Universally Composable Secure Group Communication

Youliang Tian[1,2] and Changgen Peng[1,2]

[1] College of Science, Guizhou University, Guiyang 550025, China
[2] Institute of Cryptography and Data Security, Guzhou University, Guiyang 550025, China
(Email: tianyouliang@iie.ac.cn)

Abstract. This paper analyzes group communication within the universally composable framework. We first propose the group communication model, identity-based signcrytion model and group key distribution model in the UC framework by designing the ideal functionality $\mathcal{F}_{SAGCOM}$, $\mathcal{F}_{IDSC}$ and $\mathcal{F}_{GKD}$, respectively. Then, we construct a UC secure identity-based signcryption protocol π_{IDSC}. Moreover, we shows that the identity-based signcryption π_{IDSC} securely realizes the ideal functionality $\mathcal{F}_{IDSC}$ if and only if the corresponding protocol IDSC is secure. Finally, based on the identity-based protocol, we propose a group communication scheme π_{SAGCOM}, which can securely realizes the ideal functionality $\mathcal{F}_{SAGCOM}$ in the $(\mathcal{F}_{IDSC}, \mathcal{F}_{GKD})$-hybrid model.

Keywords: Universally composable secure, Secure group communication, Identity-based signcrytion, Many-to-many group communication

1 Introduction

As is known to all, multicasting is increasingly used as an efficient communication mechanism for delivering data to multiple receivers in one-to-many or many-to-many group-oriented applications in the Internet or public broadcasting. However, the lack of security in the multicast communication model obstructs the effective and large scale deployment of multi-party applications. Much research has been done into being traditionally defined as stand-alone scheme problems without giving much attention to more complex execution environments to to multicast group communications. Therefore, it is extremely interesting to study multicast group communication scheme within the universally composable framework.

Related work Fiat and Naor [2] introduced broadcast encryption, which provided a method of securely broadcasting key information such that only a privileged set of users can decrypt the information while a coalition of up to k other users cannot know anything. Wang and Wu [4] proposed an authenticated identity-based multicast scheme from bilinear pairing , where security is also provided. However, Lin et al. [5] found that [4] is not secure against the insider forgery attack. Mu et al. [6] proposed another identity-based authenticated broadcast encryption scheme, which allows each sender to dynamically broadcast messages to its group members using a polynomial function constructed with secret keys of the members. Hur et al. [7] proposed an authenticated group communication scheme which is secure against an adaptive chosen ciphertext attack using identity-based signcryption. And solved the sender authentication problem by using an identity-based signcryption framework.

However, the above mentioned schemes do not study their security in the universally composable framework. Thus these schemes could not satisfy universally composable secure.

The UC framework The universally composable (UC) framework [1] for analyzing security of cryptographic protocols provides very strong security guarantees. In particular, a protocol proven secure in this framework is guaranteed to maintain its security even when it is run concurrently with other protocols, or when it is used as a component of a large protocol. Ideal functionality is an extremely important security concept in the UC framework; it serves as an uncorruptable trusted party and can realize the specific task of carrying out the protocol.

At present, most basic ideal functionalities have already been defined, such as the message authentication functionality $\mathcal{F}_{AUTH}$, the key-exchange functionality $\mathcal{F}_{KE}$, the public-key encryption functionality $\mathcal{F}_{PKE}$, the signature functionality $\mathcal{F}_{SIG}$, the commitment functionality $\mathcal{F}_{COM}$, the zero-knowledge functionality $\mathcal{F}_{ZK}$, the oblivious transfer functionality $\mathcal{F}_{OT}$, the anonymous hash authentication functionality $\mathcal{F}_{Cred}$ [8], the deniable authentication functionality $\mathcal{F}_{CDA}$[9], the trusted network connect (TNC) functionality $\mathcal{F}_{TNC}$ [10], the one-time signature functionality $\mathcal{F}_{OTS}$ [11], broadcast authentication functionality $\mathcal{F}_{BAUTH}$ [11] and so on.

Our contribution In this paper, we investigate secure authenticated group communication based on identity-based signcryption in the UC framework. The first step is to formalize an identity-based signcryption in the UC framework. The next step is to construct a UC secure identity-based signcryption protocol in our model. Since our identity-based signcryption only provide a communication mechanism in one-to-one applications, we design a group key distribution functionality for many-to-many applications on the Internet. In the UC frame, we can construct a secure authenticated group communication by composition of identity-based signcryption and group key distribution which are proven secure. Moreover, Data confidentiality of the group communication is guaranteed as well as the sender authentication. Our contributions are shown below.

1. We propose a universally composable group communication model including the ideal functionalities identity-based signcryption $\mathcal{F}_{IDSC}$, group key distribution $\mathcal{F}_{GKD}$ and secure authenticated group communication $\mathcal{F}_{SAGCOM}$.
2. According to $\mathcal{F}_{IDSC}$, we design a protocol π_{IDSC} to realize the identity-based signcryption functionality in the hybrid model. Meanwhile, we show that π_{IDSC} securely realizes $\mathcal{F}_{IDSC}$ if and only if IDSC is secure with respect to both IND-IDSC-CCA2 and EXT-IDSC-CMA.
3. Based on $\mathcal{F}_{IDSC}$ and $\mathcal{F}_{GKD}$, we construct a scheme π_{SAGCOM} to realize the secure authenticated group communication functionality $\mathcal{F}_{SAGCOM}$ in the $(\mathcal{F}_{IDSC}, \mathcal{F}_{GKD})$-hybrid model.

Organization The rest of this paper is organized as follows: Section 2 gives a brief introduction to the UC framework and identity-based signcryption. In section 3, we propose the ideal functionalities of identity-based signcryption,group key distribution and secure authenticated group communication. An identity-based signcryption protocol is proposed in section 4, while in section 5, we present the secure authenticated group scheme. Finally, we conclude our results in section 6.

2 Preliminaries

In this section, we briefly review the framework of UC [1] and identity-based signcryption scheme [3].

2.1 The UC framework

Firstly, the process of executing a protocol in the presence of a real-world adversary is formalized in the UC framework. Secondly, an ideal process for carrying out the task at hand is formalized. In the ideal process, the parties do not communicate with each other. Instead they have access to an ideal functionality, which is essentially an incorruptible "trusted party" that is programmed to capture the desired functionality of the task at hand.

