A Certificateless Group Authenticated Key Agreement Protocol Based on Dynamic Binary Tree

Yang Sun, Shoulin Yin, Jie Liu, and Lin Teng
(Corresponding authors: Shoulin Yin and Jie Liu)

Software College, Shenyang Normal University
Shenyang 110034, China
(Email: 352720214@qq.com; nan127@sohu.com)

(Received Mar. 24, 2018; Revised and Accepted Sept. 4, 2018; First Online June 15, 2019)

Abstract

Traditional ciphertext encryption scheme easily leaks individual data privacy information. Therefore, this paper proposes a certificateless group authenticated key agreement protocol based on dynamic binary tree. Group authentication key negotiation protocol enables multiple participants to establish a session key in an open channel. In order to provide key authentication and reduce the cost, the binary tree is introduced into the group key agreement. Due to certificateless mechanism, it simplifies the complex certificate management problem in the protocol based on certificate. And it also solves the key escrow problem based on the identity. In addition, the new protocol has made rigorously formalized proof and a comparison of calculation horizontally. The results show that the new protocol is safe and efficient.

Keywords: Certificateless; Dynamic Binary Tree; Group Authentication Key Negotiation Protocol

1 Introduction

Recently, the oriented group applications such as software video conference increase seriously with the popularity of wireless networks. In the open network communication, the most important consideration is messages safety, integrity and the certification of message source [17]. Therefore, the demand to establish a safe and effective Authenticated Group Key Agreement (AGKA) is increasing too [6, 10]. In AGKA protocol [4, 5, 16, 18, 23, 24], participants can establish a new session key for each session. In this scheme, public information is participator's public key. But the private key hosting problem has been plaguing this kind of protocol. Because it needs KGC (Key Generation Center) to generate private key, the controlled impersonator may initiate an attack on KGC [1, 3, 9, 15]. The non-certificate AGKA protocol adopts the non-certificate Public Key Cryptography. Therefore, it is not necessary to complete the PKI, and also avoids the Key trust issue, which is a more efficient ways of Key negotiation [20, 26].

Therefore, many researchers proposed amounts of new schemes to solve the above issue. Deng [7] proposed an effective PKC-based certificateless group authenticated key agreement protocol, the certificateless mechanism of the protocol simplified the complex certificate management problem and key escrow problem in ID-based protocols. The security of the scheme was proved and its computational cost was discussed. The result showed that the new protocol was secure and effective. Zhang [27] studied authenticated AGKA in certificateless and identity-based public key cryptosystems. They formalized the security model of certificateless authenticated asymmetric group key agreement and realized a one-round certificateless authenticated asymmetric group key agreement protocol to resist active attacks in the real world. They also investigated the relation between certificateless authenticated AGKA and identity-based authenticated AGKA. So a concrete conversion from certificateless authenticated AGKA was proposed to session key escrow-free identity-based authenticated AGKA. Yin [25] introduced the concept of distributed Searchable asymmetric encryption, which was useful for security and could enable search operations on encrypted data. And many other newest works by researchers [2, 11, 14].

Therefore, this paper proposes a certificateless group authenticated key agreement protocol based on dynamic binary tree. In terms of security, the protocol can prove safety in the random prediction model; For performance, the new protocol requires only one round to complete authentication and key negotiation; And for computation, compared with state-of-the-art schemes, the calculation of new protocols is also significantly reduced. The rest of the paper is organized as follows. Section 2 introduces the preliminaries used in this paper. Section 3 outlines the proposed scheme to analyze detailed processes. Ex-

periments and security analysis are given in Section 4. Finally, Section 5 concludes this paper.

2 Preliminaries

2.1 Computational Difficulties and Related Hypotheses

Definition 1. *Negligible function. For any $c > 0$, there is a b_1 satisfying $b > b_1$, and function $\varepsilon(b) \leq \frac{1}{b^c}$. Then function $\varepsilon(b0$ is negligible function.*

Definition 2. *Diffie-Hellman problem. Given three randomly numbers $P \in G_p$, aP, bP, Diffie-Hellman problem indicates that computing abP is difficulty within polynomial time ($a, b \in Z_p^*$). The advantage of solving Diffie-Hellman problem in polynomial time by adversary A can be defined as:*

$$Adv_{A,G_p}^{Diffie-Hellman} = Pr[A(P, aP, bP) = abP]$$

Meanwhile, for any polynomial time, the advantage meets $Adv_{A,G_p}^{Diffie-Hellman} < \varepsilon$.

