Journal of Mathematics and Statistics 7 (3): 165-168, 2011
ISSN 1549-3644

A New Cryptosystem Based on Factoring and Discrete Logarithm Problems

E.S. Ismail and M.S.N. Hijazi
School of Mathematical Sciences, Faculty of Science and Technology,
University Kebangsaan Malaysia, 43600 UKM Bangi, Selangor, Malaysia

Abstract: Problem statement: A cryptosystem allows a sender to send any confidential or private message using a receiver's public key and later the receiver confirms the integrity of the received message using his secret key. Currently the existing cryptosystems were developed based on a single hard problem like factoring, discrete logarithm, residuosity, knapsack or elliptic curve discrete logarithm. Although these schemes appear secure, one day in a near future they may be broken if one finds a solution of a single hard problem. **Approach:** To solve this problem, we developed a new cryptosystem based on two hard problems; factoring and discrete logarithm. We integrated the two problems in our encrypting and decrypting equations so that the former depends on two public keys whereas the latter depends on two corresponding secret keys. **Results:** The new cryptosystem is shown secure against the most three considering attacks. The efficiency performance of our scheme only requires $3T_{exp}$ $+T_{mul}$ + T_{hash} time complexity for encryption and $2T_{exp}$ + T_{mul} time complexity for decryption and this magnitude of complexity is considered minimal for multiple hard problems-like cryptosystems. **Conclusion:** The new cryptosystem based on multiple hard problems provides longer and higher security level than that schemes based on a single hard problem. The adversary has to solve the two problems simultaneously in order to recover a corresponding plaintext (message) from the received ciphertext (encrypted message).

Key words: Cryptology, cryptography, cryptosystem, factoring, discrete logarithms

INTRODUCTION

Most the existing Cryptosystems (CRS) have the common feature that they are based on a single number-theoretic cryptographic assumption (Diffie and Hellman, 1975) like Discrete Logarithms (DL) (Verkhovsky and Sadik, 2009) or Factoring (FAC) (Verkhovsky, 2009) a large composite number or Elliptic Curve Discrete Logarithm (ECDL) (Koblizt *et al*., 2000) problem. Even though such problems remain hard today, it is understood that one day in the future the FAC, DL or ECDL problems could be easily solved. As soon as this happens, CRS based on such problems will no longer be secure. This scenario has led many cryptographers to come up with CRS based on multiple number-theoretic hard problems (Baocang and Yupu, 2005; Othman *et al*., 2008; Pramod and Manju, 2010). The major motivation is that these kinds of schemes are more secure than the schemes based on a single hard problem. In other words, an adversary needs a longer period of time in order to break the two hard problems-based CRS since it is very unlikely for the adversary to obtain the solutions of these two problems simultaneously. However, how to design a public key encryption scheme based on multiple number-theoretic cryptographic assumptions is still a field in need of cultivation.

In this article, we designed a new cryptosystem based on two hard problems namely; factoring and discrete logarithm problems. With its guaranteed security, we also showed that the performance of the scheme requires reasonable numbers of operations in both encrypting and decrypting processes, which makes it very efficient to be implemented in the real world applications.

Some notations and parameters: Throughout the paper, we use the following notations and parameters unless otherwise specified:

- Two large strong random primes (Gordon, 1984) p and q which are safe primes and set the modulus n = pq
- A function $\phi(n)$ = (p-1) (q-1) is a phi-Euler function and gcd (a, b) is the greatest common divisor of a and b

Corresponding Author: E.S. Ismail, School of Mathematical Sciences, Faculty of Science and Technology, University Kebangsaan Malaysia, 43600 UKM Bangi, Selangor, Malaysia

- g is a primitive element in $\mathbb{Z}_n^* = \{z \mid \gcd(z,n) = 1\}$ with order n satisfying $g^{n-1} \equiv 1 (\bmod n)$
- h(.) is a cryptographic hash function (Schneier, 1996) whose output is a t-bit length and we assume here that t = 128

MATERIALS AND METHODS

We present a cryptosystem based on hybrid-mode problems; factoring and discrete logarithms. The scheme is described in three phases namely Setup, Encryption and Decryption. In Setup phase, the public and secret keys of users are calculated. Once computed, the public keys will be published in public directory so that anyone including the adversaries could access it while the secret keys remain secret except the owners. In Encryption phase, the original message that to be sent is first hashed using the appropriate cryptographic hash function h(.). This function determines a fixed length of output by hashing any arbitrarily length of input. Then a sender gets his hashed message encrypted. This is done by using the receiver's public key and sender's commitment of secret number. The encrypted message is then sent to the legal receiver. In Decryption phase, the receiver recovers the original message by using his own secret keys and without these secret keys no one can read the original message.

Setup phase:

- Pick randomly two integers e, x < n from $\mathbb{Z}_n^*$ such that gcd(e, n) = 1
- Solve the equation ed ≡ 1 mod $\phi(n)$ for d
- Compute the number $y \equiv g^x \bmod n$

The public key is formed by (e, y) and can be accessed in the public directory and the secret keys is given by (d, x) and only known to the legal receiver.

