A One-Way Variable Threshold Proxy Re-signature Scheme for Mobile Internet

Yanfang Lei[1], Mingsheng Hu[1(✉)], Bei Gong[2], Lipeng Wang[1], and Yage Cheng[1]

[1] College of Information Science and Technology, Zhengzhou Normal University,
Zhengzhou 450044, China
hero_jack@163.com

[2] College of Computer Sciences, Beijing University of Technology,
Beijing 100124, China

Abstract. In recent years, the mobile Internet has been rapidly developed and widely used. Aiming at the problems of the weak computing power of mobile internet mobile terminal equipment, limited energy supply and high security requirements due to the complexity of mobile Internet environment, we proposes a secure and efficient server-assisted verification threshold proxy re-signature scheme, and the correctness of the program is verified. The proposed scheme includes a threshold proxy re-signature algorithm and a server-assisted authentication protocol scheme. Threshold proxy re-signature is a technique of proxy re-signature using threshold, which can decentralize the proxy's signature rights. In the scheme, the verifier and the server send the complex signature verification operation to a semi-trusted server through the protocol, which effectively reduces the computational load of the verifier. The security analysis results show that the new scheme is safe and it is proved that the scheme is safe under collusion attack and adaptive selection message attack under the standard model. The performance analysis results show that the new scheme proposed in this paper has shorter signature length, less computational cost, higher verification efficiency and better adaptability to the mobile Internet environment.

Keywords: Threshold proxy re-signature · Server-assisted authentication protocol · Secret sharing · Unforgeability · Robustness · Completeness

1 Introduction

With the advent of the information age and the rapid development of the information technology, the Internet has penetrated into all aspects of our daily

Supported by the National Natural Science Foundation of China (Grant No. U1304614, U1204703), Henan Province Education Science Plan General Topic "Research on Trusted Degree Certification Based on Block-chain" (Grant No. (2018)-JKGHYB-0279), Zhengzhou Innovative Science and Technology Talent Team Construction Project Fund Project (Grant No. 131PCXTD597), Henan Science and Technology Project (Grant No. 162102310238).

J. Li et al. (Eds.): SPNCE 2019, LNICST 284, pp. 521–537, 2019.
https://doi.org/10.1007/978-3-030-21373-2_42

lives. The development of mobile communication technology is changing with each passing day. Mobile terminals such as Ipad, smart phones, wireless sensors, and electronic keys have become an indispensable part of our lives and work. The further development of network technology has brought more convenience to our lives. The rise of e-commerce and e-government has brought people from the real material world into a convenient electronic age. Online shopping, stock operations, communication and access to network resources can be carried out anytime, anywhere via the Internet. However, due to the limitations of the mobile Internet terminal device itself, the computing power is generally weak, which makes it necessary for people to perform a large amount of time for verification in resource request and resource access. In addition, because the mobile Internet environment is more complicated, there are higher requirements for security. Therefore, it is necessary to design a scheme that can solve terminal computing power, limited energy supply and high security to be applied in the mobile Internet environment.

Quisquater et al. first proposed a server-assisted verification signature scheme in 2000, which included a standard signature system and a server-assisted verification protocol. The scheme assigned complex signature verification operations to a semi-trusted server for execution. It effectively reduced the amount of certifier calculations and adapts to low-end electronic devices such as smartphones, electronic keys, and wireless sensors. In 2005, Girault et al. presented a formal security model for server-assisted computing at the Asia-Pacific conference. In 2008, Girault et al. proposed a security definition for server-assisted verification signatures in [1]. In 2011, a provable secure server-assisted verification signature scheme was presented in [2], but this scheme could not effectively resist the collusion attack between the server and the signer. Later, combined with the aggregate signature and server-assisted verification signatures, Wu et al. [3] proposed a cryptosystem for server-assisted authentication of aggregated signatures. This system consisted of an aggregate signature system and a server-assisted authentication protocol. It aggregated the different signatures of multiple messages into one signature, which greatly saved broadband expenditure, saved verification time and improved verification efficiency.

Proxy re-signature is an important research direction in cryptography, and scholars who at home and abroad have devoted a lot of work to this direction. In [4], the security model of proxy re-signature was first proposed and given two schemes, and a strict security proof was given based on the random prediction model. In [5], the further definition of the security attribute definition of proxy re-signature was given, and a secure proxy re-signature scheme under a standard model was constructed based on the literature [6]. The defects of this scheme were presented and an improvement method was given in [7]. In [8], the author proposed a one-way multi-purpose proxy re-signature scheme, but the signature verification overhead increased linearly with the increase of the number of re-signature layers. In recent years, proxy re-signature has attracted much attention, and some proxy re-signature schemes with special properties have been proposed, such as identity-based proxy re-signature [9], lattice-based proxy re-signature [10,11]. Although the proxy re-signature scheme has a wide range of applications, it also has some defects. For example, once the re-signature key is compromised, the security of the solution will be compromised. In addition, the agent's rights are too concentrated and need to be decentralized in order to make the solution more reliable. Then there is the concept of threshold proxy re-signature. Threshold proxy re-signature is a technique for thresholding the proxy re-signature, and dispersing the signature right of the proxy. Between two signers, the original only one agent is set as n semi-trusted agents, and the re-signature key is distributed to each agent by secret sharing technology. The re-signature can only be synthesized when at least t agents send their valid verifiable signatures. Threshold proxy re-signature schemes could be used to reduce public key management overhead, space-saving specific path traversal certificates, and generate manageable weak group signatures [12–14].

