On Concrete Security Treatment of Signatures Derived from Identification

Kazuo Ohta Tatsuaki Okamoto

NTT Laboratories
Nippon Telegraph and Telephone Corporation
1-1 Hikari-no-oka, Yokosuka, Kanagawa, 239-0847 Japan
E-mail: {ohta, okamoto}@isl.ntt.co.jp

Abstract. Signature schemes that are derived from three move identification schemes such as the Fiat-Shamir, Schnorr and modified ElGamal schemes are a typical class of the most practical signature schemes. The random oracle paradigm [1, 2, 12] is useful to prove the security of such a class of signature schemes [4, 12]. This paper presents a new key technique, "ID reduction", to show the concrete security result of this class of signature schemes under the random oracle paradigm. First, we apply this technique to the Schnorr and modified ElGamal schemes, and show the "concrete security analysis" of these schemes. We then apply it to the multi-signature schemes.

1 Introduction

1.1 Background

To realize a practical and provably secure cryptosystem is one of the most important research topics, and digital signatures are a very important ingredient in cryptography. This paper focuses on practical and provably secure signature schemes.

1.1.1 Standard Security Paradigm versus Random Oracle Paradigm

The first formal definition of the security for digital signatures ("existentially unforgeable against adaptively chosen-message attacks") was given by Goldwasser, Micali and Rivest [7], and a concrete signature scheme satisfying this security definition was shown by assuming the existence of a claw-free pair of functions [7]. Hereafter, this formal definition and model for signatures is called the "standard security paradigm", and a signature scheme with the standard security paradigm is just called a "provably secure" signature scheme.

An ultimate target in the standard security paradigm was to realize a provably secure signature scheme assuming the weakest computational assumption, the existence of a one-way function. This target was finally solved affirmatively by Naor, Yung and Rompel [9, 13]. Their solution, however, was geared towards feasibility result and thus very inefficient and far from practical. In addition, even the scheme by [7] is much less efficient than typical practical schemes such as the RSA[14] and Schnorr[15] schemes. Therefore, no provably secure scheme as efficient as typical practical schemes has been proposed.

To realize provable security and efficiency simultaneously, another paradigm to prove the security of cryptographic schemes has been proposed [1, 2, 12].

This is called the "random oracle paradigm", in which an ideally random and imaginary oracle, the "random oracle", is assumed when proving the security, and the random oracle is replaced by a practical random-like function such as a one-way hash function (e.g., SHA etc.) when realizing it in practice. Here random oracle F generates an answer randomly to a query posed to F at first. If the same query is asked later, F will answer the same value as was provided to the first query. Although the security under the *random oracle paradigm* cannot be guaranteed formally when using a practical random-like function in place of the random oracle, this paradigm yields much more efficient schemes than the *standard security paradigm*. The security with the random oracle gives an informal guarantee to the security of practical random-like functions.

In addition, the random oracle model not only provides a methodology for constructing an efficient and secure scheme, but also gives some security guarantee for schemes that practitioners intuitively constructed using a random-like functions in actual systems.

1.1.2 Asymptotic Security Analysis versus Concrete Security Analysis

The *random oracle paradigm* has another advantage over the *standard security paradigm*: it can much more easily provide "concrete security analysis", which avoids complexity theory and asymptotic property when proving the security (i.e., reducing the breaking of a primitive problem to breaking a signature scheme). Such *concrete security analysis* provides a much better guarantee than *asymptotic security analysis*, since the computational complexity currently required to break a signature scheme with a "fixed size" (e.g., 1024 bits) and "fixed key" can be estimated by the assumed lower bound of the complexity of breaking the underlying primitive with the "fixed size" and "fixed key." Note that *asymptotic security* gives no useful information on the security of a fixed size and fixed key system.

The *concrete security analysis* of the reduction from breaking a signature scheme to solving a primitive problem is usually trivial and optimal (i.e., optimally efficient). Hence, we have to obtain the *concrete security analysis* of the opposite direction of the reduction as much as optimal. If the opposite direction is as efficient as the trivial direction, then we can call such a reduction *exact*. That is, the *exact* reduction implies that the required time (and success probability) of breaking the signature scheme is *exactly* equivalent to that of breaking the primitive problem. (In other words, the signature scheme is *exactly* as secure as the primitive problem.)

The (almost) *exact security* of the RSA signature scheme along with random functions has been shown under the *random oracle paradigm* [2]. The *asymptotic security* of the Schnorr and modified ElGamal schemes has been proven under the same paradigm [12].

1.2 Main Result

This paper shows the *concrete security analysis* of the Schnorr, modified ElGamal (MEG) and multi-signature schemes under the *random oracle paradigm*. (The *concrete security analysis* of the other signature schemes based on the Fiat-Shamir conversion technique can be proven similarly.)

In order to show the *concrete security analysis* of the signature schemes, we have developed a new technique, "ID reduction", in which the identification scheme corresponding to the signature scheme is used when showing the reduction from breaking the underlying primitive to breaking the signature scheme. There are two stages of reduction. The first stage is from breaking the corresponding identification to breaking the signature scheme, and the second stage is from breaking the underlying primitive to breaking this identification.

In order to obtain a tighter (i.e., close to optimal) reduction and its tighter evaluation from breaking the underlying primitive to breaking the signature scheme, our "ID reduction" technique has an advantage over the previous technique, "forking lemma", by Pointcheval and Stern [12]. This is because the first stage of ID reduction (ID reduction lemma: Lemma 9) is optimal* in our signature scheme model and the second stage of this reduction (Lemma 13 and Lemma 15) may be more efficient than the reduction in the forking lemma of [12], since to analyze the corresponding identification scheme is easier than to analyze the signature scheme directly. Here, finding a forking pair of signatures in the forking lemma of [12] corresponds to finding two success entries in a heavy row in our approach. Therefore, the ID reduction technique seems to be more appropriate to obtain a tighter reduction than the previous technique.

In addition, the asymptotic result of the Fiat-Shamir signature scheme proven in [12] can be trivially obtained just by combining the ID reduction lemma as the first stage reduction and the well-known techniques given by [5] as the second stage reduction.

