Online/Off-line Ring Signature Scheme with Provable Security

Jayaprakash Kar

Information Security Research Group
Faculty of Computing & Information Technology
Department of Information Systems
King Abdulaziz University, Jeddah-21589, PO Box-80221
Kingdom of Saudi Arabia
jayaprakashkar@yahoo.com

Abstract

The article proposes an Online/Off-line Ring Signature Scheme in random oracle model.Security of the scheme relies on both Computational Diffie-Hellman and k-CAA problems. The proposed scheme is proven the two most important security goals Existential Unforgeability and Signer Ambiguity. Also it has robustness property where the misbehavior of the signer can be detected. Signing process is performed in two phases online and offline. All heavy computations are performed in Off-line stage. So the overall computational cost is reduced and very less than the traditional signature scheme. The scheme can be applied to Mobile Ad Hoc Networks (MANETs) that undergo node mobility.

Keywords:unforgeability, ring members, signer anonymity

1 Introduction

A ring signature scheme allows members of a group to sign messages on behalf of the group without revealing their identities. This property is known as signer anonymity. Also it can not be distinguish any two arbitrary signatures that have been generated by the member of the same group. In case of group signature scheme, generation of the group is anomalous and there does not exist any group manager to renege the signer's identity. This states that, each user is joined with a public key of a typical signature scheme, a user can construct a group by taking the public keys of all the members of the group and his own public key. In general ring signature scheme are simplified form of group signature which consists of only users and no managers. The drawback of Group signature is that, it can be applied, if there is a cooperation among the users where as ring signature are useful if the cooperation exist among the ring members.

In conventional public key infrastructure(PKI), the public key is constructed as a random bit strings is independent of user's identity. Therefore, it is required a trusted third party or certificate authority (CA) to prove the relationship between the users and the cryptographic keys. So the verifier obtains the copy of the certificate of the user to check the certificate validity prior to the verification of signature. Whereas in ring signature scheme both the verifier and the public keys are verified in the ring.

2 Previous Works

In 1991, Chaum and Van Heyst [3] proposed Group signature scheme, where a trusted group manager broadcast a specific group of users and distributes the constructed keys to their members. Each group have an individual members, they sign on behalf of their group using these keys. It is not possible for the verifiers to distinguish the signatures generated by the group members where as the group leader can. He can revoke the mischievous signers's anonymity. In 2001 Rivest et.al [4]formalize the group signature where the group contains only users and no manager and proposed ring signature instead of group signature. The drawback in group signature is there should be proper synchronization among the group members where as in ring signature scheme no need of any co-operation among the members. Security of this scheme relies on Integer factorization problem. Each member is having public key of a signature scheme like RSA or ECDSA. A novel construction of general group signatures and multiparty was proposed by Chaum et al. [3] [26] where the scheme is inspired by zero knowledge proofs is not so efficient in security. In group signature scheme, Cramer et al. [9] has described how to provides witness-indistinguishable interactive proof where the Fiat-Shamir technique is applied and generated ring signature scheme.. For random self reducible language, the interactive SZK satisfies closer property with respect to monotone Boolean operations which have been shown by DeSantis et al. [8]. This is used to design ring signature scheme.Applying Identity based cryptography, using identities of the users as public key, Zhan and Kim [6] introduced the notion of ID-based ring signature scheme. Later on many identity based ring signature scheme [11] [12] [13] [25] have been proposed by different authors. The drawback of these schemes are the size of the ring signature linearly depended on group size. Therefore it is not possible to construct on large groups. Some of the authors [16] [14] [15] have proposed ring signature scheme of constant sizes where the size of the signature, but not same level of security of the scheme based on Integer factorization problem. In 2009 Liu et al. [2]introduced online/offline signature scheme, where the security relies on Integer factorization problem. This scheme is suited to implement on mobile devices but has not been defined how to design a constant size of online/offline signature. In random oracle model an efficient online/offline signature scheme have been proposed in [24] [28] which are suited to implement on WSNs and low processor devices.

3 Preliminaries

3.1 Bilinear Pairings

A mapping is defined between two groups known as bilinear pairing. The two form of bilinear pairings are Weil and Tate pairings on elliptic curve. Let $(\mathbb{G}_1; +)$ be a cyclic additive groups of prime order q with generator P. Similarly $(\mathbb{G}_2; .)$ be a multiplicative cyclic group of same order q.

$$e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2 \text{ is a computable and non-degenerated bilinear map.}$$

which satisfies the following properties:

- **Bilinear**:
  1. $e(aP, bQ) = e(P, Q)^{ab}$, where P, Q are the group elements belongs to $\mathbb{G}_1$ and $a, b \in \mathbb{Z}_q^*$.
  2. $e(P + Q, R) = e(P, R)e(Q, R)$, for any group elements P, Q, R belongs to $\mathbb{G}_1$.
- **Non-degenerate**: $\exists$ $P, Q \in \mathbb{G}_1$ such that, there is no such pairs $(P, Q) \in \mathbb{G}_2$ for which $e(P, Q) \neq 1$.
- **Computability**: $\exists$ an algorithm which can o compute $e(P, Q)$ efficiently for all $P, Q \in \mathbb{G}_1$.

3.2 Mathematical Assumptions

Definition 1. Decision Diffie-Hellman Problem (DDHP): *Let P be a generator of group $\mathbb{G}_1$. For the elements P, aP, bP, cP decide whether $c \equiv ab \mod q$, where $a, b, c \in \mathbb{Z}_q^*$.*

Definition 2. Computational Diffie-Hellman Problem (CDHP): *Given the elements P, aP, bP compute abP, for $a, b \in \mathbb{Z}_q^*$.*

4 Security Discussions

To present the security prove of the proposed scheme, Consider collusion attack with k-traitors denoted by k-CAA. It is defined as

Definition 3. k-Collusion Attack Algorithm Assumption(k-CAA) *Let k be an integer and $x \in \mathbb{Z}_q^*$, $P \in \mathbb{G}$, given $h_1, h_2 \dots h_k \in \mathbb{Z}_q^*$, $\frac{1}{h_1+x}P$, $\frac{1}{h_2+x}P \dots \frac{1}{h_k+x}P$ to compute $\frac{1}{h+x}P$ for some $h \notin \{h_1, h_2 \dots h_k\}$.*
We say that the (t, ϵ), k-CAA assumption holds in $\mathbb{G}$, if no t-time algorithm has advantage at least ϵ in solving the k-CAA problem in $\mathbb{G}$.

k-CAA is said to be (t, ϵ)-hard $\Longleftrightarrow$

$$Pr[\mathcal{A}(P, xP, h_1, h_2 \dots h_k, \tfrac{1}{h_1+x}P, \tfrac{1}{h_2+x}P \dots \tfrac{1}{h_k+x}P | x \in \mathbb{Z}_q^*, P \in \mathbb{G}, h_1, h_2 \dots h_k \in \mathbb{Z}_q^*)] \leq \epsilon$$