With the rapid development of cloud computing and big data, cloud computing providers with powerful computing capabilities have become agents in proxy re-signing schemes, and low-end computing devices are very important cloud computing terminals [15]. These devices have weak computing power, limited energy supply and short response times. However, most of the current proxy re-signature signature verification algorithms require complex bilinear-parings operations and cannot be adapted to low-end computing devices with weak computing power. In addition, we consider that in theory, the trustee Alice should not bear any security risks, and the risks faced should be borne by the trustee Bob. In order to solve above problems, this paper combines the security attributes of server-assisted verification signature and threshold proxy re-signature, and proposes a formal model of one-way server-assisted verification variable threshold proxy re-signature, and constructs a one-way server-assisted verification variable threshold proxy re-signing scheme that can effectively resist collusion attacks. However, the one-way scheme can achieve two-way function through the conversion between the trustee Alice and the delegator Bob. The verifier completes the verification of the signature with a small computational cost by executing a server authentication protocol with the server, which improves the verification efficiency of the signature. The verification algorithm reduces complex bilinear-parings operations and has lower computational time overhead, so it can be better adapted to the mobile Internet environment.

2 Prerequisite Knowledge

2.1 Bilinear Pairings

Bilinear pairings function is a bilinear map between two cyclic groups. As all the points in elliptic curve gathered into a group, bilinear parings function applies just in elliptic curve. Let G_1 and G_2 be two cyclic groups of the prime order q, where q is a large prime number. Let g is a generator of G_1. Assume that the discrete logarithm problems in both G_1 and G_2 are hard to solve. Let $e : G_1 \times G_1 \longrightarrow G_2$ is a bilinear map that satisfies the following [16,17] properties:

a. Bilinear: For any $\tau_1, \tau_2 \in Z$, satisfy $e(\tau_1 g_1, \tau_2 g_2) = e(g_1, g_2)^{\tau_1 \tau_2}$ for all $g_1, g_2 \in G_1$.
b. Non-degenerate: There is $g_1, g_2 \in G_1$, such that $e(g_1, g_2) \neq 1$.
c. Computable: There is an efficient algorithm to compute $e(g_1, g_2)$, for any $g_1, g_2 \in G_1$.

2.2 Computational Diffie-Hellman (CDH) Problem and Corresponding Hypothesis

Based on the group, we can define the following hard cryptographic problem:

Definition 1 CDH problem [18]**:** Give a triple $(g, \tau_1 g, \tau_2 g) \in G_1^3$ for $\tau_1, \tau_2 \in Z_q^*$, find the elements $\tau_1 \tau_2 g$.

In order to obtain the following safety analysis results, we introduce the following hypothesis.

Definition 2 CDH Hypothesis [18]**:** There is no algorithm that can solve the CDH problem in group G_1 with a non-negligible probability in polynomial time.

2.3 Secret Sharing Model

Distribution Stage: Let q be a prime number and secret $s \in Z_q^*$ to be distributed. Suppose there is a threshold of (t, n): in a group with n members $P_i (i = 1, 2, ..., n)$, the secret s can be recovered when at least t members cooperate. The basic idea is: first randomly generate $\alpha_1, \alpha_2, ..., \alpha_{t-1}$ and generate the function $F(x) = s + \alpha_1 x + \alpha_2 x^2 + ... + \alpha_{t-1} x^{t-1}$, then calculate $X_i = F(i) \in Z_Q^*$ and issue (i, X_i) to each member P_i, note that we can get $X_0 = F(0) = s$ when $i = 0$.

Reconstruction Stage: Let $\Phi \subseteq 1, 2, ..., n$ and $|\Phi| \geq t$, where $|.|$ represents the order of the set Φ. Then, the function $F(x) = \sum_{j \in \Phi} \chi_{x_j}^{\Phi} x_j$ where parameter $\chi_{x_j}^{\Phi} = \prod_{k \in \Phi, k \neq j} \frac{x-k}{j-k}$ and note that $\chi_{x_j}^{\Phi} \in Z_q^*$ is the Lagrange interpolation polynomial coefficient [19]. Finally, we can recover the secret $s = F(0) = \sum_{j \in \Phi} \chi_{0_j}^{\Phi} x_j$ where $\chi_{0_j}^{\Phi} x_j = \prod_{k \in \Phi, k \neq j} \frac{0-k}{j-k}$.

3 Scenario Model and Security Definition

3.1 Model of Server Auxiliary Threshold Proxy Re-signature Scheme

The threshold proxy re-signature scheme is a tuple in a polynomial time (Setup, Keygen, Rekey, Sign, Resign, Verify, Server-setup, Sever-verify) contains the following eight algorithms.

Setup$(1^k) \rightarrow cp$**:** Given a constant k, and runs the algorithm to generate a public parameter cp.

Keygen$(cp) \rightarrow (pk, sk)$**:** Given the system parameter cp, we obtain the signer's public and private keys (pk, sk).

Rekey(pk_A, sk_A, pk_B, sk_B)**:** After inputting (pk_A, sk_A, pk_B, sk_B), generate a re-signature key $r_{A \rightarrow B}$. The re-signature key generation algorithm Rekey distributes the re-signature key $r_{A \rightarrow B}$ into n shares, and secretly distributes the n sub-keys $rk^i_{A \rightarrow B}$ to agents n for storage. The agent P_i can generate a re-signature sub-key $rk^i_{A \rightarrow B}$ and verifiable its public key $\nu k_{t,i}$ according to the provided threshold value t. It should be noted that the key sk_A is unnecessary here.

Sign (m, sk_A)**:** Given a message m and the private key sk_A of the trustee Alice, an original signature $\sigma_{A(m)}$ of the message m corresponding to the public key pk_A is generated.

Resign: The Combiner collects partial re-signatures from each agent and confirms that when at least t are legal, the Combiner synthesizes them into a re-signature $\sigma_{B(m)}$ and outputs it.

Verify: Given the public key pk, the message m and the signature σ to be verified, if σ is the valid signature of the message m corresponding to the public key pk, output 1, otherwise, output 0.

