Cryptanalysis of a technique to transform discrete logarithm based cryptosystems into identity-based cryptosystems

Qiang Tang and Chris J. Mitchell

Technical Report
RHUL–MA–2005–4
1 March 2005

Department of Mathematics
Royal Holloway, University of London
Egham, Surrey TW20 0EX, England
http://www.rhul.ac.uk/mathematics/techreports

Abstract

In this paper we analyse a technique designed to transform any discrete logarithm based cryptosystem into an identity-based cryptosystem. The transformation method is claimed to be efficient and secure and to eliminate the need to invent new identity-based cryptosystems. However, we show that the identity-based cryptosystem created by the proposed transformation method suffers from a number of security and efficiency problems.

1 Introduction

In [1], Lee and Liao propose a transformation technique which is claimed to be able to transform any discrete logarithm based cryptosystem into an identity-based cryptosystem without modifying the original discrete logarithm based cryptosystem. They also claim that the identity-based cryptosystem created by the proposed transformation scheme (referred to here as the transformed ID-based system) retains all the advantages of identity-based cryptosystems. However, we show that the transformed ID-based system loses some of the advantages of general identity-based cryptosystems. Moreover, the transformation technique suffers from a number of security and efficiency problems.

The reminder of this paper is organised as follows. In Section 2, we review the transformation method. In Section 3, we describe certain security security and efficiency problems. In Section 4, brief conclusions are provided.

2 Review of the transformation scheme

The proposed transformation technique consists of two stages: the System setup stage and the Key generation stage. A trusted center (TC) is needed to generate the system parameters and the private keys for each registered entity.

The two stages of the transformation process pre-suppose the existence of a public key cryptosystem based on logarithms to the base g modulo p, where p is a large prime, q is a large prime dividing $p-1$, and g has multiplicative order q modulo p. That is, we assume that key pairs in this cryptosystem are of the form $(s, g^s \bmod p)$, where s (the private key) is a random value satisfying $0 < s < q$. The transformation process enables each user of the system to be provided with a key pair for this public key cryptosystem, where every user's public key is based on his identity. Note that every user of the transformed system should check that a claimed identity possesses the appropriate format (see below) before using it to generate a public key — incorrect identity strings should be rejected.

The scheme works as follows.

1. System setup stage

   (a) The TC chooses a threshold value t, where $t < log_2^q$ and it is assumed that no group of t or more entities in the system will collude to attack the system. The security parameter t also determines the minimum bit-length of each entity's identity number.

   (b) The TC chooses a private key x $(0 < x < q)$; $y = g^x \bmod p$ is the corresponding public key which is published by the TC.

   (c) Let $\{k_1, k_2, k_3, \cdots, k_t\}$ be secret information randomly chosen by TC, where $\sum_{i=1}^{t} k_i < q$ and $\sum_{i=1}^{j-1} k_i < k_j$, for every j $(1 < j \leq t)$. TC publishes the corresponding public information $\{K_1$, K_2, K_3, $\cdots$, $K_t\}$, where $K_i = g^{k_i} \bmod p$, for $i = 1$, 2, $\cdots$, t.

   (d) Each entity A has a unique t-bit identity $ID_A = (\mathrm{ID}_{A1}$, ID_{A2}, $\cdots$, $\mathrm{ID}_{At})$, where $\mathrm{ID}_{Ai} \in \{0, 1\}$, for $i = 1$, 2, $\cdots$, t. Each identity is assumed to adhere to pre-specified formatting rules, although examples of such rules are not given in [1].

2. Key generation stage

   Suppose user A wants to join the system. The TC and A carry out the following procedure to generate A's private key.

   (a) A sends TC his identity $\mathrm{ID}_A = (\mathrm{ID}_{A1}$, ID_{A2}, $\cdots$, $\mathrm{ID}_{At})$.

   (b) The TC checks whether the identity ID_A conforms to the specified format. If so, then TC uses his secret information to compute $k_A = \sum_{i=1}^{t} k_i \mathrm{ID}_{Ai} \bmod q$ (note that this corrects a typographical error in [1]), and $\sigma_A = x + K_A k_A \bmod q$, where $K_A = \prod_{i=1}^{t} K_i^{\mathrm{ID}_{Ai}} \bmod p$.

   (c) TC secretly sends σ_A to A as A's private key.