Definition 1. *(UC emulation) We say that a protocol π UC-realizes an ideal functionality $\mathcal{F}$ if for any real-world adversary $\mathcal{A}$, there exists an ideal adversary $\mathcal{S}$ such that for any environment $\mathcal{Z}$, the probability that $\mathcal{Z}$ is able to distinguish between an interaction with $\mathcal{A}$ and real parties running protocol π and an interaction with $\mathcal{S}$ and dummy parties accessing $\mathcal{F}$ in the ideal process is at most a negligible probability, i.e. $REAL_{\pi,\mathcal{A},\mathcal{Z}} \approx IDEAL_{\mathcal{F},\mathcal{S},\mathcal{Z}}$.*

Theorem 1. *(Composition Theorem) Let ρ be a protocol that securely realizes the ideal functionality $\mathcal{F}$, and let π be a protocol in the $\mathcal{F}$-hybrid model. We say that $\pi^{\rho/\mathcal{F}}$ with the ideal functionality $\mathcal{F}$ which is replaced by ρ, UC-realizes π. In particular, if π securely realizes the ideal functionality ξ in the $\mathcal{F}$-hybrid model, then $\pi^{\rho/\mathcal{F}}$ securely realizes ξ from scratch.*

According to the Composition Theorem, a large protocol can be constructed by using some sub-protocols which are proven secure in the UC framework. This is very important since a complex but secure system can usually be divided into a number of sub-systems, each one performing a specific task securely.

2.2 Identity-based signcryption

Our definition of an identity-based signcryption scheme is identical to the one given in [3].

IDSC scheme An identity-based signcryption scheme consists of *Setup*, *Extract*, *Signcrypt*, and *Unsigncrypt* algorithms. The functions of these algorithms are described below.

1. *Setup* The setup algorithm produces global public parameters, the master secret key s, and the master public key on the input of security parameter 1^k.
2. *Extract* The Extract algorithm outputs a secret key of a user on the input of the master secret key and the public identity of the user.
3. *Signcrypt* The Signcrypt algorithm produces a ciphertext σ, which is the signcryption of m with signcrypter's secret key and the public identity of the receiver.
4. *Unsigncrypt* The receiver users the Unsigncrypt algorithm to unsigncrypt the received ciphertext σ with its own secret key and the identity of the signcrypter recover the corresponding plaintext m, and the receiver can verifies the signcryption by checking whether σ is the signcryption of m on using the Unsigncrypt algorithm. If it is, it outputs $\top$; and it outputs $\perp$ otherwise.

Security requirement In the identity-based signcryption scheme, confidentiality and nonrepudiation are always required. We need two experiments described as followings.

$$Exp_{IDSC,\mathcal{A}}^{IND-CCA2}(1^k)$$

$$\begin{aligned}(s, P_{pub}) &\longleftarrow Setup(1^k)\\ (sk_s, ID_s) &\longleftarrow Extract(s, P_{pub}, ID_s)\\ (sk_r, ID_r) &\longleftarrow Extract(s, P_{pub}, ID_r)\\ (m_0, m_1, state) &\longleftarrow \mathcal{A}_1^{\mathcal{O}_E,\mathcal{O}_S,\mathcal{O}_U}(ID_s, ID_r)\\ b &\longleftarrow \{0,1\}\\ \sigma &\longleftarrow Signcrypt(ID_s, ID_r,\\ &\quad P_{pub}, m_b)\\ b' &\longleftarrow \mathcal{A}_2^{\mathcal{O}_E,\mathcal{O}_S,\mathcal{O}_U}(ID_s, ID_r,\\ &\quad m_0, m_1, \sigma, state)\end{aligned}$$

If $b' \neq b$ then retutn 1, otherwise return 0.

$$Exp_{IDSC,\mathcal{A}}^{EXT-CMA}(1^k)$$

$$\begin{aligned}(s, P_{pub}) &\longleftarrow Setup(1^k)\\ (sk_s, ID_s) &\longleftarrow Extract(s, P_{pub}, ID_s)\\ (sk_r, ID_r) &\longleftarrow Extract(s, P_{pub}, ID_r)\\ \sigma &\longleftarrow \mathcal{A}^{\mathcal{O}_E,\mathcal{O}_S,\mathcal{O}_U}(ID_s, ID_r)\end{aligned}$$

If $Unsigncrypt(ID_s, ID_r, \sigma) \neq \perp$ then return 1, otherwise return 0.

Here $\mathcal{O}_E$, $\mathcal{O}_S$, $\mathcal{O}_U$ represent Extract oracle, Signcrypt oracle and Unsigncrypt oracle, respectively. The first experiment concerns privacy of messages, and adapts the notion IND-IDSC-CCA2 from public key encryption. $\mathcal{A}$ is said to win if the experiment returns 1 with non-negligible advantage. The second experiment concerns unforgeability of messages, and adapts the notion EXT-IDSC-CMA from digital signatures. $\mathcal{A}$ is said to win if the experiment returns 1 with non-negligible advantage.

Next we define the advantage of $\mathcal{A}$ in breaking IDSC with respect to IND-IDSC-CCA2 as

$$\begin{aligned}&Adv_{IDSC,\mathcal{A}}^{IND-CCA2}(1^k)\\ =&|2Pr[Exp_{IDSC,\mathcal{A}}^{IND-CCA2}(1^k) = 1] - 1|\end{aligned}$$

The scheme $IDSC$ is said to be secure with respect to IND-IDSC-CCA2 if the advantage $Adv_{IDSC,\mathcal{A}}^{IND-CCA2}(1^k)$ is negligible in ϵ, whenever $\mathcal{A}$'s runtime and number of oracle queries are polynomially bounded in ϵ.

We define the success rate of $\mathcal{A}$ in breaking IDSC with respect to EXT-IDSC-CMA as

$$Succ_{IDSC,\mathcal{A}}^{EXT-CMA}(1^k) = Pr[Exp_{IDSC,\mathcal{A}}^{EXT-CMA}(1^k) = 1]$$

The scheme IDSC is said to be secure with respect to EXT-IDSC-CMA if the success rate $Succ_{IDSC,\mathcal{A}}^{EXT-CMA}(1^k)$ is negligible in ϵ, whenever $\mathcal{A}$'s runtime and number of oracle queries are polynomially bounded in ϵ.

3 Ideal functionalities

In this section, we mainly consider the ideal procedures for a secure group communication and an identity-based signcryption scheme. Then their functionalities are presented based secure requirements of these protocols.

