A Directed Threshold Signature Scheme

Manoj Kumar

Abstract Directed signature is a solution of such problems when signed information is sensitive to message holder/signature receiver. Generally, in a directed signature, the signer is a single entity. But, when a sensitive message is signed by an organization and needs the approval of more than one entity, threshold signature scheme is a solution of this situation. To keep in mind, this paper presents a threshold directed signature scheme.

1 Introduction

Physical signature is an old and natural tool to authenticate the communication, but it does not work in electronic messages and the signer has to rely on digital signature [1]. Digital signature is a cryptographic tool to solve this problem of electronic authentication. Basically, digital signature has a self-authentication property, which means that someone has public information related to the signature, will be able to check its validity, but he/she will not able to forge this signature for other messages. This self-authentication property [2] of digital signatures is definitely suitable for many applications such as broadcasting of announcements and publication of public key certificates, but it is quiet unsuitable for some situations [3].

In some conditions, when the message are very much sensitive to the signature, receiver/message holder such that her/his medical reports, income tax related information, any personal information or most personal business transactions are these messages [4]. For these conditions, the information is signed such that only the information holder will able to verify the signature and also able to prove the

M. Kumar (✉)
Department of Mathematics, Rashtriya Kishan Postgraduate College, Shamli 247776, India
e-mail: yamu_balyan@yahoo.co.in

D.K. Lobiyal et al. (eds.), *Next-Generation Networks*, Advances in Intelligent Systems and Computing 638, https://doi.org/10.1007/978-981-10-6005-2_18

validity of the signature to a third person, whenever it is required. These types of signatures are known as directed signatures [3–6]. In a directed signature scheme [3], the receiver always has full control over the process of signature verification. No other person can check the validity of this type signature without the help of signer/receiver [1].

In most situations, generally a single identity creates signature on the message. But there are so many conditions when the message is on behalf of a group/organization, that message may require the approval or consent of several people [2]. In these conditions, the signature is created by more than one identity rather than by a single identity [5]. In case of large bank transaction, which requires the signature of more than one person [7]. In such a condition, the problem can be solved by having a separate digital signature for every required signer, but this type of solution makes the verification process very typical [8]. This problem can be solved with the help of threshold signature [8]. The (t, n) threshold signature schemes [2, 7–10] are used to solve these problems. Threshold signatures are based upon the concept of threshold cryptography [9, 11, 12].

1.1 Paper Organization

Section 2 is about some basic tools. In Sect. 3, we present a threshold directed signature scheme. Section 4 discusses the security of the proposed scheme. An illustration of the scheme is discussed in Sect. 5. Conclusion is in Sect. 6.

2 Preliminaries: Some Basic Tools

2.1 In This Paper, We Will Use the Following Public Parameters

- p: a prime number.
- q: a prime number and $q|p - 1$.
- g: a generator [3] of order q in Z_P^*.
- h: one-way hash function [13].

It is assumed that user A selects an integer $x_A \in Zq$ and will be able to compute a relative value/integer $y_A = g^{x_A} \bmod p$. Here, the integer x_A is the secret/private key of the user A, and y_A is his/her public key.

2.2 Schnorr's Signature Scheme

In the above scheme, the signature of the signer A on a message m is given by a pair (r_A, S_A), where, $r_A = h(gk_A \bmod p, m)$, and $S_A = k_A - x_A \cdot r_A \bmod p$. The integer k_A is random and secret/private to A. The signature is verified by checking the equality.

$$r_A = h(gS_A y r_A \bmod p, m).$$

3 Directed Threshold Signature Scheme

This section presents a threshold directed signature scheme [13, 14]. Suppose a group G of n designated users, out of which any t members are able to signed a message m. In our scheme, the message holder/signature receiver B will be able to check the signature authenticity, and he/she can prove this message authenticity to a third person C, whenever it is needed. It should be noted that no one other than the message holder B can check the validity of this kind of signature without the help of holder B [14]. We describe a construction of threshold directed signature scheme for this situation as follows.

