Efficient Fair Content Exchange with Robust Watermark Ownership

Wen-Shenq Juang*, Chun-I Fan† and Ming-Te Chen‡
*Department of Information Management
National Kaohsiung First University of Science and Technology
Kaohsiung 804, Taiwan
Email: wsjuang@ccms.nkfust.edu.tw
†Department of Computer Science and Engineering
National Sun Yat-sen University
70 Lien-Hai Road, Kaohsiung 804, Taiwan
Email: cifan@faculty.nsysu.edu.tw
‡ Email: ecsemtchen@gmail.com

***Abstract*—In recently years, the Internet is the major media to convey or delivery digital contents. Users can purchase or sell the digital contents on the Internet. Also some approaches can use the digital watermark to claim that some digital contents are owned by proofing the corresponding digital watermark to other users. In recently years, a lot of buyer-seller watermarking protocols were proposed. However, when users may want to exchange their digital contents by using these proposed protocols, none of them cannot cope with the watermark exchange problem. By the way, due to the security problem of network, users may not trust each other, so the mutual authentication between users must be ensured before exchanging digital contents. In addition, how to exchange the digital contents fairly on the Internet is another problem since users may not be honest. Not only our proposed scheme can exchange their watermarked digital contents securely but also keep their exchange process fairly to each other.**

***Index Terms*—digital watermark, digital content exchange, fair exchange, mutual authentication.**

I. Introduction

Data hiding is an interesting research topic in recently years. It can be defined as the process of embedding the digital content like audio or image in the cover image or host image imperceptibly. In recently years, digital watermarking is gradually growing up by the e-commerce on the Internet. It can be used for claiming the ownership on the digital content and protecting the integrity of the digital content.

In one hand, there are many papers focused on the reversible watermark [6], [10], [12], [18], [23] that watermark recovery information is embedded into the image which does not affect its resolution. It also can be used to detect if the image transmitted is modified or not. The reversible watermark is recently getting interested by researchers. The reason is that it can recover the original image without an additional help or information about the original image and detect the tampered watermarking region.

On the other hand, users may want to purchase the digital contents from the shop on the Internet and also embed the watermark into this sold digital content performed by the shop or the Watermark Certificate Authority (WCA for short). There were some papers focused on the buyer-seller watermarking technique [9], [17], [19], [25], [28]. With the help of above schemes, the user can claim her/his ownership on this digital content with the embedded watermark.

However, if one user wants to exchange their own digital content with another user from the Internet, they may have to extract their original watermark first and then embed their own watermark into the un-watermarked content received from each other. Usually, if their original watermarks are robust, then it can not be removed except the watermark embedder, or they have to perform the buyer-seller protocol again by choosing a new watermark.

Due to this reason, we propose our watermark exchange scheme for digital content exchange and keep their original watermark after the exchange protocol. Hence, we can efficiently perform the watermark exchange and also have the fair digital content delivery to each other.

II. Related Works

Digital watermark is getting gradually attentive in recently years. They can be applied to the buyer-seller watermarking protocol to claim the ownership of the owner. Up to date, many buyer-seller watermarking schemes were proposed [9], [17], [19], [25], [28].

In these buyer-seller watermarking schemes [9], [17], [19], [25], [28], we can discover that some of them can cope with the customer's copyright problem, the binding problem, the dispute problem and the buyer's anonymity, the conspiracy problem, the copy detection problem, etc. On the Internet, if some users want to exchange their digital contents with another users, how does she/he transfer the ownership to another user in their digital content ? We define this situation as the watermark exchange problem i.e., the ownership transfer problem. In such situation, one user can not exchange his/her own digital content to another users and they all may have to perform the current buyer-seller watermarking protocols again. Hence, this not only causes the inefficiency of these users but also there may have cheating problems happened among these users.

In the following, we review the related buyer-seller watermarking protocols. In [25], the proposed method does not provide the security analysis and can not solve the conspiracy problem. In [19], the proposed method can solve the above problems but if the buyer generates the empty string as the embedded watermark, it may cause that the buyer gets the final digital content and distributes to the another party. If the judge then executes the copyright violator identification and arbitration protocol, she/he discovers that the $\sigma(W) \neq \sigma(\varepsilon)$ where ε is the empty string selected by the buyer, and then she/he cannot accuse the buyer is guilty. By the way, their scheme also does not provide the watermark exchange property. In [9], the proposed method does not provide the buyer's anonymity property and also does not offer the watermark exchange property. In [28], the proposed method does not provide the buyer's anonymity during the transaction with the seller. The seller can know the buyer's public key and the buyer's certificate of this public key. By the way, this scheme does not support the watermark exchange function. In [17], the proposed scheme can solve the above problems but does not offer the watermark exchange function.

In these schemes [9], [17], [19], [25], [28], none of them can cope with the watermark exchange problem and some of them still have some problems as mentioned above.

III. The Proposed Scheme

In order to cope with the watermark exchange problem, we propose our method to solve it and have other additional nice properties. Not only our scheme is efficient but also it can ensure fair transaction between users on the Internet.

Our proposed scheme contains following nice properties including watermark exchange, authentication and key agreement, without WCA, low computation cost, and optional usage for robust watermark or reversible watermark. In one hand, the user can send her/his desired exchange digital content to the Trusted-Server(TS for short) to be used for fair exchange. On the other hand, the TS can provide the mutual authentication with both users and also generates the session key for digital content exchanging usage. In our proposed scheme, we have four phases including the setup phase, the authentication and key agreement phase, the watermark exchange phase, and the recovery phase. Notations are as follows.

