BULETINUL ACADEMIEI DE ŞTIINŢE
A REPUBLICII MOLDOVA. MATEMATICA
Number 3(85), 2017, Pages 115–119
ISSN 1024–7696

Post-quantum No-key Protocol

N. A. Moldovyan, A. A. Moldovyan, V. A. Shcherbacov

Abstract. There is proposed three-pass no-key protocol that is secure to hypothetic attacks based on computations with using quantum computers. The main operations are multiplication and exponentiation in finite ground field $GF(p)$. Sender and receiver of secret message also use representation of some value $c \in GF(p)$ as product of two other values $R_1 \in GF(p)$ and $R_2 \in GF(p)$ one of which is selected at random. Then the values R_1 and R_2 are encrypted using different local keys.

Mathematics subject classification: 94A60, 11S05.
Keywords and phrases: Post-quantum cryptography, computationally difficult problem, conjugacy search problem, discrete logarithm, commutative encryption, no-key protocol.

1 Introduction

An open problem of cryptography is design of post-quantum cryptographic algorithms and protocols [1, 2]. The most practical public-key cryptoschemes are based on difficulty of discrete logarithm [3–5] and of factoring integers containing two large prime factors [6, 7]. The three-pass no-key encryption protocol [3] based on the first problem represents significant practical interest, for example, to perform secure encryption with short shared keys [8].

Quantum computations are in progress and it is expected that in observable future it will be possible to implement polynomial algorithms solving the discrete logarithm and factoring problems [9]. Therefore researchers are looking for new cryptographic primitives and designs of cryptoschemes, for example, the hidden conjugacy search problem in finite non-commutative groups was proposed as primitive for designing post-quantum cryptoschemes [10–12].

In the present communication we propose post-quantum implementation of the three-pass no-key encryption protocol. In the proposed protocol there is used exponentiation in the finite ground field $GF(p)$, where p is a sufficiently large prime, like in the known no-key encryption protocol. However it is additionally used representation of some element of the field $GF(p)$ as product of two other elements one of which is selected at random and serves as an additional local key. Due to such representation performed independently on the side of the message sender and on the side of the receiver, solving the discrete logarithm problem (DLP) cannot be used to break the proposed protocol. No key encryption protocol [3] exploits commutative ciphers.

Encryption function E is called commutative if it satisfies the following condition

$$E_K[E_Q(M)] = E_Q[E_K(M)],$$

where K and Q are encryption keys and M is some plaintext, for arbitrary keys K and $Q \neq K$.

The appropriate commutative encryption function is provided by the exponentiation encryption method by Pohlig and Hellman [13] that is described as follows.

Suppose p is a 2048-bit prime such that number $p-1$ contains a large prime divisor q the size of which is $|q| \geq 256$ bits, for example, $p = 2q + 1$.

To select an encryption/decryption key (e, d) one needs to generate a random number e that is mutually prime with $p-1$ and has size $|e| \geq 256$ bits and then to compute $d = e^{-1} \bmod p-1$.

The encryption procedure is described with the formula

$$C = M^e \bmod p.$$

Decryption of the ciphertext C is performed as computing the value

$$M = C^d \bmod p.$$

Suppose Alice wishes to send the secret message M to Bob, using a public channel and no shared key. For this purpose they can use the following no key protocol:

(i) Alice chooses a random key (e_A, d_A) and encrypts the message M using the formula $C_1 = M^{e_A} \bmod p$. Then she sends the ciphertext C_1 to Bob;

(ii) Bob chooses a random key (e_B, d_B) and encrypts the ciphertext C_1 as follows: $C_2 = C_1^{e_B} \bmod p$ and sends the ciphertext C_2 to Alice;

(iii) Alice decrypts the ciphertext C_2 obtaining the ciphertext C_3 : $C_3 = C_2^{d_A} \bmod p$. Then she sends the ciphertext C_3 to Bob;

(iv) Bob computes the message $M = C_3^{d_B} \bmod p$.

This three-pass protocol provides security to passive attacks (potential adversary only intercepts the values sent via public channel, but does not masquerade as sender or receiver of secret message), since the used exponentiation cipher is as secure as discrete logarithm problem is hard.

However, the described protocol is not secure against attacks using hypothetic quantum computers.

We propose the following post-quantum implementation of the no-key protocol.

1. Alice generates two local keys in the form of two pairs of numbers (e_{A1}, d_{A1}) and (e_{A2}, d_{A2}) such that $d_{A1} = e_{A1}^{-1} \bmod p-1$ and $d_{A2} = e_{A2}^{-1} \bmod p-1$, and forms the pair of random numbers $R_1 < p$ and $R_2 < p$ such that $M = R_1 R_2 \bmod p$, where M is some secret message. Then she encrypts the numbers R_1 and R_2, using formulas $C_1' = R_1^{e_{A1}} \bmod p$ and $C_1'' = R_2^{e_{A2}} \bmod p$, and sends the ciphertexts C_1' and C_1'' to Bob.

