Another Method for Attaining Security Against Adaptively Chosen Ciphertext Attacks

Chae Hoon Lim and Pil Joong Lee

Department of Electrical Engineering,
Pohang University of Science and Technology (POSTECH)
Pohang, 790-784, KOREA

ABSTRACT Practical approaches to constructing public key cryptosystems secure against chosen ciphertext attacks were first initiated by Damgard and further extended by Zheng and Seberry. In this paper we first point out that in some cryptosystems proposed by Zheng and Seberry the method for adding authentication capability may fail just under known plaintext attacks. Next, we present a new method for immunizing public key cryptosystems against adaptively chosen ciphertext attacks. In the proposed immunization method, the deciphering algorithm first checks that the ciphertext is legitimate and then outputs the matching plaintext only when the check is successful. This is in contrast with the Zheng and Seberry's methods, where the deciphering algorithm first recovers the plaintext and then outputs it only when the checking condition on it is satisfied. Such a ciphertext-based validity check will be particularly useful for an application to group-oriented cryptosystems, where almost all deciphering operations are performed by third parties, not by the actual receiver.

1 Introduction

Recently much attention has been devoted to constructing public key cryptosystems secure against chosen ciphertext attacks, from the theoretical and practical points of view. Theoretically, non-interactive zero-knowledge proof was shown to be a nice tool for this purpose [3] [9] and several such concrete public key cryptosystems have been proposed [16] [18]. However, due to the enormous data expansion during the enciphering transformation, the resulting schemes are highly inefficient and thus no one would try to implement them in practice.

Practical approaches to this field were initiated by Damgard [7] and further extended by Zheng and Seberry [25]. The key idea of Damgard's approach is to construct a public key cryptosystem in such a way that an attacker cannot produce

D.R. Stinson (Ed.): Advances in Cryptology - CRYPTO '93, LNCS 773, pp. 420-434, 1994.

legitimate ciphertexts (i.e., the ciphertexts whose plaintexts he can get from the deciphering oracle) without knowing the plaintext. This makes useless the attacker's ability to gain access to the deciphering oracle under chosen ciphertext attacks. Based on this idea, Damgard [7] proposed simple methods for modifying any deterministic public key cryptosystems and the ElGamal/Diffie-Hellman cryptosystem so that the resulting cryptosystems may be more secure under chosen ciphertext attacks. Later, by refining Damgard's idea and combining the probabilistic encryption technique [10], Zheng and Seberry [25] presented three practical methods for immunizing public key cryptosystems and proved that their cryptosystems are semantically secure against adaptively chosen ciphertext attacks under reasonable assumptions.

In this paper, we first point out that in some cryptosystems presented by Zheng and Seberry the method for adding authentication capability may fail just under known plaintext attacks. Next we propose a new method for immunizing public key cryptosystems, which is illustrated by constructing cryptosystems based on the Diffie-Hellman/ElGamal scheme and the RSA scheme. In the modified cryptosystems, the deciphering algorithm first checks that ciphertexts are properly constructed according to the enciphering algorithm and only when the check is successful, does it output the matching plaintexts. A main difference of our approach from that of Zheng and Seberry is that it is determined based on the ciphertext, not on the recovered plaintext, whether or not the deciphering algorithm outputs the result.

Such a ciphertext-based validity check is especially useful for an application to group-oriented cryptosystems [8]. In group-oriented cryptosystems, ciphertexts are usually accompanied by the indicator indicating the nature of the ciphertexts and all or substantial part of deciphering operations are performed independently of the actual receiver(s). Then the partial computations are distributed to the legitimate receiver(s) according to the indicator and the security policy of the receiving company. Thus, the main threat is an illegal modification of the indicator by an inside group member who violates the security policy and tries to read the ciphertexts. This requires a concrete scheme for combining the ciphertexts and the indicator so that no one can produce a legal ciphertext by modifying the intercepted ciphertext, especially by changing the indicator. The proposed cryptosystem is well suited for this application.

This rest of this paper is organized as follows. Section 2 briefly mentions probabilistic encryption and pseudorandom number generators. Section 3 discusses the previous works of Damgard and Zheng-Seberry in this field. Here, we also point out some weakness of Zheng and Seberry's method for adding authentication capability to their cryptosystems. In section 4, applying the proposed immunization method, we present two cryptosystems based on the Diffie-Hellman/ElGamal scheme and the RSA scheme and analyze their security. Finally we conclude in section 5.

2 Random Number Generators and Probabilistic Encryption

Goldwasser and Micali [10] presented a general scheme for constructing public key probabilistic encryption schemes which hide all partial information, in the sense that whatever is efficiently computable about the plaintext given the ciphertext is also efficiently computable without the ciphertext. (This is an informal definition of semantic security which can be thought of as a polynomially bounded version of Shannon's perfect secrecy. See [13] for other equivalent notions of security for public key cryptosystems.) These encryption schemes can be thought of as the best we are seeking for, as far as passive attacks are concerned, since a polynomially bounded passive attacker can extract no information on the plaintexts from the ciphertexts. They also gave a concrete implementation under the intractability assumption of deciding quadratic residuosity modulo a large composite number. However, their scheme expands each plaintext bit into a ciphertext block of length of the composite modulus and thus is highly inefficient.

