A Hybrid Threshold Group Signature Scheme with Distinguished Signing Authority

Dao Tuan Hung[1(✉)], Nguyen Hieu Minh[2], and Nguyen Nam Hai[2]

[1] National Laboratory of Information Security, 34A Tran Phu, Ba Đinh, Hanoi, Vietnam
daotuanhung@gmail.com
[2] Academy of Cryptography Technique, Hanoi, Vietnam
{hieuminhmta, hainamnguyen}@gmail.com

Abstract. This paper proposes a new hybrid threshold group signature scheme with distinguished signing authority to provide all proof of member signing processes in case of dispute internally and internal integrity of multisignature generation process. In practical, the proposed scheme has more controls to an organization by using the threshold mechanism and allowing a limited number of members who can authorize transactions while allowing the group to grow. Moreover, the risk of losing group secret either by an APT attack or by any subset of corrupt members can be eliminated. The proposed scheme is secure based on the hardness of elliptic curve discrete logarithm problem (ECDLP).

1 Introduction

Digital signatures are widely used in many aspects of electronic life. They are designed to be part of security services such as authentication, data integrity and non-repudiation. To date, many schemes such as multisignature [1], group signature [2], traditional signature [3, 4] have been proposed. A multisignature scheme is designed so that a group of users can sign a single document [1]. Multisignatures can be categorized into such as without signing authority or with distinguished signing authority [5, 6].

Group signatures, first introduced by Chaum and van Heyst in [2]. In a group signature scheme, any group member of a given group can sign an electronic document on behalf of the group in an anonymous and unlinkable way. On the other side, anyone only needs the group public key to verify the validity of a group signature. In case of a dispute, only group manager can reveal a member who signed, while other group members neither can identify the identity of the signer nor determine whether multiple signatures are produced by the same group member. To prevent a single corrupt member illegally authorizing a transaction, the threshold signature scheme can be used. A large number of studies were published on (t, n) threshold signature schemes. Schemes at [7–11] based on various hard problems such as RSA system, discrete logarithm (DLP), Chinese Remainder Theorem (CRT), ECDLP. However, schemes at [7, 8, 10] are not secure ones [12–14]. Signature of scheme at [11] cannot be verified by just one verifier and therefore is not practical. At [15] presents an idea of masking group's private key to prevent group members who can collaborate to recover it but

V. Bhateja et al. (eds.), *Information Systems Design and Intelligent Applications*, Advances in Intelligent Systems and Computing 672,
https://doi.org/10.1007/978-981-10-7512-4_7

need a trusted party that use all member private keys to construct group signature and so one can argue that doesn't meet requirement for non-repudiation.

Moreover, previous schemes often assume the number of users being controlled by an adversary less than threshold number [9, 10, 14, 16] in order to keep group's private key safe. However, if the number of members grow, secret shared group keys will be delivered to more and more people. Therefore, there are more chances for group signature scheme to be unsecure. Previous group signature schemes lack mechanisms to maintain a balance between security and scaling of group. Especially, when considering the situation, a company might suffer Advanced Persistent Threat (APT) attacks. This leads to a valid security concern that group secret key might be lost by either corrupt members who can collaborate and recover the key. Another bad situation is many personal computers were targeted and compromised under an APT attack by hackers or state-sponsored APT campaigns that cause the group secret key being steal undetected.

Research at [17] proposed a group signature schemes that have distinguished signing authorities based on the multisignature protocols. Scheme at [17] requires a group manager to collect and issue signature.

This paper proposes a new threshold group signature protocol based on ECDLP that is highly secure, constant length and short signature, distinguished signing authority. The proposed scheme can protect group's private key from being revealed by any set of corrupt signers or hacker's threat. The proposed scheme allows group secret key shares to be kept on limited privilege signers only while allowing new people to join the group without recalculating group public key and easy revocation.

2 Proposed Group Signature Protocol

Currently, cryptographic protocols based on elliptic curves (EC) over finite field have been applied. In the proposed scheme, we use the EC, which order contains a sufficiently large prime divisor q (more than 256 bits) and a point G having order equal to q.