2. Bob generates his two local keys (e_{B1}, d_{B1}) and (e_{B2}, d_{B2}) and represents each of the numbers C_1' and C_1'' as product of the pair of random numbers (R_{11}, R_{12}), where $R_{11} < p$ and $R_{12} < p$, and (R_{21}, R_{22}), where $R_{21} < p$ and $R_{22} < p$, respectively: $R_1 = R_{11}R_{12} \bmod p$; $R_2 = R_{21}R_{22} \bmod p$.

Then he generates two random values $L_1 < p$ and $L_2 < p$ and encrypts the numbers R_{11}, R_{12}, R_{21}, and R_{22} as follows:

$$\begin{array}{ll} C_2' = R_{11}^{e_{B1}} L_1^{d_{B2}} \bmod p; & C_2''' = R_{21}^{e_{B1}} L_2^{d_{B2}} \bmod p; \\ C_2'' = R_{12}^{e_{B2}} L_1^{-d_{B1}} \bmod p; & \overline{C}_2 = R_{22}^{e_{B2}} L_2^{-d_{B1}} \bmod p, \end{array}$$

and sends the ciphertexts C_2', C_2'', C_2''', and $\overline{C}_2$ to Alice.

3. Alice generates random numbers $N_1 < p$ and $N_2 < p$ and decrypts the ciphertexts C_2', C_2'', C_2''', and $\overline{C}_2$ as follows:

$$\begin{array}{ll} C_3' = (C_2')^{d_{A1}} N_1 \bmod p; & C_3''' = (C_2''')^{d_{A2}} N_1^{-1} \bmod p; \\ C_3'' = (C_2'')^{d_{A1}} N_2 \bmod p; & \overline{C}_3 = (\overline{C}_2)^{d_{A2}} N_2^{-1} \bmod p, \end{array}$$

and sends the ciphertexts C_3', C_3'', C_3''', and $\overline{C}_3$ to Bob.

4. Bob recovers the secret message M from the values C_3', C_3'', C_3''', and $\overline{C}_3$ multiplying the numbers S', S'', S''' and $\overline{S}$ that are computed as follows: $S' = (C_3')^{d_{B1}} \bmod p$; $S'' = (C_3'')^{d_{B2}} \bmod p$; $S''' = (C_3''')^{d_{B1}} \bmod p$; $\overline{S} = (\overline{C}_3)^{d_{B2}} \bmod p$; $M = S'S''S'''\overline{S} \bmod p$.

A correctness proof of the protocol is as follows:

$$\begin{aligned} &S' \equiv (C_3')^{d_{B1}} \equiv (C_2')^{d_{B1}d_{A1}} N_1^{d_{B1}} \equiv R_{11}^{d_{B1}d_{A1}e_{B1}} L_1^{d_{B1}d_{A1}d_{B2}} N_1^{d_{B1}} \equiv \\ &R_{11}^{d_{A1}} L_1^{d_{B1}d_{A1}d_{B2}} N_1^{d_{B1}} \bmod p; \\ &S'' \equiv (C_3'')^{d_{B2}} \equiv (C_2'')^{d_{B2}d_{A1}} N_2^{d_{B2}} \equiv R_{12}^{d_{B2}d_{A1}e_{B2}} L_1^{-d_{B2}d_{A1}d_{B1}} N_2^{d_{B2}} \equiv \\ &R_{12}^{d_{A1}} L_1^{-d_{B2}d_{A1}d_{B1}} N_2^{d_{B2}} \bmod p; \\ &S''' \equiv (C_3''')^{d_{B1}} \equiv (C_2''')^{d_{B1}d_{A2}} N_1^{-d_{B1}} \equiv R_{21}^{d_{B1}d_{A2}e_{B1}} L_2^{d_{B1}d_{A1}d_{B2}} N_1^{-d_{B1}} \equiv \\ &R_{21}^{d_{A2}} L_2^{d_{B1}d_{A2}d_{B2}} N_1^{-d_{B1}} \bmod p; \\ &\overline{S} \equiv (\overline{C}_3)^{d_{B2}} \equiv (\overline{C}_2)^{d_{B2}d_{A2}} N_2^{-d_{B2}} \equiv R_{22}^{d_{B2}d_{A2}e_{B2}} L_2^{-d_{B2}d_{A2}d_{B1}} N_2^{-d_{B2}} \equiv \\ &R_{22}^{d_{A2}} L_2^{-d_{B2}d_{A2}d_{B1}} N_2^{-d_{B2}} \bmod p. \end{aligned}$$