- p: a prime number
- E: an elliptic curve defined over F_p
- q: the number of points on E
- G: a point on E having prime order q
- x_i: a private key with $0 \leq x_i \leq q-1$, where $i \in \{$Alice, Bob, Trusted-Server(TS)$\}$
- $X_{A,TS}$: a temporary symmetric shared key between Alice and TS
- $Y_{B,TS}$: a temporary symmetric shared key between Bob and TS
- m_i: the original digital content without any watermark embedded inside, where $i \in \{$Alice, Bob$\}$
- M_i: the watermarked object after watermark embedding operation by performing reversible

watermark/robust watermark method, where $i \in \{$Alice, Bob$\}$

- P_i: the public key $P_i = xG$ on E of each participant in the protocol, where $i \in \{$Alice, Bob, Trusted-Server(TS)$\}$
- W_i: the watermark for exchanging and embedding operation, where $i \in \{$Alice, Bob$\}$
- sk_{ij}: the session key which is used in each transaction, where $i, j \in \{$Alice, Bob$\}$
- $F_i(\cdot)$: the watermark embedding function with the reversible or robust watermark method, where $i \in \{$Alice, Bob$\}$
- $h(\cdot)$: a secure one-way hash function
- Arg_i: the digital content exchange agreement information agreed on the both party Alice and Bob, where $i \in \{$Alice, Bob$\}$
- $E_{X_{i,j}}(\cdot)$: the symmetric encrypting function with the temporary symmetric shared key $X_{i,j}$, where $i, j \in \{$Alice, Bob, Trusted-Server(TS)$\}$
- $D_{X_{i,j}}(\cdot)$: the symmetric decrypting function with the temporary symmetric shared key $X_{i,j}$, where $i, j \in \{$Alice, Bob, Trusted-Server(TS)$\}$
- s_{m_i}: the watermark secret value for extracting watermark based on the corresponding watermark technique(reversible or robust watermark) adopted by $i \in \{$Alice, Bob$\}$

A. The setup phase

In this phase, Alice(A for short) and Bob(B for short) use the generic ECC cryptosystem operations [7] to generate the parameters and both prepare exchange tuples including the exchanged objects and the watermark exchange agreements Arg_A and Arg_B,...etc as mentioned above.

In the beginning, in A's part, A selects $x_A \in (1 \leq x_A \leq q - 1)$ as the secret key and also computes the public key $P_A = x_A G$.

Alice:
She prepares x_A and P_A as the secret key and the public key, respectively. Alice generates necessary parameters as the following steps.

1) Compute $P_A = x_A G$ mod q.
2) Alice generates the watermark object α, where $\alpha = \{M_A || Arg_A || F_A(\cdot) || s_{m_A}\}$.
3) Then Alice computes the shared key $X_{A,TS} = h(P_A || P_{TS} || x_A P_{TS})$ using the secret key x_A. Finally, let $e = (ID_A, \alpha, N_A, h(ID_A || \alpha || N_A))$ be the exchanging information of A.
4) Alice generates $E_{X_{A,TS}}(ID_A, \alpha, N_A, h(ID_A || \alpha || N_A))$ and forwards it with ID_A and N_A to Bob.

Bob:

1) After receiving $E_{X_{A,TS}}(ID_A, \alpha, N_A, h(ID_A || \alpha || N_A))$, ID_A and N_A from Alice, Bob computes $\varepsilon = (M_B || Arg_B || F_B(\cdot) || s_{m_B})$ and $Y_{B,TS} = h(P_B || P_{TS} || x_B P_{TS})$ with the secret key x_B.
2) Then Bob also generates $\phi = (ID_B, \varepsilon, N_B, h(ID_B || \varepsilon || N_B))$ and $E_{Y_1}(\phi)$ using ID_B and the nonce variable N_B.

Then, Bob forwards $(ID_A, N_A, E_{X_1}(e))$ and $(ID_B, N_B, E_{Y_1}(\phi))$ to the Trusted Server. After TS receives this information, TS will decrypt the corresponding cipher text by the following steps.

1) First, TS derives the temporary sharing key $X_{A,TS} = h(P_A || P_{TS} || x_{TS} P_A)$ and $Y_{B,TS} = h(P_B || P_{TS} || x_{TS} P_B)$ using her/his secret key x_{TS}.
2) Compute $D_{X_{A,TS}}(E_{X_{A,TS}}(ID_A, \alpha, N_A, h(ID_A || \alpha || N_A))) = e = (ID_A, \alpha, N_A, h(ID_A || \alpha || N_A))$.
3) Compute $D_{Y_{B,TS}}(E_{Y_{B,TS}}((ID_B, \varepsilon, N_B, h(ID_B || \varepsilon || N_B)))) = \phi = (ID_B, \varepsilon, N_B, h(ID_B || \varepsilon || N_B))$.
4) The TS will check e and ϕ with $h(ID_A || \alpha || N_A)$ and $h(ID_B || \varepsilon || N_B)$. After checking e and ϕ, if they are valid, then TS computes the session key sk_{AB}, where is the $sk_{AB} = h(N_A N_B) \oplus h(x_{TS} rP)$, where $rP \in_R G$.
5) Then, TS also computes the signature Z and Z' on A's and B's exchange watermarked objects, respectively.
   - $C_A = \alpha G$, $C_B = \varepsilon G$
   - $s = h(N_A + 1 || sk_{AB} || C_A)$, $t = h(N_B + 1 || sk_{AB} || C_B)$
   - $Z = s x_{TS} + \alpha$, $Z' = t x_{TS} + \varepsilon$
6) TS forwards these cipher-texts δ and ξ where $\delta = E_{X_{A,TS}}(N_B, N_A + 1, sk_{AB}, s, Z)$, $\xi = E_{Y_{B,TS}}(N_A, N_B + 1, sk_{AB}, t, Z')$ and the session key sk_{AB} to Bob.

