2011 Seventh International Conference on Computational Intelligence and Security

Attribute-Based Authenticated Key Exchange Protocol with General Relations

Hao Wang, Qiuliang Xu†, Han Jiang, Rui Li
School of Computer Science and Technology
Shandong University
Jinan, China
whatsdu@gmail.com, xuqiuliang@sdu.edu.cn, jianghan@sdu.edu.cn, ruili@mail.sdu.edu.cn

***Abstract*—In this paper, we present a two-party attribute-based authenticated key exchange scheme for a wide class of relations, which are specified by non-monotone access structures combined with inner-produce relations. We prove the security of our scheme under the decisional linear (DLIN) assumption, without random oracle, in our ABCK model, which is a natural extension of the CK model.**

Keywords- attribute-based cryptosystem; authenticated key exchange; ABCK model; without random oracle; non-monotone access structure

I. INTRODUCTION

The aim of authenticated key exchange (AKE) is to share a common session key between the authenticated parties. Various variants of AKE are used for our daily life. As a new variant of AKE, the attribute-based AKE (ABAKE) is recently studied [1-5]. In ABAKE schemes, the authentication condition is different from other AKE variants. While parties (the initiator and the responder) in a session authenticate each other by their identities in most of AKE variants, parties authenticate each other by their attributes in ABAKE schemes. That is, the key generation center (KGC) issues the static secret key to a party by using the master secret key according to the attributes of the party in advance and parties specify their policies (i.e., the condition which the peer is expected to satisfy) respectively. If the attribute of a party satisfies the policy of the peer and vice versa, the common session key is established. Since attributes can contain an identity, the ABAKE is a generalization of ID-based AKE schemes [6]. The ABAKE is useful in the situation that some sensitive information (e.g., medical history) is sent with the secure channel established by some AKE scheme. Then, parties may hope to hide their identities from the peer of the session though the peer is needed to be a qualified registered person. By using the ABAKE, parties can establish the secure channel with a qualified registered person without revealing their identities.

However, to date, the existing ABAKE schemes [1-5] only support monotone access structure [7]. Most of them were proved in the random oracle model. In this paper, we proposed a new ABAKE scheme, which supported general relations (non-monotone access structure combined with inner-produce relations), and was proved in an extensive CK model without random oracle. This idea is inspired by the related work [8] and [9].

II. PRELIMINARIES

A. Dual Pairing Vector Spaces by Direct Product of Symmetric Pairing Groups

In this paper, we concentrate on the symmetric version of dual pairing vector spaces [10, 11]. The definitions of "Symmetric bilinear pairing groups" and "Dual pairing vector spaces" were given in [9, Definition 1, Definition 2]. Therefore, we will not repeat these definitions, and use the same symbols in this paper.

B. Span Programs and Non-Monotone Access Structures

The definition of "Span Program" was given in [7], and the definition of "Inner-products of Attribute Vectors and Access Structures" was given in [9, Definition 4].

C. Strongly Unforgeable One-Time Signature

The definition of "Strongly Unforgeable One-Time Signature" was given in [9, Definition 14].

D. Decisional Linear (DLIN) Assumption

Definition 1 [9] (DLIN: Decisional Linear Assumption) The DLIN problem is to guess $\beta \in \{0,1\}$, given $(param_{\mathbb{G}}, G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta) \xleftarrow{R} \mathcal{G}_\beta^{DLIN}(1^\lambda)$, where

$$\mathcal{G}_\beta^{DLIN}(1^\lambda): \ param_{\mathbb{G}} := (q, \mathbb{G}, \mathbb{G}_T, G, e) \xleftarrow{R} \mathcal{G}_{bpg}(1^\lambda),$$
$$\kappa, \delta, \xi, \sigma \xleftarrow{U} \mathbb{F}_q, \ Y_0 = (\delta + \sigma)G, \ Y_1 \xleftarrow{U} G,$$
$$\text{return } (param_{\mathbb{G}}, G, \xi G, \kappa G, \delta\xi G, \sigma\kappa G, Y_\beta),$$