Cryptographically strong pseudorandom number generators whose notion was first introduced by Blum and Micali [5] and extended by Yao [24] is one of the most powerful tools in many cryptographic applications. The output sequences produced by such a generator cannot be distinguished by a polynomial-time algorithm from truly random sequences of the same length (such a generator is said to be *perfect*). Thus these generators can be used for constructing more efficient probabilistic encryption schemes, as first illustrated by Blum and Goldwasser [4] : "Send the exclusive-or of a message sequence with an output sequence of the same length of a pseudorandom number generator, together with a public key encryption of a random seed used." Consequently, cryptosystems constructed like this can be proved to be secure against any passive attacks (e.g., see [4] for detailed proof) and thus as far as passive attacks are concerned, the problem of constructing a secure public key cryptosystem is settled. Furthermore, the plaintext is only expanded by a constant factor in this case, the portion of public key encryption of a random seed used.

Several perfect pseudorandom number generators have been established. Long and Wigderson [12] generalized the Blum-Micali's generator [5] based on the discrete logarithm problem and showed that O(log k) bits can be securely produced per each exponentiation where k is the bit-length of a modulus. The same result was obtained by Peralta [17] with different technique. Alexi et al. [1] showed that RSA/Rabin function can hide O(log k) bits under the intractability assumption of RSA encryption and factoring. Vazirani and Vazirani [22] showed that O(log k) bits can be securely extracted from the $x^2 \bmod N$ generator of Blum, Blum and Shub [2] as well as from the RSA/Rabin functions. Recently Micali and Schnorr [14] developed a very

efficient polynomial random number generator which can be based on an arbitrary prime modulus as well as on RSA modulus. This generator can produce more than k/2 bits per iteration at a cost of about one full modular multiplication, though it is open whether the generator with this efficiency is perfect. This, if perfect, will lead to very efficient probabilistic encryption schemes which hide all partial information.

3 Overview and Discussion of Previous Works

3.1 Damgard's Approach

A main drawback of probabilistic encryption schemes is that while being provably secure against any passive attacks, they can be completely broken under chosen ciphertext attacks. In a chosen ciphertext attack, an attacker can query the deciphering oracle with polynomially many ciphertexts, and use the information obtained from the answers to extract any useful information for the target ciphertext.

Recently, Damgard [7] made a first step into the research of practical public key cryptosystems secure against chosen ciphertext attacks. His key idea is to modify a public key cryptosystem in such a way that an attacker cannot produce ciphertexts whose plaintexts he can get from the deciphering oracle unless he starts by first choosing the plaintext. This nullifies the ability to have access to the deciphering oracle under chosen ciphertext attacks. Based on this idea, he presented two concrete examples of public key cryptosystems which appear to be secure against chosen ciphertext attacks, one using any deterministic public key cryptosystems and the other using the Diffie-Hellman /ElGamal public key cryptosystem.

To get a better insight into the Damgard's approach, we briefly describe his second scheme based on the Diffie-Hellman/ElGamal public key cryptosystem. Let a user A's secret key be a pair (x_{A1}, x_{A2}) of elements chosen at random over [1, p-1] and the corresponding public key be a pair (y_{A1}, y_{A2}), where $y_{A1} \equiv_p g^{x_{A1}}$ and $y_{A2} \equiv_p g^{x_{A2}}$. (Here, "$Y \equiv_p X$" denotes "Y is congruent to X in mod p".) Then the ciphertext for a message m to be sent to user A consists of a triple (c_1, c_2, c_3) :

$$c_1 \equiv_p g^r, \; c_2 \equiv_p y_{A1}{}^r \text{ and } c_3 = m \oplus (y_{A2}{}^r \bmod p),$$

where r is uniform in [1, p-1] and the symbol $\oplus$ denotes the bit-wise exclusive-or operation. The deciphering algorithm by user A who has the secret key (x_{A1}, x_{A2}) is as follows :

$$m = c_3 \oplus (c_1{}^{x_{A2}} \bmod p) \text{ if } c_2 \equiv_p c_1{}^{x_{A1}}, \text{ NULL otherwise.}$$

Here, NULL is a special symbol used for meaning "no plaintext output".