2 Framework

In this paper, we investigate a specific class of signature schemes that are derived from three move identification schemes, where the identification schemes are perfect zero-knowledge against an honest verifier [6]. This section shows the models and notations of such signature and identification schemes.

2.1 Signature Scheme

In the signature scheme, signer P publishes public key K_p while keeping secret key K_s. In this paper, we will adopt the following model as a signature scheme, which covers the class of the Fiat-Shamir scheme [4],Schnorr scheme [15] and the modified ElGamal scheme [12]:

Model 1. (Signature Model)
Key generation: Each signer P generates a pair, (K_p, K_s), of a secret key and a public key using a key generation algorithm $\mathcal{G}$ which, on input 1^k, where k is the security parameter, produces (K_p, K_s).
Signature generation: P generates the signature of his message m using a public random oracle function F as follows: P generates X from both K_s and random string R, accesses the random oracle function F to get $E = F(X, m) \in \mathcal{E}$, calculates Y using K_s, R and E, and sends (X, m, Y) to V.
Verification: a verifier V checks the validity of the signature of the message by the relations of (K_p, X, E, Y) and $E = F(X, m)$.

* We will show the meaning of "optimal" in the end of Section 3.

Remark. We assume that this signature scheme is derived from the following identification scheme.

2.2 Identification Scheme

Here we can define an identification scheme that produces the above-mentioned signature scheme.

In an identification scheme, prover P publishes a public key while keeping the corresponding secret key, and proves his identity to verifier V.

Model 2. (Identification Scheme)
Key generation: Prover P generates a pair, (K_p, K_s), of a secret key and a public key using a key generation algorithm $\mathcal{G}$ which, on input 1^k, where k is the security parameter, produces (K_p, K_s).
Identification Protocol: P proves his identity, and verifier V checks the validity of P's proof as follows:

Step 1 P generates X from both K_s and random string R and sends it to V.
Step 2 V generates random challenge $E \in \mathcal{E}$ and sends it to P.
Step 3 P generates an answer Y from (K_s, R, E) and sends it to V
Step 4 V checks the validity of the relations of (K_p, X, E, Y).

Remark. We assume that this three move protocol is *perfect zero-knowledge* against an honest verifier.

2.3 Security

We will adopt the quantifiable notion of *exact security* proposed in Reference [2].

2.3.1 Security of Key Searching Problem

Definition 3. A probabilistic Turing machine (adversary) A breaks a key search problem with (t, ϵ) if and only if A can find a secret key from a public key with success probability greater than ϵ within processing time t. The probability is taken over the coin flips of A.

Definition 4. A key searching problem is (t, ϵ)-secure if and only if there is no adversary that can break it with (t, ϵ).

2.3.2 Security of Identification Schemes

Definition 5. A probabilistic Turing machine (adversary) A breaks an identification scheme with (t, ϵ) if and only if A as a prover can cheat honest verifier V with a success probability greater than ϵ within processing time t. Here, A doesn't conduct any active attack**. Here, the probability is taken over the coin flips of A and V.

Definition 6. An identification scheme is (t, ϵ)-secure if and only if there is no adversary that can break it with (t, ϵ).

** As the result of Lemma 9 3), it is enough to cover this case only for discussion of the security of identification schemes, where the honest verifier is assumed.

2.3.3 Security of Signature Schemes

Next we will quantify the security of a signature scheme: Here we assume that the attacker can dynamically ask the legitimate user P to sign any message, m, using him as a kind of oracle. This model covers the very general attack of the signature situations, *adaptive chosen message attack.*

Definition 7. A probabilistic Turing machine (adversary) A breaks a signature scheme with $(t, q_{sig}, q_F, \epsilon)$ if and only if A can forge a signature of a message with success probability greater than ϵ . We allow chosen-message attacks in which A can see up to q_{sig} legitimate chosen message-signature pairs participating in the signature generating procedure, and allow q_F invocations of F, within processing time t. The probability is taken over the coin flips of A, F and signing oracle P.

Definition 8. A signature scheme is $(t, q_{sig}, q_F, \epsilon)$-secure if and only if there is no adversary that can break it with $(t, q_{sig}, q_F, \epsilon)$.

3 ID Reduction Lemma

The general techniques by which we can derive signature schemes from three move interactive protocols were proposed in [4] and hash functions are used in order to create a kind of virtual verifier, which gives the conversion from an identification scheme to a signature scheme.

To analyze the security of such a class of signature schemes, we will examine the opposite direction of conversion for adversaries in Lemma 9 in order to prove the security of signature schemes as the first stage of *ID Reduction Technique.*

Here note a signature scheme and an identification scheme in this section mean those defined in the previous section. We assume the uniform coin flips over $\mathcal{E}$ (i.e., Pr[E occurs] $= \frac{1}{\#\mathcal{E}}$) are provided.

Lemma 9. (ID Reduction Lemma)
Let $\epsilon \geq \frac{q_F(q_{sig}+4)+1}{\#\mathcal{E}}$ (i.e., $\epsilon' \geq \frac{4+q_{sig}}{\#\mathcal{E}}$, where $\epsilon' = \frac{\epsilon - \frac{1}{\#\mathcal{E}}}{q_F}$).
1) If A_1 breaks a signature with $(t, q_{sig}, q_F, \epsilon)$, there exists A_2 which breaks the signature with $(t, q_{sig}, 1, \epsilon')$, where $\epsilon' = \frac{\epsilon - \frac{1}{\#\mathcal{E}}}{q_F}$.
2) If A_2 breaks a signature with $(t, q_{sig}, 1, \epsilon')$, there exists A_3 which breaks the signature with $(t', 0, 1, \epsilon'')$, where $\epsilon'' = \epsilon' - \frac{q_{sig}}{\#\mathcal{E}}$ and $t' = t +$ (the simulation time of q_{sig} signatures).
*3) If A_3 breaks a signature with $(t', 0, 1, \epsilon'')$, there exists A_4 which breaks the corresponding identification scheme with (t', ϵ'')***.*

Here we assume that the values of q_F and q_{sig} can be employed by these reductions†. We neglect the time of reading/writing data on (random, communication, etc.) tapes, simple counting, and if-then-else controls. (Hereafter in this paper, we assume them.)

