2008

A generic construction of identity-based online/offline signcryption

Dongdong Sun
University of Wollongong

Yi Mu
University of Wollongong, ymu@uow.edu.au

Willy Susilo
University of Wollongong, wsusilo@uow.edu.au

A generic construction of identity-based online/offline signcryption

Abstract

Signcryption has clear advantage over traditional sign-then-encrypt schemes. However, the computational overhead for signcryption is still too heavy when it is applied to resource-constraint systems. In this paper, we propose a generic construction of the identity-based online/offline signcryption, where most of computations are carried out when the associated message is still unavailable and the online part of our scheme does not require any exponent computations and therefore is very efficient. Our scheme isgeneric and identity-based, in the sense it is independent of the selection of signature and encryption algorithms. Our scheme possesses the properties of ciphertext indistinguishability (IND-gCCA2) and existentially unforgeability (UF-CMA).

Disciplines

Physical Sciences and Mathematics

Publication Details

Sun, D., Mu, Y. & Susilo, W. (2008). A generic construction of identity-based online/offline signcryption. IEEE International Symposium on Parallel and Distributed Processing with Applications (pp. 707-712). Los Alamitos, CA: IEEE.

2008 International Symposium on Parallel and Distributed Processing with Applications

A Generic Construction of Identity-Based Online/Offline Signcryption

Dongdong Sun, Yi Mu, Willy Susilo
School of Computer Science and Software Engineering
University of Wollognong
Wollongong, NSW 2522, Australia
{dds03, ymu, wsusilo}@uow.edu.au

Abstract

Signcryption has clear advantage over traditional sign-then-encrypt schemes. However, the computational overhead for signcryption is still too heavy when it is applied to resource-constraint systems. In this paper, we propose a generic construction of the identity-based online/offline signcryption, where most of computations are carried out when the associated message is still unavailable and the online part of our scheme does not require any exponent computations and therefore is very efficient. Our scheme is generic and identity-based, in the sense it is independent of the selection of signature and encryption algorithms. Our scheme possesses the properties of ciphertext indistinguishability (IND-gCCA2) and existentially unforgeability (UF-CMA).

1. Introduction

Signcryption is a cryptographic primitive introduced by Zheng [13] in 1997. The main idea of this primitive is to carry out encryption and signature computations in a single logical step to obtain confidentiality and authentication more efficiently than the simple composition of encryption and signature. Since Zheng's seminal work, many efficient signcryption schemes have been proposed.

The notion of identity-based cryptography was introduced by Shamir in 1984 [10]. The system is realized by introducing a trusted third party named Private Key Generator (PKG) to produce a user's private key corresponding to the user's identity. Shamir proposed an identity-based signature scheme, but for many years identity-based encryption remained an open problem. In 2001, Boneh and Franklin [3] proposed a fully practical and functional identity-based encryption scheme with security proof.

Identity-based notion was introduced to signcryption in 2002. Malone-Lee proposed an identity-based signcryption solution [9]. However, it is not semantically secure. Libert and Quisquater [8] proposed a solution to remedy the problem. Unfortunately, the properties of public verifiability and forward security are mutually exclusive in their scheme. Boyen [4] proposed a Multipurpose Identity-Based Signcryption and gave the security notions for signcryption as: message confidentiality, signature non-repudiation, ciphertext unlinkability, ciphertext authentication, and ciphertext anonymity. Chen and Malone-Lee proposed a more efficient scheme [5] and their scheme provides a full security analysis in the model of [4].

To extend the applicability of signcryption to low-power devices, online/offline signcryption was introduced by An, Dodis, and Rabin [1]. The online/offline notion can be tracked to the earlier work due to online/offline signatures. Even, Goldreich, and Micali [7] proposed the first online/offline signature which is a generic scheme to convert any signature scheme into an online/offline counterpart. Their scheme increases the size of each signature by a quadratic factor, hence, it only makes sense in theoretical aspect. Another generic method to achieve online/offline signing was proposed by Shamir and Tauman [11] in 2001. The main advantage of the latter is that the length of the key and signature are significantly reduced which is much better than the former in practical sense. After that, a much more efficient generic online/offline signature scheme was proposed by Chen et al [6].