Definition 4. Modified k-CAA Assumption (Mk-CAA *The k-CAA problem in $\mathbb{G}$ is defined as follow: for some $x, a, b, h_1, h_2 \dots h_k \in \mathbb{Z}_p^*$ and $P \in \mathbb{G}$, given P, xP, aP, bP, xbP and k-pairs $(h_1, \frac{1}{h_1+x}(abP)), (h_2, \frac{1}{h_2+x}P(abP)) \dots (h_k, \frac{1}{h_k+x}(abP))$ output a new pair $(h^*, \frac{1}{(h^*+x)}abP)$ for some $h^* \notin \{h_1, h_2 \dots h_k\}$. We say that the (t, ϵ), k-CAA assumption holds in $\mathbb{G}$, if no t-time algorithm has advantage at least ϵ in solving the k-CAA problem in $\mathbb{G}$.*

k-CAA is said to be (t, ϵ)-hard $\Longleftrightarrow$

$$Pr[\mathcal{A}(P, xP, ap, bP, xbP, h_1, h_2 \dots h_k, \tfrac{1}{(h_1+x)}abP, \tfrac{1}{(h_2+x)}abP \dots \tfrac{1}{(h_k+x)}abP | x \in \mathbb{Z}_n, P \in \mathbb{G}_1, h_1, h_2 \dots h_k \in \mathbb{Z}_n)] \leq \epsilon$$

Definition 5. Weak Modified k-CAA Assumption (WMk-CAA) *The k-CAA problem in $\mathbb{G}$ is defined as follow: for some $x, h_1, h_2 \dots h_k \in \mathbb{Z}_p^*$ and $P \in \mathbb{G}$, given P, xP, aP, bP and k-pairs $(h_1, \frac{1}{h_1+x}P), (h_2, \frac{1}{h_2+x}P) \dots (h_k, \frac{1}{h_k+x}P))$ output a new pair $(h^*, \frac{1}{h^*+x}P)$ for some $h^* \notin \{h_1, h_2 \dots h_k\}$. We say that the (t, ϵ), k-CAA assumption holds in $\mathbb{G}$, if no t-time algorithm has advantage at least ϵ in solving the k-CAA problem in $\mathbb{G}$.*

k-CAA is said to be (t, ϵ)-hard $\Longleftrightarrow$

$$Pr[\mathcal{A}(P, xP, aP, bP, h_1, h_2 \dots h_k, \tfrac{1}{(h_1+x)}P, \tfrac{1}{(h_2+x)} \dots \tfrac{1}{(h_k+x)}P | x \in \mathbb{Z}_n, P \in \mathbb{G}_1, h_1, h_2 \dots h_k \in \mathbb{Z}_n)] \leq \epsilon$$

5 Ring Signature Scheme

Here we describe the framework of the ring signature scheme proposed by Rivest *e*t al. [1]. Let there are n possible signers belongs to a ring. The ring member who obtains the actual signature is referred as `signer` and other remaining members are `non-signer`. The possible signer's public key is P_i is accessed through PKI directory or certificate. The corresponding secret key is s_i. The ring signature comprises the following two probabilistic polynomial time solvable algorithms.

- `ring-sign` $\sigma \leftarrow$ `RingSign`$(m, P_1, P_2 \dots P_n, d, s_n)$. It takes the public keys $P_1, P_2 \dots P_n$ of the n ring members, s_n secret key of n^{th} member *i*.e actual signer returns a ring signature σ for message m.
- `ring-verify (‘‘true’’, ‘‘false’’)` $\leftarrow$ `RingVerify`$(\sigma, m, P_1, P_2 \dots P_n)$. It takes the signature σ, message m and the possible public keys $P_1, P_2 \dots P_n$ and returns either `true` or `false`.

6 Online/Off-line Ring Signature Scheme

Ring signature scheme allows member of group of users to sign a message on behalf of the group belonging to the ring, without revealing the actual signer's identities. This property is known as signer anonymity. So it allows the group of users to prove the verifier that the generator of the signature belongs to the group without disclosing identity of the generator. Also it is not possible to distinguish any two arbitrary signatures that have been generated by the same group member. Signer's anonymity is being protected because of the signatures have been issued by a member of a ring which is known to the verifier only but doesn't know exactly who has signed on the message.

In Online/Off-line Signature scheme, the signing process is performed in two phases namely online and off-line. All heavy computations are performed in off-line phase whereas in online phase comparatively light computations are performed. But the online phase is more efficient than the off-line. Total computational cost and time is very less as compare to the traditional ring signature scheme [29] [30].

6.1 Framework of ID-based Online/Off-line Ring Signature Scheme

It comprises five algorithms solvable in probabilistic polynomial time(PPT):

- `Setup`: $(param, msk) \leftarrow \mathrm{Set}(1^k)$. Let $k \in \mathbb{N}$ be a security parameter. The algorithm take k as input and returns publicly known global parameters *param* and master secret key *msk*.

- `Extract`: $d_{ID} \leftarrow Ext(1^k, param, msk, ID)$. For any user with identity ID, the algorithm takes k the security parameter, *param* the global parameter a and *msk* master secret key and returns d_{ID} as the secret key of the user.

- `Off-lineSign`: $\sigma_{off} \leftarrow Sgn_{off}(1^k, param, ID, msk, d_{ID})$ Let a group of n user participate in ring signing with identities $\{ID_1, ID_2 \dots ID_n\} \in \{0,1\}^*$. It is a probabilistic algorithm takes a security parameter k, global parameters *param* and identities $\{ID_1, ID_2 \dots ID_n\}$ as input to generate an off-line signature σ_{off}. Optionally, it may also take the actual each signer secret key d_k and public key Q_{ID_k}.

- `OnlineSign`: $\sigma \leftarrow Sgn_{on}(1^k, param, m, \sigma_{off}, \bigcup_{i=1}^{n}\{ID_i\}, d_k)$. The algorithm takes a security parameter k, the global parameters *param*, a message m, an off-line signature σ_{off} and the secret key d_k of one member who acts as actual signer. It generates a signature σ on message m.

- `Verify`: (“`accept`”, “`Reject`”) $\leftarrow Ver(1^k, param, \sigma, \bigcup_{i=1}^{n}\{ID_i\})$ takes a security parameter k, the global parameters *param*, a signature σ, message m and n users identities $\bigcup_{i=1}^{n}\{ID_i\}, \forall i = 1 \dots n$ returns“`accept`” if σ is valid else returns “`reject`”.

