Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-018-0791-6

ORIGINAL RESEARCH

Pseudo-probabilistic block ciphers and their randomization

Moldovyan Nikolay Andreevich[1] · Moldovyan Alexander Andreevich[2] · Tam Nguyen Duc[3] · Hai Nguyen Nam[3] · Manh Cong Tran[4] · Minh Nguyen Hieu[3]

Received: 26 December 2017 / Accepted: 30 March 2018

Abstract
There is considered implementation of the plan-ahead share-key deniable encryption algorithms that produce the cryptogram that satisfy criterion of the computational indistinguishability from probabilistic encryption of the fake message. This paper introduces a general design of the pseudo-probabilistic block ciphers. The proposed method includes encryption of the secret message block and the fake message block followed by a transformation procedure mapping the pair of intermediate ciphertext blocks into a single block of the output ciphertext. The transformation procedure is implemented in the following two variants: (1) simultaneous encryption of the intermediate ciphertext blocks and (2) solving the system of two linear congruencies. The second variant provides natural possibility to construct pseudo-probabilistic block ciphers in which recovering fake or secret message is performed using the same single decryption algorithm. To provide higher security there are proposed randomized pseudo-probabilistic ciphers. There are also considered designs with different size of the input data blocks corresponding to fake and secret messages.

Keywords Block ciphers · Plan-ahead deniable encryption · Shared-key deniable encryption · Pseudo-probabilistic cipher · Randomization

1 Introduction

The notions of *public-key deniable encryption and of shared-key deniable encryption* were introduced by Canetti et al (1997). These important cryptographic primitives are applied in cryptographic protocols to resist coercive attacks. In the concept of deniable encryption there are considered schemes such as sender-deniable, receiver-deniable, and sender and receiver-deniable (bi-deniable) in which coercive adversary attacks the party sending message, the party receiving message, and the both parties, correspondingly. In the model of the coercive attack it is supposed that coercive adversary has power to force a party or the both parties simultaneously to open the cryptogram (ciphertext) after it has been sent.

Paper of Canetti et al (1997) initiated a lot of investigations on developing secure and efficient methods for public-key deniable encryption by O'Neill et al (2011) in which no pre-shared information is used. Some of papers propose public-key deniable encryption combined with sharing secrete key (the sender and the receiver initially share a common secret key) and plan-ahead encryption (the fake message is selected at the stage of encryption) (Moldovyan et al

✉ Minh Nguyen Hieu
hieuminhmta@gmail.com

Moldovyan Nikolay Andreevich
nmold@mail.ru

Moldovyan Alexander Andreevich
ma@mail.ru

Tam Nguyen Duc
nguyenductamkma@gmail.com

Hai Nguyen Nam
nnhaivn61@gmail.com

Manh Cong Tran
manhtc@gmail.com

[1] St. Petersburg Institute for Informatics and Automation, Russian Academy of Sciences, 199178 St. Petersburg, Russia

[2] ITMO University, 197101 St. Petersburg, Russia

[3] Academy of Cryptography Techniques, Hanoi, Vietnam

[4] Le Quy Don Technical University, Hanoi, Vietnam

Published online: 24 April 2018

2017; Dürmuth and Freeman 2011; Barakat 2014). Detailed attention of the researchers to this direction in the area of deniable encryption is explained by the applicability of the public key deniable encryption to prevent vote buying in the internet-voting systems by Meng (2009) and to provide secure multiparty computations by Ishai et al (2011).

Practical applications of the plan-ahead shared-key deniable encryption can be attributed to the case of the information protection against unauthorized access in computer and communication systems in the case of coercive attacks. As it is set in Canetti et al (1997) for some models of such attacks plan-ahead shared-key deniability is trivially solved: use different keys, and construct the ciphertext as concatenation of encryption of all messages, where the ith message is encrypted using the ith key.

The present paper considers the coercive-attack model against which this trivial construction is not applicable. To resist the proposed coercive attack, the paper proposes the plan-ahead shared-key deniable encryption methods producing cryptogram that is computationally indistinguishable from the ciphertext produced by some probabilistic cipher. The paper introduces design of pseudo-probabilistic block ciphers that satisfy the last criterion.