The work of online/offline signcryption due to An, Dodis, and Rabin [1] mainly concentrates on the security analysis of general combination of signature and encryption scheme in asymmetric settings. No concrete scheme was provided in [1]. Zhang, Mu, and Susilo [12] proposed the first concrete online/offline signcryption scheme in 2005. In their scheme, the online part does not require any expensive computations so it is very efficient, and moreover, the size of a signature is short since they use the notion of short signature.

Motivation and Contribution

Identity-based online/offline signcryption has potential applicability to low-power devices. The reason is threefold.

978-0-7695-3471-8/08 $25.00 © 2008 IEEE
DOI 10.1109/ISPA.2008.16

(1) Identity-based system avoids distribution of public keys. (2) It allows expensive computation to be carried out in an offline phase. (3) Signcryption achieves encryption and signature in a single logical step to obtain confidentiality and authentication more efficiently than the sign-then-encrypt approach. In this paper, we propose a generic scheme of identity-based online/offline signcryption, where the signing part and encryption part are generic.

Our contributions of this paper are as follows. We first formally define the generic identity-based online/offline signcryption and related security models. We then propose a generic construction of identity-based online/offline signcryption. Our construction is based on chameleon hash function and pairing over elliptic curves. It can achieve authenticity and confidentiality simultaneously in an efficient manner. We also provide a proof that the resultant scheme is indistinguishable against adaptive chosen-ciphertext attacks (IND-gCCA2) and is existentially unforgeable against adaptive chosen-message attacks (UF-CMA). Finally, we present a new generic online/offline broadcast signcryption as an extension.

The rest of this paper is organized as follows. In Section 2, we briefly review the preliminaries required in this paper. In Section 3, we formally define the generic identity-based online/offline signcryption. We present our scheme and prove its security in our model in Section 4 and Section 5. In Section 6, we describe an extension of our scheme to broadcast signcryption. We conclude this paper in Section 7.

2. Preliminaries

2.1. Bilinear Mapping

Let k be a security parameter and q be a k-bit prime number. Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be groups of the same prime order q. There is a bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ with the following properties:

1. Bilinearity: for all $P, Q \in \mathbb{G}_1$ and $a, b \in Z_q^*$, $e(aP, bQ) = e(P, Q)^{ab}$.
2. Non-degeneracy: for any generator $P \in \mathbb{G}_1$, $e(P, P) \neq 1$.
3. Computability: there is an efficient algorithm to compute $e(P, Q)$, for $P, Q \in \mathbb{G}_1$.

2.2. Security Assumptions

The security of our scheme is based on the intractability of the following problem.

Definition 1. Gap-Bilinear Diffie-Hellman Problem (GBDH) *Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two groups of the same order q. Let P be a generator of $\mathbb{G}_1$. Assume that there is a bilinear mapping $e : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$. Let an attacker $\mathcal{B}$ to solve the following problem: Given (P, aP, bP, cP), compute a Bilinear Diffie-Hellman key $e(P, P)^{abc}$ with the help of the Decisional Bilinear Diffie-Hellman (DBDH) oracle, which given (P, aP, bP, cP, d), outputs $ture$ if $d = e(P, P)^{abc}$ and $false$ otherwise.*

We define $\mathcal{B}$'s advantage $Adv_{\mathbb{G}_1}^{GBDH}(\mathcal{B}) = \Pr[\mathcal{B}(P, aP, bP, cP) = e(P, P)^{abc}]$. We say an algorithm $\mathcal{B}$ (t, q_b, ϵ) breaks GBDH in $(\mathbb{G}_1, \mathbb{G}_2)$ if it makes q_b queries in time t, $\mathcal{B}$ has advantage greater than ϵ in solving GBDH.