6.2 ID-based Online/Off-line Ring Signature Scheme for General Access Structure

An ID-based online/off-line Ring signature scheme comprises the following five probabilistic polynomial time (PPT) algorithms:

- `Setup`: $(param, msk) \leftarrow \mathrm{Set}(1^k)$. Let $k \in \mathbb{N}$ be a security parameter. The algorithm take k as input and returns publicly known global parameters *param* and master secret key *msk*.

- `Extract`: $d_{ID} \leftarrow Ext(1^k, param, msk, ID)$. For any user with identity ID, the algorithm takes k the security parameter, *param* the global parameter a and *msk* master secret key and returns d_{ID} as the secret key of the user.

- `Off-lineSign`: $\sigma_{off} \leftarrow Sgn_{off}(1^k, param, \bigcup_{i=1}^{n}\{\mathcal{U}_i\}, \bigcup\{d_{ID_{kj}}\}, \bigcup\{Q_{ID_{kj}}\})$ takes a security parameter k and the global parameters *param* to generate an off-line signature σ_{off}. It is a probabilistic algorithm taking n groups of user's identities $\bigcup_{i=1}^{n}\{\mathcal{U}_i\}$, $\mathcal{U}_i = \bigcup_{i=1}^{n}\{ID_{ij}\}$ as input, where n is the maximum number of groups of users to be included in the ring signature and . Optionally, it may also take the actual each signer secret key $\bigcup\{d_{ID_{kj}}\}$ and public key $\bigcup\{Q_{ID_{kj}}\}$ in one of the group $\mathcal{U}_k$, where $1 \leq k \leq n$ as input; it returns an off-line signature σ_{off}.

- `OnlineSign`: $\sigma_{on} \leftarrow Sgn_{on}(1^k, param, m, \sigma_{off}, \mathcal{U}_i)$, Where $\mathcal{U}_i = \bigcup_{i=1}^{n}\{ID_{ij}\}, \forall i = 1 \dots n$ takes a security parameter k, the global parameters *param*, a message m, an off-line signature σ_{off}, n groups of users identities $\{\mathcal{U}_i\}, \mathcal{U}_i = \bigcup_{i=1}^{n}\{ID_{ij}\}, \forall i = 1 \dots n$ generates a signature σ.

- `Verify`: (“`accept`”, “`Reject`”) $\leftarrow Ver(1^k, param, \sigma)$ .This is the verification algorithm which takes the input k, *param*,σ, m and user's identities $\{\mathcal{U}_i\}, \mathcal{U}_i = \bigcup_{i=1}^{n}\{ID_{ij}\}, \forall i = 1 \dots n$ of the group and returns“`accept`” if σ is valid else returns “`reject`”.

7 Security Notions

The two most important security goals of ring signature scheme are **Signer Ambiguity** and **Existential Unforgeability**. These can be define as

Definition 6. *(***Signer Ambiguity***) An identity-based ring signature scheme for n groups of users with identities $L = \{ID_1, ID_2 \dots ID_n\}$ is said to have the unconditional signer ambiguity if for any $ID_i, 1 \leq i \leq n$, message m and signature $\sigma \leftarrow Sgn_{on}(1^k, param, m, \sigma_{off}, \bigcup_{i=1}^{n}\{ID_i\}, d_k)$, where d_k is the secret key of the actual sigher, any unbound adversary $\mathcal{A}$ takes L, m and σ as input returns the actual signer indexed by k with probability less than $\frac{1}{n}$.*

Definition 7. *(***Existential Unforgeability***)An identity-based ring signature scheme is said to be secure against existential unforgebility under adaptive chosen-message attacks (*`EUF-IDRS-CMA2`*), if there does not exist any adversary with non-negligible advantage in* `EUF-IDRS-CMA2`*-game.*

`EUF-IDRS-CMA2`-game:
The game is played between a challenge $\mathcal{C}$ and adversary $\mathcal{A}$. Formally it is defined as

- **Setup:** The challenger $\mathcal{C}$ runs the algorithm `Setup` that takes the security parameter $k \in \mathbb{N}$, generates public system parameters *params* and master secret key s. Sends public system parameter *params* to $\mathcal{A}$. the system parameters and sends to the adversary $\mathcal{A}$. The adversary $\mathcal{A}$ performs polynomially bounded number of queries in adaptive manner *i.e* each query may depend on the answer of the previous query:
- **Attack:**
  1. `Key Extraction Oracle`: when $\mathcal{A}$ requests the private key of the actual signer on an identity ID, $\mathcal{C}$ executes the algorithm **Extract** and obtains the secret key. Formally $Ext(1^k, param, msk, ID) \leftarrow d_{ID}$. Then sends to the adversary $\mathcal{A}$.
  2. `Off-line Signing Oracle`: $\mathcal{A}$ chooses n users identities $\bigcup_{i=1}^{n}\{ID_i\}, \forall i = 1 \dots n$ requests the off-line signature, $\mathcal{C}$ executes the algorithm **Off-Sign** and obtains the offline signature σ_{off}. Then Sends to $\mathcal{A}$.
  3. `Online Signing Oracle`: $\mathcal{A}$ submits signing oracle q_s number of times in adaptive manner as : On any message m and n users identities $\bigcup_{i=1}^{n}\{ID_i\}, \forall i = 1 \dots n$, $\mathcal{C}$ runs the algorithm **On-Sign** and obtains the online signature σ_{on}. Then sends to $\mathcal{A}$.
- **Forgery:**
  After executing the queries polynomial number of times, $\mathcal{A}$ obtains the signature σ^* and the identities $\bigcup_{i=1}^{n}\{ID_i^*\}$ of the n users such that
  1. $(\bigcup_{i=1}^{n}\{ID_i^*\}, m^*)$ has not been asked as one of the `off-line signing` queries and `online signing` queries.
  2. `Key Extraction` queries never returns each of the secret keys in $(\bigcup_{i=1}^{n}\{d_{ID_i^*}\}$.
  3. `Verify`$(\sigma, (\bigcup_{i=1}^{n}\{ID_i^*\}) =$ `Accept`.

$\mathcal{A}$ wins the above game with the probability

$$Pr[Suss_{\mathcal{A}}^{\texttt{EUF-IDRS-CMA2}}(k)] \leq \frac{1}{2} + \epsilon.$$

8 Proposed Online/Off-line Ring Signature Scheme

8.1 Construction

- `Setup` : Given security parameters k, the KGC chooses groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of prime order q. A generator P of $\mathbb{G}_1$, a bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ and two collision resistant hash function $\mathcal{H}_0$. Where $\mathcal{H}_0 : \{0,1\}^* \to \mathbb{G}_1$, $\mathcal{H} : \mathbb{G}_1 \times \{0,1,\}^* \to \mathbb{F}_q^*$. A key derivation function $\mathcal{F} : \mathbb{G}_2 \to \{0,1\}^k$. Where $|k|$ is the length of the key strings. It chooses a master-key $s \in \mathbb{F}_q^*$ and computes $P_{pub} = sP$. The KGC publishes the system public parameters $params = <\mathbb{G}_1, \mathbb{G}_2, e, q, p, P_{pub}, \mathcal{H}_0, \mathcal{H}>$.