Bob:

1) After Bob receives δ and ξ from TS, Bob decrypts $D_{Y_{B,TS}}(\xi)$ and checks N_B+1, sk_{AB}, C_B with $h(N_B+1||sk_{AB}||C_B)$. If the result is valid, then Bob computes $E_{sk_{AB}}(N_A+1)$ and forwards δ to Alice with $E_{sk_{AB}}(N_A+1)$.

Alice:

1) Alice decrypts $D_{X_{A,TS}}(\delta)$ and $D_{sk_{AB}}(E_{sk_{AB}}(N_A+1))$, respectively. Then Alice can check N_A+1 and δ, and she/he can also get and verify the session key sk_{AB} from δ.
2) After she/he checks the nonce N_A+1, she/he can compute $E_{sk_{AB}}(N_B+1)$ with sk_{AB} as the response to Bob. Therefore, She ends up this authentication and key agreement protocol with Bob and TS.

B. The watermark exchange phase

After the authentication with TS, Alice starts to perform the watermark exchange phase with Bob.

Alice:

1) She prepares the exchange digital content $\alpha = (M_A||ID_A||Arg_A||F_A(\cdot)||s_{m_A})$ and $\psi = H(\alpha)$. Then Alice computes the hash value on $(P_A, P_B, Arg_A, Arg_B, Z)$, where $c = h_1(P_A, P_B, Arg_A, Arg_B, Z)$.
2) Alice performs the signing operation and generates the signature $S = r_1 + x_Ac$, where $R_1 = r_1G$ and $r_1 \in_R Z_q$.
3) Let $U = \{S, Z, c, s, R_1\}$ be the exchanged information with the partial signature on α. Finally, she also forwards the final result $E_{sk_{AB}}(U, C_A)$ to Bob.

Bob:

1) He decrypts $E_{sk_{AB}}(U, C_A)$ with the session key sk_{AB}.
2) Then he can check U and C_A. If U is valid, he prepares $(\varepsilon, \phi, t, Z', C_B, \phi)$.
3) Bob finally forwards his encrpyted watermarked object $E_{sk_{AB}}(\varepsilon, \phi, t, Z', C_B, \phi)$ to Alice using the session key sk_{AB}.

Alice:

1) After Alice receives these tuples from Bob, she/he can decrypts $E_{sk_{AB}}(\varepsilon, \phi, t, Z', C_B, \phi)$ and checks whether they are valid or not.
2) If they are valid, Alice can perform her/his watermark embedding operation. She will extract the Bob's watermark W_A and embed her watermark W_A into the object M_B by using $F_B(M_B, W_A, s_{m_B}) = M'_A$.
3) After generating her own watermarked object, she prepares her watermarked object to Bob. She also prepares (α, e, Z, s, C_A) and forwards $E_{sk_{AB}}(\alpha, e, Z, s, C_A)$ to Bob.
4) If Bob receives these tuple from Alice, he will perform the watermarking operation on α as the same as Alice. If he does not receive these tuples from Alice, then Bob can carry out the recovery phase to ask TS for performing the dispute resolution on this exchanging transaction between Alice and him.

C. The recovery phase

In this phase, if Alice does not forward her exchange object to the user Bob after Bob has sent his watermarked object to her, then Bob can ask TS to perform the recovery phase. First, Bob sends the exchange object $E_{Y_{B,TS}}(U, C_A, \varepsilon, \phi)$ including the partial signature generated by Alice and Bob's watermarked object to TS. When TS receives these tuples from Bob, he performs the recover phase. If the partial signature (U, C_A) and (ε, ϕ) are both valid, then he extracts the Alice's watermark object from partial signature Z.

$$\alpha = Z - sx_{TS}$$

After extracting, he forwards the encrypted watermark objects $E_{Y_{B,TS}}(\alpha)$ to Bob. When receiving the tuple from TS, he can decrypt the cipher-text and get the desired watermark object.

IV. Security Analysis

In our proposed scheme, our method can solve the watermark exchange problem as mentioned above in Section 2. We describe the security analysis of the proposed scheme as followings.