Server-setup(cp)**:** Given the parameter cp, a string Vst is generated for the verifier.

Server-verify(Vst, m, pk, σ)**:** For the string Vst, the public key pk and the message signature pair (m, σ), if the server lets the verifier confirm that σ is a valid signature, output 1, otherwise, output 0.

Because the signer's computing power is limited, complex cryptographic operations cannot be performed. Therefore, a large computing task can be transferred to the server through an interaction protocol with the server, and the signature verification can be completed by using the server.

3.2 Security Definition

The security of the server-assisted verification threshold proxy re-signature includes at least the unforgeability and robustness of the threshold proxy re-signature and the completeness of the server-assisted authentication protocol $Server - verify$. Robustness and unforgeability means that even if an attacker colludes with $t - 1$ agents, the signature scheme will still work correctly, but the attacker cannot resign. This ensures that a legal signature of a new message cannot be generated in the case of a joint attack. The completeness of the server-assisted authentication protocol $Server - verify$ ensures that the server cannot convince the verifier that an illegal signature is legitimate.

Next, based on the definition of completeness given in document [20], this paper defines the completeness of server-assisted authentication protocol under joint attack and adaptive selection message attack by designing game rules between challenger and attacker.

In the Game1 below, since the attacker and the trustee Alice can be colluded, the attacker can obtain the trustee's private key and can generate the original signature of any message. By conducting a limited server-assisted verification challenge with the challenger, the attacker's goal is to convince the verifier that an illegal signature is legitimate. Also in Game2, since the server is allowed to collude with the agent, the attacker owns the agent's re-signature key and can convert the original signature of any message to the re-signature of the corresponding message. After a limited number of server-assisted verification queries with the challenger, the attacker's goal is to convince the verifier that an illegal re-signature is legitimate. The verifier's string Vst is secret to the attacker. If the attacker wins in Game1, the verifier cannot judge whether the original signature provided by the attacker is legal, indicating that the signature scheme cannot resist the joint attack between the server and the trustee Alice. Similarly, if the attacker wins in Game2, it means that the verifier cannot judge whether the re-signature provided by the attacker is legal, indicating that the signature scheme cannot resist the collusion attack of the server and the agent. Therefore, if the attacker wins in any of the games, the signature scheme is not complete. The specific implementation of the two games is as follows:

Game1: In this game, attacker A_1 has the private key pair (pk_A, sk_A) of the trustee Alice, which can represent the original signature of any message by the trustee Alice.

Establishment: Challenger C first runs $Setup$, $Keygen$ and $Server-setup$ three algorithms to obtain the system parameter cp, trustee Alice's public-private key pair (pk_A, sk_A) and string Vst, then cp and (pk_A, sk_A) are sent to attacker A_1.

Query: The attacker A_1 can adaptively perform q_c verification queries with the aid of the server. For each query(m_i, σ_i), Challenger C acts as the verifier, attacker A_1 acts as the server, A_1 and C first perform the server-assisted authentication protocol, and then return the output as a response to A_1.

Output: The attacker A_1 outputs a message m^* and a string σ^*. Let Γ_{m^*} is the set of all legal signatures of m^*, and σ^* is not in Γ_{m^*}. If $Verify(m^*, pk_A, \sigma^*) = 0$ is satisfied and $Server - verify(Vst, m^*, pk_A, \sigma^*) = 1$, it indicates that the attacker A_1 convinces the challenger C that σ^* is m^* corresponds to the legal signature of the public key pk_A, and the attacker A_1 wins in the game.

Game2: In this game, the attacker A_2 has the re-signature key $rk_{A \to B}$ between the trustee and the delegator, which can convert the original signature of the message to the re-signature corresponding to the same message on behalf of the agent.

Establishment: Challenger C runs $Setup, Keygen, Rekey$ and $Server - setup$ four algorithms to obtain the system parameter cp, public and private key pairs (pk_A, sk_A) and (pk_B, sk_B) of Alice and Bob, the re-signature key $rk_{A \to B}$, and the character string Vst, and send cp, pk_A, pk_B and $rk_{A \to B}$ to the attacker A_2.

Query: The attacker A_2 can adaptively perform the q_c server-assisted authentication query, which is the same as the response method in Game1.

Output: The attacker A_2 finally outputs a message m^* and a string σ^*. Let Γ_{m^*} is a set of all legal signatures corresponding to the public key pk_B of m^*, and σ^* is not in Γ_{m^*}. If $Verify(m^*, pk_B, \sigma^*) = 0$ and $Server-verify(Vst, m^*, pk_B, \sigma^*) = 1$, the attacker A_2 lets the challenger C be sure that σ^* is m^* legal signature, and attacker A_2 wins in this game.

The probability that an attacker wins the above game depends entirely on the probability of the coin toss of the challenger and the attacker.

Next, we give two definitions about security.

Definition 3: If the probability of an attacker winning in the above two games is negligible, then the server-assisted authentication protocol in the server-assisted verification proxy re-signing scheme is complete.

Definition 4: If a threshold proxy re-signature scheme has both unforgeability and robustness under adaptive selection message attack, and the server-assisted verification protocol is complete, the corresponding server-assisted verification threshold proxy re-signature scheme is said to be secure under collusion attack and selection message attack.

