* The REESSE1+ Public-key Cryptosystem

—— *A Multiproblem Public-key Cryptosystem*

Shenghui Su [1], and Shuwang Lü [2]

[1] School of Info. Engi., Univ. of Science & Technology Beijing, Beijing 100083, P. R. C.

sheenway@126.com

[2] School of Graduate, Chinese Academy of Sciences, Beijing 100039, P. R. C.

A Version before March 15 2007

Abstract: This paper gives the definition of a coprime sequence and the concept of the lever function, describes the five algorithms and six characteristics of the REESSE1+ public-key cryptosystem based on three new hardnesses: the modular subset product problem, the multivariate arrangement problem, and the super logarithm problem in a prime field, shows the correctness of the decryption algorithm, and infers that the probability that a plaintext solution is not unique is nearly zeroth. The authors analyze the security of REESSE1+ against recovering a related plaintext from a ciphertext, extracting a related private key from a public key or a signature, and faking a digital signature via a public key or a known signature with a public key, discuss the super logarithm problem, and believe that the security of REESSE1+ is at least equal to the time complexity of $O(2^n)$ at present. At last, the paper expounds the idea of optimizing REESSE1+ through binary compact sequences.

Keywords: Multiproblem public-key cryptosystem, Coprime sequence, Security, Lever function, Super logarithm problem, Double congruence theorem.

1 Introduction

The trapdoor functions for RSA [1] and ElGamal [2] public-key cryptosystems [3] are computationally one-way [4]. Along with the elevation of computer speeds, such one-wayness will be weakened. Hence, to enhance the one-wayness of a trapdoor function, sometimes a public-key cryptosystem is transplanted to a complex algebraic system from a simple one. For example, the ElGamal analogue in an elliptic curve group, namely the ECC cryptosystem, is more one-way or more secure than ElGamal itself [5]. However, this method is not suitable for all the existing cryptosystems.

In some public-key cryptosystems, trapdoor functions can prevent a related plaintext from being recovered from a ciphertext, but cannot prevent a related private key from being extracted from a public key. For instance, in the MH knapsack cryptosystem [6], the subset sum problem which serves as a trapdoor function can not preclude a private key from being inferred through the Shamir method [7].

Different from RSA, ElGamal and MH, REESSE1+ brings 3 independent variables into the general key transform. Its security is not based on classical hardnesses: the subset sum problem, the integer factorization problem and the discrete logarithm problem, but on three new hardnesses: the modular subset product problem, the multivariate arrangement problem, and the super logarithm problem. The modular subset product problem as a trapdoor function ensures the security of a plaintext encrypted, the multivariate arrangement problem triggered by the lever function $\ell(.)$ ensures the security of a private key, and the super logarithm problem in a prime field ensures the security of a digital signature.

Obviously, REESSE1+ is a type of multiproblem cryptosystem.

A Multiproblem Cryptosystem: If the security of a public-key cryptosystem is based on not less than three hardnesses each of which can not solved in polynomial time, this public-key cryptosystem is called a multiproblem cryptosystem.

* The newest version of this paper can be accessed at http://www.arxiv.org/pdf/cs/0702046.

The security of a multiproblem cryptosystem is equivalent to that hardness whose time complexity is smallest in all the involved hardnesses.

A multiproblem cryptosystem must be a multivariate one or multiparameter one because only multiple variables may bring in multiple hardnesses.

In this paper, unless otherwise specified, sign '%' means 'modulo', and $\langle a, b\rangle$ represents the greatest common divisor of two integers. Let $|x|$ denote the order of x % M, and '% –1' denote '% $(M-1)$'.

2 A Coprime Sequence and the Lever Function

Definition 1 If A_1, A_2, …, and A_n are n integers which are each greater than 1, pairwise distinct and relatively prime, this series of integers is called a coprime sequence, namely a relatively prime sequence, denoted by $\{A_1, \ldots, A_n\}$, and shortly $\{A_i\}$.

Property 1 For any positive integer $m \le n$, if we select randomly m elements from $\{A_i\}$ and construct a subset, i.e. a subsequence $\{A_{x_1}, \ldots, A_{x_m}\}$, the subset product $G = A_{x_1} \ldots A_{x_m}$ is uniquely determined, that is, the mapping from G to $\{A_{x_1}, \ldots, A_{x_m}\}$ is one-to-one. G is also called a coprime sequence product.

Proof: By reduction to absurdity.

Because $A_1, \ldots, A_n$ are pairwise relatively prime, for arbitrary $A_j, A_k \in \{A_1, \ldots, A_n\}$, there must exist $\langle A_j, A_k\rangle = 1$, namely there is not the same prime divisor between A_j and A_k. It manifests that the prime divisors of every element do not belong to any other elements.

Presume that G is acquired from two different subsequences $\{A_{x_1}, \ldots, A_{x_m}\}$ and $\{A_{y_1}, \ldots, A_{y_h}\}$, hereby

$$G = A_{x_1} \ldots A_{x_m} = A_{y_1} \ldots A_{y_h}.$$

Since the two subsequences are unequal, there must exist a certain element A_q which does not belong to the two subsequences at one time.

Without loss of generality, let $A_q \in \{A_{x_1}, \ldots, A_{x_m}\}$ and $A_q \notin \{A_{y_1}, \ldots, A_{y_h}\}$.

In terms of the fundamental theorem of arithmetic [8], there must exist a prime number p which is the divisor of A_q.

It is as above that the prime divisors of every element do not belong to any other elements, and thus the prime p must be the divisor of the product $A_{x_1}, \ldots, A_{x_m}$ but not the divisor of the product $A_{y_1}, \ldots, A_{y_h}$. It means that the integer G has two distinct prime factorizations, which is contrary to the fundamental theorem of arithmetic.

Therefore, the mapping relation between G and $\{A_{x_1}, \ldots, A_{x_m}\}$ is one-to-one.

In the REESSE1+ cryptosystem, the general key transform is $C_i \equiv A_i W^{\ell(i)}$ (% M), where $\ell(i)$ is an exponential.

Definition 2 In a public key cryptosystem, the parameter $\ell(i)$ in the key transform is called the lever function, if it has the following features:

- $\ell(.)$ is an injection from integers to integers, its domain is $[1, n]$, and codomain $(1, M)$. Let $Ł_n$ represent the collection of all injections from the domain to the codomain, then $\ell(.) \in Ł_n$ and $|Ł_n| \ge A_n^n = n(n-1) \ldots 1$.
- The mapping between i and $\ell(i)$ is established randomly without an analytical formula, so every time a public key is generated, the function $\ell(.)$ is distinct.
- There does not exist any dominant or special mapping from $\ell(.)$ to a public key.
- An attacker have to consider all the arrangements of the sequence $\{\ell(i) \mid i = 1, \ldots, n\}$ when extracting a related private key from a public key. Thus, if n is large enough, it is infeasible for the attacker to search the arrangements exhaustively.
- A receiver owning a private key only needs to consider the accumulative sum of the sequence $\{\ell(i)\}$ when recovering a related plaintext from a ciphertext. Thus, the time complexity of decryption is polynomial in n, and the decryption is feasible.

