IJNS
International Journal of Network Security

ISSN 1816-353X (Print)
ISSN 1816-3548 (Online)

Vol. 17, No. 2 (Mar. 2015)

INTERNATIONAL JOURNAL OF NETWORK SECURITY

PUBLISHING OFFICE

Min-Shiang Hwang
Department of Computer Science & Information Engineering, Asia University, Taichung 41354, Taiwan, R.O.C.
Email: mshwang@asia.edu.tw

International Journal of Network Security is published both in traditional paper form (ISSN 1816-353X) and in Internet (ISSN 1816-3548) at http://ijns.jalaxy.com.tw

PUBLISHER: Candy C. H. Lin

International Journal of Network Security

Vol. 17, No. 2 (Mar. 1, 2015)

Improvement of Camenisch-Neven-Shelat Oblivious Transfer Scheme

Zhengjun Cao and Hanyue Cao
(Corresponding author: Zhengjun Cao)

Department of Mathematics, Shanghai University
No.99, Shangda Road, Shanghai, China.
(Email: caozhj@shu.edu.cn)
(Received Aug. 15, 2013; revised and accepted Nov. 7, 2014)

Abstract

In 2007, Camenisch, Neven and Shelat proposed an adaptive oblivious transfer (OT) scheme in which a sender has n messages, of which a receiver can adaptively choose to receive k one-after-the-other. In this paper, we show that the scheme has a drawback that the sender can serve a single receiver only once. The drawback results from the deterministic encryption used. To fix it, we suggest to replace the deterministic encryption with a probabilistic encryption. The OT scheme adopts the paradigm of "encryption and proof of knowledge" in order to force the sender to keep the consistency of the transferred messages. We remark that the paradigm is unnecessary. In most reasonable applications of OT, the transferred messages must be recognizable for the receiver or the sender is willing to disclose some messages to the receiver. This property has been explicitly specified in the earlier works by Rabin, Even, Goldreich and Lempel.

Keywords: Oblivious transfer, deterministic encryption, probabilistic encryption, recognizable message

1 Introduction

The cryptographic primitive of oblivious transfer (OT) introduced by Rabin [25], is of fundamental importance in multi-party computation [12, 28]. In the model, a particpator (sender S) has only one secret m and would like to have the other participator (receiver R) to obtain m with probability 0.5. On the other hand, R does not want S to know whether it gets m or not.

There are two main OT models: 1-out-of-2 oblivious transfer (OT_1^2 for short) and k-out-of-n oblivious transfer (OT_k^n for short). OT_1^2 was suggested by Even, Goldreich and Lempel [11], as a generalization of Rabin's "oblivious transfer". For OT_1^2, the sender has two secrets m_1 and m_2 and would like to give the receiver one of them at the receiver's choice. Meanwhile, the receiver does not want the sender to know which secret he chooses. OT_k^n is a generalization of OT_1^2 where $k < n$. In the model, the sender has n secrets $m_1, \cdots, m_n$, and would like to give the receiver k of them at the receiver's choice. Again, the receiver does not want the sender to know which secrets he chooses.

In an adaptive oblivious transfer protocol, a sender commits to a database of messages and then repeatedly interacts with a receiver in such a way that the receiver obtains one message per interaction of his choice (and nothing more) while the sender learns nothing about any of the choices. At Eurocrypt'2007, Camenisch, Neven and Shelat [5] presented an adaptive oblivious transfer scheme in which a sender has n messages, of which a receiver can adaptively choose to receive k one-after-the-other. They were the first to propose a method for executing "assisted decryption" efficiently. In the scheme, the sender commits to his database by encrypting each message as $C_i = \mathrm{Enc}(M_i)$, and sends ciphertexts $C_1, \cdots, C_n$ to the receiver. The receiver then checks that each ciphertext is well-formed. To obtain a message, the sender and receiver engage in a blind decryption protocol such that the sender does not view the ciphertext he decrypts and the receiver is convinced that decryption was done correctly. To prevent the receiver from abusing the decryption protocol, the receiver has to provide a proof that his request corresponds to $C_1 \vee \cdots \vee C_n$.

The encryption used in the scheme is deterministic. Concretely, for $pk = (g, g^x, H = e(g,h))$ and $sk = h$, let $C_i = \left(g^{\frac{1}{x+i}}, M_i \cdot e(g,h)^{\frac{1}{x+i}}\right)$, where $g^{\frac{1}{x+i}}$ is a weak Boneh-Boyen signature [3] on i under g^x. The structure results in that a database manager (the sender) can only serve a single user (the receiver). Moreover, the protocol can be run only once even in the presence of a single user. In this paper, we shall improve the Camenisch-Neven-Shelat OT scheme by replacing the deterministic encryption with a probabilistic encryption.

The OT scheme follows the paradigm of "encryption and proof of knowledge" to force the sender to keep the consistency of the transferred messages. We should stress

that the paradigm is unnecessary for OT protocols. That means the sender can simply transfer the encrypted messages without any proofs of knowledge. The property has been explained in the earlier works by Rabin [25], Even, Goldreich and Lempel [11]. Based on the observation, we can further improve the Camenisch-Neven-Shelat OT scheme by removing the computations for some proofs of knowledge.

1.1 Related Works

In 1986, Brassard, et al. [4] extended 1-out-of-2 OT to 1-out-of-n OT for the case of n messages. Bellare and Micali [1], Naor and Pinkas [22, 23, 24], Mu, Zhang, and Varadharajan [21], Chu and Tzeng [9], et al. have studied the modle of k-out-of-n OT. Recently, Chang and Lai [7], Chang and Lee [6], and Liu et al. [8, 15, 19, 26, 27, 29] have presented some efficient OT_k^n schemes.

In 1999, Naor and Pinkas [23] investigated the problem of oblivious transfer with adaptive queries. Their scheme has inspired the latter works [2, 5, 9, 13, 14, 16, 17, 18, 20, 30]. In 2007, Camenisch, Neven and Shelat [5] proposed an adaptive oblivious transfer scheme. The Camenisch-Neven-Shelat scheme uses bilinear groups as the building block and adopts the paradigm of "encryption and proof of knowledge" to force the sender to keep the consistency of the transferred messages. The paradigm has been used in these OT protocols [13, 14, 17, 18, 30].

1.2 Security Requirements for k-out-of-n Oblivious Transfer

We follow the description of security requirements for k-out-of-n oblivious transfer in the work of Chang and Lai [7].

Definition 1. *A k-out-of-n OT is a two-party protocol in which Alice possesses n secrets $m_1, m_2, \cdots, m_n$ and Bob has his secret choices $\sigma = \{i_1, \cdots, i_k\} \subseteq 1, \cdots, n$. It satisfies the following requirements:*

- *Completeness: If both Alice and Bob follow the protocol, Bob gets k secrets m_j for $j \in \sigma$ after executing the protocol with Alice.*
- *Receiver's privacy: After executing the protocol with Bob, Alice shall not learn which k secrets Bob has received.*
- *Sender's privacy: After executing the protocol with Alice, Bob gets no information about the other $n-k$ secrets m_j for $j \notin \sigma$ or their combinations.*

An adaptive k-out-of-n OT scheme is a tuple of four PPT algorithms $(\mathsf{S}_I, \mathsf{R}_I, \mathsf{S}_T, \mathsf{R}_T)$. During the first phase, the sender runs S_I on input messages $m_1, \cdots, m_n$ and the receiver runs R_I without input. At the end of the phase, S_I and R_I produce local outputs S_0 and R_0, respectively. During the i-th transfer, $1 \le i \le k$, the sender and receiver engage in a selection protocol dictated by the S_T and R_T. The sender runs $\mathsf{S}_T(S_{i-1})$ to obtain updated state information S_i, while the receiver runs R_T on input state information R_{i-1} and the index σ_i of the message it wishes to receive, to obtain updated state information R_i and the retrieved message m'_{σ_i}. To capture security of an adaptive k-out-of-n OT scheme, we adopt the real-world/ideal-world paradigm [5].

$\mathrm{Real}_{\hat{S},\hat{R}}(N, k, M_1, \cdots, M_N, \sum)$. Suppose $\hat{S}$ and $\hat{R}$ are arbitrary sender and receiver algorithms. $\hat{S}$ is given messages $(M_1, \cdots, M_N)$ as input and interacts with $\hat{R}(\sum)$, where $\sum$ is an adaptive selection algorithm that, on input messages $M_{\sigma_1}, \cdots, M_{\sigma_{i-1}}$, outputs the index σ_i of the next message to be queried. In the first run, $\hat{S}$ and $\hat{R}$ produce initial states S_0 and R_0 respectively. Next, the sender and receiver engage in k interactions. In the i-th interaction for $1 \le i \le k$, the sender and receiver interact by running $S_i \leftarrow \hat{S}(S_{i-1})$ and $(R_i, M_i^*) \leftarrow \hat{R}(R_{i-1})$, and update their states to S_i and R_i, respectively. At the end of the k-th interaction, sender and receiver output strings S_k and R_k respectively.

$\mathrm{Ideal}_{\hat{S}',\hat{R}'}(N, k, M_1, \cdots, M_N, \sum)$. The (possibly cheating) sender algorithm $\hat{S}'(M_1, \cdots, M_N)$ generates messages $M_1^*, \cdots, M_N^*$ and hands these to the trusted party T. In each of the k transfer phases, T receives a bit b_i from the sender $\hat{S}'$ and an index σ_i^* from the (possibly cheating) receiver $\hat{R}'(\sum)$. If $b_i = 1$ and $\sigma_i^* \in \{1, \cdots, N\}$, then T hands $M_{\sigma_i^*}^*$ to the receiver; otherwise, it hands $\perp$ to the receiver. At the end of the k-th transfer, $\hat{S}'$ and $\hat{R}'$ output a string S_k and R_k.

An adaptive k-out-of-n OT scheme is sender-secure if for any PPT real-world cheating receiver $\hat{R}$ there exists a PPT ideal-world receiver $\hat{R}'$ such that the advantage of any PPT distinguisher in distinguishing the distributions $\mathrm{Real}_{S,\hat{R}}(N, k, M_1, \cdots, M_N, \sum)$ and $\mathrm{Ideal}_{S',\hat{R}'}(N, k, M_1, \cdots, M_N, \sum)$ is negligible. It is receiver-secure if for any PPT real-world cheating sender $\hat{S}$ there exists a PPT ideal-world sender $\hat{S}'$ such that the advantage of any PPT distinguisher in distinguishing the distributions $\mathrm{Real}_{\hat{S},R}(N, k, M_1, \cdots, M_N, \sum)$ and $\mathrm{Ideal}_{\hat{S}',R'}(N, k, M_1, \cdots, M_N, \sum)$ is negligible.

2 Preliminaries

Let Pg be a pairing group generator that on input 1^κ outputs descriptions of multiplicative groups $\mathbb{G}_1, \mathbb{G}_T$ of prime order p where $|p| = \kappa$. Let g be a generator of $\mathbb{G}_1$. The bilinear map $e : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_T$ satisfies: (1) for all $a, b \in \mathbb{Z}_p$ it holds that $e(g^a, g^b) = e(g, g)^{ab}$; (2) $e(g, g) \neq 1$; (3) the bilinear map is efficiently computable.

The notation

$$PoM\{(h) : H = e(g,\, h) \wedge W = e(h,\, V)\}$$

denotes a zero-knowledge proof of membership of a group element $h \in \mathbb{G}_1$ such that $H = e(g, h)$ and $W = e(h, V)$ hold. All values not enclosed in ()'s are assumed to be

known to the verifier. Likewise,

$$PoK\{(x,h): y = g^x \wedge H = e(y,h)\}$$

denotes a zero-knowledge proof of knowledge of an integer x and a group element $h \in \mathbb{G}_1$ such that $y = g^x$ and $H = e(y,h)$ hold.

Definition 2. *(ℓ-Strong Diffie-Hellman Assumption). We say that the ℓ-SDH assumption associated to a pairing generator Pg holds if for all PPT adversaries A, the probability that $A(g, g^x, \cdots, g^{x^\ell})$ where $(\mathbb{G}_1, \mathbb{G}_T) \leftarrow \mathsf{Pg}(1^\kappa), g \leftarrow \mathsf{G}_1^*$ and $x \leftarrow \mathbb{Z}_p$, outputs a pair $(c, g^{1/(x+c)})$ where $c \in \mathbb{Z}_p$ is negligible in κ.*

Definition 3. *(ℓ-Power Decisional Diffie-Hellman Assumption). We say that the ℓ–PDDH assumption associated to Pg holds if for all PPT adversaries A, the probability that A on input $(g, g^x, g^{x^2}, \cdots, g^{x^\ell}, H)$ where $(\mathbb{G}_1, \mathbb{G}_T) \leftarrow \mathsf{Pg}(1^\kappa), g \leftarrow \mathbb{G}_1^*, x \leftarrow \mathbb{Z}_p, H \leftarrow \mathbb{G}_T$, distinguishes the vector $T = (H^x, H^{x^2}, \cdots, H^{x^\ell})$ from a random vector $T \leftarrow \mathbb{G}_T^\ell$ is negligible in κ.*

3 Camenisch-Neven-Shelat Oblivious Transfer Scheme

3.1 Review

The protocol is in the standard model. See the following Table 1 for details. Each pair (A_i, B_i) can be seen as an ElGamal encryption [10] in $\mathbb{G}_T$ of M_i under public key H. But instead of using random elements from $\mathbb{G}_T$ as the first component, the protocol uses verifiably random values $A_i = g^{1/(x+i)}$. It allows the sender to check that the receiver is indeed asking for the decryption key for one particular ciphertext, and not for some combination of ciphertexts.

3.2 A Weakness

The encryption used in the scheme is deterministic. Concretely, for

$$pk = (g, g^x, H = e(g,h))$$

and $sk = h$, let

$$C_i = \left(g^{\frac{1}{x+i}}, M_i \cdot e(g,h)^{\frac{1}{x+i}}\right)$$

where $g^{\frac{1}{x+i}}$ is a weak Boneh-Boyen signature [3] on i under g^x. In view of that a database manager usually plays the role of the sender in an OT protocol, the structure results in that a database manager can only serve a single client only once.

Suppose that $N > 2k$ and there are two users $\mathcal{R}, \hat{\mathcal{R}}$. $\mathcal{R}$ has the ciphertexts $C_1, \cdots, C_N$ and $\hat{\mathcal{R}}$ has the ciphertexts $\hat{C}_1, \cdots, \hat{C}_N$, where

$$C_i = \left(g^{\frac{1}{x+i}}, M_i \cdot e(g,h)^{\frac{1}{x+i}}\right),$$

$$\hat{C}_i = \left(g^{\frac{1}{x+i}}, \hat{M}_i \cdot e(g,h)^{\frac{1}{x+i}}\right)$$

for $1 \le i \le N$. At the end of the two OT protocols executed by the sender, $\mathcal{R}$ and $\hat{\mathcal{R}}$, if $\mathcal{R}$ obtains $M_1, \cdots, M_k$, and $\hat{\mathcal{R}}$ obtains $\hat{M}_{k+1}, \cdots, \hat{M}_{2k}$, then $\mathcal{R}$ and $\hat{\mathcal{R}}$ can collaborate to obtain $M_{k+1}, \cdots, M_{2k}$, $\hat{M}_1, \cdots, \hat{M}_k$. Thus, they obtain $4k$ messages instead of $2k$ messages as usually supposed. In other words, the protocol can be run only once even in the presence of a single user. The drawback results from that the scheme invariably uses N blinders

$$e(g,h)^{\frac{1}{x+1}}, \cdots, e(g,h)^{\frac{1}{x+N}}.$$

We refer to the attack as *session key attack*.

4 A Modification of Camenisch-Neven-Shelat OT Scheme

In the original Camenisch-Neven-Shelat oblivious transfer scheme, the public key is set as (g, H, y), where $y = g^x$. The receiver has to use the public parameter y for the proof of knowledge (σ_i, v), i.e.,

$$PoK\{(\sigma_i, v): e(V, y) = e(V, g)^{-\sigma_i} e(g, g)^v\}.$$

The setting allows the sender to check that the receiver does not ask for some combination of ciphertexts. That is, it makes the sender believe that the queries from the receiver are well-formed. But *it is unnecessary to set y as a public parameter*. It only requires to *set y as a session helper* with respect to the session key x. The authors did not pay more attention to the differences between a public parameter and a session helper. Informally, a public parameter should be used repeatedly except that it has to be authorized by a functionally trusted TTP (trusted third party). Whereas, a session helper can only be used once. The change, removing the public parameter y and introducing a session helper y, successfully transforms the deterministic encryption into a probabilistic encryption. See the following Table 2 for details.

Theorem 1. *If the (N+1)-SDH assumption and the (N+1)-PDDH assumptions associated to Pg hold, then the OT protocol in Table 2 is sender-secure.*

Theorem 2. *The OT protocol in Table 2 is receiver-secure if the transferred messages are recognizable for the receiver.*

We refer to [5] for the proofs of these claims. It suffices to transform the public parameter y into a session helper and transform the associated signatures $A_1, \cdots, A_N$ in the related Games into knowledge proofs (see Pages 15-16 in [5]).

Note that the original proof of receiver-security does not consider that a malicious sender can launch the local-input replacement attack. That is, the sender simply sets $M_1 = M_2 = \cdots = M_N = M^{(i)}$ for some message $M^{(i)}$ during the i-th transfer. At the end of this phase, the receiver always obtains the message $M^{(i)}$. Of course, the

Table 1: Camenisch-Neven-Shelat oblivious transfer scheme

	Initialization	
$\underline{\mathsf{S}_I(1^\ell, M_1, \cdots, M_N):}$		$\underline{\mathsf{R}_I(1^\ell):}$
$(G_1, G_T) \leftarrow \mathsf{Pg}(1^\ell)$		
$g, h \leftarrow \mathbb{G}_1^*; H \leftarrow e(g,h)$		
$x \leftarrow Z_p; y \leftarrow g^x;$		
$pk \leftarrow (g, H, y)$		
For $i = 1, \cdots, N$ do		
$A_i \leftarrow g^{1/(x+i)}$		
$B_i \leftarrow e(h, A_i) \cdot M_i$		
$C_i \leftarrow (A_i, B_i)$	$\overset{pk, C_1, \cdots, C_N}{- - - - - - - \dashrightarrow}$	
$S_0 \leftarrow (h, pk)$	$\overset{PoM\{(h): H = e(g,h)\}}{- - - - - - - \dashrightarrow}$	$R_0 \leftarrow (pk, C_1, \cdots, C_N)$
	Transfer	
$\mathsf{S}_\mathsf{T}(S_{i-1}):$		$\mathsf{R}_\mathsf{T}(R_{i-1}, \sigma_i):$
	$\overset{V}{\dashleftarrow - - - - - - - - - - - - - -}$	$v \leftarrow \mathbb{Z}_p; V \leftarrow (A_{\sigma_i})^v$
	$\overset{PoK\{(\sigma_i, v): e(V,y) = e(V,g)^{-\sigma_i}\, e(g,g)^v\}}{\dashleftarrow - - - - - - - - - - - - - -}$	
$W \leftarrow e(h, V)$	$\overset{W}{- - - - - - - - - - - - \dashrightarrow}$	
	$\overset{PoM\{(h): H = e(g,h) \wedge W = e(h,V)\}}{- - - - - - - - - - - - \dashrightarrow}$	
$S_i = S_{i-1}$		$M \leftarrow B_{\sigma_i}/(W^{1/v})$
		$R_i = R_{i-1}$

Table 2: A modification of Camenisch-Neven-Shelat OT scheme

	Setup	
$(G_1, G_T) \leftarrow \mathsf{Pg}(1^\ell)$		
$g, h \leftarrow \mathbb{G}_1^*;\ H = e(g,h)$		
$pk \leftarrow (g, H);\ sk \leftarrow h$		
	Transfer	
$\underline{\mathsf{S}_I(1^\ell, M_1, \cdots, M_N):}$		$\underline{\mathsf{R}_I(1^\ell):}$
$x \leftarrow Z_p; y \leftarrow g^x$		
For $i = 1, \cdots, N$ do		
$A_i \leftarrow g^{1/(x+i)}$		
$B_i \leftarrow e(h, A_i) \cdot M_i$		
$C_i \leftarrow (A_i, B_i)$	$\overset{pk, y, C_1, \cdots, C_N}{- - - - - - - \dashrightarrow}$	
$S_0 \leftarrow (h, pk)$	$\overset{PoM\{(h): H = e(g,h)\}}{- - - - - - - \dashrightarrow}$	$R_0 \leftarrow (pk, C_1, \cdots, C_N)$
$\mathsf{S}_\mathsf{T}(S_{i-1}):$		$\mathsf{R}_\mathsf{T}(R_{i-1}, \sigma_i):$
	$\overset{V}{\dashleftarrow - - - - - - - - - - - - - -}$	$v \leftarrow \mathbb{Z}_p; V \leftarrow (A_{\sigma_i})^v$
	$\overset{PoK\{(\sigma_i, v): e(V,y) = e(V,g)^{-\sigma_i}\, e(g,g)^v\}}{\dashleftarrow - - - - - - - - - - - - - -}$	
$W \leftarrow e(h, V)$	$\overset{W}{- - - - - - - - - - - - \dashrightarrow}$	$M_{\sigma_i} \leftarrow B_{\sigma_i}/(W^{1/v})$
	$\overset{PoM\{(h): H = e(g,h) \wedge W = e(h,V)\}}{- - - - - - - - - - - - \dashrightarrow}$	
$S_i = S_{i-1}$		$R_i = R_{i-1}$

malicious sender learns which message that the receiver has received. To resist the local-input replacement attack, the transferred messages in OT schemes must be recognizable for the receiver. See the following section for the further explanations.

5 On the Paradigm of "Encryption and Proof of Knowledge"

The Camenisch-Neven-Shelat oblivious transfer scheme follows the paradigm of "encryption and proof of knowledge" to force the sender to keep the consistency of the committed messages. From the practical point of view, we should remark that the paradigm is unnecessary. In most reasonable applications of OT, *the transferred messages must be recognizable for the receiver*, or *the sender is willing to disclose some messages to the receiver*. The property has been explicitly specified in the earlier works by Rabin, Even, Goldreich and Lempel. We refer to the following descriptions.

In [25], Rabin explained that:

> Bob and Alice each have a secret, SB and SA, respectively, which they wish to exchange. For example, SB may be the password to a file that Alice wants to access (we shall refer to this file as Alice's file), and SA the password to Bob's file. To exclude the possibility of randomizing on the possible digits of the password, we assume that if an incorrect password is used then the file is erased, and that Bob and Alice want to guarantee that this will not happen to their respective files.

In [11], Even, Goldreich and Lempel stressed that:

> The notion of a "recognizable secret message" plays an important role in our definition of OT. A message is said to be a recognizable secret if, although the receiver cannot compute it, he can authenticate it once he receives it. The notion of a recognizable secret message is evidently relevant to the study of cryptographic protocols, in which the sender is reluctant to send the message while the receiver wishes to get it. In such protocols, it makes no sense to consider the transfer of messages that are either not secret (to the receiver) or not recognizable (by the receiver).

In symmetric case, such as exchanging secrets, signing contracts, both two participators can easily verify the correctness of the received messages. In unsymmetric case, such as a database manager plays the role of the sender and a client plays the role of the receiver, it is usual that the sender is willing to disclose some messages to the receiver.

To sum up, *if the transferred messages are not recognizable then the receiver can not decide to retrieve which message*. It is reasonable to assume that the transferred messages in an OT scheme are correct. It is unnecessary for the sender to provide any proofs of knowledge. By the way, the definition of "proof of knowledge" is more strong than that of "recognizable message". The following three common examples of recognizable messages come from [11]: (i) A signature of a user to some known message is a recognizable secret message for everybody else. (ii) The key K, by which the plaintext M is transformed using cryptosystem F into ciphertext $F_K(M)$. (iii) The factorization of a composite number, which has only large prime factors.

Based on the above facts, we now can improve the Camenisch-Neven-Shelat OT scheme by removing the computations for some proofs of knowledge. See Table 3 for the improvement.

Theorem 3. *If the (N+1)-SDH assumption and the (N+1)-PDDH assumptions associated to Pg hold, then the OT protocol in Table 3 is sender-secure.*

Proof (Sketch). The proof of this claim can be easily derived from that of Theorem 1, because the witnesses obtained by the receiver in the model of Table 3 consist of pk, y, $C_1, \cdots, C_N, W,$, which are strictly less than that

$$pk,\, y,\, C_1, \cdots, C_N, W, PoM\{(h) : H = e(g,\, h)\},$$

$$PoM\{(h) : H = e(g,\, h) \wedge W = e(h,\, V)\}$$

obtained by the receiver in the model of Table 2. Loosely speaking, the sender in the model of Table 3 shall leak less information to the receiver.

Theorem 4. *The OT protocol in Table 3 is receiver-secure if the sender is semi-honest and the transferred messages are recognizable for the receiver.*

Proof. Define the following distributions games Game-0, $\cdots$, Game-3 such that Game-0=$\text{Real}_{\hat{S},R}$ and Game-3=$\text{Ideal}_{\hat{S}',R'}$. Let D be a universal distinguisher which can efficiently recognize the output distributions of these games. Let $\Pr[\text{Game-i}] = \Pr[D(X) = 1 : X \leftarrow \text{Game-i}]$.

Game-0: In the game, the semi-honest sender $\hat{S}$ runs against an honest receiver R with selection strategy $\sum$. Obviously, $\Pr[\text{Game-0}] = \Pr[D(X) = 1 : X \leftarrow \text{Real}_{\hat{S},R}]$.

Game-1: In this game, an extractor $\mathcal{E}_1$ is used to extract from $\hat{S}$ the element h such that $e(g, h) = H$. If the extractor fails, then the output of Game-1 is $\perp$; otherwise, the execution of $\hat{S}$ continues as in the previous game, interacting with $R(\sum)$. The difference between the two output distributions is given by randomness of selection of h (because $\hat{S}$ is supposed to be semi-honest), i.e., $\Pr[\text{Game-1}] - \Pr[\text{Game-0}] \leq 1/p$.

Game-2: We refer to [5] for the description of this game. By investigating the games, we have that $\Pr[\text{Game-2}] = \Pr[\text{Game-1}]$.

Game-3: In this game, an ideal-world sender $\hat{S}'$ uses $\mathcal{E}_1$ to extract h from $\hat{S}$, decrypts M_i^* as $B_i/e(h, A_i)$ for $i = 1, \cdots, N$ and submits $M_1^*, \cdots, M_N^*$ to the trusted party

Table 3: An improvement of Camenisch-Neven-Shelat OT scheme

Setup		
$(G_1, G_T) \leftarrow \mathsf{Pg}(1^\ell)$ $g, h \leftarrow \mathbb{G}_1^*$ $pk \leftarrow g$; $sk \leftarrow h$		
Transfer		
$\underline{\mathsf{S}_I(1^\ell, M_1, \cdots, M_N):}$ $x \leftarrow Z_p; y \leftarrow g^x$ For $i = 1, \cdots, N$ do $A_i \leftarrow g^{1/(x+i)}$ $B_i \leftarrow e(h, A_i) \cdot M_i$ $C_i \leftarrow (A_i, B_i)$ $S_0 \leftarrow (h, pk)$	$\xrightarrow{pk, y, C_1, \cdots, C_N}$	$\underline{\mathsf{R}_I(1^\ell):}$ $R_0 \leftarrow (pk, C_1, \cdots, C_N)$
$\mathsf{S}_\mathsf{T}(S_{i-1}):$		$\mathsf{R}_\mathsf{T}(R_{i-1}, \sigma_i):$
	$\xleftarrow{V}$	$v \leftarrow \mathbb{Z}_p; V \leftarrow (A_{\sigma_i})^v$
	$\xleftarrow{PoK\{(\sigma_i, v): e(V,y)=e(V,g)^{-\sigma_i} e(g,g)^v\}}$	
$W \leftarrow e(h, V)$	$\xrightarrow{W}$	$M_{\sigma_i} \leftarrow B_{\sigma_i}/(W^{1/v})$
$S_i = S_{i-1}$		$R_i = R_{i-1}$

T. As in Game-2, during the transfer phase, $\hat{S}'$ feeds $V' \leftarrow A_1^{v'}$ to $\hat{S}$ and uses $(v', 1)$ as a witness in the PoK. It is easy to find that $\hat{S}$ can convince $\hat{S}'$ that W is correctly formed (because the transferred messages are recognizable for the receiver). Thus, $\Pr[\text{Game-3}] = \Pr[\text{Game-2}] = \Pr[D(X) = 1 : X \leftarrow \text{Ideal}_{\hat{S}', R'}]$.

Summing up, we have $\Pr[D(X) = 1 : X \leftarrow \text{Ideal}_{\hat{S}', R'}] - \Pr[D(X) = 1 : X \leftarrow \text{Real}_{\hat{S}, R}] \leq 1/p$.

6 Conclusions

We modify the Camenisch-Neven-Shelat adaptive OT protocol by replacing the deterministic encryption with a probabilistic encryption. We further improve it by removing the redundant proofs of knowledge based on the fact that the transferred messages should be recognizable or the sender is willing to disclose some messages to the receiver. We hope the presentation is helpful to clarify some misunderstandings about the primitive of oblivious transfer.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Project 61303200, 61411146001), the Shanghai Leading Academic Discipline Project (S30104), the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] M. Bellare and S. Micali, "Non-interactive oblivious transfer and applications," in *Proceedings of Advances in Cryptology - CRYPTO'89*, pp. 547–557, Santa Barbara, USA, Aug. 1989.

[2] M. K. Bhatia1, S. K. Muttoo, and M. P.Bhatia, "Secure requirement prioritized grid scheduling model," *International Journal of Network Security*, vol. 15, no. 6, pp. 478–483, 2013.

[3] D. Boneh and X. Boyen, "Short signatures without random oracles," in *Proceedings of Advances in Cryptology - EUROCRYPT 2004*, pp. 56–73, Interlaken, Switzerland, May 2004.

[4] G. Brassard, C. Crepeau, and J. Robert, "All-or-nothing disclosure of secrets," in *Proceedings of Advances in Cryptology - CRYPTO'86*, pp. 234–238, Santa Barbara, USA, Aug. 1986.

[5] J. Camenisch, G. Neven, and A. Shelat, "Simulatable adaptive oblivious transfer," in *Proceedings of Advances in Cryptology - EUROCRYPT 2007*, pp. 573–590, Barcelona, Spain, May 2007.

[6] C. C.Chang and J. S. Lee, "Robust t-out-of-n oblivious transfer mechanism based on crt," *Journal of Network and Computer Applications*, vol. 32, no. 1, pp. 226–235, 2009.

[7] C. C. Chang and Y. P. Lai, "Efficient t-out-of-n oblivious transfer schemes," in *Proceedings of the 2008 International Conference on Security Technology*, pp. 3–6, Hainan, China, Dec. 2008.

[8] C. K. Chu and W. G.Tzeng, "Efficient k-out-of-n oblivious transfer schemes," *Journal of Universal Computer Science*, vol. 14, no. 3, pp. 397–415, 2008.

[9] C. K. Chu and W. G. Tzeng, "Efficient k-out-of-n oblivious transfer schemes with adaptive and non-adaptive queries," in *Proceedings of 8th International Workshop on Theory and Practice in Public Key Cryptography (PKC05)*, pp. 172–183, Les Diablerets, Switzerland, Jan. 2005.

[10] T. ElGamal, "A public key cryptosystem and signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, pp. 469–472, 1985.

[11] S. Even, O. Goldreich, and A. Lempel, "A randomized protocol for signing contracts," *Commun. ACM*, vol. 28, no. 6, pp. 637–647, 1985.

[12] M. Green and S. Hohenberger, "How to play any mental game or a completeness theorem for protocols with honest majority," in *Proceedings of 19th Annual ACM Conference on Theory of Computing(STOC'87)*, pp. 218–229, New York,USA, May 1987.

[13] M. Green and S. Hohenberger, "Blind identity-based encryption and simulatable oblivious transfer," in *Proceedings of Advances in Cryptology - ASIACRYPT 2007*, pp. 265–282, Kuching, Malaysia, Dec. 2007.

[14] M. Green and S. Hohenberger, "Practical adaptive oblivious transfer from simple assumptions," in *Proceedings of the Eighth Theory of Cryptography Conference (TCC 2011)*, pp. 347–363, Brown University,USA, Mar. 2011.

[15] A. Jain and C. Har, "A new efficient protocol for k-out-of-n oblivious transfer," *Cryptologia*, vol. 34, no. 4, pp. 282–290, 2010.

[16] M. Kumar, M. K. Gupta, and S. Kumari, "An improved efficient remote password authentication scheme with smart card over insecure networks," *International Journal of Network Security*, vol. 13, no. 3, pp. 167–177, 2011.

[17] K. Kurosawa, R. Nojima, and T. P. Le, "Efficiency-improved fully simulatable adaptive ot under the ddh assumption," in *Proceedings of 7th Conference on Security and Cryptography for Networks (SCN'10)*, pp. 172–181, Amalfi, Italy, Sep. 2010.

[18] K. Kurosawa, R. Nojima, and T. P. Le, "Generic fully simulatable adaptive oblivious transfer," in *Proceedings of 9th International Conference on Applied Cryptography and Network Security (ACNS'11)*, pp. 274–291, Nerja, Spain, June 2011.

[19] Y. J. Liu, C. C. Chang, and S. C. Chang, "An efficient oblivious transfer protocol using residue number system," *International Journal of Network Security*, vol. 15, no. 3, pp. 212–218, 2013.

[20] G. Manikandan, M. Kamarasan, and N. Sairam, "A new approach for secure data transfer based on wavelet transform," *International Journal of Network Security*, vol. 15, no. 2, pp. 106–112, 2013.

[21] Y. Mu, J.Q. Zhang, and V. Varadharajan, "m out of n oblivious transfer," in *Proceedings of Information Security and Privacy, 7th Australian Conference (ACISP2002)*, pp. 395–405, Melbourne, Australia, July 2002.

[22] M. Naor and B. Pinkas, "Oblivious transfer and polynomial evaluation," in *Proceedings of 31th Annual ACM Conference on Theory of Computing(STOC'99)*, pp. 245–254, Atlanta, Georgia, USA, May 1999.

[23] M. Naor and B. Pinkas, "Oblivious transfer with adaptive queries," in *Proceedings of Advances in Cryptology - CRYPTO'89*, pp. 573–590, Santa Barbara,USA, Aug. 1999.

[24] M. Naor and B. Pinkas, "Efficient oblivious transfer protocols," in *Proceedings of 12th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'01)*, pp. 448–457, Washington, D.C., USA, Jan. 2001.

[25] M. Rabin. "How to exchange secrets by oblivious transfer. technical report,". Tech. Rep. TR-81, May 1981.

[26] W. G. Tzeng, "Efficient 1-out-of-n oblivious transfer protocols with universally usable parameter," *IEEE Transactions on Computers*, vol. 53, no. 2, pp. 232–240, 2004.

[27] Q. H. Wu, J. H. Zhang, and Y. M. Wang, "Practical t-out-n oblivious transfer and its applications," *Information and Communications Security*, vol. 2836, pp. 226–237, 2003.

[28] Y. Yao, "How to generate and exchange secrets," in *Proceedings of 27th Annual Symposium on Foundations of Computer Science (FOCS'86)*, pp. 162–167, Toronto, Canada, Oct. 1986.

[29] B. Zeng and et al., "A practical framework for t-out-of-n oblivious transfer with security against covert adversaries," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 465–479, 2012.

[30] B.S. Zhang, "Simulatable adaptive oblivious transfer with statistical receiver's privacy," in *Proceedings of the 5th International Conference on Provable Security (ProvSec 2011)*, pp. 52–67, Xi'an, China, Oct. 2011.

Zhengjun Cao is an associate professor of department of Mathematics at Shanghai University. He received his Ph.D. degree in applied mathematics from Academy of Mathematics and Systems Science, Chinese Academy of Sciences. He served as a post-doctor in Department of Computer Science, Universit Libre de Bruxelles, from 2008 to 2010. His current research interests include cryptography, random algorithms and quantum computation.

Hanyue Cao received the B.S. degree from Shanghai University, Shanghai, China, in 2012. She is currently pursuing her M.S. degree from Department of Mathematics, Shanghai university. Her research interests include information security and cryptography.

Revocable Identity-based Signcryption Scheme Without Random Oracles

Xiangsong Zhang[1], Zhenhua Liu[2,3], Yupu Hu[4] and Tsuyoshi Takagi[5]
(Corresponding author: Zhenhua Liu)

School of Science, Xi'an Technological University, Xi'an, Shaanxi 710032, China[1]
School of Mathematics and Statistics, Xidian University, Xi'an, Shaanxi 710071, China[2]
Guangxi Experiment Center of Information Science, Guilin University of Electronic Technology, Guilin, Guangxi 541004, China[3]
State Key Laboratory of Integrated Services Network, Xidian University, Xi'an, Shaanxi 710071, China[4]
Faculty of Mathematics, Kyushu University, Fukuoka, 819-0395, Japan[5]
(Email: zhualiu@hotmail.com)
(Received Oct. 14, 2013; revised and accepted Nov, 25, 2014)

Abstract

Revocation functionality is crucial for the practicality of the public key cryptosystems including signcryption. When a user's private key is corrupted by hacking or the period of a contract expires, the cryptosystems must provide a revocation method to revoke the misbehaving/compromised user. However, little work has been published on key revocation in identity-based signcryption. We propose a revocable identity-based signcryption scheme. In the scheme, the master key is randomly divided into two parts: one is used to construct the initial key, the other is used to generate the updated key. Furthermore, they are used to periodically and re-randomly generate full private keys for non-revoked users. Thus, the proposed scheme can revoke users and resist key exposure. In the standard model, we prove the proposed scheme with IND-CCA2 security under the DBDH hardness assumption and EUF-CMA security under the CDH hardness assumption.

Keywords: Bilinear pairings, identity-based cryptography, provable security, revocation, signcryption

1 Introduction

Confidentiality, integrity, non-repudiation and authentication are the important requirements for many cryptographic applications. A traditional approach to achieve these requirements is to sign-then-encrypt the message. Signcryption [31] combines the functionality of digital signature and that of public-key encryption in a logical step, and provides the improvements on efficiency over traditional cryptographic mechanisms. The performance advantage makes signcryption useful in many applications, such as shared secret key authentication, resource-constrained network environments and electronic commerce [8, 10, 13, 14].

In an identity-based cryptosystem [23], the public key of a user can be arbitrary strings, such as an email address that uniquely identifies the user. The private key corresponding to the public key or identity is generated by a trusted key authority called key generation center (KGC). Compared with traditional public key cryptosystems using public key infrastructure (PKI), identity-based cryptosystem simplifies the key management problem by avoiding public key certificates. Since then, a large number of papers have been published in this area, including identity-based encryption schemes [3, 27], identity-based signature schemes [1, 9, 12, 20, 26] and identity-based signcryption schemes [1, 4, 6, 11, 15, 16, 17, 29, 30].

Key revocation is critical for the practicality of any public key cryptosystems including identity-based cryptosystem. For example, the private key corresponding to the public key has been stolen, the user has lost her private key, or the user is no longer a legitimate system user. In these cases, it is important that the public/private key pair be revoked or replaced by new keys. In the traditional PKI setting, a certification authority informs the senders about expired or revoked keys of the users via publicly available digital certificates and certificate revocation lists. Many efficient way to revoke users has been studied in numerous studies. However, there are only a few studies in the identity-based cryptosystem setting. To solve the problem of key revocation in the identity-based cryptosystem, Boneh and Franklin [3] suggested that the public key of a user be composed of identity information and time information (called BF revocation technique). Let u_i be a receiver's identity, and T be the current time index. The user's public key is denoted as $\mathrm{u}_i||T$, and the private key $sk_{\mathrm{u}_i,T}$ for non-revoked user u_i on each time

index T is issued by KGC. This means that all users, regardless of whether their keys have been exposed or not, have to periodically get in connect with the KGC, prove their identity and get new private keys. Tseng et al. used the BF revocation technique to propose fully secure revocable identity-based identity-based signature (RIBS) scheme [24] and encryption (RIBE) scheme [25] in the standard model. By the BF revocation technique, the key update complexity at each time index is $\mathcal{O}(n-r)$, with n the number of users and r the number of revoked users. Thus, their solution introduces huge overheads for the KGC that linearly increased in the number of users.

Furthermore, Boldyreva, Goyal and Kumar [2] proposed a new revocable identity-based encryption scheme which used a binary-tree data structure to settle the revocation problem (called BGK revocation technique) in 2008. BGK revocation technique reduces the KGC's periodic key update workload to $\mathcal{O}(r \log \frac{n}{r})$, and their scheme is proved to be selective-identity secure in the standard model. By making use of BGK's binary-tree data structure, Libert and Vergnaud (LV) [18] described an adaptive-identity secure and revocable identity-based encryption scheme, and Chen et al. [7] proposed selective-identity secure and revocable identity-based encryption scheme from lattices. The two schemes share the same key update complexity with the BGK scheme. Liu et al. [19] proposed a low-complexity key updating algorithm, which reduced the binary tree structure of BGK scheme to a tree of depth one, and constructed an efficient revocable identity-based encryption scheme. Most recently, Seo and Emura [21] showed all prior RIBE schemes except for using the BF technique were vulnerable to decryption key exposure attack, where an adversary, who has decryption key $dk_{\mathrm{u}^*,T}$ and key update ku_T, can always recover a part $(D_{x^*,0}, D_{x^*,1})$ of initial private key sk_{u^*} for some x^* if the challenged user u^* is not revoked in time T, and can always obtain a decryption key $dk_{\mathrm{u}^*,T^*} = (D_{x^*,0}, \tilde{D}_{x^*,0}, D_{x^*,1}, \tilde{D}_{x^*,1})$ by combination of the parts $(D_{x^*,0}, D_{x^*,1})$ of sk_{u^*} and $(\tilde{D}_{x^*,0}, \tilde{D}_{x^*,1})$ of ku_{T^*} if u^* is still not revoked in the challenge time T^*. For further details, please read this reference [21]. Then they revisited the Boldyreva et al. security model and proposed the first scalable and efficient RIBE scheme with decryption key exposure resistance. Furthermore, Seo and Emura [22] extended the revocation functionality to the hierarchical identity-based encryption (HIBE).

Signcryption is an important cryptographic primitive. However, little work has been published on revocable identity-based signcryption (RIBSC) schemes. Wu et al. [28] formalized the security model of identity-based signcryption with revocation functionality and proposed the first revocable identity-based signcryption scheme in 2012. Nevertheless, their scheme makes use of the BF revocation technique. Thus this requires the KGC to do work linear in the number of users, and does not scale well as the number of users grows. Moreover, the security of their scheme is demonstrated in the random oracle model. As shown in [5], a proof in the random oracle model can only serve as a heuristic argument and does not necessarily imply the security in the real implementation. Hence, the revocable identity-based signcryption scheme in [28] is not practically secure. In this paper, we focus on efficient identity-based signcryption schemes with revocation functionality without using the random oracles.

The rest of this paper is organized as follows. Some preliminaries are presented in Section 2. The formal model of revocable identity-based signcryption scheme and a concrete construction are detailed in Sections 3 and 4, respectively. We analyze the proposed scheme in Section 5. Finally, some concluding remarks are given in Section 6.

2 Preliminaries

In this section, we briefly review bilinear maps and some complexity assumptions. Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplicative cyclic groups of order p for some large prime p, and g be a generator of $\mathbb{G}$. A bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ should satisfy the following properties:

1) Bilinear: for all $u, v \in \mathbb{G}$ and $a, b \in \mathbb{Z}_p$, $e(u^a, v^b) = e(u, v)^{ab}$;

2) Non-degenerate: $e(g, g) \neq 1_{\mathbb{G}_T}$;

3) Computable: it is efficient to compute $e(u, v)$ for any $u, v \in \mathbb{G}$.

We say that $(\mathbb{G}, \mathbb{G}_T)$ are bilinear map groups if they satisfy these requirements above. In such groups, we describe the following intractability assumptions related to the security of our scheme.

Definition 1. *The challenger chooses $a, b, c, z \in \mathbb{Z}_p$ at random and then flips a fair binary coin $\beta \in \{0, 1\}$. If $\beta = 1$, it outputs the tuple $(g, A = g^a, B = g^b, C = g^c, Z = e(g, g)^{abc})$. Otherwise, if $\beta = 0$, the challenger outputs the tuple $(g, A = g^a, B = g^b, C = g^c, Z = e(g, g)^z)$. The decisional bilinear Diffie-Hellman (DBDH) problem is to guess the value of β.*

An adversary, $\mathcal{C}$, has at least an ϵ advantage in solving the DBDH problem if

$$|Pr[\mathcal{C}(g, g^a, g^b, g^c, e(g, g)^{abc}) = 1] - Pr[\mathcal{C}(g, g^a, g^b, g^c, e(g, g)^z) = 1]| \geq 2\epsilon,$$

where the probability is oven the randomly chosen a, b, c, z and the random bits consumed by $\mathcal{C}$.

The (ϵ, t)-DBDH intractability assumption holds if no t-time adversary $\mathcal{C}$ has at least ϵ advantage in solving the DBDH problem.

Definition 2. *The challenger chooses $a, b \in \mathbb{Z}_p$ at random and outputs (g, g^a, g^b). The computational Diffie-Hellman (CDH) problem is to compute g^{ab}.*

An adversary, $\mathcal{C}$, has at least an ϵ advantage in solving the CDH problem if

$$Pr[\mathcal{C}(g, g^a, g^b) = g^{ab}] \geq \epsilon.$$

The (ϵ, t)-CDH intractability assumption holds if no t-time algorithm has the advantage at least ϵ in solving the CDH problem.

3 Formal Model of RIBSC Scheme

In this section, we define the formal definition of the syntax and the security notions of RIBSC scheme. Our syntax of RIBSC scheme is slightly different from Wu et al. [28]. The main differences are: (1) our key update (KeyUp) algorithm does not bind the identity with the time; (2) our full private key generation (FPKG) algorithm is probabilistic and supports key re-randomization, whereas Wu et al.'s one is deterministic and does not support key re-ranomization; (3) we increase a Revocation algorithm.

3.1 Generic Scheme

Let $\mathcal{M}, \mathcal{I}$ and $\mathcal{T}$ be a message space, an identity space, and a time index space, respectively. A RIBSC scheme consists of seven algorithms as follows.

- **Setup:** This is the (stateful) setup algorithm which takes as input the security parameter λ and the number of users N, and outputs public parameters mpk, a master secret key msk, an initial revocation list $RL = \phi$, and a state st.
- **Initial Private Key Generation:** This is the (stateful) initial private key generation (**IPKG**) algorithm which takes as input mpk, msk, an identity $\mathfrak{u} \in \mathcal{I}$, and outputs a secret key $sk_{\mathfrak{u}}$ associated with $\mathfrak{u}$ and an updated state st.
- **Key Update Generation:** This is the key update generation (**KeyUp**) algorithm which takes as input mpk, msk, the key update time $T \in \mathcal{T}$, the current revocation list RL, and st, and outputs a key update ku_T.
- **Full Private Key Generation:** This is the probabilistic full private key generation (**FPKG**) algorithm which takes as input $mpk, sk_{\mathfrak{u}}$, and ku_T, and outputs a decryption key $dk_{\mathfrak{u},T}$, or $\perp$ if $\mathfrak{u}$ has been revoked.
- **Signcryption:** This is the probabilistic signcryption (**SC**) algorithm which takes as input $mpk, T \in \mathcal{T}$, a sender's identity $\mathfrak{u}_s \in \mathcal{I}$ and decryption key $dk_{\mathfrak{u}_s,T}$, a receiver's identity $\mathfrak{u}_r \in \mathcal{I}$, and a message $M \in \mathcal{M}$, and outputs a ciphertext σ.
- **Designcryption:** This is the deterministic designcryption (**DSC**) algorithm which takes as input $mpk, T \in \mathcal{T}$, a sender's identity $\mathfrak{u}_s \in \mathcal{I}$, a receiver's identity $\mathfrak{u}_r \in \mathcal{I}$ and decryption key $dk_{\mathfrak{u}_r,T}$, and a ciphertext σ, and outputs M or $\perp$ if σ is an invalid ciphertext.
- **Revocation:** This is the stateful revocation (**REV**) algorithm which takes as input an identity to be revoked $\mathfrak{u} \in \mathcal{I}$, a revocation time $T \in \mathcal{T}$, the current revocation list RL, and a state st, and outputs an updated RL.

Every RIBSC scheme should satisfy the following consistency constraint that if

$$\sigma = \mathbf{SC}(mpk, \mathfrak{u}_s, \mathfrak{u}_r, T, dk_{\mathfrak{u}_s,T}, M),$$

then

$$\mathbf{DSC}(mpk, \mathfrak{u}_s, \mathfrak{u}_r, T, dk_{\mathfrak{u}_r,T}, \sigma) = M$$

holds. Next, we provide a security definition of RIBSC scheme that captures realistic threats including decryption key exposure.

3.2 Security Notions

Wu et al. [28] gave the security notions for a RIBSC scheme including the indistinguishability under adaptive chosen-ciphertext attack (*IND-RIBSC-CCA2*) and the existential unforgeablilty under adaptive chosen-message attack (*EUF-RIBSC-CMA*). This model is a natural extension of the security notions of the ordinary identity-based signcryption schemes [4, 16, 17, 30]. According to the generic scheme in Subsection 3.1, we will revise the extended security notions by allowing the adversary to access *full private key generation query* and *revocation query*.

For the *IND-RIBSC-CCA2* property, we consider the following game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

– **Initial.** $\mathcal{C}$ runs the algorithm **Setup** and obtains both the master public key parameters mpk and the master secret key msk. The adversary $\mathcal{A}$ is given mpk but the master secret is kept by the challenger.

– **Phase 1.** $\mathcal{A}$ makes a polynomially bounded number of queries to the challenger $\mathcal{C}$, in an adaptive fashion (i.e., one at time, with knowledge of the previous replies). The following queries are allowed:

  - *Initial private key generation query.* Upon receiving this query with identity $\mathfrak{u} \in \mathcal{I}$, the challenger $\mathcal{C}$ runs $\mathbf{IPKG}(mpk, msk, \mathfrak{u}, \mathsf{st}) \rightarrow sk_{\mathfrak{u}}$ and returns $sk_{\mathfrak{u}}$.
  - *Key update query.* Upon receiving this query with time index $T \in \mathcal{T}$, $\mathcal{C}$ runs $\mathbf{KeyUp}(mpk, msk, T, RL, \mathsf{st}) \rightarrow ku_T$ and returns ku_T.
  - *Revocation query.* Upon receiving this query with $\mathfrak{u} \in \mathcal{I}$ and $T \in \mathcal{T}$, $\mathcal{C}$ runs $\mathbf{REV}(mpk, \mathfrak{u}, T, RL, \mathsf{st}) \rightarrow RL$ and returns the updated revocation list RL.
  - *Full private key generation query.* Upon receiving this query with $\mathfrak{u} \in \mathcal{I}$ and $T \in$

$\mathcal{T}$, $\mathcal{C}$ runs $\mathbf{IPKG}(mpk, msk, \mathtt{u}, \mathtt{st}) \rightarrow sk_{\mathtt{u}}$, $\mathbf{KeyUp}(mpk, msk, T, RL, \mathtt{st}) \rightarrow ku_T$, and $\mathbf{FPKG}(mpk, \mathtt{u}, T, sk_{\mathtt{u}}, ku_T) \rightarrow dk_{\mathtt{u},T}$, and returns $dk_{\mathtt{u},T}$.

- *Signcryption query.* Upon receiving this query for a message $M \in \mathcal{M}$, a sender's identity $\mathtt{u}_s \in \mathcal{I}$, a receiver's identity $\mathtt{u}_r \in \mathcal{I}$, and time index $T \in \mathcal{T}$, $\mathcal{C}$ computes the sender's decryption key $dk_{\mathtt{u}_s,T} = \mathbf{FPKG}(mpk, \mathtt{u}_s, T, sk_{\mathtt{u}_s}, ku_T)$ (if necessary, first need to compute secret key $sk_{\mathtt{u}_s} = \mathbf{IPKG}(mpk, msk, \mathtt{u}_s, \mathtt{st})$ and key update $ku_T = \mathbf{KeyUp}(mpk, msk, T, RL, \mathtt{st})$), runs $\mathbf{SC}(mpk, \mathtt{u}_s, \mathtt{u}_r, T, dk_{\mathtt{u}_s,T}, M) \rightarrow \sigma$, and then returns the ciphertext σ.
- *Designcryption query.* Upon receiving this query for a ciphertext σ, a receiver's identity $\mathtt{u}_r \in \mathcal{I}$, a sender's identity $\mathtt{u}_s \in \mathcal{I}$, and time index $T \in \mathcal{T}$, $\mathcal{C}$ computes the receiver's decryption key $dk_{\mathtt{u}_r,T} = \mathbf{FPKG}(mpk, \mathtt{u}_r, T, sk_{\mathtt{u}_r}, ku_T)$ (if necessary, first need to compute secret key $sk_{\mathtt{u}_r} = \mathbf{IPKG}(mpk, msk, \mathtt{u}_r, \mathtt{st})$ and key update $ku_T = \mathbf{KeyUp}(mpk, msk, T, RL, \mathtt{st})$), runs $\mathbf{DSC}(mpk, \mathtt{u}_s, \mathtt{u}_r, T, dk_{\mathtt{u}_r,T}, \sigma)$, and returns its result to $\mathcal{A}$ (This result can be $\perp$ if σ is an invalid ciphertext).

– **Challenge.** At the end of Phase 1, $\mathcal{A}$ outputs two equal length plaintexts M_0^* and M_1^*, a time index T^*, and two identities $\mathtt{u}_s^*$ and $\mathtt{u}_r^*$, on which it wants to be challenged. $\mathcal{C}$ takes a random bit β from $\{0,1\}$ and runs signcryption algorithm on $(mpk, \mathtt{u}_s^*, \mathtt{u}_r^*, T^*, dk_{\mathtt{u}_s^*,T^*}, M_\beta^*)$ to obtain a ciphertext σ^* which is sent to $\mathcal{A}$.

– **Phase 2.** $\mathcal{A}$ can ask a polynomially bounded number of queries adaptively again as in Phase 1.

– **Guess.** $\mathcal{A}$ produces a bit β' and wins the IND-RIBSC-CCA2 game if $\beta' = \beta$ and the following restrictions are satisfied:

1) *Key update query* and *Revoke query* can be queried on time which is greater than or equal to the time of all previous queries.
2) *Revocation query* cannot be queried on time index T if *Key update query* was queried on T.
3) If *Initial private key generation query* was queried on the challenged identity $\mathtt{u}_r^*$, then *Revocation query* must be queried on $\mathtt{u}_r^*$ for $T \leq T^*$.
4) *Full private key generation query* cannot be queried on time index T before *Key update query* was queried on T.
5) *Full private generation query* cannot be queried on the challenged identity $\mathtt{u}_r^*$ and time index T^*.
6) (σ^*, T^*) was not returned by *Signcryption query* on input $(\mathtt{u}_s^*, \mathtt{u}_r^*, T^*, M_\beta^*)$ for $\beta \in \{0,1\}$.
7) *Designcryption query* cannot be queried on $(\mathtt{u}_s^*, \mathtt{u}_r^*, T^*, \sigma^*)$ to obtain the corresponding plaintext.

The advantage of $\mathcal{A}$ is defined as

$$Adv_{\mathcal{A}}^{IND-RIBSC-CCA2} = |2\Pr[\beta' = \beta] - 1|,$$

where $\Pr[\beta' = \beta]$ denotes the probability that $\beta' = \beta$.

Definition 3. *A RIBSC scheme is said to have the IND-RIBSC-CCA2 property if no polynomially bounded adversary has non-negligible advantage in the above IND-RIBSC-CCA2 game.*

For the *EUF-RIBSC-CMA* property, we consider the following game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

– **Initial.** The phase is the same one defined in the *IND-RIBSC-CCA2* game.

– **Queries.** $\mathcal{A}$ makes a polynomially bounded number of queries to the challenger $\mathcal{C}$. The queries are the same as ones defined in the *IND-RIBSC-CCA2* game.

– **Forge.** $\mathcal{A}$ outputs a new tuple $(\mathtt{u}_s^*, \mathtt{u}_r^*, T^*, \sigma^*)$, where T^* is a time index, $\mathtt{u}_s^*$ is a sender's identity, $\mathtt{u}_r^*$ is a receiver's identity, and σ^* is a ciphertext. We say that $\mathcal{A}$ wins the *EUF-RIBSC-CMA* game if the following restrictions are satisfied:

1) *Key update query* and *Revoke query* can be queried on time which is greater than or equal to the time of all previous queries.
2) *Revocation query* cannot be queried on time index T if *Key update query* was queried on T.
3) If *Initial private key generation query* was queried on the challenged identity $\mathtt{u}_s^*$, then *Revocation query* must be queried on $\mathtt{u}_s^*$ for $T \leq T^*$.
4) *Full private key generation query* cannot be queried on time index T before *Key update query* was queried on T.
5) *Full private key generation query* cannot be queried on the challenged identity $\mathtt{u}_s^*$ and time index T^*.
6) The new tuple $(\mathtt{u}_s^*, \mathtt{u}_r^*, T^*, \sigma^*)$ was not produced by *Signcryption query*.
7) The result of $\mathbf{DSC}(\mathtt{u}_s^*, \mathtt{u}_r^*, T^*, \sigma^*)$ is not the $\perp$ symbol.

The advantage of $\mathcal{A}$ is defined as the probability that it wins.

Definition 4. *A RIBSC scheme is said to have the EUF-RIBSC-CMA property if no polynomially bounded adversary has non-negligible advantage in the above EUF-RIBSC-CMA game.*

4 The Proposed Scheme

4.1 KUNode Algorithm

In the revocation process, we follow the KUNode algorithm and Boldyreva et al.'s idea to reduce the key update costs. In the actual schemes, the algorithm is used in a black-box manner.

Definition 5. *(KUNode Algorithm [2]). This algorithm takes as input a binary tree BT, revocation list RL, and time period index T, and outputs a set of nodes. A formal description of this algorithm is as follows: If η is a non-leaf node, then η_{left} and η_{right} denote the left and right child of η, respectively. Each user is assigned to a leaf node. If a user (assigned to η) is revoked on time index T, then $(\eta, T) \in RL$. Path(η) denotes the set of nodes on the path from η to root. The description of KUNode is given as follows.*

$\mathsf{KUNode}(\mathsf{BT}, RL, T)$:
$\mathsf{X}, \mathsf{Y} \leftarrow \emptyset$
$\forall(\eta_i, T_i) \in RL$
  If $T_i \leq T$ *then add* $\mathsf{Path}(\eta_i)$ *to* X
$\forall x \in \mathsf{X}$
  If $x_{left} \notin \mathsf{X}$ *then add* x_{left} *to* Y
  If $x_{right} \notin \mathsf{X}$ *then add* x_{right} *to* Y
If $\mathsf{Y} = \emptyset$ *then add* root *to* Y
Return Y.

This KUNode algorithm can be used to compute the minimal set of nodes for which key update needs to be published so that only non-revoked users at time index T are able to generate full private key. Please see a simple example in [21] to easily understand KUNode(BT,RL, T). When a user joins the system, the key authority assigns it to the leaf node η of a complete binary tree, and issues a set of keys, wherein each key is associated with each node on Path(η). At time index T, the key authority KGC publishes key updates for a set KUNode(BT, RL, T). Then, only non-revoked users have at least one key corresponding to a node in KUNode(BT, RL, T) and are able to generate decryption keys on time index T.

4.2 Our Construction

The new revocable identity-based signcryption can be described as the following algorithms.

- **Setup**(λ, N): On input (λ, N), the key authority does the followings:
  1) Generate two cyclic groups $\mathbb{G}$ and $\mathbb{G}_T$ of prime order p with a bilinear map $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ and g, g_2 the generators of $\mathbb{G}$;
  2) Choose a secret $\alpha \in \mathbb{Z}_p$, compute $g_1 = g^\alpha$ and pick up $u', m', v', v \in \mathbb{G}$ and two vectors $\overrightarrow{u} = (u_i), \overrightarrow{m} = (m_j)$ of length n_u and n_m respectively, where u_i, m_j are chosen from $\mathbb{G}$ randomly.
  3) Choose a collision-resistant hash function $H : \{0,1\}^* \rightarrow \{0,1\}^{n_m}$;
  4) Set master public parameter $mpk = \{g, g_1, g_2, u', m', \overrightarrow{u}, \overrightarrow{m}, v', v\}$, master secret key $msk = \alpha$, $RL = \emptyset$, and st = BT, where BT is a binary tree with N leaves.
- **Initial Private Key Generation**$(mpk, msk, \mathtt{u}, \mathsf{st})$: Randomly choose an unassigned leaf η from BT, and store $\mathtt{u}$ in the node η. Let $\mathcal{U} \subset \{1, 2, \cdots, n_u\}$ be the set of indices such that $\mathtt{u}[i] = 1$, where $\mathtt{u}[i]$ is the i-th bit of $\mathtt{u}$. For each node $\theta \in \mathsf{Path}(\eta)$,
  1) Recall g_θ if it was defined. Otherwise, $g_\theta \xleftarrow{\$} \mathbb{G}$ and store $(g_\theta, \tilde{g}_\theta = g_2/g_\theta)$ in the node θ.
  2) Choose $r_\theta \xleftarrow{\$} \mathbb{Z}_p$.
  3) Compute $D_{\theta,0} \leftarrow g_\theta^\alpha (u' \prod_{i \in \mathcal{U}} u_i)^{r_\theta}, D_{\theta,1} \leftarrow g^{r_\theta}$.
  4) Output secret key $sk_{\mathtt{u}} = \{(\theta, D_{\theta,0}, D_{\theta,1})\}_{\theta \in \mathsf{Path}(\eta)}$.
- **Key Update Generation**$(mpk, msk, T, RL, \mathsf{st})$: Parse st = BT. For each node $\theta \in$ KUNode (BT, RL, T),
  1) Retrieve $\tilde{g}_\theta$ (note that $\tilde{g}_\theta$ is always pre-defined in the **Initial Private Key Generation** algorithm).
  2) Choose $s_\theta \xleftarrow{\$} \mathbb{Z}_p$.
  3) Compute $\tilde{D}_{\theta,0} \leftarrow \tilde{g}_\theta^\alpha (v' v^T)^{s_\theta}, \tilde{D}_{\theta,1} \leftarrow g^{s_\theta}$.
  4) Output key update $ku_T = \{(\theta, \tilde{D}_{\theta,0}, \tilde{D}_{\theta,1})\}_{\theta \in \mathsf{KUNode}(\mathrm{BT}, RL, T)}$.
- **Full Private Key Generation**$(mpk, sk_{\mathtt{u}}, ku_T)$: Parse $sk_{\mathtt{u}} = \{(\theta, D_{\theta,0}, D_{\theta,1})\}_{\theta \in \mathtt{I}}$ and $ku_T = \{(\theta, \tilde{D}_{\theta,0}, \tilde{D}_{\theta,1})\}_{\theta \in \mathtt{J}}$, where $\mathtt{I}$ denotes Path(η) and $\mathtt{J}$ denotes KUNode(BT, RL, T). If $\mathtt{I} \cap \mathtt{J} = \emptyset$, then return $\perp$. Otherwise, choose $\theta \in \mathtt{I} \cap \mathtt{J}$ and $r, s \xleftarrow{\$} \mathbb{Z}_p$ and return full private/decryption key

$$dk_{\mathtt{u},T} = (D_{\theta,0}\tilde{D}_{\theta,0}(u' \prod_{i \in \mathcal{U}} u_i)^r (v' v^T)^s, D_{\theta,1} g^r, \tilde{D}_{\theta,1} g^s)$$
$$= (g_2^\alpha (u' \prod_{i \in \mathcal{U}} u_i)^{r_\theta + r} (v' v^T)^{s_\theta + s}, g^{r_\theta + r}, g^{s_\theta + s}).$$

Let $\mathtt{u}_A$ be sender Alice's identity and $\mathtt{u}_B$ receiver Bob's identity. Then the full private key of Alice at some time period index T is

$$dk_{\mathtt{u}_A,T} = (g_2^\alpha (u' \prod_{i \in \mathcal{U}_A} u_i)^{r_{\theta_A} + r_A} (v' v^T)^{s_{\theta_A} + s_A},$$
$$g^{r_{\theta_A} + r_A}, g^{s_{\theta_A} + s_A}).$$

And the full private key of Bob at some time period index T is

$$dk_{\mathfrak{u}_B,T} = (g_2^{\alpha}(u' \prod_{i\in\mathcal{U}_B} u_i)^{r_{\theta_B}+r_B}(v'v^T)^{s_{\theta_B}+s_B}, g^{r_{\theta_B}+r_B}, g^{s_{\theta_B}+s_B}).$$

- **Signcryption**$(mpk, \mathfrak{u}_A, \mathfrak{u}_B, T, dk_{\mathfrak{u}_A,T}, M)$: On input $M \in \mathbb{G}_T$, the receiver Bob's identity $\mathfrak{u}_B$, the sender Alice's identity $\mathfrak{u}_A$ and full private key $dk_{\mathfrak{u}_A,T} = (dk_{\mathfrak{u}_A,T,1}, dk_{\mathfrak{u}_A,T,2}, dk_{\mathfrak{u}_A,T,3})$, and the current time index T, the algorithm does the following:

  1) Randomly choose a random integer $k \in \mathbb{Z}_p$.
  2) Compute $\sigma_0 = M \cdot e(g_1, g_2)^k$, $\sigma_1 = g^{-k}$, $\sigma_2 = (u' \prod_{i\in\mathcal{U}_B} u_i)^k$, $\sigma_3 = (v'v^T)^k$, $\sigma_4 = dk_{\mathfrak{u}_A,T,2}$, and $\sigma_5 = dk_{\mathfrak{u}_A,T,3}$.
  3) Compute $\mathfrak{m} = H_1(\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \mathfrak{u}_A, \mathfrak{u}_B)$, and let $\mathcal{M} \subset \{1, \cdots, n_m\}$ be the set of indices j such that $\mathfrak{m}[j] = 1$.
  4) Compute $\sigma_6 = dk_{\mathfrak{u}_A,T,1} \cdot (m' \prod_{j\in\mathcal{M}} m_j)^k$.
  5) Output $\sigma = (\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6)$.

- **Designcryption**$(mpk, \mathfrak{u}_A, \mathfrak{u}_B, T, dk_{\mathfrak{u}_B,T}, \sigma)$: On input $\sigma = (\sigma_0, \cdots, \sigma_6)$, the time index T, the receiver's full private key $dk_{\mathfrak{u}_B,T} = (dk_{\mathfrak{u}_B,T,1}, dk_{\mathfrak{u}_B,T,2}, dk_{\mathfrak{u}_B,T,3})$ and the sender's identity $\mathfrak{u}_A$, the algorithm outputs M, or $\perp$ (if the signcryptext is not valid) as follows:

  1) Compute $\mathfrak{m} = H_1(\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \mathfrak{u}_A, \mathfrak{u}_B)$, and let $\mathcal{M} \subset \{1, \cdots, n_m\}$ be the set of indices j such that $\mathfrak{m}[j] = 1$.
  2) Check if the following equation holds:

$$e(\sigma_6, g) \stackrel{?}{=} e(g_1, g_2)e(u' \prod_{i\in\mathcal{U}_A} u_i, \sigma_4)e(v'v^T, \sigma_5) \cdot e(m' \prod_{j\in\mathcal{M}} m_j, \sigma_1^{-1}). \quad (1)$$

  if Equation (1) holds, output

$$M = \sigma_0 \cdot \prod_{i=1}^{3} e(dk_{\mathfrak{u}_B,T,i}, \sigma_i). \quad (2)$$

- **Revocation**$(mpk, \mathfrak{u}, T, RL, \mathsf{st})$: Let η be the leaf node associated with $\mathfrak{u}$. Update the revocation list by $RL \leftarrow RL \cup \{\eta, T\}$ and return the updated revocation list.

5 Security Analysis

5.1 Consistency

Now we verify the consistency of our scheme. For Equation (1), we have

$$\begin{aligned}
&e(\sigma_6, g)\\
=&e(dk_{\mathfrak{u}_A,T,1}(m' \prod_{j\in\mathcal{M}} m_j)^k, g)\\
=&e(g_2^{\alpha}(u' \prod_{i\in\mathcal{U}_A} u_i)^{r_{\theta_A}+r_A}(v'v^T)^{s_{\theta_A}+s_A}(m' \prod_{j\in\mathcal{M}} m_j)^k, g)\\
=&e(g_2^{\alpha}, g)e((u' \prod_{i\in\mathcal{U}_A} u_i)^{r_{\theta_A}+r_A}, g)e((v'v^T)^{s_{\theta_A}+s_A}, g)\\
&\cdot e((m' \prod_{j\in\mathcal{M}} m_j)^k, g)\\
=&e(g_1, g_2)e(u' \prod_{i\in\mathcal{U}_A} u_i, \sigma_4)e(v'v^T, \sigma_5)e(m' \prod_{j\in\mathcal{M}} m_j, \sigma_1^{-1}).
\end{aligned}$$

For Equation (2), we have

$$\begin{aligned}
&\sigma_0 \prod_{i=1}^{3} e(dk_{\mathfrak{u}_B,T,i}, \sigma_i)\\
=&Me(g_1, g_2)^k \frac{e(g^{r_{\theta_B}+r_B}, (u' \prod_{i\in\mathcal{U}_B} u_i)^k)e(g^{s_{\theta_B}+s_B}, (v'v^T)^k)}{e(g_2^{\alpha}(u' \prod_{i\in\mathcal{U}_B} u_i)^{r_{\theta_B}+r_B}(v'v^T)^{s_{\theta_B}+s_B}, g^k)}\\
=&\frac{Me(g_1, g_2)^k \cdot e(g^{r_{\theta_B}+r_B}, (u' \prod_{i\in\mathcal{U}_B} u_i)^k)e(g^{s_{\theta_B}+s_B}, (v'v^T)^k)}{e(g_2^{\alpha}, g^k)e((u' \prod_{i\in\mathcal{U}_B} u_i)^{r_{\theta_B}+r_B}, g^k)e((v'v^T)^{s_{\theta_B}+s_B}, g^k)}\\
=&\frac{Me(g_1, g_2)^k \cdot e(g^{r_{\theta_B}+r_B}, (u' \prod_{i\in\mathcal{U}_B} u_i)^k)e(g^{s_{\theta_B}+s_B}, (v'v^T)^k)}{e(g_2, g_1)^k e((u' \prod_{i\in\mathcal{U}_B} u_i)^k, g^{r_{\theta_B}+r_B})e((v'v^T)^k, g^{s_{\theta_B}+s_B})}\\
=&M.
\end{aligned}$$

5.2 Security

Next, we reduce the IND-RIBSC-CCA2 property to the DBDH hardness assumption and the EUF-RIBSC-CMA property to the CDH hardness assumption.