- KeyGen : The Signer submits her identity $ID \in \{0,1\}^*$ to KGC. KGC computes the Signer's public key $Q_{ID} = \mathcal{H}_0(ID) \in \mathbb{G}_1$ and private key $d_{ID} = sQ_{ID}$. Then both the keys are to be send through a secure channel to the signer.

- Ring Signing: Let $\mathcal{U} = \{\mathcal{U}_1 \ldots \mathcal{U}_n\}$ be the group of participants in the signing process. Let the actual signer is $\mathcal{U}_k$. Her public and private keys are Q_{ID_k} and d_k respectively. The signer computes the following to generate the ring signature on behalf of the group $\mathcal{U}$. The computation is carried out in the following two phases:

    1. Off-line: At the Off-line phase the signer chooses $r_i \in \mathbb{Z}_q^*$ randomly and computes

    $$V_i = r_i \cdot P_{pub}. \ \forall i = 1, 2 \ldots n.$$

    2. Online: At this phase, the signer chooses $t_i \in \mathbb{Z}_q^*$ $\forall i \neq k$ randomly and computes the online signature on message m as

    $$\sigma_k = \frac{P}{s(Q_{ID_k}\mathcal{H}(m)+r_k)} - \frac{1}{s(Q_{ID_k}\mathcal{H}(m)+r_k)} \sum_{i \neq k} t_i(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_i), \ \sigma_i = t_iP$$
    $$\forall i \in \{1, 2 \ldots n\} \setminus \{k\}$$

    The final signature $\sigma = \{\bigcup_{i \neq k}\{\sigma_i\}, \sigma_k, \bigcup_{i=1}^{n}\{V_i\}\}$.

- Verify: The signature $\sigma = \{\bigcup_{i \neq k}\{\sigma_i\}, \sigma_k, \bigcup_{i=1}^{n}\{V_i\}\}$ on message m is accepted $\Longleftrightarrow$

$$\prod_{i=1}^{n} e(P_{pub}Q_{ID_i}\mathcal{H}(m) + V_i, \sigma_i) = e(P, P)$$

8.2 Robustness

Definition 8. *A multi-party signature scheme is said to be robustness if the misbehavior of any participating signer can be detected, and the final signature will be invalid on proof of correctness of the verification equation even if there exit only one misbehaving signer.*

The following proof of correctness shows that the proposed scheme satisfies robustness property.

8.2.1 Proof of Correctness

$\prod_{i=1}^{n} e(P_{pub}Q_{ID_i}\mathcal{H}(m) + V_i, \sigma_i)$

$$\begin{aligned}
&= \prod_{i \neq k}^{n} e(P_{pub}Q_{ID_i}\mathcal{H}(m) + V_i, \sigma_i) e(t_i(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_k), \sigma_k) \\
&= e(t_i(\textstyle\sum_{i \neq k}^{n} t_iP_{pub}Q_{ID_i}\mathcal{H}(m) + V_i), P) e(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_k, \sigma_k) \\
&= e(t_i(\textstyle\sum_{i \neq k}^{n} P_{pub}Q_{ID_i}\mathcal{H}(m) + V_i), P) \cdot \\
&e(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_k, \tfrac{P}{s(Q_{ID_k}\mathcal{H}(m)+r_k)} - \tfrac{1}{s(Q_{ID_k}\mathcal{H}(m)+r_k)} \textstyle\sum_{i \neq k}(t_i(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_i) \\
&= e(t_i(\textstyle\sum_{i \neq k}^{n} P_{pub}Q_{ID_i}\mathcal{H}(m) + V_i), P) \cdot \\
&e(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_k, \tfrac{1}{s(Q_{ID_k}\mathcal{H}(m)+r_k)}\{P - \textstyle\sum_{i \neq k} t_i(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_i)\} \\
&= e(t_i(\textstyle\sum_{i \neq k}^{n} P_{pub}Q_{ID_i}\mathcal{H}(m) + V_i), P) \cdot \\
&e(P_{pub}(Q_{ID_k}\mathcal{H}(m) + r_k), \tfrac{1}{s(Q_{ID_k}\mathcal{H}(m)+r_k)}\{P - \textstyle\sum_{i \neq k} t_i(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_i)\} \\
&= e(t_i(\textstyle\sum_{i \neq k}^{n} P_{pub}Q_{ID_i}\mathcal{H}(m) + V_i), P) \cdot \\
&e(sP(Q_{ID_k}\mathcal{H}(m) + r_k), \tfrac{1}{s(Q_{ID_k}\mathcal{H}(m)+r_k)}\{P - \textstyle\sum_{i \neq k} t_i(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_i)\} \\
&= e(t_i(\textstyle\sum_{i \neq k}^{n} P_{pub}Q_{ID_i}\mathcal{H}(m) + V_i), P) \cdot \\
&e(P, \{P - \textstyle\sum_{i \neq k} t_i(P_{pub}Q_{ID_k}\mathcal{H}(m) + V_i)\} = e(P, P)
\end{aligned}$$

9 Security Analysis

Theorem 1. *The proposed scheme has unconditional signer ambiguity.*

Proof. Let the ring signature be $\sigma = \{\sigma_1, \sigma_2 \ldots \sigma_n\}$ of the set of n participant users generated by the private key d_k of the actual signer. All t_i are chosen randomly from $\mathbb{Z}_q^*$ and computed $\sigma_i = t_iP$ on message m for all $i = 1, 2 \ldots n$ except $i \neq k$. the group of signature $\{\sigma_1, \sigma_2 \ldots \sigma_n\}$ has $|\mathbb{G}_1|^{n-1}$ possible values for which can be chosen by signature generation procedure with equal possibility except the actual signer. These $t_i, \mathcal{H}$ and s_k are used to compute σ_k. V_i is computed in the off-line phase by r_i for all $i = 1, 2 \ldots n$.

