature equation refers to ElGamal signature equations, where s is calculated as $ke + dr$, i.e. $G_d(r, e, s) = r, G_k(r, e, s) = e, G_0(r, e, s) = -s$. In [11] all possible ElGamal signature equations are listed (here the difference between $+z$ and $-z$ and the difference between z and z^{-1} is neglected, where $z \in \{r, e, s, k, d\}$):

1 :	$ed = rk + s$	7 :	$red = k + s$	13 :	$(r + e)d = k + s$
2 :	$ed = sk + r$	8 :	$d = rek + s$	14 :	$d = (r + e)k + s$
3 :	$rd = ek + s$	9 :	$sd = k + re$	15 :	$sd = k + (r + e)$
4 :	$rd = sk + e$	10 :	$d = sk + re$	16 :	$d = sk + (r + e)$
5 :	$sd = rk + e$	11 :	$red = sk + 1$	17 :	$(r + e)d = sk + 1$
6 :	$sd = ek + r$	12 :	$sd = rek + 1$	18 :	$sd = (r + e)k + 1$

Figure 1: ElGamal signature equations

In the current paper we rely on this list and do not consider its completeness.

The verify procedure for the message m and the signature (r, s) assumes verifying the equality

$$r = R.x \bmod q,$$

where $R = -\dfrac{1}{G_k(r, e, s)} \left(G_d(r, e, s) \cdot Q + G_0(r, e, s) \cdot P\right),\ e = H(m).$

ElGamal blind signature scheme. We define the general ElGamal blind signature scheme. A key generation algorithm is the same as in the standard ElGamal signature scheme.

The signing protocol is defined at Figure 2. The value e is always generated on the Requester side and forwarded to the Signer. The Signer performs ElGamal signature generating algorithm.

The verify procedure is the same as in the standard ElGamal signature scheme. We denote all blind signature schemes of this type as GenEG-BS schemes.

4 Security notions

Blind signature schemes should provide two security properties: unforgeability and blindness. In the current section we introduce the corresponding security notions by defining the threat and the adversary capabilities in each case. The formal definitions of these notions for two-round blind signature schemes are introduced, for example, in [9].