Definition 3. *Bilinear Diffie-Hellman problem (BDH). Assuming that G_p and G_m are two groups with $p-order$. P is the generator of G_p. $e : G_p \times G_p \to G_m$ is a bilinear map. BDH problem indicates that computing $e(P, P)^{abc}$ is difficulty with given (P, aP, bP, cP). The advantage of solving BDH problem in polynomial time by adversary A can be defined as:*

$$Adv_{A,G_p,G_m}^{BDH} = Pr[A(P, aP, bP, cP) = e(P, P)^{abc}]$$

And for any polynomial time, the advantage meets $Adv_{A,G_p,G_m}^{BDH} < \varepsilon$.

Definition 4. *Bilinear map. Supposing G_0 and G_1 are two $p-order$ multiplicative cyclic groups. g is a generator of G_0 and e is a bilinear map, namely $e : G_0 \times G_0 \to G_1$, then for any $i, j, k \in G_0$ and $a, b \in Z_p$, the map e has the following properties:*

1) Bilinear: $e(i^a, j^b) = e(i, j)^{ab}$.

2) Non-degenerative: $e(g, g) \neq 1$.

3) Polymerizability: $e(i \cdot j, k) = e(i, k) \times e(j, k)$.

If the group operation is highly computable in G_0 and the map $e : G_0 \times G_0 \to G_1$, then the group is called bilinear. So map e is commutative: $e(g^a, g^b) = e(g, g)^{ab} = e(g^b, g^a)$.

Definition 5. *Round number of protocol. A communication protocol's round number refers to the interaction number between participants in a communication of the protocol and other participants in the process, such as single round protocol refers to participants need to interact with other participants that can achieve protocol, the protocol is designed as the single wheel in this paper.*

3 Security Model of Proposed Protocol

For AGKA protocol, the basic security target is to implement Authenticated Key Exchange (AKE) and Mutual Authentication (MA).They are defined as follows.

Definition 6. *AKE security. If the participants in each protocol can ensure that no other parties are able to obtain the information relating to the session key except legal participants, it is said that the protocol satisfies AKE security requirement.*

Definition 7. *MA security. If the participants of each protocol can ensure that only their partners can share the session key, it is said that the protocol meets the MA security requirement.*

Elkair [8] proposed a new and efficient key establishment protocol in the asymmetric (public key) setting that is based on MTI (Matsumoto, Takashima and Imai)-two pass key agreement protocol which consisted of three phases; The Transfer and Verification Phase, and The Key Generation Phase. This protocol was strong against most of potential attacks(Known-Key Security, Forward (Perfect) Secrecy, Key-Compromise Impersonation, Unknown Key-Share Attack, Small Subgroup Attack, and Man-in-the-Middle Attack) with low complexity (complexity is 4), which can be abbreviated as MTIT. In this protocol, if $x \in [1, p-1]$, then $\bar{x} = (xmod2^{f/2})+2^{f/2}$. In here, f is the bit of q. Generally, q is a prime number of 160 bit. And $\bar{x}(xmod2^{80}) + 2^{80}$. C_A denotes the certificate of A, which contains unique information string of A (such as the name, address), public key P_A ($P_A = \alpha^a modp, a \in [1, q-1]$), certificate center. The detailed negotiation processes are as follows.

1) A selects secret information $x \in [1, q-1]$ and sends $R_A = \alpha^x modp$, C_A to B.

2) B selects secret information $y \in [1, q-1]$ and sends $R_B = \alpha^y modp$, C_B to A.

3) A verifies R_B, whether it satisfies $1 < R_B < p$ and $(R_B)^q \equiv 1 modp$. If it fails, then A terminates the protocol. Otherwise, A calculates $S_A = (x + a + \bar{R_A}$ and sharing key $K = (R_B(P_B)^{\bar{R_B}})^{s_A}$. If $K = 1$, it stops protocol.