Note the distribution $\{\sigma_1, \sigma_2 \ldots \sigma_n\}$ are identical to the distribution $\{t_1P, t_2P \ldots t_nP : \sum_{i=1}^{n} t_iP = C_1\}$ and $\{r_1P_{pub}, r_2P_{pub} \ldots r_nP_{pub} : \sum_{i=1}^{n} r_iP_{pub} = C_2\}$. Where C_1 and C_2 are elements of $\mathbb{G}_1$ by closure property. These depend on n and m. Hence for any unbounded adversary Adv, any set of users and random k, the probability $Pr[Adv(\sigma) = k] \leq \frac{1}{n}$. □

Definition 9. *In random oracle model, a forger $\mathcal{F}$ is said to be $(t, q_{h_0}, q_h, q_e, q_s, \epsilon)$-break, the online/off-line signature scheme under adaptive chosen message attack, if after submitting at most q_{h_0} and q_h hash queries, q_e key extraction query, q_s online/off-line signing queries in additive manner with t-processing time, it obtains a valid forge signature with probability at least ϵ.*
$(t, q_{h_0}, q_h, q_e, q_s, \epsilon)$-secure, if there does not exist any forger who $(t, q_{h_0}, q_h, q_e, q_s, \epsilon)$-breaks the scheme.

Theorem 2. *In random oracle model, if $\exists$ a $(t, q_{h_0}, q_h, q_e, q_s, \epsilon)$-forger can obtains a valid forge online/off-line signature for n members, then $\exists$ (t^*, ϵ^*)-algorithm can solve CDH and q_s-CAA problems where*

$$\epsilon^* \geq (\tfrac{1}{q_s+1})\epsilon, t^* \leq (nq_s + 3n + 1)t_{smul} + 2(n-1)t_{add} + (n-1)t_{mul} + (n-1)t_{inv}$$

Proof. To prove, we consider the security model of Rivest *e*t al. [1] defined in section-5 and follow similar proof of Zhang *e*t al. [7]. Let there are n ring members denoted by the set $\mathcal{X}$ are associated with public keys $\{P_1, P_2 \ldots P_n\}$ are given to the adversary $\mathcal{A}$ and simulates the ring signing and hashing oracles. The target of $\mathcal{A}$ is to obtain a valid forge online/off-line ring signature in $\mathcal{X}$ with condition that prior to this, m has not been submitted to the oracle of online/off-line ring signing.

Let us assume that, $\exists$ a $(t, q_{h_0}, q_h, q_e, q_s, \epsilon)$-forger $\mathcal{F}$ algorithm can obtain a valid forge online/off-line signature of a set of ring members of size n. Let $\mathcal{F}$ construct a probabilistic polynomial time algorithm $\mathcal{B}$ and run as subroutine to solve CDH and q_s-CAA problems.

- q_s-**CAA** : Let $\mathcal{B}$ is given a challenges as :
  Given $P \in \mathbb{G}_1, Q = xP, h_1, h_2 \ldots h_{q_s} \in \mathbb{Z}_q^*$ and $\frac{1}{h_1+x}P + \frac{1}{h_2+x}P \cdots \frac{1}{h_{q_s}+x}P$.
  Compute $\frac{1}{h+x}P$ for some $h \notin \{h_1, h_2 \ldots h_{q_s}\}$
- **CDHP**: Let $\mathcal{B}$ is given a challenges as :
  Given $P, Q = aP, R = bP \in \mathbb{G}_1, a, b \in \mathbb{Z}_q^*$
  Compute abP.

Algorithm $\mathcal{B}$ performs the following simulation by interacting with the forger $\mathcal{F}$. Let $\mathcal{C}$ is given the random instances $<a, aP, bP>$ of CDHP and compute abP. $\mathcal{C}$ runs $\mathcal{B}$ as subroutine and behaves as $\mathcal{A}$'s challenger in `EUF-IDRS-CMA2`-game. While the game played between $\mathcal{A}$ and $\mathcal{C}$, $\mathcal{A}$ asks to $\mathcal{C}$ for answer to the $\mathcal{H}_0$ and $\mathcal{H}$ random oracles. In fact the answers are obtained randomly and are stored in a list to preserve consistency and avoid collision. So $\mathcal{C}$ constructs two lists as L_0 and L_1 for both the random oracles $\mathcal{H}_0$ and $\mathcal{H}$ respectively.

$\mathcal{C}$ provides $\mathcal{A}$ the system parameters with $P_{pub} = aP$. At any time $\mathcal{A}$ can submit the queries on random oracle oracles $\mathcal{H}_0$, $\mathcal{H}$, key extraction and online and off-line signing in adaptive manner. To answer these queries, $\mathcal{A}$ performs the following oracles:

- `Queries on Oracle` $\mathcal{H}_0$: In this request, we consider part of challenge aP in the answer to the series of queries on $\mathcal{H}_0$. When an identity ID is submitted to oracle $\mathcal{H}_0$, $\mathcal{A}$ toss a coin $T \in \{0, 1\}$ returns 0 and 1 with probability μ and $\mu - 1$. $\mathcal{A}$ picks randomly $\lambda_i \in \mathbb{Z}_q^*$ and continue the process until λ_i does not belongs to L_0 list. It has with the following two choices :

  1. If $T = 0$, then $\mathcal{H}_0(ID) = \lambda_iP$.
  2. If $T = 1$, then $\mathcal{H}_0(ID) = \lambda_i(aP)$.

  $\mathcal{C}$ includes the tuples $<ID, \lambda_i, T>$ in L_0 in both the above cases.

- `Queries on key extraction`: Let $\exists$ a $(t, q_{h_0}, q_h, q_e, q_s, \epsilon)$-forger $\mathcal{F}$ can obtains a valid forge online/off-line signature of the ring members of size n. Let $\mathcal{F}$ constructs an algorithm $\mathcal{A}$ to solve q_s-CAA problem, *i.*e

  Private key depends upon the identity ID. So when $\mathcal{A}$ requests the private key, $\mathcal{A}$ takes the corresponding tuple $<ID, \lambda_i, T>$ from the list L_0 and checks $T = 0$ or 1. If $T = 1$, $\mathcal{A}$ returns "`failure`" and halts it. Else $\mathcal{A}$ searches for other tuples .

  - For $T = 0$, if the tuples $<ID, \lambda_i, T>$ is in L_0 , then computes the private key as $d_{ID} = \lambda_i(bP)$ and the computation is known to $\mathcal{C}$.
  - For $T = 1$, $\mathcal{C}$ does not know both the value a and b. This results "`failure`" for this identity.

- `Queries on Oracle` $\mathcal{H}$: In $\mathcal{H}$-query, $\mathcal{A}$ runs at most q_f queries as $\{h_1, h_2 \dots h_{q_s}\}$ obtains q_h answers $\{u_1, u_2 \dots u_{q_h}\}$ on $m_i, 1 \leq i \leq q_h$. When $\mathcal{A}$ asks these queries, $\mathcal{C}$ searches the entry in L_1 list. If it is found, the answer is return to $\mathcal{A}$, else the answer is taken as a random value and given to $\mathcal{A}$. Also $\mathcal{F}$ is given the public keys $\mathcal{X} = \{P_1, P_2 \dots P_n\}$ of the n ring members.