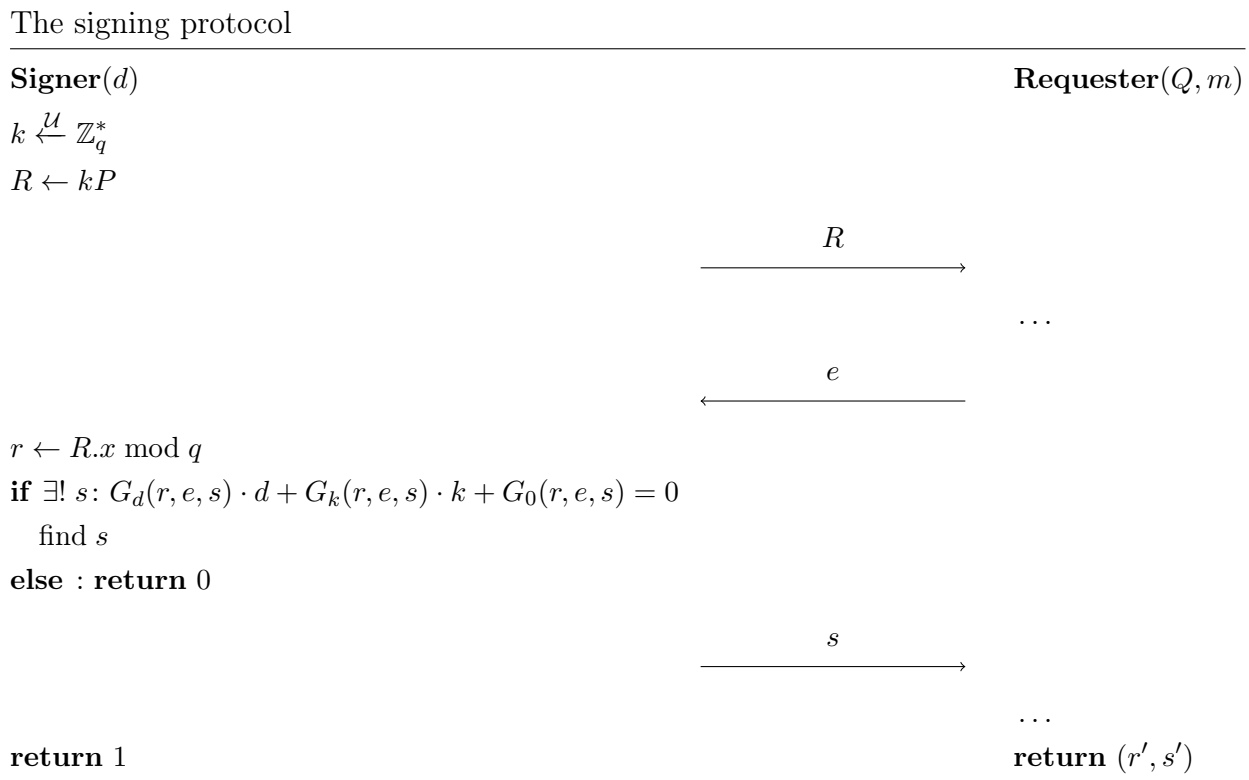

Figure 2: GenEG-BS scheme: the signing protocol

Unforgeability. An adversary acts as a malicious Requester and is powered to run the signing protocol with the Signer, scheduling and interleaving the sessions in any arbitrary way. In particular, it can open many parallel sessions with the Signer. It is assumed that the Signer behaves correctly (according to the protocol).

An adversary's task (threat), after interacting arbitrary many times with the Signer and l of these interactions were considered successful by the Signer, is to produce more than l valid (message, signature) pairs. The threat is considered strong if all messages should be distinct and weak if all (message, signature) pairs should be distinct.

In some cases the weak notion, in which the adversary is powered to open only the sequential sessions with the Signer, is considered.

Blindness. Informally, the blind signature scheme provides blindness if there is no way to link a (message, signature) pair to the certain execution of the signing protocol. In other words, the blindness is broken if the particular protocol execution for some fixed message leads to fixing the signature value in an unambiguous way or at least to significant narrowing the set of possible signature values.

Here an adversary acts as a malicious Signer and is powered to run the signing protocol with the Requester twice. It is assumed that the Requester behaves correctly (according to the protocol). After two successful interactions the Requester outputs two (message, signature) pairs simultaneously. If at least one of the interactions failed, the Requester outputs fail indicator.

An adversary's task (threat) is to link the transcription of the protocol to the corresponding (message, signature) pair with success probability significantly greater than 1/2. The unlinkability can be either computational, in which case we talk about computational blindness, or information-theoretical, we then talk about perfect blindness.

5 Security of the GenEG-BS schemes

We study the possibility of constructing secure ElGamal blind signature scheme GenEG-BS. Note that all existing GenEG-BS schemes were introduced without formal unforgeability proof, the blindness proof is presented only for some of them. Therefore, the security of these schemes remains an open question.

Well in our research, we identified two types of GenEG-BS schemes. They cover all existing schemes of such type [5, 10, 13, 14, 17, 19, 20, 21, 25]. We show that schemes of both types are not secure and do not provide either unforgeability, or blindness.

The starting point for distinguishing two types of the GenEG-BS schemes was the study of the possibility of applying the ROS attack [3] to such schemes. ROS (Random inhomogeneities in an Overdetermined, Solvable system of linear equations) problem was introduced by Schnorr [18] and was considered intractable for some time. However, later it was reduced to the $(l+1)$-sum problem, for which Wagner's [23] generalized birthday algorithm (with sub-exponential complexity) can be used. Finally, polynomial-time attack against ROS problem (ROS-attack) was proposed in 2020 in [3], that implies polynomial-time attack against blind Schnorr signature scheme in case an adversary is able to open $l \geqslant \lceil \log q \rceil$ parallel sessions with the Signer. In fact, not only the Schnorr scheme [15] was broken, but also the Okamoto-Schnorr scheme [15] and the partially blind Abe scheme [1]. Therefore, the question of the applicability of the attack to the GenEG-BS schemes is quite natural.

First type. It turned out that the modification of the ROS attack is applicable to a significant number of existing schemes [5, 12, 13, 14, 17, 19, 21, 25].

We provide the necessary condition for its applicability as the restrictions on the signature equation.