2.3. Chameleon Hash Family

Definition 2. *[11] (chameleon hash family) A chameleon hash family consists of a pair $(\mathcal{L}, \mathcal{H})$:*

- *Assume $\mathcal{L}$ is a probabilistic polynomial-time key generation algorithm that on input 1^k, outputs a pair (HK, TK) such that the sizes of HK, TK are polynomially related to k.*
- *Assume $\mathcal{H}$ is a family of randomized hash functions. Every hash function in $\mathcal{H}$ is associated with a hash key HK, and is applied to a message from a space $\mathcal{M}$ and a random element from a finite space $\mathcal{R}$. The output of hash function H_{HK} does not depend on TK.*

A chameleon hash family $(\mathcal{L}, \mathcal{H})$ has the following properties:

1. **Efficiency:** Given a hash key HK and a pair $(m, r) \in \mathcal{M} \times \mathcal{R}$, $H_{HK}(m, r)$ can be computed in polynomial time.
2. **Collision resistance:** There is no probabilistic polynomial time algorithm $\mathcal{A}$ that on input HK outputs, with a non-negligible probability, two pairs $(m_1, r_1), (m_2, r_2) \in \mathcal{M} \times \mathcal{R}$ that satisfy $H_{HK}(m_1, r_1) = H_{HK}(m_2, r_2)$ and $m_1 \neq m_2$.
3. **Trapdoor collisions:** There is a probabilistic polynomial time algorithm that given a pair $(HK, TK) \leftarrow \mathcal{L}(1^k)$, a pair $(m_1, r_1) \in \mathcal{M} \times \mathcal{R}$, and an additional message $m_2 \in \mathcal{M}$, outputs a value $r_2 \in \mathcal{R}$ such that:
   - $H_{HK}(m_1, r_1) = H_{HK}(m_2, r_2)$.
   - If r_1 is uniformly distributed in $\mathcal{R}$ then the distribution of r_2 is computationally indistinguishable from uniform in $\mathcal{R}$.

We now present a construction of chameleon hash family [11] with the elliptic curve analogue. The chameleon hash function is based on discrete logarithm assumption.

- **System Parameters Generation Algorithm $\mathcal{L}$:** Let t be a prime power, and $E(\mathbb{F}_t)$ an elliptic curve over finite field $\mathbb{F}_t$. Let $\#E(\mathbb{F}_t)$ be the number of points of $E(\mathbb{F}_t)$, and P be a point of $E(\mathbb{F}_t)$ with prime order q where $q|\#E(\mathbb{F}_t)$. Denote $\mathbb{G}$ the subgroup generated by P. Choose a random element $x \in_R Z_q^*$, and compute $Y = xP$. The public hash key is $HK = (P, Y)$, and the private trapdoor key is $TK = x$.
- **The Hash Family $\mathcal{H}$:** Given the hash key HK, the proposed chameleon hash function $H_{HK} : Z_q \times Z_q \rightarrow \mathbb{G}$ is defined as follows: $H_{HK}(m, r) \stackrel{def}{=} mP + rY$.

3. Definition and Security Models of Generic Identity-based Online/Offline Signcryption

3.1. Definition of Generic Identity-based Online/Offline Signcryption

Definition 3. *The generic identity-based online/offline signcryption scheme is comprised of five algorithms:* **System Parameters Generation**, **Key Generation**, **OffSigncrypt**, **OnSigncrypt** *and* **UnSigncrypt**.

1. **System Parameters Generation**. Given a security parameter k as input, the private key generator PKG generates the system's public parameters $params$, the master secret key s, a chameleon hash family $(\mathcal{L}, \mathcal{H})$ and an identity-based signature scheme $(\mathcal{G}, \mathcal{S}, \mathcal{V})$, where $params$ and $(\mathcal{G}, \mathcal{S}, \mathcal{V})$ are published in the system, s is kept as secret by PKG and the chameleon hash family $(\mathcal{L}, \mathcal{H})$ is sent to the designated user.
2. **Key Generation**.
   - Given an identity ID and the master secret key s as input, output d_{ID}.
   - On input 1^k, run the key generation algorithm of the trapdoor hash family $(\mathcal{L}, \mathcal{H})$ to obtain the hash/trapdoor key pair (HK, TK).
3. **OffSigncrypt**. Given $params$, ID_S's private key d_{ID_S}, hash key HK_S and the receiver's identity ID_R as input, this algorithm outputs an offline signature σ'.
4. **OnSigncrypt**. Given a message m, receiver's identity ID_R, hash/trapdoor key pair (HK_S, TK_S) and an offline signature σ' as input, this algorithm outputs the ciphertext C.
5. **UnSigncrypt**. Given $params$, a ciphertext C, the sender's identity ID_S and the receiver's private key d_{ID_R} as input, this algorithm outputs the plaintext m or the symbol "$\perp$". "$\perp$" denotes that C is an invalid ciphertext between ID_S and ID_R.