4 A New One-Way Server Auxiliary Verification Threshold Proxy Re-signature Scheme

In this part, we construct a one-way server-assisted verification variable threshold proxy re-signature scheme that is both secure and efficient and adapts to the mobile Internet environment. The participating entities of the new scheme include the delegator Bob, the trustee Alice, the verifier, the n semi-trusted agents and the server, where the trustee is responsible for generating the original signature of the message, and the semi-trusted agents convert the original signature into the re-signature of the principal, and the verifier completes the valid verification of the signature under the protocol of the semi-trusted server. The specific scheme is as follows:

Setup: Let q be a prime number of length k, G_1 and G_2 are two cyclic multiplication groups of order q, let g be the generator of group G_1, $e : G_1 \times G_1 \rightarrow G_2$ is a bilinear pairings, $H()$ is a public and anti-collision one-way hash function $H : 0, 1^* \rightarrow G_1$. Arbitrarily pick n positive integers $q_1 < q_2 < ... < q_{n-1}$ satisfying the condition $gcd(q_i, q_j) = 1$ and $gcd(q_i, q) = 1$, where $0 \leq i < j \leq n-1$, and let $F = q_0 q_1 ... q_{n-1}$, public system parameters $(cp) = (e, q, G_1, G_2, g, h, H, F, q_0, ..., q_{n-1})$.

Keygen: After entering the security parameter $cp = 1^k$, pick a random number x from Z_q and output the public key $pk = (pk^1, pk^2) = (g^x, h^{\frac{1}{x}})$ and private key $sk = x$.

Rekey: Given the public key of the trustee Alice $pk_A = (pk_A^1, pk_A^2) = (g^a, h^{\frac{1}{a}})$ and the private key of the principal Bob $sk_B = b$, then do the following:

a. Find two random numbers l_i, m_i arbitrarily in $[1, q-1]$ and calculate $\alpha_i = l_i m_i \prod_{j=0}^{i-1} q_j modF$, $i = 0, 1, ..., n-1$. From the Chinese remainder theorem, $\alpha_0 \in Z_F$ can be obtained such that $\alpha_0 = sk_B = bmodq_i, i = 0, 1, ..., n-1$. Then construct a $n-1$ degree polynomial $f(x) = \alpha_0 + \sum_{i=0}^{n-1} \alpha_i x^i$. Given a positive integer $t(1 \leq t \leq n)$, there exists a $t-1$ degree polynomial $f_t(x) = f(x)modq_{t-1} = b + \sum_{i=1}^{t-1} \alpha_i x^i$.

b. Broadcast $X_j = g^{\frac{\alpha_j}{a}}$ and $Y_j = g^{\alpha_j}, j = 0, 1, ..., n-1$. By the Chinese remainder theorem, we can obtain the re-signature key $rk^i_{A \to B} \in Z_F$, namely $rk^i_{A \to B} = (pk_A^2)^{f_t(i)} modq_{t-1} = h^{\frac{f_t(i)}{a}}, t = 1, 2, ..., n$. Then, the information $(i, rk^I_{A \to B})$ is secretly sent to the agent $P_i, i = 1, 2, ..., n$, where $X_0 = g^{\frac{b}{a}}, Y_0 = pk_B = g^b$.

c. The agent $P_i (1 \leq i \leq n)$ first calculates $rk^{n,i}_{A \to B} = rk^i_{A \to B} modq_{t-1}$, then determines whether the sub-key $rk^i_{A \to B}$ is valid by verifying whether the following two formulas are true.

$$e(rk^{n,i}_{A \to B}, g) = e(\prod_{j=0}^{n-1} X_j^{i^j}, h), \tag{1}$$

and

$$e(\prod_{j=0}^{n-1} X_j, pk_A^1) = e(\prod_{j=0}^{n-1} Y_j^{i^j}, g). \tag{2}$$

If the verifications are established, it indicates that the sub-key $rk^i_{A \to B}$ is valid. Given any positive integer $t(1 \leq t \leq n)$, the agent P_i can separately calculate $rk^{t,i}_{A \to B} = rk^i_{A \to B} modq_{t-1}$ by the initially obtained re-signature key $rk^i_{A \to B}$, and broadcast its verification public key $\nu k_{t,j} = g^{f_t(i)} = \prod_{j=0}^{t-1} Y_j^{i^j}$.

Sign: Given that the trustee's private key is a and a n_m bit long message $m = (m_1, m_2, ..., m_{n_m}) \in \{0, 1\}^{n_m}$, then pick a random constant t and let $r = h^t$, $s = a(H(m||r) + t)(modq)$, output the strong signature $\sigma = (r, s)$ and output a weak signature $\sigma = (r, h^s)$ that cannot be resigned.

Resign:

a. Partial key generation: Assuming the threshold is t, enter a threshold t, public key pk, message m, and signature σ_A, Verify this equation $Verify(pk_A, m, \sigma_A) = 0$, if it is established, enter the resigned sub-key $rk^{t,i}_{A \to B}$, and obtain the corresponding re-signature

$$\sigma_{B,i} = (r, s_i) = (r, (rk^{t,i}_{A \to B})^s) = (r, h^{f_t(i)(H(m||r)+t)}), \tag{3}$$

if it is not established, output 0.

b. Re-signature generation: After combiner obtains some partial re-signatures σ_{B,i_1}, verify the legality of partial signature by verifying whether the following equation

$$e(g, s_i) = e(\nu k_{t,i}, rh^{H(m||r)}), \tag{4}$$

is valid. If the composer obtains at least t legally partial duplicate signatures $(\sigma_{B,i_1}, ..., \sigma_{B,i_t})$, then its re-signature is where is the coefficient of Lagrange interpolation polynomial.

$$\sigma_B = (r, \prod_{i=1}^{t} s_i^{\chi_{0,i}}) = (r, \prod_{i=1}^{t} h^{\chi_{0,i} f(i)(H(m||r)+t) \sum_{i=1}^{t} \chi_{0,j} f(i)}) = (r, h^{b(H(m||r)+t)}), \tag{5}$$

where $\chi_{0,i}$ is the coefficient of Lagrange interpolation polynomial.

