Mutual-anonymity and Authentication Key Agreement Protocol

Tao Feng, Chuang Zou, Chunyan Liu and Zheng Hao
School of Computer and Communication,
School of Economics and Management, Lanzhou University of Technology, Lanzhou 730050, China

Abstract: According to the characteristics of trusted computation, we proposed an efficient pseudonym ring signature-based authentication and key agreement protocol with mutual anonymity. The use of ring signature can hide the identity information of communicating parties and effectively prevent the leakage of private information. Finally we derive a shared session key between them for their future secure communication especially in the trusted computation environment. Our protocol reaches the level of universally composable security and is more efficient.

Keywords: Anonymity, authentication and key agreement, ring signature, trusted computation, universally composable

INTRODUCTION

According to some of the more special cryptography applications, key agreement protocol also needs to protect the privacy of the communication. For instance, in trusted computing environment, protecting the privacy of the communication is one of the important functions of the trusted system. Trusted Computing Group (2009) released the TPM v 1.1 Privacy CA scheme and TPM v 1.2 Direct Anonymous testimony (DAA) scheme to realize the mutual anonymity to avoid their behavior tracking between the Trusted Platform Modules when they authenticate each other.

Anonymous digital signatures such as ring signatures (Xu et al., 2004; Bender et al., 2006), Direct Anonymous Attestation (Brickell and Li, 2010; Chen et al., 2010) and anonymous credentials (Camenisch and Lysyanskaya, 2004.) play an important role in privacy enhanced technologies. They allow an entity (e.g., a user, a computer platform, or a hardware device) to create a signature without revealing its identity. Anonymous signature also enable anonymous entity authentication.

Ring signatures, first introduced by Rivest, Shamir and Tauman, enable a user to sign a message so that a ring of possible signers (of which the user is a member) is identified, without revealing exactly which member of that ring actually generated the signature. In contrast to group signatures, ring signatures do not require any central authority or coordination among the various users (indeed, users do not even need to be aware of each other); Furthermore, ring signature schemes grant users fine-grained control over the level of anonymity associated with any particular signature.

Problems Hypothesis

The scheme of this study is mainly based on the elliptic curve cryptosystem (Enge, 2013), discrete logarithm, bilinear pairings (Su et al., 2012) and strong impact resistance of one-way hash function. The definitions and related problems hypothesis as follows: Elliptic curve E (Fp) and p is a big prime number, not less than 160 bit. Let G a multiplicative cyclic group in E (Fp) of order p, P : a generator of G.

Definition 1: Computational Diffie-Hellman Problem: Given (P, aP, bP), compute abP.
Let $C_0(S_0)$ to be the pseudonym of $C_i(S_i)$. $S$ is an adversary. Upon receiving a value $(\text{KeyGen},C_0,\text{sid})$ from $C_i$, verify that $\text{sid} = (C_0,\text{sid}')$ for $\text{sid}'$. If not, then ignore the request. Else, hand $(\text{KeyGen},C_0,\text{sid})$ to the $S$. Upon receiving $(\text{VerKey},\text{sid},C_0,\phi)$ from $S$, output them to $C_i$, record $(C_0,\phi)$. $C = \{C_0,\phi,C_0\rightarrow C_0\}$ is a identity collection of pseudonym user who has executed the key generation process, the corresponding public key is $Y = \{y_1,y_2,\ldots,y_N\}$.

Signature Generation
Upon receiving a value $(P-R-$Sign$,$sid$,$m$,$C_0,m,\sigma)'$ from $C_i$, verify that $\text{sid} = (C_0,\text{sid}')$ for $\text{sid}'$. If not, then ignore the request. Else send $(P-$Sign$,$sid$,$m$,$C_0,m,\sigma)$ to $S$. Upon receiving $(P-$Sign$,$sid$,$m$,$C_0,m,\sigma)'$ from $S$, verify that no entry $(m,\sigma,y_i,0)$ is recorded. If it is, then output an error message to $C_i$, else output $(P-$Sign$,$sid$,$C_0,m,\sigma)$ to $C_i$, and record the entry $(m,\sigma,y_i,1)$.