3.1 Functionality, $\mathcal{F}_{SAGCOM}$

A secure authenticated group communication means that a party R or a group G will receive a message m from some parties S only if S has sent the message m to R or G, and in addition adversary and non-group members have no access to the contents of the transmitted message. Obviously, the main security properties of a secure group communication are secrecy and authenticity. Specifically, the security requirements are considered to be of message confidentiality, ciphertext authentication, and signature non-repudiation.

Confidentiality Message confidentiality prevents outsiders or non-group members from decrypting the group message. It allows the communicating parties to preserve the secrecy of their communications.

Authentication Ciphertext authentication allows only the intended legitimate recipient to be convinced that the message was encrypted by the same person who signed it. That is, an outside adversary cannot re-encrypt a signed message of the sender throughout the transmission. This implies ciphertext integrity.

Non-repudiation Signature non-repudiation prevents the sender of a signed message from disavowing its signature. It can be also verified by only the intended recipient of the signature.

In the present formalization, protocols that assume ideally secure group communication can be cast as hybrid protocol with ideal access to an "authenticated message transmission functionality" and "secure message functionality". This functionality, denoted $\mathcal{F}_{SAGCOM}$, is presented in Figure 1.

3.2 Functionality, $\mathcal{F}_{IDSC}$

Here, we define an ideal functionality of identity-based signcryption, based on the security requirements given in section 2.2. The identity-based signcryption functionality, denoted $\mathcal{F}_{IDSC}$, is defined in Figure 2.

3.3 Functionality, $\mathcal{F}_{GKD}$

In this section, we formulate an ideal functionality, $\mathcal{F}_{GKD}$ (shown in Figure 3), to provide a trusted "the group key distribution service".

4 Securely realizing $\mathcal{F}_{IDSC}$

Here we present a universally composable identity-based signcryption scheme π_{IDSC} and its proof of the UC-security in the hybrid model.

4.1 Protocol, π_{IDSC}

The protocol π_{IDSC} given in Figure 4 is constructed in a natural way from the identity-based signcryption scheme IDSC.

4.2 Security proof of π_{IDSC}

Theorem 2. *Let IDSC be an identity-based signcryption scheme. π_{IDSC} securely realizes $\mathcal{F}_{IDSC}$ if and only if IDSC is secure with respect to both IND-IDSC-CCA2 and EXT-IDSC-CMA.*

Proof. $\Rightarrow$ (Reduction to absurdity) Let π_{IDSC} securely realizes $\mathcal{F}_{IDSC}$, but IDSC is not secure with respect to both IND-IDSC-CCA2 and EXT-IDSC-CMA. Since IDSC is not both IND-IDSC-CCA2 and EXT-IDSC-CMA, then we construct an environment $\mathcal{Z}$ and a real-world adversary $\mathcal{A}$ such that for any ideal adversary $\mathcal{S}$, the environment $\mathcal{Z}$ can distinguish between an interaction with $\mathcal{A}$ and real parties running protocol π_{IDSC} and an interaction with $\mathcal{S}$ and dummy parties accessing $\mathcal{F}_{IDSC}$ in the ideal process.

1. Suppose IDSC is not IND-IDSC-CCA2, i.e. there exists a adversary $\mathcal{B}$ who has non-negligible advantage $adv_{IDSC,\mathcal{B}}^{IND-CCA2}(1^k)$ in breaking IDSC. The system description follows: In the beginning of the experiment $Exp_{IDSC,\mathcal{A}}^{IND-CCA2}(1^k)$, the adversary $\mathcal{B}$ is given two public identities ID_s and ID_r belonging to the target sender and the target receiver, respectively. $\mathcal{B}$ is composed of a $find$-stage algorithm $\mathcal{B}_1$ and a $guess$-stage algorithm $\mathcal{B}_2$. $\mathcal{B}_1$ finds two messages m_0 and m_1 of the same length, while $\mathcal{B}_2$ is given a challenge ciphertext σ and guesses the bit $b' \in \{0,1\}$ correctly with probability $1/2+\epsilon$. Environment $\mathcal{Z}$ invokes an instance of $\mathcal{F}_{IDSC}$, and proceeds as follows, in a network of a trusted KGC and two uncorrupted parties S, R.
(1) Initially, $\mathcal{Z}$ activates KGC with input $(Setup, sid)$ for $sid = (KGC, sid')$, obtains system public key PK_s and hands PK_s to $\mathcal{B}$.
(2) Next $\mathcal{Z}$ activates the parties S and R with input $(Extract, sid)$ for $sid = (\{S, R\}, sid')$,

Functionality $\mathcal{F}_{SAGCOM}$

$\mathcal{F}_{SAGCOM}$ proceeds as follow, when parameterized by leakage function $l : \{0,1\}^* \rightarrow \{0,1\}^*$; and S and R are the sets of some parties.

Upon receiving an input $(Send, sid, T, m)$ from some parties S_i, where $S_i \in B, B \subset S$ and $T \subset R$, if $sid = (B, T, sid')$ for some sets of senders S, then:

- Send $(Send, sid, T, l(m))$ to the adversary.
- Generate a private delayed output $(Send, sid, m)$ to R_i, for all $R_i \in T$.
- Record B sends m to T and halts.
- Else ignore the input.

Upon receiving an input $(Receive, sid, B, m)$ from parties R_i, if $sid = (T, B, sid')$ for some sets of receivers R, then:

- Send $(Receive, sid, B, l(m))$ to the adversary.
- Generate a public output $(Received, sid, ok)$ to S_i, for all $S_i \in B$.
- Record T receives m from B and halts.
- Else ignore the input.

Upon receiving $(Corrupt, sid, P)$ from the adversary, where $P \in B \cup T$. Then:

- Disclose m to the adversary and record P is corrupted.
- If the adversary provides a value m', and $P \in S$, and no output has been yet written to the receiver, then output $(Send, sid, m')$ to the receiver, record it, and halt.