Verify: We input the public key $pk_A = (pk_A^1, pk_A^2)$, the message m and the signature σ to be verified (when σ is the signature under the weak key, let $s = h^s$), if the equation

$$e(g, s) = e(pk_B, rh^{H(m||r)}) \tag{6}$$

is established, then output 1, otherwise output 0.

Server-setup: Given a system parameter cp, the verifier picks a random element x from Z_q^* to make the string $Vst = x$.

Server-verify: Given $Vst = x$, a public key pk and a signed message pair $(m, \sigma = (\sigma_1, \sigma_2))$, the server-assisted authentication interaction protocol between the verifier and the server is as follows:

a. Firstly, the verifier calculates $\sigma^{'} = (\sigma_1^{'}, \sigma_2^{'}) = ((\sigma_1)^x, (\sigma_2)^x) = (r^x, s^x)$. Then the verifier sends $m, \sigma^{'}$ to the server.
b. The server calculates $\eta_1 = e(g, \sigma_1^{'})$ and $\eta_2 = e(pk_B, \sigma_2^{'} h^{H(m||r)})$, and sends η_1, η_2 to the verifier.
c. The verifier through the calculation to verify whether the equation

$$\eta_1 = \eta_2 \tag{7}$$

is established. If the equation is true, the verifier is convinced that σ is the legal signature of the message m, and outputs 1, otherwise, the verifier is convinced that σ is an invalid signature and outputs 0.

5 Correctness Analysis

Theorem 1. *When the threshold is t, if (1) and (2) are established, the obtained re-signature sub-key is valid.*

Proof. Because of $rk_{A\to B}^i = h^{\frac{f_t(i)}{a}}, t = 1, 2, ..., n$, Lagrange polynomial and the properties of bilinear mapping, we get

$$e(rk_{A\to B}^{n,i}, g) = e(h^{\frac{f_t(i)}{a}}, g) = e(h^{\sum_{j=0}^{n-1} \frac{\alpha_j^{i^j}}{a}}, g) = e(h, g^{\frac{\sum_{j=0}^{n-1} \alpha_j^{i^j}}{a}})$$

$$= e(h, \prod_{j=0}^{n-1} (g^{\frac{\alpha_j}{a}})^{i^j}) = e(h, \prod_{j=0}^{n-1} X_j^{i^j}).$$

Due to $A_j = g^{\frac{\alpha_j}{a}}$, we obtain

$$e(\prod_{j=0}^{n-1} X_j, pk_A^1) = e(\prod_{j=0}^{n-1} g^{\frac{\alpha_j}{a}}, g^a) = e(\prod_{j=0}^{n-1} g^{\alpha_j}, g) = e(\prod_{j=0}^{n-1} Y_j, g). \tag{8}$$

In addition, we have

$$rk_{A\to B}^{t,i} = rk_{A\to B}^{i} mod q_{t-1} = h^{\frac{f_t(i)}{i}}, \tag{9}$$

and

$$\nu k_{t,i} = \prod_{j=0}^{t-1} Y_j^{i^j} = \prod_{j=0}^{t-1} g^{\alpha_j^{i^j}} = g^{f_t(i)}. \tag{10}$$

Theorem 2. *When the threshold is t, if (4) is established, the partial re-signature is valid.*

Proof. From $s_i = h^{f_t(i)(H(m||r)+t)}$, $\nu k_{t,i} = g^{f_t(i)}$ and the properties of bilinear mapping, we obtain

$$e(g, s_i) = e(g, h^{f_t(i)(H(m||r)+t)}) = e(g^{f_t(i)}, h^{H(m||r)+t}) = e(\nu k_{t,i}, rh^{H(m||r)}). \tag{11}$$

Theorem 3. *When the threshold is t, if (6) is established, the re-signature is valid.*

Proof. From

$$s_i = h^{f_t(i)(H(m||r))}$$

and the properties of bilinear mapping, we have Displayed equations are centered and set on a separate line.

$$e(g, s) = e(g, \prod_{i=1}^{t} s_i^{\chi_{0,i}}) = e(g, \prod_{i=1}^{t} h^{\chi_{0,i} f(i)(H(m||r)+t)}) = e(g, h^{(H(m||r)+t)\sum_{i=1}^{t} \chi_{0,i} f_t(i)})$$

$$= e(g, h^{b(H(m||r)+t)}) = e(g^b, h^{H(m||r)+t}) = e(pk_B, rh^{H(m||r)}),$$

where $r = h^t$.

Theorem 4. *If the equation (7) is established, the verifier is convinced that σ is the legal signature of the message m.*

Proof. For the signature of the principal Bob $\sigma_B = (\sigma_{B1}, \sigma_{B2}) = (r, s)$ and the character string $Vst = x$, then we have Displayed equations are centered and set on a separate line.

$$\eta_1 = e(g, \sigma_{B1}^{'}) = e(g, (\prod_{i=1}^{t} s_i^{\chi_{0,i}})^{'}) = e(g, (\prod_{i=1}^{t} s_i^{\chi_{0,i}})^{x})$$

$$= e(g, (\prod_{i=1}^{t} h^{\chi_{0,i} f(i)(H(m||r)+t)})^x) = e(g, (h^{(H(m||r)+t)) \sum_{i=1}^{t} \chi_{0,i} f(i)})^x)$$

$$= e(g, (h^{b(H(m||r)+t)})^x) = e(g^{bx}, h^{H(m||r)+t}) = e((pk_B)^x, h^{H(m||r)+t})$$

$$= e((pk_B)^x, rh^{H(m||r)}) = e(pk_B, r^x h^{H(m||r)}) = e(pk_B, r^{'} h^{H(m||r)}) = \eta_2.$$

Through the above derivation process, it can be proved that when the threshold is t, the re-signature sub-key, partial re-signature and re-signature verification algorithm are effective, and the correctness of the server-assisted verification protocol is obtained. Since the original signature is the same length as the re-signature, this scheme satisfies transparency and multi-purpose rows. In addition, since the trustee's private key, the principal's private key, and the agent's re-signature key are all elements in Z_q^*, the scheme satisfies the key optimality.