Theorem 1. *If there exists an adversary $\mathcal{A}$ attacking IND-RIBSC-CCA security of the proposed RIBSC scheme, then there exists a challenger $\mathcal{C}$ breaking a DBDH problem instance.*

Proof. We suppose that an $(\epsilon, t, q_{ipk}, q_{ku}, q_{fpk}, q_r, q_s, q_d)$ adversary $\mathcal{A}$ for our scheme exists, where it has advantage at least ϵ, runs in time at most t, and makes at most q_{ipk} initial private key queries, q_{ku} key update queries, q_{fpk} full private key queries, q_r revocation queries, q_s signcryption queries, and q_d designcryption queries. From the adversary, we construct a simulator $\mathcal{C}$, which makes use of $\mathcal{A}$ to solve DBDH game with a probability at least ϵ' and in time at most t', contradicting the (ϵ', t')-DBDH

assumption. Our approach is based on Waters' idea such as [16, 17, 20, 21, 30].

$\mathcal{C}$ will take DBDH challenge $(g, A = g^a, B = g^b, C = g^c, Z)$ and output a guess, β', as to whether the challenge is a DBDH tuple. In order to use $\mathcal{A}$ to solve the problem, $\mathcal{C}$ needs to simulate a challenger and all queries for $\mathcal{A}$. $\mathcal{C}$ then simulates the queries of $\mathcal{A}$ as follows.

Setup: $\mathcal{C}$ randomly guesses the challenge time $T^* \in \mathcal{T}$. We assume that $\mathcal{C}$'s guess is right. (It holds with $1/|\mathcal{T}|$ and this is a loss of polynomial in λ.) Let $l_u = 2(q_{ipk} + q_{fpk} + q_s + q_d)$ and $l_m = 2(q_s + q_d)$.

1) $\mathcal{C}$ randomly chooses two integers k_u and k_m ($0 \leq k_u \leq n_u, 0 \leq k_m \leq n_m$). We assume that $l_u(n_u + 1) < p$ and $l_m(n_m + 1) < p$ for the given values of $q_{ipk}, q_{fpk}, q_s, q_d, n_u$ and n_m.

2) $\mathcal{C}$ picks an integer $x' \in \mathbb{Z}_{l_u}$ and a vector $\mathbf{X} = (x_i)_{n_u} (x_i \in \mathbb{Z}_{l_u})$ at random.

3) $\mathcal{C}$ randomly selects an integer $z' \in \mathbb{Z}_{l_m}$ and a vector $\mathbf{Z} = (z_j)_{n_m} (z_j \in \mathbb{Z}_{l_m})$.

4) $\mathcal{C}$ randomly picks two integers $y', w' \in \mathbb{Z}_p$ and two vectors $\mathbf{Y} = (y_i)_{n_u} (y_i \in \mathbb{Z}_p)$ and $\mathbf{W} = (w_j)_{n_m} (w_j \in \mathbb{Z}_p)$.

5) $\mathcal{C}$ randomly chooses $\nu, \nu' \in \mathbb{Z}_p$.

For convenience, we define the two pairs of functions for binary identity string $\mathfrak{u}$ and message string $\mathfrak{m}$ as follows:

$$
\begin{aligned}
F(\mathfrak{u}) &= (p - l_u k_u) + x' + \sum_{i \in \mathcal{U}} x_i, \\
J(\mathfrak{u}) &= y' + \sum_{i \in \mathcal{U}} y_i, \\
K(\mathfrak{m}) &= (p - l_m k_m) + z' + \sum_{j \in \mathcal{M}} z_j \\
L(\mathfrak{m}) &= w' + \sum_{j \in \mathcal{M}} w_j,
\end{aligned}
$$

where $\mathcal{U} \subset \{1, \cdots, n_u\}$ denotes the set of indices i such that $\mathfrak{u}[i] = 1$ and $\mathcal{M} \subset \{1, \cdots, n_m\}$ denotes the set of indices j such that $\mathfrak{m}[j] = 1$. Then the challenger assigns a set of public parameters as follows:

$$
\begin{aligned}
&g_1 = g^a, && g_2 = g^b, \\
&u' = g_2^{(p - l_u k_u) + x'} g^{y'}, && u_i = g_2^{x_i} g^{y_i} (1 \leq i \leq n_u), \\
&m' = g_2^{(p - l_m k_m) + z'} g^{w'}, && m_j = g_2^{z_j} g^{w_j} (1 \leq j \leq n_m), \\
&v' = g_1^{-T^*} \cdot g^{\nu'}, && v = g_1 \cdot g^{\nu}.
\end{aligned}
$$

Note that the master secret key is $g_2^{\alpha} = g_1^b = g^{ab}$ and the following equations hold for an identity $\mathfrak{u}$ and a message $\mathfrak{m}$:

$$
u' \prod_{i \in \mathcal{U}} u_i = g_2^{F(\mathfrak{u})} g^{J(\mathfrak{u})}, \qquad m' \prod_{j \in \mathcal{M}} m_j = g_2^{K(\mathfrak{m})} g^{L(\mathfrak{m})}.
$$

Then, it publishes $mpk = \{g, g_1, g_2, u', \overrightarrow{u} = (u_i), m', \overrightarrow{m} = (m_j), v', v\}$. The corresponding master secret key is g_2^{α}. Although $\mathcal{C}$ does not know the master secret key, it still can construct a private key (d_0, d_1) for an identity $\mathfrak{u}$ by assuming $F(\mathfrak{u}) \neq 0 \mod p$, which is the private key generation oracle $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$ of the Waters IBE scheme [27]. $\mathcal{C}$ randomly chooses $r_u \in \mathbb{Z}_p$ and computes:

$$
(d_0, d_1) = (g_1^{-J(\mathfrak{u})/F(\mathfrak{u})} (u' \prod_{i \in \mathcal{U}} u_i)^{r_u}, g_1^{-1/F(\mathfrak{u})} g^{r_u}).
$$

By writing $\hat{r}_u = r_u - a/F(\mathfrak{u})$, we can show that (d_0, d_1) is a valid private key for the identity $\mathfrak{u}$ as follows. The challenger $\mathcal{C}$ can generate such a private key (d_0, d_1) if and only if $F(\mathfrak{u}) \neq 0 \mod l_u$, which suffices to have $F(\mathfrak{u}) \neq 0 \mod p$. The simulation is perfect since

$$
\begin{aligned}
d_0 &= g_1^{-J(\mathfrak{u})/F(\mathfrak{u})} (u' \prod_{i \in \mathcal{U}} u_i)^{r_u} \\
&= g_2^a (g_2^{F(\mathfrak{u})} g^{J(\mathfrak{u})})^{-a/F(\mathfrak{u})} (g_2^{F(\mathfrak{u})} g^{J(\mathfrak{u})})^{r_u} \\
&= g_2^a (g_2^{F(\mathfrak{u})} g^{J(\mathfrak{u})})^{r_u - a/F(\mathfrak{u})} \\
&= g_2^a (u' \prod_{i \in \mathcal{U}} u_i)^{\hat{r}_u},
\end{aligned}
$$

and $d_1 = g_1^{-1/F(\mathfrak{u})} g^{r_u} = g^{r_u - a/F(\mathfrak{u})} = g^{\hat{r}_u}$. If, on the other hand, $F(\mathfrak{u}) = 0 \mod p$, $\mathcal{C}$ aborts.

Let $\mathfrak{u}^*$ be the challenge identity. $\mathcal{C}$ guesses an adversarial type among the following two types:

1. Type-1 adversary: $\mathcal{A}$ issues an initial private key generation query for $sk_{\mathfrak{u}^*}$, and so $\mathfrak{u}^*$ should be revoked before T^*. (For $T \neq T^*$, $\mathcal{A}$ may query $dk_{\mathfrak{u}^*,T}$.)

2. Type-2 adversary: $\mathcal{A}$ does not query $sk_{\mathfrak{u}^*}$, but $\mathcal{A}$ may issue $dk_{\mathfrak{u}^*,T}$ for $T \neq T^*$.

We assume that $\mathcal{C}$'s guess is right. (It holds with $1/2$ probability.) We separately describe $\mathcal{C}$'s other process according to its guess.

Type-1 Adversary. Let q be the maximum number of queries regarding initial private key generation queries, full private key generation queries, signcryption queries or designcryption queries. $\mathcal{C}$ randomly guesses $i^* \in [1, q]$ such that $\mathcal{A}$'s i^*-th query is the first query regarding $\mathfrak{u}^*$ among initial private key generation queries, full private key generation queries, signcryption queries and designcryption queries. We assume that $\mathcal{C}$'s guess is right. (It holds with $1/q$ and this is a loss of polynomial in λ.) $\mathcal{C}$ randomly choose a leaf node η^* that will be used for $\mathfrak{u}^*$ (this is not a security loss, but just a pre-assignment for $\mathfrak{u}^*$.) $\mathcal{C}$ marks η^* as a defined node. $\mathcal{C}$ keeps an integer count to count the number of queries for initial private key generation, full private key generation, signcryption or designcryption up to the current time.

Key Update Queries: For all nodes $\theta \in \mathsf{KUNode}(\mathsf{BT}, RL, T)$, $\mathcal{C}$ recalls S_θ from the node θ if it is defined. Otherwise, $\mathcal{C}$ chooses $S_\theta \xleftarrow{\$} \mathbb{G}$ and stores it in the node θ. $\mathcal{C}$ computes $\tilde{D}_{\theta,0}$ and $\tilde{D}_{\theta,1}$ as follows: if $\theta \notin \mathsf{Path}(\eta^*)$, then

$$
(\tilde{D}_{\theta,0}, \tilde{D}_{\theta,1}) = (S_\theta^{-1} (v' v^T)^{s_\theta}, g^{s_\theta}),
$$

otherwise

$$(\tilde{D}_{\theta,0}, \tilde{D}_{\theta,1})$$
$$=(S_\theta^{-1} g_2^{-\frac{\nu'+\nu T^*}{T-T^*}} g_1^{s_\theta(T-T^*)} g^{s_\theta(\nu'+\nu T^*)}, g_2^{-\frac{1}{T-T^*}} g^{s_\theta}),$$

where $s_\theta \xleftarrow{\$} \mathbb{Z}_p$. In fact, if $\theta \in \mathsf{Path}(\eta^*)$, then

$$(\tilde{D}_{\theta,0}, \tilde{D}_{\theta,1})$$
$$=(S_\theta^{-1} g_2^{-\frac{\nu'+\nu T^*}{T-T^*}} g_1^{s_\theta(T-T^*)} g^{s_\theta(\nu'+\nu T^*)}, g_2^{-\frac{1}{T-T^*}} g^{s_\theta})$$
$$=(S_\theta^{-1} g_2^a (g_1^{T-T^*} g^{\nu'+\nu T^*})^{-\frac{b}{T-T^*}+s_\theta}, g^{-\frac{b}{T-T^*}+s_\theta})$$
$$=(S_\theta^{-1} g_2^a (v'v^T)^{s'_\theta}, g^{s'_\theta})$$

where $s'_\theta = -\frac{b}{T-T^*} + s_\theta$. Output

$$ku_T = \{(\theta, \tilde{D}_{\theta,0}, \tilde{D}_{\theta,1})\}_{\theta \in \mathsf{KUNode}(\mathtt{BT}, RL, T)}.$$

When $T = T^*$, $\mathfrak{u}^*$ should be in the revocation list RL so that $\mathcal{C}$ performs the above computation for only $\theta \notin \mathsf{Path}(\eta^*)$.

Revocation Queries: Upon receiving this query on $(\mathfrak{u}, T)$, $\mathcal{C}$ runs algorithm $\mathbf{REV}(mpk, \mathfrak{u}, T, RL, \mathsf{st}) \rightarrow RL$ and returns the updated revocation list RL.

From now, we explain how $\mathcal{C}$ responds to initial private key generation queries, full private key generation queries, signcryption queries and designcryption queries according to count.

Case $\mathsf{count} < i^*$: Whenever $\mathcal{C}$ receives either initial private key generation query for $\mathfrak{u}$, full private key generation query for $(\mathfrak{u}, T)$, signcryption query for $(\mathfrak{u}, \mathfrak{u}_r, M, T)$, or designcryption query for $(\mathfrak{u}_s, \mathfrak{u}, \sigma, T)$, $\mathcal{C}$ firstly sends $\mathfrak{u}$ to $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$ oracle and obtains (d_0, d_1), and then randomly chooses an undefined leaf node η and store $\mathfrak{u}$ in η.

- **Initial Private Key Generation Queries**: For $\theta \in \mathsf{Path}(\eta^*)$, $\mathcal{C}$ recalls S_θ if it is defined. Otherwise, $S_\theta \xleftarrow{\$} \mathbb{G}$ and store it in the node θ. Compute

$$(D_{\theta,0}, D_{\theta,1})$$
$$= \begin{cases} (S_\theta (u' \prod_{i \in \mathcal{U}} u_i)^{r_\theta}, g^{r_\theta}), \text{if } \theta \in \mathsf{Path}(\eta^*), \\ (S_\theta d_0 (u' \prod_{i \in \mathcal{U}} u_i)^{r_\theta}, d_1 g^{r_\theta}), \text{otherwise}, \end{cases}$$

  where $r_\theta \xleftarrow{\$} \mathbb{Z}_p$. Return the secret key $sk_{\mathfrak{u}} = \{(\theta, D_{\theta,0}, D_{\theta,1})\}_\theta$.
- **Full Private Key Generation Queries**: Run key update query and initial private key generation query, and then run full private key generation algorithm as follows (Since $\mathsf{count} < i^*$, $\mathfrak{u} \neq \mathfrak{u}^*$ holds. So, $\mathcal{C}$ can query $\mathfrak{u}$ to $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$ oracle).

1) For the case of $\theta \in \mathsf{KUNode}(\mathtt{BT}, RL, T) \cap \neg\, \mathsf{Path}(\eta^*)$, $\mathcal{C}$ runs initial private key generation query and key update query to obtain secret key

$$sk_{\mathfrak{u}} = \{(\theta, D_{\theta,0}, D_{\theta,1})\}_\theta$$
$$= \{(\theta, S_\theta \cdot d_0 \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_\theta}, d_1 \cdot g^{r_\theta})\}_\theta,$$

and update key

$$ku_T = \{(\theta, \tilde{D}_{\theta,0}, \tilde{D}_{\theta,1})\}_\theta$$
$$= \{(\theta, S_\theta^{-1} (v'v^T)^{s_\theta}, g^{s_\theta})\}_\theta.$$

If $\mathsf{KUNode}(\mathtt{BT}, RL, T) \cap \neg\mathsf{Path}(\eta^*) = \emptyset$, then return $\perp$. Otherwise, choose $\theta \in \mathsf{KUNode}(\mathtt{BT}, RL, T) \cap \neg\mathsf{Path}(\eta^*)$ and $r, s \xleftarrow{\$} \mathbb{Z}_p$ and return the decryption key

$$dk_{\mathfrak{u},T} = (D_{\theta,0} \cdot \tilde{D}_{\theta,0} \cdot (u' \prod_{i \in \mathcal{U}} u_i)^r (v'v^T)^s,$$
$$D_{\theta,1} \cdot g^r, \tilde{D}_{\theta,1} \cdot g^s)$$
$$= (g_2^a (u' \prod_{i \in \mathcal{U}} u_i)^{\hat{r}_u + r_\theta + r} (v'v^T)^{s_\theta + s},$$
$$g^{\hat{r}_u + r_\theta + r}, g^{s_\theta + s}).$$

2) For the case of $\theta \in \mathsf{KUNode}(\mathtt{BT}, RL, T) \cap \mathsf{Path}(\eta^*)$, then $\mathcal{C}$ does the similar process as above and returns the decryption key

$$dk_{\mathfrak{u},T} = (g_2^a (u' \prod_{i \in \mathcal{U}} u_i)^{r_\theta + r} (v'v^T)^{s'_\theta + s},$$
$$g^{r_\theta + r}, g^{s'_\theta + s}).$$

- **Signcryption Queries**: When $\mathcal{A}$ queries the signcrypt oracle for a message M, a time index T, a sender's identity $\mathfrak{u}$ and a receiver's identity $\mathfrak{u}_r$, the challenger $\mathcal{C}$ proceeds as follows:
  1. Computes a decryption key $dk_{\mathfrak{u},T}$ by running a full private key generation query for $\mathfrak{u}$ and T (If it is necessary to query $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$ oracle on $\mathfrak{u}$, but $F(\mathfrak{u}) = 0 \mod l_u$, $\mathcal{C}$ will simply abort).
  2. Run the algorithm $\mathbf{SC}(\mathfrak{u}, \mathfrak{u}_r, T, M)$ and return its output as response.
- **Designcryption Queries**: At any time $\mathcal{A}$ can perform a designcryption query for a ciphertext $\sigma = (\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6)$ associated with T, $\mathfrak{u}_s$ and $\mathfrak{u}$, $\mathcal{C}$ does the following.
  1. Compute

$$\mathfrak{m} = H_1(\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \mathfrak{u}_s, \mathfrak{u}),$$

2. Set $\mathcal{M} \subset \{1, \cdots, n_m\}$ be the set of indices j such that $\mathfrak{m}[j] = 1$, where $\mathfrak{m}[j]$ is the j-th bit of $\mathfrak{m}$,
3. Check the equation

$$e(\sigma_6, g) \stackrel{?}{=} e(v'v^T, \sigma_5) e(m' \prod_{j \in \mathcal{M}} m_j, \sigma_1^{-1}) \cdot e(g_1, g_2) e(u' \prod_{i \in \mathcal{U}} u_i, \sigma_4). \quad (3)$$

4. Prepare its response according to the following situations.

(i) If Equation (3) does not hold, $\mathcal{C}$ rejects the ciphertext.

(ii) If Equation (3) holds and $F(\mathfrak{u}) = 0 \mod l_u$, but it is necessary to query $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$ oracle on $\mathfrak{u}$, then $\mathcal{C}$ will abort.

(iii) If Equation (3) holds and $F(\mathfrak{u}) \neq 0 \mod l_u$, or $F(\mathfrak{u}) = 0 \mod l_u$, but it is not necessary to query $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$ oracle on $\mathfrak{u}$, then $\mathcal{C}$ makes a full private key generation query on $\mathfrak{u}$ and T, and obtains the decryption key $dk_{\mathfrak{u},T} = (dk_{\mathfrak{u},T,1}, dk_{\mathfrak{u},T,2}, dk_{\mathfrak{u},T,3})$ and returns the message

$$M = \sigma_0 \cdot \prod_{i=1}^{3} e(dk_{\mathfrak{u},T,i}, \sigma_i).$$

Case $\mathsf{count}{=}i^*$: $\mathcal{C}$ can identify $\mathfrak{u}^*$ and store $\mathfrak{u}^*$ in the pre-assigned leaf node η^*.

- **Initial Private Key Generation Queries**: For $\theta \in \mathsf{Path}(\eta^*)$, $\mathcal{C}$ recalls S_θ if it is defined. Otherwise, $S_\theta \stackrel{\$}{\leftarrow} \mathbb{G}$ and store it in the node θ. Return

$$(S_\theta \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_\theta}, g^{r_\theta}), \text{where } r_\theta \stackrel{\$}{\leftarrow} \mathbb{Z}_p.$$

Note that it is not necessary to obtain (d_0, d_1) by sending $\mathfrak{u}$ to $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$ oracle. Thus in this case we do not need to consider whether $F(\mathfrak{u}) = 0 \mod l_u$ or not.

- **Full Private Key Generation Queries**: Run initial private key generation query for $\mathfrak{u}$ and key update query on T, and then run full private key generation algorithm for $\mathfrak{u}$ and T.
- **Signcryption Queries**: When $\mathcal{A}$ queries the signcrypt oracle for a message M, a time index T, a sender's identity $\mathfrak{u}$ and a receiver's identity $\mathfrak{u}_r$, the challenger $\mathcal{C}$ will proceeds the same as signcryption query in the case $\mathsf{count} < i^*$.
- **Designcryption Queries**: At any time $\mathcal{A}$ can perform a designcryption query for a ciphertext $\sigma = (\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6)$ associated with T, $\mathfrak{u}_s$ and $\mathfrak{u}$, $\mathcal{C}$ does the same as the designcryption query in the case $\mathsf{count} < i^*$.

Case $\mathsf{count} > i^*$: If $\mathfrak{u} \neq \mathfrak{u}^*$, then $\mathcal{C}$ does the same process as the queries in the case $\mathsf{count} < i^*$. Otherwise, $\mathcal{C}$ acts the same as the queries in the case $\mathsf{count} = i^*$.

Challenge: At the end of the first stage, $\mathcal{A}$ outputs two messages M_0^* and M_1^*, a time period index T^*, a receiver's identity $\mathfrak{u}_r^*$, and a sender's identity $\mathfrak{u}^*$ on which it wishes to be challenged. Then, $\mathcal{C}$ chooses a random bit $\beta \in \{0, 1\}$ and fails if $F(\mathfrak{u}^*) = 0 \mod l_u$, or $F(\mathfrak{u}_r^*) \neq 0 \mod l_u$, or $K(\mathfrak{m}^*) \neq 0 \mod l_m$. Otherwise, $\mathcal{C}$ first makes the full private key generation query on $(\mathfrak{u}^*, T^*)$ and obtains the decryption key $dk_{\mathfrak{u}^*,T^*} = (dk_{\mathfrak{u}^*,T^*,1}, dk_{\mathfrak{u}^*,T^*,2}, dk_{\mathfrak{u}^*,T^*,3})$, then sets

$$\begin{array}{ll} \sigma_0^* = M_\beta^* \cdot Z, & \sigma_1^* = C^{-1}, \\ \sigma_2^* = C^{J(\mathfrak{u}_r^*)}, & \sigma_3^* = C^{\nu' + \nu \cdot T^*}, \\ \sigma_4^* = dk_{\mathfrak{u}^*,T^*,2}, & \sigma_5^* = dk_{\mathfrak{u}^*,T^*,3}, \\ \mathfrak{m}^* = H(\sigma_0^*, \cdots, \sigma_5^*, \mathfrak{u}^*, \mathfrak{u}_r^*), & \sigma_6^* = dk_{\mathfrak{u}^*,T^*,1} \cdot C^{L(\mathfrak{m}^*)}. \end{array}$$

Finally, $\mathcal{C}$ sends $\sigma^* = (\sigma_0^*, \cdots, \sigma_6^*)$ to $\mathcal{A}$. It is obvious that along with the assumption that $\mathcal{C}$ does not fail, the signcryptext σ^* can pass the verification equation in the designcryption algorithm.

During the second phase, $\mathcal{A}$ may continue to make the queries to the challenger $\mathcal{C}$ as above, but with the restrictions in Subsection 3.2.

Eventually, $\mathcal{A}$ outputs a bit β'. If $\beta = \beta'$, then $\mathcal{C}$ output 1 (which means that $e(g,g)^{abc} = e(g,g)^z$), and 0 otherwise (which means that $e(g,g)^{abc} \neq e(g,g)^z$).

Type-2 Adversary: Let q be the maximum number of full private key generation queries, signcryption queries or designcryption queries. $\mathcal{C}$ randomly guesses $i^* \in [1, q]$ such that $\mathcal{A}$'s i^*-th query is the first query regarding $\mathfrak{u}^*$ among full private key generation queries, signcryption queries and designcryption queries. We assume that $\mathcal{C}$'s guess is right (It holds with $1/q$ and this is a loss of polynomial in λ). $\mathcal{C}$ keeps an integer count to count the number of full private key generation queries, signcrypiton queries or designcryption queries up to the current time.

Key Update Queries: For all nodes $\theta \in \mathsf{KUNode}(\mathsf{BT}, RL, T)$, $\mathcal{C}$ recalls S_θ from the node θ if it is defined. Otherwise, $\mathcal{C}$ chooses $S_\theta \stackrel{\$}{\leftarrow} \mathbb{G}$ and stores it in the node θ. $\mathcal{C}$ computes

$$(\tilde{D}_{\theta,0}, \tilde{D}_{\theta,1}) = (S_\theta^{-1} (v'v^T)^{s_\theta}, g^{s_\theta}),$$

where $s_\theta \stackrel{\$}{\leftarrow} \mathbb{Z}_p$. Output $ku_T = \{(\theta, \tilde{D}_{\theta,0}, \tilde{D}_{\theta,1})\}_{\theta \in \mathsf{KUNode}(\mathsf{BT}, RL, T)}$.

Revocation Queries: Upon receiving this query on $(\mathfrak{u}, T)$, $\mathcal{C}$ runs algorithm $\mathbf{REV}(mpk, \mathfrak{u}, T, RL, \mathsf{st}) \rightarrow RL$ and returns the updated revocation list RL.

Initial Private Key Generation Queries: $\mathcal{C}$ starts with receiving an identity $\mathfrak{u}$, sends it to $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$, and obtains (d_0, d_1) (if $F(\mathfrak{u}) = 0 \mod l_u$, then abort). $\mathcal{C}$ randomly chooses an undefined leaf node η and stores $\mathfrak{u}$ in η. For $\theta \in \mathsf{Path}(\eta)$, $\mathcal{C}$ recalls S_θ if it is defined. Otherwise, $S_\theta \xleftarrow{\$} \mathbb{G}$ and store it in the node θ. Then return the secret key

$$sk_{\mathfrak{u}} = \{(\theta, S_\theta \cdot d_0 \cdot (u' \prod_{i \in \mathcal{U}} u_i)^{r_\theta}, d_1 \cdot g^{r_\theta})\}_{\theta \in \mathsf{Path}(\eta)},$$

where r_θ is randomly chosen from $\mathbb{Z}_p$.

Full Private Key Generation Queries: For $\mathfrak{u} \neq \mathfrak{u}^*$ and all T, run initial private key generation query and key update query, and full private key generation algorithm (When $\mathsf{count} < i^*$, all $\mathfrak{u}$ are not equal to $\mathfrak{u}^*$. And when $\mathsf{count} = i^*$, $\mathcal{C}$ can identify $\mathfrak{u}^*$). If $\mathsf{KUNode}(\mathsf{BT}, RL, T) \cap \mathsf{Path}(\eta) = \emptyset$, then return $\perp$. Otherwise, choose $\theta \in \mathsf{KUNode}(\mathsf{BT}, RL, T) \cap \mathsf{Path}(\eta)$ and $r, s \xleftarrow{\$} \mathbb{Z}_p$ and return the decryption key

$$dk_{\mathfrak{u},T} = (d_0(u' \prod_{i \in \mathcal{U}} u_i)^{r_\theta + r}(v'v^T)^{s_\theta + s}, d_1 g^{r_\theta + r}, g^{s_\theta + s}).$$

For $\mathfrak{u} = \mathfrak{u}^*$ and $T \neq T^*$, $\mathcal{C}$ chooses random integers $r, s \xleftarrow{\$} \mathbb{Z}_p$ and outputs the decryption key

$$\begin{aligned} dk_{\mathfrak{u}^*,T} =& ((u' \prod_{i \in \mathcal{U}^*} u_i)^r g_2^{-\frac{\nu' + \nu T}{T - T^*}} (v'v^T)^s, g^r, g_2^{-\frac{1}{T - T^*}} g^s) \\ =& (g_2^a (u' \prod_{i \in \mathcal{U}^*} u_i)^r (v'v^T)^{s'}, g^r, g^{s'}), \end{aligned}$$

where $s' = -\frac{b}{T - T^*} + s$. Thus the decryption keys for $\mathfrak{u} = \mathfrak{u}^*$ and $T \neq T^*$ are identically distributed to those generated in the real experiment.

Signcryption Queries: When $\mathcal{A}$ queries the signcryption oracle for a message M, a time index T, a sender's identity $\mathfrak{u}$ and a receiver's identity $\mathfrak{u}_r$, the challenger $\mathcal{C}$ proceeds the same as signcryption oracle for Type-1 adversary.

Designcryption Queries: At any time $\mathcal{A}$ can perform a designcryption query for a ciphertext $\sigma = (\sigma_0, \sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6)$ associated with T, $\mathfrak{u}_s$ and $\mathfrak{u}$, $\mathcal{C}$ does the same as designcryption query for Type-1 adversary.

Challenge: $\mathcal{C}$ acts the same as the **Challenge** for Type-1 adversary.

Note that $\mathcal{C}$ does not query $\mathfrak{u}^*$ to $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$ oracle during the simulation. Type-2 adversary does not query the initial private key generation for $\mathfrak{u}^*$, but she may query full private key generation for $\mathfrak{u}^*$ and $T \neq T^*$. For full private key generation query for the challenged identity $\mathfrak{u}^*$ and time index $T \neq T^*$, $\mathcal{C}$ simulates queries without aid of $\mathrm{PKG}_{\mathrm{Wat}}(\cdot)$ oracle. The analysis about the challenge phase is same as the case for Type-1 adversary.

This completes the description of the simulation. It remains to analyze $\mathcal{C}$'s advantage. According to Claims 1 and 2 in [21], the distribution of all transcription between a challenger $\mathcal{C}$ and two types of adversaries $\mathcal{A}$ is identical to the real experiment. Furthermore, if $\mathcal{C}$ correctly guesses and does not abort, $\mathcal{C}$'s advantage is equal to $\mathcal{A}$'s advantage.

In the following, we firstly compute the probability of $\mathcal{C}$'s correct guess. In the setup phase, $\mathcal{C}$ randomly guesses the challenged time $T^* \in \mathcal{T}$, and so $\mathcal{C}$'s correct guess holds with $1/|\mathcal{T}|$. In the queries, $\mathcal{C}$ randomly guesses $i^* \in [1, q]$ such that $\mathcal{A}$'s i^*-th query is the first query regarding $\mathfrak{u}^*$ among initial private key generation queries, full private key generation queries, signcryption queries and designcryption queries, and so $\mathcal{C}$'s correct guess holds with $1/q$, where q is the maximum number of queries. It is obvious that $\mathcal{C}$'s guess T^* is totally independent from its guess i^*.

Then we consider the probability of $\mathcal{C}$'s not aborting. For the simulation to complete without aborting, we require that at most all initial private key generation queries, full private key generation queries on an identity $\mathfrak{u}$ have $F(\mathfrak{u}) \neq 0 \mod l_u$, that at most all signcryption queries $(\mathfrak{u}, \mathfrak{u}_r, M, T)$ have $F(\mathfrak{u}) \neq 0 \mod l_u$, that at most all designcryption queries $(\mathfrak{u}_s, \mathfrak{u}, \sigma, T)$ have $F(\mathfrak{u}) \neq 0 \mod l_u$ and that $F(\mathfrak{u}^*) \neq 0 \mod l_u$, $F(\mathfrak{u}_r^*) = 0 \mod l_u$ and $K(\mathfrak{m}^*) = 0 \mod l_m$. Similarly to the same technique in [16, 17, 20, 21, 30], we can bound the probability that $\mathcal{C}$ succeeds.

When we put the results for two types of adversaries together, we obtain a (polynomial-time) reduction from an adversary breaking IND-RIBSC-CCA security to a challenger against a DBDH instance with $\frac{1}{2q|\mathcal{T}|}$ reduction loss. Thus we obtain the following advantage of $\mathcal{C}$ in solving the DBDH problem:

$$Adv(\mathcal{C}) > \frac{\epsilon}{64q|\mathcal{T}|(q_{ipk} + q_{fpk} + q_s + q_d)^2(n_u + 1)^2(q_s + q_d)(n_m + 1)}.$$

Regarding the running time of $\mathcal{C}$, one can take into account the running time t of $\mathcal{A}$ and the multiplications, the exponentiations and the pairings computation time in the series of queries and the challenge processes above. For simplicity and due to the fact that the pairing is the most dominant component in pairing based cryptosystems, we only count the number of pairing operations required. Thus, we have the time complexity bound of $\mathcal{C}$:

$$t' \leq t + \mathcal{O}((q_s + 8q_d)\tau),$$

where τ is the time of pairing computation. Thus, the theorem follows. □

Theorem 2. *If there exists an adversary $\mathcal{A}$ attacking EUF-RIBSC-CMA security of the proposed RIBSC*

scheme, then there exists a challenger $\mathcal{C}$ breaking a CDH problem instance.

Proof. $\mathcal{C}$ receives a random instance (g, g^a, g^b) of the CDH problem. $\mathcal{C}$ uses $\mathcal{A}$ as a subroutine to solve that instance and plays the role of $\mathcal{A}$' challenger in the game of Definition 4. The simulation process is the same as that described in Theorem 1.

At the end of the game, $\mathcal{A}$ produces a ciphertext $\sigma^* = (\sigma_0^*, \cdots, \sigma_6^*)$ of message M^*, time index T^* and two identities $\mathfrak{u}_s^*$ and $\mathfrak{u}_r^*$. If σ^* is a valid forgery, then $(\sigma_1^*, \sigma_4^*, \sigma_5^*, \sigma_6^*)$ is a valid signature of $\mathfrak{u}_s^*$ on message $\mathfrak{m}^*$, where $\mathfrak{m}^* = H(\sigma_0^*, \cdots, \sigma_5^*, \mathfrak{u}_s^*, \mathfrak{u}_r^*)$. If $F(\mathfrak{u}_s^*) \neq 0 \mod l_u$ and $K(\mathfrak{m}^*) \neq 0 \mod l_m$, then $\mathcal{C}$ fails and stops. Otherwise, $\mathcal{C}$ computes and outputs

$$\frac{\sigma_6^* \cdot (\sigma_1^*)^{L(\mathfrak{m}^*)}}{(\sigma_4^*)^{J(\mathfrak{u}_s^*)} \cdot (\sigma_5^*)^{\nu' + \nu T^*}}$$

$$= \frac{g_2^a (u' \prod_{i \in \mathcal{U}_s} u_i)^{r_s} (v' v^{T^*})^{r_t} (m' \prod_{j \in \mathcal{M}} m_j)^k (g^{-k})^{L(\mathfrak{m}^*)}}{(g^{r_s})^{J(\mathfrak{u}_s^*)} (g^{r_t})^{\nu' + \nu T^*}}$$

$$= \frac{g_2^a (g^{J(\mathfrak{u}_s^*)})^{r_s} (g_1^{-T^*} g^{\nu'} g_1^{T^*} g^{\nu T^*})^{r_t} (g^{L(\mathfrak{m}^*)})^k (g^{-k})^{L(\mathfrak{m}^*)}}{(g^{r_s})^{J(\mathfrak{u}_s^*)} (g^{r_t})^{\nu' + \nu T^*}}$$

$$= g_2^a = g^{ab}$$

which is the solution to the given CDH problem.

This completes the description of the simulation. It remains to analyze the probability of $\mathcal{C}$ success. Similar to the probability analysis of $\mathcal{C}$ in the Theorem 1, if $\mathcal{C}$ correctly guesses and does not abort, $\mathcal{C}$'s advantage is equal to $\mathcal{A}$'s advantage. The probability of $\mathcal{C}$'s correct guess is $1/(2q|\mathcal{T}|)$. On the other hand, for the simulation to complete without aborting, we require that at most all initial private key generation queries, full private key generation queries on an identity $\mathfrak{u}$ have $F(\mathfrak{u}) \neq 0 \mod l_u$, that at most all signcryption queries $(\mathfrak{u}, \mathfrak{u}_r, M, T)$ have $F(\mathfrak{u}) \neq 0 \mod l_u$, that at most all designcryption queries $(\mathfrak{u}_s, \mathfrak{u}, \sigma, T)$ have $F(\mathfrak{u}) \neq 0 \mod l_u$, and that $F(\mathfrak{u}_s^*) = 0 \mod l_u$ and $K(\mathfrak{m}^*) = 0 \mod l_m$. According to the same technique in [16, 17, 20, 21, 30], we can bound the probability that $\mathcal{C}$ succeeds. Thus we obtain the following advantage of $\mathcal{C}$ in solving the CDH problem instance:

$$Adv(\mathcal{C}) > \frac{\epsilon}{32q|\mathcal{T}|(q_{ipk} + q_{fpk} + q_s + q_d)(n_u + 1)(q_s + q_d)(n_m + 1)}.$$

Regarding the running time of $\mathcal{C}$, we only count the number of pairing operations required and have the time complexity bound of $\mathcal{C}$:

$$t' \leq t + \mathcal{O}((q_s + 8q_d)\tau),$$

where τ is the time of pairing computation. Thus, the theorem follows. □

6 Conclusions

In this paper, we have proposed an identity-based signcryption scheme with revocation functionality. In the proposed scheme, the master key is randomly divided into two parts: one is used to construct the initial key, the other is used to generate the updated key. These keys are used to periodically generate full private/decryption keys for non-revoked users. Thus, our method can revoke users in time and resist key exposure. Furthermore, we prove that our scheme has the IND-CCA2 security under the DBDH hardness assumption and has the EUF-CMA property under the CDH hardness assumption in the standard model. Compared with the previous schemes, our scheme supports key re-randomization, reduces the key update complexity from $\mathcal{O}(n - r)$ to $\mathcal{O}(r \log \frac{n}{r})$ with n the number of users and r the number of revoked users, and is proved to be secure without using the random oracles.

Finally, we remark that some interesting problems remain to be solved. Our RIBSC scheme has long public parameters and loose security reduction. Therefore, constructing efficient and tightly secure RIBSC schemes is an open problem. Furthermore, one natural question is how to construct a generic transformation from IBSC to RIBSC. On the other hand, our scheme is based on bilinear pairings, but it is interesting to construct post-quantum secure schemes based on other mathematical structure such as lattices.

Acknowledgement

We would like to thank the anonymous reviewers for their valuable comments and suggestions. This work is supported by the National Natural Science Foundation of China under Grants No.61472470, 61472309, 61100229 and 61173151, the China Scholarship Council under Grants No. 201208610019, the Natural Science Foundation of Shaanxi Province under Grant No. 2014JM2-6091, the Scientific Research Plan Project of Education Department of Shaanxi Province under Grants No.12JK0852, and the State Key Laboratory of Information Security under Grants No. (GW0704127001).

References

[1] P. Barreto, B. Libert, N. McCullagh, J. Quisquater, "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps," in *Advances in Cryptology-ASIACRYPT'05*, pp. 515-532, Springer-Verlag, 2005.

[2] A. Boldyreva, V. Goyal, and V. Kumar, "Identity based encryption with efficient revocation," in *Proceedings of the 15th ACM Conference on Computer and Communications Security-CCS'08*, pp. 417-426, ACM Press, 2008.

[3] D. Boneh and M. Franklin, "Identity-based encryption from the Weil pairing," in *Advances in Cryptology-CRYPTO'01*, pp. 213-229, Springer-Verlag, 2001.

[4] X. Boyen, "Multipurpose identity-based signcryption: a swiss army knife for identity-based cryptography," in *Advances in Cryptology-CRYPTO'03*, pp. 383-399, Springer-Verlag, 2003.

[5] R. Canetti, O. Goldreich, and S. Halevi, "The random oracle methodology, revisted," *Journal of the ACM*, vol. 51, no. 4, pp. 557-594, 2004.

[6] H. Chen, Y. Li, and J. Ren, "A practical identity-based signcryption scheme," *International Journal of Network Security*, vol. 15, no. 6, pp. 484-489, 2013.

[7] J. Chen, H. Lim, S. Ling, H. Wang, and K. Nguyen, "Revocable identity-based encryption from lattices," in *17th Australasian Conference on Information Security and Privacy-ACISP'12*, pp. 390-403, Springer-Verlag, 2012.

[8] A. Dent and Y. Zheng, "Practical Signcryption," Berlin: Springer-Verlag, 2010.

[9] D. He, J. Chen, and R. Zhang, "An efficient identity-based blind signature scheme without bilinear pairings," *Computers & Electrical Engineering*, vol. 37, no. 4, pp. 444-450, 2011.

[10] M. S. Hwang, S. T. Hsu, and C. C. Lee, "A new public key encryption with conjunctive field keyword search scheme", *Information Technology and Controlitc*, vol. 43, no. 3, pp. 277-288, 2014.

[11] Z. Jin, Q. Wen, and H. Du, "An improved semantically-secure identity-based signcryption scheme in the standard model," *Computers & Electrical Engineering*, vol. 36, no. 3, pp. 545-552, 2010.

[12] J. Kar, "Provably secure online/off-line identity-based signature scheme for wireless sensor network," *International Journal of Network Security*, vol. 16, no. 1, 2014, pp. 29-39

[13] C. C. Lee, C. H. Liu, and M. S. Hwang, "Guessing attacks on strong-password authentication protocol", *International Journal of Network Security*, vol. 15, no. 1, pp. 64-67, 2013.

[14] W. T. Li, C. H. Ling, and M. S. Hwang, "Group rekeying in wireless sensor networks: a survey", *International Journal of Network Security*, vol. 16, no. 6, pp. 400-410, 2014.

[15] F. Li, Y. Liao, Z. Qin, "Further improvement of an identity-based signcryption scheme in the standard model," *Computers & Electrical Engineering*, vol. 38, no. 2, pp. 413-421, 2012.

[16] F. Li and T. Takagi, "Secure identity-based signcryption in the standard model," *Mathematical and Computer Modelling*, vol. 57, no. 11-12, pp. 2685-2694, 2013.

[17] X. Li, H. Qian, J. Weng, and Y. Yu, "Fully secure identity-based signcryption scheme with shorter signcryptext in the standard model," *Mathematical and Computer Modelling*, vol. 57, no. 3-4, pp. 503-511, 2013.

[18] B. Libert and D. Vergnaud, "Adaptie-ID secure revocable identity based encryption," in *Topics in Cryptology CT-RSA'09*, pp. 1-15. Springer-Verlag, 2009.

[19] S. Liu, Y. Long, and K. Chen, "Key updating technique in identity-based encryption," *Information Sciences*, vol. 181, no. 11, pp. 2436-2440, 2011.

[20] K. Paterson and J. Schuldt, "Efficient identity based signatures secure in the standard model," in *11th Australasian Conference Information Security and Privacy-ACISP'06*, pp. 207-222. Springer-Verlag, 2006.

[21] J. Seo and K. Emura, "Revocable identity-based encryption revisited: security model and construction," in *Public-Key Cryptography-PKC'13*, pp. 216-234. Springer-Verlag, 2013.

[22] J. Seo and K. Emura, "Efficient delegation of key generation and revocation functionalities in identity-based encryption," in *Topics in Cryptology-CT-RSA'13*, pp. 343-358. Springer-Verlag, 2103.

[23] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology-CRYPTO'84*, pp. 47-53. Springer-Verlag, 1985.

[24] Y. Tseng, T. Tsai, and T. Wu, "Provably secure revocable ID-based signature in the standard model," *Security and Communication Networks*, http://dx.doi.org/10.1002/sec.696, 2013.

[25] Y. Tseng, T. Tsai, and T. Wu, "A fully secure revocable ID-based encryption in the standard model," *Informatica*, vol. 23, no. 3, pp. 487-505, 2012.

[26] Z. Wang, L. Wang, S. Zheng, Y. Yang, and Z. Hu, "Provably secure and efficient identity-based signature scheme based on cubic residues," *International Journal of Network Security*, vol. 14, no. 1, pp. 33-38, 2012.

[27] B. Waters, "Efficient identity-based encryption without random oracles," in *Advances in Cryptology-EUROCRYPT'05*, pp. 114-127, Springer-Verlag, 2005.

[28] T. Wu, T. Tsai, and Y. Tseng, "A revocable ID-based signcryption scheme," *Journal of Information Hiding and Multimedia Signal Processing*,vol. 3, no. 2, pp. 240-251, 2012.

[29] H. Xiong, J. Hu, and Z. Chen, "Security flaw of an ECC-based signcryption scheme with anonymity," *International Journal of Network Security*, vol. 15, no. 4, pp. 317-320, 2013.

[30] Y. Yu, B. Yang, Y. Sun, and S Zhu, "Identity based signcryption scheme without random oracles," *Computer Standards & Interfaces*, vol. 31, no. 1, pp. 56-62, 2009.

[31] Y. Zheng, "Digital signcryption or how to achieve cost(signature & encryption) $\ll$ cost (signature) + cost(encryption)," in *Advances in Cryptology-CRYPTO'97*, pp. 165-179, Springer-Verlag, 1997.

Xiangsong Zhang received her B.S. degree from Henan Normal University, M.S., and Ph.D degrees from Xidian University, China, in 2004, 2007 and 2011, respectively. She is a lecturer at Xi'an Technological University, Xi'an, China. Her research interests include the mathematical problems of cryptography.

Zhenhua Liu received his B.S. degree from Henan Normal University, M.S., and Ph.D degrees from Xidian University, China, in 2000, 2003 and 2009, respectively. He is an associate professor at Xidian University, Xi'an, China. His research interests include public key cryptography and information security.

Yupu Hu received his B.S., M.S., and Ph.D. degrees from Xidian University, China, in 1982, 1987 and 1999, respectively. He is currently a professor at the State Key Laboratory of Integrated Services Network, Xidian University. His mainly research interests include cryptography and information security.

Tsuyoshi Takagi received his B.Sc. and M.Sc. degrees in mathematics from Nagoya University in 1993 and 1995, respectively. He had engaged in the research on network security at NTT Laboratories from 1995 to 2001. He received the Dr.rer.nat degree from Technische University Darmstadt in 2001. He was an Assistant Professor in the Department of Computer Science at Technische University Darmstadt until 2005, and a Professor at the School of Systems Information Science in Future University-Hakodate, Japan until 2009. He is currently a Professor in Graduate School of Mathematics, Kyushu University. His current research interests are information security and cryptography. Dr. Takagi is a memeber of International Association for Cryptologic Research (IACR).

A Novel Proactive Multi-secret Sharing Scheme

Bin Feng, Cheng Guo, Mingchu Li, and Zhihui Wang
(Corresponding author: Zhihui Wang)

School of Software Technology, Dalian University of Technology
No.8 Road, Jinzhou District, Dalian 116620, P. R. China
(Email: wangzhihui1017@gmail.com)
(Received Dec. 24, 2013; revised and accepted Jan. 16, 2015)

Abstract

A proactive secret sharing scheme is a method of sharing a secret among a set of participants. And, the corresponding shadows can be periodically renewed under the premise of never changing the shared secret. However, in the most existing proactive secret sharing schemes, only one secret can be shared during one secret sharing process. The proposed scheme describes PMSS, a new way to share multiple secrets with the proactive characteristic. In the proposed scheme, multiple secrets can be shared among many participants and shadows can be periodically renewed without changing these secrets. Meanwhile, based on the intractability of the discrete logarithm, the shadows provided by participants can be verified.

Keywords: Multi-secret sharing, proactive, secret sharing

1 Introduction

A secret sharing scheme is a technique to share a secret among a group of participants. It is mainly used to protect secret information from being lost, destroyed, or modified. In 1979, the first (t, n) threshold secret sharing schemes, based on Lagrange interpolating and liner project geometry, were proposed by Shamir [12] and Blakley [1], respectively. A secret sharing scheme contains a trusted dealer and n participants. The dealer divides the shared secret into n shadows and distributes them to these n participants through a secure channel. In the (t, n) threshold scheme, at least t honest participants can reconstruct the shared secret, but $(t-1)$ or fewer participants can obtain nothing about the secret. Therefore, even though some participants are compromised, if only the number of the compromised participants is less than t, they cannot cooperate to compute the secret.

Secret sharing schemes protect the secrecy and integrity of information by distributing the information over different locations (shadow holders). However, for long-lived and sensitive secrets, this protection may be insufficient. In many situations, such as cryptographic master keys, data files, legal documents, etc., a secret value needs to be stored for a long time. In these situations, given enough time, an adversary may compromise t servers one by one, obtain t shadows, and thus learn the shared secret. Thus, in order to protect the secrecy of the information, we need to periodically renew the shadows without changing the shared secret. To prevent such an attack, proactive secret sharing schemes are proposed. Proactive security for secret sharing was firstly suggested by Ostrovsky and Yung [10]. Herzberg et al. [5] further discussed proactive secret sharing schemes and gave a detailed proactive scheme. In their scheme, shadows can be periodically renewed (without changing the secret) in such a way that information gained by the adversary in one time period is useless for attacking the secret after the shadows are renewed. Hence, the adversary willing to learn the secret needs to break to all t locations during the same time period. Xu et al. [14] proposed a secret sharing scheme with periodic renewing shadows. In their scheme, because a trusted dealer distributes the secret information in the initialization phase and renewing phase, the amount of data communication and calculation are reduced. Zhou et al. [17] proposed in 2005 a proactive secret sharing (PSS) protocol for asynchronous systems, in which message delivery delays and processor execution speeds do not have fixed bounds. Their research extended the scope of PSS. In 2009, Ma and Ding [8] proposed a proactive verifiable linear integer secret sharing scheme. Linear integer secret sharing was firstly developed by Damgard and Thorbek [2], in which the shared secret was an integer, and each shadow was computed as an integer linear combination of the shared secret and some random integers selected by the dealer. In Ma and Ding's scheme, they introduced combinatorial structure into the proactive scheme to reduce the computational cost, meanwhile, they presented a verifiable method without a public key cryptosystem, which can prevent and detect cheating both from participants and dealers. In 2010, Schultz and Liskov [11] developed a new mobile proactive secret sharing scheme. Their scheme allows the set of participants to change.

However, in their schemes, only one secret can be shared during one secret sharing process. There are also

many papers [9, 13] that discuss proactive secret sharing. Our discussion will mainly follow the papers [5, 14].

Multi-secret sharing (MSS) schemes have been proposed by Harn [4] in 1995, in order to solve the problem that several secrets can be shared during one secret sharing process. In 2004, Yang, Chang and Hwang (YCH) [15] proposed a new MSS scheme based on Shamir's secret sharing scheme and the two-variable one-way function. Later, Li et al. [7] presented a new (t, n) threshold multi-secret sharing scheme. In 2007, Zhao et al. [16] proposed a practical verifiable multi-secret sharing scheme based on YCH and Hwang-Chang (HC) schemes [6]. The verification phase of Zhao's scheme is the same as that of HC scheme. So, a secure channel is not necessary at all. In 2008, Dehkordi and Mashhadi [3] also proposed a verifiable multi-secret sharing based on YCH, the intractability of Discrete Logarithm (DL) and RSA cryptosystem. In their scheme, there is not any need to a secure channel, and verifiable property is more efficient.

In this paper, we present a practical and efficient proactive multi-secret sharing scheme based on Xu's periodic renewing shadows secret sharing scheme. The verification phase is the same as the method introduced in [16]. In the proposed scheme, the shadows kept by participants can be updated periodically without changing these secrets. Meanwhile, multiple secrets can be shared during one secret sharing process.

To the best of our knowledge, no proactive multi-secret sharing schemes have been proposed in the literature to date. The proactive multi-secret characteristic of the proposed scheme is not available in the existing mechanisms, so the proposed scheme has the potential to work in many applications. The key features of our proposed proactive multi-secret sharing scheme are summarized below.

1) Shadows held in participants can be periodically updated without changing the shared secret;
2) The participants can shared multiple secrets during one secret sharing process;
3) Every participant can verify the validity of the shadows which he/she receives and other participants show;
4) The proposed scheme is efficiency and "$\oplus$" operation is low computing cost.

The remainder of this paper is organized as follows. In Section 2, we briefly review the multi-secret sharing scheme based on a two-variable one-way function proposed by Li et al., which is the major building block of our scheme. In the next section, we demonstrate the proposed scheme. Section 4 gives some security analysis. Finally, we presents our conclusions in Section 5.

2 Review of Li Scheme

In this section, we will review the Li scheme [7]. These schemes are based the threshold scheme proposed by Shamir [12] where the secret is embedded in an interpolating polynomial and each participant keeps a shadow associated to the interpolating polynomial.

Before presenting Li's scheme, we firstly give a definition of a two-variable one-way function $f(r, s)$ with two variables r and s. The two-variable one-way function has been used in Li's schemes.

Definition 1 [15]. Function $f(r, s)$ denotes any two-variable one-way function that maps any r and s onto a bit string $f(r, s)$ of a fixed length. The two-variable one-way function has several properties:

1) Given r and s, it is easy to compute $f(r, s)$;
2) Given s and $f(r, s)$, it is hard to compute r;
3) Having no knowledge of s, it is hard to compute $f(r, s)$ for any r;
4) Given s, it is hard to find two different values r_1 and r_2 such that $f(r_1, s) = f(r_2, s)$;
5) Given r and $f(r, s)$, it is hard to compute s;
6) Given pairs of r_i and $f(r_i, s)$, it is hard to compute $f(r_j, s)$, for $r_i \neq r_j$.

Then, we introduce a theorem that has been used in Li's scheme and will also be used in our scheme.

Theorem 1. *Given $(m+1)$ unknown variables $x_i \in GF(q)\,(i = 0, 1, ..., m)$, and m equations $x'_i = x_0 \oplus x_i\,(i = 1, 2, ..., m)$, where $GF(q)$ is a finite field. Here "$\oplus$" denotes exclusive-or bit by bit. Only the values of $x'_i\,(i = 1, 2, ..., m)$, are published. Given x_0, it is easy to find the remaining unknown symbols $x_i\,(i = 1, 2, ..., m)$; without any knowledge of x_0, it is computationally infeasible to determine the values of these unknown symbols.*

Proof. We prove Theorem 1 in two steps:

1) Given x_0, we can find $x_i\,(i = 1, 2, ..., m)$ easily by computing $x_i = x_0 \oplus x'_i$.
2) Without any knowledge of x_0, Theorem 1 is equal to solve m simultaneous equations, $x_0 \oplus x_i = x'_i\,(i = 1, 2, ..., m)$, with $(m+1)$ unknown symbols $x_i\,(i = 0, 1, ..., m)$. So, given these m equations, the values of these unknown symbols cannot be of the unknown symbols. The only thing for an adversary to do is to guess the doing it is only $1/q$ due to $x_i \in GF(q)$. If $GF(q)$ is a sufficiently large finite field, the successful probability tends to 0. So with no knowledge of x_0, it is computationally infeasible to determine the values of these unknown symbols.

□

Li's scheme can be described briefly as follows:

1) System parameters. Let $GF(q)$ denote a finite field, where q is a large prime number. All numbers are elements of $GF(q)$. The dealer randomly selects n

distinct integers $s_1, s_2, ..., s_n$, from $GF(q)$ as participants secret shadows and randomly selects n distinct integers,$u_i \in [n-t+2, q]$, for $i = 1, 2, ..., n$, as participants public identifiers. There are p secrets $k_1, k_2, ..., k_p$ to be shared among n participants. Let $f(r, s)$ be a two-variable one-way function defined above, which is used to compute pseudo shadows of participants.

2) Secret distribution. The trusted dealer performs the following steps to implement the secret distribution:

   a. Randomly choose an integer r and compute $f(r, s_i)$ for $i = 1, 2, ..., n$.

   b. Use n pairs of $(0, k_1)$ and $(u_1, f(r, s_1))$, $(u_2, f(r, s_2))$,..., $(u_n, f(r, s_n))$ to construct an nth degree polynomial $h(x) = a_0 + a_1 x + ... + a_n x^n$.

   c. Compute $z_i = h(i) \bmod q$ for $i = 1, 2, ..., n-t+1$ and $k_i' = k_1 \oplus k_i \bmod q$ where $i = 2, 3, ..., p$.

   d. Publish the values of r, z_1, z_2, ..., z_{n-t+1}, k_2', k_3', $\cdots$, k_p'.

3) Secret reconstruction. In order to reconstruct the shared secrets, at least t participants pool their pseudo shadows $f(r, s_i)$ for $i = 1', 2', ..., t'$. From these t pseudo shadows, we have t pairs of $(u_i, f(r, s_i))$ for $i = 1', 2', ..., t'$. With the knowledge of the public values $z_1, z_2, ..., z_{n-t+1}$, we can get $(n-t+1)$ pairs of (i, z_i) for $i = 1, 2, ..., n-t+1$. Therefore, there are $(n+1)$ pairs obtained altogether, by which the nth degree polynomial $h(x)$ can be uniquely determined. We use (X_i, Y_i) for $i = 1, 2, ..., n+1$ to denote these $(n+1)$ pairs, respectively. So $h(0)$ can be reconstructed through the following Lagrange interpolation polynomial:

$$h(0) = \sum_{i=1}^{n+1} Y_i \prod_{j=1, j\neq i}^{n+1} \frac{-X_j}{X_i - X_j} \bmod q.$$

We have $k_1 = h(0)$, subsequently, the remained $(p-1)$ secrets can be easily found by for $i = 2, 3, ..., p$, respectively.

3 The Proposed Scheme

In this section we will propose a new (k, n) threshold proactive multi-secret sharing scheme that are based on Xu's proactive secret sharing scheme. The sharing multiple secrets method is based on Li's multi-secret sharing scheme.

Like Li's scheme, our scheme is based on Theorem 1, and the scheme can be described as follows:

1) System parameters. The dealer (denoted as U_D) first creates a public notice board (NB), whose properties are as same as those in Type 1 scheme. We assume that $EP_i(.)$ are the public key encryption algorithm using the participants public key and the encryption process are secure and reliable. Let q be a large prime, and let $GF(q)$ denote a finite field, such that computing discrete logarithms in this field is infeasible and all the numbers are elements in the finite field $GF(q)$. Let g is the generator of the finite field $GF(q)$, $g \in GF(q)$. The dealer randomly selects n distinct integers, $u_i \in GF(q)$, for $i = 1, 2, ..., n$, as participants public identifiers. Without loss of generality, we also assume that there are n participants, $U_1, U_2, ..., U_n$, sharing p secrets $P_1, P_2, ..., P_p$, $P_1, P_2, ..., P_p \in GF(q)$.

   The notations utilized in this paper are listed in Table 1.

2) Secret distribution. The shadows computed in period t are denoted by using the superscript (t), i.e., $y_i{}^{(t)}, t = 0, 1, ...$ The polynomial corresponding to these shadows is denoted $f^{(t)}(.)$. At the beginning of the time period, the trusted dealer executes the following steps:

   a. Construct a $(k-1)$th degree polynomial $f^{(0)}(x) = a_0 + a_1^{(0)} x + \cdots + a_{k-1}^{(0)} x^{k-1} \bmod q$, where $a_0 = P_1$ and are randomly chosen from $GF(q)$.

   b. Compute $y_i^{(0)} = f^{(0)}(u_i) \bmod q, (i = 1, 2, ..., n)$, and distributes $y_i^{(0)}$ to every participants U_i for $i = 1, 2, ..., n$, over a security channel.

   c. The trusted dealer compute $G_i^{(0)} = g^{y_i^{(0)}} \bmod q$, for $i = 1, 2, ..., n$, and publish $\left\{g, G_i{}^{(0)} (i = 1, 2, ..., n)\right\}$ on the notice board.

   d. Compute $P_i' = P_1 \oplus P_i \bmod q$, for $i = 2, 3, ..., p$, and publish $\{P_i' (i = 2, 3, ..., p)\}$ on the notice board.

3) Shadow renewal. To renew the shadows at period t, $t = 1, 2, ...$, the renewed protocol will be performed as follows:

   a. Randomly select $k-1$ integers from the finite field $GF(q)$,$\varepsilon_1^{(t)}, \varepsilon_2^{(t)}, \ldots, \varepsilon_{k-1}^{(t)}$, and construct an polynomial $\varepsilon^{(t)}(x) = \varepsilon_1^{(t)} x + \varepsilon_2^{(t)} x^2 + \ldots + \varepsilon_{k-1}^{(t)} x^{k-1} \bmod q$.

   b. Compute $u_i^{(t)} = \varepsilon^{(t)}(i)$,$v_i^{(t)} = EP_i(u_i^{(t)})$,$i = 1, 2, ..., n$, and $G_i^{(t)} = g^{y_i^{(t-1)}} \cdot g^{u_i^{(t)}} \bmod q$, $(i = 1, 2, ..., n)$, and publish $\left\{v_i^{(t)}, G_i^{(t)} (i = 1, 2, ..., n)\right\}$ on the notice board.

   c. At time period t, each participant U_i will decrypt $v_i^{(t)} = EP_i(u_i^{(t)})$ using its own private key, and it will be able to obtain $u_i^{(t)}$. By the linearity of the polynomial evaluation operation, we get the renewal of the shadows $y_i^{(t)} \leftarrow y_i^{(t-1)} +$

Table 1: The notations

$u_1, u_2, ..., u_n$	participants' public identifiers
k	threshold value
$t, t = 1, 2, 3, ...$	time period
$U_1, U_2, ..., U_n$	n participants
$P_1, P_2, ..., P_p$	p shared secrets
$\varepsilon^{(t)}(x)$	the updated polynomial at t period time
$u_i^{(t)}$	the updated value at t period on ith shadow

$u_i^{(t)}$ according to $f^{(t)}(x) \leftarrow f^{(t-1)}(x) + \varepsilon^{(t)}(x)$, and destroy $y_i^{(t-1)}$.

4) Secret reconstruction. At time period t, without losing generality, suppose k participants U_i, $i = 1, 2, ..., k$, pool their shadows $y_i^{(t)*}$ (for $i = 1, 2, ..., k$), every participant U_i can check whether others secret shadows are valid by the following equations:

$$g^{y_i^{(t)*}} = G_i^{(t)} \bmod q.$$

Then, with the knowledge of k pairs $\left(u_1, y_1^{(t)}\right)$, $\left(u_2, y_2^{(t)}\right)$,..., $\left(u_k, y_k^{(t)}\right)$, the $(k-1)$th polynomial $f^{(t)}(x)$ can be uniquely determined as

$$f^{(t)}(x) = \sum_{i=1}^{k} y_i^{(t)} \prod_{j=1, j\neq i}^{k} \frac{x - u_j}{u_i - u_j}.$$

We have $P_1 = f^{(t)}(0) = a_0$, subsequently, the remained $(p-1)$ secrets can be easily found by $P_i = P_1 \oplus P_i' \bmod q$, for $i = 2, 3, ..., p$, respectively.

4 Security Analysis

In this paper, we proposed two proactive multi-secret sharing schemes based on Xu's periodic renewing shadows secret sharing scheme. The security of the proposed scheme is based on the security of Li's multi-secret sharing scheme and discrete logarithm problem. In the following, several possible attacks are investigated to demonstrate the security of the proposed scheme.

Attack 1. $(t-1)$ or fewer participants try to recover secrets.

Analysis: The security of the proposed scheme, similar to the security of Shamir's scheme is based on the Lagrange interpolation polynomial. And, any $(k-1)$ or fewer participants cannot compute the polynomial $f(x)$ and obtain anything about the secrets.

Attack 2. A malicious adversary may try to reveal k secret shadows of participants in a long time.

Analysis: According to the characteristic of proactive secret sharing and the description of the proposed scheme, the shadows kept by participants can be updated periodically. That is, the shadows will be changed at regular intervals. Therefore, a malicious adversary need to reveals k secret shadows of participants in a period time. Otherwise, if only $(k-1)$ secret shadows are revealed in a period time, and another secret shadow is revealed in the next period, the malicious adversary cannot obtain the shared secret since one secret shadow have been changed. The revealed k secret shadows cannot reconstruct the $(k-1)$th degree polynomial. The proactive characteristic of the proposed scheme increases the degree of attack difficulty.

Attack 3. A malicious participant tries to pool a fake pseudo shadow s_i' to cheat other cooperators.

Analysis: In the process of reconstructing the shared secrets, we usually assumed that the involved participants must provide their shadows honestly when they want to cooperate to recover the secrets. However, this assumption is impractical. A malicious participant can pool a fake pseudo shadow s_i' to other participants. This will lead to that other participants providing their shadows honestly cannot reconstruct the shared secrets from the $(t-1)$ corrected shadows, and only the malicious participant can obtain the secrets. In the proposed scheme, we present a verification method based on the intractability of the discrete logarithm. Every participant U_i can check whether others secret shadows $y_j^{(t)*}s$(for $j = 1, 2, ..., k, j \neq i$) are valid by the following equations:$g^{y_i^{(t)*}} = G_i^{(t)} \bmod q$.

5 Conclusions

In this paper, we present a novel proactive multi-secret sharing scheme based on Xu et al.'s scheme and the intractability of the discrete logarithm. The scheme realizes the property of proactive. That is, shadows held by every participant can be updated in a period time. As to an adversary, if he wants to attack the shared secret, he must compromise t servers one by one in one time period. How-

ever, it is difficult. Compared with the previous works, in our scheme, multiple secrets can be shared during one secret sharing process. In addition, in the reconstruction phase, the shadows can be verified.

Acknowledgments

This paper is supported by the National Science Foundation of China under grant No. 61272173, 61100194, 61401060 and the general program of Liaoning Provincial Department of Education Science Research under grants L2014017.

References

[1] G. Blakley, "Safeguarding cryptographic keys," in *In Proceedings of the National Computer Conference*, pp. 313–317, Montvale: NCC, 1979.

[2] I. Damgard and R. Thorbek, "Linear integer secret sharing and distributed exponentiation," in *In Proceedings of 9th International Conference on Theory and Practice of Public-Key Cryptography (PKC 2006)*, pp. 75–90, New York, USA, April 2006.

[3] M.H. Dehkordi and S. Mashhadi, "An efficient threshold verifiable multi-secret sharing," *Computer Standards & Interfaces*, vol. 30, no. 3, pp. 187–190, 2008.

[4] L. Harn, "Efficient sharing (broadcasting) of multiple secret," *IEE Proceedings Computers and Digital Technique*, vol. 142, no. 3, pp. 237–240, 1995.

[5] A. Herzberg, S.L. Jarecki, and H. Krawczyk et al., "Proactive secret sharing or: How to cope with perpetual leakage," in *In Advances in Cryptology-Crypto95*, pp. 339–352, Berlin: Springer-Verlag, 1995.

[6] R.J. Hwang and C.C. Chang, "An on-line secret sharing scheme for multi secrets," *Computer Communications*, vol. 21, no. 13, pp. 1170–1176, 1998.

[7] H.X. Li, C.T. Cheng, and L.J. Pang, "A new (t, n)- threshold multi-secret sharing scheme," in *In Proceedings of the 2005 International Conference on Computational Intelligence and Security (CIS 2005), Part II, LNAI 3802*, pp. 421–426, 2005.

[8] C.G. Ma and X.F. Ding, "Proactive verifiable linear integer secret sharing scheme," in *In Proceedings of 11th International Conference on Information and Communications Security (ICICS 2009)*, pp. 439–448, Beijing, China, 2009.

[9] V. Nikov, S. Nikova, B. Preneel, and J. Vandewalle, "Applying general access structure to proactive secret sharing schemes," in *In Proceedings of the 23th Symposium on Information Theory*, pp. 29–31, Benelus, 2002.

[10] R. Ostrovsky and M. Yung, "How to withstand mobile virus attacks," in *In Proceeding of 10th the ACM Symposium on Principles of Distributed Computing (PODC'91)*, pp. 51–59, New York, USA, 1979.

[11] D. Schultz and B. Liskov, "Mpss: Mobile proactive secret sharing," *ACM Transactions on Information and System Security*, vol. 13, no. 4, pp. 34–65, 2010.

[12] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[13] D.R. Stinson and R. Wei, "Unconditionally secure proactive secret sharing scheme with combinatorial structure," in *In Proceedings of the 6th International Workshop on Selected Areas in Cryptography (SAC99)), Springer-Verlag, LNCS 1758*, pp. 200–214, London, UK, 1999.

[14] C.X. Xu, S.M., and G.Z. Xiao, "A secret sharing scheme with periodic renewing to identify cheaters," *Chinese Journal of Computers*, vol. 25, no. 6, pp. 657–660, 2002.

[15] C.C. Yang, T.Y. Chang, and M.S. Hwang, "A (t, n) multi-secret sharing scheme," *Applied Mathematics and Computation*, vol. 151, no. 2, pp. 483–490, 2004.

[16] J.J Zhao, J.Z. Zhang, and R. Zhao, "A practical verifiable multi-secret sharing scheme," *Computer Standards & Interfaces*, vol. 29, no. 1, pp. 138–141, 2007.

[17] L.D. Zou, F.B. Schneider, and R.V. Renesse, "Apss: Proactive secret sharing in asynchronous systems," *ACM Transaction on Information and System Security*, vol. 8, no. 3, pp. 259–286, 2005.

Bin Feng received the BS degree in Computer Science and Technology in 2002 from the LiaoCheng University, Shandong, China, and the MS degree in software engineering in 2006 from the Dalian University of Technology, Dalian, China. He has been an assistant in TaiShan College among 2002-2004. He is currently a full engineer of Computer Science at DaLian University of Technology (DLUT) (Dalian, China), where he has been since September 2006. Since 2011 he is currently pursuing his PhD degree in computer software and theory from the Dalian University of Technology, Dalian, China. His research interests include data hiding, image processing, network and information security.

Cheng Guo received the B.S. degree in computer science from Xi'an University of Architecture and Technology in 2002. He received the M.S. degree in 2006 and his Ph.D in computer application and technology, in 2009, both from the Dalian University of Technology, Dalian, China. From July 2010 to July 2012, he was a post doc in the Department of Computer Science at the National Tsing Hua University, Hsinchu, Taiwan. Since 2013, he has been an associate professor in the School of Software Technology at the Dalian University of Technology. His current research interests include information security and cryptology.

Mingchu Li received the B.S. degree in mathematics, Jiangxi Normal University and the M.S. degree in applied science , University of Science and Technology Beijing in 1983 and 1989, respectively. He worked for University of Science and Technology Beijing in the capacity of

associate professor from 1989 to 1994. He received his doctorate in Mathematics, University of Toronto in 1997. He was engaged in research and development on information security at Longview Solution Inc, Compuware Inc. from 1997 to 2002. From 2002, he worked for School of Software of Tianjin University as a full professor, and from 2004 to now, he worked for School of Software Technology of Dalian University of Technology as a full Professor, Ph.D. supervisor, and vice dean. His main research interests include theoretical computer science and cryptography.