Multiplying the numbers S' and S'' one gets

$$\begin{aligned} &S'S'' \equiv R_{11}^{d_{A1}} L_1^{d_{B1}d_{A1}d_{B2}} N_1^{d_{B1}} R_{12}^{d_{A1}} L_1^{-d_{B2}d_{A1}d_{B1}} N_2^{d_{B2}} \equiv \\ &(R_{11}R_{12})^{d_{A1}} N_1^{d_{B1}} N_2^{d_{B2}} \equiv (C_1')^{d_{A1}} N_1^{d_{B1}} N_2^{d_{B2}} \equiv \\ &(R_1)^{d_{A1}e_{A1}} N_1^{d_{B1}} N_2^{d_{B2}} \equiv R_1 N_1^{d_{B1}} N_2^{d_{B2}} \bmod p. \end{aligned}$$

Multiplying the numbers S''' and $\overline{S}$ one gets

$$S'''\overline{S} \equiv R_{21}^{d_{A2}} L_2^{d_{B1}d_{A2}d_{B2}} N_1^{-d_{B1}} R_{22}^{d_{A2}} L_2^{-d_{B2}d_{A2}d_{B1}} N_2^{-d_{B2}} \equiv$$
$$(R_{21}R_{22})^{d_{A2}} N_1^{-d_{B1}} N_2^{-d_{B2}} \equiv (C_2')^{d_{A2}} N_1^{-d_{B1}} N_2^{-d_{B2}}$$
$$\equiv (R_2)^{d_{A2}e_{A2}} N_1^{-d_{B1}} N_2^{-d_{B2}} \equiv R_2 N_1^{-d_{B1}} N_2^{-d_{B2}} \bmod p.$$

Thus, we have

$$S'S''S'''\overline{S} \equiv R_1 N_1^{d_{B1}} N_2^{d_{B2}} R_2 N_1^{-d_{B1}} N_2^{-d_{B2}} \equiv R_1 R_2 \bmod p.$$

Therefore, $M = S'S''S'''\overline{S} \bmod p$.

We invite the reader to participate in security analysis of the proposed protocol.

References

[1] Proceedings of the 7th International Workshop on Post-Quantum Cryptography, PQCrypto 2016. Fukuoka, Japan, February 24–26, 2016 // Lecture Notes in Computer Science (LNCS) series. Springer, 2016, vol. 9606, 270 p.

[2] Federal Register: Announcing Request for Nominations for Public-Key Post-Quantum Cryptographic Algorithms // Federal Register. The Daily journal of the United States Goverment URL: https://www.gpo.gov/fdsys/pkg/FR-2016-12-20/pdf/2016-30615.pdf.

[3] Menezes A. J., Oorschot P. C., Vanstone S. A. *Handbook of applied cryptography.* CRC Press, New York, London, 1996, 780 p.

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

[5] GOST R 34.10-2012. Russian Federation Standard. Information Technology. Cryptographic data Security. Produce and check procedures of Electronic Digital Signature. Government Committee of the Russia for Standards, 2012 (in Russian).

[6] Lenstra A. K. *Integer factoring.* Designs, codes and cryptography, 2000, **19**, No. 2, 101-–128.

[7] Moldovyan N. A., Moldovyan A. A., Shcherbacov V. A. *Generating Cubic Equations as a Method for Public Encryption.* Bul. Acad. Ştiinţe Repub. Moldova, Mat., 2015, No. 3(79), 60–71.

[8] Moldovyan N. A., Moldovyan A. A., Shcherbacov A. V. *Deniable-encryption protocol using commutative transformation.* Workshop on Foundations of Informatics, July 25–29, 2016, Chisinau, Proceedings, p. 285–298.

[9] Shor P. W. *Polynomial-time algorithms for prime factorization and discrete logarithms on quantum computer.* SIAM Journal of Computing, 1997, **26,** 1484-–1509.

[10] Moldovyan D. N., Moldovyan N. A. *Cryptoschemes over hidden conjugacy search problem and attacks using homomorphisms.* Quasigroups and Related Systems, 2010, **18,** 177–186.

[11] Moldovyan D. N., Moldovyan N. A. *A New Hard Problem over Non-Commutative Finite Groups for Cryptographic Protocols.* Springer Verlag LNCS, 2010, vol. 6258, p. 183–194 / 5th Int. Conference MMM-ANCS 2010 Proceedings.

[12] Moldovyan D. N. *Non-Commutative Finite Groups as Primitive of Public-Key Cryptoschemes.* Quasigroups and Related Systems, 2010, **18,** 165–176.

[13] Hellman M. E., Pohlig S. C. *Exponentiation Cryptographic Apparatus and Method.* U.S. Patent # 4,424,414. 3 Jan. 1984.

Received July 15, 2017

N. 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*

A. A. Moldovyan
ITMO University
Kronverksky pr., 10, St.Petersburg, 197101
Russia
E-mail: *maa1305@yandex.ru*

V. A. Shcherbacov
Institute of Mathematics and Computer Science
Academy of Sciences of Moldova
Academiei str. 5, MD−2028 Chişinău
Moldova
E-mail: *scerb@math.md*