for $\beta \xleftarrow{U} \{0,1\}$. For a probabilistic machine $\mathcal{E}$, we define the advantage of $\mathcal{E}$ for the DLIN problem as:

$$Adv_{\mathcal{E}}^{DLIN}(\lambda) := |\Pr[\mathcal{E}(1^\lambda, \rho) \to 1 \mid \rho \xleftarrow{R} \mathcal{G}_0^{DLIN}(1^\lambda)] - \Pr[\mathcal{E}(1^\lambda, \rho) \to 1 \mid \rho \xleftarrow{R} \mathcal{G}_1^{DLIN}(1^\lambda)]|$$

The DLIN assumption is: For any probabilistic polynomial-time adversary $\mathcal{E}$, the advantage $Adv_{\mathcal{E}}^{DLIN}(\lambda)$ is negligible in λ.

E. Functional Authenticated Key Exchange Scheme

An ABAKE scheme consists of the following algorithm [5]. We denote a party by *P* and his associated set of attribute

† Corresponding author.

978-0-7695-4584-4/11 $26.00 © 2011 IEEE
DOI 10.1109/CIS.2011.203

IEEE computer society

by Γ_P. The party P and other parties are modeled as a probabilistic polynomial-time Turing machine.

Setup$(1^\lambda, \vec{n}) \to (MPK, MSK)$: This is a randomized algorithm that takes as input security parameter and format $\vec{n}$ of attributes. It outputs the master public key MPK and a master secret key MSK.

KeyGen$(MSK, MPK, \Gamma_P) \to SK_{\Gamma_P}$: This is a randomized algorithm that takes as input a set of attributes Γ_P, MPK and MSK. It outputs a decryption key SK_{Γ_P} corresponding to Γ_P.

Key Exchange: The party A and B share a session key by performing the following n-pass protocol. A (resp. B) selects a policy $\mathbb{S}_A := (M_A, \rho_A)$ (resp. $\mathbb{S}_B$) as an access structure, by which the attributes of his expectant communicating partner should be accepted, respectively, i.e., $\Gamma_B \in \mathbb{S}_A$, $\Gamma_A \in \mathbb{S}_B$

A start the protocol by computing the 1st message m_1 by the algorithm **Message**, that takes the master public key MPK, the set of attribute Γ_A, the long term key SK_{Γ_A} and the policy $\mathbb{S}_A$, and outputs 1st message m_1. A sents m_1 to the other party B.

For $i = 2,..,n$, upon receiving the $(i-1)$th message m_{i-1}, the party P ($P = A$ or B) computes the ith message by algorithm **Message**, that takes the master public key MPK, the set of attribute Γ_P, the long term key SK_{Γ_P}, the policy $\mathbb{S}_P$ and the sent and received message $m_1,...,m_{i-1}$, and outputs ith message m_i, i.e.,

Message$(MPK, \Gamma_P, SK_{\Gamma_P}, \mathbb{S}_P, m_1,...,m_{i-1}) \to m_i$.

The party P sends m_i to the other user $\bar{P}$ $(\bar{P} = B \text{ or } A)$.

Upon receiving or after sending the final nth message m_n, P computes a session key by algorithm **SessionKey**, that takes the master public key MPK, the set of attributes Γ_P, the static secret key SK_{Γ_P}, the policy $\mathbb{S}_P$ and the sent and received messages $m_1,...,m_n$, and outputs an session key K, i.e.,

SessionKey$(MPK, \Gamma_P, SK_{\Gamma_P}, \mathbb{S}_P, m_1,...,m_n) \to K$

Both parties P and $\bar{P}$ can compute the same session key if and only if $\Gamma_P \in \mathbb{S}_{\bar{P}}$ and $\Gamma_{\bar{P}} \in \mathbb{S}_P$.

III. SECURITY MODEL

In this section, we introduce a CK security model for the ABAKE. Our attribute-based CK (ABCK) model is an extension of the CK security model for conventional AKE [12, 13] to the ABAKE.