Correctness. The algorithm **UnSigncrypt** will output a plaintext if the ciphertext and the offline signature are generated as defined above.

$m \leftarrow$ **UnSigncrypt**$(params,$ **OnSigncrypt**$(params, m, ID_R, HK_S, TK_S,$ **OffSign**$(params, ID_R, d_{ID_S}, HK_S)), ID_S, ID_R, d_{ID_R})$

3.2. Security Models of Generic Identity-based Online/Offline Signcryption

An, Dodis, and Rabin [1] generalized IND-CCA2 notion slightly, by introducing an equivalence relation $\mathcal{R}$ with property: $\mathcal{R}(c_1, c_2) = true \Rightarrow Dec(c_1) = Dec(c_2)$ (c_1 and c_2 are ciphertexts). $\mathcal{R}$ is called decryption-respecting. We may use it to restrict the attacker from decrypting other encryptions of the target message. We say that the encryption scheme $\mathcal{E}$ is ciphertext indistinguishable against generalized CCA2 (or gCCA2) if there exists some efficient decryption-respecting relation $\mathcal{R}$ with respect to which it is CCA2-secure.

Our scheme is based on the $\mathcal{C}t\mathcal{E}\&\mathcal{S}$ which is called "commit-then-encrypt-and-sign" paradigm [1]. Before presenting our scheme, we revisit some theorems and issues addressed in [1] as follows:

Theorem 1. *[1] Assume that $\mathcal{E}$ is IND-gCCA2-secure, $\mathcal{S}$ is UF-CMA-secure and $\mathcal{C}$ satisfies the syntactic properties of a commitment scheme. Then, in the insider-security model, we have:*

- *$\mathcal{C}t\mathcal{E}\&\mathcal{S}$ is IND-gCCA2-secure $\Longleftrightarrow$ $\mathcal{C}$ satisfies the hiding property.*
- *$\mathcal{C}t\mathcal{E}\&\mathcal{S}$ is UF-CMA secure $\Longleftrightarrow$ $\mathcal{C}$ satisfies the relaxed binding property.*

Thus, $\mathcal{C}t\mathcal{E}\&\mathcal{S}$ preserves security of $\mathcal{E}$ and $\mathcal{S}$ iff $\mathcal{C}$ is a secure relaxed commitment. In particular, any secure regular commitment $\mathcal{C}$ yields secure signcryption $\mathcal{C}t\mathcal{E}\&\mathcal{S}$.

The chameleon hash function can be regarded as a commitment. It should be noted that since chameleon hash functions $\mathcal{C}$ are information-theoretically hiding, it is safe for the receiver when the sender chooses a bad commitment key (the hiding property is satisfied for all HK's, and it is in sender's interest to choose HK so that the binding is satisfied as well). It is easy to determine that our proposed chameleon hash function satisfies both properties.

We can find from the next Section that our scheme is similar to $\mathcal{C}t\mathcal{E}\&\mathcal{S}$ except that we move the expensive signature part to offline phase. We also modified the encryption part to be more suitable in identity-based system. Hence, if our encryption part is IND-gCCA2-secure and we choose some UF-CMA secure identity-based signature

scheme combined with the chameleon hash function, we can construct an IND-gCCA2 secure and UF-CMA secure identity-based online/offline signcryption scheme. A proof for the encryption part can be found in Section 5.

