Chapter 9
Chapter 9

An Efficient and Public Verifiable Identity Based Multi-Signcryption

We have discussed identity based multi-signcryption (IBMSC) schemes in Section 3.2 of Chapter 3. In this Chapter, we propose an IBMSC scheme which is more efficient than the Selvi et al.'s [SVR09] and Zhang et al.'s [ZYN09] schemes. Our scheme needs no pairing computation on the signcryption stage. The scheme also possesses public verifiability for signature. We also convert our proposed scheme for multiple receivers, which we believe is the first multi-signcryption scheme for multiple receivers.

This chapter is organized as follows: We give the proposed IBMSC scheme in section 9.1 and discuss its heuristic security in section 9.2. In section 9.3, we analyze the proposed IBMSC scheme and compare it with existing schemes. We introduce a new variant ‘identity based multi signcryption scheme for multiple receivers’ in section 9.4 and propose a scheme for the same in section 9.5. We discuss its heuristic security in section 9.6.

9.1 IBMSC1: THE PROPOSED IBMSC SCHEME

Setup: Given a security parameter $k$, the PKG chooses the system parameters that includes two groups $G_1$, $G_2$ of same prime order $p$, a bilinear map $e : G_1 \times G_1 \rightarrow G_2$ and a generator $P \in G_1$. PKG randomly chooses $s \in \mathbb{Z}_p$ and $R \in G_1 (R \neq P, \mathcal{O})$. PKG computes $P_{pub} = sP \in G_1$ and $\theta = e(P_{pub}, R)$. The PKG also chooses cryptographic hash functions $H_0 : \{0,1\}^* \rightarrow G_1$, $H_1 : G_2 \rightarrow \{0,1\}^n$, $H_2 : \{0,1\}^n \times G_1 \times G_1 \rightarrow \mathbb{Z}_p^*$ where $n$ is the length of plaintext and ciphertext.

The system public parameters are

$$ \text{params} = \langle p, G_1, G_2, e, n, P, P_{pub}, \theta, R, H_0, H_1, H_2 \rangle $$

Extract: Given a user identity $ID_U \in \{0,1\}^*$, PKG computes public key $Q_U = H_0(ID_U)$ and private key $D_U = sQ_U$. 
**Signcrypt:** To signcrypt a message $m \in \{0,1\}^n$ for the receiver with identity $ID_B$, each sender $ID_t \in \{ID_1, ..., ID_t\} = L$ executes the following steps

1. Chooses $x_t \in \mathbb{Z}_p^*$
2. Computes $X_t = x_tP$, $Y_t = \theta^{x_t}$ and $U_t = x_t(R + Q_B)$.
3. Sends $(X_t, Y_t, U_t)$ to other signers through a secure channel.
4. After receiving from the other signers $(X_j, Y_j, U_j)$, computes
   
   (i) $X = \sum_{i=1}^{t} X_i$, $Y = \prod_{i=1}^{t} Y_i$, $Q = \sum_{i=1}^{t} Q_i$ and $U = \sum_{i=1}^{t} U_i$
   
   (ii) $c = H_1(Y) \oplus m$
   
   (iii) $h = H_2(c, X, U)$
   
   (iv) $Z_t = hD_t + x_tQ$
6. Sends $Z_t$ to other users.
7. Computes $Z$ and outputs the ciphertext $\sigma$, where $Z = \sum_{i=1}^{t} Z_i$ and $\sigma = \langle c, X, Z, U, L \rangle$.