Condition 1: the signature equation can be represented in the following way:

$$k + Y_1(r, e) \cdot G_1(d) + Y_2(r, e, s) \cdot G_2(d) = 0, \tag{2}$$

where G_1 and G_2 functions are affine by d, Y_1 function significantly depends on e value and Y_2 function is linear fractional by s.

All GenEG-BS schemes with signature equation satisfying the Condition 1 will be called the schemes of Type I. For such schemes we construct generic ROS-style attack, violating unforgeability, thereby proving the following theorem.

Theorem 1. *If* GenEG-BS *scheme satisfies the Condition 1, then it does not provide unforgeability when the number of parallel sessions* $l \geqslant \lceil \log q \rceil$.

See Section 5.1 for attack description and discussion on Condition 1. Note that these attack is applicable in the standard model in which the adversary can open parallel sessions with the Signer. The security of such schemes relative to the weak adversary that can open only sequential sessions is the open question.

Second type. Consider ElGamal signature equations for which the Condition 1 is not satisfied. These are equations 2, 4, 10, 11, 16 at Figure 1, all of them have the following form:

$$sk = F_1(r, e)d + F_2(r, e) \tag{3}$$

or

$$s^{-1}k = F_1(r, e)d + F_2(r, e), \tag{4}$$

where F_1 and F_2 functions are affine functions by z or z^{-1} for all $z \in \{r, e\}$. Moreover, only one of the functions F_1 and F_2 significantly depends on r.

We obtain the result for the particular case of the GenEG-BS schemes based on these equations, in which the r' component of the signature is generated on the Requester side in the following way:

$$R' \leftarrow \alpha R + \beta Q + \gamma P, \quad r' \leftarrow R'.x \bmod q, \tag{5}$$

where each of the α, β, γ values (called blinding factors) are chosen uniformly from $\mathbb{Z}_q^*$ by the Requester or are equal to zero. We consider exactly uniform distribution on the blinding factor values, since other distributions seem not

to allow to reach perfect blindness. All existing schemes known to the authors assume exactly this way of generation of the r' component (regardless of the signature equation type). Thus, these results are important in terms of practice.

Finally, we call GenEG-BS scheme a scheme of Type II, if:

– the signature equation has the form (3) or (4);

– the r' component is generated according to (5).

The only known blind signature scheme of Type II is the scheme, introduced in [10]. However, the attack, violating blindness, on this scheme was presented in [2]. This attack leads us to consider the following condition.

Let (R, e, s) be the transcription of the signing protocol execution and $r = R.x \bmod q$. Let (r', s') be the signature value produced by the Requester for some message m with hash-value $e' = H(m)$ after that execution.

Condition 2: for all possible key pairs (d, Q) the equation $F_1(r, e) \cdot F_2(r', e') = F_1(r', e') \cdot F_2(r, e)$ holds with the overwhelming probability.

Here the probability space consists of all values representing random choices made by the Signer and the Requester randomized algorithms.

It turned out that this condition provides the criteria to link the given protocol transcription and the (message, signature) pair. We state the following theorem, see Section 5.2 for its proof.

Theorem 2. *If* GenEG-BS *scheme of Type II satisfies the Condition 2, then it does not provide blindness.*

To the best of our knowledge, there exist no GenEG-BS schemes of Type II, for which the Condition 2 is not satisfied. This observation allowed us to prove the following theorem, justifying the impossibility of constructing a secure blind signature scheme of this type.

Theorem 3. *If there exists* GenEG-BS *scheme of Type II that does not satisfy the Condition 2, then it does not provide unforgeability.*

The main idea of the proof is to show that the existence of such scheme leads either to the secret signing key recovering from the protocol transcription and the signature value obtained after the protocol execution, or to the ability to make valid signatures without secret key knowledge. See Section 5.3 for the full proof.

Summing up, we show that GenEG-BS schemes of Types I and II are not secure. Which means that if the secure GenEG-BS scheme exists, then

it is based on the equations (3) or (4) and assumes radically new way of generating the r' component, not according to (5).

5.1 ROS-style attack

According to the Condition 1, the signature equation can be represented as

$$k + Y_1(r, e) \cdot G_1(d) + Y_2(r, e, s) \cdot G_2(d) = 0,$$

where G_1 and G_2 functions are affine by d, Y_1 function significantly depends on e value and Y_2 function is linear fractional by s.

