Strong Designated Verifier Proof Signature without Hash Functions and the Same Scheme for an Ad-hoc Group Ring

Ji-Seon Lee† *and Jik Hyun Chang*†,

†Dept. of Computer Science, Sogang University, Seoul, Korea

Summary
In this paper, we propose a strong designated verifier proof signature scheme for an ad-hoc group and discuss its security properties. The proposed scheme provides a way that leaks authoritative secrets to only a designated person anonymously by one of the ad-hoc group members and no one knows that the secret is from a group member or the recipient, except the recipient. This group is called a ring. At first we propose a strong designated verifier proof signature without using hash functions for one signer and then we propose the same scheme for an ad-hoc environment.

Key words:
Designated Verifier Proof, Ad-hoc Group, Ring Signature

Introduction

In 1996, Jakobsson et al. [7] introduced the concept of the designated verifier signature scheme which makes it possible for a signer to convince only the designated verifier that the signature is made by the signer. This is achieved since a designated verifier himself can efficiently simulate signatures that are indistinguishable from the signer's signature. Since the signer's public key and the designated verifier's public key are both included in the verification step, anyone can verify the signature. However, unlike ordinary digital signature schemes, no one can be convinced that who the real signer is, except the designated verifier. When the designated verifier Bob receives a signature from a signer Alice, he certainly trusts that it is from Alice upon verifying it, since he knows that he did not generate the signature himself. A designated verifier signature scheme is useful in some situations in which the signer should specify who may be convinced by the signer's signature. However, in some circumstances, the third party may be convinced with high probability that the signature intended for the designated verifier is actually generated by the signer. For example, the signature may be captured on the line by the third party before the designated verifier receives it. The third party can then confirm that the real signer is Alice. To protect the identity of the signer in such situations, the signer encrypts the signature with the designated verifier's public key so that only the designated verifier can get the signature generated by the signer with his secret key. This stronger requirement is called a strong designated verifier signature scheme and was discussed in [7]. Saeednia et al. [9] proposed a new efficient designated verifier signature scheme which directly provides the strongness property without requiring any encryption of the signatures. In their scheme, the third party cannot even verify the signature since the secret key of the designated verifier is involved in the verification step. If the secret key of the designated verifier is exposed to the public, then anyone can verify the signature. However, still no one can confirm that the signature is from the signer or the designated verifier.

Cramer et al.[5] proposed a new scheme for achieving 1-out-of *n* group signature that allows a signer to produce a signature in the name of an ad-hoc decided group of people, without requiring the interaction of the others. That is, any single signer can choose *n*-1 members, form a temporary group of *n* members including himself, and then generate a group signature without the assistance of the other *n*-1 members. That is, the group formation and the signature generation are both spontaneous. Anyone can be convinced that the generated group signature is from one of the group members, but no one can identify the real signer among the members. Later Rivest et al.[8] formalized this kind of signature called ring signatures. Subsequently, variant 1-out-of-*n* signature schemes [1,2,3,4] have been proposed. In 2003, Herranz and Saez [6] proposed a provably secure ring signature scheme in the random oracle model.

Rivest et al. [9] noticed that the designated verifier signatures can be implemented from ring signature scheme by including the verifier's public key in the ring. However, general ring signatures with simply involving the verifier's public key is not suitable to construct a strong designated verifier signature scheme.

In this paper, at first we propose a strong designated verifier proof signature scheme without using hash functions. Next, we propose a strong designated verifier proof signature scheme for the ad-hoc group called a ring. Since the proposed scheme is a strong designated verifier signature, only the designated verifier can verify the signature and be convinced that the signature is made by one of the ring members. Since it is a ring signature, even the designated verifier does not have any idea who the real signer is among the *n* ring members. The proposed scheme

Manuscript received December 5, 2006.
Manuscript revised December 25, 2006.

would be useful in some situations. Suppose that someone wants to leak authoritative information only to a designated person or an institute in an anonymous way. He would sign that information which can be verified only by the designated recipient. The recipient knows that the information is from one of the ring members. However, except for the recipient, no one can tell from whom comes the information between a ring and a recipient since the recipient can simulate the signature in an indistinguishable way.

In section 2, we review previously proposed schemes – Saeednia et al.'s strong designated verifier signature scheme, Herranz and Saez's ring signature scheme. In section 3, we propose a strong designated verifier proof signature scheme without using hash functions and its security properties. In section 4, we propose a strong designated verifier proof for an ad-hoc group based on the scheme proposed in section 3 and discuss its security properties. Some conclusions are made in section 5.