Zhi-Hui Wang was born in Inner Mongolia in 1982. She received her B.S. degree in software engineering from the North Eastern University, Shenyang in 2004, M.S. degree in software engineering from Dalian University of Technology (DUT), Dalian in 2007, and Ph.D. degree in computer software and theory from DUT in 2010. Since 2014, she has been an associated professor in the School of Software Technology at the Dalian University of Technology. Her current research interests include information hiding and image processing.

On the Security of A Provably Secure Certificate Based Ring Signature Without Pairing

Ji Geng[1], Hu Xiong[1,2], Fagen Li[1], and Zhiguang Qin[1]
(Corresponding author: Hu Xiong)
School of Computer Science and Engineering, University of Electronic Science and Technology of China[1]
No. 4, North Jianshe Road, Chenghua District, chengdu, Sichuan 610054, China
State Key Laboratory of Information Security, Institute of Software, Chinese Academy of Sciences[2]
No. 19 Yuquan Road, Shijingshan District, Beijing 100190, China
(Email: xionghu.uestc@gmail.com)
(Received Feb. 11, 2014; revised and accepted Nov. 6, 2014)

Abstract

Featured with anonymity and spontaneity, ring signature has been widely adopted in various environments to offer anonymous authentication. To simplify the certificate management in traditional public key infrastructure (PKI) and solve the inherent key escrow problem in the Identity-based cryptography, Qin *et al.* propose a pairing-free ring signature scheme in the certificate-based cryptosystem recently. Unfortunately, we demonstrate that their scheme is not secure against the malicious certificate authority (CA) and key replacement attacks by giving concrete attack. Concretely, a malicious certificate authority (CA) can forge a signature on arbitrary message in name of any user's identity and a uncertified user is also able to forge a message.

Keywords: Certificate-based signature, forgery attack, ring signature

1 Introduction

Ring signature [19], which allows a user to issue a signature on behalf of a group of possible signers (ring), has been introduced by Rivest *et al.* in Asiacrypt 2001. The resulting ring signature can convince a verifier that one member in the ring indeed signed the message without revealing the real identity of the actual signer. Different from group signature [4], there is not group manager in the ring signature to handle the enrollment and revocation of the ring members. Specifically, the actual signer can conscript the other ring members to form the ring without their consent. Featured with anonymity and spontaneity, ring signature has been widely adopted to offer anonymous authentication in various scenarios. As a representative example, portable devices or mobile applications in the infrastructure-less mobile ad hoc networks (MANETs) can share data with the other participants to behave in intelligent manners. It is challenging to secure MANETs due to the openness and lack of the central authority. Taking MANETs as an example, there are several security requirements a practical system must satisfy, including:

- **Authenticity**: In the situation of MANETs, the data sent from the other participants would be misleading if it is forged by adversaries. Thus, it is desirable to authenticate the receiving data to resist the attacks mounted by the outside adversaries;
- **Anonymity**: The shared data in MANETs contains vast information of users, from which one can extract the location of the target users, etc. Therefore, any failures with regard to the privacy preserving may lead to the reluctance from the users to share data with others;
- ***Ad hoc***: In the MANETs, the formation of a group where the actual user hidden from is spontaneous due to the lack of central authority; and
- **Efficiency**: Taking the huge number of users in MANETs into account, a practical system must lower the computation and communication overhead as much as possible.

Ring signature can be viewed as an efficient solution on the aforementioned situation where the data authenticity and anonymity are expected. In addition to the data sharing in the MANETs (instantiated as Vehicular *ad hoc* networks [21] and wireless sensor networks [11]), ring signature can also be deployed in other environments such as routing protocol [16] and electronic auction protocol [22, 23]. Furthermore, ring signatures can also be viewed as the building block of concurrent signatures [5, 7] and optimistic fair exchange [12]. The survey of ring signatures can be found in [6, 25].

Table 1: Notations

Notations	Descriptions
MANETs:	**M**obile **A**d hoc **NET**work**s**
PKI:	**P**ublic **K**ey **I**nfrastructure
ID-PKC:	**Id**entity-based **P**ublic **K**ey **C**ryptography
CB-PKC:	**C**ertificate-**B**ased **P**ublic **K**ey **C**ryptography
CA:	**C**ertificate **A**uthority
PKG:	**P**rivate **K**ey **G**enerator
ID_i:	The identity of the user i
(upk_{ID_i}, usk_{ID_i}):	The user public/secret key pair of the user i
(R, k_i):	The certificate of the user i
$L_{ID} = \{ID_1, \cdots, ID_n\}$:	The identity set of n ring members
$L_{upk} = \{upk_{ID_1}, \cdots, upk_{ID_n}\}$:	The public key set of n ring members
$\mathcal{G}$:	A multiplicative group with order q, where q is prime number.
g:	A random generator chosen from $\mathcal{G}$
π_{u_i}:	The proof-of-knowledge (PoK) such that $PK\{(u_i) : U_1 = g^{u_i} \wedge U_2 = X^{u_i}\}$
H:	Secure hash function such as $H : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$

In traditional public key infrastructure (PKI), a semi-trusted certificate authority (CA) is involved to generate a digital certificate to bind the public key and the corresponding identity. The management overhead of the public key certificate is considered to be costly. To simplify the certificate management, the notion of Identity-based public key cryptography (ID-PKC) has been introduced [20]. In ID-PKC, the public key of user can be easily derived from its digital identity such as email address or telephone number. To enjoy the merits of ID-PKC, the notion of ID-based ring signature schemes along with the extensions have been extensively investigated [2, 8, 24]. Unfortunately, a fully-trusted private key generator (PKG) is needed to generate the private key for each user according to its respective identity in ID-PKC. Thus, the key escrow problem is introduced into ID-PKC.

To simplify the heavy certificate management in traditional PKI and solve the key escrow problem in ID-PKC, a new paradigm, certificate-based public key cryptography (CB-PKC), is proposed by Gentry [10]. In CB-PKC, each user will generate the public and private key pair itself and the CA will issue the certificate using the private key generation algorithm in ID-PKC. In this way, the certificate will be used as part of the private key and third-party queries on certificate status in traditional PKI has already been eliminated in CB-PKC. Au *et al.* [1] introduce the notion of ring signature in the CB-PKC setting to enjoy the merits of CB-PKC and ring signature together, and further proposed a concrete certificate based ring signature based on bilinear pairing.

In order to remove the costly bilinear pairing operation, Qin *et al.* [18] proposed a pairing free certificate-based ring signature recently. Furthermore, they claimed that their scheme is provably secure in the random oracle model assuming the Discrete Logarithm assumption holds. Unfortunately, in this paper, we show that their scheme cannot achieve the claimed security by demonstrating two forgery attacks. Concretely, a malicious CA equipped with the master secret key can forge a valid signature on arbitrary message. In addition, a uncertified entity without a certificate issued by CA can also forge a valid signature on arbitrary message but replacing the public keys.

The rest of this paper is organized as follows. In Section 2, we review Qin *et al.*'s pairing-free certificate based ring signature scheme. In Section 3, we show that Qin *et al.*'s scheme is not secure and analyze the basic reason for the attack. Finally, the conclusions are given in Section 4.

2 Review of Qin et al.'s Scheme

Qin *et al.*'s certificate based ring signature scheme [18] is based on certificate-based signature scheme in [17] and ID-based ring signature scheme in [13]. The notation used in [18] is listed in Table 1 to improve the readability and we review Qin *et al.*'s scheme as follows.

1) Setup: Let $\mathcal{G}$ be a multiplicative group with order q. The CA selects a random generator $g \in \mathcal{G}$ and randomly chooses $x \in_R \mathbb{Z}_q^*$ as the master secret key. It sets $X = g^x$. Let $H : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$ be a cryptographic hash function. The public parameters are given by params=$(\mathcal{G}, q, g, X, H)$ The multiplicative group can be implemented on the Elliptic curve cryptography (ECC). According to [3], to achieve the comparable level of security to 1024-bits RSA, the Koblitz elliptic curve $y^2 = x^3 + ax^2 + b$ defined on $\mathbb{F}_{2^{163}}$ providing ECC group can be adopted. Here, a is equal to 1 and b is a 163-bit random prime. Thus, the size of the element in group $\mathcal{G}$ (the master public key and the user public key) is assumed to be 163-bit.

2) UserKeyGen: User ID_i selects a secret value $u_i \in \mathbb{Z}_q^*$

as his secret key usk_{ID_i}, and computes his public key $upk_{ID_i} = (g^{u_i}, X^{u_i}, \pi_{u_i})$ where π_{u_i} is the following non-interactive proof-of-knowledge (PoK):

$$PK\{(u_i) : U_1 = g^{u_i} \wedge U_2 = X^{u_i}.\}$$

The subscript of u_i has been inadvertently omitted in [18]. This omission has been corrected to be consistent.

3) CertGen: Let $\tilde{h}_i = H(upk_{ID_i}, ID_i)$ for user ID_i with public key upk_{ID_i} and binary string ID_i which is used to identify the user. To generate a certificate for user ID_i, the CA randomly chooses $r \in_R \mathbb{Z}_q^*$, computes $R = g^r$ and $k_i = r^{-1}(\tilde{h}_i - xR) \bmod q$. The certificate is (R, k_i). Note that a correctly generated certificate should satisfy the following equality:

$$R^{k_i} X^R = g^{\tilde{h}_i}.$$

4) Ring-Sign: Suppose there is a group of n users whose identities form the set $L_{ID} = \{ID_1, \cdots, ID_n\}$, and their corresponding public keys form the set $L_{upk} = \{upk_{ID_1}, \cdots, upk_{ID_n}\}$. To sign a message $m \in \{0,1\}^*$ on behalf of the group, the actual signer, indexed by s using the secret key usk_{ID_s} and the certificate $cert_{ID_s}$, performs the following steps.

   a. For each $i \in \{1, \cdots, n\} \setminus \{s\}$, selects $y_i \in_R \mathbb{Z}_q^*$ uniformly at random and computes $Y_i = R^{-y_i}$.
   b. Compute $h_i = H(m \| L_{upk} \| L_{ID} \| Y_i)$ for $i \in \{1, \cdots, n\} \setminus \{s\}$.
   c. Choose $y_s \in_R \mathbb{Z}_q^*$, computes $Y_s = R^{-y_s} \prod_{i \neq s} (g^{u_i})^{h_i \tilde{h}_i} \prod_{i \neq s} (X^{u_i})^{-h_i R}$.
   d. Compute $h_s = H(m \| L_{upk} \| L_{ID} \| Y_s)$.
   e. Compute $z = (\sum_{i=1}^n y_i + h_s k_s u_s) \bmod q$.
   f. Output the ring signature on m as $\sigma = \{Y_1, \cdots, Y_n, R, z, \pi_{u_1}, \cdots, \pi_{u_n}\}$. Though $\{R, \pi_{u_1}, \cdots, \pi_{u_n}\}$ is needed in the Verify algorithm, it has been inadvertently omitted in the signature of [18]. This omission has been corrected to be consistent.

5) Verify: To verify a ring signature $\sigma = \{Y_1, \cdots, Y_n, R, z, \pi_{u_1}, \cdots, \pi_{u_n}\}$ on a message m with identities in L_{ID} and corresponding public keys in L_{upk}, the verifier performs the following steps.

   a. Check whether π_{u_i} is a valid PoK. If not, outputs $\perp$, Otherwise, run the next step.
   b. Compute $h_i = H(m \| L_{upk} \| L_{ID} \| Y_i)$ and $\tilde{h}_i = H(upk_{ID_i}, ID_i)$ for all $i \in \{1, \cdots, n\}$.
   c. Check whether
   $$\prod_{i=1}^n (g^{u_i})^{h_i \tilde{h}_i} \stackrel{?}{=} R^z Y_1 \cdots Y_n \prod_{i=1}^n (X^{u_i})^{h_i R}$$
   d. Accept the ring signature as valid and outputs 1 if the above equation holds, otherwise, output 0.

3 Analysis of Qin *et al.*'s Scheme

It is non-trivial to devise secure certificate-based encryption/signature scheme since the certificate of the user will no longer be used to certify the corresponding public key instead it will be implicitly used as part of private key in the decryption/signing algorithm. In fact, several certificate-based encryption scheme [26] and certificate-based signature scheme [14, 17] have been shown to be insecure against the attacks mounted by an uncertified entity or malicious CA respectively [9, 15, 27]. Motivated by these attacks, we observe that Qin *et al.*'s certificate-based ring signature [18] is also insecure against the forgery attack. Comparing with the existing attack algorithms with respect to certificate based encryption/signature schemes [9, 15, 27], our work mainly focus on the insecurity of the certificate-based ring signature, where a large number of users are involved in the process of the signature generation.

According to [14, 15, 18, 27], two different types of attacks by the malicious CA and by an uncertified user should be considered in CB-PKC. On the one hand, the malicious CA, who has the master secret key, cannot obtain the user secret key and mount the public key replacement attack. On the other hand, the uncertified user can replace public keys of any entities in the system, but is not allowed to obtain the target user's certificate.

3.1 Malicious CA Attack on Qin *et al.*'s Scheme

Given a ring signature $\sigma = \{Y_1, \cdots, Y_n, R, z, \pi_{u_1}, \cdots, \pi_{u_n}\}$ with the identities in $L_{ID} = \{ID_1, \cdots, ID_n\}$ and corresponding public keys in $L_{upk} = \{upk_{ID_1}, \cdots, upk_{ID_n}\}$, the CA equipped with the master key x can forge a valid signature on arbitrary message m' as follows:

1) Randomly choose $j \in_R \{1, \cdots, n\}$.
2) Compute $\tilde{h}_j = H(upk_{ID_j}, ID_j)$.
3) Compute $R' = x^{-1}\tilde{h}_j$, where x is the master key.
4) For each $i \in \{1, \cdots, n\} \setminus \{j\}$, selects $y_i' \in_R \mathbb{Z}_q^*$ uniformly at random and computes $Y_i' = (R')^{-y_i'}$.
5) Compute $h_i' = H(m' \| L_{upk} \| L_{ID} \| Y_i')$ for $i \in \{1, \cdots, n\} \setminus \{j\}$.
6) Choose $y_j' \in_R \mathbb{Z}_q^*$, computes $Y_j' = (R')^{-y_j'} \prod_{i \neq j} (g^{u_i})^{h_i' \tilde{h}_i} \prod_{i \neq j} (X^{u_i})^{-h_i' R'}$.
7) Compute $z' = \sum_{i=1}^n y_i' \bmod q$.
8) Output the ring signature on m' as $\sigma = \{Y_1', \cdots, Y_n', R', z', \pi_{u_1}, \cdots, \pi_{u_n}\}$.

The following equations show that the signature $\sigma = \{Y'_1, \cdots, Y'_n, R', z', \pi_{u_1}, \cdots, \pi_{u_n}\}$ is valid.

$$\begin{aligned}
\prod_{i=1}^{n}(g^{u_i})^{h_i\tilde{h}_i} &= \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} g^{xu_jh'_jx^{-1}\tilde{h}_j} \\
&= \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} g^{xu_jh'_jR'} \\
&= \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} X^{u_jh'_jR'} \\
&= (R')^{\sum_{i=1}^{n} y'_i} \prod_{i\neq j}(R')^{-y'_i} \cdot (R')^{-y'_j} \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} \\
&\quad \prod_{i\neq j}(X^{u_i})^{-h'_iR'} \prod_{i=1}^{n}(X^{u_i})^{h'_iR'} \\
&= (R')^{z'} Y'_1 \cdot \cdots \cdot Y'_n \prod_{i=1}^{n}(X^{u_i})^{h'_iR'}.
\end{aligned}$$

3.2 Key Replacement Attack on Qin *et al.*'s Scheme

In the following, we show that the scheme is not against an uncertified entity attack. Concretely, an entity without a certificate issued by CA can forge a valid signature on arbitrary message m' by replacing the public keys. The attack is depicted as follows:

1) Randomly choose $r \in_R \mathbb{Z}_q^*$ and compute $R' = g^r$.
2) Randomly choose $j \in_R \{1, \cdots, n\}$.
3) For each $i \in \{1, \cdots, n\} \setminus \{j\}$, selects $y'_i \in_R \mathbb{Z}_q^*$ uniformly at random and computes $Y'_i = g^{-y'_i}$.
4) Compute $h'_i = H(m' \| L_{upk} \| L_{ID} \| Y'_i)$ for $i \in \{1, \cdots, n\} \setminus \{j\}$.
5) Choose $y'_j \in_R \mathbb{Z}_q^*$, computes $Y'_j = X^{-aR'} g^{-y'_j} \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} \prod_{i\neq j}(X^{u_i})^{-h'_iR'}$.
6) Compute $\tilde{h}_j = H(upk_{ID_j}, ID_j)$.
7) Compute $u_j = \frac{a}{h_j}$ as the secret key of user with identity ID_j, and set $upk_{ID_j} = (g^{u_j}, X^{u_j}, \pi_{u_j})$ as the public key of this user, where π_{u_j} is the following non-interactive proof-of-knowledge (PoK):

$$PK\{(u_j). : U_1 = g^{u_j} \wedge U_2 = X^{u_j}\}.$$

8) Compute $z' = \frac{ah'_j}{r} + \frac{\sum_{i=1}^{n} y'_i}{r} \bmod q$.
9) Output the ring signature on m' as $\sigma = \{Y'_1, \cdots, Y'_n, R', z', \pi_{u_1}, \cdots, \pi_{u_n}\}$.

The following equations show that the signature $\sigma = \{Y'_1, \cdots, Y'_n, R', z', \pi_{u_1}, \cdots, \pi_{u_n}\}$ is valid.

$$\begin{aligned}
\prod_{i=1}^{n}(g^{u_i})^{h_i\tilde{h}_i} &= g^{u_j\tilde{h}_jh'_j} \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} \\
&= g^{\frac{a}{h_j}\tilde{h}_jh'_j} \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} \\
&= g^{ah'_j} \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} \\
&= g^{ah'_j} X^{-aR'} \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} X^{\frac{a}{h'_j}h'_jR'} \\
&= g^{ah'_j} X^{-aR'} \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} X^{u_jh'_jR'} \\
&= (g^r)^{\frac{ah'_j}{r} + \frac{\sum_{i=1}^{n} y'_i}{r}} \prod_{i\neq j} g^{-y'_i} X^{-aR'} g^{-y'_j} \\
&\quad \prod_{i\neq j}(g^{u_i})^{h'_i\tilde{h}_i} \prod_{i\neq j}(X^{u_i})^{-h'_iR'} \prod_{i=1}^{n}(X^{u_i})^{h'_iR'} \\
&= (R')^{z'} Y'_1 \cdot \cdots \cdot Y'_n \prod_{i=1}^{n}(X^{u_i})^{h'_iR'}.
\end{aligned}$$

4 Conclusions

In this paper, we have showed that the Qin *et al.* [18]'s certificate based ring signature scheme is not secure against the forgery attack. We consider pairing-free certificate based ring signature scheme along with provable security as an open problem and our future research work.

Acknowledgments

This work is partially supported by National Natural Science Foundation of China under Grant Nos. 61003230, 61370026, 61300191 and 61103206, the Fundamental Research Funds for the Central Universities under Grant No. ZYGX2013J073 and ZYGX2012J077, and the Applied Basic Research Program of Sichuan Province under Grant No. 2014JY0041.

References

[1] Man Ho Au, Joseph K. Liu, Willy Susilo, and Tsz Hon Yuen, "Certificate based (linkable) ring signature," in *3rd International Conference on Information Security Practice and Experience-ISPEC 2007*, pp. 79–92, Hong Kong, China, May 2007.

[2] Amit K Awasthi and Sunder Lal, "Id-based ring signature and proxy ring signature schemes from bilinear pairings," *International Journal of Network Security*, vol. 4, no. 2, pp. 187–192, 2007.

[3] Xuefei Cao, Weidong Kou, and Xiaoni Du, "A pairing-free identity-based authenticated key agreement protocol with minimal message exchanges," *In-*

formation Sciences, vol. 180, no. 15, pp. 2895–2903, 2010.

[4] David Chaum and Eugene van Hevst, "Group signature," in *Advances in Cryptology-EUROCRYPT 1991*, pp. 257–265, Brighton, UK, April 1991.

[5] Liqun Chen, Caroline Kudla, and Kenneth G. Paterson, "Concurrent signatures," in *Advances in Cryptology-EUROCRYPT 2004*, pp. 287–305, Interlaken, Switzerland, May 2004.

[6] Sherman S. M. Chow, Richard W. C. Lui, Lucas Chi Kwong Hui, and Siu-Ming Yiu, "Identity based ring signature: Why, how and what next," in *EuroPKI 2005*, pp. 144–161, Canterbury, UK, June 2005.

[7] Sherman S.M. Chow and WILLY Susilo, "Generic construction of (identity-based) perfect concurrent signatures," in *7th International Conference on Information and Communications Security-ICICS 2005*, pp. 194–206, Beijing, China, December 2005.

[8] Sherman S.M. Chow, Siu-Ming Yiu, and Lucas C.K. Hui, "Efficient identity based ring signature," in *3rd International Conference on Applied Cryptography and Network Security-ACNS 2005*, pp. 499–512, NY, USA, June 2005.

[9] David Galindo, Paz Morillo, and Carla Ràfols, "Breaking yum and lee generic constructions of certificate-less and certificate-based encryption schemes," in *3rd European PKI Workshop: Theory and Practice-EuroPKI 2006*, pp. 81–91, Turin, Italy, June 2006.

[10] Craig Gentry, "Certificate-based encryption and the certificate revocation problem," in *Advances in Cryptology-EUROCRYPT 2003*, pp. 272–293, Warsaw, Poland, May 2003.

[11] Daojing He, Jiajun Bu, Sencun Zhu, Sammy Chan, and Chun Chen, "Distributed access control with privacy support in wireless sensor networks," *IEEE Transactions on Wireless Communications*, vol. 10, no. 10, pp. 3472–3481, 2011.

[12] Qiong Huang, Guomin Yang, Duncan S. Wong, and Willy Susilo, "Efficient optimistic fair exchange secure in the multi-user setting and chosen-key model without random oracles," in *The Cryptographers' Track at the RSA Conference, CT-RSA 2008*, pp. 106–120, San Francisco, CA, USA, April 2008.

[13] Germán Sáez Javier Herranz, "New identity-based ring signature schemes," in *6th International Conference on Information and Communications Security-ICICS 2004*, pp. 27–39, Malaga, Spain, October 2004.

[14] Bo Gyeong Kang, Je Hong Park, and Sang Geun Hahn, "A certificate-based signature scheme," in *Topics in Cryptology-CT-RSA 2004*, pp. 99–111, CA, USA, February 2004.

[15] Jiguo Li, Xinyi Huang, Yi Mu, Willy Susilo, and Qianhong Wu, "Certificate-based signature: Security model and efficient construction," in *EuroPKI 2007*, pp. 110–125, Palma de Mallorca, Spain, June 2007.

[16] Xiaodong Lin, Rongxing Lu, Haojin Zhu, Pin-Han Ho, Xuemin (Sherman) Shen, and Zhenfu Cao, "Asrpake: An anonymous secure routing protocol with authenticated key exchange for wireless ad hoc networks," in *Proceedings of IEEE International Conference on Communications, ICC 2007*, pp. 1247–1253, Scotland, UK, June 2007.

[17] Joseph K. Liu, Joonsang Baek, Willy Susilo, and Jianying Zhou:, "Certificate-based signature schemes without pairings or random oracles," in *11th International Conference on Information Security-ISC 2008*, pp. 285–297, Taipei, Taiwan, September 2008.

[18] Zhiguang Qin, Hu Xiong, and Fagen Li, "A provably secure certicate based ring signature without pairing," *International Journal of Network Security*, vol. 16, no. 3, pp. 244–251, 2014.

[19] Ronald L. Rivest, Adi Shamir, and Yael Tauman, "How to leak a secret," in *Advances in Cryptology-AsiaCrypt 2001*, pp. 552–565, Gold Coast, Australia, December 2001.

[20] Adi Shamir, "Identity-based cryptosystems and signature schemes," in *Advances in Cryptology-Crypto 1984*, pp. 47–53, California, USA, August 1984.

[21] Hu Xiong, Konstantin Beznosov, Zhiguang Qin, and Matei Ripeanu, "Efficient and spontaneous privacy-preserving protocol for secure vehicular communication," in *Proceedings of IEEE International Conference on Communications, ICC 2010*, pp. 1–6, Cape Town, South Africa, May 2010.

[22] Hu Xiong, Zhong Chen, and Fagen Li, "Bidder-anonymous english auction protocol based on revocable ring signature," *Expert Systems with Applications*, vol. 39, no. 8, pp. 7062–7066, 2012.

[23] Hu Xiong, Zhiguang Qin, and Fagen Li, "An anonymous sealed-bid electronic auction based on ring signature," *International Journal of Network Security*, vol. 8, no. 3, pp. 235–242, 2009.

[24] Hu Xiong, Zhiguang Qin, and Fagen Li, "A certificateless proxy ring signature scheme with provable security," *International Journal of Network Security*, vol. 12, no. 2, pp. 92–106, 2011.

[25] Hu Xiong, Zhiguang Qin, and Fagen Li, "A taxonomy of ring signature schemes: Theory and applications," *IETE Journal Of Research*, vol. 59, no. 4, pp. 376–382, 2013.

[26] Dae Hyun Yum and Pil Joong Lee, "Identity-based cryptography in public key management," in *1st European PKI Workshop: Research and Applications-EuroPKI 2004*, pp. 71–84, Samos Island, Greece, June 2004.

[27] Jianhong Zhang, "On the security of a certificate-based signature scheme and its improvement with pairings," in *5th International Conference on Information Security Practice and Experience-ISPEC 2009*, pp. 47–58, Xi'an, China, April 2009.

Ji Geng is a professor in the School of Computer Science and Engineering, University of Electronic Science and Technology of China. He received his M.S. degree from

Southwest Jiaotong University in 1990. His research interests include: information security and system software.

Hu Xiong is an associate professor at University of Electronic Science and Technology of China (UESTC). He received his Ph.D degree from UESTC in 2009. His research interests include: cryptography and network security.

Fagen Li received his Ph.D. degree from Xidian University in 2007. He is now an associate professor in the School of Computer Science and Engineering, University of Electronic Science and Technology of China. His recent research interests include cryptography and network security.

Zhiguang Qin is the dean and professor in the School of Computer Science and Engineering, University of Electronic Science and Technology of China (UESTC). He received his PH.D. degree from UESTC in 1996. His research interests include: information security and computer network.

Privacy-preserving Communication for VANETs with Conditionally Anonymous Ring Signature

Shengke Zeng, Yuan Huang, and Xingwei Liu
(Corresponding Author: Xingwei Liu)

School of Computer and Software Engineering, Xihua University
999 Jin Zhou Road, Jin Niu District, Chengdu, Sichuan Province 610039, P. R. China
(Email: lxw@mail.xhu.edu.cm)

Abstract

In this paper, we introduce an efficient communication protocol for vehicular ad hoc networks (VANETs) based on conditionally anonymous ring signature scheme to address the issue on anonymous authentication and efficient tracking in case of a dispute. It offers low storage requirements and fast message authentication. In addition, the proposed protocol does not require Road-side Units to aid to authenticate or track. Indeed, the obvious advantage is that our construction does not depend on any fully trusted authority during the tracing phase.

Keywords: Conditional privacy, conditionally anonymous ring signature, ring signature, VANETs

1 Introduction

To reduce traffic accidents and improve driving experience, extensive efforts have been made by industry and academia. And so, a self-organized vehicular ad hoc networks (VANETs) emerged. VANETs mainly consist of wireless communication devices On-board Units (OBUs) and Road-side Units (RSUs). Through inter-vehicle communication and vehicle to OBU communication, VANETs can collect traffic and road information and deliver them to all the users after integration.

At present, one of the key issues in design and deployment of VANETs is anonymous authentication. On the one hand, we expect that a message is authenticated by a credible vehicle (sender) instead of malicious or bogus vehicle. On the other hand, the sender is reluctant to leak its identity or location information during the authentication. Clearly, the goals of privacy presentation and accountability seem conflicting. Furthermore, the conditional privacy protection should be satisfied where an involved vehicle should be revoked by Transportation Regulation Center (TRC) just in a traffic dispute [4, 8].

To tackle this conditional privacy during the communication in VANETs, there existing kinds of proposals such as pseudonyms-based approaches, group-oriented signature-based approaches and RSU-based approaches. In 2005, Raya et al. introduced a large number of anonymous keys based protocol (LAB) [9] which is a kind of pseudonyms-based approach. Although LAB protocol satisfies the conditional privacy requirement, it is inefficient in terms of storage, tracing and revocation since it requires 43800 certificates for each vehicle to meet the privacy. Some approaches have been proposed to reduce the large number of pseudonyms which are preloaded on each vehicle, such as [1]. In addition, the group-oriented signature-based approaches can avoid the inefficiency existed in pseudonyms-based approaches. For example, the GSB protocol [5] introduced by Lin et al. does not need to store large number of keys and anonymous certificates. However, it requires each remaining vehicle to calculate a new private key and group public key if the number of revoked vehicles is larger than some threshold. To verify the message, the time increases linearly as the number of revoked vehicles in the revocation list grows. Xiong et al. proposed an anonymous authentication protocol based on proxy re-signature scheme [12]. This protocol depends on the RSUs to aid to authenticate the safety messages. It enables lower computation and communication overheads compared to LAB protocol and GSB protocol. However, this kind of RSU-aided authentication is over-reliance on RSUs. As we know, RSUs are vulnerable to attackers in the real world. Furthermore, there are some other schemes, for example, PPSCP [7] used the shared keys instead of pseudonyms or anonymous certificates to authenticate vehicle safety messages. Zhang et al. [15] proposed an improved authentication scheme which needs to produce a pseudonym before the vehicle sending a message each time. The potential problems in [7] and [15] are the same as [11], which is proposed by Xiong et al. This scheme [11] introduced an efficient authentication for VANETs based on revocable ring signature [6] (denoted as RRSB). It is clear except that it relies on the absolutely honest TRC. In the realization of tracing OBU, the TRC cannot show the evidence of the validation process. Actually, the authority can slander any vehicle arbitrary, and the framed vehicle has no way to prove its innocence.

In this paper, we focus on the construction of a commu-

nication protocol based on conditionally anonymous ring signature [13, 14] to tackle the conditional privacy presentation and authentication for VENETs, called CRSB. Different from [11], our protocol does not fully depend on the authority in tracing. In other words, TRC in our scheme cannot frame any vehicles during the anonymous authentication. The remainder of this paper is organized as follows. Section 2 introduces the preliminaries. Section 3 presents the system model and design goals. Section 4 proposes the privacy-preserving authentication protocol for VANETs and the security analysis and performance evaluation are shown in Section 5. The last section concludes this paper.

2 Preliminaries

In this section, we briefly introduce the mathematical tool and the underlying signature used in our protocol.

2.1 Mathematical Tool

Bilinear maps over an elliptic curve will be our mathematic tool. Let G_1 be an additive group over an elliptic curve and G_2 be a multiplicative cyclic group. Both of them have a same prime order q. P is a generator of G_1. Let $\hat{e} : G_1 \times G_1 \rightarrow G_2$ be a computable bilinear map with the following properties:

1) *Bilinearity.* $\forall P, Q \in G_1$, $\forall a, b \in \mathbb{Z}_q$, $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ holds.

2) $Non - degeneracy.$ $\hat{e}(P, P) \neq 1$.

3) *Computability.* All the group operations and the bilinear map must be efficiently computable.

2.2 Underlying Signature Algorithm

Ring signature algorithm was first introduced by Rivest et al. in 2001 [10]. It enables the signer to sign a message anonymously. The signer in the ring signature algorithm can randomly choose members (with their public keys) to form a group without these members' consent. Through a valid ring signature, the receiver can be convinced that the message coming from this group without knowing the actual sender. Thus, the anonymity of the signer is satisfied. Different from the group signature algorithm [2], the ring signature scheme does not need any group manager to join in. There is no setup algorithm in the ring signature scheme. Therefore, the ring signature scheme has a more flexible frame. However, the anonymity of the signer in the ring signature is unconditional. Even all the private keys of members in the group are revealed, it cannot be determined who is the actual signer.

Recently, Zeng et al. [13, 14] have introduced a conditionally anonymous ring signature with additional two algorithms: confirmation algorithm and disavowal algorithm. Compared to the revocable ring signature [6], this scheme does not require the third party to trace the actual signer. If the dispute arises, the malicious signer can be revoked through the disavowal protocol. The conditional anonymity without the third party is good for the privacy-preserving communication for VANETs. We adopt their scheme as the underlying signature algorithm. On the one hand, the conditional privacy can be satisfied. On the other hand, it is more fair for the vehicles even though TRC is not absolutely honest.

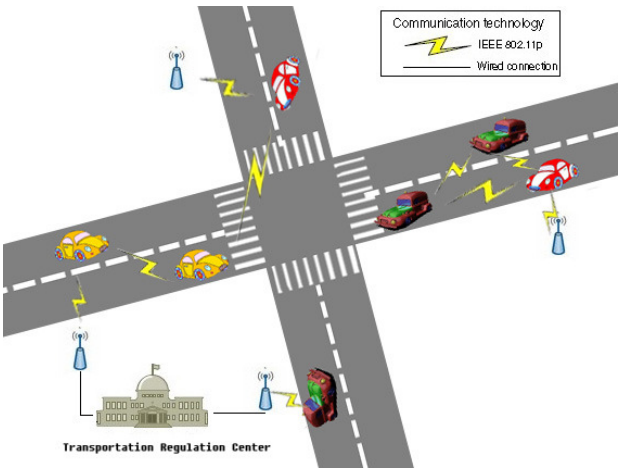


Figure 1: System model

3 System Model and Security Goals

In this section, we present the mainly entities in VANETs (Figure 1) in order to clear the later scheme. Further, we give the security requirements which should be satisfied during the secure and privacy-preserving communications in VANETs.

3.1 System Model

The common VANETs system with privacy protection mainly consists of three entities: the Transportation Regulation Center (TRC), the on-board units (OBUs) equipped on moving vehicles and the road-side units (RSUs). However, we do not employ RSUs in our system. Generally speaking, the moving vehicles in VANETS equipped with OBUs are registered with TRC which is in charge of revealing the real identity of the involved vehicle. Concretely,

- TRC. TRC in our scheme is an institution which is in charge of identity authentication, issuing and recycling certificate of each vehicle. Moreover, TRC can call out all of the vehicles in some ring to trace the target vehicle which involved in a traffic dispute. TRC has enough storage space and computational ability. However, unlike other related schemes, TRC

is not required to be fully trusted in our protocol. In other words, TRC must show a valid proof while tracing the real identity of malicious vehicle.

- OBU. After initialization with the TRC, vehicles can join in the VANETs. Each vehicle is preloaded with public system parameters, certificate issued by TRC and the public-private key pair. As the vehicle moves most of the time, so does the OBU moves constantly. Each OBU should broadcast its routine safety messages when they are on the road, such as position, current time, direction, speed, acceleration or deceleration, traffic conditions and traffic events. Thereout, the communication between two vehicles or vehicle to RSU can assist drivers to get a better awareness of their environment and take action earlier.

3.2 Security Goals

We focus on the authentication and privacy during communications in VANETs, the following aspects should be addressed.

Authentication. The messages delivered in VANETs should be authenticated. To meet the security (e.g. against impersonation attack), the accepted messages should be generated by legitimate vehicles. Therefore, all the messages must be authenticated by the receiver no matter how they are sent by RSUs or OBUs.

Anonymity. From the perspective of the vehicles, they are disinclined to leak their personal information and be tracked during the messages authentication. It seems that the anonymity and authentication are contradictory.

Traceability. The vehicles may take the advantage of anonymity to misbehave, e.g., an insider can release selfish or malicious messages since it is not afraid to be tracked. In other words, a considerate communication protocol in VANETs should meet the conditional privacy. If the dispute occurs, the malicious vehicle must be revoked. Therefore, the authority (i.e. TRC) should reveal the vehicle's actual identity if necessary. Since TRC is not fully trusted, TRC must show the valid proof when it reveals the malicious vehicle's identity.

4 Efficient and Secure Privacy-preserving Vehicular Communication Scheme

We present our authentication protocol for VANETs based on conditionally anonymous ring signature scheme in detail in this section. Each vehicle can be obtained a set of public keys from other vehicles messages during its moving. The vehicle also would update this set of public keys if old ones are changed. When a vehicle (sender) wants to authenticate a message m, it randomly chooses n valid public keys from the set to form a ring R. Then the sender generates a ring signature σ with respect to (m, R) according to the underlying ring signature scheme. If σ is a valid signature w.r.t. (m, R), then the receiver is convinced that message m is sent by one member in ring R without knowing which one. In this way, the actual identity of the sender is protected. On the other side, if the sender is involved, TRC must track the sender out. Therefore, the underlying ring signature scheme cannot be unconditionally anonymous for the signer.

Table 1: Notation and description

Notation	Description
TRC	Transportation Regulation Center
OBU	On-board Unit
CRL	Certificate Revocation List
V_i	The i-th vehicle
RID_i	The real identity of V_i
$Cert_i$	The certificate of V_i
x_i	The private key of V_i
$y_i = x_iP$	The public key of V_i
m	The authenticated message
H_0, H_1	Hash functions
$Sig(\cdot)$	Digital signature algorithm
$m\|\|n$	Concatenation of strings m and n

The proposed protocol includes four parts: system initialization and membership registration, OBU safety message generation, message verification and tracking algorithm. The notations used in the following scheme are listed in Table 1.

A. System Initialization and Membership Registration

Given the security parameter γ, TRC generates the parameters $(G_1, G_2, P, q, \hat{e})$, where G_1 is an additive group and G_2 is a multiplicative cyclic group, both of them have the same prime order q. P is the generator of G_1. $\hat{e}$ is a computable bilinear map such that $\hat{e} : G_1 \times G_1 \rightarrow G_2$. TRC also selects a secure digital signature algorithm $Sig(\cdot)$ and two cryptographic hash functions:

$$H_0 : \{0,1\}^* \rightarrow G_1 \text{ and } H_1 : \{0,1\}^* \rightarrow \mathbb{Z}_q.$$

After that, TRC randomly selects $x_{TRC} \leftarrow \mathbb{Z}_q$ as its private key and computes $y_{TRC} = x_{TRC}P$ as its public key. Finally, TRC outputs system parameters $(G_1,G_2,P,q,\hat{e},H_0,H_1,y_{TRC},Sig(\cdot))$.

To achieve more comprehensive security, each vehicle V_i with its real identity RID_i generates its key pair by itself and obtains its certificate from TRC as follows.

- V_i randomly chooses $x_i \leftarrow \mathbb{Z}_q$ as its private key, and computes $y_i = x_iP$ as its public key.

- V_i randomly selects an integer $t_i \leftarrow \mathbb{Z}_q$ to compute the verification information $a_i = H_1(t_iP||RID_i)$ and $b_i = t_i + x_ia_i$. Then V_i sends (y_i, RID_i, a_i, b_i) to TRC for registration.
- After received this message, TRC checks whether the following equation holds or not:

$$a_i \stackrel{?}{=} H_1((b_iP - a_iy_i)||RID_i).$$

If it meets, (y_i, RID_i) will be defined as valid public key and identity of V_i. After that, TRC stores (y_i, RID_i) and creates the certificate $Cert_i = Sig(y_i, RID_i; x_{TRC})$ for V_i with TRC's private key x_{TRC}. Finally, the tamper-proof device of each vehicle is preloaded with $(x_i, y_i, Cert_i, RID_i)$.

B. OBU Safety Message Generation

For each vehicle in VANETs, it should generate the signature on message m before sending it. In our scheme, we consider the common vehicles (excluding ambulance, police cars, military vehicles and so on) which need privacy protection. We take a common vehicle V_k for example. As mentioned before, when V_k moves on the road for some time, it has collected and stored many public keys of other vehicles. We suppose this set of public keys is $\mathcal{R} = \{y_1, y_2, \cdots, y_n, y_{n+1}, \cdots\}$. When V_k needs to send and authenticate a message m, it randomly chooses n public keys from set $\mathcal{R}$ to form a group (e.g. ring) R. The signature generation algorithm is listed as follows:

1) V_k randomly selects $r_0 \leftarrow \{0,1\}^\gamma$, computes $\mu_0 = H_0(0, r_0, m, R)$ and $\mu_1 = H_0(1, r_0, m, R)$.
2) V_k computes $\rho = \hat{e}(\mu_1, \mu_0)^{x_k}$. After that, V_k generates verification information Π_1 to prove $\rho = \hat{e}(\mu_0, \mu_1)^{x_k}$ is consistent with some public key in R as follows:
   - Select $d, r_1 \leftarrow \mathbb{Z}_q$, compute $M = \hat{e}(P,P)^d$, $N = \hat{e}(\mu_1, \mu_0)^d$, $R_1 = \rho^{r_1}$.
   - For $1 \leq i \leq n$ but $i \neq k$, randomly choose $U_i \leftarrow G_1$, compute $h_i = H_1(m, M, N, R_1, \rho, U_i)$.
   - Compute U_k, h_k, and e as follows:

$$\begin{aligned} U_k &= r_1y_k - \sum_{i \neq k}(U_i + h_iy_i - h_iy_k), \\ h_k &= H_1(m, M, N, R_1, \rho, U_k), \\ e &= d - (\sum_{i=1}^{n} h_i + r_1)x_k. \end{aligned}$$

The signature with respect to (m, R) is $\sigma = (\rho, r_0, \Pi_1)$ where $\Pi_1 = (M, N, R_1, \{U_i\}_{i=1}^n, e)$. Finally, V_k broadcasts (m, R, σ).

C. Message Verification

Upon received (m, R, σ), the receiver, say V_l, checks whether these public keys y_i in ring R are contained in CRL or not. If all these public keys y_i are not in CRL, then, V_l checks σ as follows:

1) For $1 \leqslant i \leqslant n$, V_l computes $h_i = H_1(m, M, N, R_1, \rho, U_i)$.
2) V_l verifies whether the following conditions are true.

$$\begin{aligned} M &\stackrel{?}{=} \hat{e}(P,P)^e \cdot \hat{e}(P, \sum_{i=1}^{n}(U_i + h_iy_i)) \\ N &\stackrel{?}{=} \rho^{\sum_{i=1}^{n} h_i} \cdot R_1 \cdot \hat{e}(\mu_1, \mu_0)^e. \end{aligned}$$

If they hold, V_l will be convinced that message m is authenticated by one member in the ring R without knowing which one.

D. Tracking Algorithm

When comes a reward or dispute, there should be some mechanisms to reveal the real identity of the message authenticator. Consider the two scenarios. If the sender will be received a reward for his signing on one message, he is willing to admit his identity for his generation σ. In this case, our *confirmation algorithm* is helpful for him. On the other hand, if his malicious signing involves dispute, TRC must trace this member to take the responsibility for his fault. In this case, the malicious sender will not admit his signing of course. Then we should take our *disavowal algorithm* to help TRC to track the sender out.

confirmation algorithm. V_k and TRC conduct the confirmation algorithm as follows to convince TRC that he is the signer of given signature σ w.r.t. (m, R).

1) V_k randomly selects $d' \leftarrow \mathbb{Z}_q$, and computes $M' = \hat{e}(P,P)^{d'}$, $N' = \hat{e}(\mu_1, \mu_0)^{d'}$, $h'_k = H_1(M', N', \rho)$, $e' = d' - h'_k \cdot x_k$.
2) V_k computes $\Pi_2 = (e', M', N')$, then sends Π_2 to TRC.

After received Π_2, TRC performs as follows.

1) TRC computes $h'_k = H_1(M', N', \rho)$;
2) TRC verifies Π_2 by checking the following equations:

$$\begin{aligned} M' &\stackrel{?}{=} \hat{e}(P,P)^{e'} \cdot \hat{e}(P, y_k)^{h'_k} \\ N' &\stackrel{?}{=} \rho^{h'_k} \cdot \hat{e}(\mu_1, \mu_0)^{e'}. \end{aligned}$$

If they hold, TRC is convinced that σ is generated by V_k.

Disavowal Algorithm. When V_k involves the dispute for his signing σ and V_k does not admit his generation. Then TRC must depend on our disavowal algorithm to trace V_k. Our strategy is that, TRC calls out all the members in ring R to execute the disavowal algorithm with him. If the member V_i is not the sender, he must pass verification of the disavowal algorithm. In this way, only V_k (who is the actual signer of σ) cannot pass the verification. Therefore, TRC tracks the malicious sender out. The detail of disavowal algorithm is as follows.

1) V_i computes $\rho_i = \hat{e}(\mu_1, \mu_0)^{x_i}$.

2) V_i generates Π_3 as confirmation algorithm to prove that ρ_i is consistent with his public key y_i, and sends (ρ_i, Π_3) to TRC.

3) TRC checks Π_3's validation according to the verification equations in confirmation algorithm and checks that $\rho_i \neq \rho$. If they hold, TRC accepts the disavowal of V_i.

Remark 1. *Our communication protocol (CRSB) does not require each vehicle to store a large number of keys and anonymous certificates like LAB protocol. Each vehicle in CRSB only needs to store its key pair and CRL. The storage overhead of CRSB is lower than pseudonyms-based protocols. CRSB protocol does not require any RSUs to aid to authenticate messages or trace the vehicle. CRSB is based on ring signature scheme, compared to GSB protocol, CRSB does not require each remaining vehicle to update any public parameters if the number of revoked vehicles is larger than some threshold. CRSB meets the conditional privacy for the confirmation protocol and disavowal protocol. Indeed, any verifiers can obtain the proof transcript if one conducts the confirmation protocol or disavowal protocol with members in the ring R. Thus, CRSB does not rely on the absolutely honest TRC during the tracing phase. While RRSB [11] is based on revocable ring signature scheme, the actual member must be revoked by authority. Therefore, RRSB is secure only on the assumption that TRC is fully honest.*

5 Security Analysis and Performance Evaluation

In this section, we give the security analysis and performance evaluation of our construction.

5.1 Security Analysis

We analyze the security of CRSB protocol in terms of message authentication, user privacy preservation and traceability of the target vehicle.

- *Message Authentication*: In our scheme, σ w.r.t. (m, R) can be generated only by a registered vehicle in the ring R. Under the unforgeability of the underlying ring signature scheme, it is infeasible for an attacker which do not belong to ring R to forge a valid ring signature σ. Therefore, as long as σ fulfills the equation in the message verification phase in section 4, we can confirm that the message m must be authenticated by one member from the ring R.

- *User Privacy Preservation*: This property holds under the anonymity of the underlying ring signature scheme. It is proven in [13, 14] that the anonymity of this underlying ring signature is satisfied if Decisional Bilinear Diffie-Hellman assumption holds. Therefore, the privacy of the vehicle (authenticator) is protected in our protocol.

- *Traceability*: CRSB protocol provides the *confirmation protocol* and the *disavowal protocol* to revoke the actual signer. Specially, the traceability and the non-frameability of the underlying ring signature guarantee that the actual signer must be traced if a generated ring signature is valid and an innocent member cannot be framed if he does not generate one signature, respectively. Therefore, TRC can reveal the real identity of the vehicle by checking the list (y_i, RID_i).

5.2 Performance Evaluation

We evaluate the performance for CRSB protocol in terms of storage requirements and computational overhead, and compare CRSB to other related privacy-preserving protocols in VANETs.

A. Storage Requirements

We focus on the comparison between the RRSB protocol [11] and our protocol (CRSB) since both two protocols are based on ring signature algorithm. According to the analysis in [11], the total storage overhead of each vehicle in RRSB protocol is $m+1$ if there are m OBUs which are revoked and each key occupies one storage unit. Likewise, each vehicle stores one keypair registered in TRC and m revoked public keys in the CRL. Thus, the total storage unit of CRSB is also $m + 1$.

For the storage overhead, ring (group) signature-based protocols are better than LAB [9] since each vehicle in LAB protocol needs to store its own anonymous key pairs (almost up to 10^4 key pairs for the security) and m revoked public keys in the CRL. In other words, $(m + 1) \cdot 10^4$ is the total storage overhead for LAB protocol. However, RSU-based protocols such as [12] is the best for the storage overhead. For example, each OBU in [12] only needs to store one key pair and a short-time key pair together with its anonymous certificate issued by RSU. The storage overhead in such RSU-based protocols is only 2 since OBU does not need to store the CRL. Although the Roadside Unit-aided case is the most efficient in the storage, it requires Road-side Units to join in the communication authentication. However, in our scheme, we do not require any RSUs to aid to authenticate or trace.

B. Computation Overheads

In CRSB protocol, the vehicle authentication phase requires 1 pairing computation, 4 exponentiations and n point multiplications, $n + 2$ hashing operations where n is the size of the ring (the number of vehicles involved in ring R). The vehicle verification phase requires 2 pairing computations, 3 exponentiations, n point multiplications and $n + 2$ hashing operations. Thus, the total computation overhead during communication for our construction requires 3 pairing computations, 7 exponentiations,

$2n$ point multiplications and $2n + 4$ hashing operations. While the RRSB protocol during the vehicle authentication phase requires 1 pairing computation, 2 exponentiations, $2n$ point multiplications and 2 hashing operations. Their vehicle verification phase also requires 1 pairing computation, 2 exponentiations, $2n$ point multiplications and 1 hashing operations. Then the total computation overhead during communication for RRSB protocol requires 2 pairing computation, 4 exponentiations, $4n$ point multiplications and 3 hashing operations.

Under the same security parameter, the time consuming for the pairing computation, exponentiation, point multiplication and the map to point hashing operation are 47.4ms, 3.13ms, 6.83ms and 3.00ms respectively with the subgroup of order prime 160-bit q in a super-singular elliptic curve $E(\mathbb{F}_p)$ with the embedding degree 2, where p is 512-bit prime [3]. This implementation of these primitives are executed on Pentium IV 2.26GHz with 256M RAM.

The total computation overhead comparison between CRSB protocol and RRSB protocol is listed in Table 2. We can find that the computation overheads of the two schemes are increasing with the growth of the number of vehicles n. In addition, with the increase of n, the computation of RRSB has a faster growth than CRSB.

Table 2: Comparison between CRSB and RRSB

	Descriptions	Execution Time
T_{CRSB}	The total execution time for CRSB protocol	$(176.11 + 19.66n)$ms
T_{RRSB}	The total execution time for RRSB protocol	$(116.32 + 27.32n)$ms

6 Conclusion

We introduce an efficient authentication protocol based on conditionally anonymous ring signature (CRSB) for privacy-preserving VANETs. Our protocol satisfies efficient authentication and conditional privacy preservation. Moreover, our protocol does not require any RSUs to participate in the authentication. Meanwhile, we also does not require any fully trusted authority during the tracing phase.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (NOs. 61402376, U1433130), the Key Technology Research and Development Program of Sichuan Province and Chengdu Municipality (NOs. 2013GZX0140, 12DXYB127JH).

References

[1] M. Burmester, E. Magkos, V. Chrissikopoulos, "Strengthening privacy protection in VANETs," in *IEEE International Conference on Networking and Communications, 2008 (WIMOB '08)*, pp. 508-513, 2008.

[2] D. Chaum and van E. Hevst, "Group signature," in *Eurocrypt'91*, pp. 257-265, Brighton, UK, 1991.

[3] S. Cui, P. Duan, C. W. Chan, "An efficient identity-based signature scheme and its applications," *International Journal of Network Security*, vol. 5, no. 1, pp. 89-98, 2007.

[4] C. T. Li, M. S. Hwang, Y. P. Chu, "A secure and efficient communication scheme with authenticated key establishment and privacy preserving for vehicular Ad hoc networks," *Computer Communications*, vol. 31, no. 12, pp. 2803-2814, 2008.

[5] X. Lin, X. Sun, P. H. Ho, "GSIS: A secure and privacy-preserving protocol for vehicular communications," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 6, pp. 3442-3456, 2007.

[6] D. Y. W. Liu, J. K. Liu, Y. Mu, "Revocable ring signature," *Journal of Computer Science Technology*, vol. 22, no. 6, pp. 785-794, 2007.

[7] M. Mikki, Y. M. Mansour, "Privacy preserving secure communication protocol for vehicular Ad hoc networks," in *2013 Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, pp. 189-195, 2013.

[8] H. H. Ou, M. S. Hwang, J. K. Jan, "The UMTS-AKA protocols for intelligent transportation systems," *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, pp. 1–12, 2009.

[9] M. Raya and J. P. Hubaux, "Securing vehicular ad hoc networks", *Journal of Computer Security*, Special Issue on Security of Ad Hoc and Sensor Networks, pp. 39-68, 2007.

[10] R. L. Rivest, A. Shamir and Y. Tauman, "How to leak a secret," in *Asiacrypt'01*, pp. 552-565, Gold Coast, Australia, 2001.

[11] H. Xiong, K. Beznosov, Z. Qin, "Efficient and spontaneous privacy-preserving protocol for secure vehicular communication," in *International Communications Conference (ICC'10)*, pp. 1-6, Cape Town, South Africa, 2010.

[12] H. Xiong, Z. Chen and F. Li, "Efficient privacy-preserving authentication protocol for vehicular communications with trustworthy," *Security and Communication Networks*, vol. 12, no. 5, pp. 1441-1451, 2012.

[13] S. Zeng, S. Jiang and Z. Qin, "A new conditionally anonymous ring signature," in *17th International Computing and Combinatorics Conference (COCOON'11)*, pp. 479-491, Dallas, USA, 2011.

[14] S. Zeng, S. Jiang and Z. Qin, "An efficient conditionally anonymous ring signature in the random oracle model," *Theoretical Computer Science*, vol. 461, pp. 106-114, 2012.

[15] J. Zhang, M. Xu, "On the security of a secure batch verification with group testing for VANET", *International Journal of Network Security*, vol. 16, no. 5, pp. 355-362, 2014.

Shengke Zeng is a Lecturer at the School of Mathematics and Computer Engineering, Xihua University. She received her Ph.D. degree from University of Electronic Science and Technology of China (UESTC) in 2013. Her research interests include: Cryptography and Network Security.

Yuan Huang is a Master Candidate at the School of Mathematics and Computer Engineering, Xihua University. Her research interest is Network Security.

Xingwei Liu is a Professor at the School of Mathematics and Computer Engineering, Xihua University since 2002 and is also the director of the Laboratory for Wireless and Mobile Networks. He received his Ph.D. degree from Sichuan University in 2001. He has been a visiting professor at the Key Laboratory of Information Coding and Transmission in the Southwest Jiaotong University, Chengdu, China. His current research includes: Wireless and Mobile Networks.

Unidirectional Proxy Re-Encryption for Access Structure Transformation in Attribute-based Encryption Schemes

Xingbing Fu

School of Computer Science and Engineering, University of Electronic Science and Technology of China
No. 2006, Xiyuan Avenue, West Hi-tech Zone, Chengdu 611731, P. R. China
(Email: uestcfuxb@126.com)

(Received March 10, 2014; revised and accepted June 16, 2014)

Abstract

In Ciphertext Policy Attribute Based Encryption (CP-ABE) scheme, a user's private key is associated with a set of attributes, and the sensitive data are encrypted under an access structure over attributes, only if the users whose attributes satisfy the access structure associated with the ciphertext can decrypt the ciphertext data. However, a limitation of the existing CP-ABE schemes is that it does not support transforming access structure provided that the encrypted data are not decrypted. In this work, we proposed Ciphertext Policy Attribute Based Proxy Re-Encryption (CP-ABPRE) scheme which allows to transform access structure associated with the original ciphertext without decrypting it through an honest and curious proxy such as the cloud storage server that re-encrypts the original ciphertext under another access structure such that only if the users whose attributes satisfy the new access structure can decrypt the re-encrypted ciphertext. Security of the proposed scheme is based on the generic bilinear group model. Performance evaluation shows the proposed scheme is efficient.

Keywords: Access structure transformation, attribute based encryption, bilinear maps, proxy re-encryption, unidirectionality

1 Introduction

Traditional public key encryption scheme is to protect the confidentiality of the sensitive data. Encryption is viewed as a mechanism through which one user can share the sensitive data with another user. The scheme is very suitable for the setting where the data owner specifically knows with whom he wants to share the data in advance. However, in many applications such as cloud storage systems, the data owners may want to share data under some access policy over the target users' attributes or credentials to achieve fine-grained access control. Recent years, the proposed Attribute Based Encryption (ABE) schemes can meet the requirements very well. ABE has the two fundamental forms: Key Policy Attribute Based Encryption (KP-ABE) schemes and Ciphertext Policy Attribute Based Encryption (CP-ABE) schemes. In CP-ABE schemes, ciphertexts are associated with access policies, whereas user keys are associated with attribute sets. For example, in cloud storage systems, after the data owner encrypts the data employing CP-ABE scheme, he uploads the encrypted data to the cloud storage server which is semi-honest, such that any data consumers can download the ciphertext data, only if the data consumers whose attributes satisfy the access structures can decrypt the encrypted data. Neither the cloud server nor the unauthorized data consumers including malicious adversaries can decrypt the encrypted data to obtain plaintext messages.

In contrast with the traditional access control schemes such as mandatory access control, discretionary access control, role-based access control et al., CP-ABE schemes have many advantages in providing data security in distributed environments, especially in cloud storage setting, in that they can specify and enforce complex access policies without online interaction with trusted or/and centralized servers. However, the existing CP-ABE schemes do not support the transformation of the access structure. The decrypt-and-encrypt method to implement such a mechanism is that the encryptor sends his private key to the proxy, renders it decrypt the original ciphertext by using his private key to recover the plaintext message, and then encrypts it under another access structure employing the CP-ABE scheme. The shortcoming of the method is that the proxy can learn his private key and access the sensitive plaintext data. To solve this problem, the data owner may carry out the re-encryption operation as follows: he downloads the ciphertext data into his local disks from the cloud server acting as a proxy, then decrypts them employing his private key, re-encrypts the decrypted plaintext data under another access structure employing

the CP-ABE scheme, and finally uploads his re-encrypted ciphertext data to the cloud server. The shortcomings of this method are that the data owner must be online in each re-encryption stage, and it incurs the great processing and communication overheads at the same time,which is inefficient.

To solve the foregoing problems, ciphertext policy attribute based proxy re-encryption scheme is presented. In the presented scheme, a delegator only needs to calculate the re-encryption key $RK_{\mathbb{A}1\rightarrow\mathbb{A}2}$ employed by a proxy to convert the original ciphertext computed under one access structure $\mathbb{A}1$ into the re-encrypted ciphertext computed under another access structure $\mathbb{A}2$ without decrypting the original ciphertext. Our scheme satisfies collusion resistance where if the two users combine their attributes, they cannot decrypt the ciphertext which they cannot decrypt individually. In the existing PRE scheme, communication model is one-to-one, whereas communication model of our scheme is one-to-many, i.e., a ciphertext is decrypted by many users whose attributes satisfy the access structure associated with the ciphertext. Our scheme is very suitable for dynamic setting such as cloud storage system in which the access structures are transformed frequently.

The remainder of our paper is organized as follows: in Section 2 we discuss related works. We introduce preliminaries in Section 3. We present scheme definition and security game in Section 4. We discuss the scheme construction in Section 5. Security proof is given in Section 6. The performance of our scheme is evaluated in Section 7. We conclude and specify the future work in Section 8.

2 Related Works

Sahai and Waters [16] proposed the first attribute based encryption scheme as a new means for access control of the encrypted data. One shortcoming of the scheme is that its initial construction is limited to handling formulas comprising one threshold gate, which makes it less expressive. Goyal et al. [11] greatly enhanced the expressiveness of attribute based encryption scheme where users' keys are associated with access policies, whereas ciphertexts are associated with attribute sets. The drawback of their schemes is that the encryptor does not exert any control over who can access the data which she encrypts, except for her choice of attribute set of the data to be encrypted. Bethencourt et al. [4] proposed the construction of ciphertext policy attribute based encryption scheme where private keys are associated with a set of attributes, and ciphertexts are associated with access policies over attributes. Decryption is enabled if and only if the user's attribute set satisfies the access policy associated with the ciphertext. With the advent of cloud computing, more and more sensitive data will be outsourced to cloud storage server to be stored in the encrypted form. In order to realize fine-grained access control over the encrypted data, attribute based encryption schemes are applied to cloud storage setting where there exist a large number of different types of users who are authorized to read different data. Lee et al. [14] surveyed on attribute-based encryption schemes of access control in cloud environments. However, these schemes are focused on the attribute revocation, not on access structure transformation.

Blaze et al. [5] presented the first bidirectional chosen plaintext attack (CPA) secure scheme where the proxy is prevented from seeing the plaintext information and private keys, and the re-encryption algorithm is bidirectional, which may be undesirable in scenarios where trust relationships are asymmetric and leaving the construction of a unidirectional scheme as an open problem. Ateniese et al. [1, 2] proposed a first unidirectional CPA-secure scheme based on bilinear maps whose re-encryption algorithm is single-hop. Their schemes achieved the master key security in that the proxy and the delegatee cannot collude to reveal the delegator's private key. Both schemes whose re-encryption algorithms are deterministic are only CPA-secure ones, which are insufficient to guarantee security in general protocol settings. Canetti et al. [8] proposed the proxy re-encryption scheme with chosen ciphertext secure, where ciphertexts remain indistinguishable even though the adversary can access the re-encryption oracle and the decryption oracle. The drawback of their scheme is that their construction is bidirectional. Dodis et al. [13] presented the unidirectional proxy encryption where the private key generator delegates decryption rights for all identities in the system. However, their scheme has the serious security vulnerabilities: collusion between the proxy and the delegatee incurs a system-wide compromise, rendering the colluders reconstruct the master secret of IBE. Boneh et al. [7] proposed the Identity-Based Proxy Re-Encryption scheme where the private key generator carries out all delegations, such that users cannot perform non-interactive delegations, and every delegation involves a costly online request to the PKG. Green et al. [12] proposed a unidirectional identity-based proxy re-encryption with chosen ciphertext attack secure. Their security is based on the random oracle model. The recipient of a re-encrypted ciphertext needs to know who the original receiver is, such that he can decrypt the re-encrypted ciphertext. These papers are based on the traditional public key encryption schemes whose communication models are one-to-one.

Yu et al. [18] proposed attribute based data sharing employing proxy re-encryption techniques for fine-grained attribute revocation. Liang et al. [15] presented a ciphertext policy attribute based proxy re-encryption scheme. Chung et al. [10] surveyed two various access policy attribute-based proxy re-encryption schemes and analyzed these schemes.These schemes are based on the CN CP-ABE scheme [9], so that they have the same drawbacks as it: they only supports AND Boolean operator as access policies, the number of system attributes is fixed in setup and the ciphertext size and encryption and decryption time increase linearly in the total number of attributes in the system, which makes them less expressive.

3 Preliminaries

3.1 Bilinear Map

Let $\mathbb{G}_0$ and $\mathbb{G}_1$ be two cyclic groups of prime order p, and g, h are a generator of $\mathbb{G}_0$, respectively. e is a bilinear map $e : \mathbb{G}_0 \times \mathbb{G}_0 \to \mathbb{G}_1$, which has the following properties:

Bilinearity. For any $a, b \in \mathbb{Z}_p$, $e(g^a, h^b) = e(g, h)^{ab}$.

Nondegenerate. $e(g, g) \neq 1_{\mathbb{G}_1}$, $e(g, g)$ is a generator of $\mathbb{G}_1$.

If the group operation on $\mathbb{G}_0$ and the bilinear map $e : \mathbb{G}_0 \times \mathbb{G}_0 \to \mathbb{G}_1$ are efficiently computable, then $\mathbb{G}_0$ is a bilinear group. Our scheme employs the symmetric bilinear map which has the following properties: $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$.

3.2 Benaloh and Leichter Secret Sharing Scheme

Benaloh and Leichter secret sharing scheme [3] shares the secret $s \in \mathbb{Z}_p^*$ as follows: convert an access structure $\mathbb{A}$ into an access policy tree $\mathbb{T}$, and assign the root node of $\mathbb{T}$ the value s. For every other internal node, the following are recursively performed: if the operator is $\wedge$, assign every child node a random $s_j \in \mathbb{Z}_p^* (j = 1, 2, \cdots, n-1)$ except the last one, and assign the last child one

$$s_n = (s - \sum_{j=1}^{n-1} s_j) \bmod p,$$

if the operator is $\vee$, assign every child node the value s.

4 Definition

4.1 Our Scheme Definition

Definition 1. *Ciphertext Policy Attribute Based Proxy Re-Encryption (CP-ABPRE) scheme comprises the six algorithms as follows:*

Setup$(1^k) \to (MS, \mathbb{PP})$: *The Setup algorithm is run by the trusted authority. It takes a security parameter κ as input. It outputs a master secret MS employed to generate the users' private keys and the public parameters $\mathbb{PP}$ defining system attribute sets $\mathbb{S}$ which are employed by all parties in the scheme.*

Encrypt $(\mathbb{PP}, \mathbb{A}1, m) \to CT_{\mathbb{A}1}$: *The Encryption algorithm is run by the sender. It takes as inputs the public parameters $\mathbb{PP}$, the plaintext message m and the access structure $\mathbb{A}1$ over a set of attributes specifying which users are able to decrypt to recover the plaintext message. It outputs the original ciphertext $CT_{\mathbb{A}1}$ associated with access structure $\mathbb{A}1$.*

PriKeyGen$(MS, S) \to PriKey_S$: *The Private Key Generation algorithm is run by the trusted authority. It takes as inputs the master secret MS, and the attribute set of user $S \subseteq \mathbb{S}$. It outputs the private key of user $PriKey_S$ associated with the attribute set of user S.*

ReKeyGen$(\mathbb{PP}, \mathbb{A}1, \mathbb{A}2, PriKey_S) \to RK_{\mathbb{A}1 \to \mathbb{A}2}$: *The Re-Encryption Key Generation algorithm is run by the delegator. It takes as inputs the public parameters $\mathbb{PP}$, the access structures $\mathbb{A}1$ and $\mathbb{A}2$, and the private key $PriKey_S$. It outputs a unidirectional re-encryption key $RK_{\mathbb{A}1 \to \mathbb{A}2}$ which is employed by the proxy to re-encrypt the original cihpertext $CT_{\mathbb{A}1}$ if the attribute set associated with $PriKey_S$ satisfies access structure $\mathbb{A}1$, else it returns NULL.*

ReEncrypt$(\mathbb{PP}, \mathrm{CT}_{\mathbb{A}1}, RK_{\mathbb{A}1 \to \mathbb{A}2}) \to CT_{\mathbb{A}2}$: *The Re-Encryption algorithm is run by the proxy. It takes as inputs the public parameters $\mathbb{PP}$, the ciphertext $CT_{\mathbb{A}1}$ and the re-encryption key $RK_{\mathbb{A}1 \to \mathbb{A}2}$. It outputs the re-encrypted ciphertext $CT_{\mathbb{A}2}$ associated with the access structure $\mathbb{A}2$.*

Decrypt$(CT_{\mathbb{A}k}, PriKey_S) \to m (k = 1, 2)$: The Decryption algorithm is run by the decryptor who is either a delegator or a delegatee. It takes as inputs the $CT_{\mathbb{A}k}$ and the private key $PriKey_S$. It outputs the plaintext message m if attribute set S satisfies the access structures $\mathbb{A}_k$ $(k = 1, 2)$, else it returns NULL.

Correctness: A CPAB-PRE scheme is correct when for all security parameters κ, all messages m, all sets of attributes S, access structures with $\mathbb{A}_k$ $(k = 1, 2)$ with $S \in \mathbb{A}_k$, all master secrets MS and public parameters $\mathbb{PP}$ output by Setup algorithm, all private keys $PriKey_S$ output by PriKeyGen algorithm, all original ciphertexts $CT_{\mathbb{A}1}$ output by Encryption algorithm, all re-encryption keys $RK_{\mathbb{A}1 \to \mathbb{A}2}$ output by ReKeyGen algorithm, all re-encrypted ciphertexts $CT_{\mathbb{A}2}$ output by Re-Encryption algorithm, if a set of attributes S satisfies access structure either $\mathbb{A}1$ or $\mathbb{A}2$,the following propositions hold: Decrypt$(CT_{\mathbb{A}1}, PriKey_s) = m$, Decrypt$(PriKey_s, \mathrm{ReEncrypt}(\mathbb{PP}, CT_{\mathbb{A}1}, RK_{\mathbb{A}1 \to \mathbb{A}2})) = m$.

4.2 Security Model for Ciphertext Policy Attribute Based Proxy Re-Encryption (CPAB-PRE) Scheme

We describe a security model for CPAB-PRE scheme using a security game between a challenger and an adversary as follows:

Setup. The challenger runs the Setup algorithm which generates $(MS, \mathbb{PP})$ and gives the adversary $\mathbb{PP}$.

Phase 1. The adversary issues a polynomial number of key queries: Private key generation oracle $\mathbb{O}_{Prikey}(S)$: on input any set of attributes S, the challenger runs PriKeyGen$(MS, S) \to PriKey_s$, and returns $PriKey_s$ to the adversary.

Re-encryption key generation oracle $\mathbb{O}_{rk}(\mathbb{A}1, \mathbb{A}2)$: on input an access structure $\mathbb{A}1$ and a new access structure $\mathbb{A}2$, the challenger returns $RK_{\mathbb{A}1\to\mathbb{A}2} \leftarrow \text{ReKeyGen}(\mathbb{PP}, \mathbb{A}1, \mathbb{A}2, PriKey_S)$ to the adversary, where $PriKey_S \leftarrow \text{PriKeyGen}(MS, S)$.