Signature Verification
Upon receiving a value $(P-$Sign$,$sid$,$m$,$S_0,\sigma,y_i)$ from $S_0$, hand it to $S$. Upon receiving $(P-$Sign$,$Verify$,$sid$,$m$,$S_0,\sigma,y_i,0)$ from $S$, verify that no entry $(m,\sigma,y_i,0)$ is recorded. If it is, then set $f = 0$ and record the entry $(m,\sigma,y_i,0)$.

Computable: There exists an efficient algorithm to compute $e(P,Q)$ for all $P,Q \in \mathbb{G}_1$.

**ANONYMOUS KEY AGREEMENT PROTOCOL SECURITY MODEL**

Universally Composable (UC) security framework is a formal model based on the computational complexity theory to design and analyze security protocols (Canetti and Krawczyk, 2002a; Canetti, 2005a). The most outstanding properties is that it adopts the designing thought of modularization: we can design cryptographic protocols separately, as long as each sub-protocol meets UC safety, it can guarantee the security of assembling, parallel running with other protocols (Canetti et al., 2005b).

Ring signature anonymous authentication makes the receiver certitude that the sending party is a legal member in the ring, but don't know the specific identity. Because the identity of the signer will be recorded in the session identification (sid), others can know his true identity through the sid. In order to achieve anonymity, we use pseudonym to instead of the true signer's identity information in the sid. Here, we learn the signature thought from Canetti (2004) and use pseudonym technology instead of the specific identity of the members; we first construct a ring signature ideal functionality based on pseudonym, as shown in Fig. 1:

As need to construct an anonymous authentication key agreement ideal functionality, the concept is learned from Canetti and Krawczyk (2002b) and Hofheinz et al. (2003), the constructed anonymous authentication key agreement ideal functionality $F_{A-AKE}$ is show in Fig. 2.
Fig. 3: The network model of the protocol

Table 1: The symbols and their meanings in this study

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>H_u: {0,1}^* \rightarrow \mathbb{G}</td>
<td>Collision-resistance one-way hash function, generate identity information</td>
</tr>
<tr>
<td>H: {0,1}^* \rightarrow \mathbb{Z}_p^*</td>
<td>Collision-resistance one-way hash function, generate message information</td>
</tr>
<tr>
<td>ID_s</td>
<td>Server identity</td>
</tr>
<tr>
<td>ID_c</td>
<td>User identity</td>
</tr>
<tr>
<td>S</td>
<td>Identity collection of all the servers in the network</td>
</tr>
<tr>
<td>C</td>
<td>Identity collection of a group of users</td>
</tr>
</tbody>
</table>

servers both communicate using pseudonym. The symbols and their meanings in this study are shown in Table 1.

In this study, the mutual anonymous authentication key agreement protocol based on pseudonym ring signature is divided into three phases:

- **System initialization**
- **Registration phase**
- **Key agreement**

**System initialization:** The Trusted third party is responsible for generating system parameters in the network. The specific steps are as follows:

- TTP chooses a large prime \( p \geq 2^{160} \), constructs the cyclic group \( \mathbb{G} \) and the elliptic curve \( E(\mathbb{F}_p) \) as in the second chapter.
- Bilinear map: \( e: \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T \), \( \mathbb{G} \) of order \( p \) is a multiplication cyclic group, \( g \) is a generator of \( \mathbb{G} \).
- Assume that there exist \( n \) servers \( S = \{ID_{s1},ID_{s2},...,ID_{sn}\} \) in the trusted network.
- TTP publishes the system parameters \( (p,g,g_1,g_2,u',U,H_u,H_m) \), master key \( kmk = g_2^{s} \).