Obviously, there is the large amount of calculation on $\ell(.)$ at 'a public terminal', and the small amount of calculation on $\ell(.)$ at 'a private terminal'.

3 Design of the REESSE1+ Public Key Cryptosystem

3.1 The Key Generation Algorithm

This algorithm is employed by a third-party authority. Every user is given a pair of keys.

Assume that S, T, $Đ$, $đ$ are pairwise coprime integers, where the binary form of S only contains two '1' bits respectively at the beginning and the end, $T \geq 2^n$, $Đ \geq 2^n$, and $đ$ is a non-large integer.

(1) Randomly generate a coprime sequence $\{A_1, \ldots, A_n\}$, and compute $G = \prod_{i=1}^{n} A_i$.

(2) Find a prime $M > G$ making $\langle S, M-1\rangle = 1$, $đĐT \mid (M-1)$,

and $q \mid (M-1)$ for any prime $q \in [1, n+4]$.

(3) Pick δ making $\langle \delta, M-1\rangle = 1$, and $|\delta| = đĐT$.

(4) Compute $\alpha \leftarrow \delta^{\delta^n}$, $W \leftarrow G^{-1}(\alpha\delta^{-1})^{1/S}$, $\beta \leftarrow \delta^{(\delta+1)WS}$, and $\gamma \leftarrow \delta^{W^n} \% M$.

(5) Produce pairwise distinct $\ell(1), \ldots, \ell(n) \in \Omega = \{i\,\delta(\%-1) \mid i = 5, \ldots, n+4\}$.

(6) Compute $\{C_1, \ldots, C_n \mid C_i \leftarrow A_i W^{\ell(i)} \% M$, for $i = 1, \ldots, n\}$.

At last, the public key is $(\{C_i\}, \alpha, \beta, \gamma)$, and the private key $(\{A_i\}, \{\ell(i)\}, W, \delta, Đ, đ)$. S, T, and M are common.

Remark: $\Omega = \{i\,\delta(\%-1) \mid i = 5, \ldots, n+4\}$ is not a unique selection — $\Omega = \{i+\delta(\%-1) \mid i = 5, \ldots, n+4\}$ for example. The principles for selecting Ω is that 1) $\ell(i) \geq 5$; 2) the elements in Ω are pairwise distinct; 3) decryption time complexity does not exceed $O(n^3)$.

We know that in degree 5 or higher, the congruence $x^n \equiv c\,(\% M)$ has a non-solvable Galois group.

To seek a certain element x of order k, first do $x \equiv c^{(M-1)/k}\ (\% M)$, where $c < M$ is an arbitrary integer, then test x by the algorithm 4.80 in section 4.6 of reference [3].

3.2 The Encryption Algorithm

Assume that $(\{C_i\}, \alpha, \beta, \gamma)$ is the public key, and $b_1 \ldots b_n$ is an n-bit plaintext block or symmetric key.

(1) Set $\hat{\boldsymbol{G}} \leftarrow 1$, $i \leftarrow 1$.

(2) If $b_i = 1$, $\hat{\boldsymbol{G}} \leftarrow \hat{\boldsymbol{G}} C_i \% M$.

(3) Let $i \leftarrow i + 1$. If $i \leq n$, go to (2), or else end.

After the algorithm is executed, the ciphertext $\hat{\boldsymbol{G}}$ is gained.

Note that in encryption, α, β, and γ are not helpful.

3.3 The Decryption Algorithm

Assume that $(\{A_i\}, \{\ell(i)\}, W, \delta, Đ, đ)$ is the private key, and $\hat{\boldsymbol{G}}$ is the ciphertext.

(1) Compute $\hat{\boldsymbol{G}} \leftarrow \hat{\boldsymbol{G}}(W^{-1})^{\delta} \% M$.

(2) Set $b_1 \ldots b_n \leftarrow 0$, $G \leftarrow \hat{\boldsymbol{G}}$, $i \leftarrow 1$.

(3) If $A_i \mid G$, set $b_i \leftarrow 1$ and $G \leftarrow G / A_i$.

(4) Let $i \leftarrow i + 1$. If $i \leq n$ and $G \neq 1$, go to (3).

(5) If $G \neq 1$, go to (1), or else end.

At last, the $b_1 \ldots b_n$ is the original plaintext block or symmetric key.

This algorithm can always terminate normally as long as $\hat{\boldsymbol{G}}$ is a true ciphertext

Note that in decryption, $\{\ell(i)\}$, $Đ$, and $đ$ are not helpful.

3.4 The Digital Signature Algorithm

Assume that $(\{A_i\}, \{\ell(i)\}, W, \delta, Đ, đ)$ is the private key, F is a file or message which will be signed, and *hash* is a one-way compression function.

(1) Let $H \leftarrow hash(F)$, whose binary form is $b_1 \ldots b_n$.

(2) Set $k_1 \leftarrow \sum_{i=1}^{n} b_i \ell(i) \,\%\, -1$, $G_0 \leftarrow \prod_{i=1}^{n} A_i^{\neg b_i}$.

(3) Pick Q making $Đ \mid (\delta Q - WH)$, $đ \nmid ((\delta Q)^n - (WH)^n) \,(\%\, -1)$.

Compute R such that $Q \equiv (R G_0)^S \delta \,(\%\, M)$.

(4) Compute $U \leftarrow (R W^{k_1 - 1} \delta^{\delta(\delta+1)})^{QT} \,\%\, M$.

If $đ \nmid ((\delta+1) SU + \sum_{i=0}^{n-1} (\delta Q)^{n-1-i} (WH)^i) \,(\%\, -1)$, go to (3).

At last, the signature (Q, U) on the file F is obtained, and sent to a receiver with F.

In terms of the double congruence theorem (see sect. 3.6), we do not need $V \equiv (R^{-1} W G_1)^{QU} \delta^{\lambda} \,(\%\, M)$ in the signature, where $G_1 = \prod_{i=1}^{n} A_i^{b_i}$, and λ satisfies $\lambda S \equiv ((\delta+1) SU + \sum_{i=0}^{n-1} (\delta Q)^{n-1-i} (WH)^i)(\delta Q - WH) \,(\%\, -1)$. It indicates $đ Đ \mid \lambda$.