*** From the condition of ϵ, $\epsilon'' \geq \frac{4}{\#\mathcal{E}}$ holds. It makes the heavy row technique available in Lemma 13 and Lemma 15, since there are at least two '1' in a heavy row of a Boolean matrix H defined in Section 4.2.2.

† For simplicity, we also assume that these values don't depend on the adversary's coin flips but only on the length of its input.

Sketch of Proof:

1) Let Q_i be the i-th query from A_1 to the random oracle F and ρ_i be the i-th answer from F to A_1. Construct a machine B using A_1 as follows:

Step 1 Select an integer i satisfying $1 \leq i \leq q_F$ randomly.

Step 2 Run A_1 with a random oracle F and get (X, m, E, Y).

Step 3 If $(X, m) = Q_i$ and $E = \rho_i$, then output (X, m, E, Y). Otherwise output (Q_i, ρ_i, r_i) where r_i is a random element of the range of Y.

If A_1 succeeds in forging a signature (X, m, E, Y), there are two cases: 1) (X, m) was not asked to the random oracle F, and 2) (X, m) was asked as the i-th query to the random oracle F $(1 \leq i \leq q_F)$.

In the former case, the success probability of A_1 is at most $1/\#\mathcal{E}$, because of the randomness of the random oracle. Thus

$$\begin{aligned}
&\Pr[B \text{ succeeds}] \\
&\geq \sum_{i=1}^{q_F} \Pr[i \text{ is selected}] \Pr[A_1 \text{ succeeds} \wedge (X, m) = Q_i] \\
&= \sum_{i=1}^{q_F} \frac{1}{q_F} \Pr[A_1 \text{ succeeds} \wedge (X, m) = Q_i] \\
&= \frac{1}{q_F} \sum_{i=1}^{q_F} \Pr[A_1 \text{ succeeds} \wedge (X, m) = Q_i] \\
&= \frac{1}{q_F}(\Pr[A_1 \text{ succeeds}] - \Pr[A_1 \text{ succeeds} \wedge (X, m) \text{ is not a query to } F]) \\
&\geq \frac{1}{q_F}(\epsilon - \frac{1}{\#\mathcal{E}}),
\end{aligned}$$

because $\Pr[A_1 \text{ succeeds}] \geq \epsilon$.

Construct a machine $\widetilde{B}$ using A_1 as follows:

Step 1 Select an integer i satisfying $1 \leq i \leq q_F$ randomly.

Step 2 Run A_1 with a random oracle F and a random working tape Θ, and get (X, m, E, Y), where only the i-th query is asked to F and the remaining $(q_F - 1)$ queries are asked to Θ. Here Θ contains of $(q_F - 1)$ random blocks used as answers from Θ.

Step 3 If $(X, m) = Q_i$ and $E = \rho_i$, then output (X, m, E, Y). Otherwise output (Q_i, ρ_i, r_i) where r_i is a random element of the range of Y.

A_1 cannot distinguish $(q_F - 1)$ random blocks of Θ from $(q_F - 1)$ answers from F, because of the randomness of F. Thus $\Pr[B \text{ succeeds}] = \Pr[\widetilde{B} \text{ succeeds}]$ holds. Therefore,

$$\Pr[\widetilde{B} \text{ succeeds}] \geq \frac{\epsilon - \frac{1}{\#\mathcal{E}}}{q_F}.$$

Put $A_2 = \widetilde{B}$.

2) Construct a machine A_3 using A_2 as follows:

```
Step 1 For $j = 1$ to $q_{sig}$ do.
  Step 1-1 Run $A_2$ with simulated $(X_i, m_i, E_i, Y_i)$ $(1 \leq i \leq j - 1)$, and get a message $m_j$ chosen by $A_2$ whose signature is requested to the signer.
  Step 1-2 Simulate $(X_j, m_j, E_j, Y_j)$ by the standard perfect ZKIP simulation technique of the corresponding identification scheme with an honest verifier. If there exists an integer $i (< j)$ satisfying $X_j = X_i$, discard $X_j$ and repeat this step.
Step 2 Run $A_2$ with a random oracle $F$ and simulated $(X_i, m_i, E_i, Y_i)$ $(1 \leq i \leq q_{sig})$, and get $(X, m, E, Y)$.
Step 3 Output $(X, m, E, Y)$.
```

If A_2 does not ask (X_i, m_i) $(1 \leq i \leq q_{sig})$ to F, then A_2 cannot distinguish the simulated message-signature pairs from legitimate pairs because of the perfect indistinguishability described in Section 2 and the the randomness of F's output. The success probability of A_3 is given as follows:

$$\begin{aligned}
\epsilon'' &= \Pr[A_3 \text{ succeeds}] \\
&= \Pr[A_2 \text{ succeeds} \wedge (X_j, m_j) \neq (\text{the query from } A_2 \text{ to } F) \text{ for } 1 \leq \forall j \leq q_{sig}] \\
&= \Pr[A_2 \text{ succeeds}] \\
&\quad - \Pr[\exists i \text{ such that } 1 \leq i \leq q_{sig} \wedge (X_i, m_i) = (\text{the query from } A_2 \text{ to } F)] \\
&\geq \epsilon' - \frac{q_{sig}}{\#\mathcal{E}},
\end{aligned}$$

while $t' = t +$ (the simulation time of q_{sig} signatures in Step 1-2).