Verify procedure for message m and signature (r, s) assumes verifying the equality

$$r = R.x \bmod q,$$

where $R = -Y_1(r, e) \cdot G_1(d)P - Y_2(r, e, s) \cdot G_2(d)P,\ e = H(m)$. Note that $G_1(d)P$ and $G_2(d)P$ can be computed since G_1, G_2 functions are affine by d and $Q = dP$ is known.

The attack, presented below, allows an adversary to construct $(l + 1)$ valid (message, signature) pairs after $l \geqslant \lceil \log q \rceil$ successful interactions with the Signer. The adversary acts as follows:

1. Selects message $m_l \in \{0, 1\}^*$ for which a signature will be forged, let $e_l = H(m_l)$.

2. Opens l parallel sessions, querying the Signer, and receives corresponding points $R_0, \ldots, R_{l-1}$.

3. Calculates $r_i = R_i.x \bmod q, 0 \leqslant i \leqslant l - 1$.

4. Selects $m_i^0, m_i^1 \in \{0, 1\}^*, 0 \leqslant i \leqslant l - 1$, such that $r'_{i0} = Y_1(r_i, e_i^0) \neq Y_1(r_i, e_i^1) = r'_{i1}$, where $e_i^0 = H(m_i^0), e_i^1 = H(m_i^1)$.

5. Defines $(\rho_0, \rho_1, \ldots, \rho_l)$ as the vector of coefficients placed before x_i in the function $f : \mathbb{Z}_q^l \to \mathbb{Z}_q;\ f(x_0, \ldots, x_{l-1}) = \sum_{i=0}^{l-1} 2^i \underbrace{\frac{x_i - r'_{i0}}{r'_{i1} - r'_{i0}}}_{b'_i} = \sum_{i=0}^{l-1} \rho_i x_i + \rho_l$.

   Note that if $x_i = r'_{i0}$ then $b'_i = 0$, if $x_i = r'_{i1}$ then $b'_i = 1$.

6. Defines $R_l = \sum_{i=0}^{l-1} \rho_i R_i - \rho_l G_1(d)P$.

7. Defines $r_l = R_l.x \mod q$.

8. Defines $b_0, \ldots, b_{l-1}$ from the following equation: $Y_1(r_l, e_l) = \sum\limits_{i=0}^{l-1} 2^i b_i$.

9. Defines $r'_i = r'_{ib_i}, e_i = e_i^{b_i}, m_i = m_i^{b_i}, 0 \leqslant i \leqslant l-1$; therefore, according to step 5, $Y_1(r_l, e_l) = \sum\limits_{i=0}^{l-1} \rho_i r'_i + \rho_l = \sum\limits_{i=0}^{l-1} \rho_i Y_1(r_i, e_i) + \rho_l$.

10. Sends $e_0, \ldots, e_{l-1}$ values to the Signer in the corresponding sessions;

11. Obtains responses $s_0, \ldots, s_{l-1}$ such that:

$$R_i + Y_1(r_i, e_i) \cdot G_1(d)P + Y_2(r_i, e_i, s_i) \cdot G_2(d)P = 0,\ 0 \leqslant i \leqslant l-1.$$

12. Defines s_l in such a way that the following equality is satisfied:

$$\sum_{i=0}^{l-1} \rho_i Y_2(r_i, e_i, s_i) = Y_2(r_l, e_l, s_l).$$

    According to the Condition 1, Y_2 function is linear fractional by s. Thus, the above equation can be represented as $a_1 s_l + a_2 = 0$, where a_1, a_2 are the fixed values from $\mathbb{Z}_q$ that depend on $d, e_l, R_i, e_i^0, e_i^1, 0 \leqslant i \leqslant l-1$, values. If $a_1 \neq 0$, it is possible to efficiently find the s_l value such that the equation is satisfied. If $a_1 = 0$, the adversary returns to step 1. For all ElGamal equations listed at Figure 1, for any fixed signing key d and for any values $e_l, e_i^0, e_i^1, 0 \leqslant i \leqslant l-1$, selected by the adversary, the condition $a_1 = 0$ holds with the negligible probability over the random choice of R_i values by the Signer algorithm.