- **Authentication**: In the authentication phase, we can find out that Alice and Bob both run the challenge-response authentication with TS. TS will authenticate the both sides and also assigns the session key to Alice and Bob. In this phase, we can know that Alice and Bob send their challenge N_A and N_B to TS. And

TS also responses the $N_A + 1$ and $N_B + 1$ back to Alice and Bob. If the attacker wants to reply the old nonce N_A or N_B to TS, then she/he must have to decrypt $E_{X_{A,TS}}(e)$ and $E_{Y_{B,TS}}(\phi)$ without having $X_{A,TS}$ or $Y_{B,TS}$ with non-negligible advantages in the polynomial time bound. It will cause the contradiction in our assumption in the appendix. On the other hand, after authentication with both parties, TS will also generate the $N_A + 1$ and $N_B + 1$ encrypted by session key sk_{AB} as a response to Alice and Bob, respectively. After receiving these encrypted nonce variables, Alice and Bob can verify them and also finish the authentication phase. If the attacker wants to reply the nonce response variables, then she/he must have to decrypt the encrypted nonce variables by guessing the session key with non-negligible advantages in the polynomial time bound. It will also cause the contradiction in our assumption in the appendix. In the appendix, we have the formal proof of this property.

- **Key-agreement**: In the authentication and key agreement phase, we can find that TS will prepare the session key sk_{AB} for usage in the watermark exchange phase. However, the attacker cannot compute sk_{AB} without having $X_{A,TS}$ or $Y_{B,TS}$, where they were computed by Alice and Bob's secret keys, respectively. Hence, we claim that the authentication and key agreement phase is secure unless the attacker can correctly guess the session key sk_{AB} with non-negligible probability in the polynomial time bound.
- **Fair-exchange**: After the authentication and key-agreement phase, Alice and Bob have sent their exchange digital contents and secret information about this digital content to TS for fair exchange. In this phase, we can know that if there is a dispute (like Alice or Bob do not send their digital content to the other party), Alice or Bob can ask TS to solve the dispute in this situation. There are many papers [1]-[24] proposed to deal with these fair transaction problems efficiently. In the appendix, we also have the formal proof on this property.
- **Watermark-exchange**: In this phase, Alice and Bob use the session key sk_{AB} to encrypt their own digital contents. We assume that the watermark s_{m_A} and s_{m_B} are generated by the secure watermark generating function and the watermarking embedding function is also a secure function as mentioned in [22]. By the way, the attacker cannot embed his/her watermark into the watermarked objects or extract the watermark from them, respectively.

V. Performance and Functionality Comparisons

We assume that p is 1024 bits and q is 160 bits for security consideration [20]. Assume the H is the computation time of one hashing operation, E is the computation time of one modular exponential operation in a 1024 modulo, M is the computation time of one modular multiplication in a 1024-bit modulo and EC_M is the computation time of the multiplication of a number over an elliptic curve [2], [13], [21]. By the way, we assume that schemes [9], [17], [19], [25], [28] whose encryption operation is about 1 RSA encryption operation and let Sig, $SymEnc$, and $SymDec$ to be the signature operation, symmetric encryption and symmetric decryption, respectively. Assume that an elliptic curve over a 163-bit field has the same security level of 1024-bit public key cryptosystems such as the RSA or the Diffie-Hellman cryptosystem[13]. Assume that $E \cong 8.24EC_M$ for the implementation with the StrongARM processor in 200MHz as referenced in[13]. We also can find the relationship $E \cong 240M, E \cong 600H, Sig \cong SymEnc, SymDec \cong SymEnc$ [3], [14], [29].

In [25], the we find that their scheme does not have the security analysis. The computation cost about the watermarking protocol is about $2180M + 1W$. Also their scheme does not provide the watermark exchange property. In [19], the proposed scheme do not provide the buyer's watermark verification function for being used by Judge. So the Judge may not be able to accuse the suspect buyer that is guilty. On the other hand, the computation cost of the watermarking protocol is about $1680M + 1W$. Also it don't provide the watermark exchange property. In [9], their scheme also does not provide the buyer's anonymity protection and watermark exchange property. Their computation cost of the watermarking protocol is about $2162M + 1W$. The computation cost is higher than that of our proposed scheme. In [28], the buyer's anonymity was not solved and computation cost is about $(3n+3) \times 240M + 2W$, where n is the number of watermarks. In [17], the proposed scheme can not provide the watermark exchange property. Table 1 and Table 2 are the functionality and performance comparisons tables.

VI. Conclusions

In our scheme, we use the lightweight authentication method combining with the watermark exchange, to provide fair digital content exchange and also to solve the watermark exchange problem. Not only users can authenticate the other party on the Internet, but also they can exchange their digital content fairly. In summary, our proposed scheme can offer the solution for the watermark exchange problem and also provide multiple properties.

References

[1] A. Alaraj and M. Munro, "An Efficient e-Commerce Fair Exchange Protocol that Encourages Customer and Merchant to Be Honest," Proceeding of Computer Safety, Reliability, and Security, LNCS 5219, pp.193-206, 2008.

[2] A. Jurisic and A.J. Menezes, "Elliptic Curves and Cryptography," pp.1-13, 1997.

[3] B. Schneier, "Applied Cryptography," 2nd edition, John Wiley & Sons Inc., 1996.

[4] D. Pointcheval and J. Stern, "Security Arguments for Digital Signatures and Blind Signatures," Journal of Cryptology, Vol. 13, pp.361-396, 2000.

[5] G. Arora, M. Hanneghan, and M. Merabti, "P2P Commercial Digital Content Exchange," Electronic Commerce Research and Applications, Vol. 4, pp.250-263, 2005.