6 Security Analysis

The following is an analysis of the scheme proposed in this paper is non-forgeable and robust, and the server verification protocol $Servier - verify$ of the scheme satisfies the completeness. Therefore, in order to prove the security of the scheme, it is necessary to prove that the scheme satisfies the non-forgeable, robustness and completeness of the server-assisted verification protocol.

The third adversary who wants to forge the proxy re-signature of message m for the proxy signers and original signer must have the original signer's signature $\sigma_A(m)$, and it cannot be forged. Next, we will explain by the proof of the following Theorem that even if the third adversary knows the pair (r, s) sent by the original signer, he still cannot make a forgery signature on any other message. So he cannot make a forgery proxy signature on m either. On the other hand, the original signer cannot create a valid proxy re-signature, because the proxy re-signature is obtained by the proxy signers using the CDH signature scheme and the proxy signers' secret proxy $\{rk^i_{A \to B}\}$ shares which contain the private key of each proxy signer.

Theorem 5. *Assuming the third adversary has the $\sigma_A(m) = (r, s)$, our scheme is still secure.*

Proof. If we want to know the re-signature $\sigma_B = (r, h^{b(H(m||r)+t)})$ of message m, we must know Bobs private key $b, H(m||r)$ and t. Although he has known the signature $\sigma_A(m = (r, s))$, where $r = h^t, s = a(H(m||r) + t)(modq)$, he still cannot get $H(m||r)$ and t, because this problem is equivalent to the discrete logarithm problem. Even he cannot get Bobs private key t. Thus, the scheme is still secure.

Theorem 6. *Under the standard model, when $n \geq 2t - 1$, our proposed scheme is robust to any attacker who can collude with the $t - 1$ agents.*

Proof. The compositor is able to verify the legitimacy of a partial re-signature and therefore can reject a malicious agent. Since there are at least t honest agents and each honest agent calculates a legal re-signature σ_i, the synthesizer Combiner can obtain the set Φ of the honest agent's serial number i and $|\Phi| \geq t$. Therefore, the compositor can always have a legal partial re-signature to calculate the re-signature of the message m. From this, we can get the scheme is strong when $n \geq 2t - 1$.

Theorem 7. *The server-assisted authentication protocol $Servier - verify$ of the proposed scheme is complete under collusion attack and adaptive selection message attack.*

We prove this theorem by the following two lemmas.

Lemma 1. *Assuming that the attacker of the server and the trustee Alice is A_1, the probability that the attacker A_1 makes the challenger C convinced that an illegal original signature is legal is negligible.*

Proof. Let A_1 act as the server in the server's secondary authentication protocol, and C acts as the certifier in the protocol. Given the illegal original signature of a message, the goal of A_1 is to convince C that the illegal signature is legitimate. The interaction process between attacker A_1 and challenger C is as follows:

Establishment: Challenger C executes the initialization algorithm to generate the system parameter cp, randomly selects two elements $x^*, \gamma \in Z_q^*$, makes $Vst = x^*$, and calculates the public-private key pair of Alice $(pk_A, sk_A) = (e(g^\gamma, h^{\frac{1}{\gamma}}))$. Then it sends $\{cp, pk_A, sk_A\}$ to attacker A_1.

Query: Attacker A_1 can adaptively perform a limited number of server-assisted verification queries. For each inquiry (m_i, σ_i), Challenger C and attacker A_1 perform a server-assisted authentication protocol, and then return the output of the protocol as a response to attacker A_1.

Output: Finally, the attacker A_1 outputs the message m^* and the string $\sigma^* = (\sigma_1^*, \sigma_2^*)$. Let Γ_{m^*} be the set of all legal signatures of the message m^* corresponding to the public key pk_A, and σ^* is not in Γ_{m^*}. After the challenger C receives (m^*, σ^*), it uses Vst to calculate $(\sigma^*)^{'} = ((\sigma_1^*)^{'}, (\sigma_2^*)^{'}) = ((\sigma_1^*)^{x^*}, (\sigma_2^*)^{x^*})$, and sends $(\sigma^*)^{'} = ((\sigma_1^*)^{'}, (\sigma_2^*)^{'})$ to attacker A_1. Then the attacker A_1 calculates $\eta_1^* = e(g, (\sigma_1^*)^{'})$ and $\eta_2^* = e(pk_B, (\sigma_2^*)^{'})$, and returns $\eta_1^* = \eta_2^*$ to Challenger C. The probability of the equation $\eta_1^* = \eta_2^*$ being established is $\frac{1}{q-1}$.

a. Since $(\sigma^*)^{'} = (\sigma^*)^{x^*}$ and $x^* \in_R Z_q^*$, Therefore, the probability that the attacker A_1 successfully forged $(\sigma^*)^{'}$ by σ^* is $\frac{1}{q-1}$.

b. Suppose the attacker A_1 returns (η_1^*, η_2^*) such that $\eta_1^* = \eta_2^*$, then $\eta_1^* = e(pk_B, r^{x^*} h^{H(m||r)}) = e(pk_B, rh^{H(m||r)})^{x^*}$. For the sake of writing, let $e(pk_B), rh^{H(m||r)}) = M$. Through a simple calculation, we get

$$x^* = \log_M \eta_1^*. \tag{12}$$

Since x^* is an element that randomly selected from Z_q^*, the attacker finds x^* such that the probability that the above equation holds is $\frac{1}{q-1}$.