The intuitive reason of the security against chosen ciphertext attacks is that given

g and y_{A1}, it seems hard to generate a pair ($g^r \bmod p$, $y_{A1}^r \bmod p$), unless one starts by simply choosing r, which in turn implies that it is hard for an attacker to generate a legitimate ciphertext (on which the deciphering algorithm produces a non-null output), unless he already knows the plaintext. Therefore, this modified ElGamal cryptosystem will be secure against chosen ciphertext attacks, if we assume that the original ElGamal is secure against passive attacks and that there is no other way to produce ciphertexts than to first choose r. This approach suggests a method of gaining more security against chosen ciphertext attacks : "Modify a public key cryptosystem in such a way that it is infeasible to generate a legitimate ciphertext without first choosing the plaintext. Then the modified cryptosystem will be as secure under a chosen ciphertext attack as under a passive attack."

Here, we have to note that Damgard considered a restricted model for chosen ciphertext attacks, known as indifferently chosen ciphertext attacks (also called a lunchtime attack or a midnight attack), where an attacker has access to the deciphering oracle only *before* seeing the ciphertext he attempts to decrypt itself. This is a less satisfying model of attacks inherent in real-life applications of cryptosystems, as illustrated by Zheng and Seberry [25]. One of the most severe type of attack against a public key cryptosystem is an adaptively chosen ciphertext attack, where an attacker is allowed to have access to the deciphering algorithm even *after* seeing the target ciphertext. Note that, in an adaptively chosen ciphertext attack, an attacker can feed the deciphering algorithm with the ciphertexts correlated to the target ciphertext and obtain the matching plaintexts. Consequently, the Damgard's scheme described above can be completely broken under this model of attacks as follows : For a given ciphertext $c = (c_1, c_2, c_3)$, an attacker feeds the deciphering algorithm with the modified ciphertext $c' = (c_1, c_2, c_3')$ where $c_3' = c_3 \oplus r$ with a random message r. Then he will get a message $m' = m \oplus r$ as an answer and thus can obtain the desired message m by computing $m = m' \oplus r$.

3.2 Zheng and Seberry's Extension

Zheng and Seberry [25] further extended and generalized the Damgard's approach in order to attain security against adaptively chosen ciphertext attacks. Extending Damgard's idea, they introduced the notion of sole-samplability, defined informally as follows : The space induced by function $f : D \rightarrow R$ is said to be *sole-samplable* if there is no other way to generate an element y in R than to first choose an element x in D and then to evaluate the function at the point x. This notion is very similar to the assumption used by Damgard in proving the security of his modified ElGamal cryptosystem under chosen ciphertext attacks. However, according to this notion, for the enciphering transformation to be a sole-samplable function, the whole ciphertext

should be hard to generate without knowing the plaintext. Note that the space induced by the enciphering algorithm of Damgard's modified ElGamal cryptosystem is not sole-samplable due to the last part c_3 of the ciphertext.

Using different techniques in order to approximate sole-samplability of the enciphering transformation, they presented three methods for immunizing public key cryptosystems against adaptively chosen ciphertext attacks. Thanks to the generation of ciphertexts in a sole-samplable way along with probabilistic encryption, they could attain semantic security of their cryptosystems against adaptively chosen ciphertext attacks, under reasonable assumptions. More generally, they proved the following : "Assume that the space induced by the enciphering algorithm of a public key cryptosystem is sole-samplable. Then the cryptosystem is semantically secure against adaptively chosen ciphertext attacks, if it is semantically secure against chosen plaintext attacks." This is a quite obvious consequence resulting from the sole-samplability assumption on the enciphering transformation.

The main point of Zheng and Seberry's immunization methods is to make the enciphering transformation into a sole-samplable function by appending to each ciphertext a tag computed as a function of the message to be enciphered, much as a manipulation detection code (MDC) under encipherment or a message authentication code (MAC) in clear is used for message authentication. Note that a MDC is computed solely as a function of the message and transmitted under encipherment, while a MAC is computed from the message using a secret key and transmitted in clear [11]. The three immunization methods proposed by Zheng and Seberry differ only in the ways of generating tags. That is, they applied three basic tools that can be used to realize message authentication : one-way hash functions, universal classes of hash functions [6] [23] and digital signature schemes. Among them, the method based on one-way hash function (the resulting cryptosystem is denoted by C_{owh}) is first explained, since it is very simple, but most reflects the approach taken by them.

Assume h hashes arbitrary input strings into t-bit output strings. In this method, the ciphertext for an n-bit message m consists of a pair (c_1, c_2), where c_2 is the exclusive-or of the (n+t)-bit concatenated message $m \| h(m)$ with an (n+t)-bit output sequence of a pseudorandom number generator on a secret seed s and c_1 is a public key encryption of the seed s. In the decryption process, the deciphering algorithm first recovers a message $m' \| h(m)'$ from the ciphertext c_2 and outputs m' as the matching plaintext only when $h(m') = h(m)'$. Due to the involvement of a tag h(m), it is reasonable to assume that a polynomially bounded adaptively chosen ciphertext attacker cannot produce a ciphertext whose plaintext passes the check of the deciphering algorithm. This justifies the sole-samplability assumption and thus the ability to have access to the deciphering oracle gives no advantage to the attacker. Therefore, the cryptosystem is as secure under adaptively chosen ciphertext attacks as

under chosen plaintext attacks. Now its semantic security against adaptively chosen ciphertext attacks follows immediately from the fact that the cryptosystem can be proved to be semantically secure under chosen plaintext attacks as in the Blum and Goldwasser's scheme [4].