- `Online/Off-line Signing Query`: $\mathcal{A}$ acts as a actual signer and chooses $\alpha_1 = 1, \alpha_2, \dots \alpha_n$ randomly from $\mathbb{Z}_q^*$. $\mathcal{A}$ initializes :

`Setup`

$P_1 = d_1Q$
$P_2 = \alpha_2Q + h(\alpha_2 - d_2)P$
$\vdots$
$P_n = \alpha_nQ + h(\alpha_n - d_n)P$

Assume that, there are n number of ring members with identities $\{ID_1, ID_2 \dots ID_n\}$ have private keys $\{d_1, d_2 \dots d_n\}$ respectively. Where $d_i = sQ_{ID_i}, Q_{ID_i} = \mathcal{H}_0(ID_i)$. The online/off-line signing oracle query for u_i are prepared by $\mathcal{F}$. $\mathcal{A}$ obtains the signatures as

$$\Sigma_i = \{\sigma_{i1}, \sigma_{i2}, \dots \sigma_{in}\}$$

and given to $\mathcal{F}$, where

$\sigma_{i1} = (1-\alpha) \cdot \frac{1}{h_j+x}P$
$\sigma_{i2} = (\alpha_2 - d_2)^{-1} \cdot \frac{1}{h_j+x}P$
$\sigma_{i3} = (\alpha_3 - d_3)^{-1} \cdot \frac{1}{h_j+x}P$
$\vdots$
$\sigma_{ik} = (-1)^k(\alpha_k - d_k)^{-1} \cdot \frac{1}{h_j+x}P$
$\vdots$
$\sigma_{in} = (\alpha_n - d_n)^{-1} \cdot \frac{1}{h_j+x}P$

`Verify`

Now we need to verify that Σ_i should pass the following verification equation, else the process halt and returns "`failure`" .

$$\prod_{k=1}^{n} e(P_{pub}Q_{ID_k}\mathcal{H}(m_i) + P_k, \sigma_{ik}) = e(P, P) \tag{1}$$

$$\begin{aligned}
&\textstyle\prod_{k=1}^{n} e(P_{pub}Q_{ID_k}\mathcal{H}(m_i) + P_k, \sigma_{ik}) \\
&= \textstyle\prod_{k=1}^{n} e(h_jd_kP + P_k, \sigma_{ik}) \\
&= e(h_jd_1P + P_1, \sigma_{i1}) \textstyle\prod_{k=2}^{n} e(h_jd_kP + d_kQ + (\alpha_k - d_k)(Q + hP), (\alpha_k - d_k)^{-1}\frac{1}{h_j+x}P) \\
&= e(h_jd_1P + d_1Q, (1-\alpha) \cdot \frac{1}{h_j+x}P) \textstyle\prod_{k=2}^{n} e(h_jd_kP + d_kQ + (\alpha_k - d_k)(Q + hP), (\alpha_k - d_k)^{-1}\frac{1}{h_j+x}P) \\
&= e(h_jd_1P + xd_1P, (1-\alpha) \cdot \frac{1}{h_j+x}P) \textstyle\prod_{k=2}^{n} e(h_jd_kP + d_kQ + (\alpha_k - d_k)(Q + hP), (\alpha_k - d_k)^{-1}\frac{1}{h_j+x}P) \\
&= e((h_j+x)d_1P, (1-\alpha) \cdot \frac{1}{h_j+x}P) \textstyle\prod_{k=2}^{n} e(h_jd_kP + d_kQ + (\alpha_k - d_k)(Q + hP), (\alpha_k - d_k)^{-1}\frac{1}{h_j+x}P) \\
&= e(P, P)^{(1-\alpha)d_1} \textstyle\prod_{k=2}^{n} e((h_j+x)d_kP, (\alpha_k - d_k)^{-1}\frac{1}{h_j+x}P)e((\alpha_k - d_k)(Q + hP), (\alpha_k - d_k)^{-1}\frac{1}{h_j+x}P) \\
&= e(P, P)^{(1-\alpha)d_1} \textstyle\prod_{k=2}^{n} e(P, P)^{(-1)^t(\alpha_k - d_k)^{-1}d_k} = e(P, P)
\end{aligned}$$

`Output`

At the end of the simulation, $\mathcal{F}$ returns a message-signature pair $(m, \sigma = \{\sigma_1, \sigma_2 \dots \sigma_n\})$ of n ring members with public keys $P_1, P_2 \dots P_n$. Note that the hash value of m is some u_k such that, there is not submitted any signature query for m. $\mathcal{A}$ returns "`failure`"

and halt for $u_k = h$,else
$\prod_{i=1}^{n} e(P_{pub}Q_{ID_i}\mathcal{H}(m) + P_i, \sigma_i)$
$= \prod_{i=1}^{n} e(d_i hP + P_i, \sigma_i)$
$= \prod_{i=1}^{n} e(d_i hP + \alpha_i Q + h(\alpha_i - d_i)P, \sigma_i)$
$= \prod_{i=1}^{n} e(d_i hP + \alpha_i dQ + h\alpha_i P - hd_i P, \sigma_i)$
$= \prod_{i=1}^{n} e(\alpha_i Q + \alpha_i hP, \sigma_i)$
$= e(Q + hP, \sum_{i=1}^{n} \alpha_i \sigma_i)$
$= e((h+x)P, \sum_{i=1}^{n} \alpha_i \sigma_i) = e(P, P)$

Hence we note that, $\mathcal{A}$ returns $\frac{1}{h+x}P = \sum_{i=1}^{n} \alpha_i \sigma_i$. □

The number of operations in $\mathbb{Z}_q^*$ and of the group elements $\mathbb{G}_1$ require for `Setup` and `Output` in the above simulation for Online/offline Signing query is given by:
`Setup`

- Number of multiplication in $\mathbb{Z}_q^*$ is $(n-1)$.
- Number addition of group elements of $\mathbb{G}_1$ is $(n-1)$.
- Number of scalar multiplication of $\mathbb{G}_1$ is $(2n+1)$.
- In signing query, the number of scalar multiplication of $\mathbb{G}_1$ is nq_s.
- Number of inversion is $(n-1)$.

`Output`

- Number of scalar multiplication of $\mathbb{G}_1$ is n.
- Number addition of group elements of $\mathbb{G}_1$ is $(n-1)$.

