On the Discrepancy between Serial and Parallel of Zero-Knowledge Protocols

(Extended Abstract)

Kouichi Sakurai

Computer &
Information Systems Laboratory,
Mitsubishi Electric Corporation,
5-1-1 Ofuna, Kamakura 247, Japan.
sakurai@isl.melco.co.jp

Toshiya Itoh

Department of Information Processing,
Interdisciplinary Graduate School
of Science and Engineering,
Tokyo Institute of Technology,
4259 Nagatsuta, Midori-ku,
Yokohama 227, Japan.
titoh@ip.titech.ac.jp

Abstract

In this paper, we investigate the discrepancy between a serial version and a parallel version of zero-knowledge protocols, and clarify the information "leaked" in the parallel version, which is not zero-knowledge unlike the case of the serial version. We consider two sides: one negative and the other positive in the parallel version of zero-knowledge protocols, especially of the Fiat-Shamir scheme.

1 Introduction and motivation

The notions of interactive proofs and zero knowledge were introduced by Goldwasser, Micali and Rackoff [GMR]. Fiat and Shamir [FiS] exhibited a practical identification scheme, which is zero-knowledge, based on the intractability of the factorization.

A common weakness in such zero-knowledge protocols is that the protocols require many iterations of a basic (three move) protocol, then such zero-knowledge protocols are not efficient.

The straightforward parallelization of the basic protocol decreases the round complexity of the protocols. However, a problem on the straightforward parallelization of zero-knowledge protocols is that a technique of the proof of zero-knowledge in the serial version, so called *resettable simulation*, fails in the parallel version.

Feige, Fiat and Shamir [FSS] showed that the parallel version of the Fiat-Shamir identification scheme releases no "useful" knowledge that could help the verifier to impersonate the prover within the identification system.

On the other hand, Goldreich and Krawczyk [GKr] observed that non zero-knowledgeness is an intrinsic property of the three move protocols, and showed that the parallel version of the Fiat-Shamir scheme is not zero-knowledge unless the factorization is tractable.

Our motivation of this study is derived from these contradictive results on the security of the parallel version of the Fiat-Shamir scheme (generally, the three move protocols).

Some researchers characterize the security of the parallel execution of the Fiat-Shamir type identification scheme [FSS, FeS, OhOk'88, BM]. However, none has investigated what kind of information is leaked by the parallel version or how useful these knowledge is for the verifier.

In this paper, we investigate the essential discrepancy between the serial version and the parallel version of the Fiat-Shamir scheme (more generally, zero-knowledge protocols), and clarify properties which the parallel version has but the serial version does not have.

E.F. Brickell (Ed.): Advances in Cryptology - CRYPTO '92, LNCS 740, pp. 246-259, 1993.

Our main observation is that the information "leaked" in the parallel version of the Fiat-Shamir identification scheme is closely related to a digital signature which is a modification of the Fiat-Shamir identification scheme, and the parallel version of zero-knowledge protocols leave a trace.

Furthermore, we consider two sides of the discrepancy, one negative and the other positive.

Organization of this paper

In section 2, we give the definitions and overview the Fiat-Shamir scheme. In section 3, we consider the reason why straightforward parallelization fail to be zero-knowledge. In section 4, we point out abuses of the parallel version. In section 5, we positively apply the parallel version. Finally, we conclude with future topics.

2 Preliminaries

In this section, we give some definitions on zero-knowledge [GMR] and overview of the Fiat-Shamir scheme [FiS, FSS]. The reader who is familiar with these topics may skip this section.

2.1 Notation and Definitions

Our model of computation is the interactive probabilistic Turing machines (both for the prover P and for the verifier V) with an auxiliary input. The common input is denoted by x and, and its length is denoted by $|x| = n$. We use $\nu(n)$ to denote any function vanishing faster than the inverse of any polynomial in n. More formally,

$$\forall k \in \mathrm{N}\ \exists n_0\ \ s.t.\ \forall n > n_0\ \ 0 \leq \nu(n) < \frac{1}{n^k}.$$

We define *negligible* probability to be the probability behaving as $\nu(n)$, and *overwhelming* probability to be the probability behaving as $1 - \nu(n)$.

Let $A(x)$ denote the output of a probabilistic algorithm A on input x. This is a random variable. When we want to make the coin tosses of A explicit, for any $\rho \in \{0,1\}^*$ we write $A[\rho]$ for the algorithm A with ρ as its random tape. Let $V_P(x)$ denote V's output after interaction with P on common input x, and let $M(x; A)$ (where A may be either P or V) denote the output of the algorithm M on input x, where M may use the algorithm A as a (blackbox) subroutine. Each call M makes to A is counted as a single computation step for M.

Definition 2.1 [GMR]: *An interactive proof for membership of the language L is a pair of interactive probabilistic Turing machines (P, V) satisfying:*

> **Membership Completeness:** *If x belongs to L, V accepts P's proof with overwhelming probability. Formally:*
>
> $$\forall x \in L \ \ Prob(V_{P(x)}(x)\, accepts) > 1 - \nu(|x|),$$
>
> *where the probability is taken over all of the possible coin tosses of P and V.*
>
> **Membership Soundness:** *If x does not belong to L and P^* may act in any way, V accepts P^*'s proof with negligible probability. Formally:*
>
> $$\forall x \notin L \forall P^* \ \ Prob(V_{P^*(x)}(x)\, accepts) < \nu(|x|),$$
>
> *where the probability is taken over all of the possible coin tosses of P^* and V.*

It should be noted that P's resource is computationally unbounded, while V's resource is bounded by probabilistic polynomial time in $|x|$.