Challenge. The adversary submits two plaintext messages m_0, m_1 of equal length and the challenge access structure $\mathbb{A}^*$ to the challenger, with the restriction that the adversary should not select a challenge access structure $\mathbb{A}^*$ if it has performed the queries in Phase 1 as follows: $\text{PriKeyGen}(S)$ queries such that the set of attributes S satisfies the challenge access structure $\mathbb{A}^*$ or any derivative challenge access structures. $\text{ReKeyGen}(\mathbb{PP}, \mathbb{A}1, \mathbb{A}2, PriKey_S)$ queries if the adversary has beforehand issued $PriKey_S$ queries such that a set of attributes S satisfies $\mathbb{A}2$ and $\mathbb{A}1$ is a derivative challenge access structure. The challenger flips a fairy binary coin $\beta \in \{0, 1\}$, and encrypts m_β under $\mathbb{A}^*$ as the challenge ciphertext $CT^* = \text{Encrypt}(\mathbb{PP}, \mathbb{A}^*, m_\beta)$ which is given to the adversary.

Phase 2. Phase 1 is repeated with the same restriction as the challenge phase.

Guess. The adversary outputs a guess $\beta' \in \{0, 1\}$ of β, if $\beta' = \beta$, the adversary wins.

Definition 2. *A CPAB-PRE scheme is secure against chosen plaintext attacks (CPA) if no probabilistic polynomial time adversaries have non-negligible advantage in the aforementioned game, where the advantage is defined as*

$$|Pr[\beta' = \beta] - \frac{1}{2}|.$$

5 Scheme Construction

5.1 Our Scheme Construction

This construction comprises the algorithms as follows:

Setup$(1^\kappa) \to (\mathbf{MS}, \mathbb{PP})$**:** The setup algorithm calls the group generator algorithm $\mathbb{G}(1^\kappa)$ and obtains the descriptions of the two groups and the bilinear map $D = (p, \mathbb{G}_0, \mathbb{G}_1, g, e)$, in which p is the prime order of the cyclic groups $\mathbb{G}_0$ and $\mathbb{G}_1$, g is a generator of $\mathbb{G}_0$ and e is a bilinear map. The trusted authority generates the universe of system attributes $\mathbb{S} = \{att_1, att_2, \cdots, att_n\}$, where n is a positive integer. It selects the random exponents $t_1, t_2, \cdots, t_n, \mu \in \mathbb{Z}_p^*$. For each attribute $att_i \in S(1 \le i \le n)$, it selects a corresponding $t_i \in \mathbb{Z}_p^*$, and sets $T_i = g^{t_i}(1 \le i \le n)$. It employs a cryptographic hash function $H : \mathbb{G}_1 \to \mathbb{G}_0$ which hashes the elements of $\mathbb{G}_1$ into the elements of $\mathbb{G}_0$. The public parameters are published as: $\mathbb{PP} = (D, e(g, g)^\mu, \{T_i\}_{1\le i\le n}, H)$, where $e(g, g)^\mu$ can be pre-computed. The master secret is $MS = (\mu, \{t_i\}_{1\le i\le n})$.

Encrypt$(\mathbb{PP}, \mathbb{A}1, m) \to CT_{\mathbb{A}1}$**:** The encryption algorithm encrypts a message $m \in \mathbb{G}_1$ under the access structure $\mathbb{A}1$. It selects a random value $s \in \mathbb{Z}_p^*$, and then assigns attributes in the access structure $\mathbb{A}1$ values s_j $(1 \le j \le n)$, where values s_j $(1 \le j \le n)$ are shares of secret s which are generated based on the aforementioned Benaloh and Leichter secret sharing scheme. The resulting ciphertext is constructed and calculated as follows:

$$\begin{aligned} CT_{\mathbb{A}1} &= (\mathbb{A}1, E_b = g^s, E_2 = m \cdot e(g, g)^{\mu s}, \\ &\quad \{E_{3,i,j} = g^{t_i s_j}\}_{att_{i,j}\in\mathbb{A}1}). \end{aligned}$$

PriKeyGen$(MS, S) \to PriKey_S$**:** The private key generation algorithm takes in the master secret MS and the attribute set of the user $S \subseteq \mathbb{S}$. For every user, it selects a random $r \in \mathbb{Z}_p^*$ employed to prevent collusion attacks through which the different users can pool their attributes to decrypt the ciphertext that they cannot decrypt individually and calculates the private key $PriKey_S$ as follows: $PriKey_S = (K_b = g^{\mu+r}, K_{2,i} = \{g^{rt_i^{-1}}\}_{att_i\in S})$.

ReKeyGen$(\mathbb{PP}, \mathbb{A}1, \mathbb{A}2, PriKey_s) \to RK_{\mathbb{A}1\to\mathbb{A}2}$**:** The Re-Encryption Key Generation algorithm produces a unidirectional re-encryption key $RK_{\mathbb{A}1\to\mathbb{A}2}$ employed by the proxy to convert the original ciphertext $CT_{\mathbb{A}1}$ computed under the access structure $\mathbb{A}1$ into the re-encrypted ciphertext $CT_{\mathbb{A}2}$ computed under the access structure $\mathbb{A}2$.Let $S' \subseteq S$ be the minimal set of attributes satisfying the access structure $\mathbb{A}1$. It selects random $\omega, \lambda \in \mathbb{Z}_p^*$, and calculates the re-encryption key $RK_{\mathbb{A}1\to\mathbb{A}2}$ as follows:

$$RK_{\mathbb{A}1\to\mathbb{A}2} = (K_b^*, K_{2,i}^*, K_3^*),$$

where

$$\begin{aligned} K_b^* &= K_b \cdot g^{-\omega} = g^{\mu+r-\omega}, \\ K_{2,i}^* &= \{K_{2,i}\}_{att_i\in S'}, \\ K_3^* &= (K_{3,1}^*, K_{3,2}^*, K_{3,i,j}^*) = \text{Encrypt}(\mathbb{PP}, \mathbb{A}2, g^\omega). \end{aligned}$$

$Encrypt(\mathbb{PP}, \mathbb{A}2, g^\omega)$ is performed as follows in the similar way as $\text{Encrypt}(\mathbb{PP}, \mathbb{A}1, m)$ in the Encryption phase:

$$\begin{aligned} K_{3,1}^* &= g^\lambda, \\ K_{3,2}^* &= g^\omega \cdot H(e(g, g)^{\mu\lambda}), \\ K_{3,i,j}^* &= \{g^{t_i\lambda_j}\}_{att_{i,j}\in\mathbb{A}2}. \end{aligned}$$

Likewise, where $\lambda_j(1 \le j \le n)$ values are shares of secret λ which are generated based on the aforementioned BL secret sharing scheme.

ReEncrypt$(\mathbb{PP}, \boldsymbol{CT}_{\mathbb{A}1}, RK_{\mathbb{A}1\to\mathbb{A}2}) \to CT_{\mathbb{A}2}$**:** The re-encryption algorithm calculates the components as follows:

Step 1. For each attribute $att_i \in S'$, it calculates

$$\begin{aligned} B_1 &= \prod_{att_i \in S'} e(E_{3,i,j}, K^*_{2,i}) \\ &= \prod_{att_i \in S'} e(g^{t_i s_j}, g^{r t_i^{-1}}) \\ &= \prod_{att_i \in S'} e(g^{s_j}, g^r) = e(g,g)^{rs}. \end{aligned}$$

Step 2. It calculates

$$\begin{aligned} B_2 &= e(E_b, K^*_b)/B_1 = e(g^s, g^{\mu+r-\omega})/e(g,g)^{rs} \\ &= e(g^s, g^{\mu+r}) \cdot e(g^s, g^{-\omega})/e(g,g)^{rs} \\ &= e(g^s, g^{\mu-\omega}). \end{aligned}$$

Step 3. It calculates

$$\begin{aligned} E^*_2 &= \frac{E_2}{B_2} \\ &= \frac{m \cdot e(g,g)^{\mu s}}{e(g^s, g^{\mu-\omega})} \\ &= \frac{m}{e(g^s, g^{-\omega})} \\ &= m \cdot e(g^s, g^{\omega}). \end{aligned}$$

Step 4. It sets

$$\begin{aligned} E^*_b &= E_b = g^s, \\ E^*_3 &= K^*_3 = (K^*_{3,1}, K^*_{3,2}, K^*_{3,i,j}). \end{aligned}$$

The resulting re-encrypted ciphertext comprises the following components:

$$CT_{\mathbb{A}2} = (\mathbb{A}_2, E^*_b, E^*_2, E^*_3).$$

Decrypt$(CT_{\mathbb{A}k}, PriKey_S) \rightarrow m (k = 1, 2)$**:** The decryption algorithm takes in the ciphertext $CT_{\mathbb{A}k}$ and the private key $PriKey_S$. If the set of attributes S does not satisfy the access structure $\mathbb{A}_k (k = 1, 2)$, the algorithm returns NULL. If the access structure $\mathbb{A}_k (k = 1, 2)$ is satisfied by S and $CT_{\mathbb{A}k}$ is a well-formed ciphertext, then the decryption algorithm performs the steps as follows:

Step 1. It selects the minimal set of attributes $S' \subseteq S$ satisfying the access structure $\mathbb{A}1$, and calculates

$$\begin{aligned} N_1 &= \prod_{att_i \in S'} e(E_{3,i,j}, K_{2,i}) \\ &= \prod_{att_i \in S'} e(g^{t_i s_j}, g^{r t_i^{-1}}) \\ &= \prod_{att_i \in S'} e(g^r, g^{s_j}) = e(g,g)^{rs}. \end{aligned}$$

Step 2. It calculates

$$\begin{aligned} N_2 &= e(E_b, K_b)/N_1 \\ &= e(g^s, g^{\mu+r})/e(g,g)^{rs} \\ &= e(g,g)^{\mu s}. \end{aligned}$$

Step 3. The message m is recovered via calculating

$$\begin{aligned} \frac{E_2}{N_2} &= \frac{m \cdot e(g,g)^{\mu s}}{e(g,g)^{\mu s}} \\ &= m. \end{aligned}$$

If S satisfies the access structure $\mathbb{A}2$, $S' \subseteq S$ be the minimal set of attributes satisfying the access structure $\mathbb{A}2$ and $CT_{\mathbb{A}2}$ is a re-encrypted ciphertext, then the decryption algorithm performs the following steps:

Step 1. For each attribute $att_i \in S'$, it calculates:

$$\begin{aligned} V_1 &= \prod_{att_i \in S'} e(K^*_{3,i,j}, K_{2,i}) \\ &= \prod_{att_i \in S'} e(g^{t_i \lambda_j}, g^{r t_i^{-1}}) \\ &= \prod_{att_i \in S'} e(g^{\lambda_j}, g^r) \\ &= e(g,g)^{r\lambda}. \end{aligned}$$

Step 2. It calculates

$$\begin{aligned} V_2 &= e(K_b, K^*_{3,1})/V_1 \\ &= e(g^{\mu+r}, g^{\lambda})/e(g,g)^{r\lambda} \\ &= e(g,g)^{\mu\lambda}. \end{aligned}$$

Step 3. It calculates

$$\begin{aligned} V_3 &= \frac{K^*_{3,2}}{H(V_2)} \\ &= \frac{g^{\omega} \cdot H(e(g,g)^{\mu\lambda})}{H(V_2)} \\ &= \frac{g^{\omega} \cdot H(e(g,g)^{\mu\lambda})}{H(e(g,g)^{\mu\lambda})} \\ &= g^{\omega}. \end{aligned}$$

Step 4. The message is recovered as follows:

$$\begin{aligned} \frac{E^*_2}{e(E^*_b, V_3)} &= \frac{m \cdot e(g^s, g^{\omega})}{e(g^s, g^{\omega})} \\ &= m. \end{aligned}$$

6 Security Proof

Proof of security is provided in the generic bilinear group model [6, 17] where group elements are encoded as unique random strings. We consider two random encodings φ_0,φ_1 of the additive group $\mathbb{F}_p$ which are injective maps $\varphi_0, \varphi_1 : \mathbb{F}_p \rightarrow \{0,1\}^l$, in which $l > 3\log_2 p$. Let $\mathbb{G}_i = \{\varphi_i(x) : x \in \mathbb{F}_p\}, i = 0, 1$. We are given oracles to calculate the induced group action on $\mathbb{G}_0$ and $\mathbb{G}_1$, and an oracle to calculate a bilinear map

$$e : \mathbb{G}_0 \times \mathbb{G}_0 \rightarrow \mathbb{G}_1.$$

Theorem 1. *Let $\varphi_0, \varphi_1, \mathbb{G}_0, \mathbb{G}_1$ be defined as aforementioned. For any adversary, let q be a bound on the total number of group elements it receives from queries that it makes to the oracles for groups $\mathbb{G}_0$ and $\mathbb{G}_1$, and the bilinear map e, and from its interaction with the CPAB-PRE security game. Then the advantage of the adversary in the CPAB-PRE security game is $\mathbb{O}(q^2/p)$.*

Proof. In the CPAB-PRE security game, the challenge ciphertext has a component that is either $m_0 e(g,g)^{\mu s}$ or $m_1 e(g,g)^{\mu s}$. Instead, we can consider a modified game where ciphertext component E_2 is either $e(g,g)^{\mu s}$ or $e(g,g)^{\sigma}$, in which σ is selected uniformly at random from $\mathbb{F}_p$ and the adversary must decide which the case is. Any adversary who has advantage ε in the CPAB-PRE security game can be converted into another adversary who has advantage at least $\varepsilon/2$ in the modified CPAB-PRE security game.

We will write g^{α} to denote $\varphi_0(\alpha)$ and $e(g,g)^{\eta}$ to denote $\varphi_1(\eta)$, where $\alpha, \eta \in \mathbb{Z}_p$. Each random encoding is associated with a rational function in the variables: $\{\mu, \sigma, s, \{t_i\}_{1\le i\le n}, \lambda, r, \omega\}$, where each variable is a random element selected in the scheme.

In Setup phase, let $g = \varphi_0(1)$, $\{g^{t_i} = \varphi_0(t_i)\}_{1\le i\le n}$,$e(g,g)^{\mu} = \varphi_1(\mu)$. The public parameters are sent to the adversary.

In Phase 1 and Phase 2 of the security game, for Private key generation oracle $\mathbb{O}_{PriKey}(S)$, let

$$\begin{aligned} K_b &= g^{\mu+r} = \varphi_0(\mu+r), \\ K_{2,i} &= \{g^{rt_i^{-1}} = \varphi_0(rt_i^{-1})\}_{att_i\in S}, \end{aligned}$$

for Re-encryption key generation oracle $\mathbb{O}_{rk}(\mathbb{A}1, \mathbb{A}2)$, let

$$\begin{aligned} K_b^* &= K_b \cdot g^{-\omega} = g^{\mu+r-\omega} \\ &= \varphi_0(\mu+r-\omega), \\ K_{2,i}^* &= \{K_{2,i}\}_{att_i\in S'}, \\ (K_{3,1}^*, K_{3,2}^*, K_{3,i,j}^*) &= \text{Encrypt}(\mathbb{PP}, \mathbb{A}2, g^{\omega}) \\ &= (\varphi_0(\lambda), \varphi_0(h), \{\varphi_0(t_i\lambda_j)\}_{att_{i,j}\in\mathbb{A}2}). \end{aligned}$$

These values are given to the adversary.

In the Challenge phase, for the Encryption oracle, when the adversary submits two challenge plaintext messages $m_0, m_1 \in \mathbb{G}_1$ and the challenge access structure $\mathbb{A}^*$, let

$$\begin{aligned} E_b &= g^s = \varphi_0(s), E_2 = e(g,g)^{\sigma} = \varphi_1(\sigma), \\ \{E_{3,i,j} &= g^{t_i t_j}\}_{att_{i,j}\in\mathbb{A}^*} = \{\varphi_0(t_i s_j)\}_{att_{i,j}\in\mathbb{A}^*}. \end{aligned}$$

These values are passed on to the adversary.

We will show the adversary cannot distinguish with non-negligible advantage the simulation of the modified game in which the challenge ciphertext is $E_2 = e(g,g)^{\sigma}$, from the simulation of the real game in which the challenge ciphertext is $E_2 = e(g,g)^{\mu s}$. Firstly, the adversary's view is given if the challenge ciphertext is $\varphi_1(\sigma)$, and the adversary's view can change if an unexpected collision occurs due to the random choices of these variables $\{\mu, \sigma, s, \{t_i\}_{1\le i\le n}, \lambda, r, \omega\}$. For any two distinct queries, the probability that any such collision happens is at most $\mathbb{O}(q^2/p)$. Secondly, what the adversary's view would have been if we had set $\varphi_1(\mu s)$. The adversary cannot obtain a query polynomial of the form μs, so such a collision cannot occur. In Table 1, we list possible queries into $\mathbb{G}_1$ based on the bilinear map and the group elements passed on to the adversary in the simulation.

Table 1: Possible query types from the adversary

$(\mu+r)t_is_j$	$(\mu+r)s$	rs_j	$\mu s+rs-rs_j$
rst_i^{-1}	$(\mu+r)\lambda$	$(\mu+r)h$	$(\mu+r)t_i\lambda_j$
$r\lambda t_i^{-1}$	$rt_i^{-1}h$	$r\lambda_j$	$(\mu+r-\omega)t_is_j$
$(\mu+r-\omega)s$	λs	$\lambda_j t_i^2 s_j$	$\lambda_j t_i s$
hs	ht_is_j	$\lambda t_i s_j$	

As seen from Table 1, the adversary can pair t_is_j with rt_i^{-1}, and $\mu+r$ with s, and then make the latter subtract the former to obtain $(\mu+r)s - \sum_{att_{i,j}\in S} rs_j$. In order to obtain μs, the adversary must make polynomial requests to cancel rs. The adversary has to pair t_is_j with rt_i^{-1} to get rs. As you can see from Table 1, the adversary has to construct a query polynomial of the form: $\mu s + rs - \sum_{att_j\in s} rs_j$. Whereas the adversary cannot construct a query polynomial of the form μs if he does not possess a private key associated with the set of attributes satisfying the access structure. There has to be one rs_j missing, in that even if the adversary has one ciphertext component $g^{t_is_j}$, he has not a private key component $g^{rt_i^{-1}}$ to pair. Therefore he is not able to reconstruct rs, as a result he cannot cancel the second term and the third term to get μs. From the foregoing analysis, we can draw a conclusion that the adversary cannot make a polynomial query of the form μs. □

7 Performance Analysis

7.1 Properties Comparison

As seen from Table 2, the distinguished property of our scheme is that the communication model of our scheme is one-to-many, i.e., a ciphertext is decrypted by many users whose attributes satisfy the access structure associated with the ciphertext, whereas the traditional proxy re-encryption schemes based on the traditional public key encryption schemes or identity based encryption schemes are one-to-one, i.e., a ciphertext is only decrypted by a private key.

7.2 Performance Evaluation

As illustrated in Table 3, where $x\mathbb{G}_0$, $y\mathbb{G}_1$, zC_e,kH and $||$ denote x exponentiations in $\mathbb{G}_0$, y exponentiations in $\mathbb{G}_1$, z times bilinear maps, k times hash, and the cardinality of the set, respectively, our scheme supports

Table 2: Property comparison of PRE schemes

References	Unidirectional	Hops	CCA security	Collusion Resistance	Non-interactive	Non-transitive	Key Optimal	Communication Model
BBS Scheme [5]	No	Multi-Hop	No	No	No	No	Yes	One-to-one
AFGH Scheme [1, 2]	Yes	Single-Hop	No	Yes	Yes	Yes	Yes	One-to-one
CH Scheme [8]	No	Multi-Hop	Yes	No	Yes	Yes	Yes	One-to-one
GA Scheme [12]	Yes	Multi-Hop	Yes	Yes	Yes	Yes	Yes	One-to-one
Our Scheme	Yes	Single-Hop	No	Yes	Yes	Yes	Yes	One-to-many

the transformation of access structure, whereas BSW scheme does not support it; furthermore, our scheme has better performances on Private Key Generation, Encryption, and Decryption operations than those of BSW scheme. Performances on Re-Encryption Key Generation, Re-Encryption, and Decryption for the Re-Encrypted Ciphertext operations are analyzed.

8 Conclusions

In this work, for the settings such as cloud storage system where the access structures are frequently changed, we proposed a ciphertext policy attribute based proxy re-encryption scheme which delegates the proxy to transform the access structure associated with the original ciphertext without decrypting it. However, in our scheme, suppose there exists a single trusted authority, which may bring about a single point of failure. A user may possess attributes issued from multiple authorities and the data owner may share the data with users administered by different authorities. In order to enhance robustness, we will design multi authority CPAB-PRE scheme to transform the access structure associated with the ciphertext. The PRE algorithm in our scheme is only CPA-secure, and CPA security is often insufficient to guarantee security in general protocol settings.So we will address the problem of achieving CPAB-PRE scheme which is secure in arbitrary protocol settings, i.e., CCA secure.

Acknowledgments

The author gratefully acknowledges the anonymous reviewers for their valuable comments.

References

[1] Giuseppe Ateniese, Kevin Fu, Matthew Green, and Susan Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," in *the 12th Annual Network and Distributed System Security Symposium*, pp. 176–180, 2005.

[2] Giuseppe Ateniese, Kevin Fu, Matthew Green, and Susan Hohenberger, "Improved proxy re-encryption schemes with applications to secure distributed storage," *ACM TISSEC*, vol. 9, no. 1, pp. 1–30, 2006.

[3] Josh Benaloh and Jerry Leichter, "Generalized secret sharing and monotone functions," in *Advances in Cryptology - CRYPTO,vol. 403 of LNCS*, p. 27C36, Springer, 1988.

[4] John Bethencourt, Amit Sahai, and Brent Waters, "Ciphertext-policy attribute-based encryption," in *IEEE Symposium on Security and Privacy*, pp. 321 – 334, 2007.

[5] Matt Blaze, Gerrit Bleumer, and Martin Strauss, "Divertible protocols and atomic proxy cryptography," in *Proceedings of Eurocrypt 98*, vol. 1403, pp. 176–180, 1998.

[6] Dan Boneh, Xavier Boyen, and Eu-Jin Goh, "Hierarchical identity based encryption with constant size ciphertext," in *R.Cramer, editor, EUROCRYPT, vol. 3494 of Lecture Notes in Computer Science*, pp. 440–456, Springer, 2005.

[7] Dan Boneh, Eu-Jin Goh, and Toshihiko Matsuo, "Proposal for p1363.3 proxy re-encryption," in *http://grouper.ieee.org/groups/1363/IBC/submissions/NTTDataProposal-for-P1363*, September 2006.

[8] Ran Canetti and Susan Hohenberger, "Chosen-ciphertext secure proxy e-encryption," in *ACM CCS07*, pp. 185–194, New York, 2007.

[9] Ling Cheung and Calvin C. Newport, "Provably secure ciphertext policy abe," in *ACM Conference on Computer and Communications Security*, pp. 456–465, 2007.

Table 3: Performance comparison of our scheme with BSW scheme

References	**Private Key Generation**	**Encryption**	**Re-Encryption Key Generation**	**Re-Encryption**	**Decryption (Original Ciphertext)**	**Decryption (Re-Encrypted Ciphertext)**
BSW Scheme [4]	$(2\|S\|+1)\mathbb{G}_0$	$(2\|\mathbb{A}_1\|+1)\mathbb{G}_0+1\mathbb{G}_1$	Not Supported	Not Supported	$\|S'\|\mathbb{G}_1+2\|S'\|C_e$	Not Supported
Our Scheme	$(\|S\|+1)\mathbb{G}_0$	$(\|\mathbb{A}_1\|+1)\mathbb{G}_0+1\mathbb{G}_1$	$(\|\mathbb{A}_2\|+3)\mathbb{G}_0+1\mathbb{G}_1+1H$	$(\|S'\|+1)C_e$	$(\|S'\|+1)C_e$	$(\|S'\|+1)C_e+1H$

[10] Pei-Shan Chung, Chi-Wei Liu, and Min-Shiang Hwang, "A study of attribute-based proxy re-encryption scheme in cloud environments," *International Journal of Network Security*, vol. 16, no. 1, pp. 1–13, Jan. 2014.

[11] Vipul Goyal, Omkant Pandey, Amit Sahai, and Brent Waters, "Attribute-based encryption for fine-grained access control of encrypted data," in *Proceedings of the 13th ACM conference on Computer and communications security*, pp. 89 – 98, 2006.

[12] Matthew Green and Giuseppe Ateniese, "Identity-based proxy re-encryption," in *ACNS0, LNCS 4521*, pp. 288–306, New York:Springer, 2007.

[13] Anca Ivan and Yevgeniy Dodis, "Proxy cryptography revisited," in *the 10th Annual Network and Distributed System Security Symposium (NDSS03)*.

[14] Cheng-Chi Lee, Pei-Shan Chung, and Min-Shiang Hwang, "A survey on attribute-based encryption schemes of access control in cloud environments," *International Journal of Network Security*, vol. 15, no. 4, pp. 231 – 240, July 2013.

[15] Xiaohui Liang, Zhenfu Cao, Huang Lin, and Jun Shao, "Attribute based proxy re-encryption with delegating capabilities," in *Proceedings of the 4th International Symposium on Information, Computer, and Communications Security*, pp. 276 – 286, 2009.

[16] Amit Sahai and Brent Waters, "Fuzzy identity based encryption," in *Advances in Cryptology (Eurocrypt)*, vol. 3494 of LNCS, pp. 457 – 473. Springer, 2005.

[17] Victor Shoup, "Lower bounds for discrete logarithms and related problems," in *EUROCRYPT*, pp. 256–266, 1997.

[18] Shucheng Yu, Cong Wang, Kui Ren, and Wenjing Lou, "Attribute based data sharing with attribute revocation," in *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, pp. 261–270, 2010.

Xingbing Fu is a lecturer, he received his M.S. degree from Southwest University in 2007. He is currently a PhD Candidate in the School of Computer Science and Engineering, University of Electronic Science and Technology of China. His research interests are information security, cloud computing, cryptography and artificial intelligence.

Firewall Policy Diagram: Structures for Firewall Behavior Comprehension

Patrick G. Clark and Arvin Agah
(Corresponding author: Patrick G. Clark)

Department of Electrical Engineering and Computer Science
University of Kansas, Lawrence, KS 66045 USA
(Email: patrick.g.clark@gmail.com and agah@ku.edu)
(Received Apr. 9, 2014; revised and accepted Dec. 24, 2014)

Abstract

Communication security and regulatory compliance have made the firewall a vital element for networked computers. They provide the protections between parties that only wish to communicate over an explicit set of channels, expressed through protocols, traveling over a network. These explicit set of channels are described and implemented in a firewall using a set of rules. The firewall implements the will of the organization through an ordered list of these rules, collectively referred to as a policy. In small test environments and networks, firewall policies may be easy to comprehend and understand; however, in real world organizations these devices and policies must be capable of handling large amounts of traffic traversing hundreds or thousands of rules in a particular policy. Added to that complexity is the tendency of a policy to grow substantially more complex over time and the result is often unintended mistakes in comprehending what is allowed, possibly leading to security breaches. Therefore, it is imperative that an organization is able to unerringly and deterministically reason about network traffic, while being presented with hundreds or thousands of rules. This work seeks to address this problem using a data structure, the Firewall Policy Diagram, in an effort to advance the state of large network behavior comprehension.

Keywords: Firewall policy, firewall policy diagram (FPD), human comprehension, policy analysis

1 Introduction

Computer networking has arguably been one of the most important advancements in modern computing. Allowing disparate applications to trade information, conduct business, exchange financial transactions, and even the routine act of sending an email are some of the most common things we do with computers today. Even with the advancement of ever faster computer chips, the trend continues to connect devices at an astounding rate. In addition, there is also a thriving mobile device market, thus increasing the amount of traffic flowing between systems. An important aspect of this interconnected system is security. Without security, the convenience and speed of networked transactions would present more risk than the majority of applications could handle. In order to mitigate that risk and provide a much more secure communication channel, the firewall device was designed and deployed. It is one of the most widely used and important networking tools and exists in virtually every organization. In fact, over the past two decades the landscape of network security has come to rely heavily on that single type of device. The primary purpose of a firewall is to act as the first line of defense against malicious and unauthorized traffic, keeping the information that the organization does not want out, while allowing approved access to flow.

1.1 Firewall Basics

Firewalls allow two entities to connect their networks together through existing infrastructure and protocols, while securing the private networks behind them [16, 20]. The typical placement of a firewall is at the entry point into a network so that all traffic must pass through the firewall to enter the network. The traffic that passes through the firewall is typically based on existing packet-based protocols, and a packet can be thought of as a tuple with a set number of fields [16]. Examples of these fields are the source/destination IP address, port number, and the protocol field. A firewall will inspect each packet that travels through it and decide if it should allow that traffic to pass based on a sequence of rules. This sequence of rules is made up of individual rules that follow the general form:

$$\langle predicate \rangle \rightarrow \langle decision \rangle$$

The *predicate* defines a boolean expression over the fields in a packet tuple that are evaluated and the physical network interface from which the packet arrives. For example, source IP is 10.2.0.1 and destination IP address

is 192.168.1.1 on eth0 (a common label for a linux interface). Then the *decision* portion of a rule is what happens if the predicate matches to a true evaluation. A *decision* is typically accept or deny with the possibility of additional actions, such as an instruction to log the action [16]. However, for the purpose of this research we are only concerned with the accept or deny decision.

A firewall policy is made up of an ordered list of these rules such that as a packet is processed by the firewall, it attempts to match the packet to the predicate one rule at a time, from beginning of the rule list to the end. Matching the packet means that the firewall evaluates a packet based on the fields in the rule tuple, a packet matches the rule if it matches all the fields identified in the predicate of the rule [14]. The predicate does not necessarily need to contain a value for all possible fields and can sometimes contain the "any" variable in a field to indicate to the rule processing software that this is a "do not care" condition of the predicate and any value for that variable will match. It must completely match all the fields for the firewall to take the appropriate action. These rules are processed in order until the firewall finds a match, at that time it will take the appropriate action identified by the decision [14].

1.2 Motivation

The cost of a security breach has the potential to negatively impact business and cause large financial losses. This has been studied in the risk management area and falls under the avoidance topic [19]. The firewall is an important avoidance tool, however, over the past decade firewall rule-bases have grown in size at a remarkably fast pace. In a study finished in 2001, it was discovered that the typical organization will have 200 firewalls under the control of its network consisting of an average of about 150 rules per device [22]. In addition, these rule sets have been shown to grow to thousands of rules controlling routing between as many as 13 distinct networks [22]. More recent statistics gathered further support that the growth has only accelerated. In a study finished in 2009 the authors determined that the growth in complexity has out paced the growth in the organization's ability to synthesize and comprehend the changes [8]. The average number of rules has substantially increased from 150 in 2001 to 793, with a largest rule set found comprised of 17,000 rules [8]. The later study did not discuss the number of firewalls deployed, but in the unlikely case that firewall deployment growth stopped and the number of firewalls at an average organization stayed at 200 [22], then approximately 160,000 rules (200×793) would be under active management. In addition, the study also discovered that the average rule turnover (change) rate for an organization is 9.9% of the rules per month [8]. This means that an organization's firewall administration team has to accurately manage about 160,000 rules where 16,000 of those are changing on a monthly basis [8]. Therefore, the ability to accurately and confidently understand firewall policies and know what changes have occurred is more difficult than ever, and continues to increase in complexity.

1.3 Key Contributions

This work presents a novel set of data structures, together called a Firewall Policy Diagram (FPD). These data structures seek to solve the problem of large network behavior comprehension as it relates to firewall policies in several key areas:

- De-correlation of the firewall policy from the source rule set to gain a holistic view of behavior. This will remove any overlapping rules that will typically exist in a firewall policy [23] and the resulting data structure will model the actual ACCEPT and DENY space.
- Provide the ability to perform arbitrary mathematical set based operations like *and*, *or*, and *not*. These operations will assist in reasoning about firewall policy changes over time. They will also provide the foundation for many other firewall operations, such as understanding the functional differences between two policies. In addition, these operations will also provide the base for querying an arbitrary policy.
- Provide the foundation data structure for the implementation of a method to query the policy.
- Once a policy has been decomposed into an FPD, allow the reconstitution of that policy into a human comprehensible form, like an equivalent policy rule set.
- Finally, the experiments will show that these operations can be executed in seconds of computation time even on large policies (up to 10,000 rules).

The remainder of the paper is organized as follows: We begin with an overview of the FPD data structure and a description of how it is constructed, operated on, and translated into a set of de-correllated rules. We will then present the results of performance related experiments in terms of creation, SET operations, and reconstitution of firewall policies of various sizes. In the final two sections, related work and conclusions about FPDs will be covered.

2 Firewall Policy Diagram

A Firewall Policy Diagram is a set of data structures and algorithms used to model a firewall policy into an entity allowing efficient mathematical SET operations. The entity also has the ability to reconstitute the policy into a set of human comprehensible rules.

Table 1 demonstrates an access list that might be defined for a particular organization [14]. As shown, firewall policy traditionally consists of a list of rules. These rules are comprised of a subset of the fields in the Internet Protocol version 4 (IPv4) packet as defined by the

Open Systems Interconnection (OSI) model [14]. In this research effort only certain fields will be analyzed and modelled. The source address, destination address, protocol, and destination port will be used. The decision was made for two reasons, the source port is not often used in most firewall products and the flag is primarily used at layer three for setup and tear down of TCP connections [14]. However, if source port is necessary, it is a straightforward process to extend a FPD to include an additional 16 variables.

The internal storage mechanism of an FPD uses Reduced Ordered Binary Decision Diagrams (ROBDD or BDD) [5, 6, 18]. These data structures were introduced as an efficient way to capture hierarchical binary data and related works have described their use in firewall policy validation [11, 13, 23]. In addition to those that support the use of the BDD compressed data structure, there are research efforts that argue against its use in favor of other combination data structures [17]. However, in our work, the BDD provides the efficient storage and the necessary operations that allow our algorithms to reason about policy changes over time and differences between policies. Also, using network address translation (NAT) methods presented by [13], multi-firewall behavior over time can be modelled using BDDs, a missing research component in other policy comprehension work to date.

In a similar manner to the FIREMAN system [23], policies and rules are modelled as variable sets represented as BDDs. Using a BDD is an efficient way to represent a Boolean expression, like $(a \vee b) \wedge c$. Extending this concept to firewall policies, the variables in the expression become the bits of the associated IPv4 field. In this research, 32 bits representing the source address, 32 bits representing the destination address, 8 bits representing the protocol, and 16 bits representing the destination port. This means that for a particular ACCEPT space, there are 88 variables and 2^{88} potential combination of variable values.

2.1 Creation and Decomposition

When an FPD is initialized and created, the process begins by iterating over the policy rule set one rule at a time. Each rule is decomposed into its constituent parts relevant to our research: source, destination, protocol and destination port. At this point, each bit of each field is converted into an input vector of an appropriate size for the field. For example, the source field is 32 bits and therefore the vector is of size 32. Once the input vectors are constructed, the four input vectors are appended and added as a constraint in the underlying BDD [23].

2.2 Operations

Based on SET mathematics, if a data structure can accurately implement the *union*, *intersection*, and *complement* operation, other operations can be derived. For the purposes of this research and experimentation there are two primary operations that are being studied, namely, Difference and Symmetric Difference.

DIFFERENCE is used in the situation where there exists a base policy P and the desire is to understand what has changed in a later version of the policy, P'.

$$\begin{aligned} \triangle &= P' - P \\ &= P' \wedge \neg P \end{aligned}$$

SYMMETRIC DIFFERENCE is used in the situation where there exists two policies $P1$ and $P2$ and the desire is to know what is not shared between the two policies.

$$\begin{aligned} \triangle &= (P1 - P2) \vee (P2 - P1) \\ &= (\neg P2 \wedge P1) \vee (\neg P1 \wedge P2) \end{aligned}$$

Using these two operations with basic SET functions, we are able to reason about policy discrepancies and understand how one policy is related to another arbitrary policy. In addition, these operations can be chained together to produce a FPD' such that changes to a policy over time or as a set of access flows through multiple policies, the data structure can then be used to extract the resulting allowed (or denied) rules in a human comprehensible form.

2.3 Human Comprehension

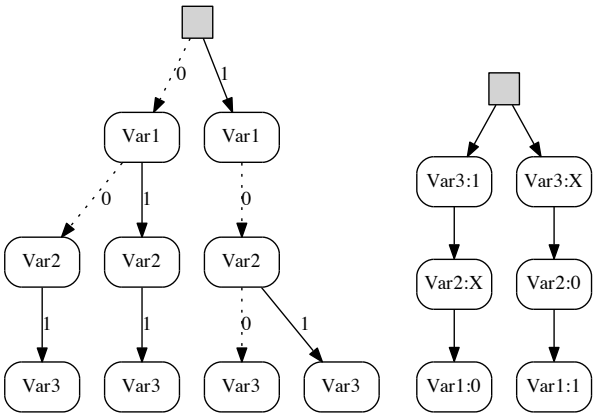
Using the resulting policy represented by $\triangle$, the procedure to reconstitute that policy into a human consumable set of rules involves transforming the BDD. The first step in the algorithm is to only explore the space necessary for the operation, typically the ACCEPT space. Therefore, starting at the root node the graph is traversed to the accept node and avoids having to explore the potentially large opposite space. During this traversal, the BDD is transformed into human comprehensible rules. To accomplish this sort of rule extraction without having to completely decompress the data structure, two additional data structures and algorithms are used.

The first data structure is a Ternary Tree, which, as the name suggests, consists of three child nodes for every parent. The purpose behind this tree is to take the fully compressed BDD and decompress portions of it without the full cost of the worst case of 2^{88} leaf nodes being represented. There is a low, high, and combination child node, such that the low represents a 0, the high represents a 1, and the combination represents both 0 and 1. The idea behind using a Ternary Tree was inspired by [5, 6] and other systems allowing the processing of a "do not care" variable, such as ternary content-addressable memory (TCAM). Therefore, for a particular sequence of ordered variables in a BDD, solutions that occupy the same space (i.e., ACCEPT) have variables of three potential values: 0, 1, or both. The representation of the potential values is *0*, *1*, and *X*, respectively. The definition of the *0* and *1* are the same as a binary tree, and the variable represented in the node assumes that value. The new edge

Table 1: Example access control list for a firewall interface

rule	action	src address	dest address	protocol	dest port
1	allow	172.16.0.0/16	10.2.0.0/16	TCP	80
2	allow	10.2.0.0/16	172.16.0.0/16	TCP	> 1023
3	allow	172.16.0.0/16	10.2.0.0/16	UDP	53
4	allow	10.2.0.0/16	172.16.0.0/16	UDP	> 1023
5	deny	all	all	all	all

value of X represents that it is both 0 and 1 for that particular variable at that particular node in the tree. The result is compression in the size of the tree by removing the need for a left and right sub tree.



(a) Binary Tree (b) Ternary Tree

Figure 1: Identical binary numbers in a (a) binary tree and (b) Ternary tree

Figure 1 illustrates the concept and what the model would resemble when a binary tree is represented as a Ternary Tree. The tree still maintains the hierarchical nature of the data, but in a more compressed format. There are two important differences as a binary tree transforms to a Ternary tree. The first is the representation of the values of a particular variable (identified by bit location) is stored with the node. The second difference is the order of the variables in the Ternary Tree, as they are representative of how a tree formed from the algorithm shown in Figure 2 resulting in the least significant bit (LSB) variable at the root. These differences allow the tree to be pruned in reverse such that the process begins at the root and generates intervals as it traverses to a leaf. The binary numbers represented in these identical trees are 1, 3, 4, and 5.

The Ternary Tree provides an intermediary between the BDD and the pruned rules by allowing ranges in data to be represented and subsequently combined as the tree is pruned. This is information that cannot be easily ascertained from a canonical BDD representation. The algorithm to transform the ROBDD into a Ternary tree is shown in Figure 2.

An additional concern is that a true tree structure has the potential to have a higher storage cost because of replicated data nodes in child trees when the pattern could be shared where appropriate. This has been addressed in the Ternary Tree by allowing it to share nodes where the underlying pattern is shared. The resulting rule set deals with any collisions that may occur when pruning by copying intervals. For example, if the pattern starting at a certain variable is common and shared by parent variables, then the Ternary Tree will share those nodes by an edge pointing to those nodes. This accomplishes the goal of reducing space requirements without sacrificing the expressiveness of the data structure.

Input: ROBDD Bdd
Output: A fully formed Ternary Tree T

```
1: procedure TRANSLATE(Bdd)
                    ▷ Bdd Variables labels start with 0
2:     T ← newTernaryTree
3:     for all R ∈ RootNodes(Bdd) do
4:         N ← createTernaryRoot(T)
5:         WALKEDGE(R.low, N)
6:         WALKEDGE(R.high, N)
7:     end for
8: end procedure
9: procedure WALKEDGE(bN, tN)
10:    for bN.parent.var to bN.var − 1 do
11:        tN ← tN.middle ← newTernaryNode(X)
12:    end for
13:    if bN.var = Bdd.One then return
14:    end if
15:    if bN.low ≠ Bdd.Zero then
16:        tN.left ← newTernaryNode(0)
17:        WALKEDGE(bN.low, tN.left)
18:    end if
19:    if bN.high ≠ Bdd.Zero then
20:        tN.right ← newTernaryNode(1)
21:        WALKEDGE(bN.high, tN.right)
22:    end if
23: end procedure
```

Figure 2: Algorithm for translation of a ROBDD to Ternary tree

for $i = Min;\ i <= Max;\ i += \ 2^{\ Conversion\ Factor}$ **do**
    $Value \leftarrow i$
end for

Figure 3: Algorithm for expressing all interval values

2.4 Pruning the Ternary Tree

The Ternary tree acts as an intermediate data structure where the primary purpose is to allow a second algorithm to collapse the tree into a set of human comprehensible data structures. The second algorithm is the pruning procedure that starts at the place-holder root of the Ternary tree and then traverses the tree to the leaves, pruning and generating intervals. The resulting data structure captures an interval and a count of bits in a conversion factor. The interval has a starting and ending number, with the conversion factor identifying how the interval sequence progresses. As an example, the source IP address Ternary Tree that represents an odd number of IP addresses is considered. In this example, the interval would be the starting and ending values of the range with a conversion factor of 1. All individual values in an interval can be expressed using the procedure shown in Figure 3.

As the tree is walked to the heuristically defined leaves of the tree, the interval is transformed by bit shifting and sometimes cloning of the interval to represent a transformation from a hierarchical data set into a rule. Additional details of the algorithm are identified in Figure 4. As the algorithm progresses, a number of these intervals are captured and get merged when appropriate. The internal representation of the ranges makes use of a red-black tree algorithm [9] to maintain a balanced structure while the intervals are assembled into a human readable form.

Notably, the entire rule is represented on the originating BDD; and therefore heuristics are used to help separate the data and make the rules more human comprehensible. This means that for a particular policy model, the data structure represents the concatenation of the source IP, destination IP, destination port, and protocol. The algorithms will separate the data structure into three separate Ternary Tree boundaries during the processing of the algorithm, namely, a source IP boundary, destination IP boundary, and service boundary. The process of extracting human comprehensible rules from an FPD runs at $O(V + E)$ where V and E are the number of vertices and edges in the Ternary trees, respectively.

The upper bound on the number of copies an interval must endure is 16. This is because a copy must occur when a variable transitions from 0 or 1 to X, and for a 32 variable number that can only occur a maximum of 16 times. This is also an upper bound on the number of intervals that could potentially exist at 2^{16}.

The final useful function achieved by using the interval data structure with conversion factor is a simple way to determine the number of addresses, ports or services in a particular rule. For an interval in a rule, it results in the

Input: Ternary Tree
Output: List of Rules that composed the SPACE

```
1: procedure PruneRules(T)
2:     for all L ∈ Children(T.root) do
3:         Interval ← create(from = 0, to = 0)
4:         Depth ← 0
5:         ParseNode(L, Interval, 0)
6:     end for
7: end procedure
8: procedure ParseNode(N, Interval, lnVal)
9:     if N.value = X and Interval.factor empty and
       lnVal ≠ X then
10:        Interval.factor ← Depth
11:    end if
12:    if N.value = X then
13:        Interval.to ← to | 1 << depth
14:    end if
15:    if N.value = 1 then
16:        Interval.from ← from | 1 << depth
17:        Interval.to ← to | 1 << depth
18:    end if
19:    if N is boundary then
20:        Rules ← Interval
21:        Interval ← create(from = 0, to = 0)
22:        Depth ← 0
23:        ParseNode(N.parent, Interval, 0)
24:    else if N not root then
25:        Depth = Depth + 1
26:        if lnVal ≠ X and N.value = X then
27:            Interval′ ← clone(Interval)
28:            ParseNode(N.left, Interval′, N.value)
29:            ParseNode(N.right, Interval′, N.value)
30:            ParseNode(N.middle, Interval′, N.value)
31:        end if
32:        ParseNode(N.parent, Interval, N.value)
33:    end if
34: end procedure
```

Figure 4: Algorithm for rule extraction

equation:

$$\left[\frac{(Interval\ Maximum - Interval\ Minimum)}{2^{\ Conversion\ Factor}}\right] + 1$$

The result of the pruning is a list of de-correlated rules with three intervals, the source IP range, destination IP range, and service range. An interesting result of the way that we have chosen to order the BDD variables is that the source IP to destination IP variable transition drives the number of distinct rules present in the final reconstituted rule set. This means that when that boundary in the variables is crossed while traversing the Ternary Tree, a new rule is generated and the remaining destination IP and service ranges are collected in that rule. Additional research could be done using the BDD variable reordering capability to find the most concise rule set representing a particular policy.

2.5 Heuristics Applied to Policies

Knowledge about the data set being modelled is important to the conversion and pruning algorithms. The heuristics provide the knowledge about where the hierarchical data sets begin and allow the summary of those data as the tree is pruned.

In this work the size of each of the fields drives the separation of the trees such that the 32^{nd}, 64^{th}, and 88^{th} variables divide the hierarchy of a solution into the source IP address, destination IP address, and service (a combination of protocol and port). Notably, these algorithms can be applied to other hierarchical data sets in different domains. They may be especially useful when dealing with large solution spaces and multiple hierarchies being combined to provide the composition of a desired "space".

2.6 Generalized Example

This section will step through an example of how the algorithms of the FPD function on a smaller solution space in an effort to both show how the boundary heuristics may be applied to other domains, as well as a better description of how the algorithms function. A example is considered where we represent the solution space as 2^8, with a fictitious rule being made up of two fields of 4 binary variables, A and B, which subsequently form an 8 binary variable solution space. As the stored ROBDD is generated, A will represent the decimal values 2 through 12 and B will represent the decimal values 3 through 13. Figure 5 visualizes the canonical ROBDD data structure representing A and B with the LSB at the root, and referred to as variable zero. As the ROBDD is traversed from root to leaf, the tree is transformed into the graph as seen in Figure 6. The variable transition values *high* and *low* are shifted to the variable values in the individual nodes. The new root node with label *1* is created and the tree is rooted at that node. In addition, the solution space removes the need for the edges leading to the *zero* value as we do not care about those numbers when extracting the stored data. The change from one tree to another is what is described in the algorithm shown in Figure 2.

As indicated in earlier explanations of the Ternary tree, it acts as an intermediary data structure as human comprehensible intervals are extracted. Therefore, the next step of process is to traverse the Ternary Tree from root to leaf in an effort to generate a set of *rules* with the described intervals that make up those rules. We have already identified our single boundary heuristic in this generalized example as variable 4 of our 8 variables (using a zero index). Therefore as the algorithm shown in Figure 4, begins with that knowledge by traversing the root to leaf in an effort to create intervals.

The initial result is two groupings or *rules* being generated. This can be visually confirmed in the Ternary tree with the existence of two nodes for variable 4. Subsequently, in each rule there are two sets of intervals, those representing the variables 0 through 3 and those representing the variables 4 through 7. Rule 1, segment 1 (variables 0-3):

$$2\ to\ 10\ every\ 2^3$$

$$3\ to\ 11\ every\ 2^3$$

$$4\ to\ 12\ every\ 2^3$$

$$5, 6, 7, 8, 9$$

Rule 1, segment 2 (variables 4-7):

$$4\ to\ 12\ every\ 2^3$$

$$6, 8, 10$$

Thus, the intervals may be merged into segment 1 (variables 0 through 3) being represented by the decimal numbers 2 through 12. Subsequently the segment 2 interval may be merged into 4 *to* 12 *every* 2, so *even* numbers 4 through 12. Based on a similar interval merging procedure for Rule 2, segment 1 (variables 0-3) becomes:

$$2\ to\ 10\ every\ 2^3$$

$$3\ to\ 11\ every\ 2^3$$

$$4\ to\ 12\ every\ 2^3$$

$$5, 6, 7, 8, 9$$

Rule 2, segment 2 (variables 4-7):

$$3\ to\ 11\ every\ 2^3$$

$$5\ to\ 13\ every\ 2^3$$

$$7, 9$$

The intervals may then be merged into segment 1 being represented by the decimal numbers 2 through 12. Then segment 2 may be merged into 3 *to* 13 *every* 2, so *odd* numbers 3 through 13.

The final merge may then be reviewed between two rules such that when a segment of a rule matches another segment, as it does with segment 1, it may become one rule with the segment that is not matching (segment 2),

being combined and reviewed for merges. Therefore, after rule 1 merges with rule 2, and since the segment 2 intervals represents overlapping odd and even ranges, i will become a contiguous decimal interval from 3 to 13. This reconstitutes our original rule that created the BDD: A = 2-12, B = 3-13; which is fit for human comprehension.

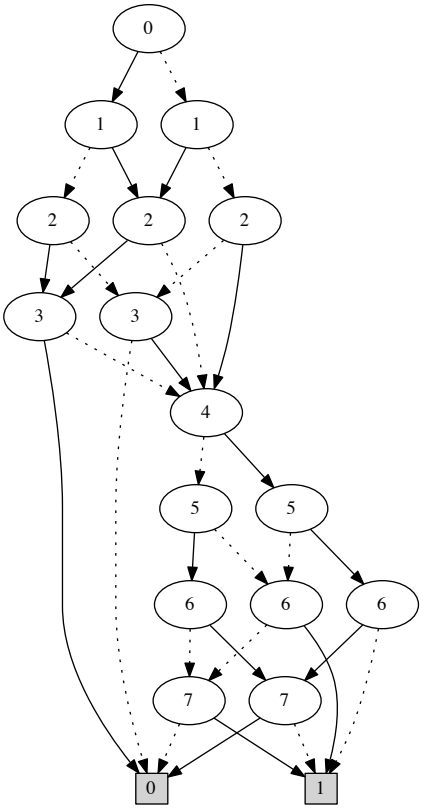


Figure 5: ROBDD generalized example

This algorithm is in contrast to a more naive, brute force approach to processing all known "solutions" to the BDD. In the example presented here, there are 64 8-bit numbers that will need to be parsed back into the known intervals that created the BDD from the start. While that number may seem small, the example and solution space are small by design. The important difference is between the number of solutions and the number of nodes in the Ternary tree. In addition, the brute force method removes the hierarchical relationships between the nodes, i.e. the heuristic value of node 4, further complicating reconstituting related intervals.

2.7 De-correlation

When a list of rules in a policy is decomposed into an FPD, a reconstituted policy that covers the same equivalent ACCEPT or DENY space will result in a rule set with none of the resulting rules overlapping in any area. The primary reason for this behavior is that as a policy is decomposed one rule at a time, all inconsistent or overlapping rules are removed and just the space is represented.

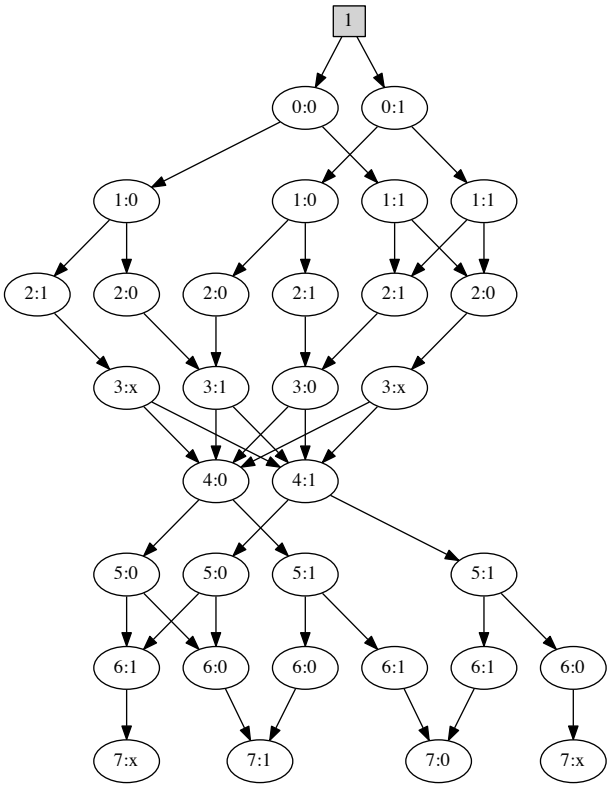


Figure 6: ROBDD as Ternary tree

The de-correlation property is useful in a number of scenarios:

- A policy no longer has a need to be processed as an ordered set of rules, since the FPD removes any overlapping rules. As a result, if the FPD is built by the rules in the policy from last to first, the resulting system can match an incoming packet to all rules simultaneously.
- A policy may be substantially smaller and take much less time to process once it has been de-correlated. This behavior is the effect of the procedure that converts rules into the FPD where it has also merged adjacent rules and removed any redundancies. In addition, the matching operation of a rule can be performed in constant time. This is because matching a rule involves walking the data structure from root to result, which is a constant 88 elements.

3 Experiments

The experiments designed for our work seek to review and address problems seen in the outside industry, i.e., large and difficult to understand firewall policies. We will first measure the time required to construct an FPD from rule sets of various sizes. We will then measure the time to extract a set of human comprehensible rules from an FPD. Finally, we will review the results of the DIFFERENCE and SYMMETRIC DIFFERENCE operations between two policies

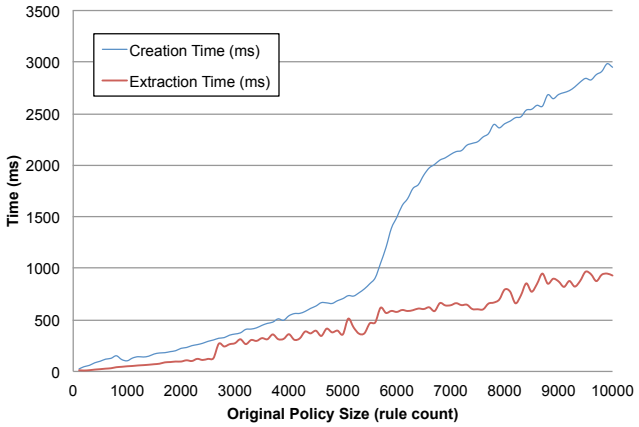


Figure 7: FPD generation and extraction of rules for policy sizes 100 to 10,000 rules

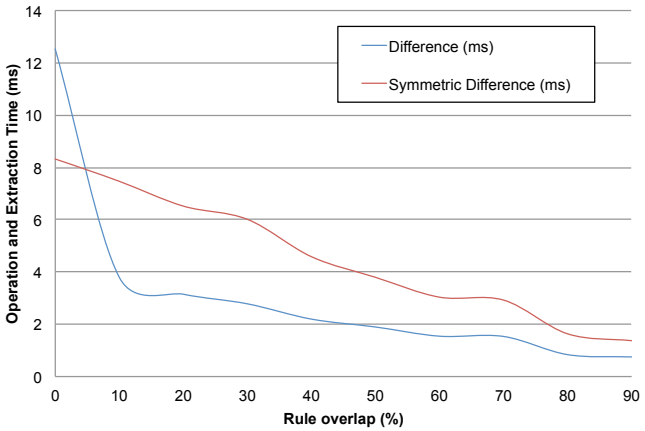


Figure 8: FPD difference and symmetric difference operations for a 200 rule policy

that share from 0% to 90% of the same space. The FPD data structure and algorithms are implemented in the Java™ 6 programming language and the tests are run on a Mac Book Pro laptop computer. For ROBDD software, the Buddy and JavaBDD libraries were used [15, 21].

For the data sets used to test creation and extraction of rules, policies of sizes 100 to 10,000 with 100 rule increments were created. When testing the DIFFERENCE and SYMMETRIC DIFFERENCE operations, we created two randomly generated 200 rule policies that share from 0% to 90% of the same space. Security reasons prevented us from being able to gain access to actual industry rule sets, as the majority of companies with firewall policies of those sizes are hesitant to allow outside parties access.

Figure 7 charts the performance of the FPD data structure for creation of a policy from a set of rules through to extraction of the policy into an equivalent set of rules. For creation of the FPD from a set of rules, the performance is consistently below one second for policies up to 5,500 rules. The creation times then begins to take more than a second until finally approximately 3 seconds to create a policy that originated from 10,000 rules. The extraction stays consistently less than a second to produce an equivalent set of rules and stays less than 500 ms for the majority of the data sets.

Figure 8 charts the performance of a DIFFERENCE operation and SYMMETRIC DIFFERENCE operation between two 200 rule policies. The experiment involved executing the appropriate operation; and then extracting the human comprehensible rules from the FPD. Notably, over 90% of the computation time is used producing human comprehensible rules. The actual DIFFERENCE and SYMMETRIC DIFFERENCE operations are averaging 5 milliseconds in all operations. SYMMETRIC DIFFERENCE operation exhibited slower processing performance due to the size of the resulting space represented by that operation. This means for that particular data set, the worst case representation of a space was very close to the maximum size of the potential *space*. DIFFERENCE operation yielded much smaller resulting data set sizes and reflected that fact in the computation times.

4 Related Work

Other work has been done modeling firewall policies, however, most of the models reflect their intended use and not all are capable of the sort of operations described in this work. Much of the research has focused on rule processing and validation of those rules where the goal is to identify redundant, shadowed, and inconsistent rules [1, 2, 3, 4, 7, 10, 16, 23]. In general the focus in those efforts is on algorithms for finding policy anomalies both from a single policy model to a multi-policy model. A portion of the related research introduced the use of BDDs for the models and became the foundation for some of the algorithms in our work [5, 6, 11, 18, 23].

In [12] the authors present a rule de-correlation algorithm with efforts to extract rules from existing firewall policies. The methods appear to be strictly for de-correlating rules with an exponential running time correlated to the size of the original policy and no overall policy comprehension.

One of the earliest works on policy comprehension built a query engine on top of a formal verification system named Voss [11]. Hazelhurst implemented a simple functional language that is capable of modeling a firewall policy in a formal verification system. While the underlying Voss hardware verification system used ordered binary decision diagrams to model a rule set, it is unclear the method in which the results were extracted from the canonical BDD form. The algorithm complexity was cited to be linear with respect to the number of rules for creation of the policy and constant time with respect to the number of variables in the BDD for any SET operations on the of the policy [11]. This is consistent with our work on processing FPDs. However, that analysis does not discuss the processing time related to extracting human comprehensible data from a binary decision diagram. This is

a large segment of processing and could involve millions of individual BDD solutions. Our analysis indicates that discussing the complexity as a function of the resulting BDD and the number of distinct solutions is a more accurate representation of the running time.

The research most similar to ours focuses on a data structure called a Firewall Decision Diagram [10, 16, 17]. The goal of the data structure is similar to ours, but the authors chose a different internal representation of a policy, specifically a combination of prefix and interval trees with additional data structures. It is capable of some DIFFERENCE operations, and is portrayed as a tool in which to achieve more accurate firewall design and change impact analysis. They specifically cite reasons for not using BDDs as the core of the data structure, although our work seeks to overcome the limitations referenced. In addition, it is not clear that they are capable of handling more complicated situations, such as network address translation, as a *space* is modelled through a real network.

5 Internet Protocol Version 6

Internet Protocol Version 6 (IPv6) is the latest iteration in the Internet protocol routing stack. It is an improvement to the current protocol IPv4, with the primary difference related to our research being the unique address space being expanded from 32 bits to 128 bits. This increase will allow many more devices to communicate and connect on the Internet. However, this also represents a challenge for firewalls and firewall administration teams. The growth of the addressable space means that these teams will need better and faster tools in which to comprehend how the firewalls under their control are configured and what sort of changes are happening to the security policies over time. This trend in the market space also supports the need for structures such as FPD for allowing the comprehension of these policies. While the experiments in this dissertation are on 32 bit addresses, expanding the FPD to use 128 bit addresses is a straightforward process and is planned for future experimentation. There is no reason that the algorithms would not function, although the size of the search space would become at least 2^{192} larger. This expansion is a result of the source and destination IP address space growing from 2^{32} to 2^{128}, therefore the number of bits needing to be represented grows $2^{96} \times 2^{96} = 2^{192}$.

6 Conclusions

In this paper we presented the Firewall Policy Diagram, an efficient and accurate data structure that serves as a basis for reasoning about firewall policy behavior and change. There are four primary contributions in this work:

- Data structures to efficiently model firewall policies that can be used to reason about them over time and modification.
- A data structure to act as the basis for the implementation of a means in which to query the policy.
- A set of algorithms in which to extract understandable rules from the FPD.
- Experimental evidence that these algorithms can perform the appropriate operations in seconds even on large firewall policy rule sets.

References

[1] E. S. Al-Shaer and H. H. Hamed. "Design and implementation of firewall policy advisor tools,". technical report, School of Computer Science, Telecommunications and Information Systems, DePaul University, Chicago, USA, August 2002.

[2] E. S. Al-Shaer and H. H. Hamed, "Discovery of policy anomalies in distributed firewalls," in *Proceedings of the 23rd Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 4, pp. 2605–2616, March 2004.

[3] E. S. Al-Shaer and H. H. Hamed, "Modeling and management of firewall policies," *IEEE Transactions on Network and Service Management*, vol. 1, pp. 2–10, April 2004.

[4] Y. Bartal, A. Mayer, K. Nissim, and A. Wool, "Firmato: a novel firewall management toolkit," in *Proceedings of the IEEE Symposium on Security and Privacy*, pp. 17–31, 1999.

[5] R. E. Bryant, "Graph-based algorithms for boolean function manipulation," *IEEE Transactions on Computers*, vol. C-35, pp. 677–691, August 1986.

[6] R. E. Bryant, "Symbolic boolean manipulation with ordered binary-decision diagrams," *ACM Computing Surveys*, vol. 24, pp. 293–318, September 1992.

[7] C. Chao, "A flexible and feasible anomaly diagnosis system for internet firewall rules," in *Proceedings of the 13th Asia-Pacific Network Operations and Management Symposium*, pp. 1–8, September 2011.

[8] M. J. Chapple, J. D'Arcy, and A. Striegel, "An analysis of firewall rulebase (mis)management practices," *ISSA Journal*, pp. 12–18, February 2009.

[9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. McGraw-Hill Higher Education, 2009.

[10] M. G. Gouda and A. X. Liu, "Firewall design: Consistency, completeness, and compactness," in *Proceedings of the 24th International Conference on Distributed Computing Systems*, pp. 320–327, 2004.

[11] S. Hazelhurst, A. Attar, and R. Sinnappan, "Algorithms for improving the dependability of firewall and filter rule lists," in *Proceedings of the 2000 International Conference on Dependable Systems and Networks*, pp. 576–585, 2000.

[12] E. Horowitz and L.C. Lamb, "A hierarchical model for firewall policy extraction," in *Proceedings of the 2009 International Conference on Advanced Information Networking and Applications*, pp. 691–698, may 2009.

[13] K. Ingols, M. Chu, R. Lippmann, S. Webster, and S. Boyer, "Modeling modern network attacks and countermeasures using attack graphs," in *Proceedings of the 2009 Computer Security Applications Conference*, pp. 117–126, December 2009.

[14] J. F. Kurose and K. W. Ross, *Computer Networking: A Top-Down Approach*. Addison Wesley, 4th edition, 2007.

[15] J. Lind-Neilsen. "Buddy version 2.4. http://sourceforge.net/projects/buddy,", 2004.

[16] A. X. Liu and M. G. Gouda, "Complete redundancy detection in firewalls," in *Proceedings of the 19th Annual IFIP Conference on Data and Applications Security*, pp. 196–209, 2005.

[17] A. X. Liu and M. G. Gouda, "Diverse firewall design," *IEEE Transactions on Parallel and Distributed Systems*, vol. 19, pp. 1237–1251, September 2008.

[18] C. E. Shannon, "A symbolic analysis of relay and switching circuits," *Transactions of the American Institute of Electrical Engineers*, vol. 57, pp. 713–723, December 1938.

[19] N. Sklavos and P. Souras, "Economic models and approaches in information security for computer networks," *International Journal of Network Security*, vol. 2, no. 1, pp. 14–20, 2006.

[20] M. Uma and G. Padmavathi, "A survey on various cyber attacks and their classification," *International Journal of Network Security*, vol. 15, no. 5, pp. 390–396, 2013.

[21] J. Whaley. "Javabdd version 1.0b2. http://javabdd.sourceforge.net,", 2007.

[22] A. Wool, "A quantitative study of firewall configuration errors," *Computer*, vol. 37, no. 6, pp. 62–67, 2004.

[23] L. Yuan, J. Mai, Z. Su, H. Chen, C. Chuah, and P. Mohapatra, "Fireman: A toolkit for firewall modeling and analysis," *IEEE Symposium on Security and Privacy*, pp. 199–213, 2006.

Patrick G. Clark is a Research Scientist for an information security analytics firm. He earned a Ph.D. degree in Computer Science with distinction from the University of Kansas in 2013 and his research interests include: rough-set theory, applied pattern recognition in medicine, IP network behavior models and big data algorithm analysis. He is a member of the ACM and of Upsilon Pi Epsilon, with over 30 publications in peer reviewed journals, conference proceedings and book chapters.

Arvin Agah is the Associate Dean for Research and Graduate Programs for the School of Engineering at the University of Kansas. He is the author of over 140 peer-reviewed articles and has earned multiple awards for research and graduate education. His research interests include algorithm analysis, artificial intelligence, applied AI in medicine and mobile robotics.

Improvement on Timestamp-based User Authentication Scheme with Smart Card Lost Attack Resistance

Heri Wijayanto[1,2], Min-Shiang Hwang[1,3]
(Corresponding author: Min-Shiang Hwang)

Department of Computer Science and Information Engineering, Asia University[1]
No. 500, Lioufeng Rd., Wufeng, Taichung, Taiwan 41354, R.O.C
Department of Information Engineering, Mataram University[2]
No. 62, Majapahit Rd., Mataram, Indonesa
Department of Medical Research, China Medical University Hospital, China Medical University[3]
No.91, Hsueh-Shih Road, Taichung, Taiwan 40402, R.O.C.
(Email: heri@te.ftunram.ac.id)
(Received Nov. 12, 2014; revised and accepted Jan. 29, 2015)

Abstract

Smart card-based user authentication is a useful mechanism for performing private session over an insecure network. Tang et al have proposed a robust and efficient scheme in 2013 that is based on elliptic curve discrete logarithm problem (ECDLP). It is for eliminating the attack in Awasthi et al's scheme. However, Tang et al's scheme is still vulnerable to denial of service attack and off-line password guessing attack. In this paper, the weakness of Tang et al's scheme is presented. Furthermore, it gives the improvement of Tang et al's scheme, and is proposed for avoiding the possible attack in Tang et al's scheme.

Keywords: Authentication, ECDLP, password, smart card

1 Introduction

A user authentication scheme based on smart card is growing rapidly in this decade. It avoids a use of user authentication table that should be kept in the server [1, 4, 6, 8, 9, 11, 13, 14, 18]. Several types of the lightweight user authentications include password-based approaches, symmetric encryption approaches, public-key encryption approaches, ID-based approaches, and the hybrid approaches [2, 3, 5, 10, 15, 19].

In 2013, Tang et al proposed a user authentication that is based on Elliptic Curve Cryptography or ECC [16]. ECC is a public key cryptography (PKC) that is better than previous PKC scheme. It is because, in the same security level, ECC has a smaller key length than RSA or El-Gamal scheme [7, 17].

However, Tang et al's scheme security only depends on the secure hash function security because the private key of server stored in the smart card is wrapped by a secure hash function. Actually, guessing a message that is compressed by a secure hash function needs a long time, but it is still not proper to store the secret key of server in all users' smart cards. The server secret key must be changed periodically for a security reason that is impossible to do in Tang et al's scheme because changing the secret key mechanism is not provided in Tang et al's scheme.

Besides that, Tang et al's scheme is also vulnerable to DoS attack. Denial of service (DoS) attack is the type of attack that exhausts a victim's resources by sending large amounts of packets or requests [5]. Therefore, the victim's computer will be lack of resources and cannot serve clients properly. In this unstable condition, the system will be vulnerable for other attacking protocols.

The remaining sections of this paper are organized as follows. Section 2 gives a brief review one of Tang et al's schemes and describes its weaknesses. In Section 3, we propose the improved scheme. In section 4, the security analysis of our scheme is given. Finally, Section 6 concludes the paper.

2 Brief Review of Tang's Scheme

Tang et al's scheme is based on ECDLP that improves Awasthi et al's scheme [16]. This scheme consists of four phases. There are system setup phase, registration phase, login phase, authentication phase, and password change phase. This section describes about Tang et al's scheme and its cryptanalysis as follows.

2.1 Tang et al's Scheme

In this step, all users and server agree on ECC parameters that will be used in this scheme. The server chooses a secret key x and computes $Q = x \cdot P$. Then server keeps x secret and publishes p, a, b, n, P, h, and Q.

This registration phase consists of three steps. In the first step, User U_i chooses identity ID_i and password PW_i freely. Then, he or she selects a random number N, and computes $HPW = h(PW_i||N)$. Next, U_i sends the ID_i and HPW to server S through a pre-established secure channel.

In the second step, S computes $V_i = h(ID_i||x) \oplus h(PW_i||N)$, stores $(V_i, h(\cdot))$ in smart card, and issues the smart card to U_i through secure channel. S also maintains an ID table that contains ID_i and status bit.

In the third step, after receiving the smart card, U_i stores N into a smart card. When U_i wants to log into a remote server S, U_i enters ID_i and PW_i. Then a smart card will do these two steps. First, the smart card computes $s = V_i \oplus h(PW_i||N)$, select a random $r_1 \in Z_n^*$, computes $R_1 = r_1 \cdot P$, $R_2 = r_1 \cdot Q$, and computes $V_1 = h(ID_i||R_1||R_2||s||T_c)$ where T_c is the timestamp at the login device. Actually, s is same with $h(ID_i||x)$ because $h(ID_i||x) \oplus h(PW_i||N) \oplus h(PW_i||N)$ is equal to $h(ID_i||x)$. Second, U_i sends $M_1 = (ID_i, R_1, V_1, T_c)$ to the server S through a common channel.