13. Outputs $\{m_i, (r_i, s_i)\}_{i=0}^{l}$.

Indeed, for $0 \leqslant i \leqslant l-1$ signature (r_i, s_i) is valid for m_i by attack construction, see step 11. Consider the case $i = l$. Summarize the equations obtained at step 11 with the corresponding coefficients:

$$\sum_{i=0}^{l-1} \rho_i R_i + \sum_{i=0}^{l-1} \rho_i Y_1(r_i, e_i) \cdot G_1(d)P + \sum_{i=0}^{l-1} \rho_i Y_2(r_i, e_i, s_i) \cdot G_2(d)P = 0.$$

Subtract and add the term $\rho_l G_1(d)P$ in the left part of the equation:

$$\underbrace{\sum_{i=0}^{l-1}\rho_i R_i - \rho_l G_1(d)P}_{=R_l} + \underbrace{\left(\sum_{i=0}^{l-1}\rho_i Y_1(r_i,e_i) + \rho_l\right)}_{=Y_1(r_l,e_l)} \cdot G_1(d)P + \\ + \underbrace{\sum_{i=0}^{l-1}\rho_i Y_2(r_i,e_i,s_i)}_{=Y_2(r_l,e_l,s_l)} \cdot G_2(d)P = 0.$$

According to the steps 6, 9, 12, this equation is equivalent to the following equation:

$$R_l = -Y_1(r_l,e_l)\cdot G_1(d)P - Y_2(r_l,e_l,s_l)\cdot G_2(d)P,$$

and $R_l.x \bmod q = r_l$ by construction at step 7. Hence, the signature (r_l, s_l) is valid for m_l.

The condition $l \geqslant \lceil \log q \rceil$ is needed to make possible the field element binary representation (see step 8) of length l.

The attack works due to the ability of varying $Y_1(r_i, e_i)$ values by message changing on step 4. This, in turn, is possible because of the summand, that does not depend on s value, in the equation (2). That explains the form of the Condition 1.

5.2 Attack on blindness

Consider GenEG-BS schemes of Type 2. Remind that for such schemes the Condition 2 is satisfied, i.e. the equation

$$F_1(r,e)\cdot F_2(r',e') = F_1(r',e')\cdot F_2(r,e) \tag{6}$$

holds with the overwhelming probability.

We claim that such schemes do not provide blindness. Namely, we show that for fixed protocol transcription and message there exists only the small set of valid signature values that could be produced during the given protocol execution. Indeed, if the protocol transcription (R, e, s) and message m are fixed, then the $r = R.x \bmod q$ and $e' = H(m)$ values are also fixed. The equation (6) is affine by r' since $F_1(r', e')$ and $F_2(r', e')$ functions are affine by r' and only one of them significantly depends on r'. Thus, r' component of the signature is defined unambiguously from equation (6). Note that α, β, γ are equal to zero or chosen uniformly at random from $\mathbb{Z}_q^*$. The probability

to choose α, β, γ during several protocol executions such that $(\alpha R + \beta Q + \gamma P).x \bmod q = r'$ is negligible. Therefore, with overwhelming probability there exists the unique signature that could be produced for message m during the given protocol transcription.

5.3 Unforgeability attack

Suppose, that there exists GenEG-BS scheme of Type II, for which the Condition 2 does not hold. It means that there exists an algorithm User, that works on the Requester side as follows. For arbitrary public key pk, outputed by key generation algorithm, arbitrary message m, point R and α, β, γ values, generated according to the distributions specified by the scheme, it outputs some value e. Then, after receiving the s value, generated according to (3) or (4), algorithm User outputs a valid signature (r', s') for message m with the overwhelming probability. Here the probability space consists of all values representing random choices made by the User randomized algorithm. Otherwise, it returns the fail indicator.