The proposed ABCK model is different from the original CK model in the following points:

1) the session is identified by a set of attributes Γ_P of party P

2) freshness conditions for queries to corrupt the long term keys are different

Session. An invocation of a protocol is called a session. A session is activated with an incoming message of the form $(\mathcal{I}, \Gamma_P, \Gamma_{\bar{P}})$ or $(\mathcal{R}, \Gamma_{\bar{P}}, \Gamma_P, m_1)$, where $\mathcal{I}$ and $\mathcal{R}$ with role identifiers, P and $\bar{P}$ with user identifiers. If P was activated with $(\mathcal{I}, \Gamma_P, \Gamma_{\bar{P}})$, then P is called the session initiator. If $\bar{P}$ was activated with $(\mathcal{R}, \Gamma_{\bar{P}}, \Gamma_P, m_1)$, then $\bar{P}$ is called the session responder. After activated with an incoming message of forms $(\mathcal{I}, \Gamma_P, \Gamma_{\bar{P}}, m_1,...,m_{k-1})$ from the responder $\bar{P}$, the initiator P outputs m_k, then may be activated next by an incoming message of forms $(\mathcal{I}, \Gamma_P, \Gamma_{\bar{P}}, m_1,...,m_{k+1})$ from the responder $\bar{P}$. After activated by an incoming message of forms $(\mathcal{R}, \Gamma_{\bar{P}}, \Gamma_P, m_1,...,m_k)$ from the initiator P, the responder $\bar{P}$ outputs m_{k+1}, then may be activated next by an incoming message of the forms $(\mathcal{R}, \Gamma_{\bar{P}}, \Gamma_P, m_1,...,m_{k+2})$ from the initiator P. Upon receiving or after sending the final nth message m_n, both parties P and $\bar{P}$ computes a session key K.

In the analysis of the protocols in this paper, we define the session-id as the concatenation of the message sent and received by the party, i.e. $sid = m_1 || ... || m_n$. If P is the initiator of a session, the session is identified by $(\mathcal{I}, \Gamma_P, \Gamma_{\bar{P}}, sid)$. If $\bar{P}$ is the responder of a session, the session is identified by $(\mathcal{R}, \Gamma_{\bar{P}}, \Gamma_P, sid)$. We say that a session is completed if a session key is computed in the session. The matching session of a completed session $(\mathcal{I}, \Gamma_P, \Gamma_{\bar{P}}, sid)$ is completed session with identifier $(\mathcal{R}, \Gamma_{\bar{P}}, \Gamma_P, sid)$ and vice versa.

Adversary. The adversary $\mathcal{A}$ that is modeled as a probabilistic polynomial-time Turing machine controls all communications between parties including the session activation by performing the following queries.

- Sent(message): The message has one of the following forms: $(\mathcal{I}, \Gamma_P, \Gamma_{\bar{P}}, m_1,...,m_k)$ or $(\mathcal{R}, \Gamma_{\bar{P}}, \Gamma_P, m_1,...,m_{k+1})$. The adversary obtains the response from the party.
- Session-Expiration $(\mathcal{I}, \Gamma_P, \Gamma_{\bar{P}}, sid)$ (or $(\mathcal{R}, \Gamma_{\bar{P}}, \Gamma_P, sid)$): This query is used for defining forward secrecy and erases from memory the session key on a completed session. The session is thereafter said to be expired.

Revealing secret information of parties is captured via the following queries.

- Corrupt(Γ_P): With this query $\mathcal{A}$ learns the long term key corresponding to the set of attributes Γ_P.
- Session-Key $(\mathcal{I},\Gamma_P,\Gamma_{\bar{P}},sid)$ (or $(\mathcal{R},\Gamma_{\bar{P}},\Gamma_P,sid)$): This query return the session key (if any) accepted by P ($\bar{P}$) during a given session sid with $\bar{P}$ (P).
- Session-State $(\mathcal{I},\Gamma_P,\Gamma_{\bar{P}},sid)$ (or $(\mathcal{R},\Gamma_{\bar{P}},\Gamma_P,sid)$): This query returns all the internal state information of party P ($\bar{P}$) associated to a particular session sid with $\bar{P}$ (P), but not includes the long term keys of P ($\bar{P}$).
- Establish(P,Γ_P): This query allows the adversary to register a static public key corresponding to the set of attributes Γ_P on behalf of the party P; the adversary totally controls the party. If a party is established by Establish(P,Γ_P) query issued by the adversary, then we call the party P dishonest. If not, we call the party honest.