Fig. 1. The secure authenticated group communication functionality, $\mathcal{F}_{SAGCOM}$

obtains their identity ID_S and ID_r, and hands ID_s and ID_r to $\mathcal{B}$.
(3) When $\mathcal{B}$ submits a message m' and two distinct identities S' and R' (one is a sender identity and the other is a receiver identity), $\mathcal{Z}$ activates S' with $(Signcrypt, sid, ID_{r'}, ID_{s'}, PK_s, m')$, obtains a ciphertext σ' and hand σ' to $\mathcal{B}$.
(4) When $\mathcal{B}$ submits a ciphertext σ' and an identity R' of the receiver, $\mathcal{Z}$ activates R' with $(Unsigncrypt, sid, ID_{s'}, ID_{r'}, PK_s, \sigma')$, obtains a plaintext m' and hand m' to $\mathcal{B}$.
(5) When $\mathcal{B}$ submits an identity ID_t, $\mathcal{Z}$ activates KGC with $(Extract, sid)$, obtains the corresponding private key sk_t and hands sk_t to $\mathcal{B}$.
(6) When $\mathcal{B}$ generates the two test plaintexts (m_0, m_1), $\mathcal{Z}$ chooses $b \xleftarrow{R} \{0,1\}$, activates S with $(Signcrypt, sid, ID_r, PK_s, m_b)$, obtains a ciphertext σ^*, and hands σ^* to $\mathcal{B}$ as the text ciphertext.
(7) When $\mathcal{B}$ returns a guess $b' \in \{0,1\}$, $\mathcal{Z}$ outputs $b \oplus b'$ and halts.
Analyzing $\mathcal{Z}$, notice that if interacts with the adversary $\mathcal{A}$ and parties running π_{IDSC}, then the instance of $\mathcal{B}$ within $\mathcal{Z}$ sees in fact a CCA2 interaction with protocol IDSC. Thus, in this case $b' = b$ with probability at least $1/2+\epsilon$. In the contrast, when $\mathcal{Z}$ interacts with the ideal protocol for $\mathcal{F}_{IDSC}$ and any adversary, the view of the instance of $\mathcal{B}$ within $\mathcal{Z}$ is statistically independent of b, thus in this case $b' = b$ with probability exactly one half.

2. Suppose IDSC is not EXT-IDSC-CMA, i.e. the success rate $Succ^{EXT-CMA}_{IDSC,\mathcal{A}} 1^k$ of the experiment $Exp^{EXT-CMA}_{IDSC,\mathcal{A}}(1^k)$ is non-negligible. We construct a simulated instance of $\mathcal{B}$, where $\mathcal{B}$ is an adversary of forgeable signcrypt with non-negligible probability.
(1) Initially, $\mathcal{Z}$ activates $\mathcal{B}$.
(2) Next, $\mathcal{Z}$ activates KGC with input $(Setup, sid)$ for $sid = (KGC, sid')$, obtains system public key PK_s and hands PK_s to $\mathcal{B}$.
(3) When $\mathcal{B}$ asks an identity ID_t, $\mathcal{Z}$ activates KGC with $(Extract, sid)$, obtains the corresponding private key sk_t and hands sk_t to $\mathcal{B}$.
(4) When $\mathcal{B}$ submits a message m' and two distinct identities S' and R' (one is a sender identity and the other is a receiver identity), $\mathcal{Z}$ activates S' with

Functionality $\mathcal{F}_{IDSC}$

$\mathcal{F}_{IDSC}$ proceeds as follows, with parties $P_1, \cdots, P_n$, KGC and an ideal adversary $\mathcal{S}$.

Upon receiving a message from a corrupted party, $\mathcal{F}_{IDSC}$ forward the message to $\mathcal{S}$, and when $\mathcal{S}$ replies to this message, $\mathcal{F}_{IDSC}$ forwards the reply to the corrupted party.

IDSC.Setup

Upon receiving the message $(IDSC.Setup, sid)$ from a KGC:

- Send $(IDSC.Setup, sid, KGC)$ to $\mathcal{S}$.
- Upon receiving $(IDSC.Setup, sid, PK_s)$ from $\mathcal{S}$, output $(IDSC.Setup, sid, PK_s)$ to the KGC.
- Record (KGC, PK_s) and label it 'fresh'.

IDSC.Extract

Upon receiving the first message $(IDSC.Extract, sid, PK'_s)$ from some party P_i, send $(IDSC.Extract, sid, ID_i, PK'_s)$ to $\mathcal{S}$, where ID_i is the identity of P_i.

Upon receiving $(IDSC.Received, sid)$, send $(IDSC.Extract, sid, ID_i, PK'_s)$ to P_i and KGC.

IDSC.Signcrypt

Upon receiving $(IDSC.Signcrypt, sid, IDr, PKs, m)$ from P_i, do:

- If $ID_r = ID_j$ for some j, $PK_s = PK_l$ for some l and KGC are uncorrupted, then send $(IDSC.Signcrypt, sid, ID_i, ID_r, PK_s, |m|)$ to $\mathcal{S}$.
- Otherwise send $(IDSC.Signcrypt, sid, ID_i, ID_r, PK_s, m)$ to $\mathcal{S}$.

Upon receiving $(IDSC.Signcrypt.Ciphertext, sid, ID_i, ID_r, PK_s, \sigma)$ from $\mathcal{S}$, do:

- If there is no recorded entry $(ID_i, ID_r, PK_s, m', \sigma)$ for any m',
  output $(IDSC.Signcrypt.Ciphertext, sid, ID_i, ID_r, PK_s, m, \sigma)$ to P_i.
- If $ID_r = ID_j$ for some j, $PK_s = PK_l$ for some l and KGC are uncorrupted, then record the entry $(ID_i, ID_r, PK_s, m', \sigma)$.

IDSC.Unsigncrypt

Upon receiving $(IDSC.Unsigncrypt, sid, ID_s, PK_s, \sigma)$ from P_j, do:

- Send $(IDSC.Unsigncrypt, sid, ID_s, ID_j, PK_s, \sigma)$ to $\mathcal{S}$.
  Upon receiving $(IDSC.Unsigncryp.Plaintext, sid, ID_s, ID_j, PK_s, m'/\perp, \sigma)$ from $\mathcal{S}$, continue.
- If an entry $(ID_s, ID_j, PK_s, m, \sigma)$ is record,
  then output $(IDSC.Unsigncrypt.Plaintext, sid, ID_s, ID_j, PK_s, m, \sigma)$ to P_j.
- Otherwise, if $ID_s = ID_i$ for some i and P_i and $PK_s = PK_l$ for some l and KGC are uncorrupted, then output $(IDSC.Unsigncrypt.Plaintext, sid, ID_s, ID_j, PK_s, \perp, \sigma)$ to P_j.
- Otherwise, output $(IDSC.Unsigncryp.Plaintext, sid, ID_s, ID_j, PK_s, m'/\perp, \sigma)$ to P_j.

Fig. 2. The identity-based signcryption functionality, $\mathcal{F}_{IDSC}$

Functionality $\mathcal{F}_{GKD}$

$\mathcal{F}_{GKD}$ proceeds as follows, with parties $P = \{P_1, \cdots, P_n\}$, KGC and an adversary $\mathcal{S}$.

