Computer Science Journal of Moldova, vol.**19**, no.1(55), 2011

Blind Collective Signature Protocol*

Nikolay A. Moldovyan

Abstract

Using the digital signature (DS) scheme specified by Belarusian DS standard there are designed the collective and blind collective DS protocols. Signature formation is performed simultaneously by all of the assigned signers, therefore the proposed protocols can be used also as protocols for simultaneous signing a contract. The proposed blind collective DS protocol represents a particular implementation of the blind multisignature schemes that is a novel type of the signature schemes. The proposed protocols are the first implementations of the multisignature schemes based on Belarusian signature standard.

Keywords. Digital signature, collective digital signature, discrete logarithm problem, blind signature, blind collective signature.

1 Introduction

The digital signatures (DS) are widely used in practical informatics to solve different problems connected with electronic documents authentication. There is proposed a variety of the DS protocols in the literature [11, 7]. Some type of the DS schemes, called multi-signature protocols, provide computing the single DS shared by several signers [1, 8]. A particular type of the multi-signature protocols, called collective DS, has been recently designed [9]. That variant of the multi-signature protocols is based on using the difficulty of finding large prime roots modulo 1024-bit prime p possessing the structure $p = Nk^z + 1$, where

* The work is supported by Russian Foundation for Basic Research grant # 10-07-90403-Ukr_a

$z \geq 2$, N is an even number, and k is a 160-bit prime. That protocol produces a fixed size collective DS for arbitrary number of signers, however the DS length is sufficiently large, actually, 1184 bits.

Using the general design of the collective DS scheme by [9] and DS algorithm specified by Belarusian DS standard, in this paper there is designed the collective DS protocol based on difficulty of finding discrete logarithm. The proposed protocol produces a 320-bit collective DS. Then the proposed collective DS protocol has been used to design the blind collective DS protocol that represents a new type of the multi-signature schemes. The blind collective signature protocol can be applied, for example, in the electronic voting systems and in the electronic money systems.

2 Collective signature protocol based on difficulty of discrete logarithm

2.1 Belarusian signature standard

Belarusian signature standard STB 1176.2−9 [6] is based on difficulty of finding the discrete logarithm in the finite group, order of which contains large prime factor q. The size of the factor q should be equal to $h \geq 160$ bits. The standard specifies the finite group as follows. Select prime p such that its size is $l \geq 1024$ bits. The group includes all numbers of the set $\{1, 2, \ldots, p-1\}$. The group operation is defined by the following formula:

$$a \circ b = abR^{-1} \bmod p,$$

where a and b are the group elements and $R = 2^{l+2}$. The standard specifies ten security levels corresponding to balanced pairs of the values h and l (see Table 1). The exponentiation operation is denoted as follows:

$$a^{(k)} = a \circ a \circ \ldots \circ a \bmod p \quad (k \text{ times}).$$

In the STB 1176.2−9 signature scheme the public key is computed using the following formula:

$$y = g^{(x)},$$

where g is the q order element of the group and x is the secret key $(1 < x < q)$. The signature generation procedure includes the following steps:

1. Generate a random number k $(1 < k < q)$ and compute $T = g^{(k)}$.
2. Concatenate the value T and message M to be signed: $M' = T\|M$.
3. Using the specified hash function F_H compute the hash value from M': $e = F_H(M') = F_H(T\|M)$, where $\|$ is the concatenation operation.
4. Compute the value $s = (k - xe) \bmod q$.

The pair of numbers (e, s) is the signature to message M. The signature verification is performed as follows:

1. If $1 < s < q$ and $0 < e < q$, then go to step 2. Otherwise the signature is false.
2. Compute value $T^* = g^{(s)} \circ y^{(e)}$.
3. Compute value $e^* = F_H(T^*\|M)$.
4. If $e^* = e$ the signature is valid, otherwise the signature is false.