Next we describe in more details the cryptosystem C_{sig} which is based on an adaptation of digital signature schemes, since it will be a basis of our proposed method. Let p be a large prime and let g be a generator of the multiplicative group $GF(p)^*$. A user A possesses a secret key $x_A \in_R [1, p-1]$ and the corresponding public key is computed as $y_A \equiv_p g^{x_A}$. Let G(n, s) be an n-bit output sequence produced on seed s by a pseudorandom number generator based on the intractability of computing discrete logarithms in finite fields [5] [12] [17] [14]. Assume that a user B wants to send in secret an n-bit message m to A. Then the enciphering and deciphering algorithms are as follows.

Enciphering Algorithm (user B) :

i) Choose $r_1, r_2 \in_R [1, p-1]$, where $\gcd(r_2, p-1) = 1$.

ii) Compute $s \equiv_p y_A^{r_1+r_2}$ and $z = G(n, s)$.

iii) Compute $c_1 \equiv_p g^{r_1}$, $c_2 \equiv_p g^{r_2}$,

$$c_3 \equiv_{p-1} (h(m) - sr_1)/r_2, \text{ and } c_4 = z \oplus m.$$

iv) Send (c_1, c_2, c_3, c_4) to user A.

Deciphering Algorithm (user A) :

i) Compute $s' \equiv_p (c_1 c_2)^{x_A}$ and $z' = G(n, s')$ where $n = |c_4|$.

ii) Recover a plaintext m' by $m' = z' \oplus c_4$.

iii) Check that $g^{h(m')} \equiv_p c_1^{s'} c_2^{c_3}$.

If ok, output m' ; Else, output NULL.

The first three parts (c_1, c_2, c_3) of the ciphertext correspond to an adaptation of the ElGamal's signature scheme. Assuming that the hash function h produces output with almost uniform distribution, the ciphertext also leaks no partial information on the message m. Note that a tag is generated like a MDC in C_{owh} whereas in C_{sig} it is generated much like a MAC. Also note that in all three methods of Zheng and Seberry the validity check is based on the recovered plaintexts. In section 4, we will present a new immunization method using a ciphertext-based validity check.

3.3 Problem of Zheng-Seberry's Authentication Method

Zheng and Seberry also presented the method for adding authentication capability

to their cryptosystems. Here, we point out that in their cryptosystems C_{owh} and C_{uhf} (based on universal class of hash functions), their method may fail to provide this capability under known plaintext attacks. First note that authentication scheme fails when a user different from the sender can create a message which the receiver will accept as being originated from the sender and that its security may be independent of the security of the cryptosystem used for secrecy. The reason of authentication failure in these two schemes is that tags are computed just as a function of the message to be enciphered and/or a pseudorandom sequence used for encryption, both of which are available under known plaintext attacks. This makes it possible for an attacker to reuse a pseudorandom sequence obtained from a ciphertext-plaintext pair to encrypt and falsely authenticate his chosen message.

Let's first consider the cryptosystem C_{owh} in which the ciphertext for an n-bit message m consists of (c_1, c_2) where $c_1 \equiv_p g^r$, $c_2 = G(n+t, s) \oplus (m \| h(m))$ and the secret seed s is computed by $s \equiv_p y_A^r$. As suggested by Zheng and Seberry, authentication capability may be added to this cryptosystem by the sender B computing a seed s as $s \equiv_p y_A^{r+x_B}$ where x_B is the secret key of user B. But in this case an attacker knowing the plaintext m for this ciphertext (mounting known plaintext attacks, for example) can obtain the (n+t)-bit pseudorandom sequence G(n+t, s) by simply exclusive-oring $m \| h(m)$ with c_2 (i.e., $G(n+t, s) = c_2 \oplus (m \| h(m))$). Then he will be able to generate a legal ciphertext for his chosen message m' of length $n' \leq n$ as (c_1, c_2') where $c_2' = G(n'+t, s) \oplus (m' \| h(m'))$. Here, the pseudorandom sequence G(n'+t, s) is just the first (n'+t) bits of G(n+t, s). Clearly this ciphertext will be correctly deciphered and the receiver A will accept the plaintext m' as a valid message sent by user B.