System initialization: Assume that a large group has n privilege signers who can keep company's secret key shares (for example: directorate board) and any number of normal staffs. Only privilege signers have shared company's secret key shares. Group's policy requires that at least t ($t < n$) privilege signers must join signing process to make a valid group signature. Here are four roles in the proposed scheme:

Group Manager (*GM*): Group manager is a trusted party of the group signature scheme. He creates the secret parameters for the group, calculates and distributes secret key shares to privilege members; add, removes group members, and reveals the identity of the group member in a special case.

Distributed Center (*DC*): special hardened servers of the group that communicate with all signers during signing process. DC calculates some secret parameters needed by signers to create signatures for each transaction. Moreover, all signer's shared signatures are safely stored on DC. Only GM can open DC when needed.

Normal signers: digitally sign on their own work inside large group document.

Privilege signers: digitally sign on their own work inside large group document. With enough t signatures of them, a signature of the group can be generated.

An example of this could be: A complex CAD design files of a construction company need to be internally signed by different people including signatures of t important people such as head of financial office, planning office, directorate board to form a valid group signature. The company wants to hide its internal structure. Head of financial office, planning office, and member of directorate board are privilege signers. In the case design defects are found, the company can traceback and see who is responsible for defect parts of the design.

System preparation phase:
Group manager (GM) chooses two random integers A_0, SE ($1 < A_0 < q$, $1 < SE < q$). A_0 is group's private key which is unchanged. SE is another secret number but can be changed to another value when GM decides to redistribute secret key shares. GM calculates secret key shares for n privilege signers following the cryptographic technique of Shamir's perfect secret sharing scheme [18].

$$\begin{aligned} &f(x) = (SE * A_0 + s_1 x + s_2 x^2 + \cdots + s_{t-1} x^{t-1}) \bmod q \\ &(v_i, y_i),\ i = 0,\ 1,\ \ldots,\ n; y_i \equiv f(v_i) \bmod q \end{aligned} \tag{1}$$

Values $s_1, s_2, \ldots, s_{t-1}$ are random integers with $(1 < s_i < q)$. These values are known only by GM. n values $(v_i,\ y_i)$ are secretly sent to n privileges, where y_i is secret shared value of signer i, and v_i is i-th signer's identity. All v_i from (1) are published inside privilege group. Value $A_0 * SE$ can be recovered by any t privilege people or devices who hold secret shares [18], while any number of privileges less than t can reveal nothing about a value $A_0 * SE$:

$$A_0 * SE \equiv \sum_{k=1}^{t} y_k \left[\prod_{i=1, i \neq k}^{t} \frac{-v_i}{v_k - v_i} \right] \bmod q \tag{2}$$

Each privilege who sign will use this equation to calculate of his share during signing process:

$$f_j \equiv y_j \left(\prod_{i=1, i \neq j}^{t} \frac{-v_i}{v_j - v_i} \right),\ (j = 1,\ 2,\ \ldots,\ n) \tag{3}$$

GM calculates public key of the group as an EC point: $P_{gm} = A_0 * G$ and another EC point $P_{DC} = (SE - 1) * A_0 * G$. Point P_{gm} is group public key, which can be used by anyone to verify group signatures. GM keeps values $h_{SE} = h(SE)$ and P_{DC} on DC.

Key generation phase:

Each member i-th in the group generates their private key as a random number $k_i (1 < k_i < q)$, and then public key computed as the point $P_i = k_i G$, i = (1, 2, …, N).

Group signature generation phase:

1. Assume N people including t privilege signers together sign the document set $M = m_1 || m_2 || \ldots || m_N$. M is sent to DC prior to the signing process, the DC

calculates values $h_i = H(m_i)$, $z_i = H(h(M)||h_i||P_i||h_{SE})$. Then values (z_i, h_i) are sent to corresponding signer i-th.

2. The DC calculates an EC point as follow:

$$U = h_1 z_1 P_1 + h_2 z_2 P_2 + \cdots + h_N z_N P_N \tag{4}$$

U is the first element of group signature.

3. Each signer i-th chooses a random integer t_i $(1 < t_i < q)$, and calculates $R_i = t_i G$, then sends R_i to DC.
4. DC calculates an EC point:

$$R = R_1 + R_2 + \cdots + R_N \tag{5}$$

and the second element of group signature:

$$e = H(M||x_R||x_U) \tag{6}$$

where x_R and x_U are x-coordinates of EC points R and U, respectively. DC sends the value e to the group members who initiated the protocol.