Clearly the probability of finding out suitable U is $1 / đ$. Since $đ$ is a non-large number, U can be found out at a good pace.

Due to $\langle S, M - 1 \rangle = 1$, computing R by $Q \equiv (R G_0)^S \delta \,(\%\, M)$ may resort to the following theorem 1.

Theorem 1 For the congruence $x^k \equiv c \,(\%\, p)$ with p is prime, if $\langle k, p - 1 \rangle = 1$, every c has just one k-th root modulo p. Especially, let μ be the multiplicative inverse of k modulo $(p - 1)$, then $c^{\mu} \,\%\, p$ is one k-th root.

Further, we have theorem 2.

Theorem 2 For the congruence $x^k \equiv c \,(\%\, p)$, if $k \mid (p - 1)$ and $\langle k, (p - 1) / k \rangle = 1$, then when c is one k-th power residue modulo p, $c^{\mu} \,\%\, p$ is one k-th root, where μ is the multiplicative inverse of k modulo $(p-1)/k$.

The proofs of theorem 1 and 2 are referred to reference [9].

The solution obtained by theorem 1 and theorem 2 is called the trivial solution to the congruence $x^k \equiv c \,(\%\, p)$, namely that solution which may be written as c to a certain power modulo p.

3.5 The Identity Verification Algorithm

Assume that $(\{C_i\}, \alpha, \beta, \gamma)$ is the public key, F is the file, and (Q, U) is a signature on it.

(1) Let $H \leftarrow hash(F)$, whose binary form is $b_1 \ldots b_n$.

(2) Compute $\hat{\boldsymbol{G}} \leftarrow \prod_{i=1}^{n} C_i^{b_i} \,\%\, M$.

(3) Compute $X \leftarrow (\alpha Q^{-1})^{QUT} \alpha^{Q^n T} \,\%\, M$, $Y \leftarrow (\hat{\boldsymbol{G}}^{QT} U^{-1})^{US} \beta^{UHT} \gamma^{H^n T} \,\%\, M$.

(4) If $X \equiv Y$, the identity is valid and F intact,

otherwise the identity is invalid or F already modified.

By running this algorithm, a verifier can judge whether the signature is genuine or fake, prevent the signatory from denying the signature, and do an attacker from modifying the file.

The discriminant $X \equiv Y \,(\%\, M)$ at (4) is explained as follows:

It is known from sect. 3.1 that $\alpha \equiv \delta^{\delta^n} \equiv \delta (W G_0 G_1)^S \,(\%\, M)$, $\beta \equiv \delta^{(\delta+1) WS} \,(\%\, M)$, and $\gamma \equiv \delta^{W^n} \,(\%\, M)$.

Let $V \equiv (R^{-1} W G_1)^{QU} \delta^{\lambda} \,(\%\, M)$. Because$\lambda$ meets $\lambda S \equiv ((\delta+1) SU + \sum_{i=0}^{n-1} (\delta Q)^{n-1-i} (WH)^i)(\delta Q - WH) \,(\%\, -1)$, may let $\lambda = k đ Đ$, where k is a certain integer, and then

$$\begin{aligned} Q^{QU} V^S &\equiv (R G_0)^{SQU} \delta^{QU} (R^{-1} W G_1)^{QUS} \delta^{\lambda S} \\ &\equiv (W G_0 G_1)^{QUS} \delta^{QU} \delta^{\lambda S} \\ &\equiv \alpha^{QU} \delta^{((\delta+1) SU + \Sigma (\delta Q)^{n-1-i} (WH)^i)(\delta Q - WH)} \end{aligned}$$

$$\equiv \alpha^{QU}\delta^{-(\delta+1)WHSU}\delta^{\delta(\delta+1)QUS}\delta^{(\delta Q)^n-(WH)^n}$$
$$\equiv \alpha^{QU}\beta^{-UH}\delta^{\delta(\delta+1)QUS}\alpha^{Q^n}\gamma^{-H^n}\ (\%\ M).$$

Transposition yields $V^S \equiv (\alpha Q^{-1})^{QU}\alpha^{Q^n}\beta^{-UH}\gamma^{-H^n}\delta^{\delta(\delta+1)QUS}\ (\%\ M)$. Therefore, we have

$$V^{ST} \equiv (\alpha Q^{-1})^{QUT}\alpha^{Q^nT}\beta^{-UHT}\gamma^{-H^nT}\delta^{\delta(\delta+1)QUST}$$
$$\equiv X\beta^{-UHT}\gamma^{-H^nT}\delta^{\delta(\delta+1)QUST}\ (\%\ M).$$

In addition,

$$U^U V^T \equiv (R\,W^{k_1-1}\delta^{\delta(\delta+1)})^{QUT}(R^{-1}WG_1)^{QUT}\delta^{\lambda T}$$
$$\equiv (W^{k_1}G_1)^{QUT}\delta^{\delta(\delta+1)QUT}\delta^{\lambda T}$$
$$\equiv \hat{\boldsymbol{G}}^{QUT}\delta^{\delta(\delta+1)QUT}\delta^{k\,đĐT}$$
$$\equiv \hat{\boldsymbol{G}}^{QUT}\delta^{\delta(\delta+1)QUT}\ (\%\ M).$$

Transposition yields $V^T \equiv (\hat{\boldsymbol{G}}^{QT}U^{-1})^U\delta^{\delta(\delta+1)QUT}\ (\%\ M)$. Hence

$$V^{ST} \equiv (\hat{\boldsymbol{G}}^{QT}U^{-1})^{US}\delta^{\delta(\delta+1)QUST}\ (\%\ M).$$

By the double congruence theorem, there is

$$V^{ST} \equiv X\beta^{-UHT}\gamma^{-H^nT}\delta^{\delta(\delta+1)QUST} \equiv (\hat{\boldsymbol{G}}^{QT}U^{-1})^{US}\delta^{\delta(\delta+1)QUST}.$$

Making transposition and counteraction gives $X \equiv (\hat{\boldsymbol{G}}^{QT}U^{-1})^{US}\beta^{UHT}\gamma^{H^nT} \equiv Y\ (\%\ M)$.

Namely, $X \equiv Y\ (\%\ M)$.

3.6 The Double Congruence Theorem

Theorem 3 (***The Double Congruence Theorem***) Assume that p is a prime, and that s, t satisfying $\langle s, t\rangle = 1$ are two constants, then simultaneous equations

$$\begin{cases} x^s \equiv a\ (\%\ p) \\ x^t \equiv b\ (\%\ p) \end{cases}$$

have the unique solution if and only if $a^t \equiv b^s\ (\%\ p)$.