Definition 2.2: *Let R be a relation $\{(x, w)\}$ testable in $\mathcal{BPP}$. Namely, given x and w, checking whether $(x, w) \in R$ is computed in probabilistic polynomial time. For any x, its witness set $w(x)$ is the set of w such that $(x, w) \in R$.*

Definition 2.3 [FSS]: *An interactive proof of knowledge for the relation R is a pair of interactive probabilistic Turing machines (P, V) satisfying:*

> **Knowledge Completeness:** *For any $(x, w) \in R$, V accepts P's proof with overwhelming probability. Formally:*
>
> $$\forall (x, w) \in R \;\; Prob(V_{P(x,w)}(x)\, accepts) > 1 - \nu(|x|),$$
>
> *where the probability is taken over all of the possible coin tosses of P and V.*
>
> **Knowledge Soundness:** *For any x, for any P^*, P^* can convince V to accept only if he actually "knows" a witness for $x \in dom\ R$. An expected polynomial time knowledge extractor M is used in order to demonstrate P^*'s ability to compute a witness. Formally:*
>
> $$\begin{aligned} &\forall a\, \exists M\, \forall P^*\, \forall x\, \forall w'\, \forall \rho \\ &Prob\Big(V_{P^*[\rho](x,w')}(x)\, accepts\Big) > 1/|x|^a \Rightarrow \\ &Prob\Big(M(x; P^*[\rho](x, w')) \in w(x)\Big) > 1 - \nu(|x|), \end{aligned}$$
>
> *where the probability is taken over all of the possible coin tosses of M and V. P^* is assumed not to toss coins, since his favorable coin tosses can be incorporated into the auxiliary input w'. The knowledge extractor M is allowed to use P^* as a blackbox subroutine and runs in expected polynomial time. Each message that P^* sends M costs a single computation step for M.*

Note that both P's and V's resource are bounded by probabilistic polynomial time in $|x|$.

We recall that the *view* of the verifier is everything he sees during an interaction with the prover, that is, his own coin tosses and the conversation between himself and the prover.

Definition 2.4 [GMR]: *Let (P, V) be an interactive protocol and let $x \in \{0, 1\}^*$. The view of V' on input x is the probability space*

$$VIEW_{(P,V')}(x) = \{(R, C) : R \leftarrow \{0, 1\}^{p(|x|)};\ C \leftarrow (P \leftrightarrow V'[R])(x)\},$$

where p is a polynomial bounding the running time of V', and $(P \leftrightarrow V'[R])(x)$ denotes the probability space of conversations between P and $V'[R]$ on input x (the probability is taken over all of the possible coin tosses of P).

Denote by $Time_P^{V'}(x)$ the running time of machine V' when interacting with P on input x.

Definition 2.5 [GO]: *An interactive proof system (P, V) of knowledge for the relation R is blackbox simulation perfect zero knowledge if there exists a universal simulator S_u which runs in expected polynomial time, such that for every polynomial Q and any pair (x, y, V') such that $(x, y) \in R$ and $Time_{P(y)}^{V'}(x) \leq Q(|x|)$, $S_u(x; V'(x))$ is exactly identical to $VIEW_{(P(y),V')}(x)$.*

Formally:

$$\exists S_u \forall Q \forall x \forall y \forall V' \ \ s.t. \ (x,y) \in R \ \& \ Time^{V'}_{P(y)}(x) \leq Q(|x|),$$
$$VIEW_{(P(y),V')}(x) = S_u(x; V'(x)).$$

Blackbox simulation zero knowledge represents the strongest notion of zero knowledge among the types of the simulation (e.g. auxiliary input zero-knowledge [GO]) although all known concrete zero knowledge protocols are in fact blackbox simulation zero knowledge. Thus these definitions above are reasonable and never too restrictive.

Throughout this paper, we use a term "zero knowledge" in the sense of *blackbox simulation* zero knowledge.

$\bar{A}$ (resp. $\bar{B}$) represents the real prover (resp. verifier) who follows its designated protocol. $\tilde{A}$ represents a polynomial time cheater who does not possess the witness (or secret) but can derive from the protocol in an arbitrary way. $\tilde{\bar{B}}$ represents an arbitrary polynomial time verifier who tries to extract additional information from $\bar{A}$.

Definition 2.6 [FSS]: *The protocol (A,B) releases no transferable information if:*

1. *It succeeds with overwhelming probability.*
2. *There is no coalition of $\tilde{A}, \tilde{\bar{B}}$ with the property that, after a polynomially many number of executions of $(\bar{A}, \tilde{\bar{B}})$ it is possible to execute $(\tilde{A}, \bar{B})$ with a non negligible probability of success.*

Ohta and Okamoto [OhOk'88] defined rigorous notions on "revealing no transferable information".

For more precise definition of *no transferable* that is suitable for the identification system, see the journal version of the reference [FSS].

2.2 The Fiat-Shamir scheme

Fiat and Shamir [FiS] exhibited a practical identification scheme and a signature scheme that are provably secure if factoring is difficult. We overview their scheme.