5. Each signer (privilege or normal) computes their signature share s_i on his assigned part (m_i) of the document differently as follow:
If i-th signer is normal signer he computes:

$$s_i = t_i + h_i k_i z_i e \bmod q \tag{7}$$

If i-th signer is privilege signer he computes an EC point $V_i = f_i e G$ and then s_i:

$$s_i = f_i e + t_i + h_i k_i z_i e \bmod q \tag{8}$$

Normal signer sends (s_i) to DC, privilege sends two values (s_i, V_i) to DC.

6. DC verifies s_i of a normal signer (s_i is sent by U_i) if DC received s_i only by checking following equation:

$$R_i = s_i G - z_i h_i e P_i \tag{9}$$

DC verifies s_i of a privilege signer (if DC received two values (s_i, V_i)) by checking following equation:

$$R_i = s_i G - V_i - z_i h_i e P_i \tag{10}$$

7. If the equation holds for all s_i, DC computes the third, fourth elements of group signature $P_V = e P_{DC}$ and:

$$s = s_1 + s_2 + \cdots + s_N \bmod q \tag{11}$$

Group signature of M is a tuple (U, P_V, e, s), which consists of two EC points and two integer values.

Group Signature verification:

1. Verifier computes the hash of the document $M = m_1||m_2||\cdots||m_N$ as $h = H(M)$.
2. Verifier uses the group public key P_{gm} and the signature $(U,\ P_V,\ e,\ s)$ to compute an EC point $\tilde{R} = sG - P_V - e(U + P_{gm})$, and value $\tilde{e} = H(M||x_{\tilde{R}}||x_U)$. Accept the signature only if $\tilde{e} \equiv e$.

3 Analysis of the Proposed Group Signature Scheme

3.1 Proof of Correctness

1. Share signature verification equation (for privilege signer i-th):

$$\begin{aligned} R_i &= s_iG - V_i - eh_iz_iP = Gf_ie + t_iG - V_i + k_ih_iz_ieG - k_ih_iz_ieG \\ &= V_i + t_iG - V_i + k_ih_iz_ieG - ez_ih_ik_iG = t_iG \equiv R_i \end{aligned}$$

2. Share signature verification equation (for normal signer i-th):

$$\begin{aligned} R_i &= s_iG - eh_iz_iP = t_iG + k_ih_iz_ieG - k_ih_iz_ieG \\ &= t_iG + k_ih_iz_ieG - k_ih_iz_ieG = t_iG \equiv R_i \end{aligned}$$

3. Signature verification equation:

With total N signers including t privilege signers, and equations at (2), (3), (7), (8) we have:

$$\begin{aligned} \tilde{R} &= sG - P_V - e(U + P_{gm}) = sG - e(U + P_{DC} + P_{gm}) \\ &= \left(\sum_{i=1}^{N} s_i\right)G - e\left(P_{gm} + P_{DC} + \sum_{i=1}^{N} h_iz_iP_i\right) \\ &= \left(\sum_{i=1}^{N}(t_i + k_ih_iz_ie) + \sum_{i}^{t} f_ie\right)G - e\left(A_0G + (SE-1)A_0G + \sum_{i=1}^{N} k_ih_iz_iG\right) \\ &= \left(\sum_{i=1}^{t} f_ie + \sum_{i=1}^{N} t_i + \sum_{i=1}^{N} k_ih_iz_ie - eA_0 + e(SE-1)A_0 - \sum_{i=1}^{N} k_ih_iz_ie\right)G \\ &= \left(\sum_{i=1}^{t} f_ie + \sum_{i=1}^{N} t_i - eA_0 + e(SE-1)A_0\right)G = \left(SEeA_0 + \sum_{i=1}^{N} t_i - SEeA_0\right)G \\ &= \sum_{i=1}^{N} t_iG = R \Rightarrow \tilde{e} = H(M||x_{\tilde{R}}||x_U) = H(M||x_R||x_U) = e. \end{aligned}$$

If number of privilege signers who participated less than t or simply absent, above equation does not hold and signers cannot create a valid group signature.