The organization of the paper is as follows. Section 2 describes the model of the coercive attack, notion of pseudo-probabilistic encryption, and criteria used for designing pseudo-probabilistic block ciphers. Section 3 proposes a simple method for pseudo-probabilistic block encryption. Section 4 introduces pseudo-probabilistic encryption algorithms satisfying an additional criterion of using the same single decryption algorithm for disclosing both the secret and the fake message. The probabilistic encryption algorithms associated with the pseudo-probabilistic ones are presented in Sect. 5. The randomized pseudo-probabilistic ciphers are introduced in Sect. 6 in which there are also considered cryptoschemes with different size of data blocks of fake and secret messages. Section 7 concludes the paper.

2 Model of adversary and notion of pseudo-probabilistic encryption

It is assumed that after ciphertext has been sent the adversary has possibility to force both the sender and the receiver to open the following:

1. The plaintext corresponding to the ciphertext;
2. Encryption and decryption algorithms;
3. The encryption key with which encryption of the opened message yields all bits of the ciphertext.

Thus, in the considered model of the coercive attack the sender and the receiver are coerced to open parameters and algorithm of the ciphering procedure with which each bit of the sent ciphertext has been produced depending on the opened message (plaintext).

Security against the described attack can be provided using the symmetric deniable encryption (SDE) algorithm that produces the ciphertext like cryptogram produced as result of probabilistic encryption of the fake message with fake key. This idea leads to an encryption method that can be called pseudo-probabilistic encryption. Correspondingly, the ciphers performing pseudo-probabilistic encryption can be called pseudo-probabilistic ciphers (PPC). Let us consider some details of the notion of pseudo-probabilistic encryption in comparison with probabilistic encryption.

Probabilistic ciphering of some message M includes mixing random data with the message, therefore for some given encryption key K the output ciphertext C depends on random data. One of possible schemes of the probabilistic encryption can be described as adding some random value R having the size $|R|$ to the message M and transforming the value $M||R$ as follows:

$$C = \overline{E}_K(M) = E_K(R||M), \tag{1}$$

where E is some deterministic encryption function. It is evident that $C \in \{C_0, C_1, \ldots, C_N\}; N = 2^{|R|} - 1$. In other words, while using the key K the message M can be transformed with the probabilistic cipher $\overline{E}$ into one of $2^{|R|}$ potentially possible output ciphertexts. Suppose, using two different independent keys K and Q, two arbitrary messages M and T are encrypted simultaneously with encryption algorithm $E_{K,Q}$ and some single ciphertext C is produced as the result of the encryption process. Besides, each of the messages M and T can be computed from the ciphertext C. If one can indicate a probabilistic encryption algorithm $\overline{E}$ that can potentially encrypt the message M into the ciphertext C, then the cipher $E_{K,Q}$ is called *pseudo-probabilistic*.

If using the source message M, the key K, and ciphertext C it is computationally infeasible to prove that some other message can be recovered from C while using some other key Q and some decryption algorithm, then the cipher E is called *computationally indistinguishable from probabilistic cipher*.

For constructing symmetric PPC we have used the following design criteria:

1. Symmetric deniable encryption should be performed as simultaneous encryption of two messages, secret one and fake one, using secret and fake keys (which are shared by sender and receiver);
2. A probabilistic encryption algorithm should be associated with the SDE algorithm;
3. The associated probabilistic encryption algorithm should transform the fake message with the fake key into the same ciphertext that is produced by the SDE algorithm;

4. Using the fixed-size shared keys should provide performing secure SDE of messages having arbitrary length.

The use of PPC is attractive to provide security of the communication session to coercive attacks, since the parties of secure communication protocol can chart plausible that they use the probabilistic encryption to get higher resistance to potential cryptographic attacks. Indeed, mixing the encrypted data with random data makes cryptanalysis more difficult. Next section describes a method for implementing pseudo-probabilistic block ciphering on the base of deterministic block encryption functions.

3 Simple method for pseudo-probabilistic block encryption

In the well known method (Moldovyan and Moldovyan 2006, 2007) for probabilistic block ciphering it is used a b-bit encryption function E and the encrypted message is divided into v-bit data blocks ($v < b$). To transform a plaintext block M it is generated a u-bit random block R ($u = b - v$) followed by composing b-bit input data block $B = R||M$, where the sign || denotes the concatenation operation of two binary vectors, R and M, and computing the ciphertext block $C = E_K(B)$, where K is the encryption key. Rationality of practical application of the probabilistic encryption relates to the following items:

1. It provides more security against hypothetic attacks based on using back doors in the used block ciphers.
2. It potentially prevents attacks using some unforeseen vulnerabilities of the used block encryption algorithm.

