http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 12 (17): 4063-4067, 2013
ISSN 1812-5638 / DOI: 10.3923/itj.2013.4063.4067

Unidirectional Multiple-times Proxy Re-signature Scheme

[1]Xuan Hong, [1]Jianhua Gao and [2]Zhongmei Wan
Department of computer science and technology, Shanghai Normal University, Shanghai, 200234, People's Republic of China
College of Science, Hehai University, Nanjing 210098, China

Abstract: Proxy re-signature, in which a semi-trusted proxy is given special information to convert the delegatee's signature into the delegator's signature on the same message, is frequently applied to many applications. Since the signing ability of the delegatee may be abused In these practical circumstances, we introduce the multiple-times proxy re-signature scheme, with the help of the binary hash tree. In this study, the proposed scheme restricted the attacker's forgery, restrained the delegatee's abuse and released the overhead of revocation. Furthermore, the scheme had only O(N) additional hashing computation overheads. The size of the proxy signature is also constant. Finally, we support our scheme with detailed security and performance analysis.

Key words: Proxy re-signature scheme, multiple-times, restriction, binary hash tree

INTRODUCTION

The proxy re-signature (Blaze *et al*., 1998) makes a proxy to converts a delegatee's signature into a delegator's signature on the same message. The semi-trusted proxy transforms the signature with some secure information but cannot generate the signature instead of the delegatee or the delegator. Generally speaking, a proxy re-signature scheme should also satisfy other properties (Ateniese and Hohenberger, 2005), Multi-use, key optimal, non-transitive and temporary. In a multi-use scheme, the re-signature can also be transformed, while in a single-use scheme, only the original signature can be transformed. In a key optimal scheme, a user is required to store only a small constant amount of secrets, regardless of how many signature delegations the user gives or accepts. In a non-transitive scheme, the proxy cannot delegate his re-signing right with itself alone. In a temporary scheme, the re-signing right is temporary.

The proxy re-signature scheme proposed by Blaze *et al*. (1998) is bidirectional, multi-use. However, the scheme is not secure. It is possible for everybody to recover the re-sign key that should be stored at the proxy. Ateniese and Hohenberger (2005) proposed another two proxy re-signature schemes. One is bidirectional multi-use and another is unidirectional single-use. The schemes did not proved secure. Later, Shao *et al*. (2007) proposed the first bidirectional proxy re-signature which is existentially unforgeable in the standard model and the first ID-based proxy re-signature scheme. Both the schemes suffered from the relatively large size of public parameters and the considerable computation overheads. However, Kim *et al*. (2009) discussed about Shao *et al*. (2007) scheme, by presenting an attack and making improvements. Furthermore, Benoit and Damien (2008) proposed the multi-use unidirectional proxy re-signature schemes based on bilinear groups in random oracle model and also in standard model. Sunitha and Amberker (2008) proposed another unidirectional proxy re-signature scheme with forward-secure. Sherman and Raphael (2008) showed how to design a generic unidirectional proxy re-signature scheme and how to incorporate the concept of forward-security into proxy re-signature.

Proxy re-signature can be very useful to simplify certificate management by constructing a secure channel between two CAs, to simplify group signature management by making each signature transformed to the group's signature, to help relieve some common key management headaches by resigning without the CA and to construct Digital Right Management (DRM) interoperable system by sharing the certification.

We consider the scenario, that when the manager B is on holiday, B delegates the signing ability to the secretary A. That is, A can sign the files on behalf of B. In this application, the signing ability of the secretary B may be abused. Once B is corrupted and signs hundreds of such signature, it will cause great damage to A. To solve

Corresponding Author: Xuan Hong, Department of computer science and technology, Shanghai Normal University, Shanghai, 200234, People's Republic of China

the above problem, we introduce the multiple-times proxy re-signature, where no more than some number re-signature can be can be securely generated.