The third is the authentication phase that is divided into four steps. Step one, server S checks ID_i, status-bit, and T_c. If those three parameters pass the checking criteria, then continues to step two. If not, S will informs U_i about the failure. Step two, S sets the status bit to be 1, computes $R_2' = x \cdot R_1 = x \cdot r_1 \cdot P = r_1 \cdot Q$ and $s' = h(ID_i||x)$. Then S constructs $V_1' = h(ID_i||R_1||R_2'||s'||T_c)$. If V_1 is not equal to V_1', S rejects the login request and informs the user about it. On the other hand, S authenticates U_i and computes $V_2 = h(S||ID_i||R_2'||s'||T_s')$ and sends $M_2 = (V_2, T_s')$ to U_i. Steps three and four, after receiving M_2, U_i checks T_s' and V_2 by the similar way. In the end of the session, S the set status-bit to zero.

In the fourth phase or Password Change Phase, firstly, U_i needs to performs the Login Phase procedure and if it passes, U_i inputs the new password PW_i^*. In the step two, the smart card selects a random number N' and computes $V_i' = V_i \oplus h(PW_i||N) \oplus h(PW_i * ||N')$, and replaces V_i and N with the new V_i' and N'.

2.2 Cryptanalysis of Tang et al's Scheme

Tang et al's scheme is based on ECC that has two weaknesses. There are DoS attack and off-line password guessing attack.

2.2.1 DoS Attack

The main purpose of denial of service attack is turning off a service. Tang et al's scheme does not hide the ID in the login phase. The attacker can guess or steal it easily from an unsecured network connection. Then attackers will try the normal login by using stolen users ID or guessing ID, current T_c, and anything R_1, and V_1. This request will pass the ID checking and the status-bit of this ID is set to be one. Then, attackers will do the same way with different guessing ID's until all legal users can not use this service.

2.2.2 Off-Line Password Guessing Attack

In the Tang et al's scheme the secret key (x) of server is transmitted even though this is wrapped by secure hash function. In the other words, this scheme security does not depend on ECC but it is only based on the secure hash function security. Therefore, by finding the collision, the complexity of secure hash function will be decreased. There are some methods for attacking secure hash function such as Birthday attack, Joux's attack, and multi-collision attack [12].

3 The Proposed Scheme

In this paper, we propose an improvement of Tang et al's scheme. We add session key (R_2) and EC digital signature scheme.

3.1 System Setup Phase

This phase is equal to Tang et al's scheme. Server selects a secret key x and computes $Q = x \cdot P$ and keeps secret key x. After that, server publishes the public keys parameters p, a, b, P, n, h, and Q. In our scheme, server also saves random numbers k_i and M_i for ECC digital signature.

3.2 Registration Phase

Figure 1 shows the registration phase. It is done by users once in the first time they log-in to the server. Similar with Tang et al's scheme, it also uses secure communication line. It consists of three steps as follows:

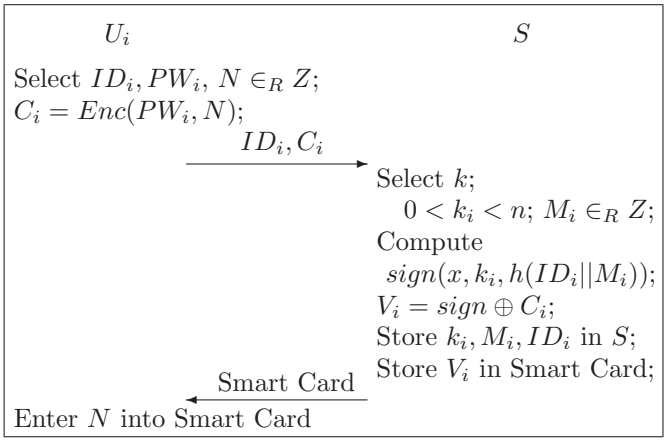


Figure 1: The registration phase of the proposed scheme

Step 1. User U_i selects an identity ID_i, password PW_i, and also a high entropy random number N. Then, users encrypt N by password PW_i as a symmetric key cryptography $C_i = Enc(PW_i, N)$. Next, user sends ID_i and C_i to the server through a secure channel.

Step 2. After receiving ID_i, and C_i, server selects a random number k_i that $0 < k_i < n$ and also a high entropy random number M_i. Next, server computes an EC digital signature by secret key x, and hash function of concatenation of ID_i and M_i as $sign(x, k_i, h(ID_i||M_i))$, for a short we call it sign. Then, server computes $V_i = sign \oplus C_i$, stores V_i into smart card and sends it back to user U_i through secure channel. Finally, server maintains an ID table that contains ID_i, status-bit, k_i, and M_i.

Step 3. After receiving a smart card, user inputs N into smart card.

3.3 Login Phase

In the login phase, the interaction between users and server utilize a common channel. Firstly, user inputs his or hers identity ID_i and password PW_i into a smart card. Then smart card computes $s = V_i \oplus C_i$ that equals to sign because $V_i = sign \oplus C_i$. Secondly, and the smart card chooses a random nonce $r_1 \in_R Z_n^*$, and computes $R_1 = r_1 \cdot P$, and $R_2 = r_1 \cdot Q$. Thirdly, the smart card encrypts $C_1 = ENC(R_2, ID_i||R_1||R_2||s||T_c)$ then sends R_1 and C_1 to server. This phase is shown in Figure 2.

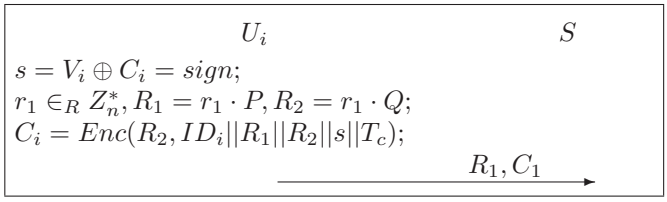


Figure 2: The login phase of the proposed scheme

3.4 Authentication Phase

The password change phase is shown in Figure 3. When a log-in requests that are R_1 and C_1 arrive to the Server S, S will do four passes that are described as bellow.

Pass 1. Server S computes the session key R_2' by secret key x as $R_2' = x \cdot R_1$. Then, Server decrypts C_1 by R_2', and this result is $ID_i||R_1||R_2||s||T_c$. If this decryption fail to produce those parameters, this login phase is rejected, and informs sender.

Pass 2. S checks the ID_i in the database. If this ID is not available in the database, S will reject this request and informs U_i in encrypted text by password R_2'.

Pass 3. S checks status-bit. If status-bit is equal to one, server rejects this request and informs U_i about it in encrypted text by password R_2', otherwise, server sets it to one.

Pass 4. S checks T_c. If $(T_s - T_c) \leq 0$ or $(T_s - T_c) > \triangle T$ server rejects this request and informs U_i in encrypted text by password R_2'.

Pass 5. Server computes its signature as $s' = sign(x, k_i, h(ID_i||M_i))$ and compares it with s. If those are not equal, S rejects this request and informs U_i about it. Otherwise, U_i has passed this authentication phase in the server side. And then, S encrypts $S||T_s$ by R_2' and sends back C_2 to user U_i in encrypted text by password R_2'. The next steps are done in the user side. U_i decrypts C_2 by R_2 and check S and T_c by the same way as server did. If those parameters do not satisfy the requirement criteria, U_i will reject this session.

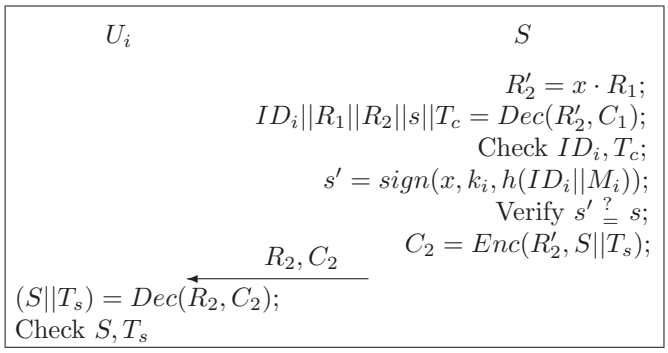


Figure 3: The authentication phase of the proposed scheme

3.5 Password Change Phase

When U_i wants to change his or her password for some reasons. U_i should keys his or her identity ID_i and password PW_i to a smart card first before changing the password. After that, Smart card will perform login protocol and if the login process is successful, U_i can input the new password $PW_{i,new}$. After that, the smart card generates new random number N_{new} and computes $V_i = V_i \oplus Enc(PW_i, N) \oplus Enc(PW_{i,new}, N_{new})$. Next, the smart card replaces V_i and N by $V_{i,new}$ and N_{new}. Finally, the smart card informs U_i that changing password is success.

4 Security Analysis

This proposed scheme also resists all attack explained in Tang et al's scheme [16]. In addition, this paper focuses to explain more about DoS attack and offline password guessing attack.

1) The Proposed Scheme Resists of DoS Attack.
In this scheme, User's ID is encrypted by using symmetric key cryptography before it is transmitted over an unsecured communication line. Therefore, attackers cannot steal it or guess it for DoS attack explained in Section 2.2.1 above. This improvement also fulfills the purpose of Chang et al's scheme [1].

2) The Proposed Scheme Resists of Offline Password Guessing Attack.
The weakness of Tang et al's scheme presented in Section 2.2.2 is storing hash value of the concatenation between user identity ID_i and server secret key x in the smart card. It is because of knowing x, the entire system will be down. In this proposed scheme, the secret key of elliptic curve cryptography (x) is not stored in the user's smart card. In this scheme, this hash value is replaced by EC signature.

5 Conclusions

In this paper, the weaknesses of a timestamp-based user authentication scheme with the smart card losing attack resistance have been discussed. Furthermore, the improvement of Tang et al's scheme is given by adding the session key and digital signature that are still based on the elliptic curve cryptography. Therefore, this scheme resists the denial of service attack and also offline password guessing attack.

Acknowledgments

This study was supported by the Ministry of Science and Technology of Taiwan under grant NSC102-2811-E-468 -001, MOST103-2622-E-468-001-CC2, and MOST103-2622-H-468-001-CC2. The authors gratefully acknowledge the anonymous re- viewers for their valuable comments.

References

[1] Chin-Chen Chang and Chia-Yin Lee, "A smart card-based authentication scheme using user identify cryptography," *International Journal of Network Security*, vol. 15, no. 2, pp. 139–147, 2013.

[2] T. H. Feng, C. H. Ling, and M. S. Hwang, "Cryptanalysis of Tan's improvement on a password authentication scheme for multi-server environ-ments," *International Journal of Network Security*, vol. 16, no. 4, pp. 318–321, 2014.

[3] D. He, W. Zhao, and S. Wu, "Security analysis of a dynamic ID-based authentication scheme for multi-server environment using smart cards," *International Journal of Network Security*, vol. 15, no. 5, pp. 282–292, 2013.

[4] Debiao He, Jianhua Chen, and Jin Hu, "Weaknesses of a remote user password authentication scheme using smart card," *International Journal of Network Security*, vol. 13, no. 1, pp. 58–60, 2011.

[5] Min-Shiang Hwang, Song-Kong Chong, and Te-Yu Chen, "Dos-resistant id-based password authentication scheme using smart cards," *Journal of Systems and Software*, vol. 83, pp. 163–172, Jan. 2010.

[6] Min-Shiang Hwang, Jung-Wen Lo, Chi-Yu Liu, and Shu-Chen Lin, "Cryptanalysis of a user friendly remote authentication scheme with smart card," *Pakistan Journal of Applied Sciences*, vol. 5, no. 1, pp. 99–100, 2005.

[7] Min-Shiang Hwang, Shiang-Feng Tzeng, and Chwei-Shyong Tsai, "Generalization of proxy signature based on elliptic curves," *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 73–84, 2004.

[8] M. Kumar, M. K. Gupta, and S. Kumari, "An improved efficient remote password authentication scheme with smart card over insecure networks," *International Journal of Network Security*, vol. 13, no. 3, pp. 167–177, 2011.

[9] Manoj Kumar, "An enhanced remote user authentication scheme with smart card," *International Journal of Network Security*, vol. 10, no. 3, pp. 175–184, 2010.

[10] C. C. Lee, C. H. Liu, and M. S. Hwang, "Guessing attacks on strong-password authentication protocol," *International Journal of Network Security*, vol. 15, no. 1, pp. 64–67, 2013.

[11] C. T. Li and Min-Shiang Hwang, "An efficient biometrics-based remote user authentication scheme using smart cards," *Journal of Network and Computer Applications*, vol. 33, pp. 1–5, 2010.

[12] M. Nandi and D. R. Stinson, "Multicollision attacks on some generalized sequential hash functions," *IEEE Transcactions on Information Theory*, vol. 53, pp. 759–767, Feb. 2007.

[13] R. Ramasamy and A. P. Muniyandi, "An efficient password authentication scheme for smart card," *International Journal of Network Security*, vol. 14, no. 3, pp. 180–186, 2012.

[14] Jau-Ji Shen, Chih-Wei Lin, and Min-Shiang Hwang, "Security enhancement for the timestamp-based password authentication scheme using smart cards," *Computers & Security*, vol. 22, no. 7, pp. 591–595, 2003.

[15] S. K. Sood, A. K. Sarje, and K. Singh, "Inverse cookie-based virtual password authentication protocol," *International Journal of Network Security*, vol. 13, no. 2, pp. 98–108, 2011.

[16] H. B. Tang, X. S. Liu, and L. Jiang, "A robust and efficient timestamp-based remote user authentication scheme with smart card lost attack resistance," *International Journal of Network Security*, vol. 15, pp. 446–454, Nov. 2013.

[17] Shiang-Feng Tzeng and Min-Shiang Hwang, "Digital signature with message recovery and its vari-

ants based on elliptic curve discrete logarithm problem," *Computer Standards & Interface*, vol. 26, no. 2, pp. 61–71, 2004.

[18] L. Yang, J. F. Ma, and Q. Jiang, "Mutual authentication scheme with smart cards and password under trusted computing," *International Journal of Network Security*, vol. 14, no. 3, pp. 156–163, 2012.

[19] X. Zhuang, C. C. Chang, Z. H. Wang, and Y. Zhu, "A simple password authentication scheme based on geometric hashing function," *International Journal of Network Security*, vol. 16, no. 4, pp. 271–277, 2014.

Heri Wijayanto earned his BS and MS degrees in Electrical Engineering Gadjah Mada University Indonesia in 1998 and 2002. He works as a lecturer in Mataram University Indonesia since 2002 and currently, he continues his study in Computer Science and Information Engineering, Asia University Taiwan started in 2014. His research interest are data mining, and computer security.

Min-Shiang Hwang received the B.S. in Electronic Engineering from National Taipei Institute of Technology, Taipei, Taiwan, Republic of China (ROC), in 1980; the M.S. in Industrial Engineering from National Tsing Hua University, Taiwan, in 1988; and the Ph.D. in Computer and Information Science from National Chiao Tung University, Taiwan, in 1995. He also studied Applied Mathematics at National Cheng Kung University, Taiwan, from 1984-1986. Dr. Hwang passed the National Higher Examination in field Electronic Engineer in 1988. He also passed the National Telecommunication Special Examination in field Information Engineering, qualified as advanced technician the first class in 1990. From 1988 to 1991, he was the leader of the Computer Center at Telecommunication Laboratories (TL), Ministry of Transportation and Communications, ROC. He was also a chairman of the Department of Information Management, Chaoyang University of Technology (CYUT), Taiwan, during 1999-2002. He was a professor and chairman of the Graduate Institute of Networking and Communications, CYUT, during 2002-2003. He is currently a chair professor of the department of Computer Science and Information Engineering, Asia University, Taiwan, ROC. He obtained 1997, 1998, 1999, 2000, and 2001 Outstanding Research Awards of the National Science Council of the Republic of China. He is a member of IEEE, ACM, and Chinese Information Security Association. His current research interests include electronic commerce, database and data security, cryptography, image compression, and mobile computing. Dr. Hwang had published 200+ articles on the above research fields in international journals.

A Novel Threshold Conference-Key Agreement Protocol Based on Generalized Chinese Remainder Theorem

Cheng Guo[1] and Chin-Chen Chang[2,3]
(Corresponding author: Chin-Chen Chang)

School of Software Technology, Dalian University of Technology, China[1]
No. 8 Road, Jinzhou District, Dalian, 116620 China
Department of Information Engineering and Computer Science, Feng Chia University[2]
No. 100, Wenhwa Road, Seatwen, Taichung, Taiwan 40724, R.O.C.
Department of Computer Science and Information Engineering, Asia University[3]
500, Lioufeng Road, Wufeng, Taichung, Taiwan 41354, R.O.C.
(Email: alan3c@gmail.com)
(Received May 24, 2014; revised and accepted Nov. 25, 2014)

Abstract

The conference-key agreement protocol is a mechanism for generating a common session key among the authorized conference members. The common session key is used to encrypt communication messages transmitted over an open network. Inspired by traditional key agreement protocols and threshold cryptosystems, we have proposed a novel threshold conference-key agreement protocol in this paper. In the proposed protocol, we used a secret sharing scheme based on the generalized Chinese remainder theorem (GCRT) to achieve the threshold characteristic, and we can alter the shared data by adjusting an additional parameter k of the GCRT. If the number of conference members involved in generating the conference key exceeds a certain number, the members can cooperate to generate a valid common session key that also can be verified and used by other authorized conference members. Compared with traditional key agreement protocols, the proposed protocol has some unique characteristics that are beneficial in real applications.

Keywords: Conference-key agreement, generalized Chinese remainder theorem, threshold cryptosystem

1 Introduction

With the rapid development of Internet technology and its growing popularity, group-oriented applications and protocols have become increasingly important. A group of people can use the Internet to communicate at anytime and from their various locations instead of having to assemble in one location, thereby saving a lot of time and money. Because of the convenience of the network, web conferencing has become the trend of future development. A web conference can be held by connecting conference members located in different areas, even different continents, via the Internet. However, the network is an open environment, which means that it is vulnerable to a variety of attacks, such as masquerade attacks, replay attacks, and the modification of messages. Regardless of the underlying environments, communication privacy and integrity in the group are essential. A general and effective method for ensuring the confidentiality of the messages transmitted among participants is to establish a common session key for encrypting their communications over an insecure channel.

Conference-key establishment protocols can be classified roughly into two categories i.e., key agreement protocols [3, 7, 9, 10, 11, 12, 16, 20, 24, 25, 26, 27] and key distribution protocols [5, 6, 8, 13, 14, 21]. In a conference-key distribution protocol, a trusted third party, called a key generation center, is responsible for generating and distributing the conference key to authenticated conference members. Conference-key agreement is a mechanism in which a group of conference members computes a function $K = f(k_1, k_2, \cdots, k_n)$ securely, where k_i is a conference member's private input. There are some potential secure problems. First, conference member's private input must be transmitted in an open channel. That could allow some non-authorized attackers to obtain the conference key and listen to or observe the content of the communication. Second, attackers also can try to impersonate authenticated members. The basic technique for solving the above security issues is the utilization of public key cryptology, such as RSA, ElGamal, or ECC, to confirm the conference members' identities and to guarantee the

confidentiality of the function $K = f(k_1, k_2, \cdots, k_n)$.

It is well known that Diffie and Hellman (DH) [7] proposed a protocol that can establish a common key between two parties. Most group key-management protocols, including those developed by Ingemarsson et al. [9]; Burmester and Desmedt [3]; Steiner et al. [24]; and Just and Vaudenay [11], have attempted to extend the elegance and simplicity of Diffie-Hellman's two-party key exchange to the group setting. Just and Vaudenay [11] proposed an authenticated, multi-party, key-agreement protocol in which the group members could interact via an exchange of messages to obtain a common session key by using DH and public-key techniques without requiring a trusted third party. There has been intensive research on conference-key agreement protocols [10, 12, 16, 20, 25, 26, 27], such as the study of key agreement protocols in dynamic peer groups [16, 25].

In 1979, the first (t, n) threshold schemes, based on Lagrange interpolation and liner project geometry, were proposed by Shamir [23] and Blakley [2], respectively. Other well-known secret sharing schemes include Mignotte's scheme [18] and the Asmuth-Bloom scheme [1], which were based on the Chinese remainder theorem (CRT). With the emergence of different kinds of applications, threshold cryptosystems have been studied extensively, and threshold cryptosystems and their many variations form an important research direction.

In traditional conference-key agreement protocols, conference members u_i, for $i = 1, 2, \cdots, n$ who want to establish a secure channel for transferring confidential information must cooperate to compute a common conference key K, using each conference member's input k_i. No information about K can be obtained from a protocol run without knowledge of at least one of the k_i. Most existing conference-key agreement protocols have followed this pattern. However, in reality, there are many situations in which the ability to hold the conference is determined by a certain number of people. Once it has been decided that the conference will take place, all conference members must participate in the conference. Consider the following scenario: In a board meeting, the condition for holding the conference is that a certain number of members of the board of directors must agree that the meeting should proceed. Then, once it has been decided that the board meeting will occur, all members of the board of directors must participate in the meeting. However, to the best of our knowledge, traditional conference-key agreement protocols cannot meet this requirement. Existing conference-key agreement protocols do not have threshold characteristics, and other members cannot obtain the conference key or verify the validity of the conference key if they were not involved in the generation of the conference key. For the above reasons, we have focused on the threshold conference-key agreement in this paper.

We propose a novel threshold conference-key agreement protocol that has the following characteristics:

1) The conference key can be obtained if and only if at least t or more conference members cooperate to generate the conference key;

2) The other authorized members who do not participate in the generation of the conference key also can obtain the conference key;

3) The other authorized members can verify the validity of the conference key and communicate with other conference members using this common session key.

The remainder of the paper is organized as follows: In the next section, we describe our main objective. In Section 3, we introduce some preliminaries. In Section 4, we propose our threshold conference-key agreement protocol based on the generalized Chinese remainder theorem. The security and performance analysis are presented in Section 5. Finally, we present our conclusions in Section 6.

2 Objective

In this section, we describe the design principles of our threshold conference-key agreement protocol and then present the security requirements for the proposed protocol.

2.1 Design Principles

In the proposed scheme, same as for the existing conference-key agreement protocols, a common key can be reached among the conference members. However, there are some distinct characteristics in the proposed threshold conference-key agreement protocol. The idea of our design is to allow a certain number of conference members to decide whether the conference should be held and what the conference key will be. In the proposed scheme, one authorized conference member can initiate a request for a conference. Then, if a sufficient number of conference members agree and are willing to cooperate to generate the common conference key, the common key can be computed. But if the number of conference members who agree is less than the threshold value, they cannot construct a valid conference key.

Our protocol uses threshold Mignotte secret sharing and the generalized Chinese remainder theorem (GCRT) as building blocks. First, we utilize the Mignotte secret sharing scheme based on CRT to achieve the threshold access structure that can be used to obtain the conference key among the conference members. It is well known that GCRT is a variation of CRT, and in a similar manner, the threshold access structure based on CRT also can be achieved by using GCRT. In the Mignotte secret sharing scheme, secret data can be represented by a corresponding congruence system. However, if we want to modify the shared data, we have to readjust the corresponding congruence system. In conference-key agreement protocols, key independence is an important property. That

is, the previous conference keys are irrelevant to the subsequent conference keys. The conference key can be obtained by evaluating a function $f(k_1, k_2, \cdots, k_n)$, where $k_1, k_2, \cdots, k_n$ are the n conference members' private inputs. Therefore, the Mignotte secret sharing scheme based on the traditional CRT is inappropriate for this situation. In GCRT, an additional modulus k is provided to strengthen and enlarge the CRT's applications. We can change the shared data by adjusting the parameter k, which can be computed by collecting a certain number of conference members' inputs, k_i. In Section 3, we provide a brief introduction to the Mignotte secret sharing scheme and the generalized Chinese remainder theorem.

First, each conference member must register at the key generation center (KGC). During the registration process, the KGC shares the necessary information for key agreement with each authorized conference member. A conference may be requested by any authorized member. For example, if conference member U_i wants to convene a conference, he or she must participate in the key-generation process and send some messages in an authenticated broadcast channel. If enough conference members participate in the key-generation phase, the conference key can be computed. Meanwhile, all authorized members can obtain the conference key and verify its validity. Because we use GCRT, a new conference key can be generated by modifying an additional parameter k, which does not require any change in the information shared with the KGC by each member.

The proposed protocol also supports changes in the conference members, such as when members join or leave the group.

2.2 Design Principles

In this section, we summarize the desired security goals for our threshold conference-key agreement protocol. Referring to the protocol developed by Kim et al. and Harn et al. [8, 12], we define the following security properties:

1) Key authentication guarantees that the conference key can be generated by authorized conference members, but not by fraudulent attackers;

2) Key confidentiality guarantees that the conference key only can be obtained by authorized conference members;

3) Forward secrecy is that a compromise of the previous conference key cannot disclose the current key;

4) Backward secrecy is that an earlier key cannot be obtained if the current conference key is compromised.

In our protocol, we utilize the RSA cryptosystem [22] to protect the authenticity and confidentiality of the key as proposed by Zhao et al.'s protocol [26]. Concerning forward and backward secrecy, the characteristic of the GCRT can ensure the conference key's freshness and independence. That is, old keys and the current key cannot be derived from each other. We will give the details in Section 5.

3 Preliminaries

In this section, we briefly introduce the threshold Mignotte secret sharing scheme [18] and the generalized Chinese remainder theorem [4, 15], which are the major building blocks of our protocol.

3.1 Threshold Mignotte Secret Sharing Scheme

In 1982, Mignotte [18] proposed a threshold secret sharing scheme that uses some special sequences of integers, referred to as the Mignotte sequences.

Let n be a positive integer and $2 \leq t \leq n$. A (t, n)-Mignotte sequence is a sequence of positive integers $p_1 < p_2 < ... < p_n$, such that $\prod_{i=0}^{t-2} p_{n-i} < \prod_{i=1}^{t} p_i$, where $p_1, p_2, \cdots, p_n$ are co-prime in pairs.

Given a public (t, n)-Mignotte sequence, the scheme works as follows:

1) The secret y can be chosen as a random integer such that $b < y < a$, where $a = \prod_{i=1}^{t} p_i$ and $b = \prod_{i=0}^{t-2} p_{n-i}$;

2) The shadows y_i are computed such that $y_i = y \bmod p_i$, for all $1 \leq i \leq n$;

3) Given t distinct shadows $y_1, y_2, \cdots, y_t$ and t corresponding modulo $p_1, p_2, \cdots, p_t$, using the Chinese remainder theorem, the secret y can be recovered in the following congruence system:

$$\begin{cases} y \equiv y_1 \bmod p_1 \\ y \equiv y_2 \bmod p_2 \\ \vdots \\ y \equiv y_t \bmod p_t. \end{cases}$$

3.2 Generalized Chinese Remainder Theorem

GCRT [4, 15] is a variation of CRT. Similar to CRT, n positive co-prime integers $m_1, m_2, \cdots, m_n$ and $\{x_1, x_2, \cdots, x_n\}$ are needed to construct a system of simultaneous congruencies. In GCRT, an additional modulus k is required during the computations, where $\text{Max}\{x_i\}_{1 \leq i \leq n} < k < \text{Min}\{m_j\}_{1 \leq j \leq n}$. A number X can be represented by using $\{x_1, x_2, \cdots, x_n\}$, where $x_i = \lfloor X/m_i \rfloor \bmod k$,for $i = 1, 2, \cdots, n$. According to the

GCRT, the number X can be computed as follows:

$$\begin{aligned} X &= \sum_{i=1}^{n} M_i * M'_i * N_i (\mathrm{mod}(k * \prod_{i=1}^{n} m_i)), \quad \text{where} \\ M_i &= k * \prod_{j=1, j\neq i}^{n} m_j, \\ M_i * M'_i &= k(\mathrm{mod} k * m_i) \\ N_i &= \lceil x_i * m_i / k \rceil . \end{aligned}$$

4 The Proposed Protocol

In this section, we develop an extension to the existing conference-key agreement protocols. In the proposed protocol, the conference key can be computed if and only if the number of involved conference members who want to cooperate to generate a conference key exceeds a certain threshold. Our protocol consists of four phases: registration, sub-key distribution and commitment, sub-key recovery, conference-key derivation and verification. Table 1 is used notations throughout the remainder of this paper.

Table 1: The notations

n	number of conference members
t	threshold
U_i	i-th conference member, $i \in [1, n]$
ID_i	identity of the conference member U_i
SN_ℓ	unique serial number for the ℓth conference-key
k_i	sub-key generated by U_i
K	conference-key
$h()$	collision-free one-way hash function

4.1 Registration Phase

As described in Sections 3.1 and 3.2, we can construct a new secret-sharing scheme based on GCRT instead of CRT. First of all, KGC generates n positive coprime numbers $m_1, m_2, \cdots, m_n$ that satisfy the Mignotte sequence characteristics and selects n positive integers $x_1, x_2, \cdots, x_n$. KGC randomly selects a number k that satisfies $\mathrm{Max}\{x_i\}_{1\leq i\leq n} < k < \mathrm{Min}\{m_j\}_{1\leq j\leq n}$. The initial value of the conference key K can be represented by n-tuple $\{x_1, x_2, \cdots, x_n\}$ and $k\prod_{i=0}^{t-2} m_{n-i} < K < k\prod_{i=1}^{t} m_i$.

We assume that there are n conference members and that each conference member is required to register at KGC before the conference-key agreement. Learning from Zhao et al.'s group key-agreement protocol [26], we also utilize the RSA public-key cryptosystem [22] to guarantee the authentication, confidentiality, and integrity of the sub-key.

Upon receiving a registration request from any conference member U_i, KGC will perform the following steps:

1) According to the properties of the RSA cryptosystem, KGC computes $N_i = p_i \times q_i$, where p_i and q_i are two large prime numbers selected randomly, and factoring N_i is hard;
2) KGC generates the corresponding private key d_i and public key e_i that satisfy $d_i \times e_i \equiv \varphi(N_i)$, where $\varphi(N_i)$ is the Euler phi-function;
3) KGC distributes the secret value pair (x_i, m_i) and private key d_i to conference member U_i over a secure channel;
4) KGC publishes the public key $\{e_i, N_i\}$.

4.2 Sub-key Distribution and Commitment Phase

The protocol starts with an initiator initiating a conference request for a set $U = \{U_1, U_2, \cdots, U_n\}$ of conference members and the initiator broadcasts a unique serial number SN_ℓ for the conference. If any member U_i wants to accept the conference request, he will perform the following steps:

1) U_i randomly selects its sub-key k_i, $\mathrm{Max}\{x_i\}_{1\leq i\leq n} < k_i < \mathrm{Min}\{m_j\}_{1\leq j\leq n}$;
2) Compute $\mu_i = (x_i||m_i||k_i||SN_\ell)^{d_i} \bmod N_i$;
3) Compute $\sigma_{ij} = (\mu_i||ID_i)^{e_j} \bmod N_j$ for $U_j (i \neq j)$ and $h_i = h(x_i||m_i||k_i||SN_\ell)$;
4) Publish $\omega_i = \{h_i, \sigma_{ij}, \mathrm{for} j = 1, 2, ..., n, j \neq i\}$.

4.3 Sub-key Recovery Phase

Without losing generality, all conference members can receive ω_i from U_i for $i = 1, 2, \cdots, t$. Then, they can compute the corresponding sub-key k_i and check the validity of the sub-key. The details are as follows:

1) U_j computes $\mu'_i||ID_i = (\sigma_{ij})^{d_j} \bmod N_j$ using her or his private key d_j and reads μ'_i and ID_i. Then, he or she uses the corresponding public key e_i to compute $(x_i||m_i||k_i||SN_\ell)' = (\mu'_i)^{e_i} \bmod N_i$;
2) U_j checks whether or not SN'_ℓ is the current conference serial number SN_ℓ and computes $h'_i = h(x'_i||m'_i||k'_i||SN'_\ell)$ and verifies the equation $h'_i = h_i$;
3) If the equation holds, then U_j can retrieve the corresponding $\{x_i, m_i, k_i\}$.

4.4 Conference-key Derivation and Verification Phase

If a sufficient number of conference members respond to the conference request, the conference key can be derived from sufficient sub-key information $\{x_i, m_i, k_i\}$ for $i = 1, 2, \cdots, t$. The procedure consists of two phases: (1) the conference-key derivation phase and (2) the conference-key verification phase.

(1) Conference-key derivation phase

1) Once U_j obtains t verified sub-key information ω_i, U_j can first compute $k = \left\lfloor \frac{\sum_{i=1}^{t} k_i}{t} \right\rfloor$ without losing generality;
2) According to the GCRT and Mignotte secret sharing, the conference key K can be computed by each conference member U_j as follows:

$$K = \sum_{i=1}^{t} M_i * M'_i * N_i (\mathrm{mod}(k * \prod_{i=1}^{t} m_i)), \quad (1)$$

where

$$\begin{aligned} M_i &= k * \prod_{j=1, j\neq i}^{t} m_j, \\ M_i * M'_i &= k(\mathrm{mod} k * m_i) \\ N_i &= \lceil x_i * m_i / k \rceil . \end{aligned}$$

(2) Conference-key verification phase

As mentioned above, all authorized conference members can compute the conference key, even though they do not participate in the generation of the conference key. Meanwhile, they also have the capacity to verify the validity of the conference key. Each authorized member U_j can use her or his (x_j, m_j) to check the following equation:

$$x_j \stackrel{?}{\equiv} \left\lfloor K/m_j \right\rfloor \bmod k \quad (2)$$

If Equation (2) holds, the conference-key K is validated. This shows that at least t authorized conference members agree to convene the network conference. That is, all conference members must participate in this conference, and all conference members can share a common secret conference key used to encrypt the conference message.

4.5 Conference Members Join and Leave

Concerning a conference group, new members may join such a group, thereby increasing the number of members; also, some members may leave the group, thereby decreasing its number of members. Therefore, the proposed scheme must support these potential occurrences.

If a new member U_j wants to join the conference group, he or she must register at the KGC. The KGC selects a corresponding m_j that meets the characteristic of the Mignotte sequence. According to the current K and k, KGC can compute $x_j = \lfloor K/m_j \rfloor \bmod k$ and send (x_j, m_j) and the private key d_j to U_j over a secure channel. Then, KGC publishes the corresponding public key $\{e_j, N_j\}$.

If a conference group member U_j leaves the group, KGC will broadcast the information about U_j's leaving the conference group. Then, in the sub-key distribution and commitment phase, the conference members will not use U_j's public key to encrypt the sub-key information.

4.6 A New Conference-key is Computed

If a new conference-key is required, we need to change the shared the secret number aiming at modifying the shared conference-key. If we utilize Shamir's secret sharing based on a Lagrange interpolation polynomial or Mignotte's secret sharing based on CRT to construct a threshold protocol, then, if we want to change the shared number, we have to modify the corresponding interpolation polynomial or the corresponding congruence system. That is, we have to update the data stored by each authorized conference member. That means that the registration phase has to be replayed and all conference members have to obtain a new secret shadow. However, using the GCRT, we can simply solve the problem instead of changing the interpolation polynomial or the whole congruence system. We can change the shared number by adjusting the parameter k. Therefore, the conference members can execute the rest of the protocol except for the registration phase to compute a new conference-key.

5 Discussions

In this section, we present the security and performance analysis of our protocol.

5.1 Security Analysis

As discussed in Section 2.2, the main security requirements of a threshold conference-key agreement are: conference-key authentication, conference-key confidentiality, and forward/backward secrecy. In this section, we show that the proposed protocol satisfies those four security requirements and can resist some types of attacks.

5.1.1 Conference-key Authentication

In the process of the generating the conference key, each conference member U_i computes her or his sub-key information and encrypts the information using her or his private key d_i. Meanwhile U_i computes $h_i = h(x_i||m_i||k_i||SN_\ell)$. The other members can decrypt this message using U_i's public key and derive the corresponding sub-key information $(x_i||m_i||k_i||SN_\ell)'$. So, the other members can confirm that this message was sent by U_i. And, the other members can compute h'_i and check whether $h'_i = h_i$ in order to verify the integrity of the sub-key information.

5.1.2 Conference-key Confidentiality

Before considering conference-key confidentiality, we briefly illustrate key freshness. The conference key K can

be computed as follows:

$$K = \sum_{i=1}^{t} M_i * M'_i * N_i (\mathrm{mod}(k * \prod_{i=1}^{t} m_i)), \quad \text{where}$$
$$k = \left\lfloor \frac{\sum_{i=1}^{t} k_i}{t} \right\rfloor.$$

Every conference key is fresh since each conference member selects a new random k_i for the generation of each conference key. In our protocol, we use the GCRT to achieve the threshold access structure instead of Shamir's secret sharing. As we shall see, our protocol is somewhat sensitive to selected parameters [17, 19]. However, it is information theoretically secure if the parameters for GCRT are selected appropriately.

As discussed in Sections 4.2 and 4.3, each conference member U_i encrypts the sub-key information using the other conference members' public keys e_j for $j = 1, 2, \cdots, n, i \neq j$. So, unauthorized members cannot compute the corresponding sub-key information, so they cannot obtain the conference key.

5.1.3 Forward/Backward Secrecy

First, we consider forward secrecy, which states that 1) if an attacker obtains the previous conference keys, he or she cannot derive the current conference key, and 2) if a member leaves the conference group, he or she also cannot compute the subsequent conference keys. During the computation of the conference key K, an additional modulus k is required. The parameter k is computed as follows:
$k = \left\lfloor \frac{\sum_{i=1}^{t} k_i}{t} \right\rfloor$, where k_i is conference member U_i's private input.

For each conference key, conference member U_i must randomly select a number k_i that satisfies $\mathrm{Max}\{x_i\}_{1 \leq i \leq n} < k_i < \mathrm{Min}\{m_j\}_{1 \leq j \leq n}$. And, each k_i is independent. Therefore, the previous conference keys and the current conference key are irrelevant. Even though an adversary may know a series of previous conference keys, he or she cannot compute the current conference key.

Then, we consider backward secrecy, meaning that if an attacker obtains the current conference key, he or she cannot derive the previous conference keys, and, if a member joins the conference group, he or she also cannot derive any previous conference keys. As mentioned in the previous paragraph concerning the generation of each conference key, conference member U_i randomly re-selects a new k_i. Then, all conference keys K are independent of each other. So, an adversary who knows the current conference key cannot derive the previous conference keys.

5.1.4 Resistance to Some Types of Attacks

In this section, we show that our protocol is secure against the following attacks:

1) Impersonation attack

The main active attack to a conference-key agreement protocol is the impersonation attack, in which an unauthorized adversary impersonates a legal conference member to take part in a conference. In the registration phase, KGC generates the corresponding private key d_i and public key e_i published to all conference members for each conference member U_i. And, in the sub-key distribution and commitment phase, U_i must use her or his private key d_i to encrypt the sub-key information. Therefore, other members can utilize the corresponding public key e_i to check whether the sub-key information is from the alleged member.

2) Forgery attack

We divide forgery attacks into two categories. First, a malicious member can not forge other members' sub-key information since he or she does not have the other members' private keys. Meanwhile, conference members also can verify whether the sub-key information received has been tampered with by checking the equation $h'_i = h_i$, where $h'_i = h(x'_i||m'_i||k'_i||SN'_\ell)$ and $h_i = h(x_i||m_i||k_i||SN_\ell)$. On the other hand, as mentioned in Section 2.1, it takes a sufficient number of conference members to cooperate to generate a valid conference key. We assume that a certain number of compromised members want to forge the conference key. According to threshold secret sharing, if the number of compromised members involved is less than the threshold, the members cannot generate a valid conference key. And, the other authorized members can use their secret value pair (x_i, m_i) to verify the validity of the conference key. Therefore, the conference key cannot be forged easily unless the number of compromised members exceeds the threshold.

3) Replay attack

A group of conference members run a threshold conference-key protocol over an open network. Even though their communications are encrypted with the corresponding public key cryptosystem, an inside attacker or an outside attacker also can intercept these data and retransmit them. But, in the proposed protocol, the initiator first broadcasts a unique serial number SN_ℓ, and the other conference members involved embed into the sub-key information. The conference members that receive the sub-key information $\omega_i = \{h_i, \sigma_{ij}, \mathrm{for} j = 1, 2, ..., n, j \neq i\}$ can check to determine whether the serial number SN_ℓ is the current serial number. So, the adversary cannot resend the previous key-agreement message.

5.2 Performance Analysis

In this section, we analyzed the computation costs for the proposed protocol. In our protocol, if the number of conference members involved in the generation of a

Table 2: Comparison of computational costs

	Tzeng's protocol	**Hung et al's protocol**	**Zhao et al's protocol**	**Our protocol**
Registration	nT_e	nT_e	0	0
Sub-key distribution and commitment phase	$(n+2)\,T_e + T_{inv} + 3T_{mul} + T_h$	$(n+2)\,T_e + T_h + (n+2)\,T_{mul}$	$nT_e + T_h$	$nT_e + T_h$
Sub-key recovery phase	$4nT_e + nT_{mul}$	$4nT_e + nT_{mul} + nT_h$	$2\,(n-1)\,T_e + (n-1)\,T_h$	$2\,(t-1)\,T_e + (t-1)\,T_h$
Conference-key derivation and verification phase	0	0	0	$tT_{inv} + 2\,(t-1)\,T_{mul}$

conference key exceeds a certain number, a common session key can be achieved. And, all authorized conference member can obtain the common session key and can verify its validity. We assume that the threshold is t. During the sub-key distribution and commitment phase, each member U_i must utilize public-key cryptosystems to confirm sub-key's authentication, confidentiality, and integrity. For facilitating the performance evaluation of our scheme, we first denote the following notations:

1) T_h:the time for performing a hash function h;

2) T_{inv}:the time for performing a modular inverse computation;

3) T_{mul}:the time for performing a modular multiplication computation;

4) T_e:the time for performing a modular exponentiation computation.

Table 2 compares the computational costs of our protocol with those of Tzeng [20], Hung et al. [10], and Zhao et al. [26].

As can be seen in Table 2, since we utilize the RSA cryptosystem applied in Zhao et al.'s protocol to confirm the authentication, confidentiality, and integrity of the sub-key, the computational costs of our protocol is the same as Zhao et al.'s protocol in the sub-key distribution and commitment phase. In the sub-key recovery phase, our protocol requires fewer modular exponentiations than Zhao et al.'s protocol since our protocol is a threshold scheme. However, in the conference-key derivation and verification phase, our protocol requires additional modular inverse computation and modular multiplication computation to reconstruct the conference key. The result is that the total computational costs of our protocol are less than those associated with Tzeng's protocol and Huang et al.'s protocol, and they are about the same as those associated with Zhao et al.'s protocol. In addition, we need to point out that the conference-key can be recovered by collecting t value pairs and that the conference-key recovery by the usual Lagrange interpolation method requires $O(t\log^2 t)$ operations, while the GCRT method of the proposed protocol requires only $O(t)$ operations.

6 Conclusions

In this paper, we propose a novel, threshold conference-key agreement protocol based on the generalized Chinese remainder theorem. In our protocol, the conference key can be generated only if the number of conference members involved exceeds a certain threshold, and other authorized conference members, who were not involved in the generation of the conference key, also can compute the conference key and check its validity. The unique threshold characteristic of our conference-key agreement protocol can fill the gaps in many applications.

The security analysis shows that our protocol is resistant to impersonation attack, forgery attack, and replay attack and that it also has some important security features, such as conference-key authentication, conference-key confidentiality, and forward/backward secrecy. The performance analysis shows that the computational cost of our protocol is satisfactory.

Acknowledgments

This paper is supported by the National Science Foundation of China under grant No. 61272173, 61100194, 61401060 and The general program of Liaoning Provincial Department of Education Science Research under grants L2014017.

References

[1] C. Asmuth and J. Bloom, "A modular approach to key safeguarding," *IEEE Transactions on Information Theory, IT-29 (2)*, pp. 208–210, 1983.

[2] G. Blakley, "Safeguarding cryptographic keys," in *The National Computer Conference*, pp. 313–317, Montvale:NCC, 1979.

[3] M. Burmester and Y. Desmedt, "A secure and efficient conference key distribution system," in *EUROCRYPT 1994, LNCS 0950*, pp. 275– 286, Springer, Berlin, 1995.
[4] C.C. Chang and H.C. Lee, "A new generalized group-oriented cryptoscheme without trusted centers," *IEEE Journal on Selected Areas in Communications 11 (5)*, pp. 725–729, 1993.
[5] C.C. Chang, T.C. Wu, and C.P. Chen, "The design of a conference key distribution system," in *AUSCRYPT 1992, LNCS 0718*, pp. 459–466, Springer, Berlin, 1993.
[6] V. Daza, J. Herranz, and G. Sáez, "On the computational security of a distributed key distribution scheme," *IEEE Transactions on Computers 57 (8)*, pp. 1087–1097, 2008.
[7] W. Diffie and M.E. Hellman, "New directions in cryptography," *IEEE Transactions on Information Theory IT-22 (6)*, pp. 644–654, 1976.
[8] L. Harn and C.L. Lin, "Authenticated group key transfer protocol based on secret sharing," *IEEE Transactions on Computers 59 (6)*, pp. 842–846, 2010.
[9] I. Ingemarsson, D.T. Tang, and C.K. Wong, "A conference key distribution system," *A conference key distribution system, IEEE Transactions on Information Theory IT-28 (5)*, pp. 714–720, 1982.
[10] Q. Jiang, J.F. Ma, G.S. Li, and et al., "Robust two-factor authentication and key agreement preserving user privacy," *International Journal of Network Security*, vol. 16, no. 3, pp. 229–240, 2014.
[11] M. Just and S. Vaudenay, "Authenticated multi-party key agreement," in *ASIACRYPT 1996, LNCS 1163*, pp. 36–49, Springer, Berlin, 1996.
[12] Y. Kim, A. Perrig, and G. Tsudik, "Group key agreement efficient in communications," *IEEE Transactions on Computers 53 (7)*, pp. 905–921, 2004.
[13] K. Koyama, "Secure conference key distribution schemes for conspiracy attack," in *EUROCRYPT 1992, LNCS 0658*, pp. 449–453, Springer, Berlin, 1993.
[14] S.S. Kulkarni and B. Bruhadeshwar, "Key-update distribution in secure group communication," *Computer Communications 33 (6)*, pp. 689–705, 2010.
[15] Y.P. Lai and C.C. Chang, "Parallel computational algorithms for generalized chinese remainder theorem," *Computers and Electrical Engineering 29 (8)*, pp. 801– 811, 2003.
[16] P.P.C. Lee, J.C.S. Lui, and D.K.Y. Yau, "Distributed collaborative key agreement and authentication protocols for dynamic peer groups," *IEEE/ACM Transactions on Networking 14 (2)*, pp. 263–276, 2006.
[17] Quisquater M, Preneel B, and Vandewalle J, "On the security of the threshold scheme based on the chinese remainder theorem," in *Proc. PKC 2002, LNCS 2274*, pp. 199–210, 2002.
[18] M. Mignotte, "How to share a secret," in *EUROCRYPT 1982, LNCS 149*, pp. 371–375, Springer, Berlin, 1982.
[19] Goldreich O, Ron D, and Sudan M, "Chinese remaindering with errors," *IEEE Transactions on Information Theory IT-46*, pp. 1330–1338, 2000.
[20] Y.K. Peker, "A new key agreement scheme based on the triple decomposition problem," *International Journal of Network Security*, vol. 16, no. 6, pp. 426–436, 2014.
[21] A. Perrig, D. Song, J.D. Tygar, and Elk, "A new protocol for efficient large group key distribution," in *IEEE Symposium Security and Privacy*, pp. 247–262, 2001.
[22] R.L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Communications of the ACM 21 (2)*, pp. 120– 126, 1978.
[23] A. Shamir, "How to share a secret," *Communications of the ACM 22 (11)*, pp. 612–613, 1979.
[24] M. Steiner, G. Tsudik, and M. Waidner, "Diffie-hellman key distribution extended to group communication," in *Third ACM Conference Computer and Communications Security*, pp. 31–37, New Delhi, India, 1996.
[25] M. Steiner, G. Tsudik, and M. Waidner, "Key agreement in dynamic peer groups," *IEEE Transactions on Parallel and Distributed Systems 11 (8)*, pp. 769–780, 2000.
[26] J.J. Zhao, D.W. Gu, and Y.L. Li, "An efficient fault-tolerant group key agreement protocol," *Computer Communications 33 (7)*, pp. 890–895, 2010.
[27] B.Q. Zhou, S.J. Li, J.X. Wang, and et al., "A pairwise key establishment scheme for multiple deployment sensor networks," *International Journal of Network Security*, vol. 16, no. 3, pp. 221–228, 2014.

Cheng Guo received the B.S. degree in computer science from Xian University of Architecture and Technology in 2002. He received the M.S. degree in 2006 and his Ph.D in computer application and technology, in 2009, both from the Dalian University of Technology, Dalian, China. From July 2010 to July 2012, he was a post doc in the Department of Computer Science at the National TsingHua University, Hsinchu, Taiwan. Since 2013, he has been an associate professor in the School of Software at the Dalian University of Technology. His current research interests include information security and cryptology

Chin-Chen Chang received his B.S. degree in applied mathematics in 1977 and the M.S. degree in computer and decision sciences in 1979, both from the National Tsing Hua University, Hsinchu, Taiwan. He received his Ph.D in computer engineering in 1982 from the National Chiao Tung University, Hsinchu, Taiwan. From 1983-1989, he was on the faculty of the Institute of Applied Mathematics, National Chung Hsing University, Taichung, Taiwan. From August 1989 to July 1992, he was the head of, and a professor in, the Institute of Computer Science and Information Engineering at the National Chung Cheng University, Chiayi, Taiwan.

From August 1992 to July 1995, he was the dean of the college of Engineering at the same university. From August 1995 to October 1997, he was the provost at the National Chung Cheng University. From September 1996 to October 1997, Dr. Chang was the Acting President at the National Chung Cheng University. From July 1998 to June 2000, he was the director of Advisory Office of the Ministry of Education of the R.O.C. From 2002 to 2005, he was a Chair Professor of National Chung Cheng University. Since February 2005, he has been a Chair Professor of Feng Chia University. In addition, he has served as a consultant to several research institutes and government departments. His current research interests include database design, computer cryptography, image compression and data structures. He is a fellow of the IEEE.

A Specification-based Intrusion Detection Framework for Cyber-physical Environment in Electric Power System

Shengyi Pan, Thomas Morris, and Uttam Adhikari
(Corresponding author: Shengyi Pan)

Department of Electrical and Computer Engineering, Mississippi State University
Box 9571, 216 Simrall Hall Mississippi State, MS 39762, USA
(Email: tommymorris@ieee.org)
(Received June 13, 2013; revised and accepted Nov. 2, 2014)

Abstract

The emergence of high-speed networks in electric power systems creates a tight interaction of cyber infrastructure with the physical infrastructure and makes the power system susceptible to cyber penetration and attacks. To address this problem, this paper proposes an innovative approach to develop a specification-based intrusion detection framework that leverages available information provided by components in a contemporary power system. A Bayesian network is used to graphically encode the causal relations among the available information to create patterns with temporal state transitions, which are used as rules in the proposed intrusion detection framework. This allows the proposed framework to detect cyber attacks and classify different substation scenarios. A case study is provided for the non-pilot directional over current relay protection scheme for a modified 2-bus 2-generator system taken from a section of the IEEE 9-bus 3-generator system. Nine power system scenarios were developed and implemented as part of the case study. Each scenario was implemented on a test bed and all scenarios were correctly classified by the IDS built using the proposed methodology.

Keywords: Bayesian network, cyber-physical, electric power system, intrusion detection system, relay protection scheme, synchrophasor

1 Introduction

The next generation electric power grid will rely on many advanced technologies such as synchrophasor systems, industrial automation control systems and advance metering infrastructure in order to meet the increasing demand on reliable energy. Due to the critical role that the electric power system plays in our society, there is a common agreement among different organizations that the electric power grid needs to be better secured to ensure continually available power being provided to the nation [35]. The North American Electric Reliability Corporation Critical Infrastructure Protection (NERC-CIP) program defines critical infrastructure and provides recommendations regarding to cyber security for electric utilities to better protect their critical infrastructures [38]. The National Institute of Standards and Technology Interagency Report (NISTIR) 7628 also documents the guidelines and requirements for industry to better secure their facilities [15]. However, the United States Government Accountability Office (GAO) has realized that current guidelines from these organizations are not sufficient to securely implement the modern electricity grid and it calls for the retrofit research and development to current security mechanisms [12].

1.1 Background and Problem Statements

The cyber-physical security issues of electric power systems have been discussed for a long time. In the past the electric power system was often isolated and used proprietary devices and software. The contemporary and future power grid uses advanced technologies which rely on the commercial off-the-shelf (COTS) components e.g. Personal Computers (PCs), Windows Operating System, and standardized communications such as IEC 61850 and IEEE C37.118. Many COTS components were designed for consumer or enterprise use and not for use in critical infrastructures such as the electric power system [10]. Power system cyber components have certain security features (password authentication) built in, however, penetration tests conducted in [30, 33] have shown that cyber attacks targeted towards substation computers and devices can interrupt the electric power system communications, prevent real time monitoring of the power system, induce physical side effects. Hence there is a need to develop security countermeasures that can be deployed to

protect the critical infrastructures in the electric power system. However, a barrier for developing cyber security countermeasures is a lack of algorithms based on the unique characteristics of electric power system where high interactions between the physical process and the cyber infrastructures are present [44]. The cyber infrastructures provide the communication media that is used between the physical automation control system and other systems such as enterprise software where control algorithms or system analysis algorithms are implemented. Since most of the cyber infrastructures use open standards without security features, once they are compromised, the attackers are able to launch attacks targeting the physical process by modifying the control algorithm. An example is the resonance attacks where an attacker who compromised the sensors or controllers causes the physical system to oscillate at its resonant frequency [7]. Another example is demonstrated in [27] where an attack can inject false data to compromise meters to bypass the existing bad data detection algorithms. An example of attacks from physical devices in real world is the Stuxnet in July 2010 where the malware targeting control system physical devices spread by USB drives [11]. Some recent works investigating the security issues in the modern electric power grid suggest that new security mechanisms should focus on the unique characteristics of the power grid to achieve comprehensive protection [32, 47].

Since attacks are always unpredictable and cannot be eliminated, it is necessary to deploy an intrusion detection system to alert operators or automated response algorithms when an attack is discovered. Traditional intrusion detection systems that only examine network traffic cannot provide enough detection abilities to the cyber-physical system where the physical process is also of concern [7]. In this paper, a new methodology that extends the ability of traditional anomaly-based intrusion detection system is proposed to design an intrusion detection system suitable for cyber-physical system by taking the system's physical process into account. The proposed intrusion detection system is able to provide a "defense-in-depth" protection by considering the following two concerns [6]: The consequences of the attacks on the cyber-physical system should be understood when planning the protection; and novel attack-detection algorithms should be developed based on how the physical process should behave so that intrusion detection systems can identify whether the control command or sensor data has been altered.

In addition, another concern from utility operators is that cyber security solutions should make minimum modification to the current facilities in the grid. This is because any changes to be made require strict recertification and evaluation to be adapted to the grid, which can be quite costly [17]. Therefore, the security solution proposed here is designed to be built upon current resources in the grid minimizing changes to its components.

1.2 Contributions of This Paper

This paper addresses the security challenges of the modern cyber-physical electric power system by proposing an intrusion detection framework that covers the aforementioned three concerns. This framework provides a specification-based intrusion detection system that complements current anomaly intrusion detection systems by leveraging time-synchronized data from synchrophasor devices as well as observable events from audit logs of network devices including protection relays, network monitoring software, and control room computers. Depending on different control schemes, this information underlines causal relations between the system behaviors in the cyber-physical system. One of the contributions of this work is to provide a methodology to map such information to the probabilistic network - Bayesian network to derive the rules for the IDS. The Bayesian network is recognized for its powerful intuitive method of modeling interdependencies between variables and its ability to graphically represent causal relations from data and workflow logs. Based on a specific control scheme, namely the over current relay protection scheme, this paper demonstrates the procedure to construct such a Bayesian network and derive the temporal-state transition patterns for different system scenarios. These patterns are used in the IDS as rules for classifying legible system scenarios and detecting intrusions that aim to interrupt the protection scheme. A model to implement the proposed IDS is also provided based on a specific power system transmission system test bed. The IDS monitors the status of one relay and the transmission line where the relay is located to provide an extension to power system situation awareness such that the operator can be informed of whether disturbances in the power grid (e.g. faults in transmission line or relay operations) are caused by system faults or cyber attacks.

1.3 Paper Organization

The rest of the paper is organized as follows. First, related works are discussed in Section 2. Section 3 provides an overview of a reference electric transmission system, the non-pilot over current protection scheme, and a hardware-in-the-loop test bed implementation of the transmission system and protection scheme. This system is used later in the paper for case study to demonstrate the effectiveness of the proposed IDS methodology. Section 3 also provides a threat model which describes 9 power system disturbances and cyber attacks which threaten the reference transmission system and protection scheme. Section 4 provides a mathematical description of the Bayesian network and discusses a procedure for creating a Bayesian network for a cyber physical system. Section 5 provides results an analysis of the IDS built for the case study. Conclusions and future work are discussed in Section VII.

2 Related Works

2.1 Wide Area Monitoring Systems

The need of electric market regulation and the connection of neighboring grids motivate the Wide Area Monitoring System (WAMS) where multiple organizations cooperate to allow real-time monitoring of the electric power system. The WAMS is a measurement system that uses information communication technology (ICT) to transmit digital and/or analogue information. The WAMS is now adopting time-synchronized data that provides microsecond time accuracy [29]. The time-stamping data in WAMS includes not only the measurements such as phasors of voltage, current (i.e. synchrophasor system) but also the status of some IEDs such as relays, breakers etc. [2]. Such accurate redundant information nowadays can be collected from PMUs, smart meters and protection relays etc. The redundant information contributed by the time-synchronized data provides benefit for reliability, efficiency, and economics in power system monitoring and control. The extreme low latency brought by time-synchronized data allows various real-time wide area control algorithms and special protection schemes to be developed to increase power grid reliability and stability [2, 18, 21, 28, 34]. This paper takes advantage of this fast and accurate information provided by synchrophasor system to build a novel intrusion detection system for the electric power system.

2.2 Specification-based Intrusion Detection System

The idea of intrusion detection systems was originally introduced to the IT system to detecting activities that violate security policy [37]. There are two types of intrusion detection systems (IDS): Misuse-based Intrusion Detection Systems and Anomaly-based Intrusion Detection Systems. Misuse-based IDS and signature-based IDS look for well-defined patterns of known attacks or vulnerabilities, and, therefore, suffer from the fact that any undefined attacks will be ignored [49]. Anomaly-based IDS consider the normal behaviors of a system [13, 54]. Any derivation from the normal system behaviors will be defined as an intrusion. The anomaly-based IDS is widely used for its ability to detect zero-day intrusions however, it has high false positive rate where some normal behaviors of the system that do not match the defined normal behaviors will be mistaken as intrusions. The specification-based IDS was introduced by Ko in 1996 [24] as a complement to the anomaly-based IDS to improve its accuracy. Specification-based IDS monitor the system according to policies specified by valid sequences of system behaviors. Any sequence of behaviors outside the predefined specifications will be regarded as a violation. Various methodologies have been applied by scientists to specify such behavior/event sequences, for example, the parallel environment grammar [24], regular expressions for events [50], or abstract state machine language [43]. The specification-based intrusion detection has also been widely applied in software engineering. Most recently specification-based intrusion detection is also used in the area of network protocol of critical infrastructures e.g. ANSI C12 protocol for advanced metering infrastructure [3], DNP3 protocol [26], IEEE C37.118 protocol [46], Modbus protocol [9]. The specification-based IDS is popular in this area because network protocols usually have standard message formats from which the specifications of the IDS can be derived. The applications of specification-based IDS also extend to more complex systems such as networked SCADA systems [5], medical cyber-physical systems [31] and real-time embedded systems [55] where intrusion detection rules are defined from system behaviors. The system behaviors in these works are represented by a sequence of system states. By keeping tracking the system state the intrusion detection techniques of these works discover the malicious activities that drive the system state from safe to unsafe. This paper puts emphasis on cyber-physical electric power grid. In addition to define a finite state machine for the electric power system, this paper uses a probabilistic network to extract knowledge about the specifications of different system behaviors from the causal relationship underlined by the transitions of these system states. Such knowledge is used to derive the rules used by the proposed IDS to classify system behaviors.

2.3 Why Bayesian Network?

Probabilistic networks provide clear semantics to allow them to be processed for extracting knowledge of a certain domain. They are able to represent the dependencies or interdependencies between variables; therefore they can be used for diagnosis, learning, explanation, and many other inference related tasks necessary for intelligent systems [4]. Among the probabilistic networks, the Bayesian network is prevalent for its explicit graphical representation of cause-effect reasoning with uncertainty. Depending on different interpretations, it can also represent causality. One of the applications of Bayesian networks is in network vulnerability assessment where "attack graph" is developed using a Bayesian network [52]. In an attack graph, two directly connected nodes represent the causal relation in which the compromise of one node will lead to the compromise of the other node. In addition, Bayesian networks have also been applied to interdependency modeling and analysis for critical infrastructure [14, 16] and for fault diagnosis in power systems [36, 53]. In the area of health care, Bayesian networks are also used to discover patterns for Hemodialysis to help medical professionals react to exceptions [25]. Tutorials about Bayesian networks can be found in [8, 23, 40]. In the cyber-physical environment, a sequence of system actions and events raised by a specific system scenario (e.g. a short circuit fault in a transmission line) imply the causal relations between system states. The causal relations are represented as tempo-

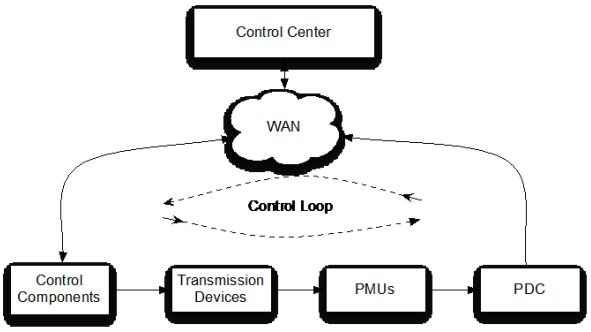


Figure 1: Structure of electric transmission system with integration of synchrophasor Technology

ral transitions between different system states. Therefore, the system states are nodes in the Bayesian network.

3 System Model and Threads

In this section, a brief introduction of the electric transmission system integrated with synchrophasor technology will be given. The potential attacks and their consequences to the transmission line protection scheme will be discussed on the basis of a 2-generator 3-bus electric transmission system. Nevertheless, all discussions are based on the assumption that the attacks are launched after the attackers compromise the substation communication network or physically penetrate the substation such as through an insider attack by a disgruntled employee authorized to access power grid facilities and or its communication network.

3.1 Cyber-Physical Environment of Synchrophasor-based Electric Transmission System

A typical power system is divided into four functional parts: generation, transmission, distribution and consumers. The electric transmission system is the backbone of the power system transmitting the electric power from generators to the load centers over a long distance. The structure of a cyber-physical environment for the electric transmission system augmented with synchrophasor technology is shown in Figure 1. The transmission system devices are mainly composed of transmission lines, breakers, and transformers that are monitored by field sensors. In the case of a synchrophasor system these field sensors are Phasor Measurement Units (PMUs). The PMUs attached to transmission lines provide synchronized data that is time-stamped to the Coordinated Universal Time (UTC) via Global Positioning System (GPS) signals for continuous real-time monitoring. Phasor Data Concentrators (PDCs) collect synchrophasor measurements from PMUs that are located in different locations and send the measurements to the control center through the wide-area network (WAN). PMUs in different locations and PDCs are key components in the synchrophasor based wide area monitoring system (WAMS). Compared to the traditional supervisory control and data acquisition (SCADA) system where the field sensors measure the system once per several seconds, the emergence of WAMS leveraging synchrophasor technology allows much faster measuring for the transmission system at the rates ranging from 30 samples per second to 120 samples per second. The control center that utilizes the high resolution measurement data aggregated by PDCs is able to evaluate the system status and perform advanced algorithms to make different real-time control decisions to control components in the field. The information flow described above is shown as the dotted line in Figure 1 and is often recognized as a control loop. All devices in this system are synchronized to UTC time via GPS signals. However, in the case of a distributed control, the protection components in the system sense the disturbance and react to it by themselves. The bi-directed-arrow lines in between control components and WAN indicates not only the command data sent from control center but also the time-synchronized audit information reported from intelligent electronic devices (IDEs) to the control center.

The system can be considered as a finite state machine. If, for example, a tripping operation is sent from control center, this will cause system state transitions because a signal-sending operation has been recorded in the control panel which is one component of the system. In general, the changes of the behaviors in different system components such as a breaker, relay, and transmission lines in a given period of time or at a definite point of time will cause the system state to transit from certain state to another. Theses changes are reflected by the transmission line sensor readings or device logs. If the system state is represented as a set of observations (from logs of different components) and measurement data (from measurement devices) inside the system, such changes along with time can be regarded as temporal state transitions.

3.2 Reference Electric Transmission System and Non-pilot Over current Protection

Non-pilot transmission system protection is chosen as a demonstration vehicle for the proposed IDS because this type of protection is fundamental to all other electrical equipment [19]. The non-pilot directional over current relay protection scheme is examined in the context of a multiple source circuit shown in Figure 2 to provide protection against short circuit faults which are the primary disturbances found in transmission lines. This paper focuses on multiple source circuits. Although there may be other types of circuits such as loop circuit and radial circuit, multiple source circuits make up the majority of the electric transmission system. The IDS proposed here is suitable only to the multiple sources circuits where our assumption that when one line is taken off the load can still get supplied from generators via other route can be

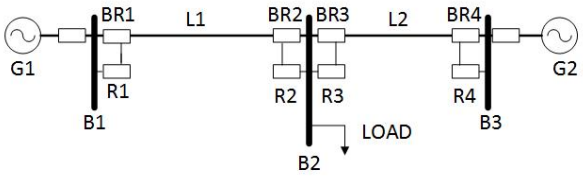


Figure 2: Single line diagram for over current protection scheme for a transmission system

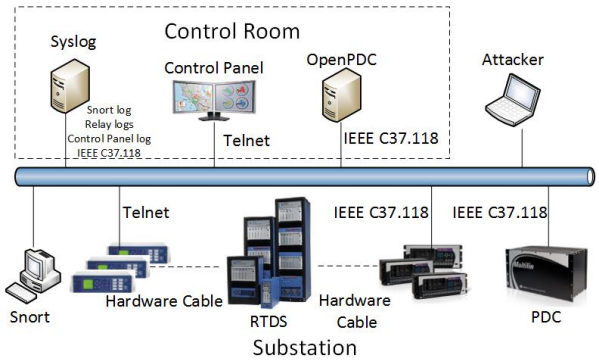


Figure 3: Test bed topology

true.

In a multiple sources transmission system, the relays are directional to provide relay coordination between all of the relays that can see a given fault. In this paper, instantaneous and time-delay over current relays are applied to the transmission system to provide a directional over current protection scheme for a maximum of 2 zones. Readers can refer to [19] for a complete introduction of non-pilot over current protection of transmission lines.

The single-line electric transmission diagram depicted in Figure 2 is a modified 2-bus 2-generator system taken from a section of the IEEE 9-bus 3-generator system [45]. In the non-pilot system, relays decisions solely depend on the measurement of electrical quantities at the near end of the protected line section. Generators G1 and G2 supply power to load L at bus B2 via the transmission line L1 and L2. Line L1 and L2 are symmetric. The load can be changed from 200 MW to 240 MW in this study. Four relays R1, R2, R3 and R4 with integrated PMU functionality reside at each end of each transmission line that control the breakers BR1, BR2, BR3 and BR4. Take line L1 for example, the relays R1 and R2 provide instantaneous over current protection for transmission line L1 while R1 also provides time-delayed over current protection for transmission line L2 in the case that the relay R3 fails for faults in L2. It is the similar case to the relays in line L2 where R3 and R4 take care of the faults in L2 and R4 provides backup protection for faults in L1 in case R2 failed tripping for the fault. For the four relays, there are two settings that should be properly configured to achieve the over current protection and relay coordination: the instantaneous over current pickup and the time-delayed over current pickup (abbreviated as IOC and TOC). The IOC specifies the threshold so that the relay will operate for all short circuits which cause the current magnitude to exceed the threshold in the line it is to provide protection for. As for the two-zone over current protection a TOC specifies the threshold according to which the relay provides backup protection for an adjacent line. In this case, the relay wait for a period of time called "delay time" if the current magnitude is in the "warning level" (magnitude in between TOC and IOC) in the local line, which indicates a short circuit fault in the adjacent line. Theoretically relay trips simultaneously when PMU displays overcurrent for instantaneous over current protection, but the simulation in this paper will insert one cycle delay between them, which is more likely in the real situation.

The four PMUs constantly monitor the power flow in the two transmission lines at the locations where the relays sit. They transmit time-stamped synchrophasor data in the IEEE C37.118 data frames [48] to a PDC in the substation via Ethernet in a frequency at 120 samples/second. The data frames including information of angles and magnitudes of line current and voltage, namely phasors, will be finally stored in a historian located in the control room. Usually, an Energy Management System (EMS) with an integrated software PDC (e.g. OpenPDC [20]) is installed in the control room for applications such as system visualizations, system situation awareness etc. [56]. The software PDC is a higher level PDC that collects synchrophasor data frames from physical PDC devices in different locations and processes them. The operator who monitors the system status through a software PDC will be kept informed to react to a contingency in the monitored system.

3.3 Test Bed Configurations

A test bed was implemented to simulate the electric transmission system shown in Figure 2. The test bed includes the real-time power system simulator (RTDS) with a hardware-in-the-loop design using commercial PMUs, PDCs and protective relays [1]. The topology of test bed is depicted in Figure 3.