Signature length: Signature of a document is a tuple of two integers and two EC point $(U,\ P_V,\ e,\ s)$, in the case of 128-bit security, q can be chosen with size around 256 bits and signature length will approximately 1536 bits. Compared with group schemes in [19], the proposed scheme has shorter signature length. If choose 80-bit security signature length will approximately 960 bits, with $|q|$ = 160 bits).

3.2 Security Analysis

Theorem 1: *Protection of private keys and secret key shares.*

Proof. Normal and privilege signer use private key k_i to sign on a partial message m_i follow (7) and (8) respectively. In both cases two secret random values are used t_i and e. Using adaptive message attack is invalid with the scheme. *Therefore, private keys and secret key shares are protected from other members.*

Theorem 2: *Any subset of t privileged signers out of n to generate a valid signature of the group, but they cannot recover private key of group* A_0.

Proof. If all privileged signers are curious, they can get a value $A_0 * SE$ by following (2). In order to find A_0 from $A_0 * SE$, they have to try each possible guess value of SE' to get A_0' and check if $A_0' * G = P_{gm}$, with assumption of Elliptic curve problem is hard, this task is computational infeasible. Compared with previous works [9, 15, 20], the proposed scheme can protect group secret with any number corrupt members. Therefore, the proposed scheme is secure against conspiracy attack [12, 21].

Theorem 3: *Signers cannot bypass DC to create signature*.

Proof. An element of signature $P_V = e * P_{DC}$,which P_{DC} is a private EC point kept on DC only and e (6) is a value related to the document and signer public keys. Without P_{DC}, a signature cannot pass verification process. Often, a group wants to keep records of all transactions. If signers in a group signature scheme can collaborate without a system to keep track of all activities, this situation might cause issues for large group. At DC, a company can place more security protections than it can do with individual personal devices.

Theorem 4: *Suffering an APT attack, company group secret remains safe*.

Proof. During signing and verification process, group secret A_0 is not reconstructed at any step. So, if suffering an APT attack many computers might be compromised, but hackers cannot use memory forensic technique or network sniffer to find A_0. Assume hacker that can get all shares secrets of n privilege signers and following (2), they can recover $A_0 * SE$. They cannot get A_0 directly from $A_0 * SE$ and P_{DC} because of ECDLP problem. Values $h_{SE} = h(SE)$ and P_{DC} are stored on DC, but they are produced of safe hash function and multiplication on elliptic curve, respectively. *Group secret is protected with APT attack.*

Traceability
In the case of dispute, group manager needs to convince that specific signers signed sessions of document. In order to identify signer, GM can show values that related to only signer i-th (*privilege or normal signer*): h_i, R_i, $z_i = H(h(M)||h_i||P_i||h_{SE})$, $s_i = t_i + h_i k_i z_i e \bmod q$ $s_i = f_i e + t_i + h_i k_i z_i e \bmod q$ and an EC point $V_i = f_i eG$. These values satisfy check equations for normal signers (9) or privilege signers (10) so only signer i-th is responsible for document session with $h_i = h(M_i)$. Thus, the scheme provides distinguished signing authority feature internally. Disclosure of R_i, V_i is safe because they are produce by the multiplication on Elliptic curve.

Unforgeability
Signer i-th needs approval from DC to get z_i to calculate his share signatures that pass a verification equation at (9) or (10) for normal or privilege signer, respectively. Generating group signature needs cooperation of DC with t privilege signer members and only DC can produce a valid group manager with valid member's shared signatures.

Unlinkability
Identifying the two different signatures generated by one member (or group of members) is impossible, except for the group manager.

Exculpability
In the proposed scheme, no member (or many corrupt members work together) can forge signatures of other. This is because signature of member is calculated not only by private key but also on R_i, z_i, e which are calculated for specific signer. Therefore, to forge the signature of a group member, they need to pass the signature check equation for each member of the group manager. That means they must break the ECDLP.

4 Conclusion

In this paper, a new threshold group signature scheme based on usage of Elliptic curve is proposed. The new scheme has these new practical benefits:

1. Scaling group without worrying about group secret loss; enables only a limited number people can hold secret key shared while allows number normal members to grow; Practical revocation and joining group.
2. Compared with previous threshold group signature schemes, no chance for an adversary or dishonest group of signers can steal group secret.
3. Reduce the risk of unexpected transaction of threshold group signature scheme by using a trusted DC.
4. Provides distinguished signing authority feature of multisignature internally.