Upon receiving an input $(\mathit{GKD.Distribute}, \mathit{sid}, s)$ from KGC, where $\mathcal{T} \in 2^P$ is an access structure and s is the group key, do:

- If there exists $\mathit{sid} = (KGC, \mathcal{T}, \mathit{sid}')$, then:
  1. send $(\mathit{GKD.Distribute}, \mathit{sid}, |s|)$ to $\mathcal{S}$.
  2. generate a private delayed output $(\mathit{GKD.Distributed}, \mathit{sid}, s_i)$ to P_i, for all $P_i \in P$.
  3. record (sid, s) and halt.
- Otherwise halt.
- Once (sid, s) is recorded, ignore any subsequent distribute inputs.

Upon receiving an input $(\mathit{GKD.Recover}, \mathit{sid})$ from P_i, do:

- Add P_i to a set T (initially $T := \phi$).
- If there is a set $T \in \mathcal{T}$ and there is a recorded distributed group key s, then:
  1. send $(\mathit{GKD.Recover}, \mathit{sid}, |s|)$ to $\mathcal{S}$.
  2. generate a private delayed output $(\mathit{GKD.Recovered}, \mathit{sid}, s)$ to the parties in T.
  3. record (sid, s) and halt.
- Else halt.

Upon receiving a message $(\mathit{Corrupt}, \mathit{sid}, P_i)$ from the adversary $\mathcal{S}$, do:

- Add P_i to a set C of the corrupted (initially $C := \phi$).
- If the adversary $\mathcal{S}$ provides an invalid s' and $(\mathit{GKD.Distributed}, \mathit{sid}, s_i)$ was not yet written on the tape of any uncorrupt party in T, then change the recorded value (sid, s').
- If the adversary $\mathcal{S}$ provides an invalid s' and $(\mathit{GKD.Recovered}, \mathit{sid}, s_i)$ was not yet written on the tape of any uncorrupt party in T, then change the recorded value (sid, s'). Else hands $(\mathit{GKD.Recovered}, \mathit{sid}, s)$ to the adversary $\mathcal{S}$.
- Else halt.

Fig. 3. The group key distribution functionality, $\mathcal{F}_{GKD}$

Protocol π_{IDSC}

π_{IDSC} proceeds as follows, with parties $P_1, \cdots, P_n$, KGC and an adversary $\mathcal{A}$.

IDSC.Setup

Upon the first input $(\mathit{IDSC.Setup}, sid)$, KGC runs the algorithms Setup. The KGC obtains a master key pair (s, P_{pub}), and outputs $(\mathit{IDSC.Setup}, sid, (s, P_{pub}))$.

IDSC.Extract

Upon the first input $(\mathit{IDSC.Extract}, sid, ID_i)$, P_i runs the algorithms Extract. P_i obtains his private key sk_i, and outputs $(\mathit{IDSC.Extract}, sid, (sk_i, ID_i))$.

IDSC.Signcrypt

Upon input $(\mathit{IDSC.Signcrypt}, sid, P_{pub}, ID_r, m)$, P_i obtains $\sigma = \mathit{Signcrypt}(sk_i, P_{pub}, ID_r, m)$ and outputs $(\mathit{IDSC.Signcrypt.Ciphertext}, sid, ID_r, m, \sigma)$.

IDSC.Unsigncrypt

Upon input $(\mathit{IDSC.Unsigncrypt}, sid, P_{pub}, ID_s, \sigma)$, P_i obtains $m/\perp = \mathit{Unsigncrypt}(ID_s, P_{pub}, sk_i, \sigma)$ and outputs $(\mathit{IDSC.Unsigncrypt.Plaintext}, sid, m/\perp, \sigma)$.

Fig. 4. The identity-based signcryption protocol, π_{IDSC}

$(\mathit{Signcrypt}, sid, ID_{r'}, ID_{s'}, PK_s, m')$, obtains a ciphertext σ' and hand σ' to $\mathcal{B}$.

(5) When $\mathcal{B}$ generates a successful forgeable ciphertext σ^* using an identity R' of the receiver, $\mathcal{Z}$ activates R' with $(\mathit{Unsigncrypt}, sid, ID_{s'}, ID_{r'}, PK_s, \sigma^*)$, obtains a plaintext m^* and hand m^* to $\mathcal{B}$.

(6) If m^* is not 'fresh', then $\mathcal{Z}$ outputs 0. Otherwise outputs whatever R' outputs.

Obviously, when $\mathcal{Z}$ interacts with the adversary $\mathcal{A}$ and parties running π_{IDSC}, $\mathcal{Z}$ outputs 1 with non-negligible probability since $\mathcal{B}$ can forge a valid signcryption. However, when $\mathcal{Z}$ interacts with the adversary $\mathcal{S}$ and parties running $\mathcal{F}_{IDSC}$, $\mathcal{Z}$ can not outputs 1.

The above analysis shows that π_{IDSC} securely realizes $\mathcal{F}_{IDSC} \Rightarrow$ IDSC is secure with respect to both IND-IDSC-CCA2 and EXT-IDSC-CMA.

$\Leftarrow$ It remains to show that if IDSC is secure with respect to both IND-IDSC-CCA2 and EXT-IDSC-CMA, then π_{IDSC} securely realizes $\mathcal{F}_{IDSC}$. We show that π_{IDSC} securely realizes $\mathcal{F}_{IDSC}$. Using the equivalent notion of security with respect to the dummy adversary, we construct an ideal-process adversary $\mathcal{S}$ such that no environment $\mathcal{Z}$ can tell with non-negligible probability whether it interacts with $\mathcal{F}_{IDSC}$ and $\mathcal{S}$ or with parties running π_{IDSC} and the dummy adversary $\mathcal{D}$.

Assume for contradiction there is an environment $\mathcal{Z}$ that distinguishes between the real and ideal interactions. We use $\mathcal{Z}$ construct an adversary $\mathcal{B}$ that break the IND-IDSC-CCA2 and EXT-IDSC-CMA security of the identity-based signcryption scheme IDSC.