4) B verifies R_A, whether it satisfies $1 < R_A < p$ and $(R_A)^q \equiv 1 modp$. If it fails, then B terminates the protocol. Otherwise, B calculates $S_B = (y + b + \bar{R_B}$ and sharing key $K = (R_A(P_A)^{\bar{R_A}})^{s_B}$. If $K = 1$, it stops protocol.

5) $k = H(K)$ is the negotiation key of A and B.

The above protocol requires two rounds communication. Under the situation of A knowing public key of B, it only needs to send one message from A to B. This protocol is suit for one online. One round communication is as:

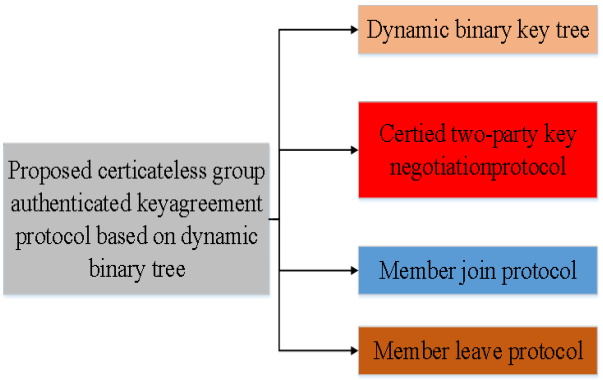

Figure 1: The composition of the proposed scheme

1) A selects secret information $x \in [1, q-1]$ and sends $R_A = \alpha^x modp$, C_A to B.

2) A computes $s_A = (x + a\bar{R_A}) modq$ and $K = (R_B(P_B)^{\bar{R_B}})^{s_A}$. If $K = 1$, A stops protocol.

3) B verifies R_A, whether it satisfies $1 < R_A < p$ and $(R_A)^q \equiv 1 modp$. If it fails, then B terminates the protocol. Otherwise, B calculates $S_B = (y + b + \bar{R_B}$ and sharing key $K = (R_A(P_A)^{\bar{R_A}})^{s_B}$. If $K = 1$, it stops protocol.

4) $k = H(K)$ is the negotiation key of A and B.

In fact, little modified in the above protocol, it can be used for the trust of A for B. B uses temporary private key and temporary public key respectively to replace long-term private key and long-term public key to verify the identity of A. So this paper gives a no authentication protocol to simplify the above processes as follows.

1) A calculates $S_A = (x + x\bar{R_A}$ and sharing key $K = (R_B(P_B)^{\bar{R_B}})^{s_A}$. If $K = 1$, A stops protocol.

2) B calculates $S_B = (y + b + \bar{R_B}$ and sharing key $K = (R_A(P_A)^{\bar{R_A}})^{s_B}$. If $K = 1$, B stops protocol.

3) $k = H(K)$ is the negotiation key of A and B.

Figure 1 shows the composition of the proposed scheme. Then we detailed introduce the process.

3.1 Key Tree

Each leaf node is associated with a group of members, the internal node is used to save the key intermediate results in the process of negotiation. In order to reduce the amount of calculation and traffic, a member is specified as a sponsor, which is responsible for the internal nodes of temporary public key and broadcasts to the members. Internal node does not correspond to the group members. There is no identity information, therefore, it cannot provide key authentication for legal group member. In order to solve the problem, group long-term public key (group key certificate) associated with internal nodes is introduced, the corresponding private key only is known for legal group members [13, 19, 21].

Temporary private key α_i of leaf node is randomly selected by group member M_i. The temporary private key of internal node is the result of two-side key negotiation that can be certified by its children nodes. The temporary private key of $j-th$ node in $i-th$ $(N_{(i,j)})$ can be denoted as $k_{(i,j)}$, the corresponding temporary public key is $b_{(i,j)}$. Children nodes of node $N_{(i,j)}$ are denoted as $N_{i+1,l}$ and $N_{i+1,l+1}$ respectively and their corresponding private key is y_{x1}, y_{x2} and y_{x3}. The $m-th$ member generate the temporary private key α_m.