The multiple-times signature scheme is another special research area, which restricts the signer's signing capability. Hwang *et al.* (2003) present a multiple-times digital signature scheme by using the C degree polynomial f(z) to restrict the signing capability. In this scheme, if the signer generates signatures more the threshold value C, then anyone can calculate the signer's secret key. Choi *et al.* (2003) present a variation of Schnorr signature scheme with restricted signing capability. But the scheme had many limitations, the delegation process is insecure against the original signer's forgery and it is only useful in the case of Schnorr-based digital signature scheme.

In basic proxy re-signature schemes, the delegatee can sign large number of signatures beyond the expectancy of the original signer. So in some applications the delegator may want to control the signing ability of the delegate. The function can be easily realized by the multiple-times proxy re-signature schemes. The proposed scheme can fully satisfy all the requirements and the overhead is just O(N) hashing function which is easy and fast. Meanwhile, the colluding delegatee and proxy cannot forge the delegator's original signature and vice versa. We protect the signing key of the delegator's and the delegatee's.

The proposed scheme enjoys the following properties: It is unforgeable in the random oracle model, assuming the CDH assumption is hard to solve; the size of signing key and the proxy re-signature is constant and independent of signing executions number; it would also solve the overhead of the revocation. After a required number of re-signing executions, the re-signing key will be out of date, so there is no need to declare that the delegatee is invalid and no additional computation for delegator.

PROPOSED SCHEME

Binary hash tree: We define the binary hash tree which can authenticate a series of numbers $L_i(0\leq_i<N)$ where $N = 2^D$ or may be $2^{D-1}{}_{<N<}2^D$. Generally speaking, The binary hash tree is a binary tree with depth $_D$. We let $L_i(0\leq_i<N)$ be the leaves and construct hash tree from the bottom, $N_{i,j}$ is the node of the tree:

- set $N_{i,j} = 0$, for $0\leq j\leq D$, $0\leq i<2^j$,
- set $N_{D,i} = L_i(0\leq_i<N)$
- $N_{j,i} = \begin{cases} h(N_{j+1,2i}, N_{j+1,2i+1}) & \text{if } N_{j+1,2i+1} \neq 0 \\ N_{j+1,2i} & \text{if } N_{j+1,2i+1} = 0 \end{cases}$

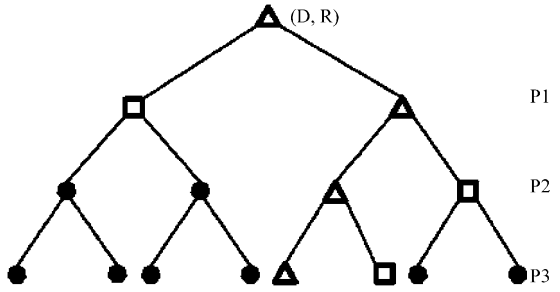

Fig. 1: Example of an I-labeled sibling path for D = 3, i = 4, which is depicted in □, the path of the leaf 4 is depicted in Δ

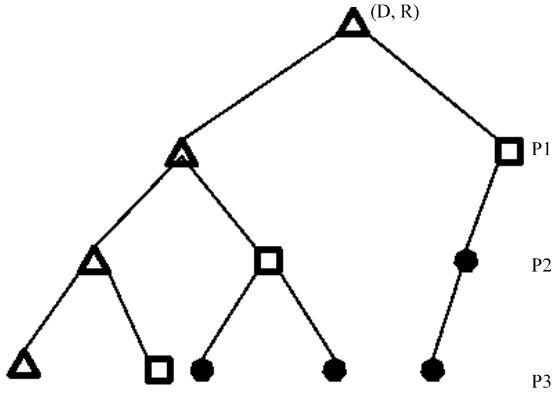

Fig. 2: Example of an i-labeled sibling path for N = 5, D = 3, i = 0, which is depicted in □, the path of the leaf 0 is depicted in Δ

where, $0\leq j< D$, $0\leq_i<2^j$.

Let $R = L_{0,0}$. For each leaf $L = N_{D,i}$ we has i-labeled sibling path $(P_N,\ldots, P_1)$, where P_i is the sibling of the node of the path from the root to the leaf L_i. If there is no sibling nodes, $P_i = 0$.