Freshness. For the security definition, we need the notion of freshness.

Definition 2 (**Freshness**). Let $(\mathcal{I},\Gamma_{P^*},\Gamma_{\bar{P}^*},sid^*)$ (or $(\mathcal{R},\Gamma_{\bar{P}^*},\Gamma_{P^*},sid^*)$) be a completed session between an honest user P^* with the set of attribute Γ_{P^*} and $\bar{P}^*$ with $\Gamma_{\bar{P}^*}$. If the matching session exists, then let $(\mathcal{R},\Gamma_{\bar{P}^*},\Gamma_{P^*},sid^*)$ (or $(\mathcal{I},\Gamma_{P^*},\Gamma_{\bar{P}^*},sid^*)$) be the matching session. We say $(\mathcal{I},\Gamma_{P^*},\Gamma_{\bar{P}^*},sid^*)$ (or $(\mathcal{R},\Gamma_{\bar{P}^*},\Gamma_{P^*},sid^*)$) to be fresh if none of the following conditions hold:

1) a session-state or session-key query to this session or to the matching session

2) a corrupt(Γ) query s.t. $\Gamma\in\mathbb{S}_{P^*}$ or $\Gamma\in\mathbb{S}_{\bar{P}^*}$ before the session exposes at that partner.

Security Experiment. For our security definition, we consider the following security experiment. Initially, adversary $\mathcal{A}$ is given a set of honest users, and makes any sequence of the queries described above. During the experiment, $\mathcal{A}$ makes the following query.

- Test $(\mathcal{I},\Gamma_{P^*},\Gamma_{\bar{P}^*},sid^*)$ (or $(\mathcal{R},\Gamma_{\bar{P}^*},\Gamma_{P^*},sid^*)$): Here, the test session must be a fresh session. Select random bit $b\in\{0,1\}$, and return the session key held by $(\mathcal{I},\Gamma_{P^*},\Gamma_{\bar{P}^*},sid^*)$ (or $(\mathcal{R},\Gamma_{\bar{P}^*},\Gamma_{P^*},sid^*)$) if $b=0$, and return a random key if $b=1$.

The experiment continues until $\mathcal{A}$ makes a guess b'. The adversary wins the game if the test session is still fresh and if $\mathcal{A}$'s guess is correct, i.e., $b'=b$. The advantage of $\mathcal{A}$ in the experiment with the FAKE scheme Π is defined as

$$\mathbf{Adv}_{\mathcal{A}}^{\mathrm{SK}} = |2\Pr[b=b']-1|$$

We define the security as follow.

Definition 3 (**ABCK Security**). A key establishment protocol Π is called session key (SK-) secure with perfect forward secrecy (PFS) if the following properties are satisfied for any adversary $\mathcal{A}$.

1) If two uncorrupted honest parties completing matching sessions and $\Gamma_P\in\mathbb{S}_{\bar{P}}$ and $\Gamma_{\bar{P}}\in\mathbb{S}_P$ hold, then, except with negligible probability, they both compute the same session key.

2) For any probabilistic polynomial-time adversary $\mathcal{A}$, $\mathbf{Adv}_{\mathcal{A}}^{\mathrm{SK}}$ is negligible.

Hence the second requirement will be met if the advantage of $\mathcal{A}$ is negligible. Canetti and Krawczyk also provide a definition of SK-security without PFS. The only difference with respect to the above definition is that now the adversary is not allowed to expire sessions.

Krawczyk [14] showed that the forward secrecy in the usual sense cannot be achieved in a two-pass protocol such as the ones that we consider. Therefore we restrict our concern to what Krawczyk calls weak forward secrecy (WFS), in which all the internal state of both partners will be erased when the session is expired. Particularity, in this paper we only consider the condition of partial weak forward secrecy (p-WFS), where we restricted the adversary to corrupt at most one party to the test session. To achieve WFS, we only need append an independent Diffie-Hellman exchange protocol concurrently, just like [8].