3) Let Q be a query from A_3 to the random oracle F and ρ be an answer from F to A_3. Construct a machine A_4 using A_3 interacting with an honest verifier V as follows:

```
Step 1 Run $A_3$ and get a query $Q = (X, m)$ which is sent to the random oracle $F$.
Step 2 Send $Q$ to $V$ and get a challenge $E$ from $V$.
Step 3 Run $A_3$ with an input $\rho = E$ and get $(X, m, E, Y)$.
Step 4 Output $Y$ to $V$.
```

Note that a valid signature (X, m, E, Y) satisfies a relation of (K_p, X, E, Y) and $E = F(X, m)$. When a verifier V checks the validity of this relation, V accepts A_4's proof with (t', ϵ''). □

Remark. When ignoring the minor terms (the simulation time and $\epsilon' - \epsilon''$), the first stage of ID reduction for the signature schemes in this paper is *optimal* in the following sense: For any strategy of A_1, $\epsilon' = \frac{\epsilon - \frac{1}{\#\mathcal{E}}}{q_F}$. On the other hand, let assume a specific $\widetilde{A_1}$, where $\Pr[\widetilde{A_1} \text{ succeeds} \wedge (X, m) = Q_i] = \frac{\epsilon - \frac{1}{\#\mathcal{E}}}{q_F}$. Then, for any strategy of the first stage reduction (signature to identification), $\epsilon' = \frac{\epsilon - \frac{1}{\#\mathcal{E}}}{q_F}$.

Since we cannot neglect the existence of such a specific $\widetilde{A_1}$, we cannot obtain the first stage reduction whose value of ϵ' is better than $\frac{\epsilon - \frac{1}{\#\mathcal{E}}}{q_F}$.

Note that this does not mean the "exact" security, since $\epsilon' \approx \epsilon$ in the "exact" security, while $\epsilon' \approx \frac{\epsilon}{q_F}$ in our "optimal" reduction.

In addition, note that this observation depends on the signature scheme model shown in Section 2.

4 Schnorr Signature Scheme

We discuss here the Schnorr scheme [15] as an example, though similar results can be obtained for the Fiat-Shamir scheme [4, 5] etc. The schemes can also be implemented using an elliptic curve [8].

4.1 Scheme

Key generation: A trusted center publishes two large primes, p and q, such that $q \mid (p-1)$, and element $g \in (\boldsymbol{Z}/p\boldsymbol{Z})^*$ of order q. A signer P chooses a secret key $s \in \boldsymbol{Z}/q\boldsymbol{Z}$ and publishes the public key I, where $I = g^s \bmod p$.

Signature generation: A signer P generates the signature of his message m using a public hash function h, and a verifier V checks the validity of signature of the message as follows: P generates a random integer $r \in \boldsymbol{Z}/q\boldsymbol{Z}$, calculates $X = g^r \bmod p$, $e = F(X, m) \in \boldsymbol{Z}/q\boldsymbol{Z}$ and $y = r + es \bmod q$, and sends (X, m, y) to V.

Verification: V checks the validity of a signature of the message by the following equations: $g^y \overset{?}{\equiv} XI^e \pmod{p}$ and $e \overset{?}{=} F(X, m)$.

4.2 Security

The following identification scheme is reduced to the Schnorr signature scheme in Section 4.1, and it will be analyzed adopting the scenario given in Section 3.

4.2.1 Identification Scheme

Key generation: A trusted center publishes two large primes p and q such that $q \mid (p-1)$, and element $g \in (\boldsymbol{Z}/p\boldsymbol{Z})^*$ of order q. A prover P chooses a secret key $s \in \boldsymbol{Z}/q\boldsymbol{Z}$ and publishes the public keys I, where $I = g^s \bmod p$.

Identification Protocol: P proves his identity and a verifier V checks the validity of P's proof as follows:

Step 1 P generates a random integer $r \in \boldsymbol{Z}/q\boldsymbol{Z}$, calculates $X = g^r \bmod p$, and sends X to verifier V.

Step 2 V generates a random integer $e \in \boldsymbol{Z}/q\boldsymbol{Z}$ and sends it to P.

Step 3 P calculates $y = r + es \bmod q$ and sends it to P.

Step 4 V checks the following equation: $g^y \overset{?}{\equiv} XI^e \pmod{p}$.

4.2.2 Heavy Row Lemma

A Boolean matrix and heavy row will be introduced in order to analyze the security of one-round identification schemes. Assume that there is a cheater A who can break a one-round identification scheme with (t, ϵ), where $\epsilon \geq \frac{4}{q}$.

Definition 10. (Boolean Matrix of (A, V))
Let's consider the possible outcomes of the execution of (A, V) as a Boolean matrix $H(RA, e)$ whose rows correspond to all possible choices of RA, where RA is a private random tape of A; its columns correspond to all possible choices of e, which means $e \in RV$. Its entries are 0 if V rejects A's proof, and 1 if V accepts A's proof.

Note that $RV = (Z/qZ)$ in Schnorr's case.

Definition 11. (Heavy Row)
A row of matrix of H is *heavy* if the fraction of 1's along the row is at least $\epsilon/2$, where ϵ is the success probability of A.

Lemma 12. (Heavy Row Lemma)
The 1's in H are located in heavy rows of H with a probability of at least $\frac{1}{2}$.

4.2.3 Security of Identification Scheme

Lemma 13. (Security of Schnorr Identification Scheme)
Let $\epsilon \geq \frac{4}{q}$. Suppose that the key searching problem of (p, g, I), that is, calculation of s from I satisfying $I = g^s \bmod p$, is (t^*, ϵ^*)-secure. Then the Schnorr identification scheme with parameter (p, g, I) is (t, ϵ)-secure, where

$$t^* = \frac{3(t + \Phi_1)}{\epsilon} + \Phi_3 \quad \textit{and} \quad \epsilon^* = \frac{1}{2}\left(1 - \frac{1}{e}\right)^2 > \frac{9}{50}.$$

Here Φ_1 is the verification time of the identification protocol, Φ_3 is the calculation time of s in the final stage of the reduction, and e is the base of the natural logarithm.

Sketch of Proof:

Assume that there is a cheater A who can break an identification with (t, ϵ). We will construct a machine A^* which breaks the key searching problem of (p, g, I) with (t^*, ϵ^*) using A.

We will discuss the following probing strategy of H to find two 1's along the same row in H [5]:

Step 1 Probe random entries in H to find an entry $a^{(0)}$ with 1. We denote the row where $a^{(0)}$ is located in H by $H^{(0)}$.