Each member needs to compute all the temporary private key from its corresponding leaf nodes to root node. Temporary public key of all the brother nodes should be obtained. For example, the following is the process of calculating root key $k_{0,0}$. First, M_1 generates temporary private key $\alpha_1(k_{2,0})$, and gets a temporary public key $b_{\alpha_2}(b_{2,1})$, $b_{\alpha_3}(b_{2,1}))$ of M_2 and M_3, respectively. Long-term public key is also obtained. So M_1 can be calculated by:

$$\begin{aligned} k_{1,0} =& e(b_{\alpha_2} + H_1(b_{\alpha_2}||y_2P)y_2P, b_{\alpha_3} \\ &+ H_1(H_1(b_{\alpha_2}||y_3P)y_3P)^{\alpha_1+H_1}. \end{aligned}$$

Therefore, the group key is calculated by using temporary public key $b_{1,1}$.

$$k_{0,0} = e(b_{1,1} + H_1(b_{1,1} + H_1 b_{1,1}||y_2P)y_2P, Q)^{\alpha_2+H_2}.$$

3.2 Certified Two-Party Key Negotiation Protocol

Assuming that the both negotiation sides are A and B. In the initial stage, a certification center (CA) provides certificate for them to binding the user's identity with the long-term key (public key). Certificate of user A is as follows:

$$Cert_A = (I_A||xP||P||Q||S_{CA}(I_A||xP||P||Q)).$$

Where I_A denotes identity string of A. $||$ is the string of data items. S_{CA} is the signature of CA. $x \in Z_q^*$ is private key. P and Q are public used for pointing out the elements for temporary public key. The executing processes of protocol are as follows:

1) $A \to B : aP||Cert_A$.

2) $B \to A : bP||Cert_B$.

3) $k_A = e(bP + H_1(bP||yP)yP, Q)^{a+H_1(aP||xP)x}$.

4) $k_B = e(aP + H_1(aP||xP)xP, Q)^{b+H_1(bP||yP)y}$.

5) $k_{AB} = e(P, Q)^{a+H_1(aP||xP)x)(b+H_1(bP||yP)y)}$

Suppose that $S = aP||bP|a, b \in Z_q^*$ and $p \in G_1$, then $H_1 : S \to Z_q^*$ is a Hash function. x, xP and y, yP are

the private and public key of A and B respectively. They randomly select integer in $a, b \in Z_q^*$ as temporary private key. Then it sends the corresponding temporary public key aP, (bP) and certificate to each other. Finally, A and B can use the their long-term and temporary keys and the other long-term public key and public key to calculate the shared secret temporarily. The protocol provides key independence and implicit key authentication.

3.3 Member Join Protocol

Suppose that there are n members $M_1, M_2, \cdots, M_n$ in group. New member M_{n+1} broadcasts a join request message including the temporary public key and certificate. Sponsors M_s verifies certificate of M_{n+1}, if the verification is correct, then after update key tree, it recalculates all the changed key in key tree.

In order to reduce computing overhead, the new node should be inserted to the nearest sub-node of the root node. The process of join protocol is:

1) $M_{n+1} \rightarrow M_1, \cdots, M_n : \alpha_{n+1}P||C_{n+1}$.

2) All members update the key tree. The new node is inserted into the leftmost node with the smallest number of nodes. If the inserted point is a leaf node, then the leaf node is the initiator M_s. Otherwise, the leftmost leaf node in the subtree with the insertion point is the initiator.

3) The initiator M_s updates its temporary private key α_s, then it calculates all the changed keys, and finally broadcasts the key tree $B_{(n+1)}$ containing all the temporary public keys to the group.

$$M_s = M_1, \cdots, M_{n+1} : B_{n+1}||C_n||E_g(y_G)||y_G Q.$$

4) All members use the temporary public key of $B_{(n+1)}$ to calculate the group key. Then it decrypts the y_G, so M_{n+1} can get y_G, while other members can verify the correctness of the new group of key.

After the initiator updates the temporary private key, the key of all the previous nodes is recomputed. Then it broadcasts the corresponding temporary public key; Finally, all members can compute the new group key using the temporary public key in their temporary private key, which contains the collection of all temporary public keys.