Adversary $\mathcal{B}$ proceeds as follows, given a extract algorithm $\mathit{Extract}$, a signcryption algorithm $\mathit{Signcrypt}$ and an unsigncryption algorithm $\mathit{Unsigncrypt}$, and having access to a extract oracle $\mathcal{O}_E$, a signcryption oracle $\mathcal{O}_S$ and an unsigncryption oracle $\mathcal{O}_U$. $\mathcal{B}$ first randomly chooses a number $e \xleftarrow{R} \{1, \cdots, q\}$ and $s \xleftarrow{R} \{1, \cdots, p\}$, where q is the total number of parties that were extracted and p is the total number of messages that were signcrypted throughout the run of the system. Next, $\mathcal{B}$ runs $\mathcal{Z}$ on the following simulated interaction with a system running π_{IDSC} (and the dummy adversary $\mathcal{D}$). Let m_i denote the i-th message that $\mathcal{Z}$ asks to signcrypt in an execution.

(1) Initially, $\mathcal{Z}$ activates $\mathcal{B}$. Next, $\mathcal{Z}$ activates KGC with input (Setup, sid) for $sid = (KGC, sid')$, obtains system public key PK_s and hands PK_s to $\mathcal{B}$.

(2) For the first $e-1$ times that $\mathcal{Z}$ asks to extract some parties' identities, ID_i, $\mathcal{B}$ let-

s the extracting party KGC outputs $sk_i = Extract(ID_i)$.

(3) At the e-th time that $\mathcal{Z}$ asks to extract a party's identity, ID_e, $\mathcal{B}$ queries its extract oracle $\mathcal{O}_E$ with the pair of identities (ID_e, ID_f), where ID_f is the fixed identity used above, and obtains the text private key sk_e. It then hands sk_e to $\mathcal{Z}$ as the private key of the party P_e (whose identity is the ID_e).

(4) For the remaining $q - e$ times that $\mathcal{Z}$ asks to extract some party, ID_i, $\mathcal{B}$ lets the KGC outputs $sk_i = Extract(ID_f)$.

(5) For the first $s - 1$ times that $\mathcal{Z}$ asks to signcrypt some message, m_i, $\mathcal{B}$ lets the signcrypting party return $\sigma_i = Signcrypt(m_i)$.

(6) At the s-th time that $\mathcal{Z}$ asks to signcrypt a message, m_s, $\mathcal{B}$ queries its signcryption oracle $\mathcal{O}_S$ with the pair of messages (m_e, m_f), where m_f is the fixed message used above, and obtains the text ciphertext σ_s. It then hands σ_e to $\mathcal{Z}$ as the signcryption of m_s.

(7) For the remaining $p - s$ times that $\mathcal{Z}$ asks to signcrypt some message, m_i, $\mathcal{B}$ lets the signcrypting party return $\sigma_i = Signcrypt(m_f)$.

(8) Whenever the unsigncrypting party P_i is activated with input $(Unsigncrypt, sid, ID_s, ID_i, \sigma)$, where ID_s is the identity of signcrypting party and $\sigma = \sigma_i$ for some i, $\mathcal{B}$ lets P_i return the corresponding plaintext m_i. If σ is different from all the σ_i's then $\mathcal{B}$ queries its unsigncryption oracle $\mathcal{O}_U$ on σ, obtains a value v, and lets P_i return v to $\mathcal{Z}$.

(9) When $\mathcal{Z}$ halts, $\mathcal{B}$ outputs whatever $\mathcal{Z}$ outputs and halts.

Here, assume that for some value of the security parameter k we have $EXEC_{\mathcal{F}_{IDSC},\mathcal{S},\mathcal{Z}}(1^k) - EXEC_{\pi_{IDSC},\mathcal{D},\mathcal{Z}}(1^k) > \epsilon$. Analyzing the success probability of $\mathcal{B}$ is done via a standard hybrid argument. Let the random variable X_i denote the output of $\mathcal{Z}$ from an interaction that is identical to an interaction with $\mathcal{S}$ in the ideal process. It is easy to see that X_0 and X_{q+1} are statistically close to the output of $\mathcal{Z}$ in the ideal process, and X_q and X_{q+p} is identical the output of $\mathcal{Z}$. Thus we can figure out that $\mathcal{B}$ guesses the bit b correctly in the IND-IDSC-CCA2 experiment with probability $1/2 + \epsilon(p + q)/2pq$, and the experiment $Exp^{EXT-CMA}_{IDSC,\mathcal{A}}(1^k)$ return 1 with probability $1/2 + \epsilon/2q$. This shows that if IDSC is secure with respect to both IND-IDSC-CCA2 and EXT-IDSC-CMA then π_{IDSC} securely realizes $\mathcal{F}_{IDSC}$.

5 Securely realizing $\mathcal{F}_{SAGCOM}$

In this section, we propose a universally composable secure authenticated group communication scheme π_{SAGCOM} and prove that it realizes the ideal functional $\mathcal{F}_{SAGCOM}$ in $(\mathcal{F}_{IDSC}, \mathcal{F}_{GKD})$-hybrid model.

5.1 System description

We consider the network environment where there are a centralized key server and multiple multicast controllers. A sender of a group $G = \{G_1, \cdots, G_m\}$ is in charge of managing the group as a multicast controller. Let $R = \{R_1, \cdots, R_n\}$ be the universe of users. The key server as KGC generates public parameters for the system and the keys for the network group senders and users.

5.2 Protocol, π_{SAGCOM}

We propose our group communication protocol π_{SAGCOM} in the $(\mathcal{F}_{IDSC}, \mathcal{F}_{GKD})$-hybrid model. The detailed description of π_{SAGCOM} is shown in Figure 5.

5.3 Security proof of π_{SAGCOM}

Theorem 3. *Protocol π_{SAGCOM} securely realizes the ideal functionality $\mathcal{F}_{SAGCOM}$ in the $(\mathcal{F}_{IDSC}, \mathcal{F}_{GKD})$-hybrid model.*

Proof. Let $\mathcal{A}$ be an adversary that interacts with the parties running the protocol π_{SAGCOM}. We construct an ideal adversary $\mathcal{S}$ such that any environment $\mathcal{Z}$ cannot distinguish with a non-negligible probability whether it is interacting $\mathcal{A}$ and π_{SAGCOM} in the $(\mathcal{F}_{IDSC}, \mathcal{F}_{GKD})$-hybrid model (denoted $REAL$) or it is interacting with $\mathcal{S}$ and $\mathcal{F}_{SAGCOM}$ in the ideal world (denoted $IDEAL$).

Construction of the ideal adversary $\mathcal{S}$. The adversary $\mathcal{S}$ shown below runs a simulated copy of the adversary $\mathcal{A}$, thus $\mathcal{S}$ is often called a simulator. Any input from $\mathcal{Z}$ is forwarded to $\mathcal{A}$ and any output of $\mathcal{A}$ is copied to the output of $\mathcal{S}$.