   (d) User A checks whether the following equation holds, $g^{\sigma_A} = y K_A^{K_A} \bmod p$, where $K_A = \prod_{i=1}^{t} K_i^{ID_{Ai}} \bmod p$.

   (e) If the checks succeed then A accepts $(\sigma_A, Y_A = g^{\sigma_A})$ as his private/public key key pair, where $Y_A = y K_A^{K_A} \bmod q$ can be computed from a combination of A's identity and publicly available information (i.e. y and $\{K_1$, K_2, $\cdots$, $K_t\}$).

3 Comments on the proposed scheme

Lee and Liao [1] claim that their transformation scheme achieves both efficiency and security. However, we now show that both the security and efficiency of the transformed ID-based system are open to question.

1. The requirement for the values k_i to be super-increasing (specified in section 3 of [1]) is very dangerous and severely weakens the scheme. This is because it drastically reduces the possible ranges of these private values. To see why this is true, suppose that q has 160 bits (as per the discussion in section 2.1 of [1]), and hence $t < 160$. Suppose, moreover, that t is chosen to equal 159, the largest possible value, in order to maximise the number of possible users in the scheme.

   By the super-increasing requirement, we know that $k_1 < k_2$, $k_1 + k_2 < k_3$, $k_1 + k_2 + k_3 < k_4$, and so on. Hence $k_2 \geq k_1 + 1$, $k_3 \geq 2k_1 + 2$, $k_4 \geq 4k_1 + 4, \cdots, k^{159} \geq 2^{157}k_1 + 2^{157}$. But we also know that $k_1 + k_2 + \cdots + k_{159} < q < 2^{160}$, i.e. $2^{158}(k_1 + 1) < 2^{160}$, i.e. $k_1 < 3$. Hence k_1 can immediately be determined, given that K_1 is public! Finding other values of k_i (for small values of i) is then almost as simple.

   This implies that choosing the values of k_i to be super-increasing is extremely risky. In fact, this requirement was only imposed, as explained in section 3 of [1], to avoid the possibility that two users, with different identity vectors, share the same key pair. However, as long as t is sufficiently large, the probability of this happening is very small, and can therefore be safely ignored. Thus as long as t is at least 120, say, the requirement for super-increasing values of k_i can, and should, be dropped.

2. Recall that, in the key generation stage of the transform scheme, a user A's secret σ_A ia computed as $\sigma_A = x + K_A k_A \bmod q = x + K_A \sum_{i=1}^{t} k_i \mathrm{ID}_{Ai} \bmod q$, where K_A is a public value. It is easy to prove that any n ($n < t$) users can collude to compute the private key x of the TC if their identity vectors are linearly dependent in $GF(q)$. From the discussion in section 3 of [1], this would appear to be the reason why t-bit identity vectors are required to conform to a special format, i.e. to prevent a set of users deliberately choosing their identity vectors to be linearly dependent. Curiously, this special format is not specified in [1], and needs to be carefully chosen to prevent attacks based on linear dependencies.

   The probability of a randomly chosen set of up to t identity vectors being linearly dependent over $GF(q)$ is very small, given that q is very large. However, this probability could be made much larger if users were permitted to choose their identity vectors, e.g. to have very small weight. One possible approach to defining a special format would simply be to use a 160-bit hash function (see e.g. [2]) to convert identifiers to 160-bit vectors, thereby preventing malicious users choosing identifiers with special properties. Identity collisions would be prevented by probabilistic arguments.

3. The transformed ID-based system possesses efficiency problems, including the following.

   (a) In the transformed ID-based system, a user cannot freely choose his name and network address as his identity. Identities must be generated according to a certain format.

   (b) In the transformed ID-based system, each user either stores y and $\{K_1, K_2, K_3, \cdots, K_t\}$ or obtains them from a public directory in order to compute the public key for another user. The storage and communication complexity of public key discovery is thus in proportion to the number of users in the system.

4 Conclusions

In this paper, we have analysed a protocol which is designed to transform any discrete logarithm based cryptosystem into an identity-based cryptosystem. We show that the transformed ID-based system loses some of the advantages of a general identity-based system and also suffers from security and efficiency issues.

References

[1] W. B. Lee and K. C. Liao. Constructing identity-based cryptosystems for discrete logarithm based cryptosystems. *Journal of Network and Computer Applications*, 27:191–199, 2004.

[2] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone. *Handbook of Applied Cryptography.* CRC Press, 1997.