Fiat-Shamir identification scheme (FSIS)

1. PREPROCESSING STAGE BETWEEN THE TRUSTED CENTER AND EACH USER
   The unique trusted center's secret key in the system is (p,q), and the public key is N, where p,q are distinct large primes, $N = p \times q$. The center generates user A's secret key s_A, where $1/s_A = \sqrt{I_A}$ (mod N). I_A is the identity of user A and is published to other users.

2. IDENTIFICATION STAGE BETWEEN USER A AND USER B

   Repeat step (a) to (d) t times.

   (a) The user A picks $r \in_{\mathrm{R}} Z^*_N$, and sends $x \equiv r^2$ (mod N) to a user B.

   (b) The user B generates $e \in_{\mathrm{R}} \{0,1\}$, and sends e to the user A.

   (c) The user A sends $y \equiv s^e_A r$ (mod N) to the user B.

   (d) The user B checks that $x \equiv y^2 I^e_A$ (mod N). If the check is not valid, the user B quits the procedure.

   The user B accepts A's proof of identity only if all t round checks are successful.

Remark 2.7: In the parallel version of the protocol above, A sends B all the x_i $(i = 1, \ldots, t)$ simultaneously, then B sends A all the e_i $(i = 1, \ldots, t)$, and finally A sends all the y_i $(i = 1, \ldots, t)$ to B.

Furthermore, Fiat and Shamir modified the identification scheme above into a non-interactive digital signature scheme by replacing the verifier B's role by the prover with a pseudo-random function f.

Fiat-Shamir digital signature scheme (FSDS)

1. Preprocessing stage between the trusted center and each user
   Same as the preprocessing stage in FSIS.

2. To sign a message M:
   The user A picks $r_i \in_R Z_N^*$ $(i = 1, \ldots, t)$, and calculates $x_i \equiv r_i^2 \pmod N$ $(i = 1, \ldots, t)$, $f(M, x_1, \ldots, x_t)$ and sets its first t bits to e_i $(i = 1, \ldots, t)$. Furthermore, the user A computes $y_i \equiv s^{e_i} r_i \pmod N$ $(i = 1, \ldots, t)$ and sends M, e_i, y_i $(i = 1, \ldots, t)$ to the user B.

3. To verify A's signature on M :
   The user B calculates $z_i = y_i^2 I_A^{e_i} \pmod N$ $(i = 1, \ldots, t)$, $f(M, z_1, \ldots, z_t)$, and checks that its first t bits are equal to e_i $(i = 1, \ldots, t)$. If the checks are valid, the user B recognizes that M is A's valid message.

2.3 Known properties of the Fiat-Shamir scheme

Feige, Fiat and Shamir [FSS] showed that FSIS is provably secure. Namely,

Proposition 2.8 [FSS]: *The serial version of* FSIS, *where* $t = O(|N|)$, *is a zero-knowledge proof of knowledge.*

Although Feige, Fiat and Shamir [FSS] did not show that the parallel version of FSIS is zero knowledge, they did show that the parallel version of FSIS releases no "useful" knowledge that could help the verifier to impersonate the prover within the identification system. Namely,

Proposition 2.9 [FSS]: *If factoring is difficult, the parallel version of* FSIS *releases no transferable information.*

Note that Proposition 2.9 does not imply that the parallel version of FSIS releases no "useful" knowledge that could help the verifier to cheat *outside* the identification system.

Goldreich and Krawczyk [GKr] observed that non-zero-knowledgeness is an intrinsic property of the parallel version of the FSIS protocol.

Proposition 2.10 [GKr]: *If factoring is difficult, the parallel version of* FSIS *is not (black-box simulation) zero knowledge.*

Although the straightforward parallel version of FSIS is not zero-knowledge, Bellare, Micali, and Ostrovsky [BMO] proposed how to parallelize FSIS with preserving zero-knowledgeness. Their scheme is not three move and needs some additional interactions between the prover and the verifier.

In this paper, we use a term "parallel" version of protocols in the sense of the (three move) straightforward parallelization as in **Remark 2.7**.

With respect to the security of FSDS, Fiat and Shamir showed

Proposition 2.11 [FiS]: *When f is a truly random function,* FSDS *is existentially unforgeable under an adaptive chosen message attack unless factoring is easy.*

Remark 2.12: A variant of the Fiat-Shamir scheme has proposed [GQ1] and the security as in Proposition 2.9 has been considered [OhOk'88]. Brickell and McCurley [BM] proposed a modified Schnorr's identification scheme [Sch] based on a special discrete logarithm problem, and gave a formal proof on the security. Probably secure three move identification scheme based on the general problems is proposed by Okamoto [Oka].

3 Why does straightforward parallelization fail to be zero-knowledge ?

Feige, Fiat and Shamir's result in Proposition 2.9 guarantees a security of the parallel version of FSIS. On the other hands, Goldreich and Krawczyk's statement in Proposition 2.10 implies the parallel version of FSIS is not (blackbox simulation) zero knowledge. Many researchers [FSS, BC] remarked that the parallel version of FSIS could leak some "partial" information on the prover's secret.

Our first question is :

Question A: What information is released in the parallel version of FSIS ?

To prove a protocol to be zero knowledge, a main technique is to reset a (cheating) verifier, so called *resettable simulation* [GMR].

Many researchers [BC, BMO] have observed that the resettable simulation may not be applied to the following cheating verifier in the parallel version of FSIS.