2.2 Collective signature scheme

Suppose that m users should sign the given message M. The collective DS protocol works as follows:

1. Each of the users generates his individual random value k_i and computes $T_i = g^{(k_i)}$.

Table 1. Ten security levels of the STB 1176.2−9 standard

Security level	h, bits	l bits	Security level	h bits	l bits
1	143	638	6	208	1534
2	154	766	7	222	1790
3	175	1022	8	235	2046
4	182	1118	9	249	2334
5	195	1310	10	257	2462

2. It is computed the common randomization parameter as the product $T = T_1 \circ T_2 \circ \ldots \circ T_m$.

3. Using the common randomization parameter T and the specified hash function F_H it is computed the first element e of the collective DS: $e = F_H(T\|M)$.

4. Each of the users computes his share s_i in the second element of the collective DS

$$s_i = k_i - x_i e \bmod q, \quad i = 1, 2, .., m.$$

5. The second element s of the collective DS (r, s) is computed as follows $s = \sum_{i=1}^{m} s_i \bmod q$.

Size of the value s is equal to h, since it is computed modulo prime q. The total size of the signature (e, s) is $h + h'$, where h' is the bit size of the specified hash function.

The signature verification is performed exactly as it is described in Section 2.1 except the collective DS verification uses the collective public key computed as follows:

$$y = y_1 \circ y_2 \circ \ldots \circ y_m.$$

The presented collective DS protocol works correctly. Indeed,

$$T^* = y^{(e)} \circ g^{(s)} = y^{(e)} \circ g^{\left(\sum_{i=1}^{m}(k_i - x_i e)\right)} =$$

$$= y^{(e)} \circ g^{\left(\sum_{i=1}^{m} k_i\right)} \circ g^{\left(-e\sum_{i=1}^{m} x_i\right)} = y^{(e)} \circ g^{\left(\sum_{i=1}^{m} k_i\right)} \circ y^{(-e)} =$$

$$= g^{(k_1)} \circ g^{(k_2)} \circ \ldots \circ g^{(k_m)} = T_1 \circ T_2 \circ \ldots \circ T_m = T \Rightarrow$$

$$\Rightarrow E^* = F_H(M, R^*) = F_H(M, R) = E.$$

Since the equality $E^* = E$ holds, the collective signature produced with the protocol satisfies the verification procedure, i.e. the described collective signature protocol is correct.

2.3 Attacks on the collective DS protocol

The participants of the collective DS protocol have significantly more possibilities to attack the protocol than outsiders. They can try to forge a collective signature (the first type of the attacks) and to compute the secret key of one of the signers that shares a collective DS.

The first attack. Suppose it is given a message M and $m-1$ signers attempt to create a collective DS corresponding to m signers owning the collective public key $y = y' \circ y_m$, where $y' = \prod_{i=1}^{m-1} y_i$, i.e. $m-1$ users unite their efforts to generate a pair of numbers (e^*, s^*) such that $T^* = y^{(e^*)} \circ g^{(s^*)}$ and $e^* = F_H(T^* \| M)$. Suppose that they are able to do this, i.e. the collective forger (i.e. the considered $m-1$ signers) is able to calculate a valid signature (e^*, s^*) corresponding to collective public key $y = y_1 \circ y_2 \circ \ldots \circ y_m$. The collective DS satisfies the following relation

$$T^* = y^{(e^*)} \circ g^{(s^*)} = \left(y' y_m\right)^{(e^*)} g^{(s^*)} =$$

$$= y'^{(e^*)} \circ y_m^{(e^*)} \circ g^{(s^*)} = g^{\left(e^* \sum_i^{m-1} x_i\right)} \circ y_m^{(e^*)} \circ g^{(s^*)} =$$

$$= y_m^{(e^*)} \circ g^{\left(s^* + e^* \sum_i^{m-1} x_i\right)} \quad \Rightarrow \quad T^* = y_m^{(e^*)} \circ g^{(s^{**})},$$

where $s^{**} = s^* - E^* \sum_i^{m-1} x_i \bmod q$. The collective forgery have computed the signature (e^*, s^{**}) which is a valid signature (to message M) of the mth signer, since e^* is equal to $F_H(M \| R^*)$ and the pair of numbers (e^*, s^{**}) satisfies the verification procedure of the underlying DS scheme. Thus, any successful attack breaking the collective DS protocol also breaks the underlying DS standard. Since the STB 1176.2−9

standard specifies secure DS scheme the proposed protocol is also secure. Otherwise two or more persons would be able to forge a signature of the STB 1176.2−9 standard.