One should note that in real encryption devices the used random number generator (RNG) have to be imbedded as an internal part, like the electronic circuite implementing the block encryption algorithm E. Thus, increasing the security is provided only in the case when potential adversary is not able to modify the RNG or its output.

With using different values of the b/v ratio, for some given encryption function E one can select required trade-off between security and encryption speed. The greater this ratio, the greater the contribution to the security level and the lower the data ciphering speed. The last can be roughly estimated with the formula $s = s_0(b - u)/b$, where s_0 is the encryption speed provided by the algorithm E. General scheme for probabilistic block encryption is illustrated in Fig. 1. The scheme for probabilistic block encryption can be easily transformed into a scheme for pseudo-probabilistic block encryption that can be used to encrypt simultaneously two independent messages, fake and secret ones,

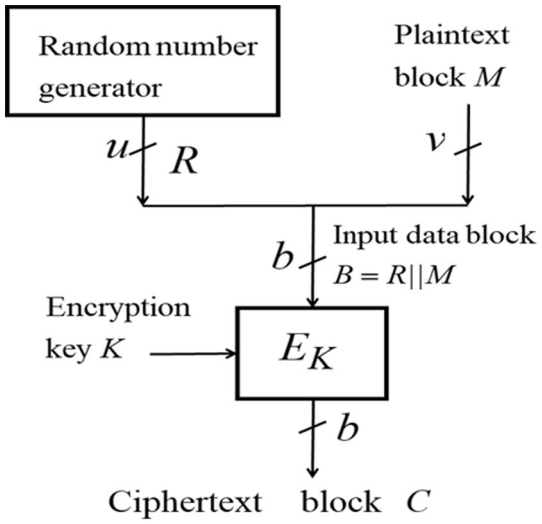

Fig. 1 Generalized scheme of probabilistic block encryption

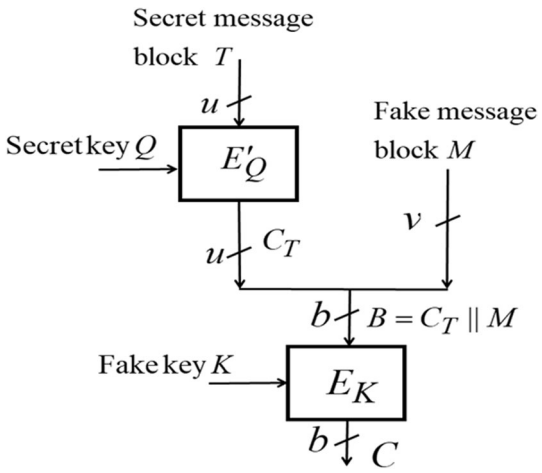

Fig. 2 Generalized scheme of pseudo-probabilistic block encryption

with using two different keys K and Q, correspondingly. For such purpose one can replace the RNG by some block encryption function E' with u-bit input data block. Instead of generating a v-bit random number R it is encrypted the v-bit data block T of secret message (see Fig. 2). While using secure block encryption algorithm E' to transform the data block T with the key Q, the produced intermediate ciphertext block $C_T = E'_Q(T)$ is computationally indistinguishable from uniformly random v-bit binary vector. Then the block C_T is concatenated with the fake-message block M and transformed into the output ciphertext block:

$$C = E_K(C_T||M). \tag{2}$$

When being coerced the sender and receiver of the message can open the fake message M and the fake key K and declare

about their using probabilistic block encryption algorithm. Thus, at time of the coercive attack the parties of secure communication session have possibility to cheat plausible, i.e. the proposed pseudo-probabilistic encryption method provides deniability. The book Moldovyan and Moldovyan (2006) presents several methods for probabilistic encryption with some deterministic block encryption function. For each of that methods one can propose respective pseudo-probabilistic block encryption scheme that is secure to ordinary coercive attacks. Such pseudo-probabilistic encryption schemes relates to the plan-ahead deniable encryption algorithms and protocols. They provide bi-deniability until the coercer has no possibility to check the decryption time required to disclose the secret message. If he has such possibility, then he will be able to establish that decryption time of the fake message is less than decryption time of the secret message. Besides, one can suppose that in some cases the coercer will have possibility to compare the execution codes of the procedures used to decrypt fake and secret messages.