In our scheme, there exists a trusted share distribution center (SDC) [13, 14], which is able to determine the secrets parameters and the secret shares $v_i, i \in G$ for all members of the group. Again assume that H be a subset of G, containing t members. We also have a designated combiner DC for collecting partial signatures of each participant of subgroup H. Any shareholders in the group/subgroup have equal authority with respect to the main secret key for signature generation. In the proposed scheme, the generation of the required directed signature needs t signers out of n signers and interaction with DC. This scheme has the following steps.

3.1 Generation of Secret Key and Secret Shares for Group

(a) SDC also selects a polynomial

$$g(x) = a_0 + a_1 x + \cdots a_{t-1} x^{t-1} \bmod q, \text{ with } a_0 = K = g(0).$$

(b) SDC compiles group public key, y_G, as, $y_G = g^{g(0)} \bmod p$.
(c) SDC computes private shares v_i for each user in group G, as,

$$v_i = g(u_i) \bmod q.$$

Here, u_i is public information related to user i in the group G.
(d) SDC transfers v_i to each user in a secret manner.

3.2 Generation of Partial Signature by Any t Signer

Let any t signers out of n signers agree to sign a message m for receiver B, they generate the signature using following steps.

(a) Each member i randomly picks K_{i_1} and $K_{i_2} \in Zq$ and then computes

$$w_i = gK_{i_2} - K_{i_1} \bmod p \quad \text{and} \quad z_i = y_B K_{i_2} \bmod p.$$

(b) Each signer computes Z, W, and R as

$$W = \prod_{i \in H} w_i \bmod q, Z = \prod_{i \in H} z_i \bmod q, \quad \text{and} \quad R = h(Z, W, m) \bmod q.$$

(c) Each signer i modifies corresponding share, as

$$\mathrm{MS}_i = v_i. \prod_{j=1, j \neq i}^{t} \frac{-u_j}{u_i - u_j} \bmod q.$$

(d) Each signer i computes

$$s_i = K_{i_1} - \mathrm{MS}_i \cdot R \bmod q.$$

(e) DC collects the partial signatures and produces

$$S = \sum_{i=1,}^{t} s_i \bmod q.$$

(f) $\{S, W, R, m\}$ is desired directed signature.

3.3 Verification of Digital Signature {S, W, R, M}

(a) The signature holder B recovers $\mu = gS(y_G)RW \bmod p$ and recovers $Z = \mu x_B \bmod p$.
(b) The signature holder B checks the validity of signature by verifying $R = h(Z, W, m) \bmod q$.

3.4 *Proof of Validity by Signature Receiver to Any Third Party* C

(a) The signature holder B sends $\{S_A, W_B, r_A, m, \mu\}$ to third party.
(b) Third party checks if $r_A = h(Z_B, W_B, m) \bmod q$.
If this does not hold third party stops the process; otherwise goes to the next steps [13, 14].
(c) Signature receiver (in a zero-knowledge fashion) proves to C that $\log_{\mu} Z_B = \log_g y_B$ as follows.

- Third party selects randomly two values u and $v \in Zp$ and then finds $w = \mu u \cdot gv \bmod p$ and passes this value w to receiver.
- The signature receiver selects randomly a value $\alpha \in Zp$ and then calculates another value $\beta = w \cdot g\alpha \bmod p$ and $\gamma = \beta x_B \bmod p$, and then passes it to third party.
- The signature receiver verifies that $w = \mu u \cdot gv \bmod p$. The third party verifies $\beta = \mu u \cdot gv + \alpha \bmod p$ and $\gamma = Z_B u y_B v + \alpha \bmod p$.

In this way, the third party ensures himself that the signature receiver is an authentic user.

4 Security Discussion

This section is about the security aspect of the proposed scheme.