**Unsigncrypt:** To unsigncrypt the ciphertext $\sigma = \langle c, X, Z, U, L \rangle$, the receiver with identity $ID_B$

1. Computes
   
   (i) $h = H_2(c, X, U)$
   
   (ii) $e(P, Z)$
   
   (iii) $e(X + hP_{pub}, Q)$
2. Rejects the ciphertext if $e(P, Z) \neq e(X + hP_{pub}, Q)$, otherwise
3. Computes $Y' = e(P_{pub}, U)e(X, D_B)^{-1}$
4. Recovers $m = c \oplus H_1(Y')$

### 9.2 HEURISTIC SECURITY OF IBMSC1

#### 9.2.1 Confidentiality of IBMSC1
Without knowing the secret key of the receiver, no one can compute
\[ Y = \prod_{i=1}^{t} Y_i = \theta(X_1 + \ldots + X_t) = e(P_{pub}, R)^{X_1 + \ldots + X_t}. \]
It is only the specific receiver who can compute the actual value of \( Y \) using secret key as
\[
Y' = e(P_{pub}, U)e(X, D_B)^{-1} = e(P_{pub}, \sum_{i=1}^{t} U_i)e(X, D_B)^{-1} \\
= e(P_{pub}, \sum_{i=1}^{t} x_i(R + Q_B))e(X, D_B)^{-1} \\
= e(P_{pub}, \sum_{i=1}^{t} x_i R)e(P_{pub}, \sum_{i=1}^{t} x_i Q_B)e(X, D_B)^{-1} \\
= e(P_{pub}, (X_1 + \ldots + X_t)R)e(sP_{pub}, (X_1 + \ldots + X_t)Q_B)e(X, D_B)^{-1} \\
= e(P_{pub}, R)^{(X_1 + \ldots + X_t)}e((X_1 + \ldots + X_t)P, sQ_B)e(X, D_B)^{-1} \\
= \theta^{(X_1 + \ldots + X_t)}e(X, D_B)e(X, D_B)^{-1} \\
= \theta^{(X_1 + \ldots + X_t)} = Y.
\]

9.2.2 Public Verifiability of IBMSC1
Anyone who has access to the signcryption can verify the signature on the ciphertext.
First the verifier computes \( h = H_2(c, X, U) \) and \( Q = \sum_{i=1}^{t} Q_i \), then checks
\[
e(P, Z) = e(P, \sum_{i=1}^{t} (hD_i + x_i Q)) = e(P, \sum_{i=1}^{t} hD_i)e(P, \sum_{i=1}^{t} x_i Q) \\
= e(P, \sum_{i=1}^{t} hQ)e(P, \sum_{i=1}^{t} x_i Q) = e(P, h\sum_{i=1}^{t} Q_i)e(P, (X_1 + \ldots + X_t)Q) \\
= e(P, hQ)e((X_1 + \ldots + X_t)P, Q) = e(hsP, Q)e(X, Q) \\
= e(hP_{pub}, Q)e(X, Q) = e(X + hP_{pub}, Q).
\]

9.2.3 Unforgeability of IBMSC1
Signcryption is generated using the secret key \( D_i \) of each of the signers. Thus no one, not even the one among the signers can generate a valid signcryption without knowing the secret key of all the signers.

9.3 EFFICIENCY COMPARISON AND REMARKS
We compare the efficiency of our scheme with the existing schemes [SVR09, ZYN09] in the following table. We consider the costly operations which include
scalar multiplications in $G_1$ ($G_1 \text{ Mul}$), exponentiations in $G_2$ ($G_2 \text{ Exp}$) and pairing operations (Pairing).

<table>
<thead>
<tr>
<th></th>
<th>Signcrypt</th>
<th>Unsigncrypt</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$G_1 \text{ Mul}$</td>
<td>$G_2 \text{ Exp}$</td>
</tr>
<tr>
<td>S. Deva et al. [SVR09]</td>
<td>$3t$</td>
<td>$t$</td>
</tr>
<tr>
<td>Zhang et al. [ZYN09]</td>
<td>$4t$</td>
<td>$0$</td>
</tr>
<tr>
<td>Proposed Scheme</td>
<td>$4t$</td>
<td>$t$</td>
</tr>
</tbody>
</table>

**Remarks:**

1. In the proposed schemes, we used the concept of Duan and Cao [DC06] which they used to construct an Identity based multi-receiver signcryption scheme.
2. One of the important advantages of the proposed scheme is that no pairing computation is needed for signcryption. This makes the scheme quite efficient.
3. To achieve efficiency in Zhang et al. scheme [ZYN09], only one signer can compute the signcrypted text but in our scheme every user can generate own copy of signcrypted text.
4. The proposed scheme is publicly verifiable. Anyone who has access to the signcrypted text can verify the signature on ciphertext $\sigma$. Thus the proposed scheme is more applicable when signing a joint confidential contract between two or more organizations. Anyone can verify the authenticity of the contract without getting any knowledge of it, however, only the authority can read the contract.