2. Preliminaries

2.1 Notations

- p, q : two large primes such that $q|p-1$
- g : a generator of a multiplicative subgroup of Z_p^*
- $H(\cdot)$: a collision resistant one-way hash function mapping $H:\{0,1\}^* \rightarrow Z_q$
- m : message to be signed where $m \in Z_p$
- (x_u, y_u) : the key pair of a user u, where $x_u \in Z_q^*$ is u's secret key and $y_u = g^{x_u} \bmod p$ is the corresponding public key

2.2 Saeednia et al.'s Strong Designated Verifier Proof Signature

We review the strong designated verifier proof signature scheme proposed by Saeednia et al. in 2003[9]. We suppose that Alice is the signer with key pair (x_A, y_A) and Bob is the designated verifier with key pair (x_B, y_B). Alice generates a strong designated verifier proof signature (r,s,t) for a message m and sends it with m to Bob.

Signature generation. Alice chooses two random number $k \in Z_q$ and $t \in Z_q^*$, and then generates a signature as follows :

$$c = {y_B}^k \bmod p$$
$$r = H(m,c)$$
$$s = kt^{-1} - rx_A \bmod q.$$

Signature verification. Upon receiving the transcript (r,s,t) with m, Bob verifies the signature by checking

$$H(m,(g^s {y_A}^r)^{tx_B} \bmod p) \stackrel{?}{=} r.$$

In this scheme, nobody else other than Bob can perform this verification since Bob's secret key is involved in the verification equation. Even if Bob reveals his secret key, he cannot convince any third party of the validity of a signature.

Transcript simulation. Bob selects $s' \in Z_q$ and $r' \in Z_q^*$ at random and computes the simulated signature as follows:

$$c = g^{s'} {y_A}^{r'} \bmod p$$
$$r = H(m,c)$$
$$\ell = r' r^{-1} \bmod q$$
$$s = s' \ell^{-1} \bmod q$$
$$t = \ell {x_B}^{-1} \bmod q.$$

Bob's simulated signature of m is (r,s,t). If Bob's secret key is given to the third party, he can verify the signature as Bob does. However, since Bob can generate the transcript in an indistinguishable way as above, the third party cannot tell who the real signer of that signature is.

2.3 Herranz and Saez's Provably Secure Ring Signature

Assume that there are n members in an ad-hoc group. Each member A_i, $1 \le i \le n$, has a key pair (x_i, y_i), the designated verifier Bob has his key pair (x_B, y_B), and $L = \{y_1, ..., y_n\}$.

Signature Generation. To generate a ring signature for a message m on behalf of n ring members $A_1, ..., A_n$, a signer A_s, where $s \in \{1, ..., n\}$, follows the below steps.

(1) For all $i \in \{1, ..., n\}$, $i \neq s$, A_s randomly chooses $a_i \in Z_q^*$ pairwise different.

A_s computes $R_i = g^{a_i} \bmod p$, for $1 \le i \le n$, $i \neq s$.

(2) A_s selects a random number $a \in Z_q$.

(3) A_s computes $R_s = g^a \prod_{i \neq s} y_i^{-H(m,R_i)} \bmod p$.

If $R_s = 1$ or $R_s = R_i$ for some $i \neq s$, go to step (2).

(4) A_s computes $\sigma = a + \sum_{i \neq s} a_i + x_s H(m, R_s) \bmod q$.

(5) The signature is then $(L, m, R_1, ..., R_n, h_1, ..., h_n, \sigma)$, where $h_i = H(m, R_i)$, for all $1 \le i \le n$.

Signature Verification. The recipient checks that $h_i \stackrel{?}{=} H(m, R_i)$, for all $1 \le i \le n$. If this holds, the recipient verifies that the following equation holds or not.

$$g^{\sigma} = \prod_{1 \le i \le n} R_i \prod_{1 \le i \le n} y_i^{h_i} \bmod p.$$

Herranz and Saez proved that their scheme satisfies anonymity and unforgeability in the random oracle model.

3. Strong Designated Verifier Proof Signature Scheme without Using Hash Functions

In this section we propose a strong designated verifier proof signature scheme without using hash functions which will be the basic scheme for the same scheme for an ad-hoc group called a ring.

3.1 Proposed Scheme

Suppose that the signer Alice wants to generate a strong designated verifier proof signature (*c*,*r*,*s*) for a message *m* and sends it with *m* to the designated verifier Bob.