> After receiving the prover's the message x_i $(i = 1, \ldots, t)$, the (cheating) verifier sends back bits e_i $(i = 1, \ldots, t)$ which are computed with dependence on x_i $(i = 1, \ldots, t)$, for example, $(e_1, \ldots, e_t) = g(x_1, \ldots, x_t)$ for a random hash function g.

In fact, Goldreich and Krawczyk's proof on the non-zero-knowledgeness of the parallel version of FSIS (generally, on the triviality of three move protocols) is based on a careful analysis of the cheating verifier with random hash function.

In the cheating strategy above, the verifier learns $(x_1, \ldots, x_t, y_1, \ldots, y_t)$ satisfying the conditions that $(e_1, \ldots, e_t) = g(x_1, \ldots, x_t)$ and $y_i \equiv s^{e_i} r_i \pmod N$ $(i = 1, \ldots, t)$. The (polynomial-time bounded) verifier without the secret s seems not to be able to generate such information by himself. Thus, we regard the information above as knowledge leaked in the parallel version of FSIS.

Our second question is as follows.

Question B: How useful is this information for the verifier ?

To clarify the role of this information above, we consider a verifier who acts as below.

> After receiving the prover's message x_i $(i = 1, \ldots, t)$, the verifier selects a message M and sends back bits e_i $(i = 1, \ldots, t)$ which are computed as $g(M, x_1, \ldots, x_t)$ for a *one-way* hash function g.

In the cheating method, the verifier learns $(x_1, \ldots, x_t, y_1, \ldots, y_t)$ satisfying the conditions that $(e_1, \ldots, e_t) = g(M, x_1, \ldots, x_t)$ and $y_i \equiv s^{e_i} r_i \pmod N$ $(i = 1, \ldots, t)$ for the message M selected by him. If g is a *one-way* hash function, we can regard $(e_1, \ldots, e_t, y_1, \ldots, y_t)$ as the prover's digital signature for the message M in FSDS with respect to the function g.

Our observation above implies that in the parallel version of FSIS a cheating verifier, who makes an access to the true prover in the parallel version of FSIS, gets the prover's digital signature of FSDS for any message M. Note that in the *serial* version of FSIS, even if a cheating verifier acts as the same as the above, the verifier cannot get any digital signature of FSDS.

4 Abuses of the parallel version

In this section we point out abuses of the parallel Fiat-Shamir scheme based on our remarks in the previous section.

4.1 Non-transferable information helps to forge secure digital signatures

We consider a practical system which consists of FSIS and FSDS.

Suppose a prover uses only one secret key s for his public information I in the system. Namely, the prover shows his identity via the *serial* version of FSIS using the secret s, and the prover signs messages via FSDS using the same secret s. This system is convenient for the prover because he keeps only one secret information.

However, if the prover shows his identity via the parallel FSIS, not via the *serial* one, this system is not secure for the prover. As we noted in the previous section, in the parallel version of FSIS a cheating verifier can get the prover's digital signature of FSDS for any message M while the verifier interacts with the prover in FSIS. In this system, FSDS(or FSIS) is not secure.

Note that "releasing no-transferable information" by Feige, Fiat, and Shamir [FSS] guarantees the security of the case only when the prover's secret information is used in the identification systems.

Remark 4.1: We may prevent the verifier's cheating above by using a different security parameter t in the signature stage and in the identification stage. However, such temporary protection never implies the provable security of the system.

4.2 Message authentication based on the public key

The message authentication is used as a data integrity mechanism to detect whether data have been altered in an unauthorized manner. An implementation of message authentication based on the conventional secret key cipher (e.g. DES) is Message Authentication Codes (MACs) [ISO]. The *public-key based message authentication* is defined as:

> **Validity:** In the authentication stage, *only* the user A can prove the validity of a message to any user B by using A's public key.

The authentication stage based on the public key needs an interaction between the prover and the verifier, while MACs is *non-interactively* verified by the only receiver who knows the same secret key as the sender has. Note that the digital signature [DH] is verified by anybody *without interaction* using only the signer's public key.

Desmedt [Des] and Guillou-Quisquater [GQ2] applied FSIS to the public-key based message authentication. Guillou and Quisquater modified the (extended) Fiat-Shamir identification scheme into a message authentication by using a one-way hash function. The one-way hash function is used to mix the message into the communication for the identification.

Guillou-Quisquater's Message Authentication
based on the (extended) Fiat-Shamir scheme

1. PREPROCESSING STAGE BETWEEN THE TRUSTED CENTER AND EACH USER
   In this system, the center's secret key is p, q (distinct large primes) and the public key is $N = pq$ and L. The center generates prover A's secret key s_A satisfying $1/s_A = (I_A)^{1/L} \pmod N$, where I_A is the identity of user A and is published to other users. Furthermore, a one-way hash function g is published to each user.

2. Authentication stage between the user A and the user B

(a) The user A sends his message M with his identify I_A to the user B.

Repeat step (b) to (e) t times.

(b) The user A picks $r \in_R Z_N^*$, and computes $x \equiv r^L \pmod N$ and $u = g(M, x)$. The user A sends x and u to the user B.

(c) The user B sends $d \in_R Z_L$ to the user A.

(d) The user A sends y such that $y \equiv s_A^d r \pmod N$ to the user B.

(e) The user B checks that $u = g(M, y^L I_A^d \pmod N))$. If the check is not valid, the user B quits the procedure.