The same attack can be applied to the cryptosystem C_{uhf} as well. To send an n-bit message m to user A using the cryptosystem C_{uhf}, a user B generates the ciphertext $c = (c_1, c_2, c_3)$ as: $c_1 \equiv_p g^r$, $c_2 = h_u(m)$, and $c_3 = z \oplus m$. Here, z denotes the first n bits of a pseudorandom sequence G(n+k, s) of length (n+k) produced on seed $s \equiv_p y_A^r$ and u denotes the remaining k bits of G(n+k, s). The function h_u denotes a hash function specified by a string u in a universal class of hash functions mapping n-bit input into t-bit output [6] [21] [23]. On receiving the ciphertext c, user A (the deciphering algorithm) computes $s' \equiv_p c_1^{x_A}$, generates $z' \| u' = G(n+k, s')$, computes $m' = z' \oplus c_3$ and finally checks that $h_{u'}(m') = c_2$. Only when the check is successful, does it output m' as the matching plaintext. Now consider the case where user B computes a secret seed s as $s \equiv_p y_A^{r+x_B}$ to provide authentication in addition, as Zheng and Seberry suggested, and suppose that an attacker obtained the ciphertext-plaintext pair (c, m). Then the attacker can extract from this pair the pseudorandom sequence z

of length n. Therefore, as in C_{owh}, he can generate a legitimate ciphertext for a message of length $n' \leq n-k'$ where k' is the key length needed to specify a hash function in a universal class of hash functions mapping n'-bit input into t-bit output.

Finally, consider possible countermeasures against the described attack. First note that tags are computed as a function of the message alone in C_{owh} or as a function of the message and pseudorandom sequence in C_{uhf}, while in C_{sig} the random numbers used to generate a seed are also involved in generating a tag. Therefore, we can see that to defeat the attack, either a secret seed itself or random numbers used to compute a seed should also be involved in generating a tag. Simple countermeasure may be such that h(m) (resp. $h_u(m)$) is replaced by h(s || m) (resp. $h_u(s \| m)$) where in C_{owh} the secret seed s may be used as an initialization variable of the hash function h.

4 Proposed Immunization method

4.1 Motivation

Before presenting our immunization method, we first give the motivation that drives us to devise such a system that the decision on deciphering can be made solely based on the ciphertext. First recall that in an adaptively chosen ciphertext attack, an attacker can query the deciphering algorithm with any ciphertexts, *except* for the target ciphertext to decipher. However, in some applications, the attacker may feed the target ciphertext itself into the deciphering oracle and directly obtain the corresponding plaintext. This is seemingly a meaningless attack, but such a case may arise in group-oriented cryptosystems [8].

In group-oriented cryptosystems, the name of the destined receiver (or an indicator denoting the nature of the ciphertext, according to which the receiving group processes the ciphertext and distributes the partial results so that the legitimate group member can read the message) is usually accompanied by the ciphertext and in particular all or substantial part of the deciphering operations are carried out apart from the actual receiver. This separation of deciphering process from the actual receiver may make inside attacks easy, unless the ciphertext and the receiver's name are inalterably combined. This is because anyone inside the group can intercept the ciphertext not directed to him and then can change the receiver's name into his, if necessary in collusion with an outside colleague. Then all partial computations for decryption will be sent to him and thus he can decipher it. In fact, this inside attack may be mounted independently of the security of cryptosystems used. Ordinary authentication or digital signature schemes do not help to prevent an illegal modification of the receiver's name, since the modification should be detected, before

decryption, by third parties not knowing the message. All that is required is to adapt the cryptosystem in such a way that any change in ciphertexts including the receiver's name can be detected before decryption by any third parties.

Motivated by the case considered above, we present a ciphertext-based immunization method and illustrate it by examples of the Diffie-Hellman /ElGamal scheme and the RSA scheme in the following two subsections. In the proposed schemes, any attempt to illegally modifying ciphertexts can be detected at the start of the deciphering process, which makes useless adaptively chosen ciphertext attacks.

4.2 Immunizing Diffie-Hellman/ElGamal Cryptosystem

Let p be a large prime (say, $\geq$ 512 bits) such that t-bit prime q divides p-1 (for example, t = 160) and let α be a generator of the unique subgroup GF(q)* of the multiplicative group GF(p)*. Let h be a one-way hash function hashing arbitrary input strings into t-bit output strings (for example, secure hash standard [20]). Denote by G(n, s) an n-bit output sequence produced on a secret random seed s by a cryptographically strong pseudorandom number generator such as [12], [14] and [17]. As before, each user A possesses a secret key $x_A \in_R GF(q)^*$ and let $y_A \equiv_p \alpha^{x_A}$ be the corresponding public key. Assume user B wants to send in secret an n-bit message m to user A. Then they can proceed as follows.

Enciphering Algorithm (user B) :

i) Choose $r_0, r_1 \in_R [1, q-1]$.

ii) Compute $c_0 \equiv_p \alpha^{r_0}$, $c_1 \equiv_p \alpha^{r_1}$ and $s \equiv_p y_A^{r_1} c_0$.

iii) Compute $z = G(n, s)$, $c_2 = z \oplus m$, $c_3 = h(c_0 \| c_2)$ and $c_4 \equiv_q r_0 + c_3 r_1$.

iv) Send $c = (c_1, c_2, c_3, c_4)$ to user A.