The second attack. Suppose that $m-1$ signers that share some collective DS (e, s) with the mth signer are attackers trying to calculate the secret key of the mth signer. The attackers know the values T_m and s_m generated by the mth signer. This values satisfy the equation $T_m = y_m^{(e)} g^{(s_m)}$, where the values T_m and e are out of the attackers' control, since the value $T_m = g^{(k_m)}$, where k_m is a random number generated by the mth signer, and e is the output of the hash function algorithm. It is supposed that the standard uses secure hash function, therefore the attackers are not able to select the value T producing some specially chosen value e. This means that, like in the case of underlying DS algorithm, computing the secret key requires solving the discrete logarithm problem, i.e. i) to find $k_m = \log T_m$ and then compute $x_m = e^{-1}(k_m - s_m) \bmod q$ or ii) to compute $x_m = \log y_m$.

3 Blind collective signature protocol based on Belarusian DS standard

3.1 Blind signatures

Blind signature schemes [2] represent a particular type of the cryptographic protocols that are especially interesting for application in the electronic money systems and in the electronic voting systems. For practical applications it is interesting to use the blind signature schemes based on the DS algorithms specified by the DS standards. Belarusian DS standard STB 1176.2−9 suites well to be used as the underlying DS scheme of the blind signature protocols.

The properties of the blind signatures are [11]:

i) the signer can't read the document during process of signature generation;

ii) the signer can't correlate the signed document with the act of signing.

Usually in the DS algorithms the signature is calculated using the hash function from the document to be signed, therefore the first property can be easily provided. It is sufficiently to present the hash function to the signer keeping the document secret. The problem of providing the second property is known as anonymity (or untraceability) problem. To solve this problem there are used specially designed DS algorithms. There are known blind signature schemes based on difficulty of the factorization problem [3] and on difficulty of finding the discrete logarithm [10].

To provide the anonymity of the signature there are used so called *blinding factors*. Prior to submit a hash function value (or message M) for signing, the user U computes the hash function value H and multiplies H (or M) by a random number (blinding factor). Then the user submits the blinded hash function value (or blinded document) for signing. The signer signs the blinded value H (or M) producing the blinded signature that is delivered to user U. The user divides out the blinding factor producing the valid signature to the original hash function value (or directly to the original document).

The blind DS protocol based on Belarusian signature standard can be constructed using the blinding factors y^{τ} and g^{ϵ} applied earlier to construct a blind signature scheme based on Schnorr's DS scheme [10, 12]. The designed protocol works as follows.

The blind signature generation procedure includes the following steps:

1. The signer generates a random number k $(1 < k < q)$, computes $T = g^{(k)}$, and sends the value T to the user U.

2. The user U generates random values τ and ϵ, computes $T' = Ty^{(\tau)}g^{(\epsilon)}$, $e' = F_H(T'\|M)$, where M is document to be signed, and $e = e' - \tau \bmod q$. Then the user sends the value e to the signer.

3. The signer computes the blinded signature $s = (k - xe) \bmod q$ and sends the value e to the user U.

4. The user U computes the signature $s' = s+\epsilon$. The pair of numbers (e', s') is the valid signature to the message M.