The user B recognizes that M is A's valid message only if all t round checks are successful.

The serial version of the protocol above (when $t = O(|N|)$ and $L = O(1)$) is zero-knowledge, and the security of parallel versions, which are not zero-knowledge, is studied by Ohta and Okamoto [OhOk'88].

However, no discrepancy between the serial and the parallel of the message authentication based on the (extended) Fiat-Shamir scheme has known. We clarify the discrepancy.

Desmedt [Des] considered the *one-time-validity* of the message authentication and Okamoto and Ohta [OkOh'90] called the same notion *non-transitive signature*:

Validity: *Only* the user A can prove the validity of a message M to any user B by A's public key.

Non-transitivity: The user B cannot transfer the proof of A's origin of the message M to another user C.

We should notice that the ordinary (transitive) digital signature [DH] does not satisfy the condition of non-transitivity, i.e, in the digital signature any user B can transfer the proof of A's origin of the message M to another user C and the user C can check the correctness of the proof of A's origin of the message M using only A's public key.

Okamoto and Ohta implemented message authentication based on the modification of the prover's randomness in the (extended) Fiat-Shamir scheme.

Desmedt [Des] mentioned that the serial version of his message authentication is non-transitive (one-time-valid), however nothing was mentioned in the case of the parallel version. Note that the serial version of Guillou-Quisquater's message authentication is non-transitive. Okamoto-Ohta [OkOh'90] claimed, without formal discussion, that both the serial and the parallel version of the message authentication are non-transitive. But, our claim is as follows.

Claim: The parallel version of the Guillou-Quisquater, Okamoto-Ohta, and Desmedt's message authentication are *not* non-transitive.

A cheating method for a verifier in the Guillou-Quisquater message authentication is as follows. (This cheating is applied to other message authentication like as the Desmedt and Okamoto-Ohta's one.)

A (cheating) verifier manages to record the history of the communication with the prover. After receiving prover's message $M, x_1, \ldots, x_t$ and $u = g(M, x_1, \ldots, x_t)$ and the verifier sends back $d_i (i = 1, \ldots, t)$ which is computed as $(d_1, \ldots, d_t) = h(x_1, \ldots, x_t)$ by a one-way hash function h. After receiving the prover's answer $y_i (i = 1, \ldots, t)$ for $d_i (i = 1, \ldots, t)$, the verifier records $H = (M, x_1, \ldots, x_t, h, d_1, \ldots, d_t, y_1, \ldots, y_t)$ as the history of the communication with prover A. Once the verifier publishes the history H, anyone can check the validity and the origin of message M by calculating $u = g(M, {y_1}^L I_A^{d_1} \pmod N), \ldots, {y_t}^L I_A^{d_t} \pmod N))$, and $(d_1, \ldots, d_t) = h(x_1, \ldots, x_t)$.

Remark 4.2: The same kind of abuse as above cannot be applied to the scheme based on the *serial* version of the extended Fiat-Shamir scheme.

5 Positive applications of the parallel version

In this section, we consider positive applications of the parallel version.

Okamoto and Ohta [OkOh'89] proposed a blind signature scheme, which was introduced by Chaum [Ch'82], based on a combination of the parallel version of FSIS and FSDS. This is the first positive application of the parallel version of Fiat-Shamir scheme although Okamoto and Ohta did not clarify the distinction between the parallel version and the serial one of the Fiat-Shamir scheme. The technique used in Okamoto-Ohta scheme is more sophisticated than one observed in subsection 4.1, however Okamoto and Ohta's technique is applied to a special class of problems which satisfy a condition, so called *random self reducibility* [TW], and seems not to be applied to the parallel version of more general zero-knowledge protocols (e.g. the references [GMW, BCC]).

We consider positive applications of the parallel version of the Fiat-Shamir scheme, which can be applied to the parallel version of the more general protocols.

5.1 The parallel version of the Fiat-Shamir scheme leaves a trace

Our observations in the previous sections suggest that the parallel version of FSIS leaves some trace, unlike the case of the serial version of zero-knowledge FSIS. We positively apply the trace to message authentication with the proof of the origin and to a protection of divertibility of interactive protocols.

5.2 Testifiable message authentication

As we pointed out in the previous section, the message authentication based on the parallel FSIS does not satisfy the non-transitivity. We positively apply the transitive trace of authentication stage in the parallel version of FSIS.

In the message authentication based on the serial FSIS, the sender (signer) can deny the fact that the signer has shown authentication, because there are no evidence of the prover's proving stage. Okamoto and Ohta [OkOh'90] remarked this property as a merit to show the distinction between non-transitive signatures and Chaum's undeniable signature [CA]. Occasionally, however, we needs an evidence to avoid prover's denying the fact of his authentication on the message. The trace in the parallel version is useful for the evidence.

Suppose that user A sends a message M to user B. A *testifiable* message-authentication has the following properties.

> **Validity:** In the authentication stage, *only* the user A can prove the validity of a message M to any user B by A's public key.
> **Testifiability:** Any user C can check the fact that the user A has given the proof of A's origin on the message M by A's public-key *without interaction with* A.

It must be noted that the digital signatures [DH] satisfy the condition of testifiability, however, the digital signatures do not have the authentication stage where A can prove to B that he is A.

We propose a message-authentication which is a modification of the verifier's randomness in the parallel version of the message authentication using the Guillou and Quisquater's idea.