**9.4 IDENTITY BASED MULTI-SIGNCRYPTION SCHEME FOR MULTIPLE RECEIVER**

An identity based multi- signcryption scheme for multi-receivers consists of the following algorithms:
Chapter 9  An Efficient and Public Verifiable Identity Based Multi Signcryption

**Setup:** Given a security parameter $k$, the Private Key Generator (PKG) chooses a secret value randomly and generates master secret key $msk$ and the public parameters $params$ of the system.

**Extract:** Given a user identity $ID_U$, the PKG computes the corresponding private key $D_U$ and transmits it to the user in a secure way.

**Signcrypt:** Any $t$ users $L = \{ID_1, ..., ID_t\}$ run this algorithm to signcrypt a message $m$ to any $t'$ receivers with identities $L' = \{ID'_1, ..., ID'_{t'}\}$, and obtain signcryption $\sigma$.

**Unsigncrypt:** Each receiver with identity $ID'_j$ and private key $D'_j$ runs this algorithm to obtain plain text $m$ if $\sigma$ is a valid signcryption from $L$ to identity $ID'_j$ otherwise return $\perp$.

### 9.5 IBMSC2: THE PROPOSED IBMSC SCHEME FOR MULTIPLE RECEIVER

**Setup:** Same as in IBMSC1 except the hash function $H_2$. In this scheme, $H_2$ is defined as $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$. The system public parameters are

$$params = \langle p, G_1, G_2, e, n, P, P_{pub}, \theta, R, H_0, H_1, H_2 \rangle$$

**Extract:** Given a user identity $ID_U \in \{0,1\}^*$ then PKG computes public key $Q_U = H_0(ID_U)$ and private key $D_U = sQ_U$.

**Signcrypt:** To signcrypt a message $m \in \{0,1\}^n$ for the $t'$ receivers with identities $L' = \{ID'_1, ..., ID'_{t'}\}$, each sender $ID_i \in \{ID_1, ..., ID_t\} = L$ execute the following steps

1. Chooses $x_i \in \mathbb{Z}_p^*$
2. Computes $X_i = x_iP$, $Y_i = \theta x_i$ and $U_{i,j} = x_i(R + Q'_j)$ for $j = 1, ..., t'$
3. Sends $(X_i, Y_i, U_{i,1}, U_{i,2}, ..., U_{i,t'})$ to other signers through a secure channel.
4. After receiving from the other signers $(X_i, Y_i, U_{i,1}, U_{i,2}, ..., U_{i,t'})$, computes
   
   $$(i) \quad X = \sum_{i=1}^{t} X_i, \quad Y = \prod_{i=1}^{t} Y_i, \quad Q = \sum_{i=1}^{t} Q_i$$
\[ (ii) \quad U_1 = \sum_{i=1}^t U_{i,1}, U_2 = \sum_{i=1}^t U_{i,2}, \ldots, U'_t = \sum_{i=1}^t U_{i,t'} \]

\[ (iii) \quad c = H_1(Y) \oplus m \]

\[ (iv) \quad h = H_2(c, X, U_1, U_2, \ldots, U'_t) \]

\[ (v) \quad Z_i = hD_i + x_iQ \]

5. Sends \( Z_i \) to other users.

6. Computes \( Z \) and outputs ciphertext \( \sigma \), where \( Z = \sum_{i=1}^t Z_i \) and \( \sigma = \langle c, X, Z, U_1, U_2, \ldots, U'_t, L, L' \rangle \).