1. Simulating the KGC. When an uncorrupted KGC is activated with input $(SAGCOM.Setup, sid)$, $\mathcal{S}$ obtains this value from $\mathcal{F}_{SAGCOM}$ and simulates for $\mathcal{A}$ the protocol π_{SAGCOM}.

(1) Whenever $\mathcal{S}$ obtains $(IDSC.Setup, sid)$ from $\mathcal{F}_{IDSC}$, $\mathcal{S}$ sends the message $(IDSC.Setup, sid)$ to $\mathcal{A}$, then forwards the response from $\mathcal{A}$ to $\mathcal{F}_{IDSC}$.

Protocol π_{SAGCOM}

π_{SAGCOM} proceeds as follows, with a group of senders $G = \{G_1, \cdots, G_m\}$, a group of recipients $R = \{R_1, \cdots, R_n\}$ and KGC.

SAGCOM.Setup

Upon the first input $(SAGCOM.Setup, sid)$, KGC runs the algorithms Setup:

- The KGC obtains two key pairs (s_G, P_G) and (s_R, P_R), and
- outputs $(SAGCOM.Setup, sid, (s_G, P_G), (s_R, P_R))$.

SAGCOM.Extract

Upon the first input $(SAGCOM.Extract, sid, ID)$, $G_i \in G$ (resp. $R_i \in R$) runs the algorithms Extract:

- The KGC sends $(GKD.Distribute, sid, s_G)$ (resp. $(GKD.Distribute, sid, s_R)$) to $\mathcal{F}_{GKD}$,
- P_i (resp. $R_i \in R$) obtains his private key s_{G_i} (resp. s_{R_i}) from $\mathcal{F}_{GKD}$, and
- outputs $(SAGCOM.Extract, sid, (s_{G_i}, ID_{G_i}))$ (resp. $(SAGCOM.Extract, sid, (s_{R_i}, ID_{R_i})))$.

SAGCOM.Signcrypt

Upon input $(SAGCOM.Signcrypt, sid, P_R, ID_R, m)$, $G_i \in G$ runs the algorithms Signcrypt:

- $G_i \in G$ obtains $\sigma_i = Signcrypt(s_{G_i}, P_R, ID_R, m)$,
- $G_i \in G$ sends σ_i to R, and
- outputs $(SAGCOM.Signcrypt.Ciphertext, sid, ID_R, m, \sigma_i)$.

SAGCOM.Unsigncrypt

Upon input $(SAGCOM.Unsigncrypt, sid, P_G, ID_G, \sigma)$, $R_i \in R$ runs the algorithms Unsigncrypt:

- Group signcryption generating phase
  1. R_i sends $(GKD.Recover, sid)$ to $\mathcal{F}_{GKD}$,
  2. R_i obtains σ, and
  3. outputs $(GKD.recover, sid, \sigma)$.
- Group signcryption verifying phase
  1. R_i sends $(GKD.Recover, sid)$ to $\mathcal{F}_{GKD}$,
  2. R_i obtains $m/\bot = Unsigncrypt(ID_G, P_G, s_{R_T}, \sigma)$ with $T \in \mathcal{T}$, and
  3. outputs $(SAGCOM.Unsigncrypt.Plaintext, sid, m/\bot, \sigma)$.

Fig. 5. The universally composable group communication protocol, π_{SAGCOM}

(2) Whenever $\mathcal{S}$ obtains $(IDSC.Extract, sid, ID)$ from $\mathcal{F}_{IDSC}$, $\mathcal{S}$ sends the message $(IDSC.Extract, sid, ID)$ to $\mathcal{A}$, then forwards the response from $\mathcal{A}$ to $\mathcal{F}_{IDSC}$.

(3) Whenever $\mathcal{S}$ receives $(GKD.Distribute, sid, |s|)$ from $\mathcal{F}_{GKD}$, $\mathcal{S}$ sends the message $(GKD.Distribute, sid, |s|)$ to $\mathcal{A}$, then forwards the response from $\mathcal{A}$ to $\mathcal{F}_{GKD}$.

2. Simulating the sender. When an uncorrupted party S_i is activated with input $(Send, sid, T, m)$, where $S_i \in B \subset S$ and $T \subset R$, $\mathcal{S}$ obtains this value from $\mathcal{F}_{SAGCOM}$ and simulates for $\mathcal{A}$ the protocol π_{SAGCOM}.

(1) Whenever $\mathcal{S}$ obtains $(Send, sid, T, l(m))$ from $\mathcal{F}_{SAGCOM}$, $\mathcal{S}$ sends the message $(Send, sid, T, l(m))$ to $\mathcal{A}$, then forwards the response from $\mathcal{A}$ to $\mathcal{F}_{SAGCOM}$.

(2) Whenever $\mathcal{S}$ receives $(IDSC.Extract, sid, PK'_s)$ from $\mathcal{F}_{IDSC}$, $\mathcal{S}$ sends the message $(IDSC.Extract, sid, PK'_s)$ to $\mathcal{A}$, then forwards the response $IDSC.Received, sid$ from $\mathcal{A}$ to $\mathcal{F}_{IDSC}$.

(3) Whenever $\mathcal{S}$ receives $(IDSC.Signcrypt, sid, ID_i, ID_r, PK_s, |m|)$ from $\mathcal{F}_{IDSC}$, $\mathcal{S}$ sends the message $(IDSC.Signcrypt, sid, ID_i, ID_r, PK_s, |m|)$ to $\mathcal{A}$, then forwards the response $(IDSC.Signcrypt.Ciphertext, sid, ID_i, ID_r, PK_s, \sigma)$ from $\mathcal{A}$ to $\mathcal{F}_{IDSC}$.

3. Simulating the receiver. When an uncorrupted party R_i is activated with input $(Receive, sid, B, m)$, where $R_i \in T \subset R$ and $B \subset S$, $\mathcal{S}$ obtains this value from $\mathcal{F}_{SAGCOM}$ and simulates for $\mathcal{A}$ the protocol π_{SAGCOM}.

(1) Whenever $\mathcal{S}$ receives $(IDSC.Extract, sid, PK'_s)$ from $\mathcal{F}_{IDSC}$, $\mathcal{S}$ sends the message $(IDSC.Extract, sid, PK'_s)$ to $\mathcal{A}$, then forwards the response $IDSC.Received, sid$ from $\mathcal{A}$ to $\mathcal{F}_{IDSC}$.