Proposed testifiable message authentication

1. PREPROCESSING STAGE BETWEEN THE TRUSTED CENTER AND EACH USER
   Same as the preprocessing stage in FSIS. Furthermore, two one-way hash function g and h are published to all users.

2. AUTHENTICATION STAGE BETWEEN THE USER A AND THE USER B

   (a) The user A sends his identity I_A and a message M_A to user B.

   (b) The user A picks $r_i \in_R Z_N^*$ $(i = 1, \ldots, t)$, and computes $x_i \equiv r_i^2 \pmod N$ $(i = 1, \ldots, t)$, and $u = g(M_A, x_1, \ldots, x_t)$. The user A sends $x_1, \ldots, x_t$ and u to the user B.

   (c) The user B selects a message R_B at random, calculates $h(R_B, x_1, \ldots, x_t)$. The user B sets its first t bits to e_i $(i = 1, \ldots, t)$ and sends e_i $(i = 1, \ldots, t)$ and R_B to the user A .

   (d) The user A computes $h(R_B, x_1, \ldots, x_t)$ and checks if the first t bits of $h(R_B, x_1, \ldots, x_t)$ are e_i $(i = 1, \ldots, t)$. If the check is not valid, the user A quits the procedure. Otherwise, the user A sends to B $y_i \equiv s^{e_i} r_i \pmod N$ $(i = 1, \ldots, t)$.

   (e) The user B checks that $u = g(M_A, x_1, \ldots, x_t)$ and $x_i \equiv y_i^2 I_A^{e_i} \pmod N$ $(i = 1, \ldots, t)$. If the check is not valid, the user B quits the procedure.

   After all procedures are passed, the user B accepts that M_A is A's valid message.

3. PUBLICATION AND VERIFICATION OF THE EVIDENCE OF THE AUTHENTICATION
   If the prover denies his authentication on the message M_A, the verifier shows
   $H = (I_A, M_A, u, x_1, \ldots, x_t, R_B, y_1, \ldots, y_t)$ as an evidence of the A's authentication on M_A. Anyone can accepts the A's authentication on M_A only if H satisfies the conditions that $u = g(M_A, x_1, \ldots, x_t)$, $(e_1, \ldots, e_t) = h(R_B, x_1, \ldots, x_t)$, and $x_i \equiv y_i^2 I_A^{e_i} \pmod N$ $(i = 1, \ldots, t)$.

The authors [SI] applied the proposed testifiable message authentication to a digital credit card system, where both the identification and the digital signature are required.

5.3 Protection against divertibility

Desmedt et al. [DGB] pointed out an abuse of FSIS, so called *Mafia fraud problem*, where an intermediate verifier B can masquerade as the genuine prover A to another (victimized) verifier C while A proves his identity to B; and B cancels any evidence which shows that B is assisted by A. This concept was formulated as divertibility of (zero-knowledge) protocols by Okamoto and Ohta [OkOh'89]. They proposed some types of measure to protect against such an abuse.

We propose a simple technique to protect against the abuse of divertibility of the parallel version of the Fiat-Shamir scheme, which cannot be applied to the serial one. Figure 1 describes the technical details on the divertibility of the parallel version of the Fiat-Shamir scheme. The divertibility is arisen from the property that there are no evidence which distinguishes two communication data,
$((x_1, \ldots, x_t), (\tilde{e}_1, \ldots, \tilde{e}_t), (y_1, \ldots, y_t))$ and $((\tilde{x}_1, \ldots, \tilde{x}_t), (e_1, \ldots, e_t), (\tilde{y}_1, \ldots, \tilde{y}_t))$.

Proposed countermeasure

The technique used in our proposed testifiable message authentication is useful to create an evidence which distinguishes the data. Consider the following modified protocol:

> After receiving the prover's first message $(x_1, \ldots, x_t)$, the verifier selects a random message R_V, computes $h(R_V, x_1, \ldots, x_t)$ and sets its first t bits to $e_1, \ldots, e_t$. Then the verifier sends the random message R_V to the prover instead of sending $e_1, \ldots, e_t$. The prover sends back the verifier $y_i \equiv u_i(y_i)^{e_i}(i = 1, \ldots, t)$, where $(e_1, \ldots, e_t) = h(R_V, x_1, \ldots, x_t)$ as the ordinary parallel Fiat-Shamir scheme. The verifier accepts the prover only if the checks $(e_1, \ldots, e_t) = h(R_V, x_1, \ldots, x_t)$, and $x_i \equiv y_i^2(I_A)^{e_i}(i = 1, \ldots, t)$ are passed.