**Registration phase:** Each user in the user group \( C = \{ID_{c1},ID_{c2},...,ID_{cm}\} \) needs to be registered in the servers of \( S \), every user group is assigned to a server group by the server administrator and produces a ring signature on the server group identity set. Accordingly every user can also produces a ring signature on the user group identity set.

Assume that user \( ID_{sa} \) (\( ID_{sa} \in C \)) needs to communicate with server \( ID_{sa} \) (\( ID_{sa} \in S \)), the registration process would be: The server administrator selects \( t \) servers as a server group from \( n \) servers randomly, records as \( S = \{ID_{s1},ID_{s2},...,ID_{st}\} \) and server \( ID_{sa} \) must be selected as the default server. Here, we learn the ring signature scheme thought from Yu et al. (2012), the specific signature steps are as follows:

- **ID_{sa}** randomly selects integer \( s \in \mathbb{Z}_p^{*} \), computes \( g_1 = g^s \), selects \( g \in \mathbb{G}, u^i \in \mathbb{Z}_p \), let \( U = (u_i) \) for \( t_u \) dimensional vector, \( u_i \in \mathbb{Z}_p \), publishes the system parameters \( P_{pub} = (p,g,g_1,g_2,u',U,H_u,H_m) \), master key \( kmk = g_2^{s} \).
- Suppose \( v_j = H_u(ID_{sj}) \), \( v_j[i] \) is the ith bit of \( v_j \), also satisfies that \( v_j[i] = 1 \) and the collection of subscript \( i \). Randomly selects \( r_{uj} \in \mathbb{Z}_p \), computes \( j = u + u \mod p \), so the secret key of \( ID_{sj} \) is \( d_{sj} = ((g_1^{s}(h_{j}^{(s)})^{v_j}),g_2^{v_j}) = (d_{j1},d_{j2}) \).
- **ID_{sa}** signs on the message \( M \), its secret key is \( d_{sa} = (d_{s1},d_{s2}) \), computes \( m = H_m(S_{π},M) \), the signature process: for \( j \in \{1,2,..,t\} \), selects \( r \in \mathbb{Z}_p \), computes \( R_j = g^r, h_j = H_m(S_{π},M) \); When \( j = π \), selects \( r, s \in \mathbb{Z}_p \), computes \( h_π = g^r \), \( X_s = h_π \times X_s \), the ring signature on message \( M \) is \( σ = (Z,S_{π},R_1,R_2,...,R_t) \).
- And further, the server ID_{sa} randomly selects integer \( s \), \( x_s,2 \leq x_s \leq q-1 \), computes \( X_s = g^{s} \mod p \), \( S_p = (S_{π},H_u(s)) \), \( S_p \) is the pseudonym of server \( ID_{sa} \), used for communication between with the users later, finally, the server ID_{sa} sends \( M, S_{π},S_p, s \) to the user ID_{sa} through a appropriate safe way.
Upon receiving the messages, ID_ex computes
\[ S' = S'_{\sum_{j=1}^{\ell} R_{x_j}^{\pi(\text{oh}_j)}} \]
Verifies whether \( e(S', g) = e(g_1, Z) \) and \( S_p = S'_p \) are established. If established, keeps \( X_S \) safely; Else, re-registers

User ID_ex uses the same signature methods and steps as the server ID_sr: let \( C_x = \{1|C_1, C_2, \ldots |C_n\} \) for \( t \) users identity list, the actual identity of the signer is \( ID_{ex} \in C_x, (\pi \in \{1, 2, \ldots, n\}) \) and then the ring signature on message \( M \) is \( \sigma = (Z', S', C', R_1, R_2, \ldots, R_n) \)

And further, the user ID_ex randomly selects integer \( c, x, 2 \leq x \leq q - 1, \) computes \( X_c = \sum_{c} \mod p \), \( S_p = (C_x, H_n(c)) \), \( C_p = \) the pseudonym of user ID_ex, used for communication between with the servers later, finally, the user ID_ex sends \( M, C_x, \sigma', X_c, c, C_p \) to the user ID_ex through a appropriate safe way

When server ID_ex receives the messages, computes \( S'' = C'_p = (C_x, H_n(c)), \) verifies whether \( e(S'', g) = e(g_1, Z') \) and \( C_p = C'_p \) are established. If established, keeps \( X_S \) safely; else, re-registers.