[6] H. L. Jin, M. Fujiyoshi, and H. Kiya, "Lossless Data Hiding in the Spatial Domain for High Quality Images," IEICE Trans. Fundamentals, Vol. E90-A, No. 4, pp.771-777, 2007.

[7] IEEE P1363, "Standard Specifications for Public-key Cryptography," Draft version D22, November 2, 2005.

[8] I. K. Jeong, O. Kwan and D. H. Lee, "A Diffie-Hellman Key Exchange Protocol without Random Oracles," Proceeding of CANS 2006, LNCS 4301, pp.37-54, 2006.

[9] I. M. Ibrahim and S. H. N. El-Din, "An Effective and Secure Buyer-seller Watermarking Protocol," Proceeding of IAS, pp.21-28, 2007.

[10] J. Fridrich, M. Goljan, and R. Du, "Lossless Data Embedding-new Paradigm in Digital Watermarking," EURASIP J. Applied Signal Process., Vol. 2002, No. 2, pp.185-196, 2002.

[11] J. Liu, R. Sun, W. Ma, Y. Li and X. Wang, "Fair Exchange Signature Schemes," Proceedings of Advanced Information Networking and Applications -Workshops, 2008., pp.422-427, 2008.

[12] J. Tian, "Reversible Data Embedding Using a Difference Expansion," IEEE Trans. Circuits Syst. Video Techno., Vol. 13, No. 8, pp.890-896, 2003.

[13] K. Lauter, "The Advantages of Elliptic Curve Cryptography for Wireless Security," IEEE Wireless Communications, Vol. 11, No. 1, pp.62-67, 2004.

[14] K. Takashima, "Scaling Security of Elliptic Curves with Fast Pairing Using Efficient Endomorphisms," IEICE Trans. on Fundamentals, Vol. E90-A, No.1, pp.152-159, 2007.

[15] M. Abdalla, M. Bellare, and P. Rogaway, "The Oracle Diffie-Hellman Assumption and an Analysis of DHIES," CT-RSA01, pp143-158, 2001.

[16] M. A. Strangio, "Effiecient Diffie-Hellman Two-Party Key Agreement Protocols based on Ellptic Curves," Proceedings of the 2005 ACM Symposium on Applied Computing, pp.324-331, 2005.

[17] M. H. Shao, "A Privacy-Preserving Buyer-seller Watermarking Protocol with Semi-trust Third Party," Proceeding of TrusBus, LNCS 4657, pp.44-53, 2007.

[18] M. U. Celik, G. Sharma, A. M. Teklp, and E. Saber, "Lossless Generalized-LSB Data Embedding," IEEE Trans. Image Process., Vol. 14, No. 2, pp.253-266, 2005.

[19] M. Deng and B. Preneel, "On Secure and Anonymous Buyer-seller Watermarking Protocol," Proceeding of ICIW, pp.524-529, 2008.

[20] NIST FIPS PUB 186-2, "Digital Signature Standard," National Institute of Standards and Technology, U. S. Department of Commerce, 2001.

[21] N. Koblitz, A. Menezes, and S. Vanstone, "The State of Elliptic Curve Cryptography," Designs, Codes and Cryptography, Vol. 19, pp.173-193, 2000.

[22] N. Hopper, D. Molnar, and D. Wagner, "From Weak to Strong Watermarking," Proceeding of Theory of Cryptography, LNCS 4392, pp.362-382, 2007.

[23] S. Han, M. Fujiyoshi, and H. Kiya, "An Efficient Reversible Image Authentication Method," IEICE Trans. on Fundamentals, Vol. E91-A, pp.1907-1914, 2008.

[24] Q. Huang, G. Yang, D. S. Wong, and W. Susilo, "Ambiguous Optimistic Fair Exchange," Advances in Cryptology - ASIACRYPT 2008, LNCS 5350, pp.74-89, 2008.

[25] V. V. Das, "Buyer-seller Watermarking Protocol for an Anonymous Network Transaction," Proceeding of ICETET, pp.807-812, 2008.

[26] W. Diffie and M. Hellman, "New Directions in Cryptography," IEEE Transactions on Information Theory, Vol. IT-22, No. 6, pp.644-654, 1976.

[27] W. S. Jaung, "Efficient Three-party Key Exchange using Smart Cards," IEEE Transactions on comsumer Electronics, Vol. 50, No. 2, pp.619-624, 2004.

[28] Y. Hu, "A Watermarking Protocol for Privacy Tracing," Proceeding of ISECS, pp.882-885, 2008.

[29] Z. Li, J. Higgins, and M. Clement, "Performance of Finite Field Arithmetic in an Elliptic Curve Cryptosystem," Ninth IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunications Systems (MASCOTS'01), pp.249-256, 2001.