Deciphering Algorithm (user A) :

i) Check that $c_3 = h(c_0' \| c_2)$ where $c_0' \equiv_p \alpha^{c_4} c_1^{-c_3}$.
If ok, continue ; Else, stop and output NULL.

ii) Compute $s \equiv_p c_1^{x_A} c_0'$ and $z = G(n, s)$ where $n = |c_2|$.

iii) Output m such that $m = z \oplus c_2$.

For efficiency reason, we applied the Schnorr's signature scheme [19], with a secret key chosen at random. If the pseudorandom number generator requires a generator g of GF(p)*, such a g can be published in addition.

Security against chosen plaintext attacks : Since among the ciphertext the only

message embedding part is $c_2 = z \oplus m$, we know that as far as it is computationally infeasible to compute the seed s from the ciphertext, no partial information on m will be released to a polynomial-time chosen plaintext attacker. First an algorithm for computing $s \equiv_p y_A^{r_1}\alpha^{r_0}$ from $y_A \equiv_p \alpha^{x_A}$, $c_1 \equiv_p \alpha^{r_1}$, $c_3 \in_R [1, 2^t-1]$ and $c_4 \equiv_q r_0 + c_3 r_1$ can be shown to be used to solve the Diffie-Hellman problem of computing $v \equiv_p \alpha^{x_1 x_2}$ from $v_1 \equiv_p \alpha^{x_1}$ and $v_2 \equiv_p \alpha^{x_2}$: On inputs $y_A = v_1$, $c_1 = v_2$, $c_3 = k_1 \in_R [1, 2^t-1]$ and $c_4 = k_2 \in_R GF(q)^*$, the algorithm will output $s \equiv_p v_1^{x_2}\alpha^r$ where r is an element of $GF(q)^*$ uniquely determined (by the algorithm) from the equation $k_2 \equiv_q r + k_1 x_2$. Therefore, one can compute $(v_1^{k_2} s^{-1})^{(k_1-1)^{-1}} \equiv_p v_1^{x_2} \equiv_p \alpha^{x_1 x_2}$, the desired result. Here we note that the relation $c_3 = h(c_0 \| c_2)$ where $c_0 \equiv_p \alpha^{c_4} c_1^{-c_3}$ does not affect the difficulty of computing a seed from the ciphertext. This shows that under the Diffie-Hellman assumption it is computationally infeasible to compute the seed s from the ciphertext and thus the cryptosystem is semantically secure against chosen plaintext attacks.

Security against adaptively chosen ciphertext attacks : In the following, we show that a polynomial-time attacker can extract no additional information on the plaintext, even if he is given the ability to have access to the deciphering algorithm under adaptively chosen ciphertext attacks. Then, the proposed cryptosystem will be as secure under adaptively chosen ciphertext attacks as under chosen plaintext attacks.

First we note that, in order to obtain any useful information on the plaintext corresponding to the ciphertext $c = (c_1, c_2, c_3, c_4)$ from the accessibility to the deciphering oracle, an attacker must generate a ciphertext $c' = (c_1', c_2', c_3', c_4')$ so that the following two conditions are satisfied at the same time : First, the ciphertext c' must satisfy the checking condition of step i) of the deciphering algorithm to get a non-null output. Second, the seed computed from the modified ciphertext by the deciphering algorithm must be equal to the seed used to produce the original ciphertext, since otherwise the output will be just another pseudorandom string indistinguishable from the truly random string. The second condition requires that no change in c_0 and c_1 should be made. Consequently, the attacker must solve the congruence equation $c_0 \equiv_p \alpha^{c_4} c_1^{-c_3} \equiv_p \alpha^{c_4'} c_1^{-c_3'}$ where $c_3' = h(c_0 \| c_2')$. Now solving this equation can be easily shown to be as difficult as solving the discrete logarithm problem : To solve $y \equiv_p \alpha^x$ in x using an algorithm for solving the equation, we provide as inputs $c_0 = c_1 = y$. Then from the outputs c_2' and c_4', we can compute the desired logarithm $x \equiv_q (1 + h(c_0 \| c_2'))^{-1} c_4'$.

In the above, we have shown that a polynomially bounded attacker gains no advantage from the accessibility to the deciphering oracle under adaptively chosen

ciphertext attacks. This, together with semantic security under chosen plaintext attacks, shows that the proposed cryptosystem is semantically secure against adaptively chosen ciphertext attacks, under the Diffie-Hellman assumption.

The main difference of our immunization method from that of Zheng and Seberry is that a signature is generated based on the ciphertext, not on the message m. Consequently, everyone can check that ciphertexts are properly constructed according to the enciphering algorithm. This property may be useful in many applications, as exemplified by an application to group-oriented cryptosystems in subsection 4.1.