To provide deniability at time of attacks performed by a coercer having the mentioned possibilities one can propose the following additional design criterion: *the fake and the secret messages are to be disclosed from the cryptogram with the same decryption algorithm.* This criterion can be fulfilled with insetting additional transformation of the fake-message block M with the block encryption function E' and setting values $v = u = b/2$ as it is shown in Fig. 3, where $e = K \; mod \; 2$ and the operational box $Transp^{(e)}$ performs the controlled transposition of two u-bit data blocks C_T and C_M: if $e = 1$, then $Transp^{(e)}(C_T||C_M) = C_M||C_T$; if $e = 0$, then $Transp^{(e)}(C_T||C_M) = C_T||C_M$. It is assumed that the keys K and Q satisfy condition $(K \; mod \; 2) \oplus (Q \; mod \; 2) = 1$ (the keys K and Q are generated so that they have different oddness) and the value e is depends on the key as follows: $e = K \; mod \; 2$ (while decrypting the fake message) and $e = Q \; mod \; 2$ (while decrypting the secret message). In such case the single algorithm discloses fake or secrete messages depending on the used key (K, K) or (K, Q).

The probabilistic encryption algorithm that can be associated with the pseudo-probabilistic encryption algorithm (see Fig. 3) is shown in Fig. 4. It is easy to see that the ciphertext produced by the first algorithm at time of simultaneous encryption of the secret message T with secret key Q and the fake message M with the fake key K can be potentially produced by the second one used to encrypt the fake message with the fake key. To distinguish pseudo-probabilistic encryption from probabilistic one requires disclosing the secret message T. When using secure block encryption functions E' and E, for example TripleDES by Pieprzyk et al (2002) with input data block having size $u = 64$ and AES by Pieprzyk et al (2002) with input data block having size $b = 128$, it is computationally impossible to distinguish pseudo-probabilistic encryption scheme Fig. 3 from probabilistic one Fig. 4.

Decryption algorithm that corresponds to both the pseudo-probabilistic and the probabilistic encryption schemes shown in Figs. 3 and 4 is as follows.

1. Set the key $K^* = (K, K')$, where $K' = K$ (for disclosing the fake message) and $K' = Q$ (for disclosing the secret message).
2. Compute the bit $e = K' mod 2$.
3. Decrypt the ciphertext block $C : B = (B_1||B_2) = E_K{}^{-1}(C)$, where the intermediate

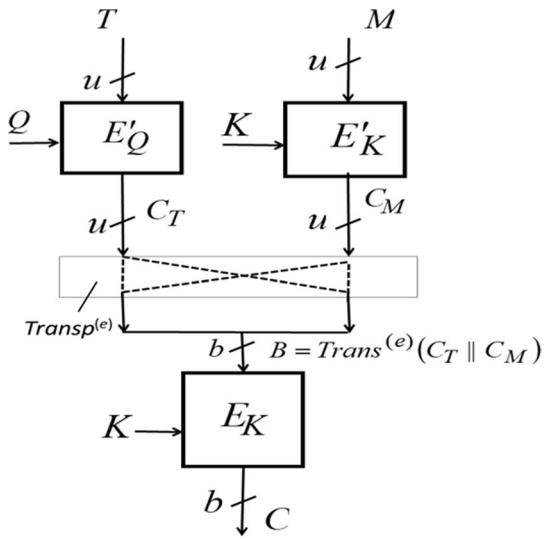

Fig. 3 Block PPC with controlled transposition operation

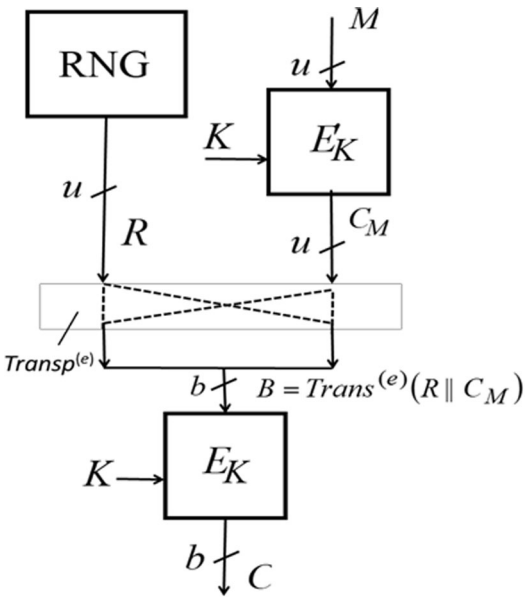

Fig. 4 The associated probabilistic encryption algorithm

ciphertext block B is represented as concatenation of the U-bit data subblocks B_1 and B_2.