4. Our Generic Identity-Based Online/Offline Signcryption Scheme

System Parameters Generation: Let t be a prime power, and $E(\mathbb{F}_t)$ an elliptic curve over finite field $\mathbb{F}_t$. Let $\#E(\mathbb{F}_t)$ be the number of points of $E(\mathbb{F}_t)$, and P be a point of $E(\mathbb{F}_t)$ with prime order q where $q|\#E(\mathbb{F}_t)$. $\mathbb{G}_1$ is the subgroup generated by P. $\mathbb{G}_2$ is a finite group of order q. Choose cryptographic hash function $H_1 : \mathbb{G}_2 \rightarrow \{0,1\}^n$. Let $(\mathcal{L}, \mathcal{H})$ be the chameleon hash family, which will be sent to the designated user on request, based on the discrete logarithm assumption and $(\mathcal{G}, \mathcal{S}, \mathcal{V})$ be any identity-based signature scheme. The system parameters are $SP = \{E(\mathbb{F}_t), t, q, P, \mathbb{G}_1, \mathbb{G}_2, (\mathcal{G}, \mathcal{S}, \mathcal{V}), H_1\}$.

Key Generation:

- Given an identity ID, run the key extract algorithm of the original identity-based signature scheme to obtain the private/public key pair (d_{ID}, Q_{ID}).
- On input 1^k, the sender runs the key generation algorithm of the trapdoor hash family $(\mathcal{L}, \mathcal{H})$ to obtain the hash/trapdoor key pair $(Y = xP, x)$.

Assume Alice sends m to Bob. Alice obtains private key and hash/trapdoor key $\{d_{ID_A}, Y, x\}$. Bob obtains private key d_{ID_B}. $\{Q_{ID_A}, Q_{ID_B}\}$ are public to both of them.

OffSigncrypt:

- Choose at random $(m', r') \in_R \mathcal{M} \times \mathcal{R}$, where $\mathcal{M}$ is a message space and $\mathcal{R}$ is a finite space, and compute the chameleon hash value $h = H_Y(m', r') = m'P + r'Y$.
- Run the signing algorithm $\mathcal{S}$ with the signing key d_{ID_A} to sign the hash value h. Let the output be $\sigma = \mathcal{S}_{d_{ID_A}}(h||H_Y)$, where H_Y is the description of the chameleon hash.
- Choose at random $y \in_R Z_q^*$ and compute $X = yP$, then compute $\omega = e(yP_{pub}, Q_{ID_B})$. Finally set $y' = H_1(\omega)$.
- Store the pair (m', r') and y' for future use.

OnSigncrypt:

- For a given message m, retrieve from the memory x^{-1} and the pair (m', r').
- Compute $r = x^{-1}(m' - m) + r' \mod q$.
- The message encryption is done with y' and a symmetric-key encryption algorithm such as AES. The ciphertext is $c = Enc_{y'}(\sigma||ID_A||m||r||H_Y)$.
- Final ciphertext is (c, X).

UnSigncrypt:

- Given ciphertext (c, X), compute $\omega = e(X, d_{ID_B})$ and $y' = H_1(\omega)$.
- Decrypt c as $\sigma||ID_A||m||r||H_Y = Dec_{y'}(c)$.
- Compute $h = H_Y(m, r) = mP + rY$.
- Verify that σ is indeed a signature of the value $h||H_Y$ with respect to the verification key Q_{ID_A}.

Correctness: The consistency is easy to verify as follows: $e(yP_{pub}, Q_{ID_B}) = e(yP, d_{ID_B}) = e(X, d_{ID_B})$.

Performance: The proposed scheme satisfies the requirement of online/offline signcryption as all expensive computations are done in the offline phase. The offline phase of our signcryption mainly consists of one evaluation of the trapdoor hash function, one invocation of the original signing algorithm and one pairing computation. The online phase consists of only a single collision finding computation and a symmetric-key encryption. The UnSigncrypt algorithm consists of one evaluation of the trapdoor hash function, one invocation of the original verification algorithm, one pairing computation and a symmetric-key decryption.

5. Security Proof

We can choose most of UF-CMA secure identity-based signatures as long as the key extraction algorithm is the same as in the encryption scheme below. We prove the encryption scheme is IND-gCCA2 secure to complete our proof.

Setup: Given security parameters k, n and $\mathbb{G}_1, \mathbb{G}_2$ of order q and generator P of $\mathbb{G}_1$, pick a random $s \in Z_q^*$, and set $P_{pub} = sP$. Choose cryptographic hash functions $H_0 : \{0,1\}^* \rightarrow \mathbb{G}_1$, $H_1 : \mathbb{G}_2 \rightarrow \{0,1\}^n$. $\mathcal{R}$ is decryption-respecting mentioned before. The system parameters are (P, P_{pub}, H_0, H_1). The master key is s. H_0 and H_1 will be regarded as random oracles in security analysis.