We construct an adversary $\mathcal{A}$ for such GenEG-BS scheme that violates unforgeability and uses algorithm User. It can interact with the Signer in the way described in Section 4. The adversary $\mathcal{A}$ knows the public key Q and acts as follows:

1. Selects message m and computes $e' = H(m)$.
2. Selects α, β, γ values uniformly from $\mathbb{Z}_q^*$ or defines them equal to zero (depending on the User algorithm).
3. Opens the session, querying the Signer, and receives point R as the response, computes $r = R.x \bmod q$.
4. Computes $r' = (\alpha R + \beta Q + \gamma P).x \bmod q$.
5. Runs algorithm User, giving it public key Q, point R, message m and α, β, γ values.
6. Receives e value from the User.
7. If $\gamma F_1(r, e) - \beta F_2(r, e) = 0$, goes to the next step.
   If $\gamma F_1(r, e) - \beta F_2(r, e) \neq 0$, computes
   $$s^* = (\gamma F_1(r, e) - \beta F_2(r, e))^{-1}(F_1(r, e)F_2(r', e') - F_2(r, e)F_1(r', e'))$$
   and checks if the signature is valid, computing $b = \mathsf{GenEG\text{-}BS.Vf}(Q, m, (r', s^*))$. If $b = 1$, the adversary $\mathcal{A}$ outputs $(m, (r', s^*))$ pair as the forgery and stops.

8. Sends e value to the Signer and forwards the obtained s value to the User.

9. Receives the signature (r', s') from the User. This signature must be valid for message m under public key Q, thus $s' \neq s^*$. If User outputs the fail indicator, the adversary $\mathcal{A}$ stops its work with the fail indicator.

10. If the equation (6) is not fulfilled, computes secret signing key d using the Algorithm 1 described below. After that, it computes valid signature (r'_1, s'_1) for arbitrary message $m_1 \neq m$, using the knowledge of d, and outputs two pairs $(m, (r', s'))$ and $(m_1, (r'_1, s'_1))$. If the equation (6) holds true, the adversary $\mathcal{A}$ stops its work with the fail indicator.

This attack is shown schematically in the Figure 3.

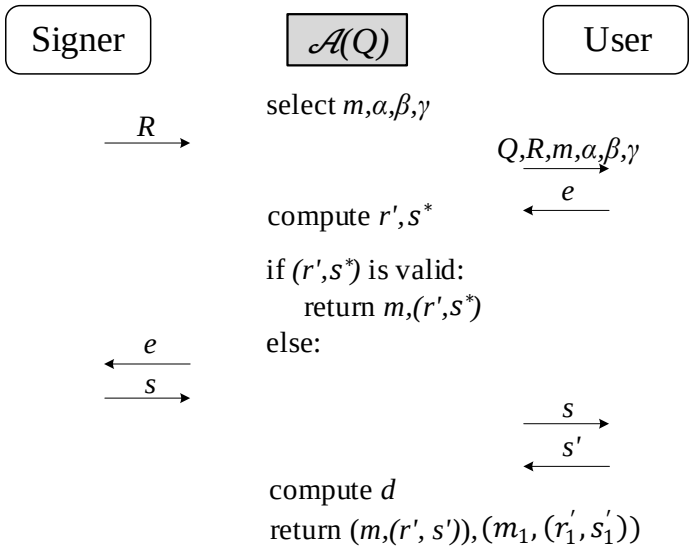

Figure 3: Attack on the GenEG-BS scheme of Type II

If the adversary $\mathcal{A}$ finishes the work on step 7, it completes successfully 0 interactions with the Signer and outputs 1 forgery. Otherwise, the adversary $\mathcal{A}$ makes 1 successful interaction with the Signer and outputs 2 forgeries, if User outputs a valid signature and the equation (6) holds true. According to the assumptions of Theorem 3, the probability of Condition 2 (and, thus, equation (6)) fulfillment and returning the fail indicator by User is negligible. Thus, the adversary $\mathcal{A}$ violates unforgeability with the overhelming probability.

Algorithm 1. We consider the case when GenEG-BS scheme of Type II is based on the equation (3), the case of the equation (4) is proved analogously.

Having a valid signature (r', s') for message m with hash-value e' and protocol transcription (R, e, s), the adversary $\mathcal{A}$ can construct the following system of linear equations with respect to unknown k and d:

$$\begin{cases} sk = F_1(r, e)d + F_2(r, e), \\ s'(\alpha k + \beta d + \gamma) = F_1(r', e')d + F_2(r', e'), \end{cases} \tag{7}$$

where $r = R.x \bmod q$. The first equation follows from the procedure of s value computation according to the equation (3). The second equation follows from the fact, that $r' = R'.x \bmod q = (\alpha R + \beta Q + \gamma P).x \bmod q$ and the signature (r', s') is valid, i.e. $s'R' = F_1(r', e')Q + F_2(r', e')P$.