- Is it possible that an antagonist retrieves group secret key $g(0)$ with the help of group public key y_G? It is computationally infeasible because this is equivalent to solve a discrete logarithm problem.
- Is it possible that an antagonist recovers the secret information v_i, from the information u_i? No, it is computationally infeasible because g is selected randomly.
- Is it possible that an antagonist recovers the secret information v_i, K_{i_1} and s_i, from the equation $s_i = K_{i_1} - MS_i \cdot R \bmod q$? No, it is computationally infeasible because unknown parameters are three and the number of equation is only one.
- Is it possible that an antagonist recovers the group secret key $g(0)$ or any partial information from the equation, $S = \sum_{i=1}^{t} s_i \bmod q$? This is again computationally infeasible due the property of the equation.
- Is it possible that an antagonist impersonates a shareholder of subgroup H? To impersonate, an antagonist needs a related secret share v_i to generate corresponding secret value s_i. To obtain this secret information from the public information is computationally infeasible.

- Is it possible that an antagonist forges the digital signature $\{S, W, R, m\}$ by using the equation

$$\mu = [gS(y_G)RW] \bmod p?$$

  To recover S from the above equation is equivalent to solving a discrete logarithm problem.
- Is it possible that a group of antagonist act in collusion to recover the polynomial $g(x)$? Yes, this is possible, but this vulnerability is not a pitfall of the proposed scheme. Actually, this is the basic characteristic of the proposed scheme.

5 Illustration

To illustrate the proposed scheme, we consider that there are four users. Out of four users A, C, E, and F any two users, say, A and F can generate the directed signature for message m. The secret and public key pair $x_B = 6$, $y_B = 8$ of the receiver B. The following steps illustrate our scheme.

5.1 Generation of Group Secret Key and Partial Secret Shares

Let SDC choose $p = 23$, $q = 11$, $g = 18$, and $g(x) = 3 + 5x \bmod 11$, where $g(0) = 3$ is the group secret key. The public values u_i and corresponding secret shares v_i of users are as follows.

Users	Public value (u_i)	Secret share (v_i)
A	9	4
C	12	8
E	14	7
F	16	6

Now, the SDC computes the private/secret key as $g(0)$ and then recovers the group public key, y_G, as $y_G = 18^3 \bmod 23 = 13$.

5.2 Signature Generation by Any t Users

Users A and F out of four users agree to sign a message m for user B, then the signature generation has the following steps.

(a) The user A randomly selects $K_{a_1} = 2$, $K_{a_2} = 7$ and computes $w_1 = 3$, $z_1 = 12$. Similarly, the user F randomly selects $K_{f_1} = 5$, $K_{f_2} = 9$ and computes $w_4 = 4$, $z_4 = 9$.
(b) Both the users A and F make (w_1, w_4) and (z_1, z_4) publicly available through a broadcast channel. Once all (w_1, w_4) and (z_1, z_4) are available, each user in H computes the product Z, W, and R as

$$W = 12, Z = 16 \quad \text{and} \quad R = h(16, 12, m) \bmod 11 = 5(\text{let}).$$

(c) The users A and F compute their modified shares as $MS_A = 6$ and $MS_G = 8$.
(d) The user A uses his/her modified share $MS_A = 6$ and random integer $K_{a_1} = 2$ and calculates his/her partial signature $s_1 = 5$.
(e) The user F uses his/her modified shadow, $MS_G = 8$, and random integer $K_{f_1} = 5$ and calculates his/her the partial signature $s_2 = 9$.
(f) Both the users A and F send their partial signature to DC who produces a group signature $S = 3$.
(g) DC sends {3, 12, 5, m} to B as signature of the group G for the message m.

5.3 Signature Verification by B

(a) B computes $\mu = [18^3.13^5.\ 12] \bmod 23 = 3$ and $Z = 16$.
(b) B checks the validity of signature by computing $R = 5$.

5.4 Proof of Validity by B to Any Third Party C

(a) B sends {3, 12, 5, m, 3} to C, and C checks that $R = 5$.
(b) Now, B proves to C that $\log_3 16 = \log_{18} 8$ in a zero-knowledge fashion [15] by using the following confirmation protocol.