The size of the output signature is comparable with known schemes. In practically, the proposed protocol provides more control to an organization by threshold mechanism and allowing a limited number of members who can authorize transactions. The scheme possesses many security advantages compared with previous works.

References

1. Harn, L.: Digital multisignature with distinguished signing authorities. Electronics Letters. Vol.35 (1999) 294–295.
2. Chaum, D., Van Heyst: Group signatures. Advances in Cryptology—EUROCRYPT 1991. LNCS. Springer Heidelberg (1991) 257–265.
3. Rivest, Ronald L., Adi Shamir, Leonard Adleman.: A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM. Vol.21 (1978): 120–126.
4. ElGamal, Taher.: A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE transactions on information theory. Vol.31 (1985): 469–472.
5. Harn, L.: New digital signature scheme based on discrete logarithm. Electronics Letter, Vol. 30. (1994) 396–398.
6. Huang, Hui-Feng, Chin-Chen Chang.: Multisignatures with distinguished signing authorities for sequential and broadcasting architectures." Computer Standards & Interfaces. Vol.27. (2005) 169–176.
7. Wang, Ching-Te, Chu-Hsing Lin, Chin-Chen Chang.: Threshold signature schemes with traceable signers in group communications. Computer Communications. Vol.21. (1998) 771–776.
8. Harn, Lein.: Group-oriented (t, n) threshold digital signature scheme and digital multisignature. IEE Proceedings-Computers and Digital Techniques 141.5 (1994): 307–313.
9. Harn, Lein, and Feng Wang.: Threshold Signature Scheme without Using Polynomial Interpolation. IJ Network Security. Vol.18. (2016): 710–717.
10. Yu, Yuan-Lung, and Tzer-Shyong Chen.: An efficient threshold group signature scheme. Applied Mathematics and Computation Vol.167. (2005) 362–371.
11. Mante, Ganesh, and S. D. Joshi.: Discrete logarithm based (t, n) threshold group signature scheme. International Journal of Computer Applications 21.2. (2011) 23–27.
12. Michels, Markus, and Patrick Horster.: On the risk of disruption in several multiparty signature schemes. International Conference on the Theory and Application of Cryptology and Information Security. Springer Berlin Heidelberg, (1996).
13. Tseng, Yuh-Min, and Jinn-Ke Jan.: Attacks on threshold signature schemes with traceable signers. Information Processing Letters Vol.71 (1999) 1–4.
14. Shao, Z.: Repairing Efficient Threshold Group Signature Scheme. International Journal of Network Security, Vol.7, No.2, (2008) 218–222.
15. Zhao, Lin-Sen; LIU, Jing-Mei. (t, n) Threshold Digital Signature Scheme with Traceable Signers against Conspiracy Attacks. In: Intelligent Networking and Collaborative Systems (INCoS), 2013 5th International Conference on. IEEE, (2013) 649–651.
16. Bozkurt, Ilker Nadi, Kamer Kaya, Ali Aydın Selçuk.: Practical threshold signatures with linear secret sharing schemes. International Conference on Cryptology in Africa. Springer Berlin Heidelberg. (2009) 167–178.
17. Tuan, H.D., Nguyen, H.M., Tran, C.M., Nguyen, H.N., Adreevich, M.N.: Integrating Multisignature Scheme into the Group Signature Protocol. Advances in Information Communication Technology: Proceedings of the International Conference, ICTA 2016. Springer International Publishing. (2017) 294–301.
18. Shamir, A.: How to Share a Secret. Communications of ACM, 22. (1979) 612–613.
19. Laguillaumie, Fabien, et al.: Lattice-based group signatures with logarithmic signature size. International Conference on the Theory and Application of Cryptology and Information Security. Springer Berlin Heidelberg. (2013) 41–61.

20. Bozkurt, Ilker Nadi; Kaya, Kamer; Selçuk, Ali Aydın.: Practical threshold signatures with linear secret sharing schemes. In: International Conference on Cryptology in Africa. Springer Berlin Heidelberg. (2009) 167–178.
21. Boldyreva, Alexandra.: Threshold signatures, multisignatures and blind signatures based on the gap-Diffie-Hellman-group signature scheme. In: International Workshop on Public Key Cryptography. Springer Berlin Heidelberg. (2003) 31–46.