The test bed is separated by the Ethernet bar into two parts. In the substation site, the RTDS is used to model the 2-bus 2-generator transmission system discussed shown in Figure 2. There are four commercial digital relays with integrated PMU functionality connected to breakers simulated in the RTDS. The RTDS provides simulated high AC voltages at buses and line currents through transmission lines for the PMU and relays to measure. The relays and PMUs are drawn as two separate components in Figure 3. In some cases the PMU and relay are integrated in the same chassis. In other cases the PMU and relay are separate devices. The RTDS simulates single line to ground and the phase to ground short circuit faults on transmission lines L1 and L2. The hardware-in-the-loop design allows the relays to react to the fault

to open the breaker. The four PMUs reside at the two ends of line L1 and line L2 constantly measure current and voltage phasors. There is one hardware PDC in the substation that aggregates IEEE C37.118 synchrophasor measurements from the four PMUs through the substation network switch and forwards the concentrated synchrophasor measurement data frames to the control room where OpenPDC is installed. OpenPDC displays and stores the synchrophasor data and serves as a data historian for the system. The substation is also equipped with Snort to monitor the substation network traffic. In our experiment Snort [51] is configured to capture Telnet frames that contains tripping commands to the relay. The content of the tripping commands varies by the relay brand and model. However, whenever the Snort rule captures remote tripping commands, it logs them with their source IP address and destination IP address (relay IP address). The content in the Snort log file is forwarded to a syslog server in the control room. In the control room, the control panel is a Microsoft Windows PC with vendor-provided configuration clients for the relays and PMUs. Relay and PMU configuration can be performed either remotely over the communications network or from the devices' faceplate.

Relays in the test bed are specifically configured to implement the directional over current protection scheme for the 2-bus 2-generator transmission system. Two relevant parameters in the relay configurations are of concern to the IDS: IOC, TOC. The PMUs and PDC are configured to stream in the data rate of 120 samples per second. Each relay is operated independently according to configurations and opens its corresponding breaker in the transmission lines. Breaker failure is also simulated in this work, in which the relay trips but the corresponding break does not open. There are four situations that relay will be tripped: (1) Relay detects fault occurring in the direction it faces; (2) Attackers send tripping command to relay via Telnet from their PCs; (3) Attackers trip the relay via its faceplate; (4) The operator sends tripping command to relay via Telnet from control panel. The test bed also provides time-synchronized audit data from logs of multiple components. In addition to Snort log which is used to record time stamped remote trip command, PMU measurements are used as a redundant source of line current measurements. Relay logs are used to record time stamped relay operations. The control panel log records time stamped commands to the energy management system to trip a relay. This information is aggregated in the syslog server and used to track series of causal events related to different system scenarios.

3.4 Scenarios for Over Current Protection Scheme in Line L1

This threat model consists of a set of scenarios which represent normal power system disturbances, normal supervisory control actions, and cyber attacks against the non-pilot directional over current protection scheme. Each scenario has been implemented using the test bed described in the previous section. A comprehensive understanding of these scenarios is important for the domain expert to identify which information is required from which resources to construct the IDS.

For this work it is assumed attackers attack only one relay or PMU at a time. There are two legitimate scenarios for over current protection scheme for line L1. Scenario Q1 is an over current fault on line 1 which causes R1 and R2 to instantaneously trip. Scenario Q2 is the removal of a line L1 for maintenance. In scenario Q2 an operator initiates a remote trip of R1 and R2 via the energy management system. The operator initiated remote trip commands will be recorded in the control panel log with the timestamps showing when the commands were sent. The remote trip events are also logged by the network traffic monitor, Snort. The trip commands cause the relays to open the breakers isolating line L1 from the power system. The operator observes the success of breaker-opening action by observing zero current in L1 from synchrophasor data frames collected by OpenPDC.

The attacks considered for this work focus on changing the control logic. To achieve this goal, attacks attempt to interrupt the over current protection scheme on line L1 by causing relays to not trip during a valid fault or by causing relays to trip when there is no fault. The operator in the control center may be aware of the implications of the aforementioned attacks via the collected synchrophasor data however he/she will still lack information on the primary cause of these failures. In this paper, we will analyze such failures from security perspective.

Scenario Q3 is a cyber command injection attack which mimics scenario Q2. For scenario Q3 the remote trip command does not originate from the control panel. Instead the remote trip command originates from another node (i.e. attacker's PC) on the communications network. As such, the control panel log will not include a trip command entry, but, the network traffic monitor will detect the remote trip command. Scenario Q4 is a physical attack in which an attacker trips the relay from the faceplate. In Scenario Q5 a valid fault occurs, but, the relay does not trip. This may occur due to hard ware error or incorrect relay setting. Scenarios Q6, Q7, and Q8 are data injection attacks which provide false current measurements to the operator by modifying synchrophasor measurements. In scenario Q6 current magnitudes above the relay pickup are injected to the PDC and transmitted to the control room. This scenario attempts to cause the operator to believe a set of relays has failed to operate and therefore cause the operator to manually trip the relay. In scenario Q7 the injected current magnitudes are less than the relay pickup. This scenario may be used to mask a fault which has been handled by relay which has been tampered with. In scenario Q8 the injected current magnitudes are 0 amperes. This scenario attempts to mimic the situation immediately after a properly handled fault. Scenario Q9 is for scenarios which target other relays. A separate instance of the IDS monitors each relay. Scenario

Q9 simulates breaker failure where the relay tripped but the current in the line remains high due to the breaker failure. According to the assumption that only one malicious activity takes place at a time, the breaker failure is simulated only when there is a fault in the line. Scenario Q10 is for scenarios which target other relays. A separate instance of the IDS monitors each relay. Scenarios Q1-Q9 are used to classify local events and scenario Q10 classifies events at other relays.

4 Constructing Bayesian Network for System Scenarios

In this section a set of terms are introduced. These terms are defined to mathematically describe Bayesian networks used to model the 10 scenarios from the threat model. Next the procedure to construct the Bayesian network with temporal state transitions for relay R1 is demonstrated. Patterns for the 10 scenarios are derived from the Bayesian network. These patterns are then used as detection rules for the IDS which monitors R2.

4.1 Definitions

A state represents system status at a point in time. A state is defined as a set of variable, each of which is a measurement. A state is denoted as $S = \{v_1, \cdots, v_n\}$. Each state variable in state S i.e. $v_1 \in S$ may have a unique range called its own domain donated as D_j. The number of possible distinct values, $\|D_j\|$, varies by components. Domains should be quantized to finite ranges to avoid infinite state space. D_j is hence denoted as a set of distinct values $\{d_j1, \cdots, d_{j\|D_j\|}\}$ with each of its elements specifying valid event or measurement values observed from a component .

An *observation* is a proposition in the form of $v_ij = d_jk$ which means the variable v_ij attains the *k-th* value d_jk in its domain D_j. Based on these concepts a system *state* can be uniquely specified by assigning different combinations of observations to all variables and write the state as $S := (v_{1i} = d_1) \vee \cdots \vee (v_{ni} = d_n)$. There are, hence, $\|D_s\| = \prod_{i=1}^{n} \|D_i\| = 0$ possible states.

An *action* is single system behavior, the occurrence of which triggers the system state to change from one state to another. Such behavior could be system inherent behavior, operator actions, attacks, or a clock timeout. An action is denoted as $A_l : S_i \rightarrow S_j$, where A_l causes the transition from S_i to S_j. Note that S_i and S_j can represent the same system state. This occurs when the expected response to an action does not occur. This would typically be an error condition.

An *event*, E_k, is a subset of state variables. The changes in the observations in these state variables are due to the corresponding action(s). An event alone with its corresponding actions is notated as $A_lE_k : S_i \rightarrow S_j$.

The *temporaldistance* specifies the period of time between two states of a state transition. Temporal distance may be defined as a specific value or as a range such as $D = T_i - T_{i-1}$. Temporal distance may be defined as a specific value or as a range such as $D = T_i - T_{i-1} > 0$. Temporal distance from the root node is always 0 since the root node is system stable state and the second state is always the first evidence of a disturbance or attack.

The *path* is a sequence of signatures that describe a specific scenario. A *signature* contains a system state, its start time, actions and events, and temporal distance to the previous signature. A signature is formally represented as $\{S, T, Action \bigcap Event, D\}$. Once the path with temporal state transitions for the corresponding scenario ID is determined, the information in all signatures involved in the path is used to create the rule for our intrusion detection framework. We denote a path as $path = (Q_i, V, E, \eta)$ where Q_i is the path name which is the scenario name, V is a finite nonempty set of vertices/nodes, and $E = V \times V$ is a set of direct edges in the Bayesian network, and the function η is in the form of assigning observations to state variables, marking labels to each vertex. In this function, S is the name of this vertex, T is the start time of this vertex, $(Action \bigcap Event)$ is the label of this vertex and D is the temporal distance to previous vertex/vertecis.

The *Bayesiannetwork* is composed of a number of paths. Therefore, it has the same composition as a path. We hence denote the Bayesian network as $\bigcup path_i = (Q_i, V, E, \eta_B)$, where $\eta_B(V) = \langle S, T, (Action \bigcap Event), Pr, D\rangle$. Note that the extra parameter Pr represents the conditional probability of the vertex. The construction of Bayesian network starts from the construction of a path. A path starts from an initial vertex standing for the system stable state. The expertise about the system under study e.g. knowledge about legitimate scenarios and threads is required when specifying the conditional probabilities. If the external impacts to our system are not considered, for example transmission errors in the telecommunication channel, it is reasonable to assign either 0 or 1 to each conditional probability so that all causal relationships become deterministic. And the action and event pair with its conditional probability of 1 will be used to mark the newly decided vertex. Paths propagate along with time until they distinct with each other

4.2 The Process of Constructing Bayesian Network for Over Current Protection Scheme

This section provides an overview of the process used to develop a Bayesian network for a set of scenarios (power system disturbances and cyber threats) which can occur for a given system. First, for a given system, a set of measurable variables are identified. Measured variables are system specific and are used to provide information about the system state. The list of measured variables

is created by examining available data sources and comparing this with data needed to detect symptoms of set of scenarios. Quantized ranges are created for each measure variable to limit the state space. Domain expertise is required to list a set of possible actions which occur in the system. For each action a corresponding measurable event is identified. The actions are arcs in the Bayesian network which cause a state transition and the measurable events are evidence that the system has changed states. A Bayesian network is built by drawing a path through a set of system states which when connected describe a scenario. The goal of this process is to create a Bayesian network with a unique path for each scenario. When first drawn the Bayesian network paths may not be unique and overlapping occurs in the process in which the domain expert searches addition actions and events which when added allow each path to become unique. Once each scenario has a unique path the Bayesian construction process is complete and the paths for each scenario represent a measurable signature for each scenario.

4.2.1 Step 1 Identifies Measurable Variables or Events

The variables measured for the over current protection case study are relay operation state, presence of a remote command to trip the relay at the control room, presence of a remote command to trip the relay on the substation network as detected by Snort, and PMU current measurements from the bus connected to the relay.

Table 1: Component ranges

Component Name	Range
Line Current Magnitude	[**H**igh, **W**arning, **N**ormal, **Z**ero]
Snort log	[True, False]
Relay log	[True, False]
Control log	[True, False]

Operators need the ability to remotely trip a relay to remove a transmission line from the power system. Transmission lines are taken out of service to allow for maintenance. The presence of the trip command in the control room is either true or false as stated in Table 1. This measurement can be extracted from the human machine interface tool used by an operator to remotely trip the relay. For this work, this value was simulated. If the remote trip command was intended to be legitimate this value was set to true. If the remote trip command was illegitimate this value was set to false. Because a relay can be remotely tripped it is possible for an attacker to direct a spurious command to the relay to trip the relay without the knowledge or approval of system operators. In this case, the remote trip command will be seen on the substation network as it travels to the relay. For this work, a Snort signature was used to detect the presence of a remote trip command. This signature alerts for both legitimate and illegitimate remote trip commands and therefore is not enough information by itself to declare the presence of an attack. If the Snort signature detects a remote trip command this measurement is True and if a remote trip is not detected this measurement is false.

Relays which have operated open or close a contact connected to the breaker. This contact state is stored in the relay log file and the state of this contact can be read to learn the intended breaker status. This is the intended breaker status due to the possibility of breaker failure. We call this variable the relay status. Relay status is true if the relay has operated and false if the relay has not operated.

PMU can be used to measure a power system bus's voltage and current at rates up to 120 times per second. For this work a PMU was used to provide a measurement of current at the transmission line as a redundant indication of the transmission line status as well as whether the relay is opened or closed. The PMU measurement is a real number which can take a continuous set of values and therefore introduces an infinite number of states in the system state space. Such continuous measurements need to be quantized to a finite set of ranges. The current magnitude can range from zero to infinity. However, for the 2-generator 3-bus transmission system and the non-pilot directional over current protection scheme described previously, the PMU current can be broken into 4 ranges. The first range is over current which covers the case that the current exceeds the pickup of the relay. The instantaneous over current pickup for relay R2 in Figure 2 is set to 800 Amperes. The over current range is defined as [800 Ampere, Infinity) and denoted as "High". The second range is a warning range to allow for the case in which a fault occurs on an adjacent line which it is not cleared due to relay(s) failures. The warning range is hence below instantaneous over current pickup and above certain value. The minimum warning level current must be above a normal operating current when the system has maximum operating load. The current was measured within RSCAD by setting to operate at maximum load. Short circuit analysis was used to predict the short circuit current for a fault on adjacent lines. From these experiments in the test bed, we conclude the warning range is (600 Ampere, 800 Ampere).

The third range is the zero-magnitude range where current magnitude is relatively small. The zero current is not necessarily zero but is a relatively small value. For example, when relays R1 and R4 trip due to a fault on line L2, the current magnitude measured by the PMU at R2 is approximately 50 Amperes. Therefore, the zero-magnitude range for this test bed is defined as [0 Ampere, 60 Ampere). The fourth range is the normal range. The normal range is for current magnitudes above the zero range and below the warning range. For this case study the normal range is set to [60 Ampere, 600 Ampere). Table 2 shows the list of the permissible ranges of all measured variables in the test bed.

4.2.2 Step 2 Specifies Actions and Events which Describe System State Transitions

Table 2: Actions and event space

Action ID	Action	Event ID	Event
A0	Normal	E0	PMU = N, Relay log = F, Snort log =F, Control panel log = F
A1	Control panel sending tripping command	E1	PMU = H
A2	Fault in the line	E2	Control panel log = T
A3	Snort detecting tripping command	E3	Snort log = T
A4	Relay tripping	E4	Relay log = T
A5	Relay operating time out	E5	Relay log = F
A6	Injecting false data beyond the pickup	E1	PMU = H
A7	Injecting false data in permissible range	E6	PMU = N
A8	Injecting false data of zero magnitude	E7	PMU = Z
A9	Breaker opened	E7	PMU = Z
A10	Breaker failed	E8	PMU = H
A11	Fault in adjacent line	E9	PMU = W

The 10 scenarios previously described in the threat model section can be broken into series of actions and corresponding measurable events for each action. For the case study 12 actions and 10 corresponding events are identified. Identification of the actions and corresponding events requires domain expertise. Actions are actions which occur as part of a larger scenario. Actions cause system state to change. Measurable events are sensor measurements are values from log files which are indicative of a power system state. Measurable events are evidence of the system state. The concatenated action identifier and corresponding event identifier is unique to a system state and is therefore used to mark the states in the Bayesian network.

To create the action and corresponding events list the domain expert attempts to describe a set of actions which occur during a given scenario. For each action, the domain expert searches for a unique corresponding measurable event. It may not always be possible to identify a measurable event. It also may not be possible to identify a unique measurable event as seen by multiple actions in Table 2 which share the same event.

The action A10 represents a fault in an adjacent line. This action is used to allow scenarios from other lines to be classified and therefore to allow differentiation between scenarios occurring at the local relay and scenarios which occur at remote relays.

4.2.3 Step 3 Determines Paths for the Set of System Scenarios

A path is a set of system states arranged in temporal order. Each scenario is described by a path. All paths start from an initial state, the system stable state. For this case study the system stable state is the case in which the current magnitude, measured by the PMU, at line L1 is normal, the relay has not operated, the operator has not sent a remote trip command from the control room, and Snort has not detected a remote trip command on the substation network. This state is marked as action A0, event E0 or A0E0. The paths for system scenarios are described below as examples.

Scenario Q1 is an over current fault on line L1. The first action which occurs is the fault. The fault will be reflected by the current magnitude changing to the high state. This is measured by the PMU. This action and event pair is A1E1. The second action for scenario Q1 is the over current relay tripping. This is measured by reading the relay state from the relay log. This action and event pair is A4E4. The final action for scenario Q1 is the breaker is opened. This is evidenced by the event current magnitude changing to 0 Amperes which is measured by the PMU. This action and event pair is A9E7.

Scenario Q2 is transmission line L1 taken out of the power system for maintenance. The first action which occurs for this scenario is operator sends a remote command to trip the relay. This command is detected by reading the operators human machine interface log. This action and event pair is A2E2. The second action which occurs is the presence of the remote trip command in the substation communications network. This is detected by a Snort alert. This action and event pair is A3E3. The third action is the over current relay tripping. This is measured by reading the relay state from the relay log. This action and event pair is A4E4. The final action for scenario Q1 is the breaker is opened. This is evidenced by the event current magnitude changing to 0 Amperes which is measured by the PMU. This action and event pair is A9E7.

Scenario Q3 is a command injection attack which remotely trips the relay. The first action which occurs for this scenario is the presence of the remote trip command in the substation communications network. This is detected by a Snort alert. This action and event pair is A3E3. The third action is the over current relay tripping. This is measured by reading the relay state from the relay log. This action and event pair is A4E4. The final action for scenario Q1 is the breaker is opened. This is evidenced by the event current magnitude changing to 0 Amperes which is measured by the PMU. This action and

event pair is A9E7.

The actions and events for a given path may occur simultaneously or may occur over a temporal distance. The temporal distance is defined as the time between action and event pairs or system states. Some paths have minimum or maximum temporal distance requirements. For example, if a fault occurs in line L1, relay R1 should trip in one cycle, then the breaker should open within the specified breaker operating time. Paths may have order requirements. This is specified by a temporal distance between an action and event pair or two states which is greater than 0.

4.2.4 Step 4 is Construction of the Bayesian Network with Temporal-state Transitions

A typical Bayesian network is represented as an acyclic directed graph (DAG) with a set of vertices and edges. The Bayesian network in this paper is distinct from traditional Bayesian networks in that its vertices contain a set of state variables, time, actions and events. An action and corresponding event pair (i.e. A#E#) is used to mark the vertices. The Bayesian network is constructed by graphing each path in temporal order. System states are represented on the Y-axis. As such a row on the graph will always have the same action and event pairs. The X-axis indicates time. Temporal distance between states or action and event pairs is shown by a path traveling from left to right on the graph. The X-axis unit is unlabeled because the different paths may have large temporal distance disparity. Therefore unit time is used to show order. The complete Bayesian network for all scenarios for relay R2 is shown in Figure 4.

In some cases the same action and event pair will lead to two different system states. For example, action and event pair A4E4 is show on two rows of the Bayesian network graph. In 3 cases the A4E4 action and event pair leads to system state S5 and on 1 case the same action and event pair leads to system state S4. The action and event pair represents a relay tripping and the resulting indication of such in the relay log. For state S5 this has occurred due to an over current fault. For state S5 this has occurred due to a command injection attack. The difference between these 2 states is that S5 has a high PMU current measurement and S4 has a normal PMU current measurement. Each path in the Bayesian network starts from the initial vertex T0S0 with label A0E0, which represents the system stable state.

The path includes all information needed for to build signatures for each scenario. The path can be used to describe the corresponding scenario by the sequences of actions and actions that cause the state transitions along with time. This Bayesian network consists 9 paths for 10 scenarios. Note that, scenarios Q5, Q6 and Q8, Q10 have the same paths. These paths contain leaf nodes, each of which has two possible actions. The two actions are mutually exclusive such that they cannot happen at the same time. However, at this stage there is not enough information to differentiate them. The extra information needed is the log from opposite relay on the same transmission line. For the scenario Q5 and Q6, if the over current is due to a fault in line L1 (A1E1), R1 will trip (R1 log = True) and open the breaker BR1 then the path will lead to scenario Q5. While in scenario Q6, the data injection (A5E5) does not cause relay R1 to trip therefore R1 log = False. Note that, this is also the case when distinguishing Q8 and Q9. The log information from relay R1 is needed again because the PMU in relay R2 reading zero current could implies two scenarios: one is due to a zero-value data injection attach (A8E7); the other results from the breaker being opened (A9E7) by the opposite relay R1. This second scenario belongs to the IDS monitoring the opposite relay, R1, and therefore is categorized as "Other" by the IDS monitoring relay R2. The path Q10 represents scenarios which have occurred on an adjacent line or on the opposite relay on the same line. All paths in Figure 4 represent all possible scenarios that could happen in the test bed.

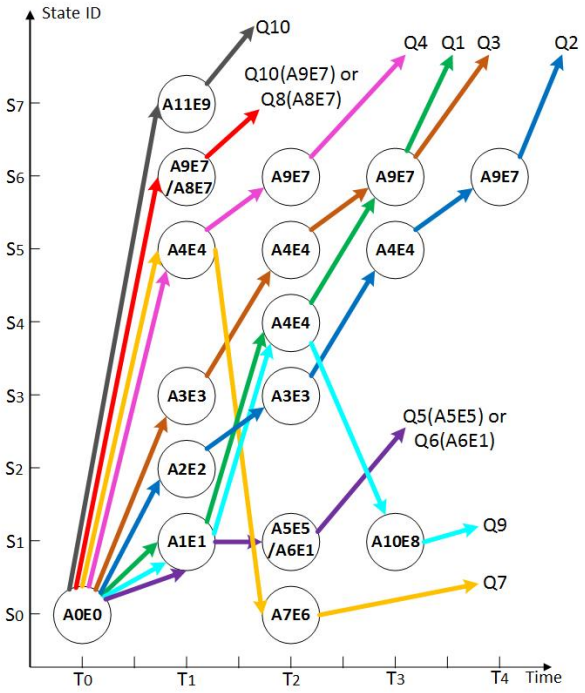


Figure 4: Complete Bayesian network for relay R2

4.3 Result and Discussion

An IDS was implemented from the Bayesian network shown in Figure 4. The IDS reads PMU current measurements, relay trip status, the snort log, and the control panel log and uses this information to track system states. The IDS monitors transitions from state to state to detect paths which match those shown in the Bayesian diagram. The result of the IDS was classification of the 10 scenarios. An experiment was conducted using the test bed shown in Figure 3. Scenarios Q1 to Q9 were simulated on a sin-

gle transmission line of the 2-generator 3-bus transmission system. Line L1 and relay R2 were the target of all power system disturbances and or cyber attacks. Each scenario was reproduced 5 times; each time with a different load. Table 3 presents the results from the experiment. Each relay in the test bed included an embedded PMU. Time stamped PMU current, relay logs, Snort logs, and control panel logs were stored separately for each relay in the test bed. The IDS built from the Bayesian network graph in Figure 4 was used to post process data collected for each relay in the system and provide classification results.

Table 3 lists classification results from the experiment for each relay. The first column of Table 3 shows the scenario simulated. The next four columns show the classification result seen from the IDS when processing data associated with relays R1 to R4 respectively. Each scenario was run 5 times. The values in the cells of table III indicate the classified scenario or are labeled with a "-" if no classification result was generated for that scenario.

The results can be analyzed in 3 groups. First, IDS at the target relay, R2, always classified the scenarios correctly. The classification results for the IDS for R1 require explanation. First, scenario Q1 is an over current fault. Since the over current fault is seen by both relays on the line and both relays are programmed to respond to the fault instantaneously, the IDS classified this fault as Q1. This result is correct. Scenario Q2 simulates an operator taking line L1 out of service for maintenance. In this case the operator will remotely trip the relays R1 and R2 simultaneously. As such the IDS at R1 correctly classified these actions as scenario Q2. Scenarios Q3 and Q4 were classified as scenario Q10. Scenario Q10 is intended as a class which represents actions which occur at another relay. The intent of this work is to develop an IDS which monitors an atomic unit, a single relay, but ignores actions at other relays since these relays will have their own IDS. Scenario Q10 includes cases when the current at the relay, as measured by the PMU, enters the warning range, below the pickup current but above the normal range. Scenario Q10 also includes when the opposite breaker opens and the local PMU current drops to 0 Amperes without a previous current measurement greater than the pickup current setting. This second case occurs to one relay when its opposite relay trips. Scenario Q3 is a cyber attack in which a remote command is used to trip relay R2. In this case there is no longer a path for current from generator G1 to the load. This causes the current at relay R1 to drop to 0 Amperes without R1 first tripping which in turn causes the Q10 classification. Scenario Q4 is a physical attack in which a substation intruder or insider uses relay R2's face to trip relay R2 without permission. Again in this case there is not path for current to flow from generator G1 to the load and this leads to a Q10 classification. For scenarios Q3 and Q4 the Q10 classification is consider correct. Scenario Q5 is an over current fault in which relay R2 fails to operate. This missed operation may be due to relay failure or incorrect setting. In this case relay R1 operates correctly and trips due to the same over current fault on line L1. This is correctly classified as scenario Q1. Scenario Q6 and Q8 are not classified by the IDS at relay R1. This occurs because scenario Q6 and Q8 are two data injection attacks targeting the PMU at R2. Since this is a false PMU measurement the actual line current measured at relay R2 remains normal and the IDS at relay R1 sees no signature of a scenario. Scenario Q7 is a data injection attack which is attempting to mask an over current fault which causes relay R2 to trip. In this case there is a valid over current fault on the line and this fault is seen by relay R1. Since the data injection attack is limited to relay R2, relay R1's IDS correctly classifies this as scenario Q1. Finally, Q9 is an over current fault on line L1 with breaker failure at BR2. In this case there is a valid over current fault on the line and this fault is seen by relay R1. Since the breaker failure is limited to relay R2, relay R1's IDS correctly classifies this as scenario Q1.

The IDS(s) at relays R3 and R4 always classified scenarios as either Q10 or did not provide a classification at all. Scenarios Q1, Q5, Q7 and Q9 were all classified as scenario Q10. Each of these scenarios involve an over current fault on line L1. This fault on the neighboring line will cause the PMU current at relays R3 and R4 to read in the warning range which leads to this scenario Q10 classification. Scenarios Q2, Q3, and Q4 all involve relay R2 tripping without a prior fault. These cases may lead to a drop in current at R3 and R4. However, since there is another source in the power system, as is expected due to N-1 generator redundancy requirements, the current at R3 and R4 does not drop to 0 Amperes and therefore no signature of a Bayesian network path is available for classification. Scenarios Q6 and Q8 are data injection attacks which alter the PMU current measurement from relay R2. This has no effect on relays R3 and R4 and therefore there is no signature of a Bayesian network path classify. All of the classification results for relays R3 and R4 are considered correct.

4.4 Conclusion and Future Work

This paper introduces a methodology for developing specification based intrusion detection systems (IDS) for cyber-physical systems. The methodology involves first developing a threat model for the system to be monitored which includes relevant cyber attacks and any expected disturbances which may occur normally in the system. The threat model is grouped into a set of scenarios to be classified by the IDS. Next, a set of actions and measurable events is created for the system. The actions and events pair to move the system from state to state. A Bayesian network graph is constructed which shows each scenario as a path which describes system state transitions and temporal order for each path. In order to provide a unique classification for each scenario the Bayesian network graph must include a separate path for each scenario. IDS designers search for actions and measurable events in a loop until separate paths exist for each sce-

Table 3: Actions and event space

Scenario Simulated	Scenario Detected IDS@R1	Scenario Detected IDS@R2	Scenario Detected IDS@R3	Scenario Detected IDS@R4
Q1: Over current fault@L1	Q1	Q1	Q10	Q10
Q2: L1 removed for maintenance	Q2	Q2	-	-
Q3: command injection attack; remotely trip R2	Q10	Q3	-	-
Q4: physical attack; trip R2 at faceplate	Q10	Q4	-	-
Q5: fault@L1; relay does not trip	Q1	Q5	Q10	Q10
Q6: data injection attack IL1 > pickup	-	Q6	-	-
Q7: data injection attack; IL1<pickup; fault@L1	Q1	Q7	Q10	Q10
Q8: data injection attack; IL1=0	-	Q8	-	-
Q9: fault@L1; BR2 breaker failure	Q1	Q9	Q10	Q10

nario. Once the Bayesian event graph is complete an IDS can be built.

The proposed method for developing specification based IDS requires system expertise. This can be a burdensome requirement. A separate version of the IDS is deployed for each relay meaning the IDS does not need to consider attacks and disturbances which occur on a separate line. For this work the non-pilot directional over current relay protection scheme was specified. Other relaying schemes would also need to be specified. Each relay requires an instance of the IDS. IDS instances may be deployed in the substation at the relay location or a single server at a central location may run multiple instances of IDS to monitor multiple relays. The computing and networking resource requirements of both deployment options should be considered in future work. Other future work will the use of clustering algorithms [39, 41, 42] and game theory approaches to learn rules from observed behavior and game theory to better model interactions between system components before learning rules. Finally, the best approach to feature selection will be researched for various target systems [22].

For this work, a case study was used to demonstrate the effectiveness of the IDS development methodology. The case study was applied to the non-pilot directional over current relay protection scheme for a modified 2-bus 2-generator system taken from a section of the IEEE 9-bus 3-generator system. Nine scenarios were developed. The scenarios include 4 power system disturbance cases and 5 cyber attacks. A Bayesian network graph for the 9 scenarios was developed and data logs were captured for each scenario from the perspective of each relay in the test bed power system. The resulting IDS was used to post process data collected from perspective of each relay separately. All case study scenarios were correctly classified.

References

[1] U. Adhikari, T. H. Morris, N. Dahal, S. Pan, R. L. King, N. H. Younan, and V. Madani, "Development of power system test bed for data mining of synchrophasors data, cyber-attack and relay testing in rtds," in *2012 IEEE Power and Energy Society General Meeting*, pp. 1–7, July 2012.

[2] D. E. Bakken, A. Bose, C. H. Hauser, E. O. Schweitzer III, D. E. Whitehead, and G. C. Zweigle. "Smart generation and transmission with coherent, real-time data," Tech. Rep. TR-GS-015, School of Electrical Engineering and Computer Science, Washington State University, Jan. 2011.

[3] R. Berthier and W. H. Sanders, "Specification-based intrusion detection for advanced metering infrastructures," in *Dependable Computing (PRDC), 2011 IEEE 17th Pacific Rim International Symposium on*, pp. 184–193, Dec. 2011.

[4] W. L. Buntine, "A guide to the literature on learning probabilistic networks from data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 8, pp. 195–210, Apr. 1996.

[5] A. Carcano, A. Coletta, M. Guglielmi, M. Masera, I. N. Fovino, and A. Trombetta, "A multidimensional critical state analysis for detecting intrusions in scada systems," *IEEE Transactions on Industrial Informatics*, vol. 7, pp. 179–186, May 2011.

[6] A. Cárdenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems," in *Pro-*

ceedings of the 3rd Conference on Hot Topics in Security, HOTSEC'08, pp. 6:1–6:6, 2008.

[7] A. Cardenas, S. Amin, B. Sinopoli, A. Giani, A. Perrig, and S. Sastry, "Challenges for securing cyber physical systems," in *Workshop on Future Directions in Cyber-physical Systems Security*, DHS, July 2009. [Online] Available: http://chess.eecs.berkeley.edu/pubs/601.html.

[8] E. Charniak, "Bayesian networks without tears," *AI Magazine*, vol. 12, no. 4, pp. 50–63, 1991.

[9] S. Cheung, B. Dutertre, M. Fong, U. Lindqvist, K. Skinner, and A. Valdes, "Using model-based intrusion detection for scada networks," in *Proceedings of the SCADA Security Scientific Symposium*, pp. 127–134, 2007.

[10] D. Dzung, M. Naedele, T. P. von Hoff, and M. Crevatin, "Security for industrial communication systems," *Proceedings of the IEEE*, vol. 93, pp. 1152–1177, June 2005.

[11] N. Falliere, L. O. Murchu, and E. Chien. "W32.stuxnet dossier,". tech. rep., Oct. 2010. [Online] Available: http://www.symantec.com/com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.

[12] U. S. Government Accountability Office (GAO). "Gao-11-117: Electricity grid modernization: Progress being made on cybersecurity guidelines, but key challenges remain to be addressed,". tech. rep., Jan. 2011.

[13] F. Geramiraz, A. S. Memaripour, and M. Abbaspour, "Adaptive anomaly-based intrusion detection system using fuzzy controller," *International Journal of Network Security*, vol. 14, pp. 352–361, Nov. 2012.

[14] A. Di Giorgio and F. Liberati, "Interdependency modeling and analysis of critical infrastructures based on dynamic bayesian networks," in *Control Automation (MED), 2011 19th Mediterranean Conference on*, pp. 791–797, June 2011.

[15] The Smart Grid Interoperability Panel Cyber Security Working Group. "Nistir 7628 guidelines for smart grid cyber security: Vol. 2, security architecture and security requirements,", Aug. 2010. [Online] Available: http://csrc.nist.gov/publications/nistir/ir7628/nistir-7628_vol2.pdf.

[16] N. Hadjsaid, C. Tranchita, B. Rozel, M. Viziteu, and R. Caire, "Modeling cyber and physical interdependencies - application in ict and power grids," in *2009 IEEE/PES Conference on Power Systems Conference and Exposition (PSCE'09)*, pp. 1–6, March 2009.

[17] D. K. Holstein and J. Diaz, "Cyber security management for utility operations," in *the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)*, 2006.

[18] S. Horowitz, D. Novosel, V. Madani, and M. Adamiak, "System-wide protection," *IEEE Power and Energy Magazine*, vol. 6, pp. 34–42, Sep. 2008.

[19] S. H. Horowitz and A. G. Phadke, *Power System Relaying*. Wiley, 2008.

[20] http://openpdc.codeplex.com. *The Open Source Phasor Data Concentrator (OpenPDC)*.

[21] E. O. Schweitzer III and H. J. Altuve, "Real-time synchrophasor applications for wide-area protection, control, and monitoring,", 2009. [Online] Available: https://www.selinc.com/WorkArea/DownloadAsset.aspx?id=6388.

[22] P. Kabiri and M. Aghaei, "Feature analysis for intrusion detection in mobile ad-hoc networks," *International Journal of Network Security*, vol. 12, pp. 42–49, Jan. 2011.

[23] U. B. Kjaerulff and A. L. Madsen, *Bayesian Network and Influence Diagrams, A Guide to Construction and Analysis*, Springer, 2012.

[24] C. Ko, M. Ruschitzka, and K. Levitt, "Execution monitoring of security-critical programs in distributed systems: a specification-based approach," in *1997 Proceedings of IEEE Symposium on Security and Privacy,* pp. 175–187, May 1997.

[25] F. R. Lin, C. H. Chiu, and S. C. Wu, "Using bayesian networks for discovering temporal-state transition patterns in hemodialysis," in *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'2002)*, pp. 1995–2002, Jan. 2002.

[26] H. Lin, A. Slagell, C. Di Martino, Z. Kalbarczyk, and R. Iyer, "Adapting bro into scada: Building a specification-based intrusion detection system for the dnp3 protocol," in *Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop (CSIIRW'13)*, pp. 5:1–5:4, 2013.

[27] Y. Liu, P. Ning, and M. Reiter, "False data injection attacks against state estimation in electric power grids," in *Proceedings of the 16th ACM Conference on Computer and Communications Security (CCS'09)*, pp. 21–32, 2009.

[28] V. Madami, M. Adamiak, and M. Thakur, "Design and implementation of wide area special protection schemes," in *Proceedings 2004 57th Annual Conference for Protective Relay Engineers*, pp. 392–402, Apr. 2004.

[29] C. Marinez, M. Parashar, J. Dyer, and J. Coroase. "Phasor data requirements for real time wide-area monitoring, control and protection applications,". tech. rep., 2005. [Online] Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.122.1737.

[30] M. Masera and I. N. Fovino, "Effects of intentional threats to power substation control systems," *International Journal of Critical Infrastructure*, vol. 4, no. 1-2, pp. 129–143, 2008.

[31] R. Mitchell and I. R. Chen, "Behavior rule based intrusion detection for supporting secure medical cyber physical systems," in *2012 21st International Conference on Computer Communications and Networks (ICCCN'2012)*, pp. 1–7, July 2012.

[32] Y. Mo, T. H. Kim, K. Brancik, D. Dickinson, H. Lee, A. Perrig, and B. Sinopoli, "Cyber physical security of a smart grid infrastructure," *Proceedings of the IEEE*, vol. 100, pp. 195–209, Jan 2012.

[33] T. Morris, S. Pan, J. Lewis, J. Moorhead, N. Younan, R. King, M. Freund, and V. Madani, "Cybersecurity risk testing of substation phasor measurement units and phasor data concentrators," in *Proceedings of the Seventh Annual Workshop on Cyber Security and Information Intelligence Research (CSIIRW'11)*, pp. 24:1–24:1, 2011.

[34] R. Moxley and D. Dolezilek, "Case studies: Synchrophasors for wide-area monitoring, protection, and control," in *Proceedings of 2nd IEEE PES International Conference and Exhibition on Innovative Smart Grid Technologies (ISGT Europe)'2011*, pp. 1–7, Dec. 2011.

[35] National Energy Technology Laboratory (NETL). "A systems view of the modern grid,". tech. rep., U.S. Department of Energy (DOE), Jan. 2007.

[36] J. P. Nieto, L. E. Garza, M. Garza, and R. Morales, "Fault diagnosis of industrial systems with bayesian networks and neural networks," in *Proceedings of the 7th Mexican International Conference on Artificial Intelligence (MICAI'08)*, pp. 998–1008, 2008.

[37] P. Ning and S. Jajodia, *Intrusion Detection Techniques*, Wiley, 2004.

[38] Critical Infrastructure Protection (NAERC-CIP) North American Electric Reliability Corporation. "Nerc standardscip-002-4 through cip-009-4,", 2012. [Online] Available: http://www.nerc.com/page.php?cid=2—20.

[39] A. Patcha and J. M. Park, "A game theoretic formulation for intrusion detection in mobile ad hoc networks," *International Journal of Network Security*, vol. 2, pp. 131–137, March 2006.

[40] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*, Morgan Kaufmann, 1988.

[41] Q. Quan, C. J. Xiao, and R. Zhang, "Grid-based data stream clustering for intrusion detection," *International Journal of Network Security*, vol. 15, pp. 1–8, Jan. 2013.

[42] Q. Quan, T. Wang, and R. Zhan, "Relative network entropy based clustering algorithm for intrusion detection," *International Journal of Network Security*, vol. 15, pp. 16–22, Jan. 2011.

[43] M. F. Raihan and M. Zulkernine, "Detecting intrusions specified in a software specification language," in *29th Annual International Conference on Computer Software and Applications (COMPSAC'2005)*, vol. 1, pp. 143–148, July 2005.

[44] S. M. Rinaldi, J. P. Peerenboom, and T. K. Kelly, "Identifying, understanding, and analyzing critical infrastructure interdependencies," *IEEE Control Systems*, vol. 21, pp. 11–25, Dec. 2001.

[45] P. Sauer and A. Pai, *Power System Dynamics and Stability*, Stipes Publishing Co., 2007.

[46] R. Sprabery, T. Morris, S. Pan, U. Adhikari, and V. Madani, "Protocol mutation intrusion detection for synchrophasor communications," in *Proceedings of the Eighth Annual Cyber Security and Information Intelligence Research Workshop (CSIIRW'13)*, pp. 41:1–41:4, 2013.

[47] S. Sridhar, A. Hahn, and M. Govindarasu, "Cyber physical system security for the electric power grid," *Proceedings of the IEEE*, vol. 100, pp. 210–224, Jan. 2012.

[48] IEEE Standard, "IEEE standard for synchrophasor data transfer for power systems," *IEEE Std C37.118.2-2011 (Revision of IEEE Std C37.118-2005)*, pp. 1–53, Dec. 2011.

[49] M. Uddin, A. A. Rehman, N. Uddin, J. Memon, R. Alsaqour, and S. Kazi, "Signature-based multi-layer distributed intrusion detection system using mobile agents," *International Journal of Network Security*, vol. 15, pp. 97–105, Jan. 2013.

[50] P. Uppuluri and R. Sekar, "Experiences with specification-based intrusion detection," in *Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection (RAID'00)*, pp. 172–189, 2001.

[51] www.snort.org. *Snort*.

[52] Y. Liu and H. Man, "Network vulnerability assessment using bayesian networks," in *Proceedings of SPIE - Data Mining, Intrusion Detection, Information Assurance and Data Networks Security (SPIE'05)*, pp. 61–71, 2005.

[53] Z. Yongli, H. Limin, and Lu Jinling, "Bayesian networks-based approach for power systems fault diagnosis," *IEEE Transactions on Power Delivery*, vol. 21, pp. 634–639, Apr. 2006.

[54] Z. Zhang, H. Shen, and Y. Sang, "An observation-centric analysis on the modeling of anomaly-based intrusion detection," *International Journal of Network Security*, vol. 4, pp. 292–305, May 2007.

[55] C. Zimmer, B. Bhat, F. Mueller, and S. Mohan, "Time-based intrusion detection in cyber-physical systems," in *Proceedings of the 1st ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS'10)*, pp. 109–118, 2010.

[56] J. Zuo, R. Carroll, P. Trachian, J. Dong, S. Affare, B. Rogers, L. Beard, and Y. Liu, "Development of tva superpdc: Phasor applications, tools, and event replay," in *Proceedings of 2008 IEEE Conference on Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, pp. 1–8, July 2008.

Shengyi Pan received the B.Eng. degree from Fuzhou University, China, in 2008 and the M.S. degree from University of Sheffield, U.K., in 2009. He is currently working toward the Ph.D. degree in Mississippi State University, Mississippi State. His research interests include security and intrusion detection for computer network, process control system, and smart grid.

Thomas Morris received the Ph.D. degree from Southern Methodist University, Dallas, TX, in 2008. He currently serves as Associate Professor of Electrical and Computer Engineering at Mississippi State University, Mississippi State, MS, USA. His research interests include industrial control system penetration testing and intrusion detection systems.

Uttam Adhikari received the B.S. degree in electrical engineering from Tribhuvan University, Nepal, in 2005, and is currently pursuing graduate studies at Mississippi State University, Mississippi State. His research interests include wide area monitoring, control, and cyber security in smart grid.

An Auto-tuning Sanitizing System for Mitigating Injection Flaws

Jan-Min Chen
(Corresponding author: Jan-Min Chen)
Department of Information Management, Yu Da University of Science and Technology
No. 168, Hsueh-fu Rd., Tanwen Village, Chaochiao Township, Miaoli County 361, Taiwan, R.O.C.
(Email: ydjames@ydu.edu.tw)
(Received July 8, 2013; revised and accepted May 20, 2014)

Abstract

Injection attacks are dangerous and ubiquitous, contributing enormously to some of the most elaborate Web hacks. Enforcing proper input validation is an effective countermeasure to improve injection flaws. Unless a web application has a strong, centralized mechanism for validating all input from HTTP requests, injection flaws are very likely to exist. However, improper constraining rules may induce some detection error. False negatives may render security risks and false positives will cause improper limits of input characters. In this paper, we design an auto-tuning system to help validating input for each vulnerable injection point. A proper validation rule can be automatically generated through an auto-tuning mechanism. The experimental results show that the system can effectively protect against injection attacks and lower false positives while compared with traditional methods.

Keywords: Constraining rule, content filtering, detection accuracy, injection flaws, input validation

1 Introduction

Injection attacks can be very easy to discover and exploit. Hackers take advantage of a weakness in the Web application design to intentionally insert some extra characters in input data to bypass or modify the originally intended functionality of the program. The consequences can run the entire range of severity, from trivial to complete system compromise or destruction. Many application's security vulnerabilities result from generic injection problems. Examples of such vulnerabilities are SQL injection, Shell injection and Cross site scripting (XSS). Enforcing input validation is an effective countermeasure to protect against injection attacks. Traditional methods usually use a generic constraining rule to strictly sanitize all input. It may cause improper limits of input characters because of some false positives.

There are numbers of the researches related to protect Web site against injection attacks: Huang et al. had developed a WebSSARI and a WAVES [8]. The Open Web Application Security Project (OWASP) had launched a WebScarab project [16]. The other available commercial scanners also included IBM Rational's AppScan and SPI Dynamics' WebInspect [9, 23]. Above-mentioned tools just focus on finding Web application flaws. Once web application vulnerabilities have been identified, the ultimate solution is to fix the vulnerabilities in the web application source code itself. However, this can render intrusion risks because proper vulnerability fixing often requires doing something else such as testing, supports coming from third parties and vendors of multiple software components. Thus, some solutions had been proposed to protect against attacks before fixing flaws. Sanctum Inc. provided an AppShield adopting Security Gateway to prevent application-level attacks [22]. Some advanced firewalls also incorporated deep packet inspection technologies for filtering application-level traffic [3]. We had proposed a fixing tool that can be used to improve injection flaws [11]. It can produce proper input validation functions related to the source codes of applications. Next an enhanced prototype adopting a security gateway in front of web server to sanitize malicious input had been proposed solving the problem as source code may not be modified [13]. The above two methods both use a generic constraining rule to validate input, so false positive will be troublesome. Recently, Web application firewalls (WAF) is a popular solution to be used to create an external security layer to improve security, detection, and prevention of attacks before they hit web applications [10, 15, 17]. For WAF, the sanitizing mechanism is a critical technique. However traditional validating methods are susceptible to error because of using single constraining rules to sanitize all input. Lately, we had proposed a heuristic mechanism that can automatically generate proper validation rules based on each vulnerable injection point. The method had been primarily proved both guarantee security (false negatives) and convenience (false positives) [2].

In this paper, we create a sanitizing system to help validating input. For each vulnerable injection point, a proper validation rule can be automatically generated and adjusted itself to new injection attacks through an auto-tuning mechanism. Thus the system can both guarantee better detection accuracy compared with other constraining strategies and better effectiveness to protect against new injection attacks.

The main contributions of this paper are summarized below:

1) It has good "scalability" when applied to Web site growth or new attack patterns because it integrates an injection vulnerability analyzer (finding target), an injection pattern generator (exploitation), and a constraining rule generator (prevention) into an auto-tuning bastion.

2) It proposes an auto-tuning mechanism to effectively improve the detection accuracy of the signature-based content filtering techniques.

3) It designs a system to automatically protect against new injection flaws through the seamless delivery of new constraining rule.

The rest of this article is organized as follows: The second section surveys a number of works relevant to improve injection flaws. In third section, we describe our technical details of auto-tuning sanitizing agent. The system implementation is shown in fourth section and its effectiveness is evaluated in fifth section. The last section concludes the whole paper.

2 The Works Relevant to Improve Injection Flaws

2.1 Injection Flaws

Injection flaws allow attackers to relay malicious code through a web application to another system. The attacker can inject special (meta) characters, malicious commands, or command modifiers into the information and the web application will blindly pass these on to the external system for execution. These attacks include calls to the operating system via system calls, the use of external programs via shell commands, as well as calls to backend databases via SQL [18].

SQL injection is a type of security exploit in which the attacker adds SQL statements through a web application's input fields or hidden parameters to gain access to resources or make changes to data. It is a particularly widespread and dangerous form of injection. It is an attack technique used to exploit web sites that construct SQL statements from user-supply input. SQL injection is a serious vulnerability, which can be found in any environment with an SQL back-end database (Microsoft SQL Server, Oracle, Access, and so on) and used to steal information from a database from which the data would normally not be available and to gain access to host computers through the database engine. As with SQL injection, XSS is also associated with undesired data flow. XSS exploit vulnerabilities in Web page validation by injecting client-side script code. The script code embeds itself in response data, which is sent back to an unsuspecting user. The user's browser then runs the script code. Because the browser downloads the script code from a trusted site, the browser has no way of recognizing that the code is not legitimate. One of the most serious examples of a XSS attack occurs when an attacker writes script to retrieve the authentication cookie that provides access to a trusted site and then posts the cookie to a Web address known to the attacker. This enables the attacker to spoof the legitimate user's identity and gain illicit access to the Web site.

2.2 Content Filtering

There are two strategies are typically employed in content filtering: signature-based and heuristic-based. Simple signature-based detection is an effective and computationally efficient method to detect viruses, but it does have a couple of shortcomings. Signature-based detection involves searching for known patterns of data within executable code. However, it is possible for a computer to be infected with new malware for which no signature is yet known. One type of heuristic approach is intended to overcome the shortcoming. Heuristic-based detection, like malicious activity detection, can be used to identify unknown viruses. Because viruses tend to perform certain actions that legitimate programs do not, they can usually be identified by those actions. If heuristic detection was employed, success depends on achieving the right balance between false positives and false negatives. Due to the existence of the possibility of false positives and false negatives, the identification process is subject to human assistance which may include user decisions, but also analysis from an expert of the antivirus software company [1].

Signature-based intrusion detection system such as the popular Snort program is typically configured with a set of rules to detect popular attack patterns. These rules look almost exactly like firewall rule sets in that patterns can be specified on packet header fields using the usual flexibility of specifying prefixes, wildcarded fields, and port ranges [5]. However, signature detection systems go one step beyond packet filters in complexity by also allowing an arbitrary string that can appear anywhere in the packet payload. String matching in packet content, is also of interest to many applications that make use of content-based forwarding. Radwan et al. show a new implementation of a gateway capable of applying content-based security on attachments of messages, where a single gateway serves several web servers in a web farm [24].

2.3 Constraining Input

Input validation is a secure input handling way for verifying user input to ensure that input is safe prior to use. In general, it checks the user input through constraining rule based on two types of security model (a whitelist and a blacklist). Validation based on whitelist can ensure that all requests are denied unless specifically allowed. A whitelist is a set of all allowed items. The list may involve setting character sets, type, length, format, and range. On the other hand, a blacklist defines what is disallowed, while implicitly allowing everything else.

The benefit of using a whitelist is that new attacks, not anticipated by the developer, will be prevented. A generic solution may not be easily implemented but we can know that is acceptable input data for application program in a localized way. It is much easier to validate data for known valid patterns but it may induce more false positives. On the other hand, a blacklist can clearly dictates that you should specify the characteristics of input that will be denied. Ultimately, however, you'll never be quite sure that you've addressed everything through the blacklist. That is to say, it is an unrealistic idea assuming that all the variations of malicious injection had been known. The blacklist may be quite tempting when you're trying to prevent an attack on your site. However, it allows more abundant input data than a whitelist. In summary, a blacklist often can guarantee fewer false positives than a whitelist but it may induce more false negatives.

2.4 Improving Injection Flaws

Once injection flaws have been identified, the ultimate solution is to fix the vulnerabilities in the web application source code itself. However, this can't be reachable immediately because proper vulnerability fixing often requires doing something else such as supports coming from third parties and vendors of multiple software components. Thus, some solutions had been proposed instead to improve injection flaws before fixing the vulnerabilities.

The Open Web Application Security Project recommends that a thorough validation of any input data needs to be made in order to ensure that the data does not contain any malicious content [16]. SPI Dynamics also suggest using regular expressions for sanitizing data before it is executed by a back-end database [23]. There has been other research into improving injection flaws. Salem et al. had outlined an intercepting filter approach aimed at increasing the security and reliability of web applications by eliminating injection flaw exploitations. The use of filter components, in conjunction with the Intercepting Filter design pattern, can be used to sanitize HTTP Request information before it is ever processed by the web application and had been carried out on Java and .NET based platforms [21]. DOME uses a filter that looks for and marks the locations of system calls and then watches the result of the execution of the actual code [19]. Halfond et al. had proposed a technique that uses a program to automatically build a model of the legitimate queries that could be generated by the application [6].

3 Protecting Against Injection Attacks

An injection flaw is the result of an invalidated input and thus, proper input validation is an effective countermeasure for protecting against injection attacks. In particular, some input validation programs are poorly written, lacking even the most basic security procedures for sanitizing input. Furthermore, some legacy applications may not be able to modify the source of such components. Currently, a WAF is a common solution that can be used in addition to the protected Web site to prevent an immediate injection attack. Although a WAF is language independent and requires no modification to the application source code, it may induce false recognition due to the use of a generic constraining rule.

3.1 Sanitizing Agent

To help performing proper input validation is an effective countermeasure for mitigating injection flaws. It can be achieved by a two steps approach: first, to find all vulnerable injection points and second, to automatically and accurately validate input.

Injection flaws can be found via the Input Validation Testing (IVT). Here the IVT is defined as choosing proper test case that attempt to show the presence or absence of specific errors pertaining to input data [7]. A Web application vulnerability scanner is a common IVT tool. We also had proposed a feasible method for performing IVT. The method not only uncovers vulnerability but also ensures the location where the vulnerability occurs [12]. Upon completion of processes of Web crawling and analysis, an injection point list can be created and filled in the fields of URL and parameter with the result of analysis. And then, IVT will be launched to uncover different vulnerabilities according to the injection point list. The completion of IVT allows us to fill in flaw type field of an injection point list with flaw name to enable the recording of flaws of each injection point.

Next, the task to automatically assist on validating input for each vulnerable injection point is achieved by a sanitizing agent. The sanitizing agent can help validating input via meta-programs. The meta-programs can be translated by a code generator. Each parameter of the same URL in the vulnerable injection point table can generate a meta-program to constrain inputs. For example, if there are some records having the content of a URL field as 'verify.php' in the vulnerable injection point table, our mechanism can automatically generate a meta-program named 'verify.php' to help sanitizing the http requests which surfing destination is 'verify.php'. The Algorithm1 is used for generating a meta-program and the code snippet of the meta-program is presented in Table 1. It needs

to be noted that the italic words should be replaced by a parameter field in the vulnerable injection point table and the constraining rule while generating a meta-program. The constraining rule can be gotten through looking up the constraining rule table according to the flaw type field in the vulnerable injection point list. The meta-program can perform input validation instead of the Web applications having injection flaws. However the meta-program only embodies a sanitizing functionality and so it can't replace the initial program. Thus, after completing the sanitizing procedure, the http request needs to be redirected to original program to obtain prime service.

Algorithm 1 Generating a meta-program

```
//A algorithm for generating a meta-program named as url
OPEN a vulnerable injection point table
GET url
FOR EACH distinct url in a vulnerable injection point table
  FOR EACH parameter having same url
//To generate a program segment of validating input for each vulnerable injection point
   GET constraining rule, parameter's value
   STORE parameter's value to parameterValue // input data
//To search the parameterValue for the number of times of match to the regular expression given in constraining rule
   FOR EACH parameterValue // validation logic begin
    COUNT the number of times of match to the regular expression given in con-straining rule
    STORE the number of times of match to Counter
   END FOR
   IF (Counter > legalSpecialCharCount)
    ECHO error message
    EXIT
   END IF
   SESSION parameter's value // validation logic end
   FILEWRITER url // append the validation logic
  END FOR
  REDIRECT url
END FOR
```

3.2 Generating Ideal Constraining Rules

In general, the constraining rule was based on a generic whitelist or blacklist. A generic whitelist usually only allows case-sensitive alphanumeric characters. A generic blacklist always does its best to include all possible malicious characters to guarantee same false negative. A blacklist allows more abundant input data than a whitelist so validation based on a blacklist can cause fewer false positives than a whitelist. However, it may induce more false negatives. In general case, a configurable set of malicious characters is used to reject the input but it is an unrealistic idea assuming that all the variations of malicious injection had been known. Therefore a validation based on a blacklist should guarantee acceptable false negative, and then do everything possible to reduce false positive. The generic constraining rule can guarantee security, but it may cause more false positives, causing inconvenient because of improper limitation of input characters. Thus, we need an intelligent method for gathering necessary characters in a blacklist to generate an ideal constraining rule according to the actual situation. While inspecting some injection attack strings, we find that most special (not case-sensitive alphanumeric) characters in the strings don't appear alone. For the special characters appearing in an injection string, just only one of them is required to be added in a blacklist. Thus we think that a better method is only put necessary special characters in a blacklist.

Table 1: Example of a simple meta-program

```
<? // a simple meta-program
  $id0=$_POST['myusername'];
  $pattern0="/[=;_'>%<\(@:&\\-\|!\.\+\/]/";
  $i = 0;
  while((preg_match($pattern0, $id0, $matches))
    && $i < 2 )
    {
        $i++;
        $temp = preg_split("/[$matches[0]]/",
          $id0);
        $id0 = implode($temp);
    }
    if ($i >= 2)
    {
        echo "illegal character detected";
        exit;
    }
  $_SESSION["myusername"] =$id0;
  redirect ($url);
?>
```

We think that the necessary special character is which having most appearing rate in comparison with other special characters in an injection string. According to the clue, we propose the Algorithm2 for choosing the special characters really having to be added in the blacklist. The algorithm can be used to generate an ideal constraining rule according to various types of injection patterns. Thus the ideal constraining rule can be used in the meta-program to not only guarantee same false negative but also reduce more false positives in comparison with using a generic blacklist while performing input validation.

Algorithm 2 Generating a ideal constraining rule

```
FOR EACH injectionString(i)
  Let candidateString(i) = all specialCharacters in an injection string
  //specialCharacter i.e. not case-sensitive alphanumeric
  Remove same characters from candidateString(i)
  Let specialCharacter (i, j) = the specialCharacter had appeared in candidateString(i)
  For EACH specialCharacter (i, j)
    Count the amount of the specialCharacter appearing in all candidateStrings and STORE to appearRate(i, j)
  END FOR
END FOR
FOR EACH candidateString(i)
  SORT the specialCharacter(i, j) in the candidateString by appearRate (i, j) into descending order
  IF (the first character in the candidateString can't be found in a blackList)
    ADD the character to the blackList
  END IF
END FOR
```

3.3 An Auto-tuning Mechanism

Generating an ideal constraining rule is a critical technique for mitigating injection flaws. Ideal sanitizing input depends on achieving the right balance between false positives and false negatives. Traditional methods usually use a comprehensive constraining rule to strictly sanitize input. It may cause improper limits of input characters because of some false positives. A looser rule may lower false positives than a generic rule but it may induce false negatives. False negatives may render security risks. Thus, from a security defense viewpoint, the least false negatives should have a higher priority than the false positive and it follows that, in general, the false negative is zero. Due to the existence of the possibility of false positives and false negatives, the identification process is subject to human assistance which may include user decisions.

Although a blacklist is likely to support more elastic input data than a whitelist, it may induce more false negatives. The blacklist may be quite tempting so it is an unrealistic idea assuming that all the variations of malicious injection had been known. Thus a validation based on blacklist should prefer assuring of acceptable false negative, and do everything possible to reduce false positive. Our auto-tuning approach is intended to automatically achieve the right balance between false positives and false negatives. It guarantees same number of false negatives as well as reduces more false positives while sanitizing inputs.

False positives may cause improper limits of input characters. There are some normal input including special characters such as compound name (jan-min including "−") and domain name (www.ydu.edu.tw including "."). Adding these special characters in the blacklist must induce many false positives, on the contrary, removing these special characters must render many false negatives. While inspecting some malicious injection strings, we find that most special characters in the strings don't appear alone. Thus we can pretend that it is a normal string if only one type of special character appearing in an injection string. In the algorithm1, generating a meta-program, we can let the legalSpecialCharCount equal to 1 for effectively lowering false positives.

The known malicious injection string must be detected and removed by the sanitizing agent. That is to say, there are normal data and new malicious injection strings will be kept in the Web access log of the protected Web server. The new malicious injection strings can be quickly filtered and categorized according to attack types and then add to the testing pattern table. Finally, new constraining rules can be generated to protect against new injection attack. Thus, the auto-tuning mechanism can help generating ideal constraining rules to achieve the right balance between false positives and false negatives.

4 System Implementation

To verify the effectiveness of our scheme, we implement an auto-tuning sanitizing system and present its architecture diagram and interactions between each component in Figure 1. The system consists of three main components: sanitizing agent, injection vulnerability analyzer and injection pattern generator. The injection vulnerability analyzer is responsible for finding injection flaws and generating a vulnerable injection point table [13]. The injection pattern generator is dedicated to organizing new injection patterns by analyzing the navigational information kept in the Web access log. The sanitizing agent is capable of help validating input. The sanitizing agent is allocated in front of the protected Web servers. All HTTP\HTTPS requests to the protected Web servers are routed through the sanitizer that can either deal with the request itself or pass the request partially to Web servers. After passing all check, the requests are forwarded to the Web server. The sanitizing agent deals with both the requests coming from client and the response pages coming from the Web server and then forward to server\clients. It can help validating input via meta-programs. Any malicious injections must be blocked by the sanitizing agent. The meta-programs can be translated by a code generator named translator. Each parameter of the same URL in the vulnerable injection point table can generate a meta-program to constrain inputs. Furthermore, the meta-program only has sanitizing functionality so it can't replace original Web application. Thus after completing constraining work, the http request needs to be redirected to original program to obtain prime service. In general, the known malicious injection string must be detected and removed by the sanitizing agent. That is to say, there are only normal data and new malicious injection strings will be kept in the

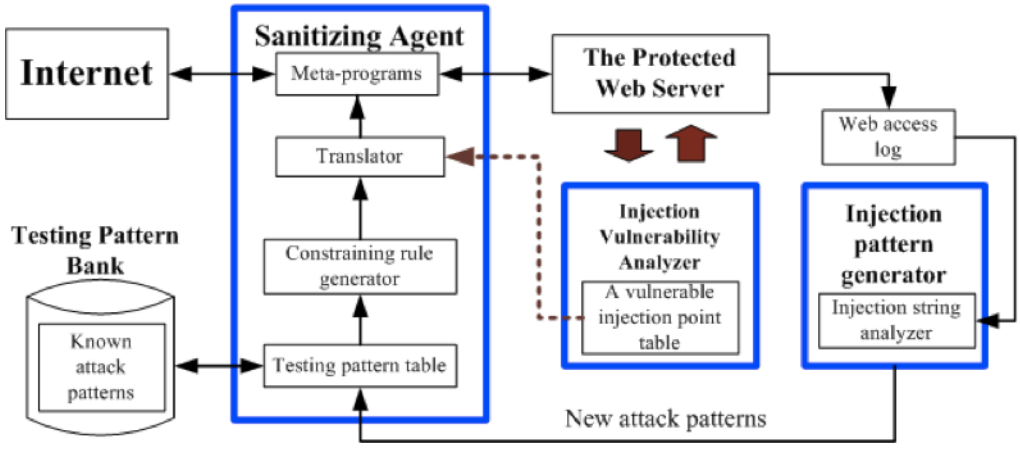


Figure 1: A architecture diagram of the auto-tuning sanitizing system

Web access log of the protected Web server. The injection pattern generator can analyze the Web access log and generate various injection patterns. These patterns can be categorized according to attack types and then kept in the testing pattern table. The constraining rule generator is capable of generating various ideal constraining rules according to testing pattern table. The detection accuracy of a sanitizing agent is dependent on constraining rules. The constraining rule generator may automatically organize new constraining rules while new injection patterns added in the testing pattern table. Thus the sanitizing rule can automatically adjust itself to constrain new malicious injection.

In Figure 1, the solid lines show the process of the auto-tuning mechanism. The sanitizing system can achieve the right balance between false positives and false negatives through auto-tuning mechanism. A false positive occurs when the normal input data is mistakenly blocked by a sanitizer, while a false negative occurs when the sanitizer cannot constrain a malicious injection. From a security defense viewpoint, the least false negative should have a higher priority than the false positive and it follows that, in general, the false negative is zero. A looser constraining rule may induce more false negatives. Thus some malicious injection strings will pass sanitizer and are kept in Web access log. These injection strings will be translate to new injection patterns by the injection pattern generator and then sent to testing pattern table. Next the constraining rule generator will organize stricter constraining rule to lower false negatives. Above processes will recursively go on until none of false negative. Therefore the auto-tuning sanitizing system can automatically adjust itself to effectively protect against new injection attacks.

We created an experimental website having SQL-injection and XSS vulnerabilities. The experimental website included 13 Web pages, 47 injection points, and 11 vulnerable injection points. We used the experimental website to assist in fine-tuning the detection accuracy of the injection vulnerability analyzer. To emphasize the importance of individual constraining rules, some injection points have been designated as special cases having only one specific flaw. For example, after the vulnerability assessment, we discovered that the third injection point of the experimental Web site only has XSS injection vulnerability and the fourth injection point has SQL vulnerability. This is then considered as a false positive if some pattern designed to locate SQL injection is filtered on the third injection point. We also chose six web applications from the National Vulnerability Database to enrich the evaluation of the injection vulnerability analyzer and the auto-tuning system [14]. After the injection vulnerability analyzer finishes the process, the detailed information about all the programs used in protected the Web site is presented in Table 2.

All primary tests were performed on an experimental Website including some client-side Web pages and vulnerable Web applications. The tests relevant to verifying the effectiveness of the sanitizer were divided into two phases. In the first phase, the injection vulnerability analyzer began to directly inspect vulnerability to create a vulnerable injection point table. In the second phase, at first the meta-program in the sanitizing agent adopted a looser constraining rule to verify the effectiveness of the auto-tuning mechanism. All known injection attack patterns were used as input. After completing the auto-tuning process, the false negative of the system is zero and the false positive is at strict minimum.

5 Experimental Evaluation

The effectiveness of the auto-tuning sanitizing system should be verified via two phases. One is to evaluate the detection accuracy of the system and next to show the auto-tuning mechanism can automatically adjust constraining rule to effectively sanitizing new injection patterns. There are none standard experimental data for verifying the performance of the system. In order to be sure of specific vulnerability, we need to create a testing

Table 2: The detail information about all programs used in protected website

Application Name/CVE#	Web page amount	Injection points amount	Vulnerable injection point amount	Vulnerable page Name	Flaw type
Experimental program	13	47	11	bbs.php	SQL injection/XSS
CVE-2010-0122	72	324	5	add_user.php	SQL injection
CVE-2009-4669	748	2947	2	Login.php	SQL injection
CVE-2009-4595	30	22	2	index.php	SQL injection
CVE-2010-1742	13	15	1	projects.php	XSS
CVE-2009-4456	9	10	1	news_detail.php	SQL injection
CVE-2009-3716	7	8	2	admin_login.php	SQL injection

Table 3: The constraining rule for various testing pattern sub-banks

Name	Description	constraining Rule_Name	Rule_Expression	String length of Rule_Expression
Testing bank 1	Sql_pattern_basic	Sanitizing_Sql_Rule 1	=;’	3
Testing bank 2	Sql_pattern_rich	Sanitizing_Sql_Rule 2	=%>(-*\’@:	10
Testing bank 3	XSS_pattern_basic	Sanitizing_XSS_Rule 1	<%	2
Testing bank 4	XSS_pattern_rich	Sanitizing_XSS_Rule 2	<%&(/=;\	8
Testing bank 5	Sql_pattern_rich & XSS__pattern_rich	Sanitizing_Sql&XSS_ Rule	’%>;-,=_/<(@:&	14
Testing bank 6	All malicious injection patterns	Sanitizing_all_Rule	=;_’>%/<(@:&\-|!.+	18

pattern bank including various testing patterns, relevant vulnerability types and comments. Each testing pattern should be meticulously designed to get expected output which can be clearly identified. AT first we had collected 1000 experimental data in a testing pattern bank. They came from access logs of websites which were scanned by Web vulnerability scanners or Web sites describing cheat sheet about injection attack [4, 20]. In Table 3, we present some testing pattern sub-banks for example. Each sub-bank is a part of the testing pattern bank and composed of patterns having specific types of vulnerability. The purpose of generating various sub-banks is to generate different constraining rules for specific injection flaws. For example, the Testing Bank1 only includes some popular SQL injection strings (10 classic patterns) and the Testing Bank2 put all SQL injection strings (453 patterns) together. The Testing Bank3 only includes some popular XSS injection strings (10 classic patterns) and the Testing Bank2 put all XSS injection strings (350 patterns) together. Others are the other types of malicious injection strings. In Table 3, we can find that the length of Rule_Expression of the Sanitizing Sql_Rule 2 is bigger than the length of Rule_Expression of the Sanitizing Sql_Rule 1 and the length of Rule_Expression of the Sanitizing all_Rule is biggest. It implies that the length of Rule_Expression may depend on the quantity and types of the injection strings.

We try to show that the detection accuracy of the sanitizer is dependent on the constraining rule. That is to say, the ideal constraining rules produced by the sanitizer can lower errors (false positives and false negatives) in comparison with the generic rules (generic whitelist or generic blacklist). From a security defense viewpoint, the least false negatives should have a higher priority than the false positive and it follows that, in general, the false negative is zero. In general, a useful sanitizing method must ensure that the false negative is zero. Thus the effectiveness of various sanitizing methods can be simplified to which having minimum false positive. A false positive occurs while these normal strings are mistakenly limited by a constraining rule. In order to be sure of false positive, we need to add some normal input strings (10 classic patterns) including at least one special character that appears only the first time in a testing pattern bank. All relevant parts of the testing results are presented as follows.

Table 4 and Table 5 show the amount of false negative and false positive induced by different constraining rules. A more flexible rule may induce less false positives than a generic rule, made apparent by the fact that the injection point only has single one specific flaw. For example, Table 4 shows that validation of using Sanitizing_ Sql_Rule 1 will only render less false positives in comparison with using Sanitizing_Sql_Rule 2 for the CVE-2009-4456.

From the principle of validating input and the experimental results we can conclude the following rules.