What follows is the proof of theorem 3.

Necessity: Assume that the simultaneous equations $x^s \equiv a\ (\%\ p)$ and $x^t \equiv b\ (\%\ p)$ have solutions.

Let x_0 be a solution to the two equations, then ${x_0}^s \equiv a\ (\%\ p)$ and ${x_0}^t \equiv b\ (\%\ p)$.

Further, ${x_0}^{st} \equiv a^t\ (\%\ p)$ and ${x_0}^{ts} \equiv b^s\ (\%\ p)$ can be obtained.

Therefore, ${x_0}^{st} \equiv a^t \equiv b^s\ (\%\ p)$.

Sufficiency: Assume that $a^t \equiv b^s\ (\%\ p)$.

According to the greatest common divisor theorem [8], there exists a pair of integers u and v making $us + vt = 1$. Thus,

$$x^{us} \equiv a^u\ (\%\ p),$$
$$x^{vt} \equiv b^v\ (\%\ p).$$

The above two equations multiplying yields

$$x^{us+vt} \equiv x \equiv a^u\ b^v\ (\%\ p).$$

Furthermore, we have

$$(a^u b^v)^s \equiv a^{us}b^{vs} \equiv a^{us}a^{vt} \equiv a^{us+vt} \equiv a,$$
$$(a^u b^v)^t \equiv a^{ut}b^{vt} \equiv b^{us}b^{vt} \equiv b^{us+vt} \equiv b.$$

Accordingly, $a^u\ b^v$ is a solution to the original simultaneous equations.

Uniqueness: Let $x_0 \equiv a^u b^v\ (\%\ p)$.

Assume that another value x_1 meets the equations $x^s \equiv a\ (\%\ p)$ and $x^t \equiv b\ (\%\ p)$ at one time.

Then, it holds that

$$x_1^{\,s} \equiv a \ (\%\ p), \text{ and } x_1^{\,t} \equiv b \ (\%\ p).$$

By comparison, we have $x_1^{\,s} \equiv x_0^{\,s}$ and $x_1^{\,t} \equiv x_0^{\,t}$ (% p). Transposing gives

$$(x_0 x_1^{-1})^s \equiv 1 \text{ and } (x_0 x_1^{-1})^t \equiv 1 \ (\%\ p).$$

If at least one between s and t is relatively prime to $p - 1$, by theorem 1, there must be $x_0 x_1^{-1} \equiv 1$ (% p), namely $x_0 \equiv x_1$ (% p).

If neither s nor t is relatively prime to $p - 1$, let $k = \langle s, p - 1\rangle$, $l = \langle t, p - 1\rangle$. Then we see $\langle s / k, p - 1\rangle = 1$ and $\langle t / l, p - 1\rangle = 1$. Thus, there are $(x_0 x_1^{-1})^k \equiv 1$ and $(x_0 x_1^{-1})^l \equiv 1$. It is known from $\langle s, t\rangle = 1$ that $\langle k, l\rangle = 1$. In terms of the group theory [10], when $\langle k, l\rangle = 1$, only the element '1' belongs to two different sub-group at same time. Therefore, $x_0 x_1^{-1} \equiv 1$, namely $x_1 = x_0$, and x_0 bears uniqueness.

To sum up, we prove theorem 3.

3.7 Characteristics of REESSE1+

REESSE1+ owes the following characteristics compared with classical MH, RSA and ElGamal cryptosystems.

- The security of REESSE1+ is not based on a single hardness, but on multiple hardnesses: the modular subset product problem, multivariate arrangement problem, and super logarithm problem. Thus, it is a multiproblem public-key cryptosystem.
- The key transform $C_i \equiv A_i W^{\ell(i)}$ (% M) is a compound function, and contains three independent variables, that is, n equations contain $2n + 1$ unknown variables. Hence, REESSE1+ is also a multivariate cryptosystem.
- If any of A_i, W and $\ell(i)$ is determined, the relation between the two remainders is still nonlinear ─ thus there is very complicated nonlinear relations among A_i, W and $\ell(i)$.
- There is indeterminacy of $\ell(i)$. On condition that C_i and W are determined, A_i and $\ell(i)$ can not be determined, and even have no one-to-one relation when W is a non-generator. On condition that C_i and A_i are determined, W and $\ell(i)$ can not be determined, and also have no one-to-one relation for $(\ell(i), M - 1) > 1$. This is the radical reason that the continued fraction analysis method is ineffectual.
- There is insufficiency of the key mapping. A private key in REESSE1+ includes $\{A_i\}$, $\{\ell(i)\}$, W and δ four main parts, but there is only a dominant mapping from $\{A_i\}$ to $\{C_i\}$. Thereby, the reversibility of the function is not obvious, and inferring a private key is intractable through mathematical methods.
- Since the length and the elements of the set Ω are not fixed, and different combinations among variables may bring in different hardnesses, REESSE1+ is a sort of flexible public key cryptosystem.

3.8 Correctness of the Decryption Algorithm

Because $(\mathbb{Z}_M^*, \cdot)$ is an Abelian, namely commutative group, $\forall k \in [1, M)$, there is

$$W^k (W^{-1})^k \equiv W^k W^{-k} \equiv 1 \ (\%\ M).$$

Let $b_1 \ldots b_n$ be an n-bit plaintext, and $k = (\sum_{i=1}^{n} \ell(i) b_i) \delta^{-1}$ % –1.

Note that due to $\langle \delta, M - 1\rangle = 1$, there exists δ^{-1} % –1.

We need to prove that $\hat{G}(W^{-\delta})^k \equiv G$ (% M).

According to sect. 3.2, $\hat{G} \equiv \prod_{i=1}^{n} C_i^{b_i}$ (% M), where $C_i \equiv A_i W^{\ell(i)}$ (% M), hence

$$\begin{aligned}
\hat{G}(W^{-\delta})^k &\equiv \prod_{i=1}^{n} C_i^{b_i} (W^{-\delta})^k \equiv \prod_{i=1}^{n} (A_i W^{\ell(i)})^{b_i} (W^{-\delta})^k \\
&\equiv \prod_{i=1}^{n} A_i^{b_i} (W^{\sum (\ell(i) b_i)})(W^{-\delta})^k \\
&\equiv \prod_{i=1}^{n} A_i^{b_i} (W^{\delta})^k (W^{-\delta})^k \\
&\equiv \prod_{i=1}^{n} A_i^{b_i} \equiv G \ (\%\ M).
\end{aligned}$$

The above process gives out a method for seeking G.