Adding authentication capability : Authentication capability is easily incorporated into the system : just replace (r_1, c_1) by the secret key/public key pair (x_B, y_B) of the sender (user B). In this case, the last two parts (c_3 , c_4) of the ciphertext constitute a signature for the message embedding part c_2 of the ciphertext. Though the signature is generated on the ciphertext, not on the message m, generally considered as a bad practice for signing, it does not raise any problem as a signature for the message m (for example, the authorship problem raised in the CCITT X.509 token structure [15]). This is because message encryption and signature generation are tightly combined through the same random number r_0, which ensures that no one can produce a signature for the ciphertext c_2 without knowing the plaintext m.

Some additional cleartexts may be included as arguments of the hash function, e.g., the name of the sender and receiver, time information, and especially the indicator of the ciphertext when used for a group-oriented cryptosystem. As shown above, these cleartexts cannot be modified by an attacker. In particular, time information may be used to prevent the playback attack within the (predetermined) clock skew limit. We think that adding these additional cleartexts can provide much convenience in most communications, since the source and destination and the timeliness of a ciphertext can be easily verified by using only the ciphertext.

4.3 Immunizing RSA Cryptosystem

Let $N_A = p_A q_A$ be the modulus of user A in the RSA scheme, where p_A and q_A are large primes of the same size. Each user A chooses the public exponent e_A as a t-bit prime (t = 64, for example) and keeps secret d_A such that $e_A d_A = 1 \bmod \phi(N_A)$ where ϕ denotes the Euler phi function. Let h be a one-way hash function hashing arbitrary input strings into output values less than e_A. Let G(n, s) be the same as before. But it can be based on the modulus N_A of the receiver, such as the RSA/Rabin scheme based generators [1] [14] or the $x^2 \bmod N$ generator [2] [22]. Of course, a common, possibly standardized, pseudorandom number generator may be used independently of the individual modulus. Assume that user B wants to send user A an n-bit message

m. Then the enciphering and deciphering algorithms are as follows.

Enciphering Algorithm (user B) :

i) Choose $s \in_R [1, N_A-1]$.

ii) Compute $c_1 = s^{3e_A} \bmod N_A$ and $z = G(n, s)$.

iii) Compute $c_2 = z \oplus m$, $c_3 = h(c_1 \| c_2)$ and $c_0 = s^{3c_3} \bmod N_A$.

iv) Send $c = (c_0, c_1, c_2)$ to user A.

Deciphering Algorithm (user A) :

i) Check that $c_0^{e_A} = c_1^{c_3'} \bmod N_A$ where $c_3' = h(c_1 \| c_2)$.
If ok, continue ; Else, stop and output NULL.

ii) Compute $s = c_1^{d_A'} \bmod N_A$ and $z = G(n, s)$,
where $d_A' = 3^{-1} d_A \bmod \phi(N_A)$ and $n = |c_2|$.

iii) Output m such that $m = z \oplus c_2$.

With the same argument as before, under the assumption that RSA is secure, the ciphertext (c_0, c_1, c_2) leaks no partial information on m and thus the cryptosystem is semantically secure against chosen plaintext attacks. Under adaptively chosen ciphertext attacks, an attacker should not change the second part c_1 of the ciphertext in order to get a useful output from the accessibility to the deciphering algorithm. Therefore the attacker is faced with the problem of solving $c_0^{e_A} = c_1^{h(c_1 \| c_2)} \bmod N_A$ in c_0 and c_2 for a fixed c_1. Solving this equation can be shown to be at least as difficult as inverting the RSA function : To compute x such that $y = x^{e_A} \bmod N_A$, one provides as inputs e_A, N_A and y to an algorithm for solving the equation, which then will output c_0 and c_2 such that $c_0 = x^{h(y \| c_2)} \bmod N_A$. Now one can easily compute the desired number x from two equations $y = x^{e_A} \bmod N_A$ and $c_0 = x^{h(y \| c_2)} \bmod N_A$ using the extended Euclidean algorithm, since e_A and $h(y \| c_2)$ is relatively prime. This shows that an algorithm for computing c_0 and c_2 from $c_0^{e_A} = c_1^{h(c_1 \| c_2)} \bmod N_A$ for a fixed c_1 can be used to invert the RSA function. Therefore, we have shown that under the assumption that RSA is secure, the above cryptosystem can be proved to be secure against adaptively chosen ciphertext attacks.

The cryptosystem can also provide authentication capability. This can be done by the sender generating c_0 as follows. Assume user B also has the RSA keys, public key (N_B, e_B) and secret key d_B. Then he can compute c_0 by $c_0 = (s^{3c_3} \bmod N_A)^{d_B} \bmod N_B$ where we assume that $N_B > N_A$. The checking condition can be changed accordingly : check that $(c_0^{e_B} \bmod N_B)^{e_A} = c_1^{c_3'} \bmod N_A$ with $c_3' = h(c_1 \| c_2)$.