Extract: Given an identity ID, compute $d_{ID} = sH_0(ID)$ and output it as the private key related to ID corresponding to $Q_{ID} = H_0(ID)$.

Encrypt: Given a message m, choose at random $y \in_R Z_q^*$ and compute $X = yP$, then compute $\omega = (yP_{pub}, Q_{ID_B})$,

finally set $y' = H_1(\omega)$. The message encryption is done with y' and a symmetric-key encryption algorithm such as AES. The ciphertext is (c, X), where $c = Enc_{y'}(m)$.

Decrypt: Given a ciphertext (c, X), Compute $\omega = e(X, d_{ID_B})$ and Set $y' = H_1(\omega)$, then decrypt the message $Dec_{y'}(c) = m$.

Theorem 2. *In the random oracle model, assume we have an IND-gCCA2 adversary called $\mathcal{A}$ that is able to distinguish ciphertexts that succeeds with probability ϵ and asking H_0, H_1 and decryption oracle q_0, q_1 and q_d times respectively. Then, there exists a simulator $\mathcal{B}$ that can solve the GBDH problem with the probability at least $\epsilon \cdot \frac{1}{q_0} \cdot \frac{1}{q_1} \cdot \left(1 - \frac{1}{q_0}\right)$.*

Proof. Let (P, aP, bP, cP) be the instance of the GBDH problem to be solved, the aim is to compute $e(P, P)^{abc}$ where a, b, c are chosen at random from Z_q^* and P generates $\mathbb{G}_1$. $\mathcal{B}$ will run $\mathcal{A}$ as a subroutine and act as $\mathcal{A}$'s challenger in the IND-gCCA2 game. $\mathcal{B}$ needs to maintain lists L_0 and L_1 that are initially empty and are used to keep track of answers to queries asked by $\mathcal{A}$ to oracles H_0, H_1. $\mathcal{B}$ gives $\mathcal{A}$ the system parameters with $P_{pub} = bP$.

We describe how the requests are treated below.

H_0 **requests**: At the beginning of the simulation, choose i_β uniformly at random from $\{1, \ldots, q_0\}$. If $i = i_\beta$ then respond with $H_0(ID_U) = aP$ and set $ID_\beta = ID_U$, else choose x uniformly at random from Z_q^*; compute $Q_U = xP$; compute $d_U = xP_{pub}$; store (ID_U, Q_U, d_U, x) in L_0 and respond with Q_U.

H_1 **requests**: for a query $H_1(\omega)$, $\mathcal{B}$ first ensures the list L_1 does not contain a tuple (ω, y'). If such a tuple is found, $\mathcal{B}$ answers y', otherwise he chooses $y' \in_R Z_q^*$, gives it as an answer to the query and puts the tuple (ω, y') into L_1.

Key extraction requests : We assume that $\mathcal{A}$ makes the query $H_0(ID_U)$ before it makes the extraction query for ID_U. When $\mathcal{A}$ asks a query **Extract**(ID_U), if $ID_U = ID_\beta$, then abort the simulation, otherwise $\mathcal{B}$ searches L_0 for the entry (ID_U, Q_U, d_U, x) corresponding to ID_U and returns d_U.

Decryption requests: When receiving an decryption query for a ciphertext (c, X) for identities ID_U that are not ID_β, find the entry (ID_U, Q_U, d_U, x) in L_0 and compute $\omega = (X, d_U)$, then run the H_1 simulation algorithm to find $y' = H_1(\omega)$, finally decrypt the ciphertext $Dec_{y'}(c) = m$.