Note that in practice, the plaintext $b_1 \ldots b_n$ is unknowable in advance, so we have no way to directly compute k. However, because the range of $k \in [5, \delta^{-1}\sum_{i=1}^{n} \ell(i)]$ is very narrow, we may search k heuristically by multiplying $(W^{-1})^{\delta}$ % M, and verify whether G is equal to 1 after it is divided exactly by some items of $\{A_i\}$. It is known from sect. 3.3 that the original plaintext $b_1 \ldots b_n$ is acquired at the same time the condition $G = 1$ is satisfied.

3.9 Uniqueness of a Plaintext Solution to a Ciphertext

Because $\{C_i\}$ is one non-coprime sequence, the mapping from the subsequence $\{C_{x_1}, \ldots, C_{x_m}\}$ to the product $\hat{\boldsymbol{G}}$ is theoretically many-to-one. It might possibly result in the nonuniqueness of the plaintext solution $b_1 \ldots b_n$ when $\hat{\boldsymbol{G}}$ is being unveiled.

Suppose that the ciphertext $\hat{\boldsymbol{G}}$ can be obtained from two different subsequence products, that is,

$$\hat{\boldsymbol{G}} \equiv C_{x_1} \ldots C_{x_m} \equiv C_{y_1} \ldots C_{y_h} \ (\% M).$$

Then,

$$(A_{x_1} \ldots A_{x_m}) W^{k_1} \equiv (A_{y_1} \ldots A_{y_h}) W^{k_2} \ (\% M),$$

where $k_1 = \ell(x_1) + \ldots + \ell(x_m)$, and $k_2 = \ell(y_1) + \ldots + \ell(y_h)$.

Without loss of generality, let $k_1 \geq k_2$. Because $(\mathbb{Z}_M^*, \cdot)$ is an Abelian group, there is

$$W^{k_1 - k_2} \equiv (A_{y_1} \ldots A_{y_h})(A_{x_1} \ldots A_{x_m})^{-1} \ (\% M),$$

which is written shortly as $W^{k_1 - k_2} \equiv \prod_{i=1}^{h} A_{y_i} \prod_{j=1}^{m} A_{x_j}^{-1} \ (\% M)$.

Let $\theta \equiv W^{k_1 - k_2} \equiv (W^{\delta})^{(k_1 - k_2)\,\delta^{-1}} \ (\% M)$.

This formula means when the plaintext $b_1 \ldots b_n$ is not unique, the value of W^{δ} must be relevant to θ. The contrapositive assertion equivalent to it is that if the value of W^{δ} is irrelevant to θ, $b_1 \ldots b_n$ will be unique. Thus, we need to consider the probability that W^{δ} takes a value relevant to θ.

If an adversary tries to attack an 80-bit symmetric key through the exhaustive search, and a computer can verify trillion values per second, it will take 38334 years for the adversary to verify up all the potential values. Hence, currently 80 bits are quite enough for the security of a symmetric key.

When the length n of a key sequence is equal to 80, the number of the values containing the repeated in the form $\prod_{i=1}^{h} A_{y_i} \prod_{j=1}^{m} A_{x_j}^{-1}$ is at most $3^{80} \approx 2^{1.585 \times 80} = 2^{127}$. Because $A_1^{-1} \ldots A_n^{-1}$ are not necessarily coprime, the value of $\prod_{i=1}^{h} A_{y_i} \prod_{j=1}^{m} A_{x_j}^{-1}$ may possibly occur repeatedly.

On the other hand, the first 80 primes may constitute a coprime sequence with a minimum sequence product, which makes it possible that the modulus M is roughly equal to 2^{552} or M roughly equal to 2^{384} with binary compact sequence optimization. Therefore, when $n \geq 80$, the probability that W^{δ} takes values relevant to θ is less than $(2^{13}2^{127}) / 2^{384} = 1 / 2^{244}$, where 2^{13} is greater than or roughly equal to $(k_1 - k_2)\delta^{-1}$, namely the number of W^{δ} meeting $(W^{\delta})^{(k_1 - k_2)\delta^{-1}} \equiv \prod_{i=1}^{h} A_{y_i} \prod_{j=1}^{m} A_{x_j}^{-1} \ (\% M)$ is at most 2^{13}. Clearly, it is almost zero. This probability will further decrease when W^{δ} is prime.

The above analysis shows that the probability that the plaintext solution $b_1 \ldots b_n$ is not unique is almost zero; thus the decryption algorithm can always recover the original plaintext from the ciphertext $\hat{\boldsymbol{G}}$, which is also verified by the program in C language.

4 Securities of Encryption and Decryption

4.1 Extracting a Private Key from a Public Key Being the Multivariate Arrangement Hardness

A public key may be regarded as the special cipher of a related private key. Since a ciphertext is the effect of a public key and a plaintext, the ciphertext has no direct help to inferring the private key.

In the REESSE1+ system, the key transform is $C_i \equiv A_i W^{\ell(i)} \ (\% M)$, and $\ell(i) \in \{i\,\delta\,(\% -1) \mid i = 5, \ldots, n + 4\}$.

For a specific C_i, assume that the corresponding A_i and W are revealed under some extreme condition. Due to $\ell(i) \in (1, M)$, obviously, by $W^{\ell(i)} \equiv C_i A_i^{-1}$ (% M) seeking $\ell(i)$ is the DLP. Thus, under normal situations, inferring a related private key from a public key is harder than the DLP.

In what follows, we discuss the case the n items of $\{C_i\}$ are considered all together.

If an attacker tries to extract a related private key $\{A_i\}$ from a public key, it is equivalent to solve the simultaneous equations

$$\begin{cases} C_1 \equiv A_1 W^{\ell(1)} \ (\% M) \\ C_2 \equiv A_2 W^{\ell(2)} \ (\% M) \\ \quad \cdots\cdots \\ C_n \equiv A_n W^{\ell(n)} \ (\% M). \end{cases}$$

The above equation system contains n known variables, and $2n + 1$ unknown variables.

Assume that the $\Phi > n$ is the largest prime constant in the REESSE1+ cryptosystem, then each $A_i \in \Gamma = \{2, \ldots, \Phi\}$, where Γ contains at least n primes. Let $\tilde{N}$ be the number of the potential coprime sequences in the interval $[2, \Phi]$, then $\tilde{N} > \mathrm{A}_n^n = n!$.