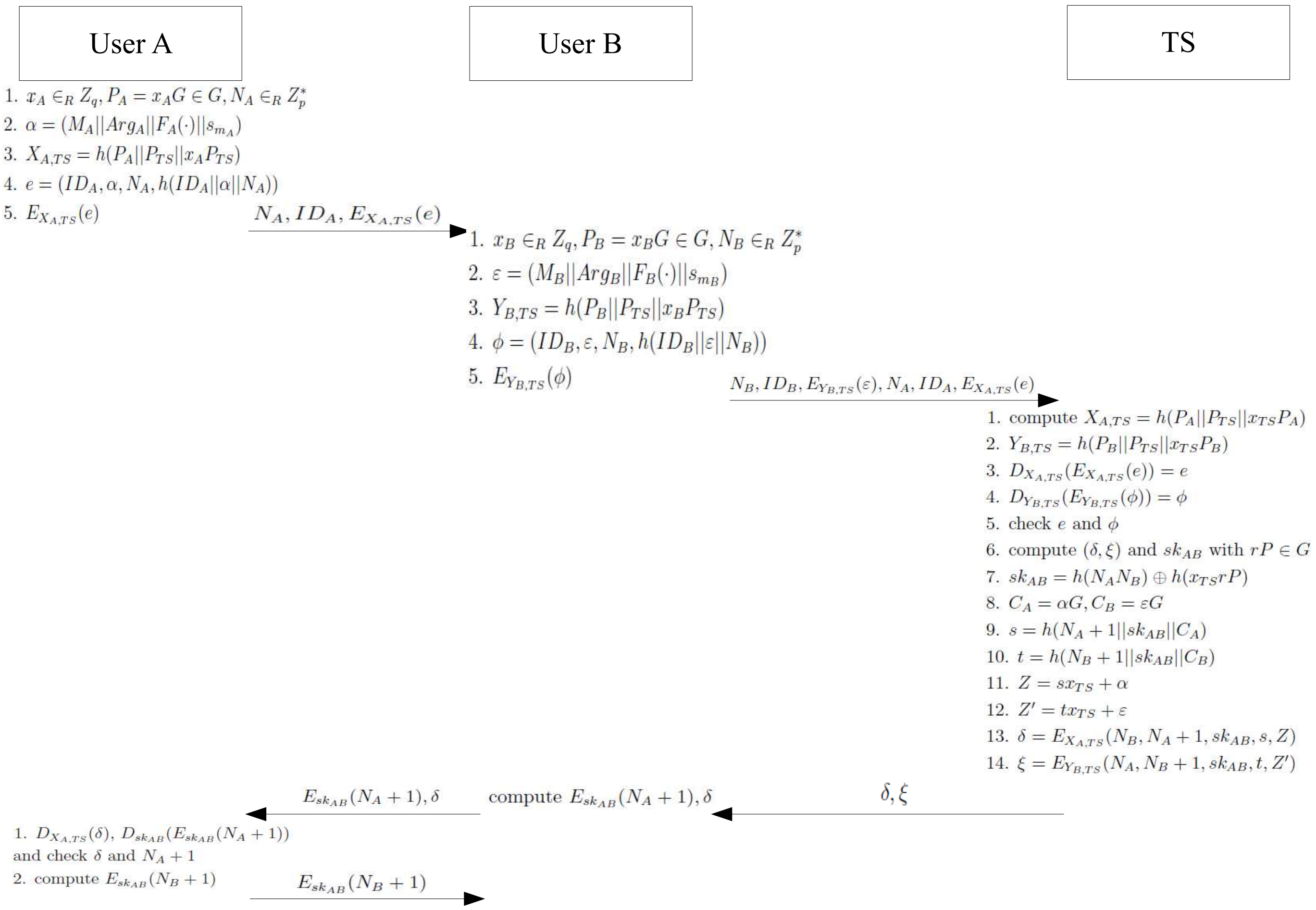
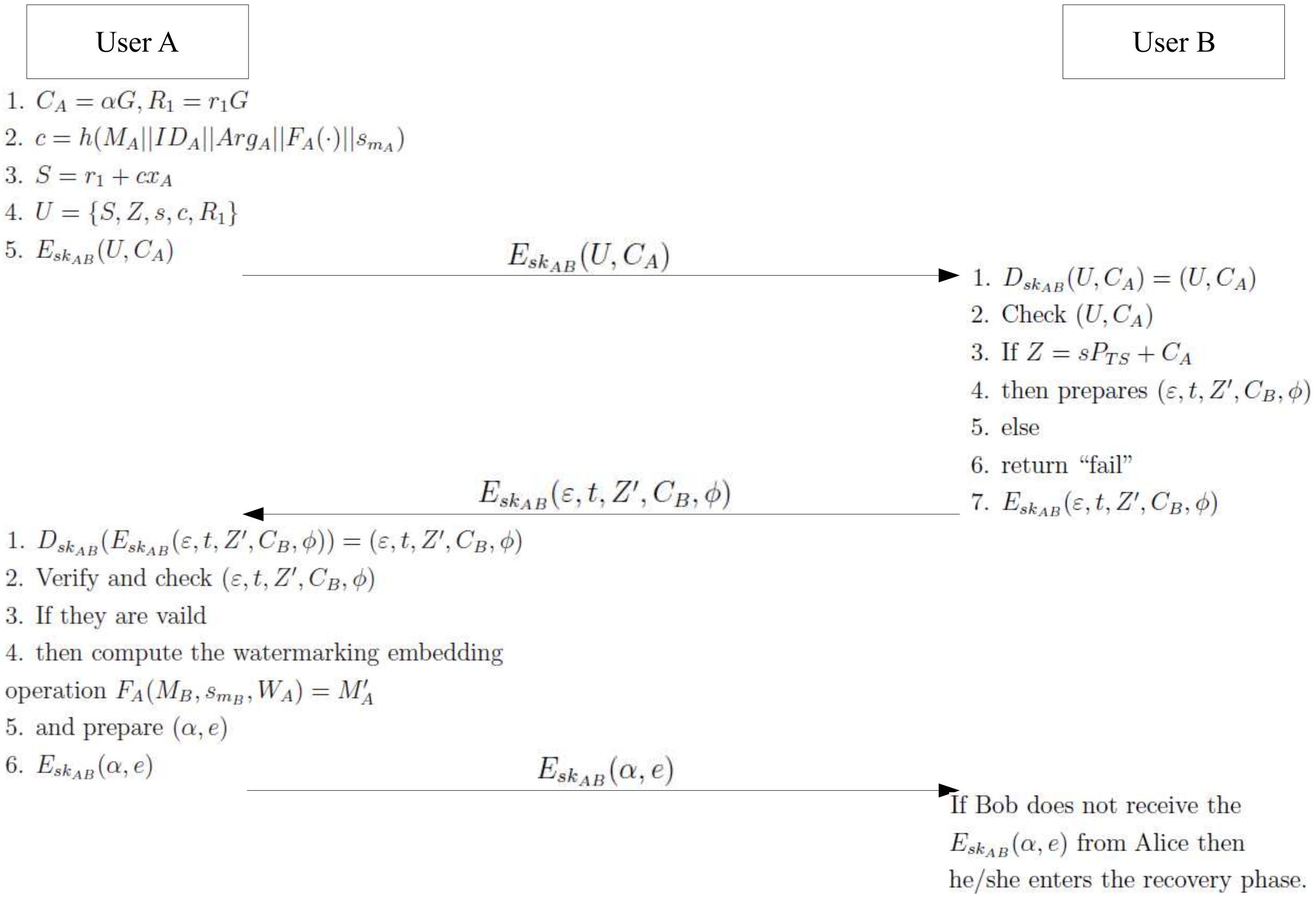