3.4 Member Leave Protocol

Assuming the current group has n members, member $M_d (d \leq n)$ will leave the group. The M_s is the member of nearest and leftmost node of M_d parent node. Implementation process of leave protocol is as follows:

1) All members update the key tree and delete the nodes corresponding to M_d.

2) The initiator M_s generates the new temporary private key α_s and the new group long-term private key y'_G, calculates all the changed temporary keys, and then encrypts the y'_G with the new group key.

$$M_s \rightarrow M_1, M_2, \cdots, M_n - M_d : B_{n-1}||E_g(y'_G).$$

3) Each member calculates the group key separately and updates the group's long-term key.

4 Security and Protocol Performance Analysis

4.1 Security Analysis

New protocol's security is based on the BDH assumption. Under all the group members can execute protocol correctly, it provides security properties with key independence, perfect forward secrecy, implicit key authentication, and has the ability to resist attacks of middlemen.

When members join or leave group, new group key contains a randomly generated new information. This ensures that the new key and other key are independent of each other, it provides the key independent and perfect forward secrecy. The implicit key authentication can be divided into the following two types to analyze. For the passive attack, an attacker can get information which is limited to transmission message in the process of protocol. Through these information to get private information and group public key of members is impossible. So it also cannot get any group key. And active attacker can insert, remove or modify the message of protocol. Due to in the process of computing key, it needs to combine long-term key closely with temporary key, and simply modify the message. This cannot help an attacker to calculate any key information for a long time. Although this does not make legal group members eventually calculate the shared secret key, an attacker cannot get any group of keys too.

The introduction of the group long-term private key y_G makes originally middle node not corresponding to the group members and no identity information that has the authentication method for other group members other than the sponsor. Therefore, active attacker does not know the y_G, it only replaces the blind key of middle nodes, this cannot lead to other group members' calculation error.

Theorem 1. *Proposed certificateless group authenticated key agreement protocol can satisfy authenticated key exchange (AKE) security.*

Proof. Supposing that the adversary A with the non-negligible advantage $Adv_{AI}^{AKE}(k)$ in polynomial time breaks AKE security of the protocol, which means that the adversary can win the game with non-negligible probability. Then we prove that if adversary can win the game, then there is an algorithm AL which can help adversary solve the BDH problem. Namely, given $< P, aP, bP, cP >$, the adversary can obtain $e(P)^{abc}$. □

Before starting the game, AL random selects $< P, aP, bP, cP >$ and sets $P_0 = aP$ as the public key of PKG, $a \in_R R_p^*$ is unknown for adversary. AL sends A system parameters $pa = \{E_p, G_p, G_m, e, P, P_0 = aP, g, H_1, H_2, H_3, H_4\}$. At the same time AL keeps the following lists for quick response when the adversary initiates the query.

1) H_1^{list} holds array $< ID_i, P_i, Q_i, x_i, D_i >$.
2) H_2^{list} keeps array $< M_{ij}, N_{ij} >$.
3) key^{list} saves array $< ID_i, II_i^t, Q_i, x_i, P_i >$.

The above lists are initially empty and only recorded as the latest list values when the protocol is executed. Algorithm AL simulates the following queries.