If let $\ell(1)\delta^{-1} = \ldots = \ell(n)\delta^{-1} = 5$, and each A_i traverse Γ, then can obtain theoretically $5n\Phi$ values where exists the true value of W^{δ}. Therefore, the number of potential values of W^{δ} will decrease to $5n\Phi$. Note that in fact computing all five W^{δ} from $(W^{\delta})^5 = C_i A_i^{-1}$ (% M) is intractable when $5 \mid (M - 1)$ and $(M - 1) / 5 > 2^{80}$ (see sect. 5.1).

Suppose that the attacker guesses the sequences $\{A_i\}$ and $\{\ell(i)\delta^{-1}\}$, then figures out the n values of W^{δ} in $\mathrm{O}(T_W)$ time. If these values are all identical, the guessing is thought right. Notice that for i-th equation, A_i is allowed to take any element of Γ as long as it is pairwise coprime to $A_1, \ldots, A_{i-1}$, and $\ell(i)\delta^{-1}$ is allowed to take any element of $\{5, \ldots, n + 4\}$ as long as it is pairwise different from $\ell(1), \ldots, \ell(i - 1)$, which means that guessing $\{A_i\}$ or $\{\ell(i)\delta^{-1}\}$ is an arrangement problem. Thereby, the time complexity of this attack is $\mathrm{O}(\tilde{N}(n!)T_W) > \mathrm{O}(2^n)$.

Suppose that the attacker guesses W^{δ} and the sequences $\{\ell(i)\delta^{-1}\}$, then compute the sequence $\{A_i\}$ in $\mathrm{O}(T_A)$ time. If $\{A_i\}$ is a coprime sequence, the guessing is thought successful. Because guessing $\{\ell(i)\delta^{-1}\}$ is also an arrangement problem, the time complexity of this attack is at least $\mathrm{O}(5n\Phi(n!)T_A) > \mathrm{O}(2^n)$.

Suppose that the attacker guesses W^{δ} and the sequences $\{A_i\}$, then find out the sequence $\{\ell(i)\delta^{-1}\}$ in $\mathrm{O}(T_\ell)$ time. If every $\ell(i)\delta^{-1} \in \{5, \ldots, n + 4\}$ and is pairwise distinct, the guessing is thought successful. Because guessing $\{A_i\}$ is likewise an arrangement problem, the time complexity of this attack is at least $\mathrm{O}(5n\Phi\tilde{N}T_\ell) > \mathrm{O}(5n\Phi(n!)T_\ell) > \mathrm{O}(2^n)$.

Since W, δ and (G_0G_1) are all unknown, it is impossible to infer W, δ or (G_0G_1) from $\alpha \equiv \delta(WG_0G_1)^S$ (% M).

The expressions $\alpha \equiv \delta^{\delta^n}$, $\beta \equiv \delta^{(\delta+1)WS}$ and $\gamma \equiv \delta^{W^n}$ (% M) only contain two unknown variables, but the time complexity of finding δ and W will be at least $\mathrm{O}(2^n)$ (see sect. 5.2.3).

Further, we can argue that the time complexity of the continued fraction attack is $\mathrm{O}(n!) > \mathrm{O}(2^n)$, for which we have a curt explanation as follows:

Theorem 4 If α is an irrational number and if r / s is a rational number in lowest terms, where r and s are integers with $s > 0$ such that $\mathrm{abs}(\alpha - r / s) < 1 / (2s^2)$, then r / s is a convergent of the simple continued fraction expansion of α [11].

Reference [11] does not offer a similar theorem with α being a rational number. Therefore, when α is a rational number, theorem 4 does not necessarily hold.

Additionally, due to the indeterminacy of $\ell(i)$, which indicates there exists $W^{\ell(k)} \equiv W^{\ell(k)+\tau}$ (% M) or $W^{\ell(k)} \equiv A_m W^{\ell(h)}$ (% M), $\ell(i) + \ell(j) = \ell(k)$ can not be determined in polynomial time, that is to say, determining $\ell(i) + \ell(j) = \ell(k)$ is unmeaning.

In summary, the time complexity of inferring a related private key from a public key is at least $\mathrm{O}(2^n)$.

4.2 Recovering a Plaintext from a Ciphertext and a Public Key Being the Modular Subset Product Hardness

In terms of sect. 3.2, the ciphertext is a modular subset product, namely $\hat{\boldsymbol{G}} \equiv \prod_{i=1}^{n} C_i^{b_i} \ (\% \ M)$, where $b_1 \ldots b_n$ is a plaintext block or a symmetric key, and $\{C_1, \ldots, C_n\}$ is a public key.

Obviously, $\prod_{i=1}^{n} C_i^{b_i} = LM + \hat{\boldsymbol{G}}$. Due to $L \in [1, M-1]$, deriving $\prod_{i=1}^{n} C_i^{b_i}$ from $\hat{\boldsymbol{G}}$ is infeasible, which indicates inferring $b_1 \ldots b_n$ from $\hat{\boldsymbol{G}}$ is not a factorization problem.

Observe an extreme case. Assume that $C_1 = \ldots = C_n = C$, then $\hat{\boldsymbol{G}} \equiv \prod_{i=1}^{n} C^{b_i} \ (\% \ M)$. It can be written as

$$\hat{\boldsymbol{G}} \equiv C^{\sum b_i} \ (\% \ M),$$

where i is from 1 to n.

Because we need not only to figure out the value of $\sum_{i=1}^{n} b_i$ but also to find out the position of every $b_i = 1$, we express equivalently the sum $\sum_{i=1}^{n} b_i$ as $\sum_{i=1}^{n} b_i 2^{i-1}$, and let $x = \sum_{i=1}^{n} b_i 2^{i-1}$. Correspondingly,

$$\hat{\boldsymbol{G}} \equiv C^{x} \ (\% \ M).$$

The above manifests that seeking the exponent x of C is the DLP.

The above process is reversible. It shows that if the plaintext recovery problem can be solved, the DLP can be solved. Therefore, when $C_1 \neq \ldots \neq C_n$, attempting to recover a related plaintext from a known ciphertext and public key is more intractable than the DLP, which is essentially different from the subset sum problem or the knapsack problem.

On the other hand, there exists the exhaustive search attack on $b_1 \ldots b_n$, and clearly, the time complexity of this attack is $O(2^n)$.

Notice that when $\hat{\boldsymbol{G}}$ is converted into the coprime sequence product G, computing the plaintext $b_1 \ldots b_n$ is tractable. Namely there is a trapdoor for $\hat{\boldsymbol{G}}$. Hence, the plaintext security of REESSE1+ is built on a trapdoor function such that computing a subset product from subset elements is tractable while seeking the involved elements from the subset product is intractable.