**Unsigncrypt:** To unsigncrypt the ciphertext \( \sigma = \langle c, X, Z, U_1, U_2, \ldots, U'_t, L, L' \rangle \), the receiver with identity \( ID_j' \) computes

1. Computes
   
   (i) \( h = H_2(c, X, U_1, U_2, \ldots, U'_t) \)

   (ii) \( e(P, Z) \)

   (iii) \( e(X + hP_{pub}, Q) \)

2. Reject the ciphertext if \( e(P, Z) \neq e(X + hP_{pub}, Q) \), otherwise

3. Computes \( Y' = e(P_{pub}, U_j)e(X, D'_j)^{-1} \)

4. Recovers \( m = c \oplus H_1(Y') \)

**9.6 HEURISTIC SECURITY OF IBMSC2**

**9.6.1 Confidentiality of IBMSC2**

Without knowing the secret key of the receiver, no one can compute

\[ Y = \prod_{i=1}^t Y_i = \theta^{(x_1 + \ldots + x_i)} = e(P_{pub}, R)^{(x_1 + \ldots + x_i)}. \]

It is only the specific receiver who can compute the actual value of \( Y \) using the secret key as

\[ Y' = e(P_{pub}, U_j)e(X, D'_j)^{-1} = e(P_{pub}, \sum_{i=1}^t U_{i,j})e(X, D'_j)^{-1} \]

\[ = e(P_{pub}, \sum_{i=1}^t x_i(R + Q'_j))e(X, D'_j)^{-1} \]
Chapter 9  

An Efficient and Public Verifiable Identity Based Multi Signcryption

\[ e(P_{pub}, \sum_{i=1}^{t} x_i R) e(P_{pub} + \sum_{i=1}^{t} x_i Q'_j) e(X, D'_j)^{-1} \]

\[ = e(P_{pub}, (x_1 + ... + x_t) R) e(sP, (x_1 + ... + x_t) Q'_j) e(X, D'_j)^{-1} \]

\[ = e(P_{pub}, R)^{(x_1 + ... + x_t)} e((x_1 + ... + x_t) P, sQ'_j) e(X, D'_j)^{-1} \]

\[ = \theta^{(x_1 + ... + x_t)} e(X, D'_j)^{-1} = \theta^{(x_1 + ... + x_t)} Y. \]

9.6.2 Public Verifiability of IBMSC2

Anyone who has access to the signcryptext can verify the signature on the ciphertext which it contains. First the verifier computes \( h = H_2(c, X, U_1, U_2, ..., U_t), \)

\[ Q = \sum_{i=1}^{t} Q_i, \] then checks

\[ e(P, Z) = e(P, \sum_{i=1}^{t} (hD_i + x_i Q)) = e(P, \sum_{i=1}^{t} hD_i) e(P, \sum_{i=1}^{t} x_i Q) \]

\[ = e(P, \sum_{i=1}^{t} hS_i) e(P, \sum_{i=1}^{t} x_i Q) = e(P, hS \sum_{i=1}^{t} Q_i) e(P, (x_1 + ... + x_t) Q) \]

\[ = e(P, hS Q) e((x_1 + ... + x_t) P, Q) = e(hS P, Q) e(X, Q) \]

\[ = e(hP_{pub}, Q) e(X, Q) = e(X + hP_{pub}, Q). \]

9.6.3 Unforgeability of IBMSC2

Signcryptext is generated using the secret key \( D_i \) of each signer. Thus no one, not even the one among signers can generate a valid signcryptext without knowing the secret key of all the signers \( Z_i \).

CONCLUSION

In this chapter, we have proposed an efficient identity based multisigncryption scheme. We discussed its confidentiality, unforgeability and public verifiability in heuristic way and compared it with two existing ID-based multisigncryption schemes. We also extended the proposed scheme to multi-signcryption scheme for multiple receivers.