1) Query of H_1. If A can query q_1 times at most and sends $H_1 < ID_i, P_i >$ to AL, then AL executes the following:
   - If $< ID_i, P_i >$ had been in H_1^{list}, then AL returns the computed Q_i.
   - If $< ID_i, P_i > = < ID_A, P_A >$, then $Q_A = bP$, the array will be updated as $< ID_A, P_A, Q_A, x_A, \perp >$. Q_A is returned.
   - If $< ID_i, P_i > =< ID_B, P_B >$, then $Q_B = cP$, the array will be updated as $< ID_B, P_B, Q_B, x_B, \perp >$. Q_B is returned.
   - Otherwise, AL random selects $r_i \in_R R_p^*$ and stores the $< ID_i, P_i, Q_i = r_i P, x_i, D_i = r_i aP >$ in H_1^{list}. Then AL returns $Q_i = H_1(ID_i || P_i)$.
2) Query of H_2. If A can query q_2 times at most and sends $H_2 < M_{ij} >$ to AL, then AL executes the following:
   - If $< M_{ij}, N_{ij} >$ had been in H_2^{list}, then AL returns the computed $H_2(M_{ij}) = N_{ij}$.
   - Otherwise, AL random selects $N_{ij} \in_R R_p^*$ and stores the new $< M_{ij}, N_{ij} >$ in H_2^{list}. Then AL returns $H_2(M_{ij}) = N_{ij}$.
3) Query of Key^{list}. If A sends query $< ID_i, II_i^t >$ to AL. AL executes the following response:
   - If $< ID_i, II_i^t >$ had been in Key^{list}, then AL returns the P_i.
   - Otherwise, AL random selects $x_i \in_R R_p^*$ and computes the $< P_i = x_i P$ and updates the Key^{list}. Then AL updates H_1^{list} as $< ID_i, P_i, Q_i, x_i, \perp >$.

Assuming that adversary executes the protocol and sends the guess value to AL when $i = A, j = B$, then AL computes $h_{AB} = H_2(x_A, P_B)$ and $g_{AB} = e(h_{AB} D_A, Q_B) = e(D_A, Q_B)^{h_{AB}} = e(aQ_A, Q_B)^{h_{AB}} = e(abP, cP)^{h_{AB}} = e(P, P)^{abch_{AB}}$. Therefore, for $< P, aP, bP, cP >$, BDH is solved: $e(P, P)^{abc} = g_{AB}^{-h_{AB}}$. This is impossible. So the adversary cannot break the protocol.

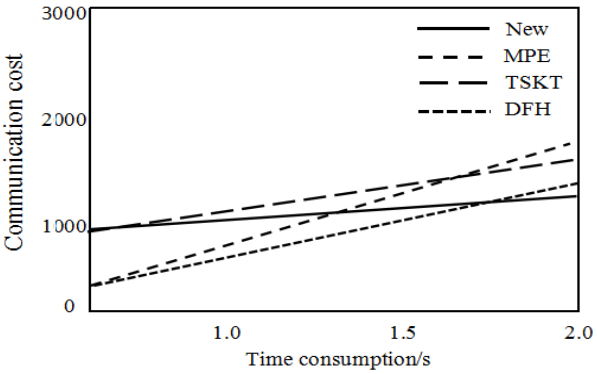

Figure 2: Comparison of tome overhead

Table 1: Functionality comparisons with different methods

Scheme	P	B	NT
DFH	YES	NO	NO
TSKT-ORAM	YES	NO	YES
MPE	YES	NO	NO
Proposed	YES	YES	YES

4.2 Communication Cost

To illustrate the effectiveness of our proposed protocol, we conduct comparison experiments at the 64-bit Intel i5-4200U processor with running speed 2.30GHz, the overhead is a constant. Join protocol requires two rounds of broadcasting, leave protocol only needs one round of broadcasting. They are all $O(log^3 n)$. Calculating one encryption process needs about 23.16ms. In addition, the certification takes about 19.84ms. Note that we omit the computational overhead of hash operation and symmetric encryption operation. So they have a significantly lower computational cost. DFH [22], TSKT-ORAM [28], MPE [12] are compared with our proposed protocol.

Figure 2 shows the results of compared schemes. From the curve, our scheme has a low computational overhead and is not affected by other factors.

4.3 Comparative Study

In this subsection, Table 1 shows the functionality comparisons between our proposed scheme and related above schemes about three aspects including Privacy protection (P), Biometrics certification (B) and No timestamp mechanism (NT). *Annotation.* YES/NO: Support/Not support.

Table 1 shows that in proposed scheme, we use dynamic binary tree as the key protection, not only can improve the security of our scheme, but also can increase the practicability of our scheme.