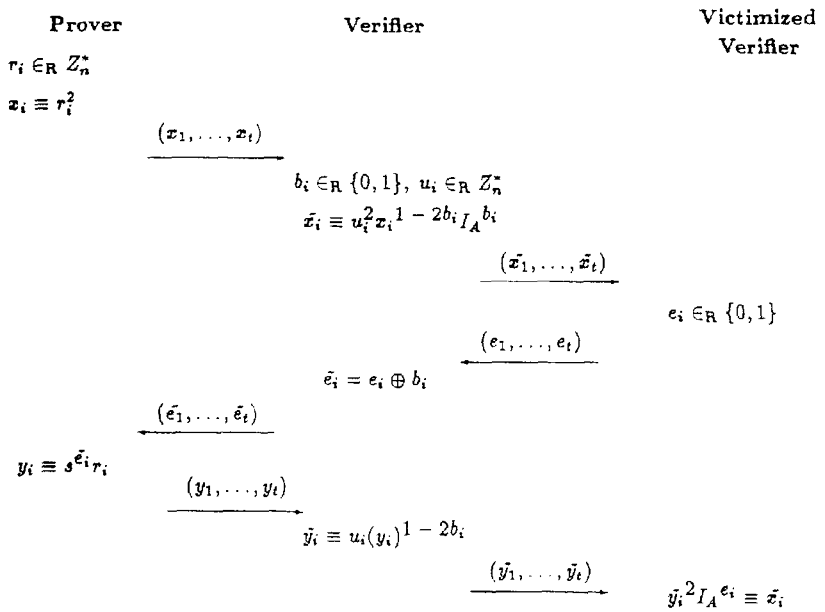
Prover
Verifier
Victimized Verifier
$r_i \in_R Z_n^*$
$x_i \equiv r_i^2$
$(x_1, \ldots, x_t)$
$b_i \in_R \{0,1\}$, $u_i \in_R Z_n^*$
$\tilde{x}_i \equiv u_i^2 x_i^{1-2b_i} I_A^{b_i}$
$(\tilde{x}_1, \ldots, \tilde{x}_t)$
$e_i \in_R \{0,1\}$
$(e_1, \ldots, e_t)$
$\tilde{e}_i = e_i \oplus b_i$
$(\tilde{e}_1, \ldots, \tilde{e}_t)$
$y_i \equiv s^{\tilde{e}_i} r_i$
$(y_1, \ldots, y_t)$
$\tilde{y}_i \equiv u_i(y_i)^{1-2b_i}$
$(\tilde{y}_1, \ldots, \tilde{y}_t)$
$\tilde{y}_i^2 I_A^{e_i} \equiv \tilde{x}_i$

Figure 1: Divertible ZK on the parallel Fiat-Shamir scheme

In this modified protocol, the way of the verifier's generating the challenge bits $(e_1, \ldots, e_t)$ is restricted and the verifier's computation in the original divertible protocol (Figure 1) cannot be apply to the modified protocol.

The proof on the correctness of our countermeasure is obtained from the same argument as the proof of the security of FSDS (Proposition 2.11). The protection is rather practical than theoretical because it is assumed in a way similar to Proposition 2.11 that the function h is a (blackbox) truly random function.

Remark 5.1: Ohta, Okamoto, and Fujioka [OOF] proposed how to protect the divertibility by using a bit commitment function. Their countermeasure is useful for both the serial version and the parallel version. However, our proposed countermeasure is applied to only the *parallel* version.

6 Concluding remarks

In this paper, we clarify the discrepancy between the serial version and the parallel version of zero-knowledge protocols, especially point out the relation between the "information" leaked in the parallel version of the Fiat-Shamir identification scheme and the Fiat-Shamir digital signature scheme. Furthermore, we consider the merit and demerit of the parallel version with comparing to the serial one. Note that our observation is applied to general zero-knowledge protocols, which is a sequential iteration of a three move protocol.

The security of the straightforward parallel execution of the Fiat-Shamir type identification scheme is characterized by some researchers [FSS, OhOk'88, FeS, BM, Oka]. However, their results heavily depend on the structure of the underlying problems (e.g. factorization, or discrete logarithm), and the technique of the proofs fails in the case of the straightforward parallel execution of the zero-knowledge protocol for general problems like as Graph-3-Colourability [GMW, BCC]. The security of these protocols are still unclear.

The security of three move protocols [FSS, OhOk'88, FeS, BM, Oka], which are based on some algebraic problems, guarantees only the case within the identification system, and nothing is mentioned outside the identification system.

The security of an identification and a signature is one of the central topics in modern cryptography, and many results are known. However, the aspect of these researches on the security is irrelevant to each other. We must study the security of the combination of the different objects.

Acknowledgments

The authors wish to thank Dr. Tatsuaki Okamoto of NTT Laboratories for his advice on digital signatures. The authors also would like to thank Dr. Hiroki Shizuya of Tohoku University. His observation on a weakness of zero-knowledge proofs in electronic fund systems inspired us to study this work. The authors are grateful to Prof. James L. Massey of Swiss Federal Institute of Technology Zurich for affording the first author an opportunity to present this research at ETH and for his encouragement of this work.

References

[BC] Brassard, G. and Crépeau, C., "Sorting out zero-knowledge," Advances in Cryptology - Eurocrypto'89, Lecture Notes in Computer Science 434, *Springer-Verlag*, Berlin, pp.181-191 (1990).

[BCC] Brassard, G., Chaum, D., and Crépeau, C., "Minimum Disclosure Proofs of Knowledge," *Journal of Computer and System Sciences*, Vol.37, No.2, pp.156-189 (October 1988).

[BM] Brickell, E. F. and McCurley, K.S "An Interactive Identification Scheme Based on Discrete Logarithms and Factoring," *Journal of Cryptology*, Vol.5, pp.29-40 (1992).

[BMO] Bellare, M., Micali, S., and Ostrovsky, R., "Perfect Zero-Knowledge in Constant Rounds," *ACM Annual Symposium on Theory of Computing*, pp.482-493 (May 1990).