Then, the verification value are (D, R) and for each L_i, there is a authentication (Li, $P_D,\ldots,P_1$). Verifier accept L_i unless $W_0 = R$, where $W_D = L_I$ and:

$$W_{i-1} = \begin{cases} h(W_i \| P_i) & P_i \neq 0 \\ W_i & P_i = 0 \end{cases}$$

Figure 1-3 describe the three leaves and their verification values.

Proposed scheme: Present scheme requires a bilinear map e: $G_1HG_1\rightarrow G_2$, operates over two groups G_1, G_2 of prime order p = Å(2^k). The global parameters are (e,p., G_1, G_2, g, h, H), where g and h are generators of G_1 and H is a hash

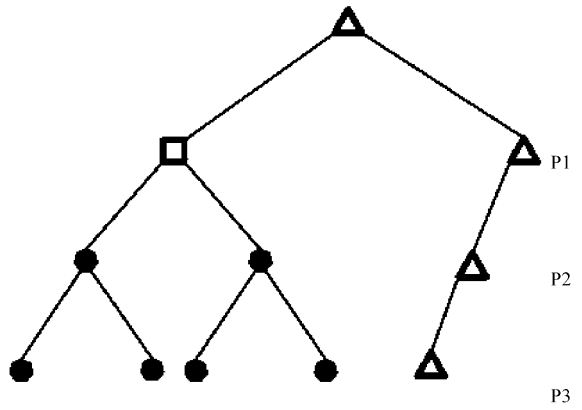

Fig. 3: Example of an i-labeled sibling path for N = 5, D = 3, i = 4, which is depicted in □, the path of the leaf 0 is depicted in Δ, so $P_2 = 0$, $P_3 = 0$

function from arbitrary strings to elements in Z_q. Scheme PRS = (KeyGen, ReKey, Sign, ReSign, Verify) is describedas follows:

KeyGen: On input the security parameter 1^k, select a random $a \in Z^*_p$ and output the key pair $pk = g^a$ and $sk = a$.

ReKey: The delegator B, the delegate A and the semi-trusted proxy execute the following steps to obtain the re-sign key rk_{A_B}.

- The delegator B selects the restriction number N which is the times of the delegatee A can sign on behalf of him. Then he sends N to the delegate and his secret key sk_B to the proxy
- The delegatee A selects N random numbers $l_i \in Z^*_p (1 \le i \le N)$, which will be used as the temporary parameters and computers

$$L_i = (g^{li} \bmod p) \bmod q (1 \le i \le N)$$

It uses these L_i as the leaves to construct a binary hash tree and obtain the verification value (D, Root), where $D = j\log_2 Nk$, Root is the root of the binary hash tree. And for every L_i there is the corresponding authentication $(L_i, P_D, \ldots P_1)$, which is the i-sibling path. Then the delegatee A put L_i and the corresponding authentication $(L_i, P_D, \ldots P_1)$ at a public dictionary and sends the parameters (D, Root) and his secret key sk_A to the proxy.

The semi-trusted proxy computes the re-sign key $rk_{A\_B} = h^{b/a} \bmod p$.

Sign: If the delegatee A wants to sign the message m on behalf of the delegator B, the delegate A chooses a unused parameter l_i and L_i, set $K = h^{l/a}$, then the delegatee A computes $\sigma = a\text{-}H(m||L||K)+l_i \bmod p\text{-}1$ and output $s = (\sigma, L, K)$. We call this form the original signature.

If the delegatee A uses the random temporary parameter l_i twice, then his secret key sk_A is revealed as follows:

$$\sigma = lG^1(H(m||L||K)+l_i\, sk_A.L) \bmod p\text{-}1$$

$$\sigma' = lG^1(H(m'||L||K)+l_i\, sk_A.L) \bmod p\text{-}1$$

By these equations, we can compute the temporary parameter:

$$1 = \frac{H(m)-H(m')}{\sigma-\sigma'} \bmod p-1$$

Then the secret key:

$$sk_A = \frac{\sigma l - h(m)}{L} \bmod p-1$$

The delegatee A can use l_i for only one time to generate signature without revealing of his secret key. The delegator B controls the number of the proxy re-signature in this way.