Signature generation. Alice chooses two random numbers k_1 from Z_q^* and k_2 from Z_q and generates a signature (*c*,*r*,*s*) as follows:

$$c = g^{k_1} \bmod p$$
$$r = m y_B^{k_2} \bmod p$$
$$s = k_1^{-1}(x_A r - k_2) \bmod q.$$

Signature verification. Upon receiving (*c*,*r*,*s*) from Alice, Bob verifies the signature by checking the validity of the following equation:

$$r = m(y_A^{r} c^{-s})^{x_B} \bmod p.$$

We can see that this verification works correctly, since $r = m(g^{k_2})^{x_B} = m(g^{x_A r - s k_1})^{x_B} = m(y_A^{r} c^{-s})^{x_B} \bmod p$.

Transcript simulation. Bob can simulate the designated verifier signature (*c*,*r*,*s*) of *m*. Bob randomly chooses t_1 from Z_q^* and t_2 from Z_q. Then he computes (*c*,*r*,*s*) as follows:

$$c = y_A^{t_1^{-1}} \bmod p$$
$$r = m c^{t_2 x_B} \bmod p$$
$$s = t_1 r - t_2 \bmod q$$

This simulated signature can be verified correctly, since

$$\begin{aligned} r &= m(y_A^{r} c^{-s})^{x_B} \\ &= m(y_A^{r} (y_A^{t_1^{-1}})^{t_2 - t_1 r})^{x_B} \\ &= m y_A^{t_1^{-1} t_2 x_B} \\ &= m c^{t_2 x_B}. \end{aligned}$$

3.2 Security Analysis

In this subsection, we show that the proposed scheme is a strong designated verifier proof signature and it is unforgeable for any third party without Alice's secret key or Bob's secret key.

Strong designated verifier property: The proposed scheme is a designated verifier signature scheme. To prove this, we show that the simulated transcripts by Bob are indistinguishable from the transcripts generated by Alice. To simulate a signature, Bob randomly chooses t_1 from Z_q^* and t_2 from Z_q. The simulated signature (c, r, s) is generated from these two values. The probability that this simulated transcript by Bob is a signature randomly chosen from the set of all possible Alice's signature is $1/q(q-1)$. Therefore two signatures have the same probability distribution and hence the proposed scheme is a designated verifier signature scheme. The proposed scheme also satisfies the strongness property by involving Bob's secret key in the verification step.

Unforgeability: While the signature should be forgeable by Bob, it should not be forgeable by any third party. However, the attacker could forge a signature by setting $c = y_A \bmod p$, $r = m \bmod p$, and $s = r \bmod q$. This forged signature satisfies a verification step, i.e., $m(y_A^{r} c^{-s})^{x_B} = m(y_A^{r} y_A^{-r})^{x_B} = m = r$. This forgery is possible by setting the value of *c* with the public key of the signer. We do not consider this as a serious attack. Actually the signer would not generate the value of *c* with his public key. We can think of two scenarios that the attacker could try out to forge a signature.

Scenario 1. the attacker would try to generate a signature as Alice does. The attacker randomly chooses k_1 from Z_q^*, k_2 from Z_q and computes *c* and *r* by $c = g^{k_1} \bmod p$ and

$r = my_B{}^{k_2} \bmod p$. Next the attacker tries to find s which satisfies the verification step. To do this, the attacker should know the secret key of Alice.
Scenario 2. the attacker would try to simulate a signature as Bob does. The attacker randomly chooses t_1 from Z_q^*, t_2 from Z_q and computes c by $c = y_A{}^{t_1^{-1}} \bmod p$. Next the attacker tries to compute r by the formula $r = mc^{t_2 x_B} \bmod p$. Since Bob's secret key is needed to compute r, it is not feasible for the attacker to compute r and s accordingly. In both scenarios, the successful forgery by any third party means that the attacker solves the discrete logarithm problem.

In comparison with Saeednia et al.' scheme, since our scheme does not use any hash functions, the computational complexity is reduced in our scheme. And the security assumption of our scheme only depends on a public hard problem – the discrete logarithm problem. Our scheme is advantageous in environments where implementing hash function is difficult.

4. Proposed Strong Designated Verifier Proof for Ad-hoc Group Called a Ring

4.1 Proposed Scheme

We also assume that there are n members in an ad-hoc group and each group member has his key pair. The designated verifier also has his key pair (x_B, y_B), and $L = \{y_1, ..., y_n\}$.

Signature Generation. Among the n ring members, the signer A_s generates strong designated verifier proof ring signature as follows :

(1) A_s randomly chooses $a_i \in Z_q^*$ pairwise different and computes $R_i = g^{a_i} \bmod p$, for all $1 \le i \le n$, $i \ne s$.