4.3 Avoiding the Adaptive-chosen-ciphertext Attack

Absolute most of public key cryptographies will probably be faced by the adaptive-chosen-ciphertext attack apart from ElGamal, ECC and so on, although this attack method still stays on a concept level, and is not implemented in the concrete applications.

It is lucky that REESSE1+ can avoid the adaptive-chosen- ciphertext attack. In the REESSE1+ cryptosystem, a public key is a sequence, and the number of its potential arrangements is $n!$, and thus a secret time function may be chosen. Every time encryption is done, a different arrangement of the public key according to the time is employed to encrypt a plaintext. In this way, even if the plaintexts encrypted are the same, due to different encrypting times, the related ciphertexts are likely unequal. Of course, should let those C_i to which values of the plaintext bits corresponding are all 1 permute as much as possible. So and so only can the multiple ciphertexts of the identical plaintext be different from one another.

Another approach to avoiding the above attack is to append a stochastic fixed-length binary sequence to the terminal of every plaintext block when it is encrypted.

5 Securities of Digital Signature and Identity Verification

5.1 Extracting a Related Private Key from a Signature Being a Hardness

Assume that p is a prime, and $k \mid (p-1)$. In terms of the probabilistic algorithm in section 1.6 of reference [12], the time complexity of finding out a random solution to $x^k \equiv c \ (\% \ p)$ is at least $\max(O(2^{k-1}), O(p/k))$. Thus, when $k > 80$ or $p/k > 2^{80}$, this algorithm is ineffectual currently.

However, when $\langle k, p-1 \rangle = 1$ or $\langle k, (p-1)/k \rangle = 1$ with $k \mid (p-1)$, the trivial solution to $x^k \equiv c \ (\% \ p)$ can be acquired in terms of theorem 1 and 2.

It is known from the digital signature algorithm that

$$Q \equiv (R G_0)^S \delta (\% M), \text{ and } U \equiv (R W^{k_1-1} \delta^{\delta(\delta+1)})^{QT} (\% M).$$

When an attacker wants to seek $R G_0$, or $R W^{k_1-1} \delta^{\delta(\delta+1)}$, it is equivalent to solving the congruences

$$x^S \equiv Q\delta^{-1} (\% M), \text{ and } x^{QT} \equiv U (\% M).$$

For the first equation, because δ is unknown, and the right of the equation is not a constant, it is impossible to solve the equation for $R G_0$. If δ is guessed, the probability of hitting δ is $1 / |\delta| < 1 / 2^n$.

For the second equation, due to $T \mid (M-1)$, there is $\langle QT, M-1 \rangle \geq T$. If there exists the trivial solution to the equation, the probability that it is just the specific solution $(R W^{k_1-1} \delta^{\delta(\delta+1)})$ is equal to or less than $1 / T \leq 1/2^n$. Even though $(R W^{k_1-1} \delta^{\delta(\delta+1)})$ is found out, due to the randomicity of $R \in (1, M-1)$, neither of W^{k_1-1} and $\delta^{\delta(\delta+1)}$ can be determined.

Therefore, the time complexity of extracting a related private key from a signature is at least $O(2^n)$.

5.2 Faking a Digital Signature only through a Public Key Being a Hardness

Assume H is the hash value of F, (Q, U) is a signature on F, then the condition discriminant

$$(\alpha Q^{-1})^{QUT} \alpha^{Q^n T} \equiv (\hat{G}^{QT} U^{-1})^{US} \beta^{UHT} \gamma^{H^n T} (\% M)$$

holds.

Due to an equation with the two variables, the value of a variable may be supposed by an attacker. However, supposing a value and seeking the other is the super logarithm problem.

5.2.1 The Super Logarithm Problem

Assume that $g \in \mathbb{Z}_p^*$ is a generator, where p is prime, then $\{y \mid y \equiv g^x (\% p), x = 1, \ldots, p-1\} = \mathbb{Z}_p^*$ [10].

Assume that k satisfying $(k, p-1) = 1$ is an integer, then also $\{y \mid y \equiv x^k (\% p), x = 1, \ldots, p-1\} = \mathbb{Z}_p^*$ [9].

Namely, $\forall x \in [1, p-1]$, $y \equiv g^x (\% p)$ or $y \equiv x^k (\% p)$ with $(k, p-1) = 1$ is a self-isomorph of the group $\mathbb{Z}_p^*$.

However, for the x^x operation, $\{y \mid y \equiv x^x (\% p), x = 1, \ldots, p-1\} = \mathbb{Z}_p^*$ does not hold, that is,

$$\{y \mid y \equiv x^x (\% p), \ x = 1, \ldots, p-1\} \neq \mathbb{Z}_p^*.$$

For example, when $p = 11$, $\{y \mid y \equiv x^x (\% p), x = 1, \ldots, p-1\} = \{1, 3, 4, 5, 6\}$, where $3^3 \equiv 6^6 \equiv 8^8 \equiv 5 (\% 11)$.

When $p = 13$, $\{y \mid y \equiv x^x (\% p), x = 1, \ldots, p-1\} = \{1, 3, 4, 5, 6, 9, 12\}$, where $7^7 \equiv 11^{11} \equiv 6 (\% 13)$, and $1^1 \equiv 3^3 \equiv 8^8 \equiv 9^9 \equiv 12^{12} \equiv 1 (\% 13)$.

When $p = 17$, $\{y \mid y \equiv x^x (\% p), x = 1, \ldots, p-1\} = \{1, 2, 4, 8, 9, 10, 12, 13, 14\}$, where $2^2 \equiv 12^{12} \equiv 4 (\% 17)$, $6^6 \equiv 15^{15} \equiv 2 (\% 17)$, and $10^{10} \equiv 14^{14} \equiv 2 (\% 17)$.

The above examples illustrate that $\{y \equiv x^x (\% p) \mid x = 1, \ldots, p-1\}$ cannot construct a complete set for a group. Furthermore, mapping from x to y is one-to-one sometimes, and many-to-one sometimes. That is, inferring x from y is indeterminate, x is non-unique, and even inexistent. Thus, x^x has extremely strong irregularity, and is essentially distinct from g^x and x^k.

We consider two functions over the real set $\mathbb{R}$: $y = f(x) = g^x$, and $y = \varphi(x) = x^x$, where $x > 0$.

Their inverse functions, the derivations of which are omitted, are respectively

$$x = f^{-1}(y) = \log_g y,$$

$$\text{and } x = \varphi^{-1}(y) = y \log_g y / ((y' - y) \log_g e),$$

where $y' = x^x (1 + \log_g x / \log_g e)$ denotes a derivative, and the constant $e = 2.7182818\ldots$.