ReSign: On input the original signature $s = (\sigma, L, K)$ and message m, the semi-trusted proxy first confirms the validity of the temporary parameter L. If there is $W_0 = R$, where $W_D = L$ and:

$$W_{i-1} = \begin{cases} h(W_i \,||\, P_i) & P_i \neq 0 \\ W_i & P_i = 0 \end{cases}$$

For $(0<i\le D)$, where $l_t - W_t$, $r_t = P_t$.

Then the proxy check Verify $(pk_A, m, s) = 1$. If these verification equations hold simultaneously, the semi-trusted proxy chooses $r \in Z^*_p$, computes $R = (rk_{A-B})^r$ and $\sigma' = (rk_{A-B})^{\sigma+r}$, outputs $s' = (\sigma', L, K, R)$. We call this form the re-signature.

Since $\sigma = a.H(m||L||K)+l_i$, the re-signature $\sigma' = (rk_{A-B})^{\sigma}$, $h^{b.H(m||L||K)}.J^b.R^b$.

Verify: On input public key pk, message m and the purported re-signature s, set $\omega = H(m||R||K)$. Since $s = (\sigma, L, K, R)$ is the re-signature, $\sigma = h^{b.w}.K^b.R^b$. That is to say, we check whether $e(\sigma, g) = e(h, pk_B)$. $e(R, pk_B)$, it is a

correct re-signature if the equation holds, then Verify outputs 1, otherwise output 0 for instead.

SECURITY ANALYSIS OF THE PROPOSED SCHEME

In this section, we will demonstrate that the proposed scheme is secure and can work correctly.

Theorem 3.1: The proposed scheme is verifiable, if the delegator, the delegate and the semi-trusted proxy follow the issuing protocol.

Proof: If all the delegator B, the delegate A and the proxy follow the issuing protocol, then obviously, the proxy re-signature can be verified. Hence, the proposed scheme satisfies verifiability.

Theorem 3.2: If Computational Diffie-Hellman (CDH) assumption holds in G_1, then the proposed unidirectional single-use proxy re-signature is correct and existentially unforgeable in the random oracle model.

Proof: If there exists an adversary Adv that can break the above proxy re-signature scheme with non-negligible probability å in time t after making at most q_S sign queries, q_{RS} resign queries, q_K corrupted key queries, q_{RK} rekey queries and q_H hash queries, then there also exists an simulator adversary Sim that can solve the mCDH problem in G_1 with probability $1/\sqrt{q_H}$ in $t+O(t(k)+q+H+\tau$.

On input (g, g^a, g^b, h, h^a), the CDH adversary Sim's goal is to compute g^{ab}. Sim sets up the global parameters for Adv: the security parameter $k = |p|$, the groups G_1, G_2, their prime order p and the mapping e. The system parameters are (e, p, G_1, G_2, g, h, H). Sim builds the following oracles:

- $\mathbf{O_{Hash}}$**:** On input (m,L,K), Sim checks if (m,R,K) is recorded in database D_H. If not, selects random $\omega \in Z_P$ and record (m, L, K, ω) sim outputs ω
- $\mathbf{O_{CKeyGen}}$**:** Sim chooses random $x_i \in Z_P$ and outputs $(pk_i, sk^i) = (g^{xi}, x_i)$
- $\mathbf{O_{ReKey}}$**:** On input (pk_i, pk_j), Sim returns $rk_{i?j} = h^{j/I} = (h^j)^{1/i}$. if pk_i and pk_j are both corrupted, or pk_i is uncorrupted and pk_j is corrupted, Sim returns $rk_{i?j} = h^{j/I} = (h^j)^{1/I}$; else, input is illegal
- $\mathbf{O_{Sign}}$**:** On input (pk, m), if pk is corrupted, Sim returns the signature $\sigma = (\delta, L, K)$, $\delta = a.H(m||L||K)+l$. Otherwise, Sim randomly selects u, v. Computes $R = g^u pk^v \bmod p$, $\delta = u$ and $K = h^{r/a} = g^b$. The challenger records $\omega = H(m||L||K) = -v$ to the D_H as the hash response to (m, L, K). $\sigma = (\delta, L, K)$ has the correct signature as in the actual scheme.
- $\mathbf{O_{reSign}}$**:** On input (pk_i, pk_j, m, σ), if Verify $(pk_i, m, \sigma) = 1$, Sim invokes the algorithm ReSign $(O_{ReKey}(pk_i, pk_j), pk_i, m, \sigma)$ and outputs the result; otherwise, outputs 0