Correctness of the described blind signature protocol is proved as follows. Computing the value T^* (see signature verification procedure in subsection 2.1) gives

$$T^* = g^{(s')} \circ y^{(e')} = g^{(k-xe+\epsilon)} \circ y^{(e+\tau)} =$$

$$= g^{(k)} \circ g^{(-xe)} \circ g^{(\epsilon)} \circ y^{(e)} \circ y^{(\tau)} = g^{(k)} \circ y^{(-e)} \circ g^{(\epsilon)} \circ y^{(e)} \circ y^{(\tau)} =$$

$$= g^{(k)} \circ y^{(\tau)} \circ g^{(\epsilon)} = T \circ y^{(\tau)} \circ g^{(\epsilon)} = T' \Rightarrow$$

$$\Rightarrow e^* = F_H(T^* \| M) = F_H(T' \| M) = e'.$$

Thus, the signature (e', s') satisfies the equations of the STB 1176.2−9 standard verification procedure.

3.2 Blind collective signature

Belarusian standard suits also to be used as underlying DS scheme of the blind collective DS scheme. Suppose some user U is intended to get a collective DS (corresponding to message M) of some set of m signers using a blind signature generation procedure. To solve this problem the user can apply the following protocol:

1. Each signer generates a random value $k_i < q$ and computes $T_i = g^{(k_i)}$, and presents the value T_i to each of the signers.

2. It is computed a common randomization parameter R as the product $T = T_1 \circ T_2 \circ \ldots \circ T_m$.

3. The value T is send to the user U.

4. The user U generates random values $\tau < q$ and $\epsilon < q$ and computes the values $T' = Ty^{(\tau)}g^{(\epsilon)}$ and $e' = F_H(T' \| M)$. The value e' is the first element of the collective DS.

5. The user U calculates the value $e = e' - \tau \bmod q$ and presents the value e to the signers.

6. Each signer, using his individual value k_i and his secret key x_i, computes his share in the blind collective DS: $s_i = k_i - x_i e \bmod q$.

7. It is computed the second part s of the blind collective DS: $s = \sum_{i=1}^{m} s_i \bmod q$.

8. The user U computes the second parameter of the collective DS: $s' = s + \epsilon \bmod q$.

The signature verification procedure is exactly the same as described in the case of collective DS based on Belarusian standard (see subsection 2.2). The signature (e', s') is a valid collective DS corresponding to the message M. Indeed, using the collective public key

$$y = y_1 \circ y_2 \circ \ldots \circ y_m = g^{(\sum_{i=1}^{m} x_i)}$$

we get

$$T^* = y^{(e')} \circ g^{(s')} = y^{(e+\tau)} \circ g^{(s+\epsilon)} = y^{(e)} \circ y^{\tau} \circ g^{(s)} \circ g^{(\epsilon)} =$$

$$= g^{(e\sum_{i=1}^{m} x_i)} \circ y^{(\tau)} \circ g^{(\sum_{i=1}^{m}(k_i - x_i e))} \circ g^{(\epsilon)} = g^{(\sum_{i=1}^{m} k_i)} \circ y^{(\tau)} g^{(\epsilon)} =$$

$$= T \circ y^{(\tau)} \circ g^{(\epsilon)} = T' \Rightarrow e^* = F_H(T^* \| M) = F_H(T' \| M) = e'.$$

Thus, the protocol yields a valid collective DS (e', s') that is known to the user U and unknown to each of the signers. The protocol provides anonymity of the user in the case when the message M and collective signature (e', s') will be presented to the signers. Anonymity means that the signers are not able to correlate the disclosed signature with only one act of the blind signing, if the signers have participated in two or more procedures of blind signing. Indeed, suppose the signers save in a data base all triples (e, s, T) that are produced while performing the protocol.

Accordingly to the blind collective DS protocol the elements of each triple satisfy the expression:

$$T = y^{(e)} \circ g^{(s)}. \tag{1}$$

The signature (e', s') satisfies the expression:

$$T' = y^{(e')} \circ g^{(s')}. \tag{2}$$

From formulas (1) and (2) we get

$$T' \circ T^{-1} = y^{(e'-e)} \circ g^{(s'-s)} \Rightarrow T' = Ty^{(\tau)} \circ g^{(\epsilon)},$$

where $\tau = e' - e \bmod q$ and $\epsilon = s' - s \bmod q$. Since the values τ and ϵ are generated at random while performing the protocol, each of the triples has equal rights to be associated with the given disclosed signature.