User A
User B
TS
1. $x_A \in_R Z_q, P_A = x_A G \in G, N_A \in_R Z_p^*$
2. $\alpha = (M_A||Arg_A||F_A(\cdot)||s_{m_A})$
3. $X_{A,TS} = h(P_A||P_{TS}||x_A P_{TS})$
4. $e = (ID_A, \alpha, N_A, h(ID_A||\alpha||N_A))$
5. $E_{X_{A,TS}}(e)$
$N_A, ID_A, E_{X_{A,TS}}(e)$
1. $x_B \in_R Z_q, P_B = x_B G \in G, N_B \in_R Z_p^*$
2. $\varepsilon = (M_B||Arg_B||F_B(\cdot)||s_{m_B})$
3. $Y_{B,TS} = h(P_B||P_{TS}||x_B P_{TS})$
4. $\phi = (ID_B, \varepsilon, N_B, h(ID_B||\varepsilon||N_B))$
5. $E_{Y_{B,TS}}(\phi)$
$N_B, ID_B, E_{Y_{B,TS}}(\varepsilon), N_A, ID_A, E_{X_{A,TS}}(e)$
1. compute $X_{A,TS} = h(P_A||P_{TS}||x_{TS} P_A)$
2. $Y_{B,TS} = h(P_B||P_{TS}||x_{TS} P_B)$
3. $D_{X_{A,TS}}(E_{X_{A,TS}}(e)) = e$
4. $D_{Y_{B,TS}}(E_{Y_{B,TS}}(\phi)) = \phi$
5. check e and ϕ
6. compute (δ, ξ) and sk_{AB} with $rP \in G$
7. $sk_{AB} = h(N_A N_B) \oplus h(x_{TS} rP)$
8. $C_A = \alpha G, C_B = \varepsilon G$
9. $s = h(N_A + 1||sk_{AB}||C_A)$
10. $t = h(N_B + 1||sk_{AB}||C_B)$
11. $Z = s x_{TS} + \alpha$
12. $Z' = t x_{TS} + \varepsilon$
13. $\delta = E_{X_{A,TS}}(N_B, N_A + 1, sk_{AB}, s, Z)$
14. $\xi = E_{Y_{B,TS}}(N_A, N_B + 1, sk_{AB}, t, Z')$
δ, ξ
compute $E_{sk_{AB}}(N_A + 1), \delta$
$E_{sk_{AB}}(N_A + 1), \delta$
1. $D_{X_{A,TS}}(\delta)$, $D_{sk_{AB}}(E_{sk_{AB}}(N_A + 1))$
and check δ and $N_A + 1$
2. compute $E_{sk_{AB}}(N_B + 1)$
$E_{sk_{AB}}(N_B + 1)$

Fig. 1. The authentication and key agreement phase



User A:

1. $C_A = \alpha G, R_1 = r_1 G$
2. $c = h(M_A||ID_A||Arg_A||F_A(\cdot)||s_{m_A})$
3. $S = r_1 + cx_A$
4. $U = \{S, Z, s, c, R_1\}$
5. $E_{sk_{AB}}(U, C_A)$

User B:

1. $D_{sk_{AB}}(U, C_A) = (U, C_A)$
2. Check (U, C_A)
3. If $Z = sP_{TS} + C_A$
4. then prepares $(\varepsilon, t, Z', C_B, \phi)$
5. else
6. return "fail"
7. $E_{sk_{AB}}(\varepsilon, t, Z', C_B, \phi)$

User A:

1. $D_{sk_{AB}}(E_{sk_{AB}}(\varepsilon, t, Z', C_B, \phi)) = (\varepsilon, t, Z', C_B, \phi)$
2. Verify and check $(\varepsilon, t, Z', C_B, \phi)$
3. If they are vaild
4. then compute the watermarking embedding operation $F_A(M_B, s_{m_B}, W_A) = M'_A$
5. and prepare (α, e)
6. $E_{sk_{AB}}(\alpha, e)$

If Bob does not receive the $E_{sk_{AB}}(\alpha, e)$ from Alice then he/she enters the recovery phase.