4. Perform the controlled transposition operation $Transp^{(e)}(B_1||B_2)$: $(B'_1||B'_2) = Transp^{(e)}(B_1||B_2)$.
5. Compute the u-bit plaintext data block M': $M' = {E'_{K'}}^{-1}(B'_2)$.

At time of the coercive attack the sender and the receiver of the secret message have possibility to cheat plausible they used probabilistic block encryption algorithm. They will open the fake message and the encryption key K with which the fake message was encrypted. The coercer can decrypt the intercepted ciphertext with key K and obtain the fake message. He is also able to get pseudo-random bit string C_T but for him is computationally infeasible to distinguish the pseudo-random value C_T from random one and to demonstrate that the ciphertext contains one message more.

In the next section we consider pseudo-probabilistic block encryption methods that also provide security to coercive attacks with measuring the decryption time.

4 Method for pseudo-probabilistic block encryption

It is proposed to implement pseudo-probabilistic encryption as simultaneous ciphering two messages $Mess1$ (fake) and $Text2$ (secret) of the equal size using the shared keys (K_1, m_1) and (K_2, m_2), where K_1 and K_2 are keys of some block cipher E with v-bit input; m_1 and m_2 are two mutually prime numbers. The messages are divided into v-bit data blocks: $Mess1 = (M_1, M_2, \dots M_z)$ and $Text2 = (T_1, T_2, \dots T_z)$ and then pairs of the respective blocks Mi and Ti are consecutively encrypted as follows:

1. Using the block cipher E and key K_1, it is encrypted the block M of the first message: $C_M = E_{K1}(M)$.
2. Using the block cipher E and key K_2, it is encrypted the block T of the second message: $C_T = E_{K2}(T)$.
3. Using additional secret values m_1 and m_2 compute the block C of output ciphertext as solution of the following system if two congruencies:

$$\begin{cases} C \equiv C_M \mod m_1 \\ C \equiv C_T \mod m_2 \end{cases} \tag{3}$$

where blocks C_T and C_M of the intermediate ciphertexts are interpreted as binary numbers; m_1 and m_2 are mutually prime numbers having size $v + 1$ bits. The size of the output ciphertext block C is equal to $2v + 2$ bits (i.e. the size of the block C is two bits larger than the sum of sizes of the blocks C_T and C_M). Solution of the system (3) is described as follows:

$$C = [C_M m_2 (m_2^{-1} \mod m_1) + C_T m_1 (m_1^{-1} \mod m_2)] \mod m_1 m_2. \tag{4}$$

The values $m_2(m_2^{-1} mod\, m_1)$ and $m_1(m_1^{-1} mod\, m_2)$ can be precomputed at moment of generating the secret keys, therefore the main contribution in computational difficulty of calculating the value C is defined by the operation of dividing the value in square brackets by the modulus $m_1 m_2$.

From practical point of view it is preferable to use the pseudo-probabilistic block encryption method that outputs the ciphertext block that have size equal exactly to $2v$ bits. This requirement can be met using the procedure of combining two blocks C_T and C_M into one block C which consists in solving the following system of two congruencies defined over binary polynomials:

$$\begin{cases} C \equiv E_{K_1}(M) \mod \mu(x) \\ C \equiv E_{K_2}(T) \mod \lambda(x) \end{cases} \tag{5}$$

where $\mu(x)$ and $\lambda(x)$ are mutually irreducible binary polynomials of the degree v (these two polynomials are secret elements); the v-bit blocks C_T and C_M of the intermediate ciphertexts are interpreted as binary polynomials of the degree$v - 1$. Solution of system (5) represents the binary polynomial of the degree $2v$ which is given by the following formula:

$$\begin{aligned} C = &[E_{K1}(M)\lambda(x)(\lambda^{-1}(x) \mod \mu(x)) \\ &+ E_{K2}(T)\mu(x)(\mu^{-1}(x) \mod \lambda(x))] \mod \mu(x)\lambda(x) \end{aligned} \tag{6}$$

Like in the first block encryption method, the polynomials $\lambda^{-1}(x) \mod \mu(x)$ and $\mu^{-1}(x) \mod \lambda(x)$ can be pre-computed to increase the encryption rate.