(2) A_s chooses a random number $a \in Z_q^*$.

(3) A_s computes $R_s = g^a \prod_{i \ne s} y_i{}^{-H(m,R_i)} \bmod p$. If $R_s = 1$ or $R_s = R_i$ for some $i \ne s$, then go to step (2).

(4) A_s selects a random number $k_1 \in Z_q^*$ different from any of a_i s and computes $t = g^{k_1} \bmod p$.

(5) A_s selects a random number $k_2 \in Z_q$ and computes $r = my_B{}^{k_2} \bmod p$.

(6) A_s computes $K = a + \sum_{i \ne s} a_i + x_s H(m, R_s) \bmod q$.

(7) A_s computes $s = k_1^{-1}(Kr - k_2) \bmod q$.

(8) The signature is $(L, m, R_1, ..., R_n, h_1, ..., h_n, t, r, s)$, where $h_i = H(m, R_i)$, for all $1 \le i \le n$.

Signature Verification. The designated verifier checks whether $h_i = H(m, R_i)$ holds, for all $1 \le i \le n$. If this holds, the verifier computes A and checks the equality of the following formula:

$$A = \prod_{1 \le i \le n} R_i \prod_{1 \le i \le n} y_i{}^{h_i}$$
$$r \stackrel{?}{=} m(A^r t^{-s})^{x_B}.$$

If this holds, the designated verifier accepts that the signature is from one of the ring members.
We can see that this verification works correctly, since

$$\begin{aligned} r &= my_B{}^{k_2} \\ &= m(g^{Kr - sk_1})^{x_B} \\ &= m\{g^{Kr}(g^{k_1})^{-s}\}^{x_B} \\ &= m\{(\prod_{1 \le i \le n} R_i \prod_{1 \le i \le n} y_i{}^{H(m,R_i)})^r t^{-s}\}^{x_B}. \end{aligned}$$

Transcript Simulation. The designated verifier Bob simulates a transcript for the message m with his private key in an indistinguishable way as follows:

(1) Bob randomly chooses $a_i' \in Z_q^*$ pairwise different and computes $R_i' = g^{a_i'} \bmod p$, for all $1 \le i \le n$.

(2) Bob computes $A' = \prod_{1 \le i \le n} R_i' \prod_{1 \le i \le n} y_i{}^{H(m,R_i')}$.

(3) Bob selects a random number $t_1 \in Z_q^*$ different from any of a_i' and computes $t' = (A')^{t_1^{-1}} \bmod p$.

(4) Bob selects a random number $t_2 \in Z_q$ and computes $r' = m(A')^{x_B t_1^{-1} t_2} \bmod p$..

(5) Bob computes $s' = t_1 r' - t_2 \bmod q$.

(6) The signature is $(L, m, R_1', ..., R_n', h_1', ..., h_n', t', r', s')$, where $h_i' = H(m, R_i')$, for all $1 \le i \le n$.

If Bob reveals his secret key to the public, the simulated signature can be verified correctly by any third party as follows.

$$\begin{aligned} & m(A'^{r'} t'^{-s'})^{x_B} \\ &= m\{(A')^{r'}(A')^{t_1^{-1}(t_2 - t_1 r')}\}^{x_B} \\ &= m(A')^{x_B t_1^{-1} t_2} \\ &= r' \end{aligned}$$

4.2 Security Analysis

We show that the proposed scheme satisfies security requirements.

Signer Anonymity for the Designated Verifier : In order to prove that the signature is anonymous to the designated verifier , we show that the designated verifier has probability $1/n$ to guess which member of the ring actually generates a given signature. With similar approach to the proof of signer anonymity in the ring signature scheme [6], the probability that one of the ring member generates a signature correctly is $1/\{q(q-1)(q-2)...(q-n)(q-n-1)\}$. Since any member of the ring has the same probability to generate a signature, the proposed scheme preserves the anonymity property.

Signer Anonymity for the Third Party (Strong Designated Verifier Property): To prove that the proposed scheme is signer anonymous for the third party, we show that the simulated transcripts by Bob are indistinguishable from the transcripts generated by any of the ring members. Since the simulated signature depends on the random values of $t_1 \in Z_q^*$, $t_2 \in Z_q$, and a_i , for all $1 \le i \le n$, the probability that the transcript simulated by Bob correctly is $1/\{q(q-1)(q-2)...(q-n)(q-n-1)\}$. This probability is the same as the probability that a signature is generated by a ring member discussed above. Therefore, the probability that the third party can guess the real signer among $n+1$ participants – n ring members and the designated verifier – is $1/(n+1)$.