(i) C chooses at random $u = 11$, $v = 13$ and computes $w = 2$ and sends w to B.
(ii) B chooses at random $\alpha = 17$ and computes $\beta = 16$ and $\gamma = 4$ and sends β, γ to C.
(iii) C sends u, v to B, by which B can verify that $w = 2$.
(iv) B sends α to C, by which she can verify that $\beta = 16$ and $\gamma = 4$.

6 Conclusion

The security of this cryptosystem is [16–18] based on the discrete log problem. Only $t - 1$ shadows are not sufficient to obtain the group secret key and they will also get no information about the group secret key, until t individuals act in collusion. In this scheme, there is a designated combiner *DC* who collects the partial signature of the signer [19, 20]. We should note that there is no secret information associated with the *DC* [21–24]. Every user can compute his/her modified share under mod q. If q is not prime, then the calculation of the exponents is performed by mod $\Phi(q)$, which is not a prime. This implies that Lagrange interpolation for calculating the modified shadows will not work (except when $q = 3$, in which case we are not interested). Consider the situation, when $\prod_{j=1,j\neq i}^{t}(u_i - u_j)$ and q are co-prime. In this case, there is no way to find out the multiplicative inverse of $\prod_{j=1,j\neq i}^{t}(u_i - u_j) \bmod q$. There is only possibility of selecting the large prime q numbers in order for each person to get around this difficulty. These signature schemes are meaningless to any third party because there is no way for him to prove its validity. The only knowledge of Z is not sufficient to prove the validity of signature. Signature receiver also has to perform the confirmation protocol in a zero-knowledge fashion to prove the validity of signature [25–32]. No doubt, the communication cost of the proposed scheme is very high, so in future, we should try to reduce its cost without compromising the security of the scheme.

References

1. Diffie, W., Hellman, M.: New directions in cryptography. IEEE Trans. Ingo. Theor. **31**, 644–654 (1976)
2. Desmedt, Y., Frankel, Y.: Shared generation of authenticators and signatures. In: Proceedings of Advances in Cryptology—Crypto 91, pp. 457–469. Springer, New York (1991)
3. Lim, C.H., Lee, P.J.: Security protocol. In: Proceedings of International Workshop, LNCS, vol. 1189. Springer, Cambridge, United Kingdom (1996)
4. Lim, C.H., Lee, P.J.: Modified Maurer-Yacobi, scheme and its applications. In: Advance in Cryptology—Auscrypt, LNCS, vol. 718, pp. 308–323 (1993)
5. Boyar, J., Chaum D., Damgard, I., Pederson, T.: Convertible undeniable signatures. In: Advances in Cryptology—Crypto 90, LNCS, vol. 537, pp. 189–205 (1991)
6. Chaum, D.: Designated confirmer signatures. In: Advances in Cryptology Euro crypt 94, LNCS, vol. 950, pp. 86–91 (1995)
7. Desmedt, Y.: Society and group oriented cryptography. In Proceedings of Advances in Cryptology—Crypto 87, pp. 457–469. Springer, New York (1988)
8. Harn, L.: (*t, n*) Threshold signature scheme and digital multisignature. In: Proceedings of workshop on cryptography and data security, 7–9 June, pp. 61–73. Chung Cheng Institute of Technology, ROC (1993)
9. Desmedt, Y.: Threshold cryptography. In: European Transactions on Telecommunications and Related Technologies. vol. 5, no. 4, pp. 35–43 (1994)