Fig. 2. The watermark exchange phase

TABLE I
PROPERTIES COMPARISONS

	The security requirements					
	P1	P2	P3	P4	P5	P6
Ours	Yes	Yes	Yes	Yes	Low	$Optinal$
[9]	No	No	No	Yes	$High$	$Robust$
[17]	No	No	No	Yes	$High$	$Robust$
[19]	No	No	No	Yes	$High$	$Robust$
[25]	No	No	No	Yes	$High$	$Robust$
[28]	No	No	No	Yes	$High$	$Robust$

Yes: Satisfied; No: Not satisfied
P1: Watermark Exchange
P2: Authenticaiton and Key Agreement
P3: Without Watermark Certificate Authority
P4: Watermark Ownership
P5: Computation Cost (Low/Medium/High)
P6: Optional(robust or fragile watermark)/Robust Watermark

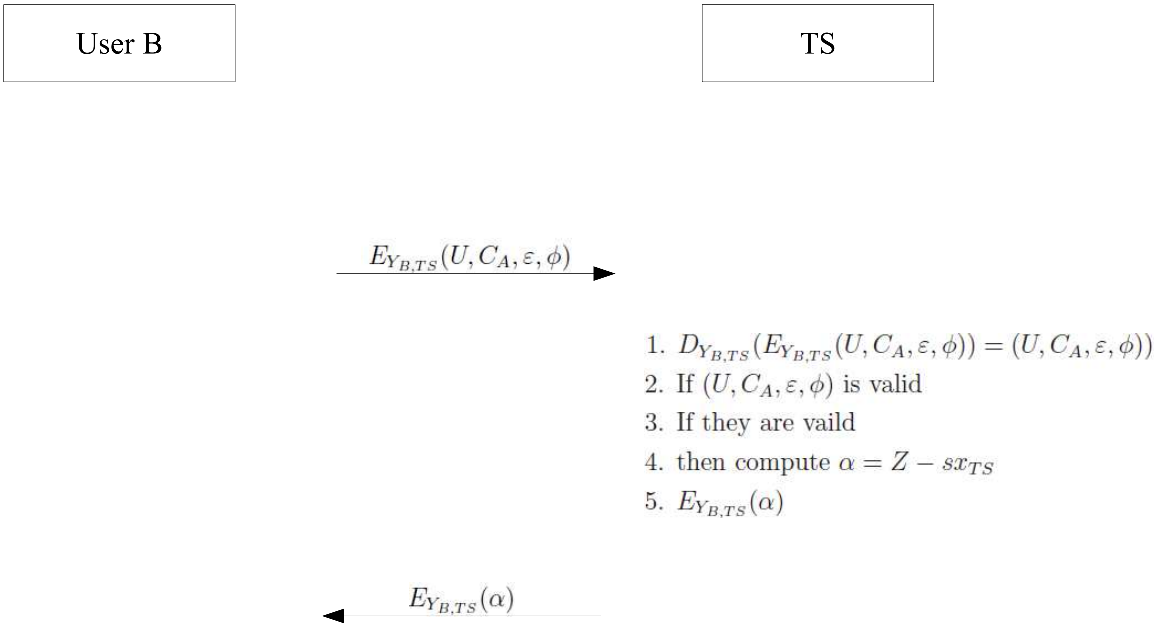

User B
TS
$E_{Y_{B,TS}}(U, C_A, \varepsilon, \phi)$
1. $D_{Y_{B,TS}}(E_{Y_{B,TS}}(U, C_A, \varepsilon, \phi)) = (U, C_A, \varepsilon, \phi))$
2. If $(U, C_A, \varepsilon, \phi)$ is valid
3. If they are vaild
4. then compute $\alpha = Z - sx_{TS}$
5. $E_{Y_{B,TS}}(\alpha)$
$E_{Y_{B,TS}}(\alpha)$

Fig. 3. The recovery phase

TABLE II
EFFICIENCY COMPARISONS

Ours	5H+14EC_M+19$SymEnc/Dec$+2W	$\cong$ 421M+2W
[9]	9Exp+1D+4H+1W	$\cong$ 2162M+1W
[17]	6Exp+2D+3S+1W	$\cong$ 2160M+1W
[19]	5Exp+1D+2S+1W	$\cong$ 1680M+1W
[25]	6Exp+3S+1W	$\cong$ 2180M+1W
[28]	(2n+3)Exp+1D+(1n+2)S+2W	$\cong$ (3n+3)×240M+2W

M: Modular Multiplication Operation
Exp: Exponential Operation
E: Public Key Encryption Operation
D: Public Key Decryption Operation
S: Signature Operation
H: Hash Operation
W: Watermark Embedding Operation
$SymEnc/Dec$: Symmetric Encryption or Decryption
EC_M: Scalar Multiplication of Elliptic Curve Point
n: The number of Watermarks