Step 2 After $a^{(0)}$ is found, probe random entries along $H^{(0)}$ to find another entry with 1. We denote it by $a^{(1)}$.

It is proven that this strategy succeeds with constant probability in just $O(1/\epsilon)$ probes, using Lemma 12 concerning a useful concept, *heavy* row, defined in Definition 11.

Let p_1 be the success probability of step 1 with $\frac{1}{\epsilon}$ repetition. $p_1 \geq 1 - (1 - \epsilon)^{1/\epsilon} = p'_1 > 1 - \frac{1}{e} > \frac{3}{5}$, because the fraction of 1's in H is ϵ. Let p_2 be the success probability of step 2 with $\frac{2}{\epsilon}$ repetition. $p_2 \geq \frac{1}{2} \times \left(1 - (1 - \frac{\epsilon}{2})^{2/\epsilon}\right) = p'_2 > \frac{1}{2}(1 - \frac{1}{e}) > \frac{3}{10}$, because the probability that $H^{(0)}$ is heavy is at least $\frac{1}{2}$

by Lemma 12 and the fraction of 1's along a heavy row is at least $\frac{\epsilon}{2}$. Therefore $\epsilon^* = p_1 \times p_2 \geq p'_1 \times p'_2 > \frac{1}{2}(1-\frac{1}{e})^2 > \frac{9}{50}$ and $t^* = t \times (\frac{1}{\epsilon} + \frac{2}{\epsilon}) = \frac{3t}{\epsilon}$.

$a^{(i)}$ represents $(X^{(i)}, e^{(i)}, y^{(i)})$. $g^{y^{(i)}} \equiv X^{(i)^{d^{(i)}}} I \pmod{p}$ $(i = 0, 1)$ holds, since $a^{(i)}$ is an entry with 1. Two 1's, $a^{(0)}$ and $a^{(1)}$, in the same row $H^{(0)}$ means $X^{(1)} = X^{(0)}$. Since there are two unknown variables, $r^{(0)}$ and s, and two equations are obtained, a secret key s can be calculated by $s = \frac{y^{(0)} - y^{(1)}}{e^{(0)} - e^{(1)}} \bmod q$ in Schnorr's scheme, since q is prime and $0 < e^{(0)} - e^{(1)} < q$. □

4.2.4 Security of Signature Scheme

The following theorem is proven by combining Lemma 9 and Lemma 13.

Theorem 14. (Security of Schnorr Signature Scheme)
Let $\epsilon' \geq \frac{4+q_{sig}}{q}$, where $\epsilon' = \frac{\epsilon - \frac{1}{q}}{q_F}$. Suppose that key searching problem of (p, g, I) is (t^*, ϵ^*)-secure. Then the Schnorr signature scheme with parameter (p, g, I) is $(t, q_{sig}, q_F, \epsilon)$-secure, where

$$t^* = \frac{3t'}{\epsilon''} + \Phi_3 \quad \textit{and} \quad \epsilon^* = \frac{1}{2}\left(1 - \frac{1}{e}\right)^2 > \frac{9}{50}.$$

Here

$$t' = t + \Phi_1 + \Phi_2 \quad \textit{and} \quad \epsilon'' = \frac{\epsilon - \frac{1}{q}}{q_F} - \frac{q_{sig}}{q},$$

where Φ_1 is the verification time of the identification protocol, Φ_2 is the simulation time of q_{sig} signatures, Φ_3 is the calculation time of s in the final stage of the reduction, and q is the order of $g \in (Z/pZ)^*$.

4.3 Discussion on the Efficiency of Our Reduction

We have proven that if the key searching problem is (t^*, ϵ^*)-secure, then the Schnorr signature scheme is $(t, q_{sig}, q_F, \epsilon)$-secure. On the other hand, if the key searching problem is breakable with (t, ϵ), then the signature scheme is breakable with $(t, 0, 1, \epsilon)$ by the trivial reduction. If our reduction is "exact (optimally efficient)," (t^*, ϵ^*) should be the same quantity as (t, ϵ) for any values of q_{sig} and q_F. Here note that is does not always imply $t = t^*$ and $\epsilon = \epsilon^*$, since (t, ϵ) and (t^*, ϵ^*) are considered to have the same quantity when $t^* = \beta t$ and $\epsilon^* = 1 - (1 - \epsilon)^{\beta}$.

Here we will estimate the degree of "exactness" of our reduction (i.e., how much close is the above mentioned reduction to the exact case) by comparing the quantities of (t^*, ϵ^*) and (t, ϵ). For the purpose, we normalize (t, ϵ) into (t^+, ϵ^+) with $\epsilon^+ = \epsilon^*$.

Let $\beta = \frac{\alpha}{\epsilon}$ be the number of repetition of (t, ϵ)-breakable algorithm, in order to attain the same success probability as ϵ^*. Since $\epsilon^* = \frac{1}{2}(1 - \frac{1}{e})^2 > 9/50$, $\alpha \approx 0.223$ holds because of the requirement of $1 - (1 - \epsilon)^{\frac{\alpha}{\epsilon}} = \epsilon^* > 9/50$. Therefore, $t^+ = \frac{\alpha t}{\epsilon} (\alpha \approx 0.223)$ and the ratio of t^* and t^+ gives the degree of exactness of our reduction.

If we assume that $t \approx t'$ and $\epsilon' \approx \frac{\epsilon}{q_F}$, since q_{sig} is small and q is large, then its ratio is $\frac{3q_F}{\alpha} \approx 13.5q_F$. Thus, our reduction is still efficient, though it is not exact. Here note that q_F can not be eliminated from this ratio because of the optimality of the ID reduction lemma.

5 Modified ElGamal Signature Scheme

We will discuss the modified ElGamal (MEG) signature scheme [12] in this section.