Forgery: If Sim does not abort as a consequence of one of the queries above, Adv will, with probability at least å, return a valid σ^* on m^*. If the forgery is the original signature, we have the conclusion that the triplet ElGamal-family signature which is provably unforgeable under ROM following the forking lemma. If the forgery is the re-signature, the forgery must be of the form $\delta^* = (h^\omega.K)^a = (ha)^\omega \cdot g^{ab}$. To solve the CDH instance, Sim outputs $g^{ab} = \delta^*.(ha)^{-\omega}$.

To conclude, we analyze the probability that Sim completes the simulation without aborting. The probability of Sim is Pr [B] = Pr[E1∨E2], where event E1 denotes forge the original signature, event E2 denotes forge the re-signature. $Pr[E1] = 1/\sqrt{q_H}$, Pr [E2] = $1/q_H$, hence $Pr[B] \geq 1/\sqrt{q_H}$. The time complexity of B is $t+(t(k)+qH.\tau)$.

Thus, the theorem follows.

Theorem 3.3: Let H(.): $\{0, 1\}^* \rightarrow \{0, 1\}^s$ be a hash function, for $0 \leq I < 2^D$. If PA = $(P_D, \ldots, P_1)$ is an i-labeled sibling path of depth D for a leaf L and PB = $(P'_D, \ldots, P'_1)$ is an i-labeled sibling path of depth D for a leaf L', both of them have the same root Root, then either PA≠PB or L≠L' implies an explicit collision.

Proof: We define W_D: = L and W_{I-1}: = $H(W_i||P_i)$. In a analogous way, W'_i for $0 < i \leq D$.

If PA≠PB. As both of them have the same root $R = W_0 = W'_0$, we know that there must be some i, where $W_{I-1} = W'_{i-1}$ but $W_i \neq W'_i$. Set k: = $\min_{1 \leq i \leq D}\{W_i \neq W'_i\}$ and $H(W_k||P_k) = W_{k-1} = W'_{k-1} = (W'_k||P'_k)$. No matter whether $P_k = P'_k$ or not, it is a collision for hash function H(.)

If PA = PB and L≠L', PA = PB implies that $W_{D-1} = W'_{D-1}$ (because of $W_0 = W'_0$ and $P_i = P'_i$). So $H(W_D||P_D) = W_{D-1} = W'_{D-1} = H(W'_D||P'_D)$. So it is also a collision for H(.).

Corollary 3.4: Proposed multiple-times proxy re-signature scheme is secure and can work correctly.

PERFORMANCE OF THE PROPOSED SCHEME

This scheme is unidirectional, single-use and non-transitive. The proxy transforms the delegatee A's signature to the delegator B's signature.