We also analyze the efficiency of the proposed scheme, According to the required operations for computational cost in different phases, Table 2 summarizes the computa-

Table 2: Computational costs comparisons with different methods

Scheme	DFH	TSKT-ORAM	MPE	Proposed
P_1	$2h+2s$	$3h+2s$	$3h+s$	$h+s$
P_2	$3s+2r$	$2s+4r$	$3s+2r$	$s+r$
P_3	$2s+4r$	$3r+3s$	$2r+2s$	r
Total	$7s+2h+6r$	$7s+2h+7r$	$6s+3h+4r$	$2s+h+2r$

tional costs of our proposed scheme and related schemes in all the authenticated key agreement protocol phase. *Annotation.* P_1: Certified two-party key negotiation protocol phase; P_2: Member join protocol phase; P_3: Member leave protocol phase. h: Hash operation; s: symmetric encryption; r: Round time of protocol.

5 Conclusion

In this paper, a certificateless group authenticated key agreement protocol based on dynamic binary tree is proposed. The new scheme encrypts the data through dynamic binary tree, which guarantees the security of the stored data, and associates the user key with a set of attributes. Associating the sharing key with a set of attribute discrimination criteria, the user can decrypt the ciphertext only if the attribute discrimination condition is satisfied avoiding the cost of distributing the sharing key for each user. Finally, experiments for the proposed scheme, the results show that our new scheme has very low computational and communication overhead. In the future work, we will carry out the proposed program, so as to further improve the effectiveness of privacy protection.

References

[1] T. Y. Chang, M. S. Hwang and C. C. Yang, "Password authenticated key exchange and protected password change protocols," *Symmetry*, vol. 9, no. 8, pp. 1–12, 2017.

[2] T. Y. Chang, M. S. Hwang, W. P. Yang, "A communication-efficient three-party password authenticated key exchange protocol," *Information Sciences*, vol. 181, pp. 217-226, 2011.

[3] T. Y. Chang, W. P. Yang, M. S. Hwang, "Simple authenticated key agreement and protected password change protocol", *Computers & Mathematics with Applications*, vol. 49, pp. 703–714, 2005.

[4] T. Y. Chang, C. C. Yang, M. S. Hwang, "A threshold signature scheme for group communications without a shared distribution center", *Future Generation Computer Systems*, vol. 20, no. 6, pp. 1013–1021, Aug. 2004.

[5] T. Y. Chang, C. C. Yang, M. S. Hwang, "Threshold untraceable signature for group communications", *IEE Proceedings - Communications*, vol. 151, no. 2, pp. 179–184, April 2004.

[6] S. M. Chen, C. R. Yang, and M. S. Hwang, "Using a new structure in group key management for pay-TV", *International Journal of Network Security*, vol. 19, no. 1, pp. 112–117, Jan. 2017.

[7] F. Deng, Y. Zhu, "Novel one-round certificateless group authenticated key agreement protocol," *Computer Engineering & Applications*, vol. 53, no. 5, pp. 111, 2017. (`http://cea.ceaj.org/EN/abstract/article_35411.shtml`)

[8] H. M. Elkamchouchi, E. F. A. Elkair, "An efficient protocol for authenticated key agreement," in *Radio Science Conference*, pp. 119-134, 2011.

[9] M. S. Hwang, S. Y. Hsiao, W. P. Yang, "Security on improvement of modified authenticated key agreement protocol," *Information - An International Interdisciplinary Journal*, vol. 17, no. 4, pp.1173–1178, Apr. 2014.

[10] M. S. Hwang, C. C. Lee, S. F. Tzeng, "A new proxy signature scheme for a specified group of verifiers," *Information Sciences*, vol. 227, pp. 102–115, 2013.

[11] M. S. Hwang, T. H. Sun, C. C. Lee, "Achieving dynamic data guarantee and data confidentiality of public auditing in cloud storage service," *Journal of Circuits Systems and Computers*, vol. 26, no. 5, 2017.

[12] J. Kar, "A study of key management protocols for multicast encryption," *International Journal of Innovative Computing Information & Control Ijicic*, vol. 13, no. 2, pp .559-574, 2017.

[13] K. M. Kim, K. S. Sohn, S. Y. Nam, "Key generation and management scheme for partial encryption based on hash tree chain," *Journal of the Korean Statistical Society*, vol. 25, no. 3, pp. 77-83, 2016.