5.1 Scheme

Key generation: the same as the Schnorr scheme.
Signature generation: A signer P generates the signature of his message m using a public hash function h as follows: P generates a random integer $r \in (Z/qZ)^*$, calculates $X = g^r \bmod p$, $e = F(X, m) \in Z/qZ$ and $y = \frac{e-sX}{r} \bmod q$, and sends (X, m, y) to V.
Verification: a verifier V checks the validity of the signature of the message by the following equations: $g^e \overset{?}{\equiv} X^y I^X \pmod p$ and $e \overset{?}{=} F(X, m)$.
Note: In the original ElGamal scheme, the order of $g \in (Z/pZ)^*$ is $p-1$. Although we can prove the security of the MEG with $\mathrm{ord}(g) = p-1$ in a manner similar to that with $\mathrm{ord}(g) = q$, here for simplicity of description we assume $\mathrm{ord}(g) = q$.

5.2 Security

5.2.1 Identification Scheme

The following identification scheme is reduced to the MEG signature scheme in Section 5.1, and it will be analyzed adopting the scenario given in Section 3.
Key generation: the same as the Schnorr scheme.
Identification Protocol: P proves his identity and verifier V checks the validity of P's proof as follows:

Step 1 P generates a random integer $r \in (Z/qZ)^*$, calculates $X = g^r \bmod p$, and sends X to verifier V.

Step 2 V generates a random integer $e \in Z/qZ$ and sends it to P.

Step 3 P calculates $y = \frac{e-sX}{r} \bmod q$, and sends it to P.

Step 4 V checks the following equation: $g^e \overset{?}{\equiv} X^y I^X \pmod p$.

5.2.2 Security of Identification Scheme

Lemma 15. (Security of ElGamal Identification Scheme)
Let $\epsilon \geq \frac{4}{q}$. Suppose that the key searching problem of (p, g, I) is (t^*, ϵ^*)-secure. Then the ElGamal identification scheme with parameter $(p, \tilde{g}, I)$[‡] *is (t, ϵ)-secure, where*

$$t^* = \left(\frac{3(t + \Phi_1)}{\epsilon} + \Phi_3\right)\sqrt{R} \quad \text{and} \quad \epsilon^* = \left(\frac{1}{2}(1 - \frac{1}{e})^2\right)^{\sqrt{R}} > \left(\frac{9}{50}\right)^{\sqrt{R}}.$$

[‡] $\tilde{g}$ is an appropriate element in the subgroup, $< g >$, generated by g.

Here Φ_1 is the verification time of the identification protocol, Φ_3 is the calculation time of r and s (or $\tilde{g}$) at Step 3 and Step 4, $R = \frac{p-1}{q}$ and q is the order of $g \in (Z/pZ)^*$.

Sketch of Proof:

Assume that cheater A breaks the ElGamal identification with (t, ϵ) for (p, I) and all $\tilde{g} \in < g >$. We will construct a machine A^* that breaks the key searching problem of (p, g, I) with (t^*, ϵ^*) using A.

We will discuss the following probing strategy of H to find two 1's along the same row in H [5] for the identification scheme with parameter (p, g, I):

Step 1 Probe random entries in H to find an entry $a^{(0)}$ with 1. We denote the row where $a^{(0)}$ is located in H by $H^{(0)}$.

Step 2 After $a^{(0)}$ is found, probe random entries along $H^{(0)}$ to find another entry $a^{(1)}$ with 1.

Step 3 Calculate the value of r as follows,

$$r = \frac{e^{(0)} - e^{(1)}}{y^{(0)} - y^{(1)}} \bmod q$$

where $a^{(i)}$ represents $(X^{(i)}, e^{(i)}, y^{(i)})$ and $X = X^{(0)} = X^{(1)}$ $(i = 0, 1)$ holds. Note that r is coprime to q.

In Case 1 with $\gcd(X, q) = 1$, calculate a secret value of s as follows, output it and halt:

$$s = \frac{e^{(0)} - ry^{(0)}}{X} \bmod q.$$

In Case 2 with $\gcd(X, q) \neq 1$, obtain b satisfying $X = bq (= g^r \bmod p)$, where $0 < b < \frac{p-1}{q} = R$, and go to Step 4.

Step 4 (For Case 2 only) Run A with input $\tilde{g} = Ig^l \bmod p$ applying Step 1 to Step 3, where $l \in Z/qZ$ is randomly selected. There are two cases, Case 1 and Case 2.

In Case 1 with $\gcd(\tilde{X}, q) = 1$, calculate s by the same way as Step3.

In Case 2 with $\gcd(\tilde{X}, q) \neq 1$, obtain $\tilde{b}$ as well as $\tilde{r}$ satisfying $\tilde{b}q = \tilde{g}^{\tilde{r}} \bmod p$ by the same way in Step 3.

If $\tilde{b} = b$ holds, calculate a secret value of s as follows, output it and halt:

$$s = \frac{r - l\tilde{r}}{\tilde{r}} \bmod q.$$

Otherwise, repeat Step 4 with another input.

The worst case for finding two values of b that collide is that these $R - 1$ events occur with equal probability $\frac{1}{R-1}$ within Case 2.

Let p_1 be the success probability of step 1 with $\frac{1}{\epsilon}$ repetition, and p_2 be the success probability of step 2 with $\frac{2}{\epsilon}$ repetition. Let p_{3-1} be the success probability of Case 1 in step 3, and $p_{3-1} = 0$ in the worst case. Let p_4 be the success probability of step 4 with $\sqrt{R}$ repetition. Then $p_4 \approx 1$ because of the birthday paradox of finding $b = b'$ satisfying $0 < b, b' < R$.

Therefore

$$\epsilon^* = (p_1 \times p_2)^{\sqrt{R}} p_4 \geq \left(\frac{1}{2}(1 - \frac{1}{e})^2\right)^{\sqrt{R}}$$

and

$$t^* = \left((t + \Phi_1) \times (\frac{1}{\epsilon} + \frac{2}{\epsilon}) + \Phi_3\right) \sqrt{R} = \left(\frac{3(t + \Phi_1)}{\epsilon} + \Phi_3\right) \sqrt{R}.$$

□

5.2.3 Security of Signature Scheme

The following theorem is proven by combining Lemma 9 and Lemma 15.