Encryption: The sender encrypts his message h(m) as follows:

- Select at random an integer c<n from $\mathbb{Z}_n^*$
- Get the original message hashed and assume that the resultant becomes h(m)
- Disguise the message by computing:

$$c_1 \equiv (h(m)y^{-c})^e \bmod n \tag{1a}$$

- Calculate the number:

$$c_2 \equiv g^c \bmod n \tag{1b}$$

In the original El-Gamal (1985) cryptosystem, we compute the number c_1 in Eq. 1 without the exponent e. In our scheme, we need this exponent to disguise our message 'twice' and to realize the hybrid-mode problems-based cryptosystem.

Decryption: The receiver decrypts the obtained encrypted message (c_1, c_2) as below:

- Compute the following:

$$c_1^d c_2^x \equiv h(m) \bmod n \tag{2}$$

A simple example: For purpose of validation, we illustrate an example to show the basic principle of our developed cryptosystem. Practitioners are not recommended to choose keys or parameters computed in this example in practice since inappropriate parameters would make this scheme vulnerable to attacks.

Assume that p = 29, q = 43. Then the modulus and its Euler-function are now given by n = 1247 and $\phi(n)$ = 1176. Next chooses the number e = 11, x = 19 and g = 17. Thus our public and secret keys of the scheme are (11, 1143) and (107, 19) respectively. To encrypt the message h(m) = 1122, the sender selects c = 3 and computes and sends receiver:

$c_1 \equiv (1122 \times 1143^{-3})^{11} \equiv 322 \bmod 1247$ and
$c_2 \equiv 17^3 \equiv 1172 \bmod 1247$

The receiver recovers the original message as below:

$322^{107}\ 1172^{19} \equiv 1122 \bmod 1247$

RESULTS

We discuss our results according to the following criterion:

- Verification of the new cryptosystem
- Security analysis
- Efficiency performance

To verify our scheme, we prove that the decrypting Eq. 2 is correct. For security consideration, we use a technique from heuristic security to show that the scheme is secure. We do this by delivering the scheme to the literature for attacks. We consider three possible attacks by which an adversary (Adv) may try to take down the new cryptosystem. We define each attack and give the corresponding analysis of why this attack would fail. For efficiency performance, we evaluate the time complexity for both phases; encryption and decryption and also the communication cost for our scheme.

Verification: We validate our new scheme by proving the following theorem.

Theorem: If the algorithms of Setup and Encryption run smoothly then the decryption of the encrypted message in Decryption is correct.

Proof: The Eq. 2 above is true for all encrypted message (c_1, c_2) since:

$$\begin{aligned}&c_1^d c_2^x\\&\equiv [h(m)y^{-c})^e]^d (g^c)^x \equiv (h(m)y^{-c})g^{cx}\\&\equiv (h(m)g^{-cx})g^{cx} \equiv h(m) \bmod n\end{aligned}$$

Security attack: We show that our scheme is heuristically secure by considering the following three most common attacks.

Direct attack: Adv wishes to obtain all secret keys using all information available from the system. In this case, Adv needs to solve FAC and DL. The best way to factorize the modulus n = pq, is by using the number field sieve method (Lenstra *et al*., 1990). However, this method is just dependent on the size of modulus n and it is computationally infeasible to factor an integer of size 1024-bit and above. Next, to increase the security of our scheme, we must select strong primes (Gordon, 1984) to avoid attacks using special-purpose factorization algorithms. We can achieve and maintain the same security level for DL by selecting the modulus n = pq with $\frac{p-1}{2}$ and $\frac{q-1}{2}$ respectively are product of two 512-bit strong primes.

Factoring attack: Assume that the Adv successfully solves the factoring problem so that he knows the secret d. With this information in hand, he learns that

$$c_1^d \equiv (h(m)y^{-c})^{ed} \equiv h(m)g^{-cx} \bmod n$$

From the above equation, to recover the original message h(m), one has to remove the term g^{-cx} from c_1^d and this only can be done if one knows the secret number x. Since at this stage the DL problem remains hard to solve then the Adv would fail.

Discrete logarithm attack: Assume that the Adv is able to solve the DL problem and thus obtain the secret integer x. He then knows that

$$c_2^x \equiv g^{cx} \bmod n$$

By knowing this number the Adv tries to recover the original message h(m) from the equation

$$c_1 \equiv (h(m)y^{-c})^e \equiv h(m)^e g^{-cxe} \bmod n$$

Since the exponent e is public, he manages to remove the term g^{-cxe} from c_1 and obtains $h(m)^e$. Unfortunately, to read the original message he must has the secret d in hand but this is impossible since the FAC is hard to solve.

Efficiency performance: Next, we investigate the performance of our scheme in terms of number of keys, computational complexity and communication costs. The following notations are used to analyse the performance of the scheme.