The related decryption algorithms are evident for the described two variants of the proposed pseudo-probabilistic block encryption method. Decryption algorithms connected with the pseudo-probabilistic encryption algorithms described in this section coincide with the decryption algorithms connected with the associated probabilistic encryption algorithms (see Sect. 5).

5 Associated probabilistic block encryption algorithms

Let us show that the block encryption method described in Sect. 3 met criterion of computational indistinguishability from probabilistic block encryption. For this purpose one should propose a probabilistic block encryption algorithm such that, when being applied to encrypt the fake message, it can potentially produce the ciphertext coinciding with

the ciphertext produced by the pseudo-probabilistic block encryption algorithm.

Probabilistic block encryption algorithm associated with the PPC including procedure of solving the system of congruencies (3) is described as follows. The fake key represents the pair of secret values (K_1, m_1). To encrypt the fake message data block M the following steps are performed:

1. The data block M is encrypted with the block cipher algorithm E: $C_M = E_{K1}(M)$.
2. It is generated a random value $R < 2v$ and a random prime number r such that $2^v < r < 2^{v+1}$.
3. It is computed the output ciphertext block C as solution of the following system of congruencies:

$$\begin{cases} C \equiv C_M & \mod m_1 \\ C \equiv R & \mod r. \end{cases} \tag{7}$$

It is easy to see that the arbitrary value C^* such that $C^* \equiv C_M \mod m_1$ can be obtained as solution of system (7) at different pairs of the values $R < 2^v$ and $r < 2^{v+1}$. Indeed, let us select a random number r^* such that $2^v < r^* < 2^{v+1}$. The respective value R^* is computed as $R^* \equiv C^* \mod r$. Decryption of the ciphertext block is performed as follows.

Algorithm for disclosing the fake message.

1. Compute the intermediate ciphertext block C_M: $C_M = C \mod m_1$.
2. Compute the data block M: $M = E_{K1}^{-1}(C_M)$.

Algorithm for disclosing the secret message.

1. Compute the intermediate ciphertext block C_T: $C_T = C \mod m2$.
2. Compute the data block T: $T = E_{K1}^{-1}(C_T)$.

Probabilistic block encryption algorithm associated with the PPC including procedure of solving the system of congruencies (5) is described as follows. The fake key represents the pair of secret values $(K_1, \mu(x))$. Encryption of the data block M of the fake message is performed as follows:

1. The data block M is encrypted with the block cipher algorithm E: $C_M = E_{K1}(M)$.
2. It is generated a random binary polynomials $\Psi(x)$ (of the degree equal to $v - 1$ or less) and $\rho(x)$ (of the degree v).
3. It is computed the output ciphertext block C as solution of the following system of congruencies (the ciphertext block C_M is considered as binary polynomial):

$$\begin{cases} C \equiv C_M & \mod \mu(x) \\ C \equiv \Psi(x) & \mod \rho(r). \end{cases} \tag{8}$$

Evidently, a bit string C^* such that $C^* = C_M \mod \mu(x)$ can be obtained as solution of system (8) at different pairs of the polynomials $\Psi(x)$ and $\rho(x)$. Indeed, for arbitrary polynomial $\rho(x)$ of the degree v the related polynomial is $\Psi(x) = C^* \mod \rho(x)$. Decryption of the ciphertext block is performed as follows.

Algorithm for disclosing the fake message.

1. Compute the intermediate ciphertext block C_M: $C_M = C \mod \mu(x)$.
2. Compute the data block M: $M = E_{K1}^{-1}(C_M)$.

Algorithm for disclosing the secret message.

1. Compute the intermediate ciphertext block C_T: $C_T = C \mod \lambda(x)$.
2. Compute the data block T: $T = E_{K2}^{-1}(C_T)$.