**Key agreement:** After registration, it is assumed that user ID_ex and server ID_sr need to conduct the anonymous key agreement, procedure is as follows:

Server ID_ex selects two random integer \( r_S, a, 2 \leq r_S, a \leq q - 1 \) and then computes \( M_a = X_a \mod p = g^{r_S} \mod p, N_S = g^a \mod p, S_S = x_S r_S + x_S a \mod q \) and next sends \( (M_a, N_S, S_S) \) to user ID_ex

After user ID_ex received \( (M_a, N_S, S_S) \), Computes \( n \cdot S'_S = g^{S_S} \mod p = \sum_{x_S} \mod p \), Computes \( X_S = S'_S \cdot M_a^{x_S} \mod p \) and compares it with \( X_S \), if equal, user ID_ex selects two random integer \( r_c, b \leq 2c < b \leq q - 1 \) and then computes \( M_c = X_c^{r_c} \mod p = g^{x_c + r_c} \mod p \)

Computes \( X'_S = S'_S X_c \cdot N_c \mod p \) and compares it with \( X_S \), if not equal, terminates the agreement, else, calculates the session key: \( k_e = M_c^{r_c} \mod p = g^{x_c r_c} \mod p \)

Finally, the user and the server consult out a consistent session key \( k_S = H_m(k_e, M_S, M_c, C_t, S_S, S_C) \).

**SECURITY ANALYSIS OF THE PROTOCOL**

There are several tests to prove the security of the protocol:

- Any input from the environment machine \( Z \) will be transmitted to \( A \), any output of \( A \) is copied to \( Z \)'s output (to be read by \( Z \))
- Whenever \( S \) receives a message (KeyGen, \( \text{sid}, C, \text{ps} \) from FP_P-SIG, it does: if \( \text{sid} \) is not of the form \( (C_p, \text{sid}) \) then ignores this request. Otherwise, \( S \) selects \( y_i \) and records it, returns (Verification key, \( \text{sid}, C_p, y_i \)) to FP_P-SIG.
- Whenever \( S \) receives a message (P-R-Sign, \( \text{sid}, C_p, C', m \)) from FP_P-SIG, if \( \text{sid} = (C_s, \text{sid}) \) and there is a recorded signing key \( y_i \), then \( S \) computes \( \sigma = sig(g, m) \), and hands (P-R-Sign, \( \text{sid}, C_p, m, C', \sigma \)) back to FP_P-SIG. Otherwise, it does nothing.
- Whenever \( S \) receives (P-R-Sign-Verify, \( \text{sid}, C_p, m, C', y_i \)) from FP_P-SIG, it returns (P-R-Sign-Verified, \( \text{sid}, C_p, m, C', y_i \))
- When \( A \) corrupts some party \( C_i \), \( S \) corrupts \( C_i \) in the ideal process. If \( C_i \) is the signer, then \( S \) reveals the signing key \( y_i \) as the internal state of \( C_i \)

**Fig. 4: Simulator \( S \)**

- Prove that our pseudonym-based ring signature protocol \( \rho_{rs} \) safely realizes the ideal functionality FP_P-SIG.
- Prove that our anonymous authentication key agreement protocol \( \pi' \) safely realizes the ideal functionality FA_AKE in the FP_P-SIG-hybrid model
- Use universally composable theorem, put \( \rho_{rs} \) and \( \pi' \) together and prove that the combined protocol is equivalent of protocol \( \pi \): pseudonym ring signature-based authentication and key agreement protocol with mutual anonymity and safely realizes FP_P-SIG and FA_AKE in the real life model

**Lemma 1:** If CDH assumption is established, the corresponding ring signature protocol \( \rho_{rs} \) UC realizes the ideal ring signature functionality FP_P-SIG.