- SK and PK denote the number of secret and public keys respectively
- T_{exp} is the time taken for a modular exponentiation and T_{mul} is the time taken for a modular multiplication
- T_{squ} is the time taken for a modular square computation and T_{srt} is the time taken for a modular square-root computation
- T_{inv} is the time taken for a modular inverse computation and T_{hash} is the time taken for performing a hash function,
- |x| denotes the bit length of x

Here we ignore the time performing modular addition or subtraction computation and we assume that the probability of the bit being selected as 0 or 1 is $\frac{1}{2}$. The performance of our new cryptosystem is summarized as in Table 1.

From Table 1, the sender performs $721T_{mul} + T_{hash}$ time complexity for encryption and the receiver performs $481T_{mul}$ time complexity for decryption using the conversion $T_{exp} = 240T_{mul}$ (Koblizt *et al*., 2000). Finally the communication costs or size of parameters of the scheme is 3|n|.

Table 1: The performance of our new cryptosystem

Our new cryptosystem		
The number of keys	SK	2
	PK	2
Computational complexity	Encryption	$3T_{exp} + T_{mul} + T_{hash}$
	Decryption	$2T_{exp} + T_{mul}$
Communication cost	Encryption	2n
	Decryption	n

DISCUSSION

Most of the designated cryptosystems are based on a single hard problem like factoring, discrete logarithm and elliptic curve discrete logarithm problems. If one day an enemy could find a polynomial algorithm solving this problem, he then can read the original message from any corresponding encrypted message.

Our new developed cryptosystem is prevented from this type of problem. This is because our scheme is designed based on two hard problems namely factoring and discrete logarithm. The enemy only can break this scheme if he can solve the two problems simultaneously and this is very unlikely to happen. If he manages to find a solution to one of the underlying hard problem, our scheme remains secure as the other problem remains hard to solve for at least another period of time.

Our scheme next is protected from the most common considering attacks for scheme based on two hard problems. The performance analysis reveals that the developed scheme requires only minimal operations in encryption and decryption phases and thus makes it very efficient.

CONCLUSION

We presented a new cryptosystem based on factoring and discrete logarithms. The proposed scheme requires respectively $721T_{mul} + T_{hash}$ and $481T_{mul}$ for encryption and decryption. Some possible attacks have also been considered and we showed that the scheme is secure from those attacks.

ACKNOWLEDGEMENT

We acknowledge the financial support received from University Kebangsaan Malaysia under the Research University Grant UKM-OUP-ICT-36-177/2009.

REFERENCES

Baocang, W. and H. Yupu, 2005. Public key cryptosystem based on two cryptographic assumptions. IEE Proc. Commun., 152: 861-865. http://ieeexplore.ieee.org/stamp/ stamp.jsp?tp=&arnumber=1561963

Diffie, W. and M.E. Hellman, 1975. New direction in cryptography. IEEE Trans. Inform. Network. Appli., 557-560. http://citeseer.ist.psu.edu/old/diffie76 new.html.

El-Gamal, T., 1985. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans. Inform. Theory, 31: 469-472. http://dsns.csie.nctu.edu.tw/ research/crypto/HTML/PDF/C84/10.PDF

Gordon, J., 1984. Strong RSA keys. Elect. Lett., 20: 514-516. http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=04248822

Koblizt, N., A. Menezes, S. Vanstone, 2000. The state of elliptic curve cryptography. Design, Codes Cryptography, 19: 173-193. http://modular.fas.harvard.edu/edu/Fall2001/124/misc/koblitz_ecc.pdf

Lenstra, A.K., H.W. Lenstra. Jr, M.S. Manesse and J.M. Pollard, 1990. The number field sieve. Proceedings of the 22nd ACM Symposium on Theory of Computing (ACMSTC'90), Baltimore, Maryland, USA., pp: 564-572. http://www.std.org/~msm/common/nfspaper.pdf

Othman, M., E.M. Abulhirat, Z.M. Ali, M.R.M. Said and R. Johari, 2008. A new computation algorithm for a cryptosystem based on Lucas functions. J. Comput. Sci., 4: 1056-1060. DOI: 10.3844/jcssp.2008.1056.1060

Pramod, K.V. and C. Manju, 2010. A cryptosystem using the concepts of algebraic geometric code. J. Comput. Sci., 6: 244-249. DOI: 10.3844/jcssp.2010.244.249

Schneier, B., 1996. Applied Cryptography: Protocols, Algorithms and Source Code in C. 2nd Edn., Wiley, ISBN: 0471128457, 9780471128458

Verkhovsky, B.S. and M.S. Sadik, 2009. Accelerated search for Gaussian generator based on triple prime integers. J. Comput. Sci., 5: 614-618. DOI: 10.3844/jcssp.2009.614.618

Verkhovsky, B.S., 2009. Integer factorization: solution via algorithm for constrained discrete logarithm problem. J. Comput. Sci., 5: 674-679. DOI: 10.3844/jcssp.2009.674.679