The original CK model dose not consider key compromise impersonation (KCI) attacks, where the adversary, after compromising the long-term key of party A, engages in a successful protocol run with A posing as a third party B, i.e., A accepts a session key in the belief that is shared with B, when in fact is shared with the adversary. Thus in a KCI attack there is no matching session to the test session. To model KCI resistance for our protocol we modify the definition of security to allow the adversary to corrupt the owner Γ_A of the test session (Γ_A,Γ_B,s).

IV. Review of Functional Encryption Scheme

Before presenting our ABAKE scheme, we first review the functional encryption scheme, introduced by Okamoto and Takashima in [9]. Avoiding duplication, we only describe the **Setup** and **KeyGen** algorithm here, the detail of **Encryption** and **Decryption** algorithm was given in [9, section 8.2].

A. CP-FE Scheme

Define function $\tilde{\rho}:\{1,...,l\}\to\{1,...,d\}$ by $\tilde{\rho}(i):=t$ if $\rho(i)=(t,\vec{v})$ or $\rho(i)=\neg(t,\vec{v})$, where ρ is given in access structure $\mathbb{S}:=(M,\rho)$. In the proposed scheme, we assume that $\tilde{\rho}$ is injective for $\mathbb{S}:=(M,\rho)$.

In the description of the scheme, we assume that an input vector, $\vec{x}_t:=(x_{t,1},...,x_{t,n_t})$ is normalized such that $x_{t,1}:=1$. (If $x_{t,1}$ is not normalized, change it to a normalized one by $(1/x_{t,1})\cdot\vec{x}_t$, assuming that $x_{t,1}$ is non-zero). In addition, we

assume that input vector $\vec{v}_t := (v_{i,1},...,v_{i,n_t})$ satisfies that $v_{i,n_t} \neq 0$.

Random dual bases generator $\mathcal{G}_{\mathrm{ob}}(1^\lambda, \vec{n})$ is defined at the end of Section 2. We refer to Section 1.3 for notations on DPVS, e.g., $(x_1,...,x_N)_{\mathbb{B}}$, $(y_1,...,y_N)_{\mathbb{B}^*}$ for $x_i, y_i \in \mathbb{F}_q$, and $\vec{e}_{t,j}$.

For simplicity, we assume verification key *verk* is an element in $\mathbb{F}_q$. (We can extend the construction to verification key over any distribution $\mathbb{D}$ by first hashing *verk* using a collision resistant hash $H: \mathbb{D} \to \mathbb{F}_q$.)

Setup$(1^\lambda, \vec{n} := (d; n_1,...n_d))$:

$n_{d+1} := 2$, $n := (d+1; \vec{n}' := \{n_t\}_{t=1,...,d+1})$,

$(param_{\vec{n}'}, \{\mathbb{B}_t, \mathbb{B}_t^*\}_{t=0,...,d+1}) \leftarrow \mathcal{G}_{\mathrm{ob}}(1^\lambda, \vec{n}')$,

$\hat{\mathbb{B}}_0 := (\mathbf{b}_{0,1}, \mathbf{b}_{0,3}, \mathbf{b}_{0,5})$, $\hat{\mathbb{B}}_t := (\mathbf{b}_{t,1},..., \mathbf{b}_{t,n_t}, \mathbf{b}_{t,3n_t+1})$ for $t = 1,..., d+1$,

$\hat{\mathbb{B}}_0^* := (\mathbf{b}_{0,1}^*, \mathbf{b}_{0,3}^*, \mathbf{b}_{0,4}^*)$, $\hat{\mathbb{B}}_t^* := (\mathbf{b}_{t,1}^*,..., \mathbf{b}_{t,n_t}^*, \mathbf{b}_{t,2n_t+1}^*,..., \mathbf{b}_{t,3n_t}^*)$ for $t = 1,..., d+1$,

$MPK := (1, param_{\vec{n}}, \{\hat{\mathbb{B}}_t\}_{t=0,...,d+1})$, $MSK := \{\mathbb{B}_t^*\}_{t=0,...,d+1}$.

return MPK, MSK.