Table 1: Comparison of different schemes

Property	BBS[1]	S_{uni}	Our scheme
Unidirectional	No	Yes	Yes
Single-use	No	Yes	Yes
Private Proxy	No	No	No
Transparent	Yes	Yes	Yes
Unlinkable	No	No	Yes
Key Optimal	Yes	Yes	Yes
Non-interactive	No	Yes	Yes
Non-transitive	No	Yes	Yes
Multi-times	No	No	Yes

The length of the signing key of the resulting multiple-times scheme is constant and independent of the number of signing executions. Since the resulting scheme only needs extra O(N) hashing operations in ReKey phase and extra $O(\log_2 N)$ hashing operations in Verify phase, it is efficient and practical. We compute the length of our multiple-times proxy re-signature. Let $Length_O$ be the size of an original proxy signature, $Length_H$ be the size of the hash value. Let $Length_M$ be the maximal size of a multiple-times proxy signature. We get. But if we put the leaf's verification signature into a public dictionary, we don't need to endure the expansion of the signature.

Now, we consider the computation. Especially, we use additional hash binary tree, whose elements are hash value of the pre-selected random values. Compared with the previous schemes, it is simple and efficient. In the signature algorithm and re-signature algorithms, the parties only need modular and multiplicative operations within the group G_1. The scheme does the time consuming paring operation only when verifying the re-signature. Unlike Kim et al.'s scheme, this has relatively large size of public parameters. The security of our scheme can be reduced to Computational Diffie-Hellman (CDH) assumption in the random oracle model. Furthermore, since each user just stores one signing key, the scheme is also key optimal.

This scheme is unidirectional, single-use and non-transitive. The proxy transforms the delegatee A's signature to the delegator B's signature.

CONCLUSION

In this study, we proposed a secure and efficient multiple-time proxy re-signature scheme to solve the delegatee's abuse. The proposed scheme is unidirectional, single-use, key optimal and non-transitive. Cooperated with binary hash tree, the proposed proxy re-signature is simpler and the security can be reduced to computational Diffie-Hellman assumption in the random oracle model. Moreover, compared to some other schemes, our scheme implement the restriction in less time with shorter length of signature.

ACKNOWLEDGMENT

The authors would like to thank the support of National Natural Science Foundation of China (NSFC) under Grant No. 61003215. We also thank the support of Innovation Program of Shanghai Municipal Education Commission No.12YZ072.

REFERENCES

Ateniese, G. and S. Hohenberger, 2005. Proxy re-signatures: New definitions, algorithms and applications. Proceeding of the 12th ACM Conference on Computer and Communication Security, November 7-11, 2005, Alexandria, VA., USA., pp: 310-319.

Blaze, M., G. Bleumer and M. Strauss, 1998. Divertible protocols and atomic proxy cryptography. Proceeding of the International Conference on the Theory and Application of Cryptographic Techniques, May 31-June 4, 1998, Finland, pp: 127-177.

Choi, C., Z. Kim and K. Kim, 2003. Schnorr signature scheme with restricted signing capability and its application. Proceeding of the 2nd International Conference on Signals Circuits and Systems, December 14-17, 2003, United Arab Emirates.

Sherman, C. and P. Raphael, 2008. Proxy re-signatures in the standard model. Proceedings of the 11th Information Security Conference, Sep. 15-19, China, pp: 260-276.

Hwang, J.Y., H.J. Kim, D.H. Lee and J. Lim, 2003. Digital signature schemes with restriction on signing capability. Proceeding of the 8th Australasian Conference of Information Security and Privacy, July 9-11, 2003, Australia, pp: 324-335.

Kim, K., I. Yie and S. Lim, 2009. Remark on Shao *et al* bidirectional proxy re-signature scheme in indocrypt'07. Int. J. Network Security, 9: 8-11.

Benoit, L. and V. Damien, 2008. Multi-use unidirectional proxy re-signatures. Proceedings of the 15th Conference on Computer and Communications Security, October 27-31, 2008, USA., pp: 511-520.

Shao, J., Z.F. Chao, L. Wang and X. Liang, 2007. Proxy re-signature schemes without random oracles. Proceeding of the 8th International Conference on Cryptology in India, December 9-13, 2007, Chennai, India, pp: 197-209.

Sunitha, N.R. and B.B. Amberker, 2008. Proxy re-signature schemes. Proceeding of the 4th Interna-tional Conference on Information Science and Security, December 16-20, 2008, Hyderabad, India, pp: 156-157.