Let the maximum time requires for scalar multiplication and inverse operation in $\mathbb{Z}_q^*$ are t_{mul} and t_{inv} respectively. Scalar multiplication and addition of group elements of $\mathbb{G}_1$ are t_{smul} and t_{add} respectively. Hence the running time t^* of $\mathcal{A}$ is sum of running time of $\mathcal{F}$ and $(nq_s + 3n + 1)t_{smul} + 2(n-1)t_{add} + (n-1)t_{mul} + (n-1)t_{inv}$. *i*.e

$$t^* \leq (nq_s + 3n + 1)t_{smul} + 2(n-1)t_{add} + (n-1)t_{mul} + (n-1)t_{inv}$$

`Probability Analysis`

$\mathcal{B}$ is successful if the following events holds

1. E_1: $\mathcal{A}$ returns a valid forged signature (m_f, σ_f);
2. E_2 : During the simulation of oracles, the processes is being not aborted or halt.
3. E_3: If $T = 0$ and the event E_2 occurs *i.e* the conditional probability of E_3 given the event E_1 and E_2 occurs.

Probability of successes is

$$P[E_1 \cap E_3] = P[E_1]P[E_2 \mid E_1]P[E_3 \mid E_1 \cap E_2]$$

From the following claims, we can evaluate the lower bound of the probability.

Claim 1. *The probability that $\mathcal{B}$ does abort or halt during the simulation of the oracle is at least $\frac{1}{\epsilon_1}$, where ϵ_1 is a small positive integer.*

Proof. Let us assume without loss of generality, $\mathcal{A}$ does not submit signature query twice for the same messages. By the method of induction, we prove that, the probability $\mathcal{B}$ does not abort is at least $(1 - \frac{1}{q_s-1})^i$ after submitting i number of signature queries.
Let $i = 0$, the claim is trivially true. *i.e* $P[E_1] \geq \frac{1}{\epsilon_1}$. For i^{th} signature query, let the tuple $(m_i, ID, \lambda, T_i) \in L_0$, the bit T_i does not depend on $\mathcal{A}$'s views prior to the submission of signature query. The value that could be submitted to $\mathcal{A}$ that depend on T_i is $\mathcal{H}(m_i)$. But the distribution on $\mathcal{H}(m_i)$ is same whether $T_i = 0$ or $T_i = 1$. Hence the probability that the query causes $\mathcal{B}$ is at most $\frac{1}{q_s+1}$.
Therefore by method of induction hypothesis, the probability that $\mathcal{B}$ does not abort after submitting i times is at least $(1 - \frac{1}{q_s+1}) + \ldots + (1 - \frac{1}{q_s+1})$, ($i$ times) *i.e*

$$P[\neg abort] = (1 - \frac{1}{q_s+1})^i$$

Since $\mathcal{A}$ submits q_s times, the probability that $\mathcal{B}$ does not abort is at least $(1 - \frac{1}{q_s+1})^{q_s}$. Hence

$$(1 - \frac{1}{q_s+1})^{q_s} \geq \frac{1}{\epsilon_1}$$

□

Claim 2. *$\mathcal{A}$ generates a valid forged signature provided $\mathcal{B}$ does not abort during the time of $\mathcal{A}$'s signature query and the probability is at most ϵ_2, where ϵ_2 is a small positive integer. Hence $P[E_2 \mid E_1] \geq \epsilon_2$.*

Proof. □

Claim 3. *The probability of even E_3 occurs and $T = 0$ with $\mathcal{B}$ does not abort during the simulation of signature oracle and $\mathcal{A}$ generates a valid forged signature (m_f, σ_f) is at least $\frac{1}{q_s+1}$.*

Proof. Event E_1 and E_2 occurs simultaneously. $\mathcal{B}$ does not abort and generates a valid forged signature (m_f, σ_f) for which the tuple (m_f, ID, λ, T_i) in L_0 list has $T_i = 1$. When $\mathcal{A}$ returns the output, at that time it knows the value of T_i for those m_i for which it submits the signature query. All the remaining T_is are not associated with $\mathcal{A}$'s views. In fact, $\mathcal{A}$ did not submit a signature query for m_i, then the value of $\mathcal{H}(m_i)$ for these T_i. But the distribution on $\mathcal{H}(m_i)$ is same for both $T_i = 0$ and $T_i = 1$. Since $\mathcal{A}$ could not have submitted a signature query for m_f, we know T depend on $\mathcal{A}$'s correct views. Hence

$$P[T = 0 \mid E_1 \cap E_2] = \frac{1}{q_s+1}$$

□

$$P[E_1 \cap E_3] = P[E_1]P[E_2 \mid E_1]P[E_3 \mid E_1 \cap E_2]$$
$$= (1 - \frac{1}{q_s+1})\epsilon_1(\frac{1}{q_s+1}) \geq \frac{\epsilon_1}{\epsilon_2}(\frac{1}{q_s+1}) \geq \epsilon$$

Running time of $\mathcal{B}$

$\mathcal{B}$'s running time is the sum of the running time of $\mathcal{H}_0$-query, $\mathcal{H}_1$-query and q_s-query.

10 Performance

In this section, we compare the computational cost for each signature generation and verification of our scheme with the schemes proposed by the authors Zhang *e*t al., Lin *e*t al., Awasthi *et al.* and Chow *et al.* in random oracle model. Let denote the following notation for the cost of operations that are performed in these schemes.

- `Mul`$(\mathbb{G}_1)$: Scalar multiplications in $\mathbb{G}_1$
- `Mul`$(\mathbb{G}_2)$: Scalar multiplications in $\mathbb{G}_2$
- `Add`$(\mathbb{G}_1)$: Addition of group elements in $\mathbb{G}_1$
- `Add`$(\mathbb{G}_2)$: Addition of group elements in $\mathbb{G}_2$
- `P` : Bilinear pairings
- `H` : Hashing

Table 1: Comparison of Computational Cost

Scheme	Generation					Verification				
	$\mathtt{P}$	$\mathtt{H}$	$\mathtt{Add}(\mathbb{G}_1)$	$\mathtt{Mul}(\mathbb{G}_1)$	$\mathtt{Mul}(\mathbb{G}_2)$	$\mathtt{P}$	$\mathtt{H}$	$\mathtt{Add}(\mathbb{G}_1)$	$\mathtt{Mul}(\mathbb{G}_1)$	$\mathtt{Mul}(\mathbb{G}_2)$
Zhang *et* al. [6]	$2n-1$	n	n	n	$n-1$	$2n$	n	$-$	n	n
Lin *et* al. [10]	$2n-1$	1	n	$2n-1$	n	2	1	$n-1$	$n+1$	n
Awasthi *et al.* [5]	$2n-1$	n	$n+1$	$2n$	$n-1$	2	$n+1$	$n+1$	$n+1$	1
Chow *et al.* [12]	$-$	$-$	$2n-2$	n	$-$	2	$-$	$2n$	n	$-$
Proposed Scheme	$2n-1$	n	n	n	$n-1$	$2n$	n	$-$	n	$-$

11 Conclusion

In this paper, we have proposed a novel construction of identity based online/off-line signature scheme in random oracle model. Security of the scheme is proven on the assumption of k-CAA and computational Diffie-Hellman problem. Our scheme is more efficient in computational cost and security. Since no heavy computations such as pairing are performed in the online and offline stages. Hence overall computational is reduced in our scheme. This scheme is suited to implement on low processor devices for many application such as whistle blowing, authentication etc.