In summary, the attacker A_1 makes the challenger C convinced that the probability that the message (m^*, σ^*) is a legal signature is $\frac{1}{q-1}$. Since q is a large prime number, the attacker A_1 makes the challenger C convinced that the probability of (m^*, σ^*) being a legal signature is negligible.

Lemma 2. *Assuming that A_2 is an attacker colluded by the server and t agents, A_2 makes Challenge C believe that the probability that an illegal re-signature is legal is negligible.*

Proof. Let A_2 acts as the server role in the server-assisted authentication protocol, and C is the certifier role in the protocol. Given the illegal signature of a message, the goal of A_2 is to convince C that the illegal signature is legitimate. The interaction between the two is as follows:

Establishment: Challenger C runs the system initialization algorithm to get the system parameter *cp*, randomly selects the three elements in Z_q^* as x^*, α, β, and makes $Vst = x^*$. The public-private key pairs of Alice and Bob are calculated as $(pk_A, sk_A) = (e(g^\alpha, h^{\frac{1}{\alpha}}), \alpha)$, $(pk_B, sk_B) = (e(g^\beta, h^{\frac{1}{\beta}}), \alpha)$ and the re-signature key $rk_{A\to B} = \beta/\alpha$. Send cp, pk_A, pk_B, and $rk_{A\to B}$ to attacker A_2.

Query: Same as the interrogation response process in Lemma 1.

Output: Attacker A_2 outputs message m^* and string $\sigma^* = (\sigma_1^*, \sigma_2^*)$. Let Γ_{m^*} be the set of legal signatures of message m^* corresponding to public key pk_B, and σ^* is not in Γ_{m^*}. Similar to the analysis process in Lemma 1, the attacker A_2 makes the challenger C convinced that (m^*, σ^*) is a legal signature with a probability of $1/(q-1)$, so the probability that the attacker A_2 convinced the challenger C that (m^*, σ^*) is a legal signature is negligible.

In summary, we get the one-way server-assisted verification threshold proxy re-signature scheme proposed in this paper is safe under collusion attack and adaptive selection message attack.

Next, we present a performance analysis of the server-assisted verification threshold proxy re-signature scheme.

7 Performance Analysis

The computational difficulty of the server-assisted verification threshold proxy re-signature scheme adapted to the mobile Internet proposed in this paper is equivalent to the CDH problem. In order to compare performance with the currently existing threshold proxy re-signature algorithm, the following symbols are defined in this paper. It should be noted that since the calculation amount of the addition, multiplication, HMAC algorithm and hash function is relatively small, we only consider the exponential operation and the bilinear pair operation with large calculation amount when considering the calculation overhead (Table 1).

Table 1. Calculation symbol representation

Symbol	Description
C_m	$Multiplicationcalculation$
C_n	$Additioncalculation$
C_o	$HMACalgorithmcalculation$
C_h	$Hashfunctioncalculation$
$\|G_1\|$	$ThelengthoftheelementinG_1$
$\|G_2\|$	$ThelengthoftheelementinG_2$
C_p	$Indexcalculation$
C_q	$Bilinearpairingcalculation$

The following analysis will be carried out from four aspects: secret segmentation, signature algorithm, re-signature algorithm and signature verification. The re-signature algorithm includes two parts: partial re-signature algorithm and synthetic re-signature algorithm. The calculation amount of the server-assisted verification threshold proxy re-signature algorithm adapted to the mobile Internet is as shown in Table 2.

Table 2. Calculated amount of the program in this paper

Procedure	Calculated amount
$Rekey$	$(4+2t)C_p$
$Sign$	C_p
$Re-sign$	$2t(C_p+C_q)$
$Verify$	$2C_q$
$Server-verify$	$3C_p+2C_q$

The literature [21] and [22] respectively propose the threshold proxy re-signature scheme. The server-assisted verification threshold proxy re-signature algorithm proposed in this paper is compared with the existing two algorithms based on its signature length and computational cost. The comparison results are as follows:

From the results of Table 3, it can be seen that compared with the literature [21,22], the computational cost of the new scheme in this paper is much smaller than that of the literature [21,22]. In the re-signature algorithm, the exponential operation of this scheme is only $2t$ operations, far less than other schemes, and the bilinear pairing operation in this procedure is a bit greater than that of the literature [21,22]. However, in the verification procedure However, in the verification process of this scheme, through the interaction protocol between the verifier and the server, the bilinear pairing operation with high computational

Table 3. Calculation overhead of the threshold proxy re-signature algorithm

Scheme	Signature length	Re-signature generation	Verification
Alg.in [21]	$4\|G_1\|$	$(4t+6)C_p+5C_q$	$5C_q$
Alg.in [22]	$3\|G_1\|$	$(3t+2)C_p$	$4C_q$
Ours	$\|G_1\|$	$2t(C_p+C_q)$	0

complexity is transferred to the server for execution, which reduces the computational burden of the verifier, thus saving the verification time and improving the efficiency of verification. In addition, the signature length of this article is much shorter than that of the literature [21,22], saving storage space. Therefore, the new algorithm proposed in this paper is more advantageous than the previous algorithm.

In the new scheme of this paper, the verifier transfers the complex bilinear pairing operation task to the server through the server-assisted verification protocol, so the signature verification does not need to perform a computationally intensive bilinear pairing operation. Therefore, the problem of limited computing power of mobile terminals in the mobile Internet environment is solved. In addition, under the standard model, the proposed scheme is non-forgeable under the adaptive selection message, and the server-assisted verification protocol process is complete. Therefore, the server-assisted verification threshold proxy re-signature scheme proposed in this paper is safe under collusion attacks and adaptive selective message attacks, so as to meet the requirements for high security requirements due to the complexity of the mobile Internet environment. In summary, this paper proposes that the server-assisted verification threshold proxy re-signature scheme can be better adapted to the mobile Internet environment.