(2) Whenever $\mathcal{S}$ receives $(IDSC.Unigncrypt, sid, ID_s, ID_j, PK_s, \sigma)$ from $\mathcal{F}_{IDSC}$, $\mathcal{S}$ sends the message $(IDSC.Unigncrypt, sid, ID_s, ID_j, PK_s, \sigma)$ to $\mathcal{A}$, then forwards the response $(IDSC.Unigncrypt.Plaintext, sid, ID_s, ID_j, PK_s, m'/\perp, \sigma)$ from $\mathcal{A}$ to $\mathcal{F}_{IDSC}$.

(3) Whenever $\mathcal{S}$ obtains $(Receive, sid, B, l(m))$ from $\mathcal{F}_{SAGCOM}$, $\mathcal{S}$ sends the message $(Receive, sid, B, l(m))$ to $\mathcal{A}$, then forwards the response $(Received, sid, ok)$ from $\mathcal{A}$ to $\mathcal{F}_{SAGCOM}$.

3. Simulating party corruption. Whenever $\mathcal{A}$ corrupts a party, $\mathcal{S}$ corrupts the same party and provides $\mathcal{A}$ with the internal state of the corrupted party. This poses on problem since none of the parties maintains any secret information.

$IDEAL$ **and** $REAL$ **are indistinguishable.** Based on our construction of $\mathcal{S}$, we define three events and show that $REAL$ and $IDEAL$ are indistinguishable no matter which one of the three events happens.

Event 1: When a party $P_i \in S \cup R$ is corrupted, it is easily observable that $\mathcal{S}$ can perfectly simulate the operations of protocols in $REAL$ according to the logics of $\mathcal{F}_{IDSC}$, $\mathcal{F}_{GKD}$ and $\mathcal{F}_{SAGCOM}$. Thus, in this case, $REAL$ and $IDEAL$ are indistinguishable.

Event 2: When some parties $C \subset S \cup R$ are corrupted, if $C \in \mathcal{T}$ then the adversary learns the plaintex m of the ciphertext σ; Else the adversary can not learn the plaintex m of the ciphertext σ. According to $\mathcal{F}_{GKD}$, obviously, $\mathcal{S}$ can perfectly simulate the operations of protocols in $REAL$. Therefore, $REAL$ and $IDEAL$ are indistinguishable.

Event 3: The receiver R_i obtains $(Received, sid, ok)$ from $\mathcal{F}_{SAGCOM}$ for an incoming message $(Send, sid, T, m)$, where $R_i \in T$, while party S_i is uncorrupted at the time when the message is delivered, and has never sent $(Send, sid, T, m)$. However, according to the protocol and the logics of $\mathcal{F}_{IDSC}$, $\mathcal{F}_{GKD}$ and $\mathcal{F}_{SAGCOM}$. The reason being is that, firstly the receiver should obtain a valid signcrypted ciphertext from $\mathcal{F}_{IDSC}$ since the protocol $IDSC$ is IND-IDSC-CCA2 and EXT-IDSC-CMA ; secondly, if an uncorrupted S_i never sent $(Send, sid, T, m)$, then the message m is never signcrypted by $\mathcal{F}_{IDSC}$. Thus, R_i would obtain $(Received, sid, ok)$ from $\mathcal{F}_{SAGCOM}$.

The above analysis shows that π_{SAGCOM} securely realizes the functionality $\mathcal{F}_{SAGCOM}$ in the $(\mathcal{F}_{IDSC}, \mathcal{F}_{GKD})$-hybrid model.

6 Conclusion

We have proposed the first UC frame for secure authenticated group communication model. We have designed the identity-based signcryption functionality $\mathcal{F}_{IDSC}$, the group key distribution functional $\mathcal{F}_{GKD}$ and group communication functionality $\mathcal{F}_{SAGCOM}$ and subsequently, proposed a UC-secure group communication scheme that realizes $\mathcal{F}_{SAGCOM}$ under the $(\mathcal{F}_{IDSC}, \mathcal{F}_{GKD})$-hybrid model. We have also analyzed the security of protocol π_{IDSC} under UC framework, which an identity-based signcryption

π_{IDSC} to realize the ideal functionality $\mathcal{F}_{IDSC}$ is proved equals to the IDSC being both IND-IDSC-CCA2 and EXT-IDSC-CMA.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Nos. 61170280, 61272398, 61262073, and 61303068; China Postdoctoral Science Foundation under Grant No. 2013M530705; The Natural Science Foundation of Guizhou of China under Grant No. 20132112; the Doctor Foundation of the Guizhou University under Grant No. 2012-024.

References

1. Canetti R. Universally composable security: A new paradigm for cryptographic protocols. A revised version (2005) is available at IACR Eprint Archive, http://eprint.iacr.org/2000/067.
2. Fiat A, Naor M. Broadcast encryption, in: CRYPTO 1993, in: LNCS, vol. 773, 1993, pp. 480-491.
3. Malone Lee J. Identity-Based Signcrytion. Available online: http://www.signcryption.org/ publications/pdffiles/ MaloneLee-eprint2002-098.pdf.
4. Wang L, Wu C.-K, Efficient identity-based multicast scheme from bilinear pairing, IEE Proceedings. Communications, 152(6), (2005), pp. 877-882.
5. Lin X.-J, Wu C.-K, Liu F. Analysis of an authenticated identity-based multicast scheme, I-ET Communications, 2(7), (2008), pp. 935-937.
6. Mu Y, Susilo W, Lin W.-X, Ruan C. Identity-based authenticated broadcast encryption and distributed authenticated encryption, in: ASIAN, in: LNCS, vol. 3321, 2004, pp. 169-181.
7. Hur J, Park C and Yoon H. Chosen ciphertext secure authenticated group communication using identity-based signcryption. Computers and Mathematics with Applications, Vol. 60, Issue 2, 2010, pp. 362-375.
8. Zhang F, Ma J F, Moon S J. Universally composable anonymous Hash certification model. Sci China Ser F-Inf Sci, 2007, 50: 440C455.
9. Feng T, Li F H, Ma J F, et al. A new approach for UC security concurrent deniable authentication. Sci China Ser F-Inf Sci, 2008, 51: 352C367.
10. Zhang J W, Ma J F, Moon S J. Universally composable secure TNC model and EAP-TNC protocol in IF-T. Sci China Inf Sci, 2010, 53: 465-482.
11. Zhang J W, Ma J F, Moon S J. Universally composable one-time signature and broadcast authentication. Sci China Inf Sci, 2010, 53: 567-580.