10. Gennaro, R., Jarecki Hkrawczyk, S., Rabin, T.: Robust threshold DSS signature. In: Proceedings of Advances in Cryptology—Euro Crypto 96, pp. 354–371. Springer, Berlin (1996)
11. Schnorr, C.P.: Efficient signature generation by smart cards. J. Cryptol. **4**(3), 161–174 (1994)
12. Shamir, A.: How to share a secret. Commun. ACM **22**, 612–613 (1979)
13. Lal, S., Kumar, M.: Application of directed signature scheme. South East Asian J. Math. Math. Sci. **2**(1), 13–26. Also available at http://arxiv.org/ftp/cs/papers/0409/o409050.pdf
14. Lal, S., Kumar, M.: A directed signature scheme and its applications. In: Proceedings of National Conference on Information Security, New Delhi, 8–9 Jan 2003, pp. 124–132. Also available at http://arXiv.org/ftp/cs/papers/0409/o4090036.pdf (2003)
15. Guillou, L.C., Quisquater, J.J.: A practical zero-knowledge protocol fitted to security microprocessors minimizing both transmission and memory. In: Advances in Cryptology—Eurocrypt 88, LNCS, vol. 330, pp. 123–128 (1988)
16. Umaprasada Rao, B., Vasudeva Reddy, P.: ID-based directed multi-proxy signature scheme from bilinear pairings. Int. J. Comput. Sci. Secur. (IJCSS) **5**(1) (2011)
17. Zhang, J., Yang, Y., Niu, X.: Efficient provable secure ID-based directed signature scheme without random oracle. In: Proceedings of the 6th International Symposium on Neural Networks: Advances in Neural Networks, LNCS, vol. 5553, pp. 318–327. Springer (2009)
18. Vasudeva Reddy, P., Umaprasada Rao, B., Gowri, T.: ID-based directed threshold multisignature scheme from bilinear pairings. Int. J. Comput. Sci. Eng. **2**(1), 74–79 (2009)
19. Sun, X., Li, J., Chen, G., Yung, S.: Identity-based directed signature scheme from bilinear pairings. eprint.iacr.org/2008/305.pdg
20. Lu, R., Lin, X., Cao, Z., Shao, J., Liang, X.: New (t, n) threshold directed signature scheme with provable security. Inf. Sci. **178**(3), 756–765 (2008)
21. Lu, R., Cao, Z.: A directed signature scheme based on RSA assumption. Int. J. Netw. Secur. **2** (3), 182–186 (May 2006)
22. Wang, Y.: Directed signature based on identity. J. Yulin College **15**(5), 1–3 (2005)
23. Blakely, G.R.: Safeguarding cryptographic keys. In: Proceedings of AGIPS 1979 National Computer Congress, vol. 48, pp. 313–317 (1979)
24. Blake, I.F., Van Oorschot, P.C., Vanstone, S.: Complexity issues for public key cryptography. In: Skwirzynski, J.K. (ed) Performance Limits in Communication, Theory and Practice, NATO ASI Series E: Applied Science, vol. 142, pp. 75–97. Kluwer Academic Publishers [Proceedings of the NATO Advanced Study Institute Ciocco, Castelvecchio Pascoli, Tuscany, Italy] (1986)
25. Chaum, D.: Zero- knowledge undeniable signatures. Advances in Cryptology—Eurocrypt 90, LNCS, vol. 473, pp. 458–464 (1991)
26. Mullin, R.C., Blake, I.F., Fuji-Hara, R., Vanstone, S.A.: Computing logarithms in a finite field of characteristic two. SIAM J. Alg. Disc. Math. 276–285 (1985)
27. NIST.: Digital signature standard. GIPS PUB 186 (1994)
28. Okamoto, T.: Designated confirmer signatures and public key encryption are equivalent. In: Advances in Cryptology—Crypto 94, LNCS, vol. 839, pp. 61–74 (1994)
29. Odlyzko. A.M.: Discrete logs in a finite field and their cryptographic significance. In: Cot, N., Beth, T., Ingemarsson, I. (eds.) Advances in Cryptology—Eurocrypt 84, LNCS, vol. 209, pp. 224–314 (1984)
30. Rabin, T.: A simplified approach to threshold and proactive RSA. In: Proceedings of Advances in Cryptology—Crypto, vol. 98, pp. 89–104. Springer, New York (1998)
31. Yen, S.M., Laih, C.S.: New digital signature scheme based on discrete logarithm. Electron. Lett. **29**(12), 1120–1121 (1993)
32. Zheng, Y., Matsummoto, T., Imai, H.: Structural properties of one-way hash functions. In: Proceedings of Advances in Cryptology—Crypto 90, pp. 285–302. Springer (1990)