8 Conclusion

At present, mobile Internet technology and its applications have been rapidly developed, and some low-end computing devices such as smart phones have been widely used. However, the corresponding information security mechanism issues and low-end computing power, limited energy supply, etc. problems have not yet found an effective solution. Aiming at these problems, this paper proposes a formal model of server-assisted verification threshold proxy re-signature, constructs a specific implementation scheme, and gives corresponding security proof. The scheme is based on threshold proxy re-signature and server-assisted authentication scheme. The threshold proxy re-signature algorithm can resist joint attacks and overcome various security defects. Verifiers and servers transfer complex bilinear pairing operations to servers through the interaction protocol between them, which greatly reduce the computational complexity of verifiers, improve the verification efficiency, and satisfy the needs of low-end computing devices with weak computing power and limited energy supply. The comprehensive analysis shows that the scheme can be well applied to the application environment of mobile internet.

References

1. Girault, M., Lefranc, D.: Server-aided verification: theory and practice. In: Roy, B. (ed.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 605–623. Springer, Heidelberg (2005). https://doi.org/10.1007/11593447_33
2. Wei, W., Yi, M., Willy, S., et al.: Provably secure server-aided verification signatures. Comput. Math. Appl. **61**(7), 1705–1723 (2011)
3. Wu, H., Xu, C.X., Deng, J.: A server-aided aggregate verification signature scheme from bilinear pairing. In: Proceedings of INCS, China, Xi'an, pp. 503–506 (2013)
4. Ateniese, G., Hohenberger, S.: Proxy re-signatures: new definitions, algorithms, and applications. In: Proceedings of the 12th ACM CCS, Alexandria, USA, pp. 310–319 (2005). https://doi.org/10.1145/1102120.1102161
5. Shao, J., Cao, Z., Wang, L., Liang, X.: Proxy re-signature schemes without random Oracles. In: Srinathan, K., Rangan, C.P., Yung, M. (eds.) INDOCRYPT 2007. LNCS, vol. 4859, pp. 197–209. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-77026-8_15
6. Waters, B.: Efficient identity-based encryption without random Oracles. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 114–127. Springer, Heidelberg (2005). https://doi.org/10.1007/11426639_7
7. Kiiate, K., Ikkwon, Y., Secogan, L.: Remark on Shao et al.'s bidirectional proxy re-signature scheme in indocrypt'07. Int. J. Netw. Secur. **9**(1), 8–11 (2009). https://doi.org/10.6633/IJNS.200907.9(1).02
8. Libert, B., Vergnaud, D.: Multi-use unidirectional proxy re-signatures. In: Proceedings of the 15th ACM Conference on Computer and Communications Security, Alexandria, USA, pp. 511–520 (2008). https://doi.org/10.1145/1455770
9. Wang, W.P.: An identity-based blind proxy re-signature scheme. Comput. Appl. Softw. **29**(10), 308–313 (2012). https://doi.org/10.3969/j.issn.1000
10. Tian, M.M.: Identity-based proxy re-signatures from lattices. Inf. Process. Lett. **115**(4), 462–467 (2015). https://doi.org/10.1016/j.ipl.2014.12.002
11. Jiang, M.M., Hu, Y.P., Wang, B.C., et al.: Identity-based unidirectional proxy re-signature over lattice. J. Electron. Inf. Technol. **36**(3), 645–649 (2014). https://doi.org/10.3724/SP.J.1146.2013.00818
12. Hao, S.G., Zhang, L., Muhammad, G.: A union authentication protocol of cross-domain based on bilinear pairing. J. Softw. **8**(5), 1094–1100 (2013). https://doi.org/10.4304/jsw.8.5.1094-1100
13. Nguyen, T.C., Shen, W., Luo, Z., Lei, Z., Xu, W.: Novel data integrity verification schemes in cloud storage. In: Lee, R. (ed.) Computer and Information Science. SCI, vol. 566, pp. 115–125. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-10509-3_9
14. Sun, Y., Chen, X.Y., Du, X.H.: A proxy re-signature scheme for stream switching. J. Softw. **26**(1), 129–144 (2015). https://doi.org/10.13328/j.cnki.jos.004553
15. Long, Z.H., Gong, J., Wang, B.: Energy efficiency study of clustered secure routing protocol secure communication method in wireless sensor networks. J. Electron. Inf. **37**(8), 2000–2006 (2015). https://doi.org/10.11999/JEIT141284

16. Boneh, D., Franklin, M.: Identity-based encryption from the weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44647-8_13
17. Long, Z.H., Gong, J., Wang, B., et al.: Energy efficiency study of secret communication method on clustering. J. Electron. Inf. Technol. **37**(8), 2000–2006 (2015). https://doi.org/10.11999/JEIT141284
18. Bao, F., Deng, R.H., Zhu, H.F.: Variations of Diffie-Hellman problem. In: Qing, S., Gollmann, D., Zhou, J. (eds.) ICICS 2003. LNCS, vol. 2836, pp. 301–312. Springer, Heidelberg (2003). https://doi.org/10.1007/978-3-540-39927-8_28
19. Shamir, A.: How to share a secret. Commun. ACM **22**, 612–613 (1979)
20. Wang, Z., Lu, W.: Server-aided verification proxy re-signature. In: Proceedings of Trust, Security and Privacy in Computing and Communications, Melbourne, Australia, pp. 1704–1707 (2013). https://doi.org/10.1109/TrustCom.2013.211.
21. Yang, X.D., Wang, C.F.: Flexible threshold proxy re-signature schemes. Chin. J. Electron. **20**(4), 691–696 (2011)
22. Li, H.Y., Yang, X.D.: One-way variable threshold proxy re-signature scheme under standard model. Comput. Appl. Softw. **12**, 307–310 (2014)