KeyGen(MPK, MSK, Γ_P), where

$\Gamma_P := \{(x_{P,t,1},..., x_{P,t,n_t}) \in \mathbb{F}_q^{n_t} \setminus \{\vec{0}\} \mid 1 \le t \le d, x_{P,t,1} := 1\}$:

$\delta_P, \varphi_{P,0} \leftarrow \mathbb{F}_q$, $\vec{\varphi}_{P,t} \leftarrow \mathbb{F}_q^{n_t}$ such that $(t, \vec{x}_{P,t}) \in \Gamma_P$,

$\vec{\varphi}_{P,d+1,1}, \vec{\varphi}_{P,d+1,2} \leftarrow \mathbb{F}_q^2$

$\mathbf{k}_{P,0} := (\delta_P, 0, 1, \varphi_{P,0}, 0)_{\mathbb{B}_0^*}$,

$\mathbf{k}_{P,t}^* := (\overbrace{\delta_P \vec{x}_{P,t}}^{n_t}, \overbrace{0^{n_t}}^{n_t}, \overbrace{\vec{\varphi}_{P,t}}^{n_t}, \overbrace{0}^{1})_{\mathbb{B}_t^*}$ for $(t, \vec{x}_{P,t}) \in \Gamma_P$,

$\mathbf{k}_{P,d+1,1}^* := (\delta_P(1,0), 0^2, \vec{\varphi}_{P,d+1,1}, 0)_{\mathbb{B}_{d+1}^*}$,

$\mathbf{k}_{P,d+1,2}^* := (\delta_P(1,0), 0^2, \vec{\varphi}_{P,d+1,2}, 0)_{\mathbb{B}_{d+1}^*}$,

$SK_{\Gamma_P} := (\Gamma, \mathbf{k}_{P,0}, \{\mathbf{k}_{P,t}^*\}_{(t,\vec{x}_{P,t}) \in \Gamma_P}, \mathbf{k}_{P,d+1,1}^*, \mathbf{k}_{P,d+1,2}^*)$

return SK_{Γ_P}.

Encryption$(MPK, m, \mathbb{S} := (M, \rho)) \to ct_{\mathbb{S}}$

Decryption$(MPK, sk_\Gamma, ct_{\mathbb{S}}) = m$

B. Security

Theorem 1 [9, section 8.3] The above CP-FE scheme is adaptively payload-hiding against chosen-ciphertext attacks under the DLIN assumption provide that the underlying signature scheme (Gen, Sig, Ver) is a strongly unforgeable one-time signature scheme.

V. Our Attribute-based Authenticated Key Exchange Protocol

The **Setup** and **KeyGen** algorithms are same as those in the CP-FE system, we describe the **Key Exchange** algorithm in Fig. 1.

- $\{\mathrm{Expd}_K(\cdot)\}_{K \in \mathbb{U}_1} : \{0,1\}^* \to \mathbb{U}_2$ is a pseudorandom function family, (as described in [8, Definition 3]),
- $\mathrm{Exct}_\kappa(\cdot): \mathbb{K} \to \mathbb{U}_1$ is chosen uniformly at random from a strong (m, ε) -strong randomness extractor for appropriate m and ε (as described in [8, Definition 2]).

VI. Security of Our FAKE Scheme

Theorem 2 The proposed ABAKE scheme is SK-security (with partial WPS and KCI resistance) under the DLIN assumption provide that the underlying signature scheme (Gen, Sig, Ver) is a strongly unforgeable one-time signature scheme.

Proof. We omit the proof of this theorem, because it can be proved using the method introduced in [8, Theorem 1]. (The CP-FE scheme is obviously a CCA-secure functional-KEM.)

VII. Conclusions

In this paper, we present a new two-party attribute-based authenticated key exchange scheme for a wide class of relations, which are specified by non-monotone access structures combined with inner-produce relations. We prove the security of our scheme under the decisional linear (DLIN) assumption, without random oracle, in our ABCK model, which is a natural extension of the CK model.