3.3 Application as a protocol for simultaneous signing a contract

Due to the fact, that individual shares of the collective DS formed with the protocols described in subsections 2.2 and 3.2 are valid only in the frame of the given set of m signers, the mentioned protocols can be used to solve efficiently the problem of simultaneous signing a contract. The collective signature protocols solve the problem of signing simultaneously a contract being free of any trusted party. A scenario of practical application of the blind simultaneous signing some electronic messages can be attributed to the electronic money systems in which the electronic banknotes are issued by several banks.

4 Conclusion

Belarusian DS standard is recommended for practical application in information technologies connected with exchange and processing electronic documents accompanied by the usual-type digital signatures. The results of this paper show that the signature generation and signature verification procedures specified by Belarusian DS standard can be additionally used as underlying algorithms in the following protocols:

i) blind signature,

ii) collective signature;

iii) blind collective signature.

Besides, the collective DS protocols can be efficiently used as protocols for signing simultaneously a contract.

It is interesting to study possibility to implement such protocols using other official DS standards. Our preliminary investigation of this problem has shown that Ukrainian and Russian [4] DS standards provide such possibility, however American signature standards DSA and ECDSA [5] do not suite to this purpose. More detailed investigation of the proposed problem represents a subject of independent research.

References

[1] Boldyreva A., *Efficient Threshold Signature, Multisignature and Blind Signature Shemes Based on the Gap-Diffi-Hellman-Group Signature Sheme*, Springer-Verlag Lecture Notes in Computer Science, vol. 2139, pp. 31–46, 2003.

[2] Chaum D., *Blind Signature Systems*, U.S. Patent # 4,759,063. 19 July 1988.

[3] Chaum D., *Security without identification: Transaction systems to make big brother obsolete*, Communications of the AMS, vol. 28, no 10, pp. 1030–1044, 1985.

[4] GOST R 34.10-2001. Russian Federation Standard. Information Technology. Cryptographic data Security. Produce and check procedures of Electronic Digital Signature. Government Committee of the Russia for Standards, 2001 (in Russian).

[5] International Standard ISO/IEC 14888-3:2006(E). Information technology – Security techniques – Digital Signatures with appendix – Part 3: Discrete logarithm based mechanisms.

[6] Kharin Yu.S., Bernik V.I., Matveev G.V., Aguievich S.V. *Mathematic and computer foundations of cryptology*, Novoe znanie, Minsk, 2003. 381 p. (in Russian).

[7] Menezes A.J., Van Oorschot P.C., and Vanstone S.A., *Handbook of Applied Cryptography*, CRC Press, Boca Raton, FL, 1997. 780 p.

[8] Min-Shiang Hwang and Cheng-Chi Lee, *Research Issues and Challenges for Multiple Digital Signatures*, International Journal of Network Security, vol. 1. no 1, pp. 1–7, 2005.

[9] Moldovyan N.A., *Digital Signature Scheme Based on a New Hard Problem*, Computer Science Journal of Moldova, vol. 16, no 2(47), pp. 163–182, 2008.

[10] Pointcheval D. and Stern J., *Security Arguments for Digital Signatures and Blind Signatures*, Journal of Cryptology, vol. 13, p. 361–396, 2000.

[11] Schneier B., *Applied Cryptography*, Second Edition, John Wiley & Sons, Inc. New York, 1996. 758 p.

[12] C.P. Schnorr, *Efficient signature generation by smart cards*, Journal of Cryptology, vol. 4, pp. 161–174, 1991.

Received September 2, 2009

Nikolay A. Moldovyan

St. Petersburg Institute for Informatics and Automation of
Russian Academy of Sciences
14 Liniya, 39, St. Petersburg 199178, Russia
E–mail: *nmold@mail.ru*