[14] 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, Jan. 2013.

[15] 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.

[16] 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. 401–410, 2014.

[17] C. H. Ling, S. M. Chen, and M. S. Hwang, "Cryptanalysis of Tseng-Wu group key exchange protocol," *International Journal of Network Security*, vol. 18, no. 3, pp. 590-593, 2016.

[18] J. Liu, S. L. Yin, H.Li and L. Teng, "A density-based clustering method for K-anonymity privacy protection," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 8, no. 1, pp. 12-18, Jan. 2017.

[19] A. Souyah, K. M. Faraoun, "Fast and efficient randomized encryption scheme for digital images based on Quadtree decomposition and reversible memory cellular automata," *Nonlinear Dynamics*, vol. 84, no. 2, pp. 715-732, 2016.

[20] L. Teng, H. Li, S. Yin, "A multi-keyword search algorithm based on polynomial dunction and safety inner-product method in secure cloud environment," *International Journal of Network Security*, vol. 8, no. 2, pp. 413-422, 2017.

[21] L. I. Xin, C. G. Peng, C. C. Niu, "Attribute-based encryption scheme with hidden tree access structures," *Journal of Cryptologic Research*, vol. 3, no. 5, pp. 471-479, 2016.

[22] J. Xu, L. Wei, Y. Zhang, *et al.* "Dynamic fully homomorphic encryption-based merkle tree for lightweight streaming authenticated data structures," *Journal of Network & Computer Applications*, 2018. DOI: 10.1016/j.jnca.2018.01.014

[23] C. C. Yang, T. Y. Chang, J. W. Li, M. S. Hwang, "Simple generalized group-oriented cryptosystems using ElGamal cryptosystem", *Informatica*, vol. 14, no. 1, pp. 111–120, 2003.

[24] S. L. Yin and J. Liu, "A K-means approach for map-reduce model and social network privacy protection," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 7, no. 6, pp. 1215-1221, Nov. 2016.

[25] S. Yin, L. Teng, J. Liu, "Distributed searchable asymmetric encryption," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 4, no. 3, pp. 684-694, 2016.

[26] Q. Zhang, L. T. Yang, X. Liu, Z. Chen, and P. Li, "A tucker deep computation model for mobile multimedia feature learning," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 13, no. 3, pp. 1-39:18, 2017.

[27] L. Zhang, Q. Wu, B. Qin, *et al.* "Certificateless and identity-based authenticated asymmetric group key agreement," *International Journal of Information Security*, vol. 16, no. 5, pp. 559-576, 2017.

[28] J. Zhang, Q. Ma, W. Zhang, *et al.* "TSKT-ORAM: A two-server k-ary tree oblivious RAM without homomorphic encryption," *Future Internet*, vol. 9, no. 4, 2017.

Biography

Yang Sun obtained his master degree in Information Science and Engineering from Northeastern University. Yang Sun is a full associate professor of the Kexin software college at Shenyang Normal University. He is also a department head of network engineering. He has research interests in wireless networks, mobile computing, cloud computing, social networks and network security. Yang Sun had published more than 15 international journal and conference papers on the above research fields.

Shoulin Yin received the B.Eng. degree from Shenyang Normal University, Shenyang, Liaoning province, China in 2016. Now, he is a doctor in Harbin Institute of Technology. His research interests include Multimedia Security, Network Security, image processing and Data Mining. Email:352720214@qq.com.

Jie Liu is a full professor in Software College, Shenyang Normal University. He received his B.S. and M.S. degrees from Harbin Institute of Technology. He is also a master's supervisor. He has research interests in wireless networks, mobile computing, cloud computing,social networks, network security and quantum cryptography. Professor Liu had published more than 30 international journal papers (SCI or EI journals) on the above research fields. Email: nan127@sohu.com.

Lin Teng received the B.Eng. degree from Shenyang Normal University, Shenyang , Liaoning province, China in 2016 . Now, she is a laboratory assistant in Shenyang Normal University. Her research interests include Multimedia Security, Network Security, Filter Algorithm and Data Mining. She had published more than 10 international journal papers on the above research fields. Email:1532554069@qq.com.