References

[1] R.Rivest,A.Shamir, and Y.Tauman How to Leak a Secret: Theory and Applications of Ring Signatures. in: Theoretical Computer Science. LNCS, vol.3895, Springer Berlin, pp.164-186, 2006.

[2] Joseph K. Liu, Man Ho Au, Willy Susilo, Jianying Zhou Online/Offline Ring Signature Scheme, Information and Communications Security Lecture Notes in Computer Science Volume 5927, pp 80-90, 2009.

[3] David Chaum and Eugene Van.Heyst Group signatures. In D.W. Davies, editor, Advances in Cryptology Eurocrypt '91, pages 257-265, Berlin, 1991. Springer-Verlag. Lecture Notes in Computer Science No. 547.

[4] Ronald L. Rivest , Adi Shamir, and Yael Tauman How to Leak a Secret ASIACRYPT 2001, LNCS 2248, pp. 552-565, 2001. Springer-Verlag Berlin Heidelberg 2001.

[5] Amit K Awasthi and Sunder Lal ID-based Ring Signature and Proxy Ring Signature Schemes from Bilinear Pairings IACR archieve ePrint report 2004/184, 2004.

[6] F. Zheng and K. Kim Id-based blind signature and ring signature from pairings, Advances in cryptology Asiacrypt 2002 LNCS 2501, pp-533–547, 2002.

[7] F.Zhang, R.Safavi-Naini and Willy Susilo An Efficient Signature Scheme from Bilinear Pairings and Its Applications, In Proceeding of Public Key Cryptography - PKC 2004, Lecture Notes in Computer Science Volume 2947, pp 277-290, 2004.

[8] Alfredo De Santis, Giovanni Di Crescenzo, Giuseppe Persiano, and Moti Yung On monotone formula closure of SZK. In Proc. 35th FOCS, pages 454–465. IEEE, 1994.

[9] Ronald Cramer, Ivan Damgard, and Berry Schoenmakers Proofs of partial knowledge and simplified design of witness hiding protocols. In Yvo Desmedt, editor,Advances in Cryptology - CRYPTO '94, pages 174-187, Berlin, 1994. Springer-Verlag. Lecture Notes in Computer Science Volume 839.

[10] Chih-Yin Lin and Tzong-Chen Wu An Identity-based Ring Signature Scheme from Bilinear Pairings. in: AINA'04, Also appear in http://eprint.iacr.org/2003/117.

[11] Javier Herranz and German Saez New Identity-Based Ring Signature Schemes. in: Information and Communications Security. LNCS, vol.3269, Springer Berlin, pp 269–274, 2004.

[12] Sherman S.M.Chow, S.M.Yiu, and Lucas C.K.Hui Efficient Identity Based Ring Signature. in: Applied Cryptography and Network Security. LNCS, vol.3531, Springer Berlin,2005, pp.499-512.

[13] M.Abe, M.Ohkubo, K.Suzuki 1-out-of-n Signatures from a Variety of Keys. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 415-432. Springer, Heidelberg (2002)

[14] D.Boneh, C.Gentry, B.Lynn and H.Shacham Aggregate and Verifiably Encrypted Signatures from Bilinear Maps. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 416-432. Springer, Heidelberg (2003).

[15] E.Bresson, J.Stern and M.Szydlo Threshold Ring Signatures and Applications to Ad-hoc Groups. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 465-480, Springer, Heidelberg (2002).

[16] Y.Dodis, A.Kiayias, A.Nicolosi and V.Shoup Anonymous Identification in Ad Hoc Groups, in proceeding of EUROCRYPT 2004. LNCS, vol. 3027, pp. 609-626. Springer, Heidelberg (2004).

[17] J. Herranz and G. Saez Ring Signature Schemes for General Ad-Hoc Access Structures. In Proceedings of ESAS 2004, pages 54-65.

[18] J. Herranz and G. Saez New Identity-Based Ring Signature Schemes. In Information and Communications Security (ICICS 2004), pages 27-39. Springer-Verlag, 2004.

[19] J. Herranz and G. Saez Distributed Ring Signatures for Identity-Based Scenarios, In Cryptology ePrint Archive: Report 2004/190.

[20] G. H. Hardy and E. M. Wright An Introduction to the Theory of Numbers, Oxford, fifth edition, 1979.

[21] M. Jakobsson, K. Sako, and R. Impagliazzo Designated verifier proofs and their applications. In Ueli Maurer, editor, Advances in Cryptology - EuroCrypt '96, pages 143-154, Berlin, 1996. Springer-Verlag. Lecture Notes in Computer Science, Volume 1070.

[22] Hidenori Kuwakado, Hatsukazu Tanaka Threshold Ring Signature Scheme Based on the Curve. In IPSJ JOURNAL Abstract Vol.44, pages 8-32.

[23] Zavier Herrenz and German Sacz Forking Lemma in Ring Signature Scenarios, IACR archive ePrint report no-2003/067, 2003.

[24] J.Kar Provably Secure Online/Off-line Identity-Based Signature Scheme for Wireless Sensor Network, IACR Archive ePrint report no-2012/162, 2012.

[25] H. Shacham and B.Waters Efficient ring signatures without random oracles. In: Okamoto, T., Wang, X. (eds.) PKC 2007. LNCS, vol. 4450, pp. 166-180. Springer, Heidelberg (2007).

[26] J.K. Liu and D.S. Wong Enhanced security models and a generic construction approach for link able ring signature. Int. J. Found. Comput. Sci. 17(6), 1403-1422 (2006).

[27] J.Kar A Novel Construction of Aggregate Signcryption Scheme for Smart Card Proceedings of the IEEE 2013 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, pp 6-13, 2013.

[28] M.Joye An efficient on-line/off-line signature scheme without random oracles. In: Franklin, M.K., Hui, L.C.K., Wong, D.S. (eds.) CANS 2008. LNCS, vol. 5339, pp. 98-107. Springer, Heidelberg (2008).

[29] D.Boneh and X. Boyen Short signatures without random oracles the SDH assumption in bilinear groups. Journal of Cryptology 2, 149-177 (2008).

[30] M.H. Au, J.K.Liu, W. Susilo and T.H. Yuen Constant-size ID-based linkable and revocable-iff-linked ring signature. In: Barua, R., Lange, T. (eds.) INDOCRYPT 2006. LNCS, vol. 4329, pp. 364-378. Springer, Heidelberg (2006).