Table 4: The partial results of system training and detection experiment

CVE #	Program Name	Threat type	Sql_Rule 1+ #	Sql_Rule 1− #	Sql_Rule 2+ #	Sql_Rule 2− #
Experimental Web site	TestingInjection	SQL/XSS	1	1	2	1
CVE-2010-0122	timeclocksoftware	SQL	1	1	2	0
CVE-2009-4669	RoomPHPlanning	SQL	1	1	2	0
CVE-2009-4595	PHP Inventory	SQL	1	1	2	0
CVE-2009-1742	Scratcher	XSS	N/A	N/A	N/A	N/A
CVE-2009-4456	Green Desktiny	SQL	1	0	2	0
CVE-2009-3716	MCshoutbox	SQL	1	1	2	0

PS:
Sql_Rule 1+ #: the false positive amount of rending by the Sanitizing_SQL Rule 1
Sql_Rule 1− #: the false negative amount of rending by the Sanitizing_SQL Rule 1

Table 5: The partial results of system training and detection experiment

CVE #	Threat type	Sql& XSSRule+ #	Sql& XSS_Rule− #	All_Rule+ #	All_Rule− #
Experimental Web site	SQL/XSS	2	0	2	0
CVE-2010-0122	SQL injection	2	0	2	0
CVE-2009-4669	SQL injection	2	0	2	0
CVE-2009-4595	SQL injection	2	0	2	0
CVE-2009-1742	XSS	2	0	2	0
CVE-2009-4456	SQL injection	2	0	2	0
CVE-2009-3716	SQL injection	2	0	2	0

Table 6: Summary of system sanitizing effectiveness

CVE #	A: Vulnerable injection point amount (before protection)	B: Vulnerable injection point amount (after protection)	Protection effectiveness (A-B/A)
Experimental Web site	11	0	100%
CVE-2010-0122	5	0	100%
CVE-2009-4669	2	0	100%
CVE-2009-4595	2	0	100%
CVE-2010-1742	1	0	100%
CVE-2009-4456	1	0	100%
CVE-2009-3716	2	0	100%

The maximum amount of the false positives for a generic blacklist will equal to the amount of all special characters appearing in the Testing Bank 6 (i.e. greater than 18). The maximum amount of the false positives for a general whitelist will equal to the amount of all normal input strings including special characters (i.e. greater than 18). Table 5 shows that the maximum amount of the false positives for the sanitizing system equals to 2 while false negative is zero if the auto-tuning mechanism automatically adjusts the value of legalSpecialCharCount to 1. The results show that the sanitizing system can effectively lower flase positive. That is to say, the sanitizing system renders fewer false positives, compared to other systems using general whitelist or blacklist.

In order to show that the auto-tuning mechanism can automatically adjust constraining rule to effectively sanitizing new injection patterns, a looser constraining rule, such as Sanitizing_Sql_Rule 1, was used in meta-program at first. And then injection attack tools will launch injection attack to the vulnerable web applications listed in Table 2.

A looser constraining rule may induce more false negatives. Thus some malicious injection strings will pass sanitizer and are kept in Web access log. These injec-

tion strings will be translate to new injection patterns by the injection pattern generator and then sent to testing pattern table. Next the constraining rule generator will organize stricter constraining rule to lower false negatives.

Finally, we need to verify the effectiveness of the sanitizing agent for protecting against injection attacks. Table 6 presents the amount of vulnerable injection points before and after the protection of our system. For each vulnerable website, the amount of vulnerable injection points after protection is zero. The results prove that our sanitizer can effectively protect against injection attacks caused by improper input validation.

6 Conclusion

Vulnerable websites with the injection flaws are being attacked and damaged every day. Injection flaws are increasingly vulnerable and protecting them requires a system that can both ensure compliance today and meet the evolving needs of an organization for tomorrow. To meet the challenge, organizations should continue to be diligent by regularly performing vulnerability scanning and penetration testing. Unless a web application has a strong, centralized mechanism for validating all input from HTTP requests (and any other sources), injection flaws are very likely to exist. Therefore, organizations should select and deploy a system providing rapid protection to close the vulnerability gap, with minimal impact on operations. In this paper, we come up with a proposal that can be used as compensating controls to protect web applications while vulnerabilities exist and patching is not an immediate option. To improve the risk of injection flaws and reduce unnecessary limitation of input characters, we design an auto-tuning system to help validating input for each vulnerable injection point. The experimental results show that the system can effectively protect against injection attacks and lower false positives while compared with traditional methods.

Acknowledgments

This work was supported by the National Science Council of Taiwan under grants NSC 100-2218-E-412-001. I gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Antivirus Solutions, http://en.wikipedia.org/wiki/Antivirus_software, accessed on April 7, 2012.

[2] J. M. Chen, "An improved sanitizing mechanism based on heuristic constraining method," *Advanced Research on Electronic Commerce, Web Application, and Communication*, Communications in Computer and Information Science, vol. 144, pp.153-159, 2011.

[3] S. Dharmapurikar, P. Krishnamurthy, T. Sproull, J. D. Lockwood, "Deep packet inspection using parallel bloom filters", in *Proceedings of the 11th Symposium for High Performance Interconnect*, pp.44-51, 2003.

[4] Ferruh, SQL Injection Cheat Sheet, http://ferruh.mavituna.com/sql-injection-cheatsheet-oku/, accessed on June 14, 2010.

[5] M. Fisk and G. Varghese, "Fast content-based packet handling for intrusion detection", *UCSD Technical Report CS2001-0670*, 2001.

[6] W. Halfond and A. Orso, "Combining static analysis and runtime monitoring to counter SQL injection attacks," in *Proceedings of the Third International ICSE Workshop on Dynamic Analysis (WODA'05)*, 2005.

[7] J. H. Hayes, A. J. Offutt, "Increased software reliability through input validation analysis and testing software reliability engineering", in *Proceedings of the 10th International Symposium on Software Reliability Engineering*, pp.199–209, 1999.

[8] Y. W. Huang, C. H. Tsa, T. P. Lin, S. K. Huang, D. T. Lee, S. Y. Kuo, "A testing framework for Web application security assessment," *Journal of Computer Networks*, vol.48, no.5, pp.739-761, 2005.

[9] IBM Rational Corp., Web Application Security Testing—App-Scan, http: //www-01.ibm.com/software/rational/offerings/websecurity/, accessed on Jan. 10, 2009.

[10] I. M. Kim, "Using Web application firewall to detect and block common web application attacks," *SAN Institute Technical Report*, 2011.

[11] J. C. Lin, J. M. Chen, "An automatic revised tool for anti-malicious injection," in *The Sixth IEEE International Conference on Computer and Information Technology*, pp.164, 2006.

[12] J. C. Lin, J. M. Chen and C. H. Liu, "An automatic mechanism for sanitizing malicious injection", in *Proceedings of the 9th International Conference for Young Computer Scientists*, pp.1470-1475, 2008.

[13] J. C. Lin, J. M. Chen, and H. K. Wong, "An automatic meta-revised mechanism for anti-malicious injection," in *Proceedings of Network-Based Information Systems*, LNCS 4658, pp.98-107, 2007.

[14] NVD (National Vulnerability Database), http: //nvd.nist.gov/, accessed on June 10, 2010.

[15] Open Source Web Application Firewall: ModSecurity, http://www.webresourcesdepot.com/open-source-we-application-firewall-modsecurity/, accessed on Jan. 15, 2009.

[16] OWASP, WebScarab Project, http: //www.owasp.org/webscarab/, accessed on Jan. 18, 2009.

[17] OWASP, http://www.owasp.org/index.php/Web_Application_Firewall, accessed on Jan. 11, 2010.

[18] OWASP, https://www.owasp.org/index.php/ Injection_Flaws, accessed on Jan. 11, 2011.

[19] J. C. Rabek, R. I. Khazan, S. M. Lewandowski, R. K. CunninghamRabek, "Detection of injected, dynamically generated, and obfuscated malicious code," *Defense Advanced Project Agency (DARPA)*, Copyright Association for Computing Machinery, ACM, 2003.

[20] RSnake, XSS (Cross Site Scripting) Cheat Sheet, http://ha.ckers.org/xss.html, accessed on Jun 12, 2010.

[21] A. Salem, "Intercepting filter approach to injection flaws," *Journal of Information Processing Systems*, vol. 6, no. 4, pp.563–574, 2010.

[22] Sanctum Inc., AppShield white paper, http://www.sanctuminc.com/, accessed on Jan. 11, 2009.

[23] SPI Dynamics, Web Application Security Assessment, SPI Dynamics Whitepaper, http://www.spidynamics.com/, accessed on Jan. 20, 2009.

[24] Z. Radwan, C. Gaspard, A. Kayssi, and A. Chehab, "Policy-driven and content-based Web services security gateway", *International Journal of Network Security*, vol. 8, no. 3, pp. 253–265, May 2009.

Jan-Min Chen received the Ph.D. degree in the Department of Computer Science and Engineer from Tatung University, Taiwan, in 2010. He is currently an assistant professor in the Department of Information Management at Yu Da University, Taiwan. His research interests include computer network security, computer network management and web application security. His email is ydjames@ydu.edu.tw.

Group Authentication and Group Key Distribution for Ad Hoc Networks

Feng Wang[1], Chin-Chen Chang[2,3] and Yeh-Chieh Chou[2]
(Corresponding author: Chin-Chen Chang)
Department of Mathematics and Physics, Fujian University of Technology[1]
Fuzhou, Fujian, 350118, China
Department of Information Engineering and Computer Science, Feng Chia University[2]
No. 100, Wenhwa Rd., Seatwen, Taichung, Taiwan 40724, R.O.C.
Department of Computer Science and Information Engineering, Asia University[3]
500, Lioufeng Rd., Wufeng, Taichung, Taiwan 41354, R.O.C.
(Email: alan3c@gmail.com)
(Received Oct. 3, 2014; revised and accepted Nov. 15, 2014)

Abstract

Group authentication and group key distribution ensure the security of group communication. Most existing schemes of group authentication and group key distribution need the assistance of a group manager. However, deciding upon a group manager can be difficult work for some practical applications, especially in an Ad Hoc network. Therefore, we proposed a group authentication and group key distribution scheme that does not require a group manager. Our proposed scheme is an identity-based scheme based on bilinear pairing. In our proposed scheme, any user can easily generate a group for communication purposes. All or part of a group can authenticate each other and obtain a group key without foreknowledge or limiting the number of individuals attending the communication session. Any group member can join or quit the group communication securely in the duration of the meeting. Our proposed scheme requires little communication and computation cost and is resistant to common attacks. Furthermore, in order to take full advantage of the properties of computing ability, which can differ within Ad Hoc networks, our proposed scheme can designate the user with the greatest computing ability to distribute the group key.

Keywords: Ad Hoc networks, bilinear pairing, group authentication, group key distribution

1 Introduction

In recent years, Group communication [3, 16] has become more and more popular in many applications. It involves many-to-many communication, in contrast to the one-to-one or one-to-many communication forums in conventional communication. With this kind of communication, several members of a group can exchange messages to each other securely. To achieve this goal, mutual authentication and sharing of a session key among the group members takes place. The properties of group communication are as follows: 1) The communication users in the group must belong to the same group, 2) the session key sharing among group members needs to be the same, and 3) only group members can get the transmitted message from the group communication.

There are two group communication models for different applications. One group communication model, such as the Wireless Sensor Network [10], only requires group members to authenticate each other. Under the conventional authentication scheme, if there are n members in the group, the user needs to perform authentication for the other users that belong to the group $n-1$ times, for which the time complexity is $O(n)$. In 2013, Harn [7] proposed a group authentication scheme based on Shamir's secret sharing method [17], which facilitates authentication for all users in the group with only a one-time interaction, for which the time complexity is $O(1)$. Under Harn's scheme, a group manager (GM) first registers as a user. Then the users can authenticate each other without the assistance of the GM if they know the number of participants to authenticate.

In the other group communication model, users need to share a session key, such as in the case of a group conference. There are two kinds of methods for sharing a session key in group communication [5]. One is the group key agreement protocol [8], and the other is group key distribution protocol [19]. With regard to group key agreement protocol, all members in the group will consult together to determine and distribute the session key, which requires several rounds of interaction. Although there are some one round group key agreement proto-

cols [22, 23], the group keys generated in those protocols are for Asymmetric cryptosystem, which is not suit for a large number of data encryption. Under group key distribution protocol, a group manager decides the session key. Generally, the latter is more efficient than the former, because in the latter case, the group manager does most of the key distribution work.

However, the models we aforementioned are not suitable for the Ad Hoc network [11, 14] environment. This type of environment does not rely on pre-existing network architecture, and each node in the network has the capability to transmit the message to the other node. Predetermining a group manager (GM) is difficult in this scenario. Therefore, foreknowledge of the number of participants in Harn's scheme [7] or distribution of the session key with the help of the GM present inconveniences. Furthermore, a group member may want to join the group communication after the group communication has begun in some practical applications. However, we did not find a group authentication key setup scheme with a join phase in the process of communication in the reviewed literature.

To solve the problems mentioned above, we proposed an identity-based group authentication and key distribution scheme based on bilinear pairing. The main contributions can be summarized below.

1) Our proposed scheme doesn't need the selection of a group manager, and can designate the user with the most advanced computing ability to distribute group key, therefore, it is suitable for Ad Hoc networks.

2) Our proposed scheme separates the authentication phase from the key distribution phase for different applications.

3) A join phase and revocation phase are employed in our proposed scheme to enable group members to join or leave the meeting before or during the process.

4) Our proposed scheme requires little communication and computation cost. It only calls for two rounds of interaction in the authentication phase and one round of interaction in the key distribution phase. And it require less computation cost compared with Zhang et al.'s scheme [23].

5) Our proposed scheme can fulfill several security requirements, such as mutual authentication, consistency of group key, and perfect forward security. Moreover, the scheme can counteract several well-known attacks, such as impersonation attack, man-in-the-middle attack, and replay attack.

The rest of this paper is given as follows. The preliminaries are provided in Section 2, and we describe our proposed scheme in Section 3. The security analysis is given in Section 4. In Section 5, a comparison with other schemes is given. Lastly, Section 6 gives the conclusion.

2 Preliminaries

In this section, we review some preliminaries including bilinear pairing [15, 20, 21], Diffie-Hellman Assumption [6, 15, 20, 21], and Gentry and Ramzan's identity based multisignature scheme [6].

2.1 Bilinear Pairing

Let G_1 be additive group and G_2 be a multiplicative group with the same prime order q, while P is a generator of G_1 [15, 20, 21]. The map $e : G_1 \times G_1 \longrightarrow G_2$ is called a bilinear map if the following three properties are held:

1) Bilinear: For all $a, b \in Z_q^*$, the equation $e(a \cdot P, b \cdot P) = e(P, P)^{a \cdot b}$ is held.

2) Non-degenerate: $e(P, P) \neq 1$.

3) For any $P_1, P_2 \in G_1$, there is an efficient algorithm to compute $e(P_1, P_2)$.

2.2 Diffie-Hellman Assumption

1) Computational Diffie-Hellman assumption [6].
Given that $a \cdot P, b \cdot P \in G_1$ with $a, b \in Z_q^*$ is unknown, there is no probabilistic polynomial-time algorithm to compute $a \cdot b \cdot P \in G_1$.

2) Bilinear Diffie-Hellman assumption [15, 20, 21].
Given that $P, a \cdot P, b \cdot P, c \cdot P \in G_1$ with $a, b, c \in Z_q^*$ are unknown, there is no probabilistic polynomial-time algorithm to compute $e(P, P)^{a \cdot b \cdot c} \in G_2$ Note that if we know anyone among a, b, c, we can compute $e(P, P)^{a \cdot b \cdot c}$ easily. For example, if we know parameter a, then we can compute $e(P, P)^{a \cdot b \cdot c}$ easily by $e(P, P)^{a \cdot b \cdot c} = e(b \cdot P, c \cdot P)^a$.

2.3 Gentry and Ramzan's Identity-based Multisignature Scheme

A multisignature approach [9] means that there are several signers cooperatively to sign on the same message to generate a single and valid signature, then the verifier can verify the signature using the public key of all of the signers. To combine the multisignature and identity-based cryptosystems [18], Gentry and Ramzan [6] proposed an identity-based multisignature scheme using bilinear pairing in 2006. Their scheme is secure in the random oracle model under the computational Diffie-Hellman assumption. The scheme consists of five phases: setup phase, private key extraction phase, individual signing phase, aggregation phase, and verification phase, which are described in detail as follows.

Setup Phase. The private key generator (PKG) chooses an additive group G_1 and a multiplicative group G_2 with the same prime order q. This also includes an admissible bilinear map $e : G_1 \times G_1 \longrightarrow G_2$, an arbitrary generator P of G_1, and two hash functions

$H_1, H_2 : \{0,1\}^* \longrightarrow G_1$. Then the PKG picks a random number $s \in Z_q^*$ as the master secret key, then computes $P_{pub} = s \cdot P$ and publishes the parameters $(G_1, G_2, q, P, P_{pub}, e, H_1, H_2)$.

Private Key Extraction Phase. Given the user U_i's identity ID_i, the PKG picks its master secret key $s \in Z_q^*$ and computes $SK_i = s \cdot H_1(ID_i)$ as U_i's private key. Then it sends SK_i to U_i via a secure channel.

Individual Signing Phase. Given a message m, the user U_i picks a random number $r_i \in Z_q^*$. Then computes $R_i = r_i \cdot P$ and $\sigma_i = r_i \cdot H_2(m) + SK_i$. Afterward, the couple (R_i, σ_i) is U_i's individual signature of message m.

Aggregation Phase. Anyone who collected n users' individual signatures (R_i, σ_i) of the same message m, for $i = 1, 2, ..., n$, can generate the n users' multisignature (R, σ), where $R = \sum_{i=1}^{n} R_i$, $\sigma = \sum_{i=1}^{n} \sigma_i$.

Verification Phase. Upon receipt of the multisignature (R, σ), the verifier computes $Q = \sum_{i=1}^{n} H_1(ID_i)$ and checks if the equation $e(\sigma, P) = e(R, H_2(m) \cdot e(P_{pub}, Q))$ holds. If so, he/she accepts the multisignature; otherwise, he/she rejects it.

3 The Proposed Scheme

In this section, we propose a group authentication and group key distribution scheme for Ad Hoc networks which are based on bilinear pairing. The scheme can be divided into five phases; i.e., 1) the initialization phase, 2) the group authentication phase, 3) the group key distribution phase, 4) the join phase, and 5) the revocation phase. When the user wants to generate a group to transmit a message, he/she can use the group authentication phase to authenticate the users that belong to the group. Then, the user can use the group distribution phase to distribute the session key to each user. In addition, if there is a new group member who wants to join the communication during the process of the communication, he/she can execute the join phase. Finally, if there is a group member who wants to exit the communication, he/she can execute the revocation phase to release this group member.

3.1 The Initialization Phase

Before communicating with others, a user must perform this phase to obtain his/her private key. The PKG selects the system parameters. The user provides his/her identity to the PKG, and the PKG generates the user's private key and sends it to the user via a secure channel.

Step 1. (Set up) This is identical to the setup phase of Gentry and Ramzan's multisignature in Subsection 2.3, except the PKG chooses a symmetric encryption/decryption algorithm E/D and a group key space GK, and publishes the parameters (E, D, GK) also.

Step 2. (Private key extraction) This is identical to the private key extraction phase of Gentry and Ramzan's multisignature in Subsection 2.3.

3.2 The Group Authentication Phase

Suppose a user U_1 wants to generate a group with n users including him- or herself. He/She broadcasts the request. Let $U = \{U_1, U_2, ..., U_n\}$ denote n users, m denote the purpose, and T denote the current time. There are $t-1$ users who respond to the activity, denoted by $U_2, ..., U_t$ They can perform the following steps for authentication. We give an example for $t = 4$ to explain this phase in Figure 1 too.

Step 1. The initiator user U_1 first picks a random number $r_1 \in Z_q^*$, then computes $R_1 = r_1 \cdot P$, $h = H_2(m||U||T)$, and $\sigma_1 = r_1 \cdot h + SK_1$. After that, (R_1, σ_1, m, U, T) is broadcast to all n users.

Step 2. After the other users receive the message, each user $U_i, (i = 2, 3, ..., t)$ picks a random number $r_i \in Z_q^*$, then computes $R_i = r_i \cdot P$, $h = H_2(m||U||T)$, and $\sigma_i = r_i \cdot h + SK_i$. After that, the message (R_i, σ_i, m, U, T) is broadcast to all t users.

Step 3. For $i = 1, 2, 3, ..., t$, each user U_i computes $R = \sum_{i=1}^{t} R_i$, $\sigma = \sum_{i=1}^{t} \sigma_i$, and $Q = \sum_{i=1}^{t} H_1(ID_i)$. Then he/she checks to determine if the equation $e(\sigma, P) = e(R, h) \cdot e(P_{pub}, Q))$ holds. If so, the t users accept the procedure, and otherwise, terminate the procedure.

3.3 The Group Key Distribution Phase

Note that the users computing abilities are different from each other in Ad Hoc networks. Therefore, after having succeeded in the group authentication phase, the initiator user U_1 to designate one user $U_j, (1 \leq j \leq t)$ who has the greatest computing ability to distribute the group key. Without loss of generality, we assume that user U_1 performs the group key distribution work, and he/she distributes the group key in the following two steps. Furthermore, we give an example for $t = 4$ to explain this phase in Figure 2.

Step 1. The initiator, user U_1, picks a random number $gk \in GK$ as the group key. For $i = 2, 3, ..., t$, User U_1 computes $k_i = e(r_1 \cdot R_i, R_{[(i-1) \bmod (t-1)]+2})$, $gk' = U_1||T||gk$, and $c_i = E_{k_i}(gk')$ and broadcasts c_i to all other $t-1$ users.

Step 2. After receiving the message, for $i = 2, 3, ..., t$,

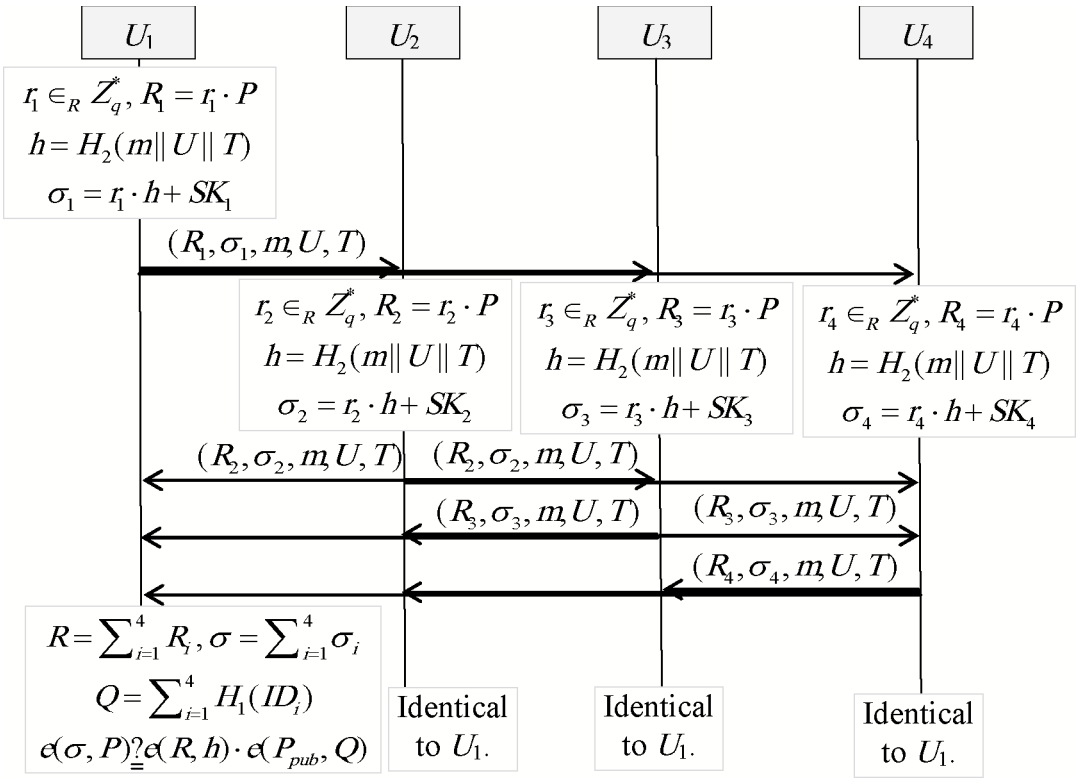


Figure 1: The proposed scheme

each user U_i computes

$$\begin{aligned} k_i' &= e(r_i \cdot R_1, R_{[(i-1) \bmod (t-1)]+2}), \\ gk_i' &= D_{k_i'}(c_i), \\ k_i'' &= e(r_i \cdot R_1, R_{[(i-3) \bmod (t-1)]+2}), \\ gk_i'' &= D_{k_i''}(c_{[(i-3) \bmod (t-1)]+2}). \end{aligned}$$

Then each user U_i checks to see if the equation $gk_i' = gk_i''$ holds and U_1, T are correct. If so, he/she can share the group key gk, and otherwise terminate the procedure. Note that gk, U_1, T satisfies $gk_i' = U_1||T||gk$ or $gk_i'' = U_1||T||gk$.

3.4 The Join Phase

Suppose that there is a user $U_j \in U$ who doesn't attend the group communication at the beginning of the communication for some reason, and he/she wants to join the communication of $\{U_1, U_2, ..., U_t\}$ during the process of the communication. The user U_j can execute the join phase and attend the communication without knowing the content of the previous communication. We describe this phase below and provide an example for U_5 attending the group communication of $\{U_1, U_2, U_3, U_4\}$ to explain this phase in Figure 3.

Step 1. User U_j picks a random number $r_j \in Z_q^*$ and then computes $R_j = r_j \cdot P$, $h_j = H_2(m||U||T||T_j)$, and $\sigma_j = r_j \cdot h_j + SK_j$. After that, this user broadcasts the message $(R_j, \sigma_j, m, U, T, T_j)$ to users $U_1, U_2, ..., U_t$, who have begun the communication, where T_j is the current time.

Step 2. For $i = 1, 2, ..., t$, each user U_i computes $h_j = H_2(m||U||T||T_j)$ and checks if the timestamp T_j is fresh and the equation $e(\sigma_j, P) = e(R_j, h_j) \cdot e(P_{pub}, H_1(ID_j))$ holds. If so, he/she accepts and performs Step 3; otherwise, he/she terminates the procedure.

Step 3. After that, user U_1 picks a new random number $gk_{new} \in GK$ as a group key. Then he/she computes $c = E_{gk}(U_1||T_j||gk_{new})$, $k_{2,j} = e(r_1 \cdot R_j, R_2)$, $c_{2,j} = E_{k_{2,j}}(U_1||T_j||gk_{new})$, $k_{t,j} = e(r_1 \cdot R_j, R_t)$, and $c_{t,j} = E_{k_{t,j}}(U_1||T_j||gk_{new})$ and broadcasts $c, c_{2,j}, c_{t,j}$ to all $t+1$ users.

Step 4. For $i = 2, 3, ..., t$, each user U_i computes $D_{gk}(c)$. Then user U_2 computes $k_{2,j}' = e(r_2 \cdot R_j, R_1)$ and checks whether the equation $D_{gk}(c) = D_{k_{2,j}'}(c_{2,j})$ holds. User U_t computes $k_{t,j}' = e(r_t \cdot R_j, R_1)$ and checks whether the equation $D_{gk}(c) = D_{k_{t,j}'}(c_{t,j})$ holds. User U_j computes $k_{t,j}'' = e(r_j \cdot R_t, R_1)$ and $k_{2,j}'' = e(r_j \cdot R_2, R_1)$ and checks to determine whether the equation $D_{k_{2,j}''}(c_{2,j}) = D_{k_{t,j}''}(c_{t,j})$ holds. If all of the equation holds and U_1, T is correct, the $t+1$ users can share the group key gk_{new}, and otherwise, terminate the procedure.

The Revocation Phase. If a user U_k who wants to leave the group communication, the remaining users

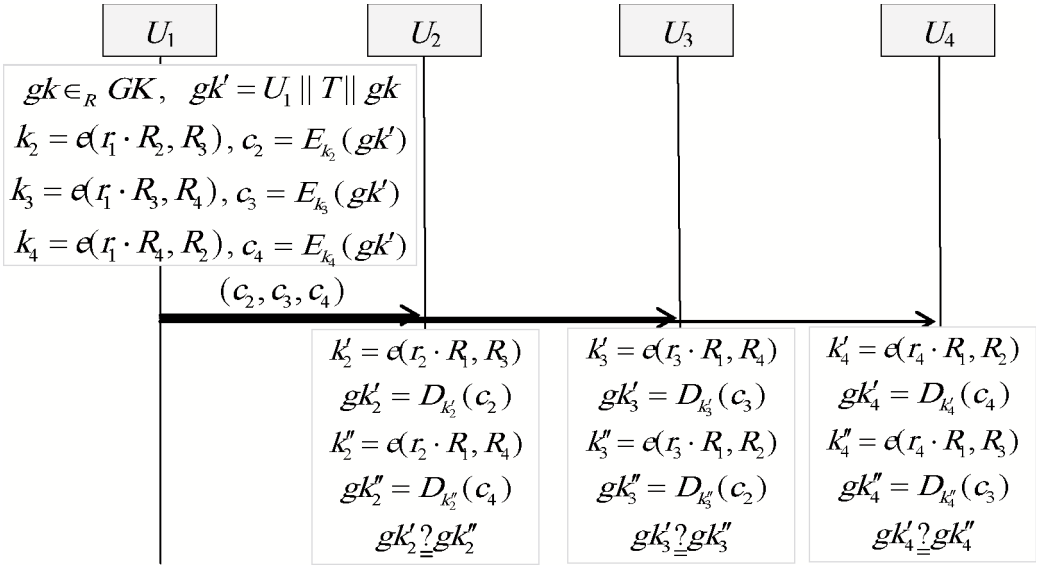


Figure 2: The example of group key distribution phase

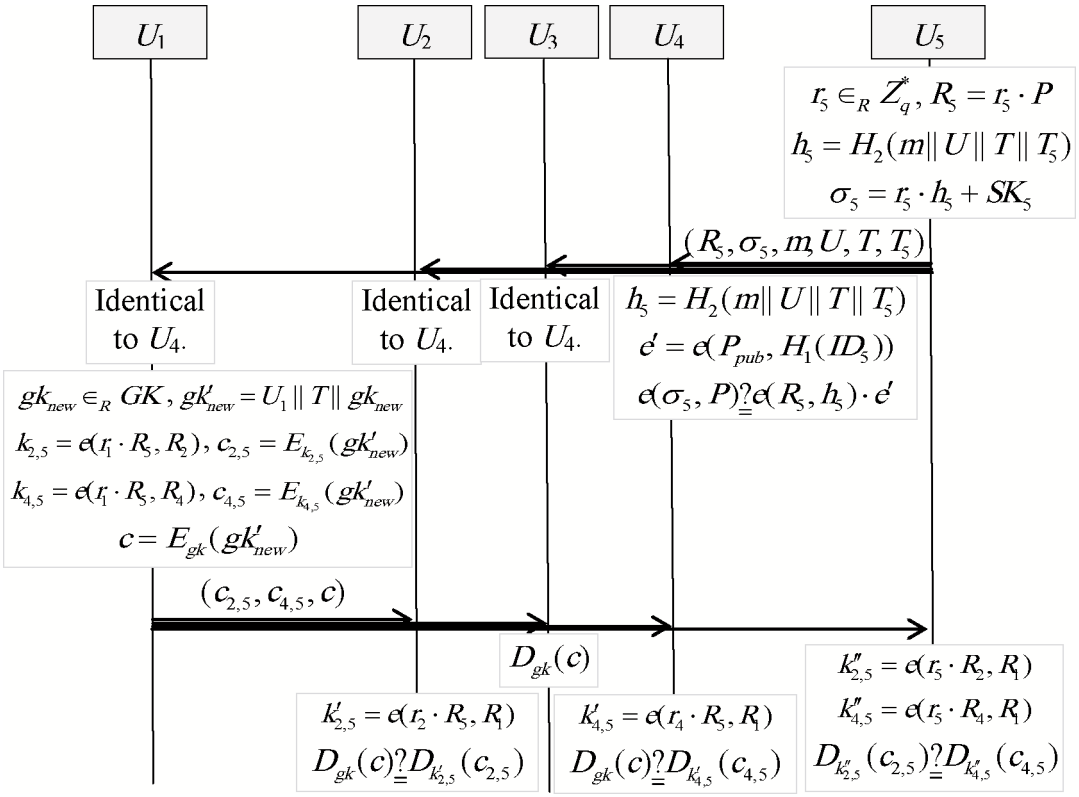


Figure 3: The example of join phase

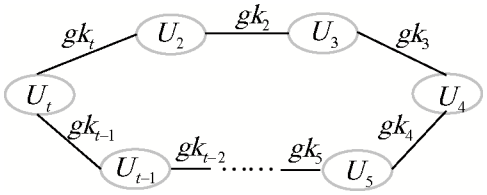


Figure 4: Ring generated by the authenticated group members

perform the group key distribution phase to reinstitute a new group key. Thus, user U_k cannot know the content of the remaining users' communication.

4 Security Analysis

In this section, we analyze several secure requirements that our proposed scheme possesses; i.e., mutual authentication, consistency of the group key, perfect forward security, withstanding the impersonation attack, withstanding the man-in-the-middle attack, and withstanding the replay attack.

4.1 Mutual Authentication

Mutual authentication means that each group member who attends the group authentication must authenticate the validity of all of the other users. The (R, σ) is actually $m||U||T$'s multisignature signed by users $U' = \{U_1, U_2, ..., U_t\}$ during Step 3 of the group authentication phase in our proposed scheme according to [6]. Each user U_i checks the validity of all the other users U' by verifying the multisignature. Any adversary cannot forge the multisignature (R, σ), according to Theorem 1 in [6]. Therefore, the scheme achieves mutual authentication.

4.2 Consistency of the Group Key

The consistency of the group key means that the group key obtained by each group member is identical. All of the authenticated group members (except the distributer) generate a ring described in Figure 4 in the group key distribution phase of our scheme. In Step 2 of the group key distribute phase, each member U_i can obtain two group keys, gk_i and $gk_{[(i-3) \bmod (t-1)]+2}$, which are equal to the value of one of the adjacent members in Figure 4. Therefore, our scheme can ensure the consistency of the group key by each member U_i checking the equality of gk_i and $gk_{[(i-3) \bmod (t-1)]+2}$. Furthermore, the group members can detect that the distributer allocates a different group key even if she/he colludes with one member U_k.

4.3 Perfect Forward Security

Perfect forward security refers to the inability of the adversary to obtain any previous group key, even if she/he knows all the participants private keys [2, 24]. For $i = 1, 2, ..., t$, if an adversary knows user U_i's private key SK_i and the interaction record including R_i and c_i, she/he cannot obtain the group key gk. The R_i is generated by $R_i = r_i \cdot P$, where r_i is selected randomly and kept secret by U_i; Therefore, the adversary cannot compute $k' = e(r_i \cdot R_1, R_{[(i-1) \bmod (t-1)]+2})$ and $k'' = e(r_i \cdot R_1, R_{[(i-3) \bmod (t-1)]+2})$, without knowing r_i, and hence cannot obtain group key gk by computing $D_{k'_i}(c_i)$ or $D_{k''_i}(c_{[(i-3) \bmod (t-1)]+2})$. Therefore, our scheme maintains perfect forward security.

4.4 Withstanding the Impersonation Attack

In Harn's group authentication scheme [7], there are two types of adversaries, including outside attackers and inside attackers. The group management generates a group with n members. The outside attacker tries to impersonate a valid group member to bypass the group authentication. The inside attacker is actually a group member who tries to obtain the secret information of the group.

In our scheme, there is no secret information of the group except each member's private key, so we consider the outside attacker only. However, without knowing user U_i's private key, anyone cannot forge the user U_i's signature. Therefore, it is impossible for anyone to impersonate a valid group member and pass the group authentication.

4.5 Withstanding the Man-in-the-Middle Attack

In the man-in-middle attack, attacker Eve interrupts, eavesdrops, and modifies the message between users Alice and Bob and builds a channel with each one. After that, Alice and Bob still believe that they are in direct communication with each other and in a private channel.

Fortunately, even attacker Eve can change the message R_i to R'_i, She still cannot achieve the purpose because the R'_i cannot pass the multisignature verification. Therefore, our scheme can resist the man-in-the-middle attack.

4.6 Withstanding the Replay Attack

Replay attack refers to the attempt by an adversary to imitate a group member in order to pass the group authentication by replaying the eavesdropped foregone message in group communication. In our scheme, a timestamp is added as a part of signed message in the group authentication phase. The user can resist the replay attack by checking whether the timestamp is fresh. This is the same as in the group key distribute phase. For this reason, our scheme can resist the replay attack.

5 Comparison

In this section, we give the comparison with Harn's group authentication scheme [7], Zhang et al.'s group key agree-

Table 1: Features comparison with the other schemes

Scheme	F1	F2	F3	F4	F5	F6	F7	F8
Harn's [7]	Y	N	Y	Y	-	Difficult	N	Shamir's secret sharing [17]
Zhang et al.'s [23]	N	N	N	N	N	Difficult	Y	CDH and k-BDHE [4]
Liu et al.'s [12]	Y	Y	N	Y	N	Difficult	N	Asmuth and Bloom's secret sharing [1]
Our	N	N	N	N	N	Easy	Y	Gentry and Ramzan's multisignature [6]

F1: Whether needs a group manager to setup the group.
F2: Whether needs a group manager to attend in the group authentication phase or the group key distribution phase.
F3: Whether needs to foreknow the number of members.
F4: Whether needs to limit the least number.
F5: Whether allows to be added or reduced the members in the process of group communication.
F6: Adding or revoking member from the group.
F7: Whether the scheme is an identity based scheme.
F8: What cryptography tool is based on.

Table 2: Efficiency comparison of group key generated with the other scheme

Scheme	Communication rounds	Computation efficiency	Type of generated group key
Zhang et al.'s [23]	1	8 pairing operations	Asymmetric cryptosystem
Liu et al.'s [12]	5	-	Symmetric cryptosystem
Our	1	2 pairing operations	Symmetric cryptosystem

ment protocol [23], and Liu et al.'s group key distribution scheme [12]. The features of those schemes are compared in Table 1. Our proposed scheme does not need a group manager to set up the group or attend the group authentication phase or group key distribution phase. In addition, it is not necessary to foreknow or limit the number of members who attend the group authentication or group key distribution in our proposed scheme. In our proposed scheme, anyone can initiate a group for communication easily. If a member of the group delays attendance of the group communication, he/she can join the group communication late without knowing the previous communication content. If there is a member who wants to quit the meeting, he/she can exit without knowing the later content. Furthermore, our proposed scheme is an identity-based scheme. Therefore, our proposed scheme is more flexible for practical application, especially in the Ad Hoc networks.

As for the efficiency of the group key generated in our scheme, we compare it in Table 2. Note that Harn's group authentication scheme does not give the algorithm for generating the group key, therefore we compare our scheme with Zhang et al.'s group key agreement protocol [23] and Liu et al.'s group key distribution scheme [12] only. Usually, a step of communication is more costly than a step of local computation [13]. Our scheme and Zhang et al.'s group key agreement protocol need 1 communication round compared with 5 communication rounds in Liu et al.'s group key distribution scheme, and therefore are more efficient. So we don't give the computation cost of Liu et al.'s group key distribution scheme. Each user in our scheme needs 2 pairing operations compared with 8 pairing operations in Zhang et al.'s protocol. Although the group key dealer needs $t-1$ pairing operations in our scheme, those operations can be pre-computed. Furthermore, the generated group key in our scheme is for symmetric cryptosystem, which is suit for a large number of data encryption than asymmetric cryptosystem generated in Zhang et al.'s protocol. Therefore, our scheme is more efficient than the other two schemes.

6 Conclusions

In this paper, we proposed a group authentication and key distribution scheme for Ad Hoc networks which is based on bilinear pairing. In our proposed scheme, any user can easily generate a group for communication without a group manager. All or part of the group members can complete the authentication and group key distribution without foreknowledge or limit to the number of members who attend the communication. Any group member can join or quit the group communication easily in the process of the communication without leaking the content of the communication. Our proposed scheme is an identity-based scheme, with little communication and computation cost, properties of mutual authentication, consistency of the group key and perfect forward security, and resistance to impersonation attack, man-in-the-middle attack, and replay attack. Furthermore, our scheme can designate the user with the greatest comput-

ing ability to distribute the group key, which is suitable for the property of computing power asymmetry in the Ad Hoc network environment.

References

[1] C. Asmuth and J. Bloom, "A modular approach to key safeguarding," *IEEE Transactions on Information Theory*, vol. IT-29, no. 2, pp. 208-210, 1983.

[2] A. K. Awasthi and S. Lal, "A remote user authentication scheme using smart cards with forward secrecy," *IEEE Transactions on Consumer Electronics*, vol. 49, no. 4, pp. 1246-1248, Nov. 2003.

[3] B. Bruhadeshwar and S. S. Kulkarni, "Balancing revocation and storage trade-offs in secure group communication," *IEEE Transactions on Dependable and Secure Computing*, vol. 8, no. 1, pp. 58-73, Feb. 2011.

[4] D. Boneh, X. Boyen, E. J. Goh, "Hierarchical identity based encryption with constant size ciphertext," in *Proceedings of EUROCRYPT 2005*, Aarhus, Denmark, pp. 440-456, May, 2005.

[5] C. Boyd, "On key agreement and conference key agreement," in *Proceedings of Second Australasian Conference on Information Security and Privacy*, Sydney, Australia, pp. 294-302, July 1997.

[6] C. Gentry and Z. Ramzan, "Identity-based aggregate signatures," in *Proceedings on 9th International Conference on Theory and Practice in Public-Key Cryptography (PKC'2006)*, New York, USA, vol. 3958, pp. 257-273, Apr. 2006.

[7] L. Harn, "Group authentication," *IEEE Transactions on Computers*, vol. 62, no. 9, pp. 1893-1898, 2013.

[8] D. He, J. Chen and J. Hu, "A pairing-free certificateless authenticated key agreement protocol," *International Journal of Communication Systems*, vol. 25, no. 2, pp. 221-230, 2012.

[9] S. K. H. Islam and G. P. Biswas, "Certificateless short sequential and broadcast multisignature schemes using elliptic curve bilinear pairings," *Journal of King Saud University - Computer and Information Sciences*, vol. 26, no. 1, pp. 89-97, Jan. 2014.

[10] C. T. Li, M. S. Hwang and Y. P. Chu, "An efficient sensor-to-sensor authenticated path-key establishment scheme for secure communications in wireless sensor networks", *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 8, pp. 2107-2124, Aug. 2009.

[11] C. T. Li, M. S. Hwang, "A lightweight anonymous routing protocol without public key en/decryptions for wireless Ad hoc networks", *Information Sciences*, vol. 181, no. 23, pp. 5333V5347, Dec. 2011.

[12] Y. J. Liu, L. Harn and C. C. Chang, "An authenticated group key distribution mechanism using theory of numbers," *International Journal of Communication Systems*, vol. 27, no. 11, pp. 3502-3512, Nov. 2014.

[13] W. Mao, Modern Cryptography: Theory and Practice, Publishing House of Electronic Industry, Beijing, China, 2004.

[14] S. A. E. Mohamed, "Secure position verification approach for wireless Ad-hoc networks," *International Journal of Network Security*, vol. 15, no. 4, pp. 248-255, July 2013.

[15] J. Nam, Y. Lee, S. Kim and D. Won, "Security weakness in a three-party pairing-based protocol for password authenticated key exchange," *Information Sciences*, vol. 177, no. 6, pp. 1364-1375, Mar. 2007.

[16] P. Sakarindr and N. Ansari, "Survey of security services on group communications," *IET Information Security*, vol. 4, no. 4, pp. 258-272, Dec. 2010.

[17] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612-613, Nov. 1979.

[18] A. Shamir, "Identity based cryptosystems and signature schemes," in *Proceedings of CRYPTO'84 on Advances in Cryptology*, Santa Barbara, California, U.S.A., vol. 196, pp. 47-53, Aug. 1984.

[19] A. T. Sherman and D. A. McGrew, "Key establishment in large dynamic groups using one-way function trees," *IEEE Transactions on Software Engineering*, vol. 29, no. 5, pp. 444-458, 2003.

[20] Y. L. Tian, C. G. Peng, and J. F. Ma, "Publicly verifiable secret sharing schemes using bilinear pairings," *International Journal of Network Security*, vol. 14, no. 3, pp. 142-148, May 2012.

[21] L. H. Wang, J. Shao, Z. F. Cao, M. Mambo, A. Yamamura, and L. C. Wang, "Certificate-based proxy decryption systems with revocability in the standard model," *Information Sciences*, vol. 247, pp. 188-201, Oct. 2013.

[22] Q. Wu, Y. Mu, W. Susilo, B. Qin and J. Domingo-Ferrer, "Asymmetric group key agreement," in *Proceedings of EUROCRYPT 2009*, Cologne, Germany, pp. 153-170, Apr. 2009.

[23] L. Zhang, Q. Wu, B. Qin and J. Domingo-Ferrer, "Provably secure one-round identity-based authenticated asymmetric group key agreement protocol," *Information Sciences*, vol. 181, no. 19, pp. 4318-4329, 2011.

[24] L. Zhang, Q. Wu, B. Qin, J. Domingo-Ferrer, and ú. González-Nicolás, "Asymmetric group key agreement protocol for open networks and its application to broadcast encryption," *Computer Networks*, vol. 55, pp. 3246-3255, 2011.

Feng Wang was born in Shandong province, China, in 1978. He received his B.S. degree in Mathematics from Yantai Normal University (now named Ludong University), Yantai, in 2000 and the M.S. degree in Applied Mathematics from the Guangzhou University, Guangzhou, in 2006. Currently, he is a Lecturer in the Department of Mathematics and Physics at Fujian University of Technology and a visiting scholar in Department of Information Engineering and Computer Science at Feng Chia University. His research interests

include computer cryptography and information security.

Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. His first degree is Bachelor of Science in Applied Mathematics and master degree is Master of Science in computer and decision sciences. Both were awarded in National Tsing Hua University. Dr. Chang served in National Chung Cheng University from 1989 to 2005. His current title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from Feb. 2005. Prior to joining Feng Chia University, Professor Chang was an associate professor in Chiao Tung University, professor in National Chung Hsing University, chair professor in National Chung Cheng University. He had also been Visiting Researcher and Visiting Scientist to Tokyo University and Kyoto University, Japan. During his service in Chung Cheng, Professor Chang served as Chairman of the Institute of Computer Science and Information Engineering, Dean of College of Engineering, Provost and then Acting President of Chung Cheng University and Director of Advisory Office in Ministry of Education, Taiwan. Professor Chang has won many research awards and honorary positions by and in prestigious organizations both nationally and internationally. He is currently a Fellow of IEEE and a Fellow of IEE, UK. On numerous occasions, he was invited to serve as Visiting Professor, Chair Professor, Honorary Professor, Honorary Director, Honorary Chairman, Distinguished Alumnus, Distinguished Researcher, Research Fellow by universities and research institutes. His current research interests include database design, computer cryptography, image compression and data structures.

Yeh-Chieh Chou was born in Taichung, Taiwan, in 1990. He received his Bachelor's Degree in Information Engineering form Chang Jung Christian University, Tainan. Currently, he is the second grade student for master's program of Department of Information Engineering in Feng Chia University. His research interests include computer cryptography and information security.

Internet of Things: Hotspot-based Discovery Service Architecture with Security Mechanism

Degang Xu[1,2], Zhao Wu[1], Zhongbo Wu[1], Qilin Zhang[1], Leihua Qin[2], Jingli Zhou[2]
(Corresponding author:Degang Xu)

School of Mathematics and Computer Science, Hubei University of Arts and Science[1]
No. 296, Longzhong Road, Xiangyang, 441053, China
School of Computer, Huazhong University of Science and Technology[2]
(Email: dgx.hust@gmail.com)
(Received Mar. 25, 2013; revised and accepted Nov. 6, 2014)

Abstract

In the emerging Internet of Things (IoT), as a means to fulfill item-level lookup, apart from the functional requirements with high performance and robustness, lookup service or discovery service playing a critical role should meet security and privacy requirements. However, existing lookup service and discovery service of IoT mainly rely on a centralized or a chain-style framework, have some drawbacks or bottlenecks to prevent them from being widely adopted, while the issue of locating hotspot resource has received much less attention, as well as the item-level lookup service is still missing. Therefore, we first present a distributed hotspot-based discovery service architecture based on double-Chord-ring for IoT, and then give its framework and some relevant mechanisms. Here we primarily focus on the goals of meeting security and privacy requirements. Additionally, we further discuss and analyze our solution.

Keywords: Internet of Things, Object Name Service, Discovery Service, Object Discovery Service, Security Mechanism

1 Introduction

The Internet of Things (IoT), an emerging global Internet-information architecture, has the purpose of providing an IT-infrastructure facilitating the exchanges of goods and services in global supply chain networks in a secure and reliable manner [3, 31], where lookup service or discovery service plays a critical role. Therefore, as an essential and critical component for a variety of application scenarios of the IoT (specifically, the EPCglobal Network, an industry proposal to build a global information architecture for objects carrying RFID tags with Electronic Product Codes (EPC)), lookup service should take some measures to enhance the security and privacy of the architecture.

Two of the key components of IoT lookup service architecture required to implement track and trace capabilities are the Object Name Service (ONS) and the Discovery Service (DS) envisaged to provide pointers to multiple providers of information across a supply chain not only the manufacturer. In the EPCglobal architecture [16], the most influential architecture and potential future nucleus of IoT, DS is still in development, ONS only provides a pointer to the information service provided by the manufacturer of the object. Moreover, ONS [15] is based on the well-known Domain Name System (DNS), each query must start from Root ONS. Thus, the ONS will inherit all of the well-documented DNS weaknesses, such as the limited redundancy in practical implementations and the creation of single points of failure [29]. For ONS, this architecture will have a deep impact on the reliability, security and privacy of the involved stake holders and their business processes, especially for information clients.

In IoT, a lookup service to locate item-level information stored at potentially unknown supply chain partners is still missing, and current lookup service and discovery service mainly rely on a centralized or a chain-style framework, e.g. EPCglobal Architecture, Affilias DS [1, 25], BRIDGE Project [24] and the Distributed ePedigree Architecture [17]. Apart from the issue of security and privacy, these systems have some drawbacks, such as poor scalability, load imbalance, poor reliability owing to the presence of single points of failure, or bottlenecks, which prevent them from being widely adopted. Moreover, to the best of our knowledge, the issue of locating hotspot resource in IoT has received much less attention.

In the last years, Peer-to-Peer (P2P) network has become one of the most popular applications in the Internet, and the P2P paradigm has emerged as an alternative to centralized and hierarchical architectures. The approaches to enhance the performance and robustness of lookup service by using structured P2P systems (e.g. Chord [28]) based on Distributed Hash Tables (DHT) that have a high potential as a replacement for ONS as

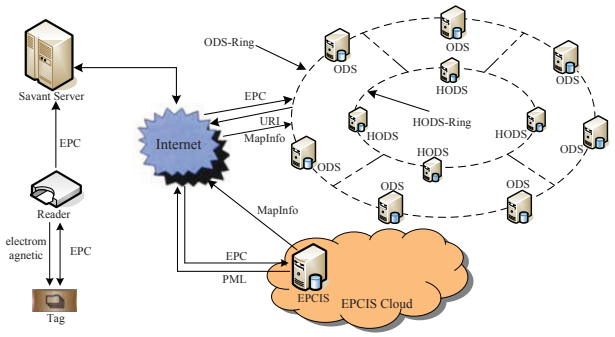


Figure 1: The framework of HDSA

well. Therefore, we present HDSA, a distributed hotspot-based IoT discovery service architecture adopting double-Chord-ring, give its framework and some relevant mechanisms. Here we mainly focus on the goals of meeting security and privacy requirements.

The rest of this paper is structured as follows. In Section 2, we give the framework of HDSA and some relevant mechanisms for security consideration. Then, we further discuss and analysis our solution in Section 3. Section 4 describes some work in related disciplines. Finally, we draw conclusions and outline future work in Section 5.

2 Hotspot-based Discovery Service Architecture

In this section, we present a distributed hotspot-based IoT discovery service architecture HDSA. The basic framework of HDSA is shown in Figure 1.

2.1 The Base Composition of HDSA

HDSA consists of three major parts Savant Server, ODS (Object Discovery Service) System and EPCIS Cloud.

2.1.1 Savant Server

Savant [6], designed to process the streams of tag or sensor data (event data) coming from of one or more reader devices, is a middleware software system that sits between tag readers and enterprise applications. Its intent is to address the unique computational requirements presented by EPC applications. Savant performs filtering, aggregation, and counting of tag data, reducing the volume of data prior to sending to Enterprise Applications. The Savant itself is a container for processing modules defined by Auto-ID standards or users and other third parties. More details about Savant can be found in [6]. The Savant server is a server installed with savant software. Each company deploys only one Savant Server logically, although in practice, every one may deploy more than one Savant Server.

2.1.2 ODS System

In HDSA, from functionality, we combine ONS and DS into ODS, and introduce Chord into ODS, the structured Chord overlay networks are the network substrate of the applications about ODS and information interaction. Every participant, such as manufacture, distributor, or retailer, deploys a dedicated ODS node, and all the ODS nodes are organized in a Chord ring and ordered following the hash values of their IPs.

Within a continuous period T, when the accessed times of an EPC exceeds the accessed times threshold of EPC (AT), the EPC is called hotspot resource (HR). Only depends on ODS-Ring, as the HR objects are frequently accessed by the client application of some organizations, it generates vast network traffic flows that may lead to the network congestion of ODS-Ring. To relieve this problem and balance the flows of query, we add a HR ODS-Chord-Ring (HODS-Ring) into HDSA. On the other hand, HODS brings appropriate hotspot data redundancy and backup, helps to enhance the reliability of the system. ODS-Ring and HODS-Ring together form an ODS System. The ODS-Ring is responsible for the queries of all objects (include HR), and the HODS-Ring is only responsible for the queries of HR objects. In real world, HODS nodes, may be derived from the participants'ODS servers which have better hardware configuration, also may are responsible by the third party or be constructed and maintained by the government, according to network region. Whichever method to be selected, it is depended on the concrete situation or the relevant regulation of real world.

In one company, the ODS/HODS node may each be implemented by multiple physically separate servers that act as backups for each other to increase the scalability and reliability of the entire system. It is important to note that the number of HODS nodes must be far smaller than the one of ODS nodes, which is analyzed in detail in our another paper.

2.1.3 EPCIS Cloud

The EPCIS is a role defined in EPCglobal Network Architecture Framework [16], which provide for storage and retrieval of filtered and processed information related to EPC-tagged objects about different events within the supply-chain. In HDSA, each participant (company or organization) maintains at least one EPCIS server. All the EPCIS servers constitute an EPCIS Cloud.

Normally, a company needs only one EPCIS Server in theory. As a kind of service, each company may deploy more than one EPCIS Server (multiple redundant servers, one for backup of another one) as needed, but logically, the external feature of multiple EPCIS Servers within a company is still one server through the relevant mechanism of main-backup. For the sake of cost saving, for every stakeholder, its ODS may be combined together with its EPCIS server.

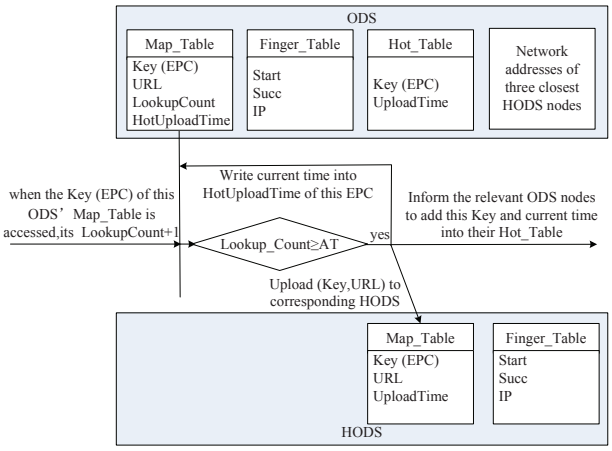


Figure 2: ODS storage & data flow chart

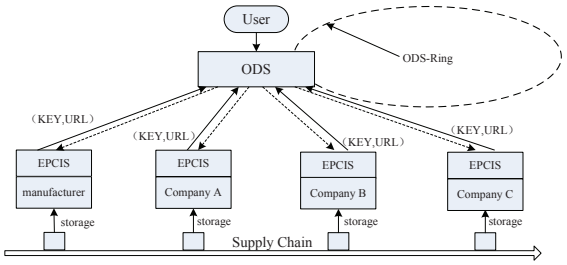


Figure 3: Information publishing principle chart

2.2 Storage Mechanism

In our HDSA framework, the DS is designed not to duplicate or aggregate information stored within each individual EPCIS repository, but to store only sufficient relevant information to be able to create the list of links.

Each of the ODS node has and maintains a finger table, a mapping table (map-table), a hotspot table (hot-table), and holds the network addresses of three closest HODS nodes to facilitate the information interaction between it and HODS-Ring. Here, finger table is similar to the one of Chord [28]. Hot-table includes two fields: key represents the hash value of hotspot EPC, and upload time represents the time when the EPC become as HR. Map-table mainly stores the list of mappings between EPC and the network addresses (IP address or URL) of its corresponding EPCIS servers, the lookup count (i.e. the accessed times of the EPC). The information storage in ODS System are shown in Figure 2, where Key (EPC) represents the hash value of EPC. Only one finger table and one map-table are in HODS node, where the roles and content of these tables are similar to the ones in ODS node.

2.3 Information Interaction Mechanism

In our HDSA, the information interactions principally include three aspects: publishing of resource information, interaction of hotspot information and switch of lookup between ODS-Ring and HODS-Ring.

2.3.1 Information Publishing

The current EPCIS standard [9, 21] does not involve a specific communication mechanism between an EPCIS and a lookup service. Thus, here, we give an information publishing mechanism to send the association between an EPC and the URL of the relevant EPCISs to corresponding ODS node.

When an EPC-tagged object flows along the supply chain, the relevant EPCIS servers must publish in time the relevant event information related to the EPC to the corresponding ODS node that determined by hashing the object EPC. The event information registered or published to ODS by EPCIS (see Figure 3) is mainly the mapping information associated to the current EPC that is passing the company that possesses this EPCIS. The mapping information is a 2-tuple (KEY, URL), includes two fields: the SHA1 (or MD5) hash value of the EPC, the URL where the information related to the EPC is available.

2.3.2 Interaction of Hotspot Information

As shown in Figure 2, for one ODS node, once someone EPC of its map-table becomes as HR, it simultaneously do three things. They are: (1) uploading the relevant information of this EPC to corresponding HODS nodes map-table, (2) writing current time into the field of Hot Upload Time of its map-table, (3) and informing the relevant ODS nodes which are consulting the EPC to add this Key and current time into their hot-table. On the other hand, with the moving through the supply chain, for a HR, once new mapping information about it is published to the corresponding ODS node by relevant EPCIS Server, this ODS node upload the newest information to relevant HODS node to assure the performance of real-time of mapping information.

2.3.3 Switch of Lookup

When an ODS node receives an EPC query of subscriber or end user, if it finds that the EPC is in its hot-table, it will forward the query to its available closest HODS node (one of the three closest HODS nodes). Then, the HODS node performs the query in inside ring, and returns query result to this ODS node.

2.4 Lookup Mechanism

As the complete information about an individual object may be fragmented across multiple organizations, the role of lookup service is to locate all the providers of the fragments of information that constitute the complete supply-chain or lifecycle history for an object. Here, we give relevant lookup mechanism: to obtain the address list of EPCIS server related to EPC-tagged object being queried

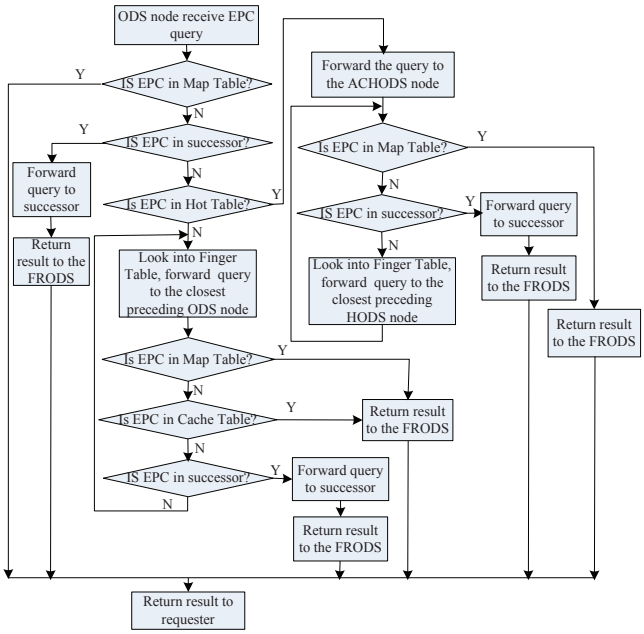


Figure 4: Flow chart of routing in chord-based ODS system

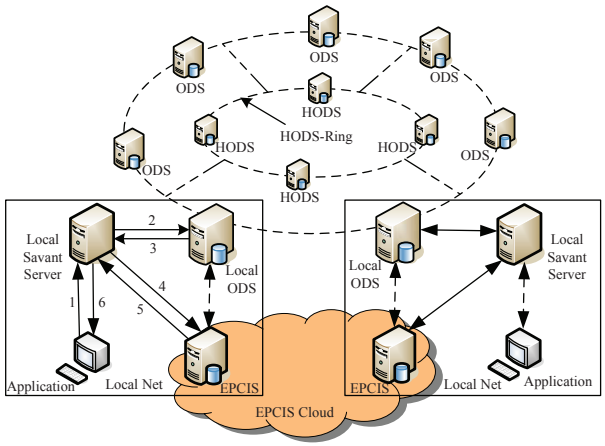


Figure 5: Base composition of company deployment & lookup data flow of application level

through the routing in ODS system, and then to get the complete information about this object through the lookup of application level. In our architecture framework, the hash value of EPC of object is used as an index of query.

2.4.1 Routing in Chord-based ODS System

As shown in Figure 4, routing in Chord-based ODS System is based on the following EPC lookup algorithm.

Firstly, the ODS node which is the first receiver of a query from a requester (Savant server) on a given EPC, looks into its map-table (we call this ODS node FRODS). If this EPC is found, it returns directly relevant address information to requester. Otherwise, it judges whether the key (the hash value of the EPC) is located between itself and its immediate successor. If is, it forwards the query to the immediate successor that is responsible for the requested EPC-tagged object, and then, the immediate successor returns directly desired result to FRODS. If not, it looks into hot-table, if this EPC is found, it forwards the query to ACHODS (one of the three closest HODS node and is available) which is responsible for this query within HODS-Ring and return net result (relevant address information) to it. If FRODS does not find EPC in hot-table, it looks into its finger table for the closest ODS node to the key (the hash value of the EPC) that has a lower or equal identifier, and forwards the query message to this ODS node which is called the closest predecessor refer to this key.

The closest predecessor does the work similar to what done by the FRODS. The main difference between them is that the closest predecessor does not look into its hot-table. In this way, the query message is routed through the ODS-Ring until it reaches a node containing mapping information related to the requested EPC-tagged object, this node then returns directly desired result to FRODS. Finally, FRODS returns the result to the requester.

Likewise, when HODS node receives the query forwarded by other node (FRODS or other HODS node), it takes the handling similar to what done by the closest predecessor in ODS-Ring. The main difference is that HODS lookups EPC only in its map-table.

2.4.2 Lookup of Application Level

In Figure 5, the parts within rectangle line represent the base composition of a company deployment, and the first to the sixth step of the query procedure of application level are described as follows:

1) The application issues EPC query to local Savant server through whose application software interface.

2) The local Savant server calls the query module to forward the query to the local ODS server.

3) The local ODS server calls lookup module to consult ODS system (as depicted in Section 2.4 1). If it finds successfully the desired result a list of network addresses of relevant EPCIS servers related to the EPC, it returns the result to the local Savant server. Otherwise, it analyzes the causes of failure and sends this analysis result to the local Savant server.

4) The local Savant server analyzes the result from the local ODS Server. If the result is failure causes, it switches to exception handling, and informs application. If is the list of EPCIS addresses, it accesses in parallel the corresponding EPCISs of EPCIS Cloud according to the list of addresses, to collect the detail information related to the EPC.

5) Every corresponding EPCIS sends its lookup result using PML (Physical Markup Language) file format to the local Savant server.

6) The local Savant server resolves these lookup results from previous step and sends them to the application through the application software interface.

Finally, the application integrates the collated results and displays them through the user-friendly interface.

2.5 Security Mechanism

In IoT, lookup services or Discovery Services are not only critically dependent upon the high efficiency of lookup and the integrity, but also the confidentiality they offer their customers. Therefore, in HDSA, apart from all information interaction use secure channels, the following security measures are given to meet the corresponding security and privacy requirement.

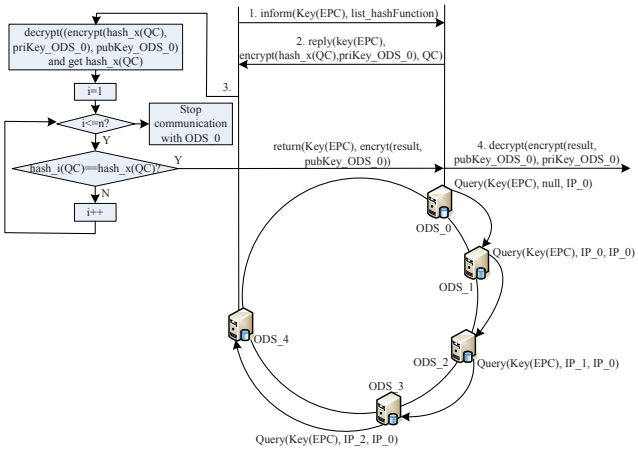


Figure 6: Base composition of company deployment & lookup data flow of application level

2.5.1 Mutual Authentication

To enable the retrieved address and object information could be authenticated, the responder and requester of object information must be authenticated mutually via certificate services. In HDSA, all participants (organization/company) need to obtain two certificates from certificate authority (CA), one certificate used for query is called query certificate (QC) and installed on the Savant server and the ODS server, another certificate used for sending lookup result to requester is called response certificate (RC) and installed on EPCIS server.

When a participant approaches CA to obtain certificates, its identities (for responder or requester) are verified and after complete verification, the corresponding certificate is issued. Communicating participants use them for mutual authentication. Before Savant server communicates with the corresponding EPCIS server, a mutual verification of them is performed. The certificates (QC of Savant server and RC of EPCIS server) which they obtained from Certificate authority are verified.

2.5.2 User Account Management

Some information about some item is confidential to some user and should only be accessed by the user has the corresponding right. Therefore, we provide corresponding access right for user to prevent unauthorized read of information.

Before beginning to perform the operation of lookup, every user must have an account including user name, password, and access right. The account information of user is stored in the user-account-table in the relevant Savant server.

In our solution, the Savant servers of all participants are assigned a corresponding level right (is called lookup-right) to lookup object information. The levels of their lookup-rights do not have to be same or different. Whatever level to be assigned, it is depended on the concrete situation or the relevant regulation of real world. When registering to the Savant server of someone participant, user must provide some necessary identity information and credentials to be verified whether the user has the qualification to obtain the level of lookup-right. If has, the Savant server creates corresponding user account for the user in its user-account-table, give user name and password, and assign its lookup-right to the access right of the user, and add the base identity information of the user into this table. Otherwise, the Savant server refuses to create account for the user, so the user could not to issue lookup via the Savant server.

2.5.3 Security Measure in Communication between Savant and EPCIS

After mutual authentication between Savant server and the corresponding EPCIS server is completed, a session key is created, Savant sever begin to issue query encrypted by the session key to EPCIS server. EPCIS server receives and decrypts the query information from the Savant server by the session key to get the EPC and the access right of the query user. Then, according to the access right, the corresponding information of item related to the EPC is extracted from the EPCIS servers lookup result, and sent to Savant server after being encrypted by the session key. Finally, the Savant server receives and decrypts the information from EPCIS server by the session key to obtain the desired lookup result.

2.5.4 Security Measure in ODS System

In ODS system, for the sake of backtracking processing when the lookup fails, the query message mainly include three parts: the hash value of EPC, the IP address of FRODS and the IP address of the closest source ODS node (e.g. ODS_1 is the closest source ODS node of ODS_2 in Figure 6) of current ODS node. When the destination ODS node (e.g. ODS_4 in Figure 6) is found, it

immediately initiates communication to FRODS (ODS_0 is the FRODS in Figure 6). As shown in Figure 6, the detailed procedures are described as follow:

1) ODS_4 informs ODS_0 that it has the desired information about Key(EPC) and the hash algorithms supported by it include: hash_0, hash_1, $\cdots$, hash_n.

2) ODS_4 decrypts the encrypted data from ODS_0 by the public key of ODS_0 and get hash_x(QC). Then, it uses hash_i ($i = 1, 2, \cdots, n$) one by one to get hash_i(QC), and compares hash_i(QC) and hash_x(QC). If they are equal, ODS_4 returns the lookup result (list of addresses) encrypted by the public key of ODS_0 on the Key(EPC) to ODS_0; Otherwise, when i is equal to n, hash_i(QC) is still not equal to hash_x(QC), ODS_4 stops communication with ODS_0 and disconnects with it.

3) ODS_0 decrypts the encrypted lookup result from ODS_4 by its private key and gets the desired lookup result about Key(EPC).

3 Discussion and Analysis

In the following, according to privacy enhancing technologies and Security and privacy needs of IoT, we give corresponding discussion and analysis for our solution and security mechanism.

3.1 Privacy Enhancing Technologies

It is quite difficult to enhance the privacy of user of network and service infrastructures. In order to achieve information privacy goals, a number of Privacy Enhancing Technologies (PET) have been developed. They can be described in brief as follows [12, 30]:

- Virtual Private Network (VPN) is extranet established by close groups of business partners. This solution may reduce the confidentiality and integrity risks, but it does not allow for a dynamic global information exchange and is impractical with regard to third parties beyond the borders of the extranet.

- Transport Layer Security (TLS) could improve confidentiality and integrity of the IoT on the base of an appropriate global trust structure, but it would be negatively affected the performance of lookup.

- DNS Security Extensions (DNSSEC) use public-key cryptography to guarantee origin authenticity and integrity of delivered information. However, DNSSEC could only assure global ONS information authenticity if the Internet community as a whole adopts it.

- Onion Routing cryptographically transform and mix Inter-net traffic from many different sources to impede matching a particular IP packet to a particular source. However, onion routing would affect the usability of ONS and Discovery Services because of the latency and performance issues.