Theorem 16. (Security of ElGamal Signature Scheme)
Let $\epsilon' \geq \frac{4+q_{sig}}{q}$, where $\epsilon' = \frac{\epsilon - \frac{1}{q}}{q_F}$. Suppose that the key searching problem of (p, g, I) is (t^*, ϵ^*)-secure. Then the ElGamal signature scheme with parameter $(p, \tilde{g}, I)$ is $(t, q_{sig}, q_F, \epsilon)$-secure, where

$$t^* = \left(\frac{3(t + \Phi_1 + \Phi_2)}{\epsilon''} + \Phi_3\right) \sqrt{R} \quad and \quad \epsilon^* = \left(\frac{1}{2}(1 - \frac{1}{e})^2\right)^{\sqrt{R}} > \left(\frac{9}{50}\right)^{\sqrt{R}}.$$

Here Φ_1 is the verification time of the identification protocol, Φ_2 is the simulation time of q_{sig} signatures, and Φ_3 is the calculation time of r and s (or $\tilde{g}$) at Step 3 and Step 4. $\epsilon'' = \frac{\epsilon - \frac{1}{q}}{q_F} - \frac{q_{sig}}{q}$, where q is the order of $g \in (Z/pZ)^*$.

Remark. The simulation time of q_{sig} signatures can be obtained in a manner similar to that in Lemma 8 in Reference [12].

5.3 More Efficient Reduction of MEG

Clearly the reduction for the MEG signature scheme is much less efficient than that of the Schnorr scheme, and the reduction does not preserve the parameter, (p, g, I). If we modify the MEG scheme as follows, the reduction can be almost as efficient as that of the Schnorr scheme and can preserve the parameter.

The modified version of the MEG scheme is the same as the MEG scheme except: Verifier V checks whether $\gcd(X, q) = 1$, and if it does not hold, V rejects the signature, (m, X, y). Note that when a valid signer generates (m, X, y), the probability that $\gcd(X, q) \neq 1$ is $1/q$ (negligible probability).

6 Multi-Signature Schemes

Multi-signature schemes are signature schemes in which plural signers (e.g., L signers) jointly generate a signature (multi-signature) of a message under the condition that the length of the multi-signature is less than the total length of ordinary (single) signatures by plural signers (e.g., $L \times |s|$, where $|s|$ is the ordinary signature length).

We can apply our ID reduction technique to the "one-round type" of multi-signature schemes[§]. This section briefly introduces our results regarding multi-signature schemes. Due to the space limitation, we omit a detailed description of the results [11].

6.1 The Proposed Multi-Signature Schemes

We propose provably secure multi-signature schemes against the most general attack, adaptively chosen message insider attacks [7] with the random oracle model. The proposed schemes are as follows[¶]:
Key generation: A trusted center publishes two large primes p and q such that $q \mid (p-1)$, and element $g \in (\boldsymbol{Z}/p\boldsymbol{Z})^*$ of order q. Each signer P_i chooses a secret key $s_i \in \boldsymbol{Z}/q\boldsymbol{Z}$ and publishes the public key I_i, where $I_i = g^{s_i} \bmod p$ $(1 \leq i \leq L)$ and L is the number of signers.
Multi-Signature: Each signer P_i generates the signature of his message m using two public hash functions F_i and H_i as follows $(1 \leq i \leq L)$:

Step 1 For $i = 1$ to L do, where $y_0 = 0$ and $V = P_{L+1}$:
P_i generates a random integer $r_i \in \boldsymbol{Z}/q\boldsymbol{Z}$, calculates $X_i = g^{r_i} \bmod p$, $e_i = F_i(X_1, \ldots, X_i, m) \in \boldsymbol{Z}/q\boldsymbol{Z}$, $d_i = H_i(X_1, \ldots, X_i, m) \in \boldsymbol{Z}/q\boldsymbol{Z}$ and $y_i = y_{i-1} + d_i r_i + e_i s_i \bmod q$, and sends $(X_1, \ldots, X_i, m, y_i)$ to P_{i+1}.

Step 2 V checks the following equations: $g^{y_L} \overset{?}{\equiv} X_1^{d_1} \cdots X_L^{d_L} I_1^{e_1} \cdots I_L^{e_L} \pmod p$, and $e_i \overset{?}{=} F_i(X_1, \ldots, X_i, m)$, $d_i \overset{?}{=} H_i(X_1, \ldots, X_i, m)$ $(1 \leq i \leq L)$.

Remark. 1) We call the scheme where $d_i = 1$ Type I, the scheme where $e_i = 1$ Type II, and the scheme where there is no restriction on d_i, e_i Type III.
2) The schemes can also be implemented using an elliptic curve [8].
3) It is possible for each P_i to check the validity of $(I_1, \ldots, I_{i-1}, X_1, \ldots, X_{i-1}, m, E_1, \ldots, E_{i-1}, Y_{i-1})$ before generating his signature.

6.2 Security of the schemes

The main results are as follows:

[§] The "two-round type" of multi-signature schemes have been proposed [10]. Our technique can also be applied to these schemes easily.

[¶] For simplicity of explanation, in this paper we use the multiplicative group $(Z/pZ)^*$ to present our schemes and the security proofs. Only the implementations over elliptic curves [8], however, are feasible in light of the multi-signature size. Note that the security of the elliptic curve versions can be proven in the same manner as those of the multiplicative group versions.