Acknowledgment

This work is supported by the National Natural Science Foundation of China under Grant No.61173139, China Post-doctoral Science Foundation under Grant No. 20090461220, the National Natural Science Foundation of Shandong Province under Grant No.ZR2010FM045, Graduate Independent Innovation Foundation of Shandong University under Grant No. yzc09043 and No. yzc10066, Independent Innovation Foundation of Shandong University.

References

[1] H. Wang, Q. Xu, T. Ban, "A Provably Secure Two-Party Attribute-Based Key Agreement Protocol," IIH-MSP 2009, IEEE Computer Society, Sep. 2009, pp. 1042−1045, doi: 10.1109/IIH-MSP.2009.92.

[2] H. Wang, Q. Xu, X. Fu, "Revocable Attribute-based Key Agreement Protocol without Random Oracles," Journal of Networks, vol. 4, Oct. 2009, pp. 787−794, doi: doi:10.4304/jnw.4.8.787-794.

[3] M.C. Gorantla, C. Boyd, J.M.G. Nieto, "Attribute-based Authenticated Key Exchange," ACISP 2010, LNCS, vol. 6168, Springer, 2010, pp. 300−317, doi: 10.1007/978-3-642-14081-5_19.

[4] J. Birkett, D. Stebila, "Predicate-Based Key Exchange," ACISP 2010, LNCS, vol. 6168, Springer, 2010, pp. 282−299, doi: 10.1007/978-3-642-14081-5_18.

[5] K. Yoneyama, "Strongly Secure Two-Pass Attribute-Based Authenticated Key Exchange," Pairing 2010, LNCS, vol. 6487, Springer, Springer, 2010, pp. 147-166, doi: 10.1007/978-3-642-17455-1_10.

[6] N.P. Smart, "An Identity Based Authenticated Key Agreement Protocol Based on the Weil Pairing," Electronics Letters, vol. 38, Jun. 2002, pp. 630−632, doi: 10.1049/el:20020387.

[7] A. Beimel, "Secure schemes for secret sharing and key distribution," PhD thesis, Israel Institute of Technology, Technion, Haifa, Isreal, 1996.

[8] C. Boyd, Y. Cliff, J.M.G. Nieto, K.G. Paterson, "Efficient One-Round Key Exchange in the Standard Model," ACISP 2008, LNCS, vol. 5107, Springer, 2008, pp. 69-83, doi: 10.1007/978-3-540-70500-0_6.

[9] T. Okamoto, K. Takashima, "Fully Secure Functional Encryption with General Relations from the Decisional Linear Assumption," CRYPTO 2010, LNCS, vol. 6223, Springer, 2010, pp. 191-208, doi: 10.1007/978-3-642-14623-7_11.

[10] T. Okamoto, K. Takashima, "Homomorphic Encryption and Signatures from Vector Decomposition," Pairing 2008, LNCS, vol. 5209, Springer, 2008, pp. 57-74, doi: 10.1007/978-3-540-85538-5_4.

[11] T. Okamoto, K. Takashima, "Hierarchical Predicate Encryption for Inner-Products," ASIACRYPT 2009, LNCS, vol. 5912, Springer, 2009, 214-231, doi: 10.1007/978-3-642-10366-7_13.

[12] M. Bellare, R. Canetti, H. Krawczyk, "A Modular Approach to the Design and Analysis of Authentication and Key Exchange Protocols (Extended Abstract)," STOC 1998, ACM, 1998, pp. 419-428, doi: 10.1145/276698.276854.

[13] R. Canetti, H. Krawczyk, "Analysis of Key-Exchange Protocols and Their Use for Building Secure Channels," EUROCRYPT 2001, LNCS, vol. 2045, Springer, 2001, pp. 453-474, doi: 10.1007/3-540-44987-6_28.

[14] H. Krawczyk, "HMQV: A High-Performance Secure Diffie-Hellman Protocol," CRYPTO 2005, LNCS, vol. 3621, Springer, 2005, pp. 546-566, doi: 10.1007/11535218_33.