When receiving an decryption query for a ciphertext (c, X) for identities $ID_U = ID_\beta$, $\mathcal{B}$ steps through the list L_1 with entries (ω, y'). For each pair in L_1, $\mathcal{B}$ submits the tuple $(P, H_0(ID_\beta), P_{pub}, X, \omega)$ to DBDH oracle. The DBDH oracle returns 1 if $\omega = e(X, d_{ID_\beta})$ and 0 otherwise. If the returned value is 1, $\mathcal{B}$ will use the corresponding y' to decrypt the ciphertext $Dec_{y'}(c) = m$. Otherwise $\mathcal{B}$ takes a random pair (ω, y') such that no $(\omega, .)$ already exists in L_1, then decrypt the message $Dec_{y'}(c) = m$ and put the tuple (ω, y') into L_1.

After a polynomially bounded number of queries, $\mathcal{A}$ chooses an identity ID_B on which he wishes to be challenged and produces his two plaintexts m_0 and m_1. The restriction is that $\mathcal{A}$ cannot have chosen ID_B as one of key extraction requests. If $ID_B \neq ID_\beta$, $\mathcal{B}$ aborts the simulation. Otherwise $\mathcal{B}$ chooses $c^* \in_R \{0,1\}^*$ and sets $X^* = cP$. It returns the challenge ciphertext (c^*, X^*) to $\mathcal{A}$. $\mathcal{A}$ then performs a second series of queries which is treated in the same way as the first one. This time, he can not make a key extraction request on ID_B and he can not make an decrypt query of (c', X') equivalent to (c^*, X^*), i. e. $\mathcal{R}((c', X'), (c^*, X^*)) = true$.

At the end of the simulation, $\mathcal{A}$ produces a bit b. The simulator ignores this bit. It chooses some ω at random from L_1 and returns ω as its guess at the solution to the GBDH problem for (P, aP, bP, cP).

Let us now consider how our simulation could fail, i.e. describe events that could cause $\mathcal{A}$'s view to differ when run by $\mathcal{B}$ from its view in a real attack. It is clear that the simulations for H_0, H_1 and Decryption oracle are indistinguishable from real random oracles. $\mathcal{B}$ will abort the key extraction oracle, if d_{ID_β} was asked. The probability for the oracle to abort is at most $1/q_0$. With a probability exactly $1/q_0$, $\mathcal{A}$ chooses to be challenged on ID_β. If $\mathcal{A}$ queries the H_1 oracle for $\omega = e(P, P)^{abc}$, the simulation would fail. However, if $\mathcal{A}$ has any advantage it must make this query, and once it has done so we have trapped it into leaving enough information in L_1 to solve the GBDH problem with probability $1/q_1$.

We conclude from the above that $\mathcal{B}$ succeeds with probability as follows:

$$Adv[\mathcal{B}] > \epsilon \cdot \frac{1}{q_0} \cdot \frac{1}{q_1} \cdot \left(1 - \frac{1}{q_0}\right).$$

□

6. Application to Broadcast Signcryption

We now present a generic identity-based online/offline broadcast signcryption scheme based on the work in [2].

System Parameters Generation, Key Generation: As in our proposed scheme.

Assume user 1 broadcasts message m to a group of N users.

Initialization: Let Q_{ID_1} be the public key of the user 1, and $K_{1N} \in_R Z_q^*$ be the broadcast secret of user 1 for a group of N users. User 1 will compute the broadcast parameter $P_{1-brdcst}$ as: $P_{1-brdcst} = K_{1N} Q_{ID_1}$. User 1 will

deliver the parameter $P_{1-brdcst}$ to other users in the group by encrypting in each group-member's pairwise shared key ($k = e(d_{ID_1}, Q_{ID_i})\ i \in N$) with user 1.

OffSigncrypt:

- Choose at random $(m', r') \in_R \mathcal{M} \times \mathcal{R}$, where $\mathcal{M}$ is a message space and $\mathcal{R}$ is a finite space, and compute the chameleon hash value $h = H_Y(m', r') = m'P + r'Y$.
- Run the signing algorithm $\mathcal{S}$ with the signing key d_{ID_1} to sign the hash value h. Let the output be $\sigma = \mathcal{S}_{d_{ID_1}}(h||H_Y)$, where H_Y is the description of the chameleon hash.
- Choose at random $y \in_R Z_q^*$ and compute $\omega = e(Q_{ID_1}, P)^y$, then compute $X = yK_{1N}^{-1}P$. Finally Set $y' = H_1(\omega)$.
- Store the pair (m', r') and y' for future use.