6 Randomization of the pseudo-probabilistic ciphers

Since the ciphertexts produced by pseudo-probabilistic cipher and by probabilistic cipher associated with the first one are indistinguishable, the potential adversary faces the following problem while trying to decrypt a cryptogram. Suppose he find the key with which a sensible message M is recovered from the cryptogram. Then he tries unsuccessfully to find another key with which the cryptogram could be decrypted into another sensible message T. In such situation cryptanalyst do not know whether he is solving unsolvable problem (if the message M had been encrypted with probabilistic cipher) or he should continue cryptanalysis (if two messages M and T had been encrypted with probabilistic cipher).

Like in the case of deterministic block encryption algorithms, the pseudo-probabilistic ciphers can be strengthened by embedding randomization in the encryption process (see beginning of Sect. 3). As regards to pseudo-probabilistic ciphers described in Sect. 3 randomization can be embedded with concatenation of a random bit string with input block M or with input block T, like in the case of simple randomization of block ciphers Moldovyan and Moldovyan (2006). To provide possibility to recover input data blocks M or T with the same single decryption algorithm two independent random bit strings R_M and R_T are to be concatenated to M and T respectively. Such simple randomization mechanism can be also applied to pseudo-probabilistic block ciphers described in Sect. 4.

However in the last case it is more efficient to embed randomization as generation of the third congruency (in the system of congruencies solution of which represents the

ciphertext) with random parameters. This method transforms two pseudo-probabilistic block ciphers from Sect. 4 into the following two randomized algorithms:

The first randomized pseudo-probabilistic block cipher.

1. Using the key K_1, it is encrypted the input data block M: $C_M = E_{K1}(M)$.
2. Using the key K_2, it is encrypted the input block T: $C_T = E_{K2}(T)$.
3. Generate two random numbers R and r such that $R < r < m_1$ and r is mutually prime with m_1 and m_2.
4. Compute the ciphertext block C as solution of the following system of three congruencies

$$\begin{cases} C \equiv C_M & \mod m_1 \\ C \equiv C_T & \mod m_2 \\ C \equiv R & \mod r \end{cases} \tag{9}$$

Solution of the system (9) is described as follows:

$$\begin{aligned} C = [C_M r m_2 (r^{-1} m_2^{-1} \mod m_1) + C_T r m_1 (r^{-1} m_1^{-1} \mod m_2) \\ + R m_2 m_1 (m_1^{-1} m_1 - 1 \mod r)] \mod m_1 m_2 r \end{aligned} \tag{10}$$

The second randomized pseudo-probabilistic block cipher.

1. Using the key K_1, it is encrypted the input data block M: $C_M = E_{K1}(M)$.
2. Using the key K_2, it is encrypted the input block T: $C_T = E_{K2}(T)$.
3. Generate random binary polynomial $\rho(x)$ of the degree $v' \le v$ which is mutually irreducible with $\mu(x)$ and with $\lambda(x)$.
4. Generate random binary polynomial $\theta(x)$ of the degree less than v'.
5. Compute the ciphertext block C as solution of the following system of three congruencies

$$\begin{cases} C \equiv C_M & \mod \mu(x) \\ C \equiv C_T & \mod \lambda(x) \\ C \equiv \theta(x) & \mod \Psi(x) \end{cases} \tag{11}$$

Solution of the system (11) is described as follows:

$$\begin{aligned} C = [C_M \lambda(x)\rho(x)(\lambda^{-1}(x)\rho^{-1}(x) & \mod \mu(x)) \\ + C_T \mu(x)\rho(x)(\mu^{-1}(x)\rho^{-1}(x) & \mod \lambda(x)) \\ + \theta(x)\mu(x)\lambda(x)(\mu^{-1}(x)\lambda^{-1}(x) & \mod \rho(x))] \\ & \mod \mu(x)\lambda(x)\rho(x) \end{aligned} \tag{12}$$

It is evident that embedding randomization leads to increase of the size of the ciphertext block. For example in the case of the second randomized pseudo-probabilistic block cipher the size of the block C is equal to $2v + v'$ bits, whereas for the source deterministic version of the algorithm the ciphertext block has size equal to $2v$ bits.