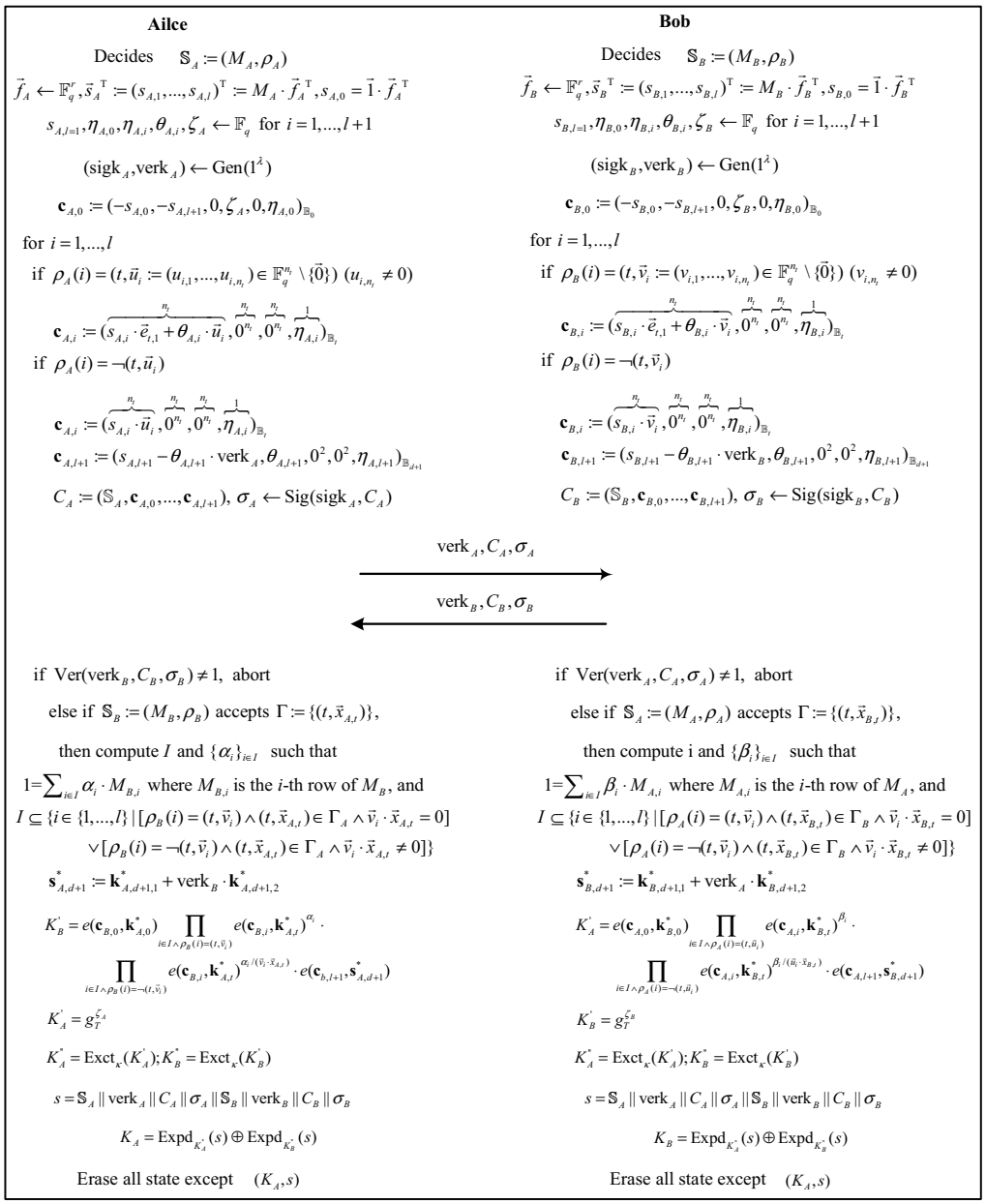

Figure 1. The **Key Exchange** algorithm of our ABAKE.