Theorem 17. (Security of the Proposed Multi-Signature Scheme (Type II)) *Let $\epsilon' \geq \frac{2^{(L+1)}+q_{sig}}{q}$. Here $\epsilon' = \epsilon_{H_L}$, where $\epsilon_{H_0} = \epsilon$, and $\epsilon_{H_i} = \frac{\epsilon_{H_{i-1}} - \frac{1}{q}}{q_{H_i}}$ $(1 \leq i \leq L)$. Suppose that the calculation of s from $I_1, \ldots, I_L$ satisfying $I_1 \times \cdots \times I_L = g^s \bmod p$ is $(t^{II}(L), \epsilon^{II}(L))$-secure. Then the proposed multi-signature scheme with the same parameter is $(t, q_{sig}, q_{H_1}, q_{H_2}, \epsilon)$-secure, where*

$$t^{II}(L) = \frac{t'}{3\epsilon''}\left(2^{(2L+1)} + 1\right) + \Phi_3,$$

$$\epsilon^{II}(L) = \left(\frac{1}{2}\right)^{(2^L-1)}\left(1 - \frac{1}{e}\right)^{2^L} > \left(\frac{1}{2}\right)^{(2^L-1)}\left(\frac{3}{5}\right)^{2^L}.$$

Here $t' = t + \Phi_1 + \Phi_2$ and $\epsilon'' = \epsilon_{H_L} - \frac{q_{sig}}{q}$. Φ_1 is the verification time of the identification protocol, Φ_2 is the simulation time of q_{sig} signatures, Φ_3 is the calculation time of s in the final stage of the reduction, and q is the order of $g \in (Z/pZ)^*$.

Theorem 18. (Security of the Proposed Multi-Signature (Type III)) *Let $\epsilon' \geq \frac{2^{(L+2)}+q_{sig}}{q}$. Here $\epsilon' = \epsilon_{H_L}$, where $\epsilon_{H_0} = \epsilon$, $\epsilon_{F_i} = \frac{\epsilon_{H_{i-1}} - \frac{1}{q}}{q_{F_i}}$, and $\epsilon_{H_i} = \frac{\epsilon_{F_i} - \frac{1}{q}}{q_{H_i}}$ $(1 \leq i \leq L)$. Suppose that the calculation of s_i from $I_1, \ldots, I_L$ satisfying $I_i = g^{s_i} \bmod p$ is $(t^{III}(L), \epsilon^{III}(L))$-secure. Then the proposed multi-signature scheme with the same parameter is $(t, q_{sig}, q_{F_1}, q_{H_1}, \ldots, q_{F_L}, q_{H_L}, \epsilon)$-secure, where*

$$t^{III}(L) = \frac{t'}{3\epsilon''}\left(2^{(2L+1)} + 3L \times 2^{(L+1)} - 3 \times 2^{(L+1)} + 1\right) + \Phi_3,$$

$$\epsilon^{III}(L) = \epsilon^{II}(L)\left(\frac{1}{2}(1 - \frac{1}{e})\right)^{(L-1)} > \left(\frac{1}{2}\right)^{(2^L+L-2)}\left(\frac{3}{5}\right)^{(2^L+L-1)}.$$

Here $t' = t + \Phi_1 + \Phi_2$ and $\epsilon'' = \epsilon_{H_L} - \frac{q_{sig}}{q}$. Φ_1 is the verification time of the identification protocol, Φ_2 is the simulation time of q_{sig} signatures, Φ_3 is the calculation time of s in the final stage of the reduction, and q is the order of $g \in (Z/pZ)^*$.

Remark. The multi-signature scheme of Type I is forgeable by a true signer, for example, signer L can make a multi-signature of arbitrary message m without coalition of other $(L-1)$ signers.

7 Conclusion

This paper presented a new key technique, "ID reduction", to show the concrete security result of a class of practical signature schemes under the random oracle paradigm. We applied this technique to the Schnorr and modified ElGamal schemes, and showed the "concrete security" of these schemes. We also applied it to the multi-signature schemes. This technique should be useful in proving the concrete security of various types of signatures such as blind signatures, group signatures and undeniable signatures.

Acknowledgments

We would like to thank Adi Shamir for his talks at NTT in1988, which inspired us to create the reduction technique introduced in this paper. We would also like to greatly thank Moti Yung for his invaluable support in revising our manuscript. We wish to thank anonymous reviewers for useful comments.

References

1. M. Bellare and P. Rogaway, "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols," Proc. of the First ACM Conference on Computer and Communications Security, pp.62–73.
2. M. Bellare and P. Rogaway, "The Exact Security of Digital Signatures –How to Sign with RSA and Rabin," Advances in Cryptology –EUROCRYPT'96, Springer-Verlag, pp.399–416.
3. T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," IEEE Transactions on Information Theory, IT-31, 4, pp.469–472, 1985.
4. A. Fiat and A. Shamir, "How to Prove Yourself," Advances in Cryptology – CRYPTO'86, Springer-Verlag, pp.186–194.
5. U. Feige, A. Fiat and A. Shamir, "Zero-Knowledge Proofs of Identity," J. of Cryptology, 1, p.77–94.
6. S. Goldwasser, S. Micali and C. Rackoff, "The Knowledge Complexity of Interactive Proof Systems," SIAM J. on Computing, 18, pp.186-208, 1989.
7. S. Goldwasser, S. Micali and R. Rivest, "A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks," SIAM J. on Computing, 17, pp.281–308, 1988.
8. N. Koblitz, "Elliptic Curve Cryptosystems," Mathematics of Computation, 48, pp.203–209, 1987.
9. M. Naor and M. Yung, "Universal One-Way Hash Functions and Their Cryptographic Applications," Proc. of STOC, pp.33–43, 1989.
10. K. Ohta and T. Okamoto, "A Digital Multisignature Scheme Based on the Fiat-Shamir Scheme," Advances in Cryptology –ASIACRYPT'91, Springer-Verlag, pp. 139–148.
11. K. Ohta and T. Okamoto, "The Exact Security of Multi-Signature Schemes," Technical Report of IEICE, ISEC97-27 (July, 1997), pp.41-52.
12. D. Pointcheval and J. Stern, "Security Proofs for Signature Schemes," Advances in Cryptology –EUROCRYPT'96, Springer-Verlag, pp.387–398.
13. J. Rompel, "One-Way Functions are Necessary and Sufficient for Secure Signature," Proc. of STOC, pp.387–394, 1990.
14. R. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public Key Cryptosystems," Communications of ACM, 21, 2, pp.120-126, 1978.
15. C.P. Schnorr, "Efficient Identification and Signatures for Smart Card," Advances in Cryptology –EUROCRYPT'89, Springer-Verlag, pp.235–251.