Due to the construction of the scheme the system (7) must have a solution relative to k and d. According to the Kronecker-Capelli theorem [22], a system has a solution iff the rank of its coefficient matrix A is equal to the rank of its augmented matrix A'. We write out these matrices for system (7):

$$A = \begin{pmatrix} s & -F_1(r, e) \\ s'\alpha & s'\beta - F_1(r', e') \end{pmatrix},$$

$$A' = \begin{pmatrix} s & -F_1(r, e) & F_2(r, e) \\ s'\alpha & s'\beta - F_1(r', e') & F_2(r', e') - s'\gamma \end{pmatrix}.$$

Further we show that $\mathsf{rank}(A) = \mathsf{rank}(A') = 2$. Then the solution of the system is unique, and $\mathcal{A}$ finds secret key d by solving the system.

Suppose the opposite. Let $\mathsf{rank}(A) = \mathsf{rank}(A') \leqslant 1$. Then any two columns of matrix A', in particular, second and third columns, are linearly dependent. This means that the determinant of the square submatrix formed by these columns is equal to zero. We write out this condition:

$$\begin{aligned} 0 = \begin{vmatrix} -F_1(r, e) & F_2(r, e) \\ s'\beta - F_1(r', e') & F_2(r', e') - s'\gamma \end{vmatrix} = \\ = F_1(r, e)(s'\gamma - F_2(r', e')) - (s'\beta - F_1(r', e'))F_2(r, e) = \\ = s' \left(\gamma F_1(r, e) - \beta F_2(r, e)\right) - \left(F_1(r, e)F_2(r', e') - F_2(r, e)F_1(r', e')\right). \end{aligned}$$

Since the equation (6) is not fulfilled, $F_1(r, e)F_2(r', e') - F_2(r, e)F_1(r', e') \neq 0$. Then if $\gamma F_1(r, e) - \beta F_2(r, e) = 0$, the determinant can not be equal to zero and we come to the contradiction, from where $\mathsf{rank}(A) = \mathsf{rank}(A') = 2$. Let $\gamma F_1(r, e) - \beta F_2(r, e) \neq 0$, then we have the following condition on s':

$$s' = (\gamma F_1(r, e) - \beta F_2(r, e))^{-1}(F_1(r, e)F_2(r', e') - F_2(r, e)F_1(r', e')) = s^*.$$

However, $s' \neq s^*$ according to the adversary $\mathcal{A}$ construction (see step 9), so we come to the contradiction and $\mathsf{rank}(A) = \mathsf{rank}(A') = 2$.

6 Conclusion

The obtained results show that the development of secure ElGamal blind signature scheme is non-trivial task. There exist no such schemes to date. If such a scheme potentially exists, then either its Signer side differs from the one defined in the GenEG-BS scheme, or the method of generating the first component of the signature on the Requester side is entirely new and signature equation necessarily has the form (3) or (4).

Therefore, the direction for further research is the analysis of more general blind signature constructions based on ElGamal signature equations and providing either the attacks on them, or their formal security proof.

References

[1] Abe M., Okamoto T., "Provably secure partially blind signatures", *Advances in Cryptology — CRYPTO 2000*, Springer, Berlin, Heidelberg, 2000, 271–286.

[2] Babueva, A. A., Akhmetzyanova, L. R., Alekseev, E. K., Taraskin, O. G., "On Blindness of Several ElGamal-Type Blind Signatures", *Proceedings of the 6th International Conference "Convergent Cognitive Information Technologies"*, Convergent 2021. Communications in Computer and Information Science.

[3] Benhamouda, F., Lepoint, T., Loss, J., Orru, M., Raykova, M., "On the (in) security of ROS", *Advances in Cryptology – EUROCRYPT 2021*, Springer, Cham, 2021, 33–53.

[4] Chaum D., "Blind signatures for untraceable payments", *Advances in cryptology*, ed. Chaum D., Rivest R.L., Sherman A.T., Springer, Boston, MA, 1983, 199–203.

[5] Camenisch, J. L., Piveteau, J. M., Stadler, M. A., "Blind signatures based on the discrete logarithm problem", *LNCS*, Advances in Cryptology — EUROCRYPT'94, **950**, ed. De Santis A, Springer, Berlin, Heidelberg, 1994.