**Proof:** Assume that ring signature protocol \( \rho_{rs} \) can’t UC realizes the ideal ring signature functionality FP_P-SIG, this is done by constructing an environment \( Z \) and a real-life adversary \( A \) such that for any ideal-process adversary \( S, Z \) can tell whether it is interacting with \( A \) and \( \rho_{rs} \) or with \( S \) in the ideal process for FP_P-SIG. The simulation process is shown in Fig. 4:

If an attacker can forge a ring signature, for the given input (P-R-Sign-Verify, \( \text{sid}, m, C', y_i \)), according to the output record under the execution of \( S \), \( Z \) can tell whether it is interacting with \( A \) and \( \rho_{rs} \) or with \( S \) in the ideal process for FP_P-SIG. The simulation process is shown in Fig. 4:

If an attacker can forge a ring signature, for the given input (P-R-Sign-Verify, \( \text{sid}, m, C', \sigma, Y' \)), according to the output record under the execution of \( S \), \( Z \) can tell whether it is interacting with real-life protocol \( \rho_{rs} \) or the ideal protocol FP_P-SIG. Therefore, the probability of forging successfully is negligible, contradicting with the assumption. So the lemma 1 is proved.

**Lemma 2:** If DDH assumption is established, Then protocol \( \pi' \) securely realizes the FA_AKE in the hybrid model

**Proof:** Construct an attacker \( S \) (Fig. 5) in the ideal environment first, make any of the environment machine \( Z \) can’t tell whether it is interacting with attacker \( H \) and \( \pi' \) in the FP_P-SIG-hybrid model, or with \( S \)
S runs H, H is an attack in the hybrid model, the rules are as follows:

- Any input from Z will be passed to H, and all the outputs of H will be seen as the outputs of S, Z can read their outputs.
- When S receives (sid, Cpi, Spj, role) from FPAKE, it indicates that Cj has launched the authentication key agreement, so let S simulate output $\pi'$ that interacts with H in the $FP_{R,SIG}$ and $FP_{R,SIG-hybrid}$ model. And given the same input, S lets Cj and H interact with Z according to the execution rules of $\pi'$.
- In order to simulate the implementation of $\pi'$, $FP_{R,SIG}$ can be activated by S to get the corresponding signature value $\sigma$, S can also compute $k = \text{prf}(\tau, \cdot)$, $\tau$ is the output key of Cj and S in $FP_{R,SIG}$.
- When Cj produces a local output, and S is not corrupted, S will sends the output of FPAKE to Cj. If S has been corrupted, the key value is decided by S, and S uses the previous output of Cj to determine the local output of simulated Cj and S. When H executes the operation of capturing Cj, S also captures Cj. If FPAKE has sent a key to Cj, S will get the key; If both of the Cj and S do not produce a local output, S sends its internal state to H, as well as their secret selected value; If either Cj or S has produced a local output, their temporary private keys will be wiped out, S directly passes the local key to H.

**Theorem 1:** In the real-life model, the protocol $\pi$ securely realizes ideal functionality $F_{\text{FA-AKE}}$ and for any environment machine $Z'$, equation $\text{REAL}_{Z',\pi,A,Z} \approx \text{IDEAL}_{F_{\text{FA-AKE}},S,Z}$ is established, so the mutual-anonymity and authentication key agreement protocol is safety under UC model.

**EFFICIENCY ANALYSIS**

The system initialization and registration process of the new protocol can be obtained by pretreatment, we compare our protocol with Chow and Choo (2007) and Wei et al. (2011) and only consider the operations that the calculation cost is relatively large including modular exponentiation, point multiplication, inverse, bilinear pairings and modular multiplication operations. Let $T_{\text{exp}}, T_{\text{emul}}, T_{\text{ebp}}, T_{\text{mul}}$ respectively denote the cost of modular exponentiation, point multiplication, bilinear pairings and modular multiplication operations and m is the size of the group.