OnSigncrypt:

- For a given message m, retrieve from the memory x^{-1} and the pair (m', r').
- Compute $r = x^{-1}(m' - m) + r' \mod q$.
- The message encryption is done with y' and a symmetric-key encryption algorithm such as AES. The ciphertext is $c = Enc_{y'}(\sigma||m||r||H_Y)$.
- Final ciphertext is (c, X).

UnSigncrypt:

Given a ciphertext (c, X), the authorized receivers (i.e., members of the group provided with broadcast parameter $P_{1-brdcst} = K_{1N}Q_{ID_1}$) will compute the key y'.

- Given ciphertext (c, X), compute $\omega = e(P_{1-brdcst}, X)$ and $y' = H_1(\omega)$.
- Decrypt c as $\sigma||m||r||H_Y = Dec_{y'}(c)$.
- Compute $h = H_Y(m, r) = mP + rY$.
- Verify that σ is indeed a signature of the hash value $h||H_Y$ with respect to the verification key Q_{ID_1}.

7. Conclusion

In this paper, we have proposed a generic identity-based online/offline signcryption scheme. In our scheme, the online computation is very efficient and all the expensive computation is performed offline. Our scheme is generic and does not require specific identity-based signatures and symmetric-key encryption schemes. Our scheme is secure against adaptive chosen-ciphertext attacks (IND-gCCA2) and is existentially unforgeable against adaptive chosen-message attacks (UF-CMA) provided that the signature part is UF-CMA secure. We also present a broadcast signcryption scheme as an application of the scheme.

References

[1] J. An, Y. Dodis, and T. Rabin. On the security of joint signature and encryption. In *Eurocrypt 02*, volume 2332 of *Lecture Notes in Computer Science*, pages 83–107. Springer-Verlag, 2002.

[2] M. Bohio and A. Miri. An authenticated broadcasting scheme for wireless ad hoc network. In *Proceedings of the Second Annual Conference on Communication Networks and Services Research*, pages 69–74, 2004.

[3] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. In *Crypto 01*, volume 2139 of *LNCS*. Springer, 2001.

[4] X. Boyen. Multipurpose identity-based signcryption: A swiss army knife for identity-based cryptography. In *Crypto 03*, volume 2729 of *Lecture Notes in Computer Science*, pages 383–399. Springer-Verlag, 2003.

[5] L. Chen and J. Malone-Lee. Improved identity-based signcryption. Cryptology ePrint Archive, Report 2004/114, 2004. http://eprint.iacr.org.

[6] X. Chen, F. Zhang, W. Susilo, and Y. Mu. Efficient generic on-line/off-line signatures without key exposure. In *ACNS 07*, volume 4521 of *Lecture Notes in Computer Science*, pages 18–30, Zhuhai, China, 2007. Springer-Verlag.

[7] S. Even, O. Goldreich, and S. Macali. On-line/off-line digital signatures. In *Crypto 89*. Springer, 1990.

[8] B. Libert and J.-J. Quisquater. New identity-based signcryption schemes based on pairings. In *IEEE Information Theory Workshop*, Paris, France, 2003.

[9] J. Malone-Lee. Identity-based signcryption. Cryptology ePrint Archive, Report 2002/098, 2002. http://eprint.iacr.org.

[10] A. Shamir. Identity-based cryptosystems and signature schemes. In *Crypto 84*, volume 0196 of *Lecture Notes in Computer Science*, pages 47–53. Springer-Verlag, 1984.

[11] A. Shamir and Y. Tauman. Improved online/offline signature schemes. In *Crypto 01*, volume 2139 of *Lecture Notes in Computer Science*, pages 355–367. Springer-Verlag, 2001.

[12] F. Zhang, Y. Mu, and W. Susilo. Reducing security overhead for mobile networks. In *AINA 05*, volume 1, pages 398–403, 2005.

[13] Y. Zheng. Digital signcryption or how to achieve cost(signature & encryption) $\ll$ cost(signature) + cost(encryption). In *Crypto 97*, volume 1294 of *Lecture Notes in Computer Science*, pages 165–179. Springer-Verlag, 1997.