[6] ElGamal T., "A public key cryptosystem and a signature scheme based on discrete logarithms", *IEEE transactions on information theory*, **31**:4 (1985), 469–472.

[7] Fersch M., *The provable security of Elgamal-type signature schemes*, Diss. Bochum, Ruhr-Universität Bochum, 2018.

[8] Fujioka A., Okamoto T., Ohta K., "A practical secret voting scheme for large scale elections", *LNCS*, Advances in Cryptology — AUSCRYPT '92, **718**, Springer, Berlin, Heidelberg, 1992.

[9] Fuchsbauer G., Plouviez A., Seurin Y., "Blind Schnorr signatures and signed ElGamal encryption in the algebraic group model", *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Springer, Cham, 2020, 63–95.

[10] Gorbenko I., Yesina M., Ponomar V., "Anonymous electronic signature method", 2016 Third International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T), 2016, 47–50.

[11] Harn, L., Xu, Y., "Design of generalised ElGamal type digital signature schemes based on discrete logarithm", *Electronics Letters*, **30**:24 (1994), 2025–2026.

[12] Jena D., Panigrahy S. K., Acharya B., Jena S. K., "A Novel ECDLP-Based Blind Signature Scheme", National Conference on Information Security – Issues & Challenges, NCISIC 08, 2008.

[13] Khater, M. M., Al-Ahwal, A., Selim, M. M., Zayed, H. H., "New Blind Signature Scheme Based on Modified ElGamal Signature for Secure Electronic Voting", *International Journal of Scientific & Engineering Research*, **9**:3 (2018).

[14] Moldovyan, N. A. Blind Signature Protocols from Digital Signature Standards. In: Int. J. Netw. Secur., 13(1), pp. 22–30. 2011.

[15] Pointcheval D., Stern J., "Security arguments for digital signatures and blind signatures", *Journal of cryptology*, **13**:3 (2000), 361–396.

[16] Qin X., Cai C., Yuen T.H., "One-More Unforgeability of Blind ECDSA", *LNCS*, Computer Security – ESORICS 2021, **12973**, ed. Bertino E., Shulman H., Waidner M., Springer, Cham, 2021.

[17] Rostovtsev, A. G., "Blind signature on elliptic curve for e-cash.", *Information Security Problems. Computer Systems*, **1** (2000), 40–45, In Russian.

[18] Schnorr, C. P., "Security of blind discrete log signatures against interactive attacks", *LNCS*, International Conference on Information and Communications Security, **2229**, ed. Qing S., Okamoto T., Zhou J., Springer, Berlin, Heidelberg, 2001.

[19] Shen, V. R., Chung, Y. F., Chen, T. S., Lin, Y. A., "A blind signature based on discrete logarithm problem", *International Journal of Innovative Computing, Information and Control*, **7**:9 (2011), 5403–5416.

[20] Tan D. N., Nam H. N., Van H. N., Thi, L. T., Hieu M. N., "New blind mutisignature schemes based on signature standards", 2017 International Conference on Advanced Computing and Applications (ACOMP), 2017, 23–27.

[21] Tan, D. N., Nam, H. N., Hieu, M. N., Van, H. N., "New Blind Muti-signature Schemes based on ECDLP", *International Journal of Electrical and Computer Engineering*, **8**:2 (2018), 1074–1083.

[22] Vinberg E. B., *A course in algebra*, **56**, American Mathematical Soc, 2003.

[23] Wagner D., "A generalized birthday problem", *LNCS*, Advances in Cryptology — CRYPTO 2002, **2442**, Springer, Berlin, Heidelberg, 2002.

[24] Yi X., Lam K. Y., "A new blind ECDSA scheme for bitcoin transaction anonymity", *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*, 2019, 613–620.

[25] Zhang Y., He D., Zhang F., Huang X., Li D., "An efficient blind signature scheme based on SM2 signature algorithm", *LNCS*, International Conference on Information Security and Cryptology, **12612**, Springer, Cham, 2020, 368–384.

[26] *GOST R 34.10-2012. Information technology. Cryptographic data security. Signature and verification processes of electronic digital signature. National standard of the Russian Federation, STANDARTINFORM*, 2012, In Russian.