While considering pseudo-probabilistic block ciphers in which the pair of the intermediate ciphertext blocks C_M and C_T is transformed into the output ciphertext block representing solution of the system congruencies, up to this point we deal with the case of equal size of the input data blocks T and M. In general case in the algorithms described in this section and Sect. 4. one can define arbitrary ratio of the lengths of the input data blocks. For example, the deniability of encryption can be strengthened defining smaller size of the data block T. This corresponds to defining size of the modulus m_2 (and $\lambda(x)$) less that size of the modulus m_1(and $\mu(x)$). In the case of the non-randomized and randomized ciphers based on computations over binary polynomials the size of the data blocks M and T is defined by the degree of the polynomials $\mu(x)$ and $\lambda(x)$, correspondingly. Thus, in the last case the size of the output ciphertext block is exactly equal to sum of sizes of input data blocks in the non-randomized pseudo-probabilistic block cipher and to sum of sizes of input data blocks plus size of the binary polynomial $\rho(x)$ in the randomized ones.

7 Conclusion

It has been proposed to construct the block deniable encryption as process of pseudo-probabilistic block encryption of secret and fake messages. At time of the coercive attack the sender or/and receiver open the fake message and fake encryption key and declare their using the probabilistic block encryption of the opened message. The coercer is computationally unable to distinguish the intercepted ciphertext from ciphertext produced by probabilistic encryption. The proposed deniable encryption methods are fast and provide bi-deniability (resistance to simultaneous attack on both the sender and the receiver of the message).

To provide higher security one can use the randomized pseudo-probabilistic encryption algorithms described in Sect. 6. The considered pseudo probabilistic block ciphers with different size of input data blocks of fake and secret messages also represent interest for some practical applications. The main result of the paper is its contribution to the class of pseudo-probabilistic ciphers to which one can attribute the proposed block cryptoschemes and introduced earlier pseudo-probabilistic stream ciphers by Moldovyan et al (2015, 2016).

Acknowledgements The reported study was funded by Russian Foundation for Basic Research (project #18-57-54002-Viet_a) and by Vietnam Academy of Science and Technology (project # QTRU01.08/18-19).

References

Barakat TM (2014) A new sender-side public-key deniable encryption scheme with fast decryption. KSII Trans Internet Inf Syst 8(9):3231–3249. https://doi.org/10.3837/tiis.2014.09.016

Canetti R, Dwork C, Naor M, Ostrovsky R (1997) Deniable encryption. In: Kaliski BS (ed) Advances in cryptology—CRYPTO '97. Springer, Berlin, pp 90–104. https://doi.org/10.1007/BFb0052229

Dürmuth M, Freeman DM (2011) Deniable encryption with negligible detection probability: an interactive construction. In: Paterson KG (ed) Advances in cryptology—EUROCRYPT 2011. Springer, Berlin, pp 610–626. https://doi.org/10.1007/978-3-642-20465-4_33

Ishai Y, Kushilevitz E, Ostrovsky R, Prabhakaran M, Sahai A (2011) Efficient non-interactive secure computation. In: Paterson KG (ed) Advances in cryptology—EUROCRYPT 2011. Springer, Berlin, pp 406–425. https://doi.org/10.1007/978-3-642-20465-4_23

Meng B (2009) A secure internet voting protocol based on non-interactive deniable authentication protocol and proof protocol that two ciphertexts are encryption of the same plaintext. J Netw 4(5):370–377. https://doi.org/10.4304/jnw.4.5.370-377

Moldovyan NA, Moldovyan AA (2006) Innovative cryptography (programming series). Charles River Media Inc, Rockland

Moldovyan NA, Moldovyan AA (2007) Data-driven block ciphers for fast telecommunication systems, 1st edn. Auerbach Publications, Boca Raton

Moldovyan AA, Moldovyan DN, Shcherbacov VA (2015) Stream deniable-encryption algorithm satisfying criterion of the computational indistinguishability from probabilistic ciphering. Workshop Found Inf I:318–330

Moldovyan NA, Moldovyan AA, Moldovyan DN, Shcherbacov VA (2016) Stream deniable–encryption algorithms. Comput Sci J Moldova 24(1(70)):68–82

Moldovyan NA, Shcherbacov VA, Eremeev MA (2017) Deniable encryption protocols based on commutative ciphers. Quasigroups Relat Syst:95–108.

O'Neill A, Peikert C, Waters B (2011) Bi-deniable public-key encryption. In: Rogaway P (ed) Advances in cryptology—CRYPTO 2011. Springer, Berlin, pp 525–542. https://doi.org/10.1007/978-3-642-22792-9_30

Pieprzyk J, Hardjono T, Seberry J (2002) Fundamentals of computer security. Springer, Berlin. https://doi.org/10.1007/978-3-662-07324-7