Since Bob has the ability to simulate the transcript in an indistinguishable way, no one can tell that the signature is from any of the ring members or Bob. The proposed scheme satisfies the strongness property of a designated verifier signature scheme since Bob's secret key is included in the signature verification step. That is, only Bob can verify the signature. If Bob's secret key is compromised, then anyone can verify the signature. However, still no one can tell that the signature is from a ring or Bob.

Unforgeability : While the signature should be simulated by Bob, it should not be forged or simulated by any third party. We can think of two scenarios that the attacker could try out to forge a signature.

Scenario 1. The attacker would try to generate a signature as any of the ring members does. The attacker follows the steps (1) through (5) in the signature generation. Next the attacker tries to compute K followed by s which should satisfy the verification step. However, to do this, the attacker should have any of the ring members' secret key which means that he should solve the discrete logarithm problem.

Scenario 2. The attacker would try to simulate a signature as Bob does. The attacker follows the steps (1) and (3) in the transcript simulation. He then tries to compute r' followed by s' which should satisfy the verification step. Likewise in scenario 1, to do this, the attacker should have the designated verifier's secret key which means that he should solve the discrete logarithm problem.

In both scenarios, the successful forgery by any third party means that the attacker solves the discrete logarithm problem.

5. Conclusions

In this paper we propose a strong designated verifier proof signature scheme without using hash functions and the same scheme for an ad-hoc group called a ring. The proposed scheme for an ad-hoc group provides a way to leak authoritative secrets anonymously by one of the ring members and no one knows that the secret is from a ring member or a designated verifier, except the designated verifier. Our scheme guarantees that only the designated verifier Bob can be convinced that the signature is really from one of the ring members. Even if Bob's secret key is exposed to the public, no one can tell that the signature is from one of the ring members or Bob. Furthermore, since it is a ring signature, even Bob does not know who the real signer is among the n ring members. We show that our scheme provides signer anonymity, unforgeability, and the strong designated verifier property.

References

[1] M. Abe, M. Ohkubo, and K. Suzuki, "1-out-of-n signatures from a variety of keys," Advancesin Cryptology - ASIACRYPT 2002, LNCS 2501, pp. 415-432, 2001.

[2] E. Bresson, J. Stern, and M. Szydlo, "Threshold ring signatures and applications to ad-hoc groups," Advances in Cryptology – CRYPTO 2002, LNCS 2442, pp. 465-480, 2002.

[3] S. Chow, S. Yiu, and L. Hui, "Efficient identity based ring signature," Applied Cryptography and Network Security, Third International Conference, ACNS 2005, Proceedings, LNCS 3531, pp.499-512, 2005.

[4] S.Chow, K. Liu, K. Wei, and T. Yuen, "Ring signatures without random oracles," Proceedings of AsiaCCS 2006, pp. 297-302, 2006.

[5] R. Cramer, I. Damgard, and B. Schoenmakers, "Proofs of partial knowledge and simplified design of witness hiding protocols," Advances in Cryptology – CRYPTO'94, LNCS 839, pp. 174-187, 1994.

[6] J. Herranz, and G. Saez, "Forking lemmas for ring signature schemes," Indocrypt'03, LNCS 1403, pp. 406-421, 2003.

[7] M. Jakobsson, K. Sako, and R. Impagliazzo, "Designated verifier proofs and their applications," Advances in

Cryptology - EUROCRYPT '96, LNCS 1070, pp. 143-154, 1996.
[8] R.L. Rivest, A. Shamir, and Y. Tauman, "How to leak a secret," Advances in Cryptology – ASIACRYPT 2001, LNCS 2248, pp.257-265, 2001.
[9] S. Saeednia, S. Kremer, and O. Markowitch, "An effcient strong designated verifier signature scheme," Information Security and Cryptology - ICISC'03, LNCS 2971, pp. 40-54, 2003.

Ji-Seon Lee received the B.S. and M.S. degrees in Computer Science from Sogang University. She is currently a Ph.D. candidate at Sogang University, Korea. Her research interests include cryptographic protocols and network security.

Jik Hyun Chang received the B.S. and M.S. degrees in Mathematics from Seoul National University, Korea. He received his Ph.D degree in the department of Computer Science & engineering from University of Minnesota, USA. Since 1986, he serves as a professor at Sogang University, Korea. His research interests include algorithms design and analysis, and cryptographic algorithms.