[CA] Chaum,D. and van Antwerpen, H "Undeniable Signatures," Advances in Cryptology - Crypto'89, Lecture Notes in Computer Science 435, *Springer-Verlag*, Berlin, pp.212-216 (1989).

[Ch'82] Chaum,D., "Blind signature for Untraceable Payments," Advances in Cryptology - Crypto'82, *Plenum Press*, New York, pp.199-203 (1983).

[Des] Desmedt, Y., "Major security problems with the "unforgeable" (Feige-)Fiat-Shamir proofs of identity and how to overcome them,"In *Securicom 88, 6th worldwide congress on computer and communications security and protection,* pp.147-159, (March 1988).

[DGB] Desmedt, Y., Goutier, C. and Bengio,S.: "Special Uses and abuses of the Fiat-Shamir Passport Protocol,"Advances in Cryptology - Crypto'87, Lecture Notes in Computer Science 293, *Springer-Verlag*, Berlin, pp.21-39 (1988).

[DH] Diffie, W., and Hellman, M. "New Directions in Cryptology", IEEE Trans. on Info. Technology, vol. IT-22, 6(1976) pp.644-654 (1976).

[FeS] Feige, U. and Shamir, A., "Witness Indistinguishable and Witness Hiding Protocols," *ACM Annual Symposium on Theory of Computing*, pp.416-426 (May 1990).

[FiS] Fiat, A. and Shamir, A., "How to Prove Yourself," Advances in Cryptology - Crypto'86, Lecture Notes in Computer Science 263, *Springer-Verlag*, Berlin, pp.186-199 (1987).

[FSS] Feige, U., Fiat, A., and Shamir, A., "Zero-Knowledge Proofs of Identity," *ACM Annual Symposium on Theory of Computing*, pp.210-217 (May 1988), the final version: *Journal of Cryptology*, Vol.1, pp.179-194 (1988). v

[GKr] Goldreich, O. and Krawczyk, H., "On the Composition of Zero-Knowledge Proof Systems," ICALP'90, Lecture Notes in Computer Science 443, *Springer-Verlag*, Berlin, pp.268-282 (1990).

[GMR] Goldwasser, S., Micali, S., and Rackoff, C., "The Knowledge Complexity of Interactive Proof Systems," *SIAM Journal of Computing*, Vol.18, No.1, pp.186-208 (February 1989).

[GMW] Goldreich, O., Micali, S., and Wigderson, A., "Proofs that Yield Nothing But Their Validity and a Methodology of Cryptographic Protocol Design," *IEEE Annual Symposium on Foundations of Computer Science*, pp.174-187 (October 1986).

[GO] Goldreich, O. and Oren, Y., "Definitions and Properties of Zero-Knowledge Proof Systems," *Technical Report* #610, Technion - Israel Institute of Technology, Department of Computer Science, Haifa, Israel (February 1990).

[GQ1] Guillou,L.C., and Quisquater,J.J, "A Practical Zero-Knowledge Protocol Fitted to Security Microprocessors Minimizing Both Transmission and Memory," Advances in Cryptology - Eurocrypt'88, Lecture Notes in Computer Science 330, *Springer-Verlag*, Berlin, pp.123-128 (1988).

[GQ2] L.C.Guillou, and J.J.Quisquater, "A "Paradoxical" Identity-Based Signature Scheme Resulting from Zero-Knowledge," Advances in Cryptology - Crypto'88, Lecture Notes in Computer Science 403, *Springer-Verlag*, Berlin, pp.216-231 (1990).

[ISO] International Standard, "Banking - Approved algorithm for message authentication - ," *ISO 8731-1* (1987).

[Oka] Okamoto,T., "Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes," Entenxed Abstract for CRYPTO'92 (1992).

[OhOk'88] Ohta,K., and Okamoto,T., "A Modification of the Fiat-Shamir Scheme," Advances in Cryptology - Crypto'88, Lecture Notes in Computer Science 403, *Springer-Verlag*, Berlin, pp.232-243 (1990).

[OkOh'89] Okamoto,T., and Ohta,K., "Divertible Zero-Knowledge Interactive Proofs and Commutative Random Self-Reducibility," Advances in Cryptology - Eurocrypt'89, Lecture Notes in Computer Science 434, *Springer-Verlag*, Berlin, pp.134-149 (1989).

[OkOh'90] Okamoto,T., and Ohta,K., "How to utilize the randomness of Zero-Knowledge Proofs," Advances in Cryptology - Crypto'90, Lecture Notes in Computer Science 537, *Springer-Verlag*, Berlin, pp.456-475 (1991).

[OOF] Ohta,K., Okamoto,T., and Fujioka,A., "Secure bit commitment function against divertibility," *EUROCRYPTO'92 Extended Abstracts*, (May 1992).

[SI] Sakurai,K. and Itoh,T "Testifiable Identification and Its application to a Digital Credit Card," Proc. of the 1992 Symposium on Cryptography and Information Security, **1D**, Japan (April 1992).

[Sch] Schnorr, C. P., "Efficient identification and signatures for smart cards," Advances in Cryptology - Crypto'89, Lecture Notes in Computer Science 435, *Springer-Verlag*, Berlin, pp.239-252 (1990).

[TW] Tompa, M. and Woll, H., "Random Self-Reducibility and Zero-Knowledge Interactive Proofs of Possession of Information," *IEEE Annual Symposium on Foundations of Computer Science*, pp.472-482 (October 1987).