- Private Information Retrieval (PIR) systems conceal which client is interested in what information, once the EPCIS have been located. However, in the global lookup system such as the ONS, problems of scalability and key management, as well as performance issues, make this method impractical.

Some of the above- mentioned methods could be combined to create alternatives for enhancing security and private of IoT.

To further increase security and privacy, DHT-based (Distributed Hash Tables) Peer-to-Peer (P2P) system that generally shows good scalability and performance in real-world applications, is a good method. In recent years, as one of the most popular applications in the Internet, the P2P paradigm has emerged as an alternative to centralized and hierarchical architectures. The advantages of DHTs include, among other aspects, self-organizing, load-balance, less traffic that data placement and search procedures generate. Chord, one of the most popular DHT-based systems, has not only simple design idea and good features of distribution, scalability, stability, and load balancing, and it is the focus of the present research, such as [7, 8, 14, 18, 19, 26, 27]. Though both existing EPCIS and DHT have not offered any encryption measure or access control [11], the encryption of EPCIS connection and the authentication of customer could be implemented without major difficulties, using common Internet and web service security frameworks [12], and the authentication of customer can be done by issuing shared secrets or using public-key cryptography [11]. Therefore, combining some of the above- mentioned methods and implementing access control at the actual EPCIS itself, a distributed hotspot-based discovery service architecture based on Chord for IoT, is a good solution to meet the functional requirements (with high performance and robustness) and security & privacy requirements.

3.2 Security and Privacy Requirements

From the technologies point of view, the lookup service (or discovery service) architecture of the IoT has an impact on the security and privacy of the involved stake holders. As described in [11] and [30], a high degree of reliability is needed in business processes and private enterprises using IoT technology will have to include the following security and privacy requirements into their risk management concept governing the business activities in general:

- Resilience to attacks: The system can avoid single points of failure and adjust itself to node failures.

- Data authentication: Retrieved address and object information should be authenticated.

- Access control: Information providers must be able to implement access control on the data provided [3].

- Client privacy: Measures need to be taken that only the information provider is able to infer from observing the use of the lookup system related to a specific customer; at least, inference should be very hard to conduct.

As HDSA is constructed on the base of Chord protocol, all storage-load and lookup-load are distributed evenly on the entire system, which make it can avoid single of failure and enhance its adaptivity by performing the stabilization operation inherited from Chord. The mutual authentication between the Savant server performing query operation and the corresponding EPCIS server enables the retrieved address and object information could be authenticated in HDSA. Though the user account management, HDSA provides corresponding access right for user to prevent unauthorized read of information, thus obtains some access control capabilities.

In our solution, information publishing, query, and storage are all done using the hash value of EPC, which make the information is not understandable or meaningful to non-requester (non-subscriber) of query or eavesdroppers or attackers, and thus enhance the protection on privacy of client and security of information to certain extent. On the other hand, due to the adoption of P2P, the number of ODS (and HODS) nodes the adversary or attackers have to monitor is very large and is increasing, which make the inference be very hard to conduct for the attackers. Additionally, the security measures in ODS System ensure the correctness and completeness of the returned discovery service information, in particular the addresses of the relevant EPCIS, and the security measure in communication between Savant and EPCIS ensures the correctness and completeness of the object information itself, which be able to reduce the whole systems confidentiality and integrity risks.

In ODS system, the hash value of EPC is used as the index to locate the corresponding EPCIS servers address and the number of ODS (or HODS) nodes is very large, which make the eavesdroppers cannot know the actual value of EPC and eavesdropping more difficult. Therefore, the mutual authentication between ODS (or HODS) nodes is not very necessary. So, our solution does not require the mutual authentication, which can help the improvement of lookup performance.

Technically, measures ensuring the IoT architectures resilience to attacks, data authentication, access control and client privacy need to be established. On the other hand, international legislator [30] would best establish an adequate legal framework considering the underlying technology, so the requirements of security and privacy can better meet.

4 Related Works

In the last years, the achievements of EPCglobal standardization efforts are substantial and the diffusion of the EPCglobal network continues. However, DS is not yet specified, the granularity of ONS lookup is still limited to product type, rather than serial-level lookup in the up-to-date version of ONS 2.0.1 [15], and ONS is based on DNS suffering from well-studied weaknesses in robustness, configuration complexity, and security. Relying heavily on Root ONS to implement traceability applications, makes the centralized architecture of hierarchical ONS is vulnerable to single point failure and workload-imbalanced due to excessive lookup-load of Root ONS. Alias DS is compliant to the architecture framework of EPCglobal, its basic characteristics are hierarchical lookups and DNS-based naming and translation. The BRIDGE project, supported by the EU and coordinated by GS1, addressed a wide spectrum of problems related to the implementation of RFID in Europe, whose prototype is very similar to the EPCglobal approach. The two kinds of approach and the EPCglobal approach share the same advantages and disadvantages [10].

Barchetti et al. [2] focused on the implementation of DS developed as an extension of FossTrak open framework [13] based on centralized framework. MUTLER et al. [22] presented a centralized aggregating DS for the EPCglobal Network and showed how to overcome scalability challenges through notification in a real-world environment. MANZANARES-LOPEZ et al. [21] proposed an distributed discovery service for EPCglobal network in nested package scenarios, which is based on the implementation of a DHT-based (Pastry) ONS and a totally distributed DS. Although they solve some problems, they all ignore the enhancement of security and privacy.

HUANG et al. [17] propose a distributed ONS architecture by combining ONS and DHT to find out drug counterfeit points in the pharmacy supply chains, and the distributed ONS be constructed involving the EPCIS servers of all the participants. This approach of locating drug counterfeit points through forward tracing or reverse tracing is called Daisy Chain approach. Although the approach is implemented in a distributed and relatively secure manner, all the processes rely on rewritable tags and must be supervised by an entity to be able to identify the offender [4, 21]. In addition, to collect information about an item, traversing all relevant EPCISs along the supply chain for a given EPC cannot be parallelized and therefore raises high lookup latency because each EPCIS must be queried sequentially [4, 22].

In preceding studies, researchers pay less or no attention on the issue of locating hotspot resource object in IoT. In our solution, ODS and DS are combined into ODS from functionality and Chord lookup algorithm is introduced into ODS, which not only make the entire system is easier and more efficient to implement item-level tracking & tracing, but also remove the faults DNS-based ONS. The hotspot-based HODS-Ring further improves the efficiency of lookup, balances the flows of query and reduces the probability of congestion of ODS-Ring, which make the whole system is more efficient and robust. In contrast to the Distributed ePedigree Architecture proposed by Huang et al. in HDSA, the information collec-

tion from all relevant EPCISs along the supply chain for a given EPC can be parallelized and therefore achieves low lookup latency because each EPCIS can be queried concurrently, which further improves the lookup efficiency in practice. Additionally, the introduction of security mechanism ensures our architectures resilience to attacks, data authentication, access control and client privacy.

5 Conclusions

For the current development status and existing problems of lookup service in IOT, especially for the issue of security and privacy, this paper proposes HDSA with security mechanism, which points out one of viable directions to foster the development of IOT. So far, many research problems and implementation issues are still unsolved, and require more efforts from both academia researchers and industrial practitioners, though lookup service is a vital research direction in IOT. Future work should focus on the latest security techniques and protocols (such as [5, 20, 23]) and further enhance the security of the Internet of Things. Additionally, to introduce Cloud Storage and Cloud Computation into HDSA will also be our next work.

Acknowledgments

This study was supported by the National Natural Science Funds Fund of China [61172084, 61202046], Natural Science Foundation of Hubei Province of China [2013CFC026], and the project of Study on Information Security of Chord-based Object Discovery for the Internet of Things. The authors gratefully acknowledge the anonymous reviewers for their valuable comments.

References

[1] Afilias, "Afilias discovery services: Enabling secure, selective visibility in global supply chains," Tech. Rep. White Paper, 2008.

[2] U. Barchetti, A. Bucciero, M. De Blasi, L. Mainetti, and L. Patrono, "Implementation and testing of an epcglobal-aware discovery service for item-level traceability," in *Proceedings of International Conference on Ultra Modern Telecommunications and Workshops (ICUMT'09)*, pp. 1–8, St. Petersburg, Russia, Oct. 2009.

[3] C. Floerkemeier, M. Langheinrich, E. Fleisch, F. Mattern, and S. E. Sarma, *The Internet of Things*, Berlin/Heidelberg: Springer, 2008.

[4] J. J. Cantero, M. A. Guijarro, G. Arrebola, E. Garcia, J. Baos, M. Harrison, and T. Kelepouris, "Traceability applications based on discovery services," in *Proceedings of IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008)*, pp. 1332–1337, Hamburg, Sept. 2008.

[5] C. H. Wei, M. S. Hwang, and A. Y.H. Chin, "An authentication protocol for low-cost rfid tags," *International Journal of Mobile Communications*, vol. 9, no. 2, pp. 208–223, 2011.

[6] S. Clark, K. Traub, D. Anarkat, and T. Osinski, "Auto-id savant specification 1.0,", Tech. Rep. Auto-ID Center, 2003.

[7] R. Cuevas, M. Uruena, and A. Banchs, "Routing fairness in chord: Analysis and enhancement," in *Proceedings of IEEE INFOCOM'09*, pp. 1449–1457, April 2009.

[8] Y. P. Deng and H. Du, "Improved chord algorithm based on physical topology," *Computer Engineering and Design*, vol. 33, no. 10, pp. 3734–3738, 2012.

[9] EPCglobal, "EPC information service (EPCIS) ver. 1.0.1," tech. rep., Sept. 2007.

[10] S. Evdokimov, B. Fabian, S. Kunz, and N. Schoenemann, "Comparison of discovery service architectures for the internet of things," in *Proceedings of 2010 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'10)*, pp. 237–244, California, USA, June 2010.

[11] B. Fabian and O. Gunther, "Distributed ons and its impact on privacy," in *Proceedings of IEEE International Conference on Communications (ICC'07)*, pp. 1223–1228, Glasgow, Scotland, June 2007.

[12] B. Fabian and O. Gunther, "Security challenges of the epcglobal network," *Communications of the ACM*, vol. 52, no. 7, pp. 121–125, 2009.

[13] C. Floerkemeier, C. Roduner, and M. Lampe, "Rfid application development with the accada middleware platform," *Systems Journal*, vol. 1, no. 2, pp. 82–94, 2007.

[14] A. Forestiero, E. Leonardi, C. Mastroianni, and M. Meo, "Self-chord: A bio-inspired p2p framework for self-organizing distributed systems," *IEEE/ACM Transactions on Networking*, vol. 18, no. 5, pp. 1651–1664, 2010.

[15] GS1, "GS1 object name service (ons) 2.0.1," tech. rep., Jan. 2013.

[16] GS1, "The GS1 epcglobal architecture framework-ver. 1.5," tech. rep., Mar. 2013.

[17] D. J. Huang, M. Verma, A. Ramachandran, and Z. B. Zhou, "A distributed epedigree architecture," in *Proceedings of the 11th IEEE International Workshop on Future Trends of Distributed Computing Systems (FTDCS'07)*, pp. 220–230, Mar. 2007.

[18] W. A. Imtiaz, S. Shil, and A. M. Rahman, "Three layer hierarchical model for chord," *International Journal of Advanced Computer Science and Applications*, vol. 3, no. 11, pp. 96–101, 2012.

[19] C. Jennings, B. Lowekamp, E. Rescorla, S. A. Baset, and H. Schulzrinne, "Resource location and discovery (reload)," Tech. Rep. IETF Internet-Draft, 2008.

[20] W. Khedr, "On the security of moessners and khans authentication scheme for passive epcglobal c1g2 rfid tags," *International Journal of Network Security*, vol. 16, no. 5, pp. 369–375, 2014.

[21] P. Manzanares-Lopez, J. P. Munoz-Gea, J. Malgosa-Sanahuja, and J. C. Sanchez-Aarnoutse, "An effcient distributed discovery service for epcglobal network in nested package scenarios," *Journal of Network and Computer Applications*, vol. 34, no. 3, pp. 925–937, 2011.

[22] J. Muller, J. Oberst, S. Wehrmeyer, J. Witt, A. Zeier, and H. Plattner, "An aggregating discovery service for the epcglobal network," in *Proceedings the 43rd Hawaii International Conference on System Sciences*, pp. 1–9, Honolulu, HI, Jan. 2010.

[23] A. Nitaj, "Cryptanalysis of ntru with two public keys," *International Journal of Network Security*, vol. 16, no. 2, pp. 112–117, 2014.

[24] University of Cambridge, AT4 wireless, BT Research, and SAP Research, "Bridge wp02-working prototype of serial-level lookup service," Tech. Rep. BRIDGE project, 2008.

[25] A. Rezafard, "Extensible supply-chain discovery service problem statement," Tech. Rep. IETF Internet-Draft, 2008.

[26] L. Schmidt, R. Dagher, R. Quilez, N. Mitton, and D. Simplot-Ryl, "Dht-based distributed ale engine in rfid middleware," Tech. Rep. Research Report 7316, INRIA, 2010.

[27] Y. Shimano and F. Sato, "Dynamic reconfiguration of chord ring based on physical network and finger table information," in *Proceedings of 2012 15th International Conference on Network-Based Information Systems (NBiS'12)*, pp. 66–73, Melbourne, Australia, Sept. 2012.

[28] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 4, pp. 149–160, 2001.

[29] R. H. Weber, "Internet of things - need for a new legal environment," *Computer Law and Security Review*, vol. 25, no. 6, pp. 522–527, 2009.

[30] R. H. Weber, "Internet of things new security and privacy challenges," *Computer Law and Security Review*, vol. 26, no. 1, pp. 23–30, 2010.

[31] Lu Yan, Y. Zhang, L. T. Yang, and H. Ning, *The Internet of things: from RFID to the next-generation pervasive networked systems*, New York/London: Auerbach, 2008.

Degang Xu received the BE degree in computer science and technology, M.S. degree in computer application technology and the Ph.D. degree in computer system architecture from Huazhong University of Science and Technology in 2000, 2007 and 2013, respectively. He is currently a lecturer in the School of Mathematics and Computer Science, Hubei University of Arts and Science, CHINA. His research interests include computer network and security, cloud computing, DHT-based P2P network, and Internet of Things.

Zhao Wu received the BE degree in computer application from China University of Geoscience, M.S, degree in computer application from Wuhan University of Technology and the Ph.D. degree in Computer software and theory from Wuhan University, in 1999, 2003 and 2007, respectively. He is currently a associate professor in the School of Mathematics and Computer Science, Hubei University of Arts and Science, CHINA. His research interests include cloud computing, service computing, and internet of things. He is a member of the Association for Computing Machinery.

Zhongbo Wu received the BE degree in information technology from Central China Normal University, MS degree in computer application from Huazhong University of Science and Technology and the PhD degree in computer application from Renmin University of China, in 2001, 2004 and 2010, respectively. He is currently an associate professor in the School of Mathematics and Computer Science, Hubei University of Arts and Science, CHINA. His research interests include cloud computing, database, and internet of things. He is a member of the IEEE.

Qilin Zhang received the BE degree in computer science and technology from Huazhong University of Science and Technology in 2003, MS degree in computer application technology from Wuhan university of technology in 2006 and the PhD degree in communication and information system from Wuhan University 2013, respectively. He is currently a lecturer in the School of Mathematics and Computer Science, Hubei University of Arts and Science, CHINA. His research interests include computer network and security, smart grid and Internet of Things.

Leihua Qin received the Ph.D. degree in computer system architecture in 2007 from Huazhong University of Science and Technology. He is now an associate professor at School of Computer Science and Technology of HUST. His main research interests include computer network and cloud computing, network storage system and Internet of Things.

Jingli Zhou received the B.E. degree in 1969. She is a Professor and doctor advisor at Huazhong University of Science and Technology. She had been a visiting scholar in USA from 1995 to 1996 and has been honor of the State Department Special Allowance since 1999. Her main field of research: computer network, network storage system and multimedia signal processing.

Provably Secure Partially Blind Signature Scheme Based on Quadratic Residue

Yi Zhao[1], Qiliang Yang[2], and Bo Yang[1]
(Corresponding author: Yi Zhao)

[1]School of Computer Science, Shaanxi Normal University
Xi'an 710062, China
[2]College of Informatics, Shu-Te University
Kaohsiung City 82445, Taiwan
(Email: yizhaore@gmail.com)
(Received Dec. 11, 2013; revised and accepted Nov. 6, 2014)

Abstract

Partially blind signature schemes are the most important ingredient for anonymity in off-line e-cash system. In this paper, a new approach to setup formal security arguments in random oracle model for factorization based partially blind signature schemes is presented. Then a provably secure and efficient scheme based on quadratic residue is proposed. The approach also allows one to give formal proofs in the random oracle model for all the factorization based fully blind signature schemes. Our scheme takes an outstanding performance in computational costs compared to the existing schemes.

Keywords: Partially blind signature, Provable security, Quadratic residue, Random oracle model

1 Introduction

1.1 Background

In an e-cash system, customers are always not willing to reveal their privacies like transaction records to others while trading on line. To offer protection for privacy by the blindness property, Chaum first introduced the technique called "blind signature" in 1983 [4]. Though blind signature was sufficient to solve the problem of privacy protection, two other new matters appeared next. The first is that the bank has to keep an unlimited database which stores all the transaction records in history to check the occurrence of double-spending issues. Apparently the cost of storage space and searching goes higher at fast speed as the period of blind signature scheme used is getting longer. The second is that the signer can not assure that the message to sign includes the right information without seeing it. The signature may be used for illegal purposes.

The above two shortcomings can be eliminated by partially blind signatures which is introduced by Abe and Fujisaki in 1996 [1]. By injecting some agreed common information such as the expiration date and the face value to the signature which can't be replaced by users, the scheme gets the property of partial blindness. So the bank can delete all the expired records to keep a constant size of the database. And also the bank will confirm that the message to sign contains the information it really concerns. Without loss of privacy of other information, the user just needs to renew the e-cash when the old one is close to expire.

The first partially blind signature scheme based on RSA was proposed by Abe and Fujisaki along with the concept which is vulnerable to one-more-forgery attack. It was realized that a signature scheme should be enhanced by adding random factors to get the randomization property which was suggested by Ferguson [7]. Until now, numerous security enhanced partial blind signature schemes have been proposed. Abe and Okamoto proposed a provably secure partially blind signature based on Schnorr signature scheme [2] and then Okamoto also presented a scheme under standard model based on bilinear groups [14]. The computational costs of both schemes above are so high to be in application. Wu et al. gave an improved Abe scheme and a inverse Schnorr based scheme with higher efficiency [19]. There are also some discrete logarithm based schemes which is not of Schnorr type like [9, 11, 12]. Tianjie Cao et al. proposed a partial blind signature scheme based on RSA [3], which turned out to be insecure [8, 13]. Some other RSA based schemes were proposed by Tahat et al. [16] and Fang et al. [6]. Fan et al. proposed a scheme based on quadratic residue and emphasized on reducing the cost in verification [5]. It is notable that these factorization related schemes [3, 5, 6, 16] didn't give a formal proof. We can see that some successful attacks on blind signatures like [8, 13, 17] are due to the unproved constructions. A formal proof which rigorously claims the security under certain condition is neces-

sary. Besides these standard assumption based schemes, other interesting schemes like [18] which is based on Braid groups were also presented.

1.2 Our Contribution

As Pointcheval said [15], general methods of proofs used to establish security arguments for signature schemes no longer work in the blind context since we lose control over the value that the signer receives. The value doesn't only come from the random oracle but also the attacker(blinding factor). As a consequence, the signer can't be simulated without the secret key as usual. Thus, the ability of the attacker to make a forgery is hard to be related to a difficult problem. To overcome this problem, the existing DDH-based schemes like [1, 2, 15] use the concept of witness indistinguishable proofs which requires that many secret keys are associated to the same public key and the knowledge of two distinct secret keys provides the solution of a difficult problem. So the simulation can be constructed with the key pair generated by simulator, but the forgery output by the attacker may be associated to one secret key indistinguishable to the one simulator uses. The fact that one forgery can be implemented by two distinct secret keys provides the solution to a difficult problem. This approach is useful for the DDH-based schemes but the factorization based schemes don't satisfy the requirement of nontrivial witness indistinguishability since the public key modulus is one-to-one corresponded to the secret key factorization. The second scheme proposed in [19] didn't use witness indistinguishable proofs by employing key evolution to construct multiple public key environment to simulate the signer. We try to apply this thought to give proofs for the factorization based schemes.

Our approach uses a simple fact which we prove later that if computing factorization of one of a polynomial bounded number of moduli randomly generated is feasible with non-negligible probability, then computing factorization of one modulus randomly generated is also feasible with non-negligible probability. We setup security definition of unforgeability on the former problem. This is a kind of computational indistinguishability obtained by randomness. Our security definition is more close to the real applications of the schemes where several public keys are used on line in the same time.

In this paper, we design a partial blind signature scheme based on quadratic residue with low computational cost. We first introduce the basic theory and definitions, then we describe our scheme and give a formal proof under random oracle model. Also, we make a comparison of the computational cost between our scheme with existing ones. Our scheme is quite applicable in e-cash system especially for resource-limited user device like smart card.

2 Preliminaries

2.1 Legendre Symbol and Jacobi Symbol

Let p be a prime, Q_p denote the set of quadratic residues modulo n, $\overline{Q_p}$ the set of quadratic non-residues. For any $a \in Z_p^*$, the Legendre symbol of a is denoted by

$$\left(\frac{a}{p}\right) = \begin{cases} 0, & p|a. \\ 1, & a \in Q_p. \\ -1, & a \in \overline{Q_p}. \end{cases}$$

Let $Z_n^* = \{k \in Z_n, \gcd(k,n) = 1\}$ denote the multiplicative group with $n = pq$, where p and q are two large primes of the same size. the Jacobi symbol of $a \in Z_n^*$ is denoted by

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p}\right)\left(\frac{a}{q}\right)$$

The Jacobi symbol can be efficiently computed without the factorization of n.

2.2 Blum Integers

An integer $n = pq$ is called Blum integer if $p \equiv 3 \bmod 4$ and $q \equiv 3 \bmod 4$. When n is a Blum integer, the function $f(x) = x^2 \bmod n$ is a permutation over Q_n. If n is a Blum integer, $\left(\frac{-1}{n}\right) = 1$. That means the Jacobi symbols of any x and $-x$ are the same, which makes it hard to distinguish x and $-x$ by computing Jacobi symbols.

2.3 Quadratic Residues Modulo a Composite

An element $x \in Z_n^*$ is a quadratic residue if there exists a $y \in Z_n^*$ with $y^2 \equiv x \bmod n$, otherwise is called quadratic non-residue. Let Q_n denote the set of quadratic residues modulo n, $\overline{Q_n}$ the set of quadratic non-residues and $\overline{Q_n}^+$ the set of quadratic non-residues whose Jacobi symbol is $+1$. There are four square roots for each quadratic residue and only one of them is also a quadratic residue when the modulus is a Blum integer.

2.4 Related Complexity Assumptions and Results

Claim 1. *(decisional quadratic residue assumption). Without the factorization of n, deciding quadratic residuosity of any $x \in \overline{Q_n}^+$ is computationally infeasible to succeed with probability more than 1/2 for all probabilistic polynomial time algorithms [10].*

Definition 1. *(computational quadratic residue problem, denoted CQR). Without the factorization of n, compute square roots of x mod n where x is randomly chosen in Q_n and $\gcd(x,n) = 1$.*

Claim 2. *If factoring is computationally infeasible, then solving CQR is also computationally infeasible [10].*

Definition 2. *(multiple computational quadratic residue assumption, denoted MCQR). For a set S of moduli $n_1, \cdots, n_N$ randomly generated, where N is polynomial bounded, the factorization of n_i for any i is unknown, and $(x, n_i) = 1$, compute square roots of x modulo n_i for any one i.*

Claim 3. *If solving CQR is computationally infeasible, then solving MCQR is also computationally infeasible, and vice versa.*

Proof. It is trivial that if one can solve CQR, then one can solve MCQR by picking any n_i as modulo. Let's focus on the other direction.

If there exists a machine M that can solve MCQR with probability ϵ in polynomial time t in a uniform way (which means the probability of the modulo of the solution is uniformly distributed over S), then we can construct M' which uses M as a subroutine to solve CQR. Let n^* and x be the parameters in CQR. M' randomly generates $n_1, \cdots, n_{N-1}$ and obtains a set of N moduli by adding n^*. Then M' invokes M by giving it the set and x as inputs. The output of M is a tuple $(x^{1/2}, x, n_i)$. Since M works in a uniform way, the probability that M outputs $(x^{1/2}, x, n^*)$ is at least ϵ/N. In this case, $x^{1/2}$ is the solution of the CQR problem we want to solve. □

2.5 Using Chinese Remainder Theorem to Compute Square Roots [10]

Theorem 1. *(Chinese remainder theorem). Let $n = pq$ where p and q are relatively prime. Then $Z_n \cong Z_p \times Z_q$ and $Z_n^* \cong Z_p^* \times Z_q^*$. Moreover, let f be the function mapping elements $x \in \{0, 1, \cdots, N-1\}$ to pairs (x_p, x_q) with $x_p \in \{0, 1, \cdots, p-1\}$ and $x_q \in \{0, 1, \cdots, q-1\}$ defined by*

$$f(x) \stackrel{def}{=\!=} ([x \bmod p], [x \bmod q])$$

Then f is an isomorphism from Z_n to $Z_p \times Z_q$ as well as an isomorphism from Z_n^* to $Z_p^* \times Z_q^*$.

To compute a square root of x modulo a Blum integer $n = pq$ of known factorization, we use Chinese remainder theorem to find the presentation of x in $Z_p^* \times Z_q^*$, then compute the square roots in the same presentation and convert the result back to the presentation in Z_n^*. Namely, do as follows.

1) Compute $x_p = x \bmod p$ and $x_q = x \bmod q$.

2) Compute a square root $y_p = x_p^{(p+1)/4}$ of x_p and a square root $y_q = x_q^{(p+1)/4}$ of x_q, by the fact that one square root of any z modulo a prime $p' = 3 \bmod 4$ is $z^{(p'+1)/4}$.

3) Using Chinese Remainder Theorem to convert from the presentation $(y_p, y_q) \in Z_p^* \times Z_q^*$ to $y \in Z_n^*$. Output y.

3 Definitions

In a partially blind signature scheme, the signer and the user are assumed to agree on a piece of common information outside the signature issuing procedure, denoted as c. And a randomizing factor should be negotiated during the procedure for the randomization property. We formalize this notion by introducing function $R()$ which is defined outside the scheme. Function $R()$ is a polynomial-time deterministic algorithm that takes two arbitrary strings x and y that belong to the signer and the user respectively, and outputs u as the randomization factor. To compute $R()$, the signer and the user will exchange x and y with each other. Some parts of the following definitions refer to [2].

Definition 3. *(Partially blind signature scheme). A partially blind signature scheme is a four tuple $(\mathcal{G}, S, U, V)$.*

- *$\mathcal{G}$ is a probabilistic polynomial-time algorithm that takes security parameter n and outputs a public and secret key pair (pk, sk).*
- *S and U are two parties who follow the interactive signature issuing protocol. U takes message m , pk, the description of $R()$, and the common information c as initial inputs. S takes sk, the description of $R()$ and c as initial inputs. Then S and U engage in the signature issuing protocol and stop in polynomial-time. When they stop, S outputs either completed or not completed. If it is completed, U outputs a signature (m, s, u, c) or $\perp$ in private.*
- *V is a probabilistic polynomial-time algorithm that takes (pk, m, s, u, c) and outputs either accept or reject.*

Definition 4. *(Completeness). If the signer and the user follow the signature issuing protocol, then with probability at least $1 - \epsilon$ for negligible ϵ, S outputs completed and the user outputs (pk, m, s, u, c) that satisfies $V(pk, m, s, u, c) = accept$.*

To define the partial blindness property, let us introduce the following game.

Game A. Let U_0 and U_1 are two honest users who follow the signature issuing protocol with the same common information c.

1) The signer S does the key generation and publishes pk.

2) U_0 and U_1 engage the protocol with S respectively and get the message-signature tuple $(m_0, \pm s_0, u_0, c)$ and $(m_1, \pm s_1, u_1, c)$.

3) A random bit $b \in \{0, 1\}$ is selected and U_b sends $(m_b, \pm s_b, u_b, c)$ to S.

4) S outputs $b' \in \{0, 1\}$.

We say that S wins if $b' = b$. We define the advantage as $adv = |Pr[b' = b] - 1/2|$.

Definition 5. *(Partial Blindness). A signature scheme is partially blind if for any probabilistic polynomial-time algorithm S, S wins in Game A with negligible advantage. The probability is taken over the coin flips of S, U_0 and U_1.*

The unforgeability property is defined through the following game.

Game B. Let U^* be the user who tries to forge a signature after issuing the protocol with the signer S.

1) The signer S generates a number of pairs of keys $(n_i, (p_i, q_i))$ and publishes the public keys $n_i (i \in [1, N])$.

2) U^* randomly and independently chooses public keys and engages in the signature issuing protocol in a concurrent way. Let l be the total number of executions of the protocol and $c_{i,j}$ be the common information used corresponding to the j-th execution of the public key N_i.

3) U^* outputs $l+1$ public key-message-signature tuples $(n_i, m_{i,j}, \pm s_{i,j}, u_{i,j}, c_{i,j})$.

Definition 6. *(Unforgeability). A partially blind signature scheme is unforgeable if for any probabilistic polynomial-time algorithm U^* that plays Game B, the probability that all the $l+1$ signatures which U^* outputs after l interactions with S are valid is negligible. The probability is taken over the coin flips of S and U^*.*

4 The Proposed Partially Blinded Signature Scheme

Z_n^* is defined as that of Definition 2.3. Let c be the common information with constant length r containing the message like an expiration date and the e-cash value which is negotiated by the user and the signer. We assume that the messages to be signed can be expressed by the elements in Z_n^*. H is a public hash function: $H : \{0,1\}^r \times \mathbb{N} \to Q_{\mathbb{N}}$ and H_0 is a public hash function: $H_0 : \{0,1\}^* \times \mathbb{N} \to Q_{\mathbb{N}}$. (The input of the parameter $\mathbb{N}$ is n to ensure that the hash value is in the valid range when multiple public keys are in use at the same time. We directly use $H(r)$ instead of $H(r, n)$ for short if there is no confusion.)

Key Generation. The signer randomly selects two large primes p and q, computes $n = pq$ and publishes n as the public key. Here the secret key is (p, q).

Blinding. A user submits the common information c to the signer. After checking the validity of the common information, the signer randomly selects a randomizing factor $x \in Q_n$ and sends x^4 to the user as the commitment(signer can choose a random $x^{1/2} \in Z_n^*$ to generate x and compute x^4, and save $x^{1/2}$ to decommit later).

The user randomly selects his randomizing factor $y \in Q_n$ and blinding factor $k \in Q_n$. With the received commitment x^4 and the message m, the user computes the blinded message

$$\hat{m} = k^2 y^2 H_0(m \parallel x^4 y^4 \parallel c)$$

and sends it to the signer.

Signing. After receiving $\hat{m}$, The signer injects his randomizing factor and the common information into the blinded message, computes

$$\hat{m}' = x^2 \hat{m} H^{1/2}(c).$$

Let h_0 and h_1 denote the output by $H_0(m \parallel x^4 y^4 \parallel c)$ and $H(c)$ respectively. Then the signer calculates the square roots of $\hat{m}'$ by Chinese remainder theorem. We pick the square root which is also a quadratic residue of $\hat{m}'$ as the blinded signature. So the blinded signature $\hat{s}$ is $kxyh_0^{1/2}h_1^{1/4}$. Then the signer sends $\hat{s}$ and $(x, x^{1/2})$ to the user.

Unblinding. The user confirms that x is a quadratic residue, then computes $s = k^{-1}\hat{s}$ to remove the blinding factor, and computes $u = xy$ as the randomizing factor in the output. The tuple $(\pm s, u, c)$ is the signer's signature on the message m. (We denote $(s, -s)$ as $\pm s$ for short and treat $(s, -s)$ and $(-s, s)$ as the same signature).

Verifying. $V(pk, m, \pm s, u, c) = accept$ if

$$s^4 = u^4 H_0(m \parallel u^4 \parallel c)^2 H(c).$$

5 Security Analysis

5.1 Completeness

The completeness of our scheme can be easily conformed as follows.

$$\begin{aligned} s^4 &= (xyh_0^{1/2}h_1^{1/4})^4 \\ &= (xy)^4 h_0^2 h_1 \\ &= u^4 H_0(m \parallel x^4 y^4 \parallel c)^2 H(c). \end{aligned}$$

We can see that with probability 1, the signature output by issuing protocol legally satisfies the equation. That is perfect completeness.

5.2 Partial Blindness

Theorem 2. *Our proposed scheme is partially blind.*

Proof. Let S be a player of Game A. Let x_i, $\hat{m}_i$, $\hat{s}_i$ be the data recorded in the view of S during the execution of the protocol for $i = 0, 1$.

S receives $(m_b, \pm s_b, u_b, c)$ and tries to match it to the views. It is sufficient to show that for either view, there always exists a tuple of corresponding random factors

(y_i, k_i) to match the verification equation. We see that $y_i = x_i^{-1}u_b$, $k_i^2 = y_i^{-2}H_0(m_b \parallel u_b^2 \parallel c)^{-1}$, S can obtain k_i by the Chinese remainder theorem. Thus (y_i, k_i), $(x_i, \hat{m}_i, \hat{s}_i)$, and (m_b, s_b, u_b, c) have exactly the same relation as the scheme defined. And such (y_i, k_i) always exists regardless of what the value of $(m_b, \pm s_b, u_b, c)$. That implies that the signature $(m_b, \pm s_b, u_b, c)$ is independent from the blinded signature. So, even an infinitely powerful S wins Game A with probability exactly 1/2. Then the blindness property follows. □

5.3 Unforgeability

Theorem 3. *Our proposed scheme is unforgeable if $l < N\log p(n)$ for sufficiently large n. In other words, let q be the maximum number of queries to H in the simulation, if there exists a forger who can make a forgery in l executions with probability ϵ, then we can solve CQR with probability $\epsilon/(qp(n))$.*

Proof. Let U^* be the forger who plays Game B and produces a valid public key-message-signature tuple $(n_i, m^*, \pm s^*, u^*, c^*)$ that never appeared in the l executions of the protocol with probability ϵ which is not negligible. By using U^*, we construct a machine M to solve the problem of finding square roots in a passive environment. Notice that every n_i is generated randomly so that they are all identically distributed. As a result, the probability that every n_i appears in the output forgery is the same.

Notice that H_0 is only queried by users and blind to the signer. So we can't treat it as a random oracle when we use M to simulate S. The value of H_0 is just treated like a random factor which multiples another random factor k in the message $\hat{m}$ in the signer's view. Let q be the maximum number of queries asked from U^* to H. All the parameters are limited by a polynomial in a security parameter k_s. Let (n^*, x) be the instance that we want to find a square root of x in $Z_{n^*}^*$ without the factorization of n^*. M simulates Game B as follows.

1) Generate N pairs of large primes (p_i, q_i) and randomly select $I \in \{1, \cdots, N\}$. Let

$$n_i = \begin{cases} p_i q_i, & i \neq I. \\ n^*, & i = I. \end{cases}$$

2) Run U^* with those keys and simulate H and S as follows.

   For the query $c_{i,j}$ to H, return z such that

   - If $i = I$, choose $j \in_R Q_{n_i}$, return $z = j^4 x$ and record $(c_{i,j}, j)$.
   - If $i \neq I$, choose $j \in_R Q_{n_i}$, return $z = j^4$.

   For the requests to S,

   - If U^* requests signatures under modulo n_i that $i = I$, the simulation fails and aborts.
   - If U^* requests signatures under modulo n_i that $i \neq I$, return $\hat{s}$ by using Chinese remainder theorem because the factorizations of these n_i where $i \neq I$ are known (they are generated by M). So that M simulates S completely under this condition.

3) If U^* eventually forges a signature (n_i, m^*, $\pm$ s^*, u^*, c^*), output them.

Then we evaluate the probability that the simulation doesn't abort.

Claim 4. *If $l < N\log p(n)$, the probability that the simulation doesn't abort is at least $1/p(n)$.*

Proof. We assume that U^* chooses n_i in a uniformly random way. Such that the probability that the simulation doesn't abort is

$$\begin{aligned} (1-1/N)^l &= (1-1/N)^{N*l/N} \\ &> (1/e)^{l/N} \\ &> (1/e)^{\log p(n)} \\ &> 1/p(n). \end{aligned}$$

□

The probability that U^* makes a forgery successfully without asking H is negligible because of the unpredictability of the hash function. Thus, the success probability of M that doesn't abort to get a forgery on n_I is at least ϵ/q which is not negligible. According to Claim 4, we know that the probability that M make a forgery is $\epsilon/(qp(n))$.

Now we use M to solve the problem (n^*, x). When M obtains a forgery $(n_I, m^*, \pm s^*, u^*, c^*)$, it checks the records $(c_{i,j}, j)$ to find out the $c_{i,j} = c^*$. Then it establishes an equation

$$(s^*)^2 = (u^*)^2 H_0(m^* \parallel (u^*)^4 \parallel c^*) H(c^*)^{1/2}$$

where all the values except $H(c^*)^{1/2}$ are known. We denote the value of $H(c^*)^{1/2}$ by y which can be computed from the equation. M answered the query $c_{i,j}$ by $H(c_{i,j}) = j^4 x$ in the simulation. So

$$y = H(c^*)^{1/2} = j^2 x^{1/2}, x^{1/2} = yj^{-2}$$

Thus M finds a square root of x without the factorization of n_I. That contradicts the fact that the problem can't be computed efficiently.

□

6 Performance

We make a computational performance comparison between our scheme and several former schemes in Table 1 as follow. Using Chinese remainder theorem to compute roots costs about 1/4 of the time that one exponentiation

Table 1: Computation costs

Scheme	Exponentiation	Inverse	Hashing	Multiplication
Fan [5]	3/4	1	4	34
Wu [19]	7	1	5	5
Cao [3]	8	1	4	7
Fang [6]	5	2	4	27
Zhao	2/4	1	4	20

modulo costs [10]. For the reason that we focus on reducing the costs for user in verification, we don't consider the costs of pairing based schemes. Our scheme takes 9 multiplications and 1 hashing in the blinding step, 2 multiplications and 2 square root computations in the signing step, 1 inverse and 2 multiplications in the unblinding step, and 7 multiplications and 2 hashing in verification. We could see that the quadratic residue based schemes have the least number of modular exponentiations and further more, no modular exponentiations in verification which is applicable in the resource limited environment.

7 Conclusions

We have presented an efficient partially blind signature scheme based on the assumption that finding square roots modulo a composite is intractable. We then gave a formal proof of security including blindness and unforgeability in the random oracle model. Also our approach is easily transformed to give formal proofs for other factorization based schemes.

Notice that unlike some other schemes based on quadratic residue [20], the signature space of our scheme is limited in the quadratic residues of the group. It's easy to expand the signature space by adding a "label" in the signature like other schemes, but the blindness property will be weakened by those "label"s. So we still constructed the scheme based on permutations rather than 4-to-1 mappings.

Acknowledgments

This work is supported by the National Nature Science Foundation of China under grants 61272436 and 61363080, the Chunhui Project of Ministry of Education under Grant Z2012094 and the Item of Information Security Innovation Team of Central Finance.

References

[1] M. Abe and E. Fujisaki, "How to date blind signatures," in *Asiacrypt'96*, vol. 1, no. LNCS 1163, pp. 244–251, 1996.

[2] M. Abe and T. Okamoto, "Provably secure partially blind signatures," in *Crypto'00*, vol. 1, no. LNCS 1880, pp. 271–286, 2000.

[3] T. J. Cao, L. D. Dai, and R. Xue, "A randomized rsa-based partially blind signature scheme for electronic cash," *Computer & Security*, vol. 24, no. 1, pp. 44–49, 2005.

[4] D. Chaum, "Blind signatures for untraceable payments," *Advances in cryptology (CRYPTO'82)*, vol. 1, no. 1, pp. 199–203, 1983.

[5] C. I. Fan and C. L. Lei, "Low-computation partially blind signatures for electronic cash," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E81-A, no. 5, pp. 818–824, 1998.

[6] D. Fang, Na Wang, and C. Liu, "An enhanced rsa-based partially blind signature," *International Conference on Computer and Communication Technologies in Agriculture Engineering*, vol. 1, no. 1, pp. 565–567, 2010.

[7] N. Ferguson, "Single term off-line coins," *Advances in Cryptology (EUROCRYPT'93)*, no. LNCS 765, pp. 318–328, 1994.

[8] M. S. Hwang, C. C. Lee, and Y. C. Lai, "Traceability on low-computation partially blind signatures for electronic cash," *IEICE Fundamentals on Electronics, Communications and Computer Sciences*, vol. E85-A, no. 5, pp. 1181–1182, 2002.

[9] M. S. Hwang, C. C. Lee, and Y. C. Lai, "An untraceable blind signature scheme," *IEICE Transactions on Foundations*, vol. E86-A, no. 7, pp. 1902–1906, 2003.

[10] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*. US: Chapman and Hall/CRC, 2007.

[11] C. C. Lee, M. S. Hwang, and W. P. Yang, "Untractable blind signature schemes based on discrete logarithm problem," *Fundamenta Informaticae*, vol. 55, no. 3-4, pp. 307–320, 2003.

[12] C. C. Lee, M. S. Hwang, and W. P. Yang, "A new blind signature based on the discrete logarithm problem for untraceability," *Applied Mathematics and Computation*, vol. 164, no. 3, pp. 837–841, 2005.

[13] G. Martinet, G. Poupard, P. Sola, "Cryptanalysis of a partially blind signature scheme or how to make $ 100 bills with $ 1 and $ 2 ones," *Proceedings of Financial Cryptography and Data Security*, LNCS vol. 4107, no. 1, pp. 171–176, 2006.

[14] T. Okamoto, "Efficient blind and partially blind signatures without random oracles," in *Theory of Cryptography (TCC'2006)*, vol. LNCS 3876, pp. 80–99, 2006.

[15] D. Pointcheval and J. Stern, "Security arguments for digital signatures and blind signatures," *Journal of Cryptology*, vol. 13, no. 3, pp. 361–396, 2000.

[16] N. M. F. Tahat, S. M. A. Shatnawi, and E. S. Ismail, "A new partially blind signature based on factoring and discrete logarithms," *Journal of Mathematics and Statistics*, vol. 4, no. 2, pp. 124–129, 2008.

[17] M. Tian, Y. Zhu, and Z. Chen, "Two simple attacks on a blind signature scheme," *International Journal of Network Security*, vol. 16, no. 6, pp. 498–500, 2014.

[18] G. K. Verma, "Probable security proof of a blind signature scheme over braid groups," *International Journal of Network Security*, vol. 12, no. 2, pp. 118-120, 2011.

[19] Q. Wu, W. Susilo, Yi Mu, and F. Zhang, "Efficient partially blind signatures with provable security," in *Computational Science and Its Applications (ICCSA'07),* LNCS vol. 4707, pp. 1096–1105, 2007.

[20] Y. Yu, Yi Mu, W. Susilo, Y. Sun, and Y. Ji, "Provably secure proxy signature scheme from factorization," *Mathematical and Computer Modelling*, vol. 55, no. 1, pp. 1160–1168, 2012.

Yi Zhao graduated from Zhejiang University in 2008. He is a master student of Shaanxi Normal University. His main research interests include cryptography and information security.

Qiliang Yang is currently a student at College of Informatics, Shu-Te University. Email:s11115263@stu.edu.tw

Bo Yang received the B.S. degree from Peking University in 1986, and the M.S. and Ph.D. degrees from Xidian University in 1993 and 1999, respectively. From July 1986 to July 2005, he had been at Xidian University. From 2002, he had been a professor of National Key Lab. of ISN in Xidian University. He has served as a program chair for the fourth China Conference on Information and Communications Security in 2005, the vice-chair for ChinaCrypt 2009, and the general co-chair for the Joint Workshop on Information Security since 2010. He is currently a professor and supervisor of Ph.D. candidates at the School of Computer Science, Shaanxi Normal University, a Bai-Ren project special-term professor of Shaanxi Province, and a member of the Council of Chinese Association for Cryptologic Research. His research interests include information theory and cryptography.

Cryptanalysis of Attribute-based Ring Signcryption Scheme

Hu Xiong, Ji Geng, Zhiguang Qin, and Guobin Zhu
(Corresponding author: Hu Xiong)
School of Computer Science and Engineering & University of Electronic Science and Technology of China[1]
No. 4, North Jianshe Road, Chenghua District, Chengdu, Sichuan 610054, China
(Email: xionghu.uestc@gmail.com)
(Received Apr. 8, 2013; revised and accepted Nov. 3, 2014)

Abstract

Signcryption can offer authentication and confidentiality simultaneously with better efficiency than traditional signature-then-encryption approach. Ring signature enables a user to conscribe arbitrarily a group of ring members and sign a message on behalf of the ring (which includes himself) without revealing his real identity. By integrating the notion of signcryption and ring signature, ring signcryption has been initialized to leak secrets in an authenticated and confidential way anonymously. Recently, Guo *et al.* (Guo Z, Li M, Fan X. Attribute-based ring signcryption scheme. *Security and Communication Networks*, vol. 6, no. 6, pp. 790-796, 2013) proposed a ring signcryption scheme in attribute-based cryptography. Furthermore, they claimed that their scheme can satisfy confidentiality and unforgeability in the random oracle model. Unfortunately, by giving concrete attacks, we indicate that Guo *et al.*'s attribute-based ring signcryption scheme doesn't provide confidentiality and unforgeability.

Keywords: Attribute-based cryptography; cryptanalysis; ring signcryption; provable security

1 Introduction

To offer authenticity and confidentiality simultaneously with better efficiency than traditional "sign-then-encrypt" approach, Zheng [24] initially formalized the notion of signcryption. Since Zheng's pioneering work, dozens of signcryption schemes have been proposed following various research lines. Firstly, the existing signcryption scheme can be classified as RSA-based [10], IF-based [20], elliptic curves-based [21, 25], pairing-based [4], lattice-based [12] according to the underlying keys. Secondly, ID-based [3, 5, 6], certificateless [2, 7], self-certified [13] and certificate-based [15] signcryption also have been proposed to simplify the public key certificates in the traditional public key infrastructure. Thirdly, the extensions of signcryption have been proposed by integrating the pure signcryption with other cryptographic primitives, such as ring signcryption [1, 22], group signcryption [11], threshold unsigncryption [14, 23] and proxy signcryption [17, 18]. The survey of signcryption and related applications can be found in [8].

As one of the extension of signcryption, ring signcryption was initially formalized by Huang *et al.* [1] and allows a signer conscripts a group of ring members and signcrypts one message on behalf of the ring without revealing his real identity. Furthermore, the procedure of signcryption does not need the cooperation of other ring members. Thus, ring signcryption can be applied in some concrete applications where authenticity, confidentiality and anonymity receive concern simultaneously. On the other hand, to use biometric-based identities in the Identity-based cryptosystem, attribute-based cryptography has been proposed in 2005 [19]. Recently, Guo et al. [9] introduced ring signcryption in the attribute-based cryptography by integrating the notion of attribute-based ring signature [16] and attribute-based encryption [19]. In an attribute-based signcryption, a signer can get its private key for attributes set ω from a trusted private key generator. Then, this signer can signcrypt message on behalf of a subset $\omega' \subseteq \omega$. Here, all users with this attributes subset ω' can be considered as the ring. After that, a concrete attribute-based ring signcryption based on bilinear pairings has also been suggested in this paper. They claimed that their scheme can achieve unforgeability and confidentiality in the random oracle model. However, in this paper, we show that their scheme cannot provide confidentiality and unforgeability at all by giving concrete attacks. Furthermore, the basic reason behind our attack has also been analyzed.

The rest of this paper is organized as follows. In Section 2, we review the Guo-Li-Fan attribute-based ring signcryption scheme. After that, we explain why their scheme can not provide unforgeability and confidentiality in Sections 3 and 4 respectively. Finally, the conclusions are given in Section 5.

2 Overview of the Guo-Li-Fan Scheme

We describe Guo-Li-Fan's attribute-based ring signcryption scheme [9] as follows. In their scheme, the signer can signcrypt a message on behalf of d attributes, where d will be defined in the **Setup** algorithm. We then review Lagrange interpolation as follows. Given d points $q(1), \cdots, q(d)$ on a $d-1$ degree polynomial, $q(i)$ for any $i \in \mathbb{Z}_p$ can be computed by adopting Lagrange interpolation technique. Assume S be a set in $\mathbb{Z}_p$ with d-elements and the Lagrange coefficient $\Delta_{i,S}$ for $i \in \mathbb{Z}_p$ as follows.

$$\Delta_{i,S}(x) = \prod_{j \in S, j \neq i} \frac{x-i}{i-j}$$

Setup(κ): Given a security parameter κ, the trusted private key generator (PKG) first defines the set of universal attributes $\mathcal{U}$ in $\mathbb{Z}_p$, where $|\mathcal{U}| = l$. After that, a $d-1$ default attributes set from $\mathbb{Z}_p$ is given as $\Omega = \{\Omega_1, \cdots, \Omega_{d-1}\}$. Furthermore, PKG selects a pairing $e : G_1 \times G_1 \rightarrow G_2$ where the order of G_1 and G_2 is prime $p > 2^\kappa$, and a generator g of G_1. PKG then chooses $t_1, \cdots, t_l, t_{l+1}, \cdots, t_{l+d-1} \in \mathbb{Z}_p$ randomly and computes $T_i = g^{t_i}$ where $1 \leq i \leq l+d-1$. PKG also picks $\alpha \in \mathbb{Z}_p$ at random and computes $Y = e(g,g)^\alpha$. Finally, PKG selects three cryptographic hash functions: $H_1 : G_2 \rightarrow \{0,1\}^{|M|} \times \mathbb{Z}_p^* \times G_1$, $H_2 : \{0,1\}^* \rightarrow \mathbb{Z}_p^*$, and $H_3 : \{0,1\}^{|M|} \times \mathbb{Z}_p^* \rightarrow \mathbb{Z}_p^*$, where $|M|$ denotes the length of the ciphertext. The public parameters PK are published as follows:

$$PK = (G_1, G_2, e, g, \{T_i\}_{i=1}^{l+d-1}, Y, H_1, H_2, H_3).$$

The master secret key MK is denoted as $MK = (\alpha, \{t_i\}_{i=1}^{l+d-1})$.

Key Extract(MK, ω): Given the user with attribute set $\omega \subseteq \mathcal{U}$, the PKG generates the private key for ω as follows:

- A $d-1$ degree polynomial $q(x)$ is picked at random such that $q(0) = \alpha$.
- Generates a new attribute set $\hat{\omega} = \omega \cup \Omega$ and computes $D_i = g^{\frac{q(i)}{t_i}}$ for each $i \in \hat{\omega}$.
- Outputs the private key D_i for each $i \in \hat{\omega}$.

Signcryption(m, ω_S, ω_R): To signcrypt a message m to a receiver $\mathcal{R}$, the sender $\mathcal{S}$ follows the steps below:

- Chooses a subset ω_S' with d elements from $\hat{\omega}_S$ (where f attributes $\{i_1, \cdots, i_f\}$ are chosen from ω_S to signcrypt the message, and $d-f$ attributes are chosen from default attributes set Ω).
- The sender $\mathcal{S}$ randomly chooses $r \in \mathbb{Z}_p^*$, and set $s = H_3(m, r)$, $U = g^s$, and $X = Y^s = e(g,g)^{\alpha \cdot s}$. $\mathcal{S}$ then computes $E_i = T_i^s$ for each $i \in \omega_S'$ and for each $j \in \omega_R$.
- Let $\omega_S' = \{1, \cdots, d\}$, and chooses $k \in \omega_S'$ randomly. Defines the elements in set $\omega_S' \cup \omega_R$ to be the ring. For $l \in \omega_S' \cup \omega_R$ and $l \neq k$, chooses $U_l \in \mathbb{Z}_p^*$ at random and computes $h_l = H_2(m, U_l, X, \omega_S' \cup \omega_R, l)$, where $|\omega_S' \cup \omega_R| = n_R + d$. For $l = k$, chooses r_k from $\mathbb{Z}_p^*$ randomly and computes

$$\begin{aligned} U_k &= E_k^{r_k} / \prod_{l \in \omega_S' \cup \omega_R, l \neq k} U_l \cdot E_l \\ &= g^{t_k \cdot r_k \cdot s} / \prod_{l \in \omega_S' \cup \omega_R, l \neq k} U_l \cdot g^{t_l \cdot h_l \cdot s} \\ h_k &= H_2(m, U_k, X, \omega_S' \cup \omega_R, k) \\ V &= E_k^{r_k + h_k} \end{aligned}$$

- Compute $y = (m \| r \| V) \oplus H_1(X)$.
- Finally, the ciphertext CT is denoted as $CT = (y, \omega_S', \omega_R, U, \{U_l\}_{l=1}^{n_R+d}, \{E_i\}_{i=1}^{d}, \{E_i\}_{i=1}^{n_R})$.

UnsigncryptionCT: After receiving the ciphertext CT, $\mathcal{R}$ decrypts the ciphertext as follows.

- For $CT = (y, \omega_S', \omega_R, U, \{U_l\}_{l=1}^{n_R+d}, \{E_i\}_{i=1}^{d}, \{E_i\}_{i=1}^{n_R})$, select a subset $\omega_{R'}$ with d-elements subset from attribute set ω_R.
- Computes

$$\begin{aligned} X' &= \prod_{j \in \omega_{R'}} e(D_j, E_j)^{\Delta_{j,S}(0)} \\ &= \prod_{j \in \omega_{R'}} e(g^{\frac{q(j)}{t_j}}, g^{t_j \cdot s})^{\Delta_{j,S}(0)} \\ &= e(g,g)^{\alpha \cdot s} \end{aligned}$$

  and retrieves m', r', V' as $(m' \| r' \| V') = y \oplus H_1(X')$.
- Computes $s' = H_3(m', r')$ and verifies whether $U \stackrel{?}{=} g^{s'}$ holds or not.
- For $l \in \{1, \cdots, n_R + d\}$, computes $h_l' = H_2(m, U_l, X, \omega_S' \cup \omega_R, l)$ and verifies

$$e(g, \prod_{l=1}^{n_R+d} U_l \cdot g^{t_l \cdot h_l' \cdot s'}) \stackrel{?}{=} e(g, V')$$

  holds or not. If so, $\mathcal{R}$ accepts CT as the valid ring signcryption on the message m'; $\mathcal{R}$ rejects otherwise.

Note that the original scheme in [9] has several typos. In the Step 1 of **Signcryption** algorithm, instead of writing Chooses a subset ω_S' with d elements from $\hat{\omega}_S$, it was written as Chooses a subset ω_S' with d elements from ω_S. In the Step 2 of **Signcryption** algorithm, instead of writing for each $i \in \omega_S'$, it was written as for each $i \in \omega_{S'}$. In the Step 3 of **Signcryption** algorithm, instead of writing $g^{t_k \cdot r_k \cdot s} / \prod_{l \in \omega_S' \cup \omega_R, l \neq k} U_l \cdot g^{t_l \cdot h_l \cdot s}$, it was written as $g^{t_k \cdot r_k \cdot s} / \prod_{l \in \omega_S' \cup \omega_R, l \neq k} U_l \cdot g^{t_l \cdot h_i \cdot s}$. We have corrected these typos to maintain consistency of the scheme.

3 On the Unforgeability of the Guo-Li-Fan Scheme

In this section, we show that the Guo-Li-Fan's certificate-based ring signcryption scheme is not secure against forgery attacks. After receiving a valid ciphertext $CT = (y, \omega'_S, \omega_R, U, \{U_l\}_{l=1}^{n_R+d}, \{E_i\}_{i=1}^{d}, \{E_i\}_{i=1}^{n_R})$, the adversary $\mathcal{A}$ can forge a valid ciphertext $CT^* = (y^*, \omega'_S, \omega_R, U^*, \{U_l^*\}_{l=1}^{n_R+d}, \{E_i^*\}_{i=1}^{d}, \{E_i^*\}_{i=1}^{n_R})$ on message m^* as follows:

- $\mathcal{A}$ randomly chooses $r^* \in \mathbb{Z}_p^*$, and set $s^* = H_3(m^*, r^*)$, $U^* = g^{s^*}$, and $X^* = Y^{s^*} = e(g,g)^{\alpha \cdot s^*}$. $\mathcal{S}$ then computes $E_i^* = T_i^{s^*}$ for each $i \in \omega'_S$ and for each $j \in \omega_R$.
- Let $\omega'_S = \{1, \cdots, d\}$, and chooses $k \in \omega'_S$ randomly. Defines the elements in set $\omega'_S \cup \omega_R$ to be the ring. For $l \in \omega'_S \cup \omega_R$ and $l \neq k$, chooses $U_l^* \in \mathbb{Z}_p^*$ at random and computes $h_l^* = H_2(m^*, U_l^*, X^*, \omega'_S \cup \omega_R, l)$, where $|\omega'_S \cup \omega_R| = n_R + d$. For $l = k$, chooses r_k^* from $\mathbb{Z}_p^*$ randomly and computes

$$\begin{aligned} U_k^* &= (E_k^*)^{r_k^*} / \prod\nolimits_{l \in \omega'_S \cup \omega_R, l \neq k} U_l^* \cdot E_l^* \\ &= g^{t_k \cdot r_k^* \cdot s^*} / \prod\nolimits_{l \in \omega'_S \cup \omega_R, l \neq k} U_l^* \cdot g^{t_l \cdot h_l^* \cdot s^*} \\ h_k^* &= H_2(m^*, U_k^*, X^*, \omega'_S \cup \omega_R, k) \\ V^* &= (E_k^*)^{r_k^* + h_k^*} \end{aligned}$$

- Compute $y^* = (m^* \| r^* \| V^*) \oplus H_1(X^*)$.
- Finally, the ciphertext CT^* on message m^* is denoted as $CT^* = (y^*, \omega'_S, \omega_R, U^*, \{U_l^*\}_{l=1}^{n_R+d}, \{E_i^*\}_{i=1}^{d}, \{E_i^*\}_{i=1}^{n_R})$.

The ring signcryption is correct because of the following:

- X^* can be reconstructed as follows:

$$\begin{aligned} X^* &= \prod\nolimits_{j \in \omega_{R'}} e(D_j, E_j^*)^{\Delta_{j,S}(0)} \\ &= \prod\nolimits_{j \in \omega_{R'}} e(g^{\frac{q(j)}{t_j}}, g^{t_j \cdot s^*})^{\Delta_{j,S}(0)} \\ &= e(g,g)^{\alpha \cdot s^*} \end{aligned}$$

- After retrieving $(m^* \| r^* \| V^*) = y^* \oplus H_1(X^*)$, it is easy to verify that $s^* = H_3(m^*, r^*)$ and $U^* = g^{s^*}$.
- Finally, it is obvious that

$$e(g, \prod_{l=1}^{n_R+d} U_l^* \cdot g^{t_l \cdot h_l^* \cdot s^*}) \stackrel{?}{=} e(g, V^*),$$

where $h_l^* = H_2(m^*, U_l^*, X^*, \omega'_S \cup \omega_R, l)$ for $l \in \{1, \cdots, n_R + d\}$.

The basic reason of our attack works is that the private key of the signer has not been mentioned in the **Signcryption** algorithm. Thus, anyone can generate valid ciphertext on any message on behalf of the ring without the knowledge of any ring member's private key by executing **Signcryption** algorithm directly.

4 On the Confidentiality of the Guo-Li-Fan Scheme

In this section, we show that the Guo-Li-Fan's certificate-based ring signcryption scheme cannot offer confidentiality. After receiving a valid ciphertext $CT = (y, \omega'_S, \omega_R, U, \{U_l\}_{l=1}^{n_R+d}, \{E_i\}_{i=1}^{d}, \{E_i\}_{i=1}^{n_R})$ generated by a user $\mathcal{S}$. Here, ω'_S denotes $\mathcal{S}$'s attributes subset and depicts the ring. Assume that the adversary $\mathcal{A}$ is one of the member in the ring, and therefore $\mathcal{A}$ has the private key D_j for $j \in \omega'_S$ corresponding to the attributes sets ω'_S with d elements according to the definition in [9, 16].

Thus, $\mathcal{A}$ can decrypt the ciphertext CT as follows.

$$\begin{aligned} X &= \prod\nolimits_{j \in \omega'_S} e(D_j, E_j)^{\Delta_{j,S}(0)} \\ &= \prod\nolimits_{j \in \omega'_S} e(g^{\frac{q(j)}{t_j}}, g^{t_j \cdot s})^{\Delta_{j,S}(0)} \\ &= e(g,g)^{\alpha \cdot s}. \end{aligned}$$

Then, $\mathcal{A}$ can obtain m by executing $(m \| r \| V) = y \oplus H_1(X)$.

The basic reason about our attack works is that the blind factor $X = Y^s = e(g,g)^{\alpha \cdot s}$ is computed independent of the receiver's public key. Thus, any of the ring member can decrypt the ciphertext by computing the blinded factor using its own private key.

5 Conclusions

In this paper, we identified security flaws in Guo-Li-Fan's attribute-based ring signcryption scheme proposed in [9]. Our results showed that this signcryption scheme fails to provide unforgeability and confidentiality. Specifically, anyone can forge the valid ciphertext without the knowledge of the ring member's private key. On the other hand, any ring member can decrypt the ciphertext which should only be decrypted by the receiver. Furthermore, the basic reason about our attack has also been analyzed. We remark that it is still an open problem to construct a provably-secure and efficient attribute-based ring signcryption scheme.

Acknowledgments

The authors thank the editors and the anonymous referees for their valuable comments and suggestions. This work is partially supported by National Natural Science Foundation of China under Grant Nos. 61003230, 61370026, 61300191 and 61103206, the Fundamental Research Funds for the Central Universities under Grant No. ZYGX2013J073 and ZYGX2012J077, and the Applied Basic Research Program of Sichuan Province under Grant No. 2014JY0041.

References

[1] M. Barbosa and P. Farshim, "Identity-based ring signcryption schemes: Cryptographic primitives for preserving privacy and authenticity in the ubiquitous world," in *19th International Conference on Advanced Information Networking and Applications (AINA'05)*, pp. 649–654, Taipei, Taiwan, Mar. 2005.

[2] M. Barbosa and P. Farshim, "Certificateless signcryption," in *Proceedings of the 2008 ACM Symposium on Information, Computer and Communications Security (ASIACCS'08)*, pp. 369–372, Tokyo, Japan, Mar. 2008.

[3] P. S. L. M. Barreto, B. Libert, N. McCullagh, and J. J. Quisquater, "Efficient and provably-secure identity-based signatures and signcryption from bilinear maps," in *11th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT'05)*, pp. 515–532, Chennai, India, Dec. 2005.

[4] X. Boyen, "Multipurpose identity-based signcryption: a swiss army knife for identity-based cryptography," in *Proceedings of Advances in Cryptology (CRYPTO'03)*, pp. 383–399, Santa Barbara, California, USA, Aug. 2003.

[5] H. Chen, Y. Li, and J. Ren, "A practical identity-based signcryption scheme," *International Journal of Network Security*, vol. 15, no. 6, pp. 484–489, 2013.

[6] L. Chen and J. M. Lee, "Improved identity-based signcryption," in *8th International Workshop on Theory and Practice in Public Key Cryptography (PKC'05)*, pp. 362–379, Les Diablerets, Switzerland, Jan. 2005.

[7] L. Cheng and Q. Wen, "An improved certificateless signcryption in the standard model," *International Journal of Network Security*, vol. 17, no. 3, pp. 229-237, 2015.

[8] A. W. Dent and Y. Zheng, *Practical Signcryption*, Springer, 2010.

[9] Z. Guo, M. Li, and X. Fan, "Attribute-based ring signcryption scheme," *Security and Communication Networks*, vol. 6, no. 6, pp. 790–796, 2013.

[10] J. M. Lee and W. Mao, "Two birds one stone: signcryption using RSA," in *Proceedings of Topics in Cryptology (CT-RSA'03)*, LNCS, vol. 2612, pp. 211–226, Apr. 2003.

[11] D. J. Kwak, S. J. Moon, G. Wang, and R. H. Deng, "A secure extension of the kwakc-moon group signcryption scheme," *Computers & Security*, vol. 25, no. 6, pp. 435–444, 2006.

[12] F. Li, F. T. B. Muhaya, M. K. Khan, and T. Takagi, "Lattice-based signcryption," *Concurrency and Computation: Practice and Experience*, vol. 25, no. 14, pp. 2112–2122, 2013.

[13] F. Li, X. Xin, and Y. Hu, "A pairing-based signcryption scheme using self-certified public keys," *International Journal of Computers and Applications*, vol. 29, no. 3, pp. 278–282, 2007.

[14] F. Li, X. Xin, and Y. Hu, "Id-based signcryption scheme with (t, n) shared unsigncryption," *International Journal of Network Security*, vol. 3, no. 2, pp. 155–159, 2006.

[15] J. Li, X. Huang, M. Hong, and Y. Zhang, "Certificate-based signcryption with enhanced security features," *Computers and Mathematics with Applications*, vol. 64, no. 6, pp. 1587–1601, 2012.

[16] J. Li and K. Kim, "Attribute-based ring signatures," *Cryptology ePrint Archive*, 2008.

[17] H. Y. Lin, T. S. Wu, S. K. Huang, and Y. S. Yeh, "Efficient proxy signcryption scheme with provable cca and cma security," *Computers & Mathematics with Applications*, vol. 60, no. 7, pp. 1850–1858, 2010.

[18] C. Pan, S. Li, Q. Zhu, C. Wang, and M. Zhang, "Notes on proxy signcryption and multi-proxy signature schemes," *International Journal of Network Security*, vol. 17, no. 1, pp. 29-33, 2015

[19] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *Advances in Cryptology (EUROCRYPT'05)*, pp. 457–473, Aarhus, Denmark, May 2005.

[20] R. Steinfeld and Y. Zheng, "A signcryption scheme based on integer factorization," in *Third International Workshop on Information Security (ISW'00)*, pp. 308–322, Wollongong, Australia, Dec. 2000.

[21] M. Toorani and A. A. B. Shirazi, "Cryptanalysis of an elliptic curve-based signcryption scheme," *International Journal of Network Security*, vol. 10, no. 1, pp. 51–56, 2010.

[22] H. Xiong, J. Hu, and Z. Chen, "Security flaw of an ecc-based signcryption scheme with anonymity," *International Journal of Network Security*, vol. 15, no. 4, pp. 317–320, 2013.

[23] Bo Yang, Y. Yu, F. Li, and Y. Sun, "Provably secure identity-based threshold unsigncryption scheme," in *4th International Conference on Autonomic and Trusted Computing (ATC'07)*, pp. 114–122, Hong Kong, China, July 2007.

[24] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) $\ll$ cost (signature) + cost(encryption)," in *Proceedings of Advances in Cryptology (CRYPTO'97)*, pp. 165–179, Santa Barbara, California, USA, Aug. 1997.

[25] Y. Zheng and H. Imai, "How to construct efficient signcryption scheme on elliptic curves," *Information Processing Letters*, vol. 68, no. 5, pp. 277–233, 1998.

Hu Xiong is an associate professor at University of Electronic Science and Technology of China (UESTC). He received his Ph.D degree from UESTC in 2009. His research interests include: cryptographic protocol, and network security.

Ji Geng is a professor in the School of Computer Science and Engineering, University of Electronic Science and Technology of China. He received his Ph.D degree from UESTC in 2014. His research interests include:

information security and system software.

Zhiguang Qin is the dean and professor in the School of Computer Science and Engineering, UESTC. He received his PH.D. degree from UESTC in 1996. His research interests include: information security and computer network.

Guobin Zhu is an assistant professor at University of Electronic Science and Technology of China (UESTC). He received his Ph.D degree from UESTC in 2014. His research interests include: network security and applied cryptography.

Guide for Authors
International Journal of Network Security

IJNS will be committed to the timely publication of very high-quality, peer-reviewed, original papers that advance the state-of-the art and applications of network security. Topics will include, but not be limited to, the following: Biometric Security, Communications and Networks Security, Cryptography, Database Security, Electronic Commerce Security, Multimedia Security, System Security, etc.

1. Submission Procedure

Authors are strongly encouraged to submit their papers electronically by using online manuscript submission at http://ijns.femto.com.tw/.

2. General

Articles must be written in good English. Submission of an article implies that the work described has not been published previously, that it is not under consideration for publication elsewhere. It will not be published elsewhere in the same form, in English or in any other language, without the written consent of the Publisher.

2.1 Length Limitation:

All papers should be concisely written and be no longer than 30 double-spaced pages (12-point font, approximately 26 lines/page) including figures.

2.2 Title page

The title page should contain the article title, author(s) names and affiliations, address, an abstract not exceeding 100 words, and a list of three to five keywords.

2.3 Corresponding author

Clearly indicate who is willing to handle correspondence at all stages of refereeing and publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address.

2.4 References

References should be listed alphabetically, in the same way as follows:
For a paper in a journal: M. S. Hwang, C. C. Chang, and K. F. Hwang, ``An ElGamal-like cryptosystem for enciphering large messages," *IEEE Transactions on Knowledge and Data Engineering*, vol. 14, no. 2, pp. 445--446, 2002.

For a book: Dorothy E. R. Denning, *Cryptography and Data Security*. Massachusetts: Addison-Wesley, 1982.

For a paper in a proceeding: M. S. Hwang, C. C. Lee, and Y. L. Tang, ``Two simple batch verifying multiple digital signatures," in *The Third International Conference on Information and Communication Security (ICICS2001)*, pp. 13--16, Xian, China, 2001.

In text, references should be indicated by [number].

2.5 Author benefits

No page charge is made.

Subscription Information

Individual subscriptions to IJNS are available at the annual rate of US$ 200.00 or NT 6,000 (Taiwan). The rate is US$600.00 or NT 19,800 (Taiwan) for institutional subscriptions. Price includes surface postage, packing and handling charges worldwide. Please make your payment payable to "Femto Technique Co., LTD." For detailed information, please refer to http://ijns.femto.com.tw or Email to ijns.publishing@gmail.com.