From Koblitz et al. (2000) and Chen et al. (2007), we can deduce: $T_{\text{exp}} \approx 240 T_{\text{mul}}, T_{\text{emul}} \approx 29 T_{\text{mul}}, T_{\text{inv}} \approx [0.3843nq + 1.47] T_{\text{mul}}, T_{\text{ebp}} \approx 7 T_{\text{exp}}$. Table 2 shows that our protocol has a higher execution efficiency.
the scheme satisfies unconditional anonymity between
the communicating parties and protects the privacy of
communications, achieves universally composable
security. The system initialization and registration
process of the new program can be obtained by
pretreatment and it has a high efficiency, can better
meet the scenarios of the trusted computing
environment.

ACKNOWLEDGMENT

This study is supported by The National Natural
Science Foundation of China (60972078) and
Universities Basic Scientific Research Operation Cost
of Gansu Province (0914ZTB186) and The Ph.D.
Programs Foundation of Lanzhou University of
Technology (BS14200901).

REFERENCES

Bender, A., J. Katz and R. Morselli, 2006. Ring
signatures: Stronger definitions and constructions
without random oracles. In: Halevi, S., and T.
Rabin (Eds.), TCC 2006. Springer, Heidelberg,
LNCS, 3876: 60-79.

Brickell, E. and J.T. Li, 2010. A pairing-based DAA
scheme further reducing TPM resources.
Proceeding of the 3rd International Conference
Trust and Trustworthy Computing. Springer, 6101:
181-195.

Camenisch, J. and A. Lysyanskaya, 2004. Signature
schemes and anonymous credentials from bilinear

Canetti, R. and H. Krawczyk, 2002a. Universally
composable notions of key exchange and secure

of IKE’s signature-based key-exchange protocol.

Canetti, R., 2004. Universally composable signature,
certification and authentication. Proceeding of 17th
IEEE Computer Security Foundations Workshop,
iacr.org/2003/239.

Canetti, R., 2005a. Universally composable security: A
new paradigm for cryptographic protocols.
Proceedings of the 42nd Symposium on
Foundations of Computer Science (FOCS), pp:
136-145. Full Version. Retrieved from:

composable password based key exchange. Adv.
Cryptol., 3494: 404-421.

design and implementation of an efficient DAA
scheme. Proceeding of the 9th Smart Card
Research Conference Advanced Application IFIP.
Springer, Heidelberg.

Chen, L., Z. Cheng and N.P. Smart, 2007. Identity-
based key agreement protocols from pairings. Int.

identity-based key agreement and anonymous

Handbook of Finite Fields.

Hofheinz, D., J. Muller-Quade and R. Steinwandt,
2003. Initiator-Resilient Universally Composable
Key Exchange. Proceeding of European Symposium on Research in Computer Security
(ESORICS, 2003).

Koblitz, N., A. Menezes and S. Vanstone, 2000. The
state of elliptic curve cryptography. Designs, Codes

for efficient parallel computation of Tate pairing.

trustedcomputinggroup.org/home.

Walker, J. and J. Li, 2010. Key exchange with
anonymous authentication using DAA-SIGMA
protocol. Proceedings of the 2nd International

gateway-oriented password-based authenticated
key exchange based on RSA. EURASIP J. Wirel.
Comm., pp: 1-12, Doi: 10.1186/1687-1499-2011-
162.

Xu, J., Z. Zhang and D. Feng, 2004. A ring signature
scheme using bilinear pairings. Proceedings of the
5th International Conference on Information
Security Applications. Springer-Verlag Berlin,
3325: 160-169.

Yu, T., Z. M. Zhao and X.F. Ren, 2012. Efficient
identity-based ring signature in standard model. J.