5 Conclusion

This paper presented a new method for immunizing public key cryptosystems against adaptively chosen ciphertext attacks, which is illustrated by examples of the Diffie-Hellman/ElGamal cryptosystem and the RSA cryptosystem. In the proposed immunization method, everyone can check that ciphertexts are properly constructed according to the enciphering algorithm. This property is particularly useful for an application to group-oriented cryptosystems, where deciphering operations are separated from the actual receiver and the main threat is an illegal modification of the indicator by an inside group member. We also pointed out that in some of the cryptosystems presented by Zheng and Seberry the method for adding authentication capability may fail just under known plaintext attacks and presented a simple countermeasure.

References

[1] W.Alexi, B.Chor, O.Goldreich and C.P.Schnorr, *RSA and Rabin functions : certain parts are as hard as the whole*, SIAM J. Computing vol.17 no.2 (1988), 194-208.

[2] L.Blum, M.Blum and M.Shub, *A simple unpredictable pseudo-random number generator*, SIAM J. Computing vol.15 no. 2 (1986), 364-383.

[3] M.Blum, P.Feldman and S.Micali, *Non-interactive zero-knowledge proof systems and applications*, Proc. 20th Annual ACM Symposium on Theory of Computing (STOC) (1988), 103-112.

[4] M.Blum and S.Goldwasser, *An efficient probabilistic public key encryption scheme which hides all partial information*, Advances in Cryptology - Crypto'84, Lecture Notes in Computer Science vol.196, Springer-Verlag (1985), 289-299.

[5] M.Blum and S.Micali, *How to generate cryptographically strong sequences of pseudo-random bits*, SIAM J. Computing vol.13 no.4 (1984), 850-864.

[6] J.Carter and M.Wegman, *Universal classes of hash functions*, J. Computer and System Sciences vol.18 (1979), 143-154.

[7] I.Damgard, *Towards practical public key systems secure against chosen ciphertext attacks*, Advances in Cryptology - Crypto'91, LNCS vol.576, Springer-Verlag (1992), 445-456.

[8] Y.Desmedt, *Society and group oriented cryptography : a new concept*, Advances in Cryptology - Crypto'87, LNCS vol.293, Springer-Verlag (1988), 120-127.

[9] Z.Galil, S.Haber and M.Yung, *Symmetric public key cryptosystems*, submitted to J. Cryptology.

[10] S.Goldwasser and S.Micali, *Probabilistic encryption*, J. Computer and System Sciences vol.28 no.2 (1984), 270-299.

[11] R.R.Jueneman, S.M.Matyas and C.H.Meyer, *Message authentication with manipulation detection codes*, Proc. 1983 Symposium on Security and Privacy, IEEE Computer Society Press, 33-54.

[12] D.L.Long and A.Wigderson, *The discrete logarithm hides O(log n) bits*, SIAM J. Computing vol.17 no.2 (1988), 363-372.

[13] S.Micali, C.Rackoff and B.Sloan, *The notion of security for probabilistic cryptosystems*, SIAM J. Computing vol.17 no.2 (1988), 412-426.

[14] S.Micali and C.P.Schnorr, *Efficient, perfect polynomial random number generators*, J. Cryptology vol.3 no.3 (1991), 157-172.

[15] C.Mitchell, M.Walker and D.Rush, *CCITT/ISO standard for secure message handling*, IEEE J. Selected Areas on Commun. vol.7 no.4 (1989), 517-524.

[16] M.Naor and M.Yung, *Public key cryptosystems provably secure against chosen ciphertext attacks*, Proc. 22th Annual ACM Symposium on Theory of Computing (STOC) (1990), 427-437.

[17] R.Peralta, *Simultaneous security of bits in the discrete log*, Advances in Cryptology - Eurocrypt'85, LNCS vol.219, Springer-Verlag (1986), 62-72.

[18] C.Rackoff and D.Simon, *Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attacks*, Advances in Cryptology - Crypto'91, LNCS vol.576, Springer-Verlag (1992), 433-444.

[19] C.P.Schnorr, *Efficient signature generation by smart cards*, J. Cryptology vol.4 no.3 (1991), 161-174.

[20] *A proposed federal information processing standard for secure hash standard*, Federal Register Announcement (Jan. 31, 1992), 3747-3749.

[21] D.R.Stinson, *Combinatorial techniques for universal hashing*, submitted to J. Computer and System Sciences.

[22] U.V.Vazirani and V.V.Vazirani, *Efficient and secure pseudo-random number generation*, Advances in Cryptology - Crypto'84, LNCS vol.196, Springer-Verlag (1985), 193-202.

[23] M.Wegman and J.Carter, *New hash functions and their use in authentication and set equality*, J. Computer and System Sciences vol.22 (1981), 265-279.

[24] A.C.Yao, *Theory and applications of trapdoor functions*, Proc. 23rd Annual IEEE Symposium on Foundations of Computer Science (FOCS), IEEE Computer Society Press (1982), 80-91.

[25] Y.Zheng and J.Seberry, *Practical approaches to attaining security against adaptively chosen ciphertext attacks*, Proc. Crypto'92.