Further, $\varphi^{-1}(y) = y f^{-1}(y) / ((y' - y) \log_g e)$.

Assume that we already know y_0 satisfying $y_0 = x_0^{x_0}$ and $y_0 = g^{x_0}$. Apparently, if $\varphi^{-1}(y_0)$, namely x_0 can be found out, $f^{-1}(y_0)$, namely x_0, can be figured out. This means that the time complexity of computing $f^{-1}(y)$ is less than or equal to that of computing $\varphi^{-1}(y)$.

Contrariwise, if $f^{-1}(y_0)$, namely x_0, can be found out, $\varphi^{-1}(y_0)$, namely x_0 can not be figured out since y_0' is the function of x_0, and has no solution in polynomial time. This means that the time complexity of computing $f^{-1}(y)$ is not equal to that of computing $\varphi^{-1}(y)$.

To sum up, the time complexity of seeking $\varphi^{-1}(y)$ is greater than that of seeking $f^{-1}(y)$.

Similarly, this fact should hold in the finite field $\mathbb{GF}(p)$, because the discreteness of a finite field does not weaken the computational complexity of the identical problem over a continuous interval. For instance, computing $x = f^{-1}(y) = \log_g y$ is easy in a continuous interval while hard in a finite field.

In summary, we think that the $x^x \equiv c \ (\% \ p)$ problem is harder than the $g^x \equiv c \ (\% \ p)$ problem. Hence, the former is called the super logarithm hardness. It is emphasized that the super logarithm hardness is more suitable for doing signature since it owns non-uniqueness.

Note that to attempt to solve the super logarithm problem in light of the Chinese Remainder Theorem is specious.

At present there is no better method for seeking a super logarithm than the exhaustive search, and thus the time complexity of the solution to $x^x \equiv c \ (\% \ p)$ may be expected to be $O(p) > O(2^n)$, where n is the length of a message digest.

5.2.2 Faking a Signature by the Verification Algorithm Being the Super Logarithm Problem

Assume that F is an arbitrary file, and H is its hash output. In terms of the discriminant

$$(\alpha Q^{-1})^{QUT} \alpha^{Q^n T} \equiv (\hat{\boldsymbol{G}}^{QT} U^{-1})^{US} \beta^{UHT} \gamma^{H^n T} \ (\% \ M),$$

an attacker may suppose the value of a variable.

If suppose the value of Q, no matter whether U exists or not, seeking U is the super logarithm hardness.

Similarly, if suppose the value of U, seeking Q is the super logarithm hardness.

If the attacker hits the small $đ$, obtains $Đ$ by factorizing $M - 1$, and raises either side of the discriminant to the power of $đ$, then when $Đ \mid (\delta Q - WH)$, there is

$$(\alpha Q^{-1})^{đQUT} \equiv (\hat{\boldsymbol{G}}^{QT} U^{-1})^{đUS} \beta^{đUHT} \ (\% \ M).$$

Further,

$$(\alpha Q^{-1})^{đQT} \equiv (\hat{\boldsymbol{G}}^{QT} U^{-1})^{đS} \beta^{đHT} \ (\% \ M).$$

Hereat, if suppose the value of Q which is unknown, U may possibly be worked out in polynomial time. However, Q and U must satisfy both the discriminant and the condition $Đ \mid (\delta Q - WH)$, which is impossible since both δ and W are unknown.

5.2.3 Faking a Signature by the Signature Algorithm Being the Exponential Time Problem

Due to $Q \equiv (R G_0)^S \delta$, $U \equiv (R W^{k_1 - 1} \delta^{\delta(\delta+1)})^{QT}$ and $V \equiv (R^{-1} W G_1)^{QU} \delta^{\lambda} \ (\% \ M)$, an attacker may attempt the following attack method.

Let $a \, c \equiv (\alpha \delta^{-1})^{1/S}$, and $b \, c \equiv \hat{\boldsymbol{G}} \ (\% \ M)$. When δ is sought, c and b may be figured out with supposing the value of a.

Let $Q \equiv a^S \delta$, $U \equiv b^{QT} \delta^{\delta(\delta+1)QT}$, and $V \equiv c^{QU} \delta^{\lambda} \ (\% \ M)$, where λ meets

$$\lambda S \equiv ((\delta+1) SU + \textstyle\sum_{i=0}^{n-1} (\delta Q)^{n-1-i} (HW)^i)(\delta Q - WH) \ (\% -1),$$

and Q, U make $Đ \mid (\delta Q - WH)$ as well as $đ \mid ((\delta+1) SU + \sum_{i=0}^{n-1} (\delta Q)^{n-1-i} (HW)^i) \ (\% -1)$. If the attacker seeks δ and W such taht

$$\begin{cases} \alpha \equiv \delta^{\delta^n} \ (\% \ M) \\ \beta \equiv \delta^{(\delta+1)WS} \ (\% \ M) \\ \gamma \equiv \delta^{W^n} \ (\% \ M), \end{cases} \tag{1*}$$

then the attack is likely successful.

Can δ and W be sought? If δ exists, there is

$$\begin{cases} \alpha^{W^n} \equiv \gamma^{\delta^n} \ (\%\ M) \\ \beta^{W^n} \equiv \gamma^{(\delta+1)WS} \quad (\%\ M). \end{cases} \tag{2*}$$

Note that because possibly $\langle W^n, \delta^n \rangle > 1$, (2*) is not a sufficient condition for (1*) to have solutions.

Let g be a generator of $(\mathbb{Z}_M^*, \cdot)$, by means of Index-calculus method for discrete logarithms [3], figure out q, u, and v such that $g^q \equiv \alpha$, $g^u \equiv \beta$, and $g^v \equiv \gamma\ (\%\ M)$, obtain

$$\begin{cases} g^{qW^n} \equiv g^{v\delta^n} \ (\%\ M) \\ g^{uW^n} \equiv g^{v(\delta+1)WS} \quad (\%\ M). \end{cases}$$

Accordingly,

$$\begin{cases} qW^n \equiv v\delta^n \ (\%\ -1) \\ uW^n \equiv v(\delta+1)WS \ (\%\ -1). \end{cases}$$

If $\langle q, M-1 \rangle \mid v$, and the trivial solution to $x^n \equiv (q^{-1}v)\delta^n \ (\%\ -1)$ exists, then $W \equiv (q^{-1}v)^{n^{-1}}\delta \ (\%\ -1)$, where computing n^{-1} is referred to theorem 1, theorem 2 and reference [8]. Further, due to $W = 0$ is not a solution to the equations, there is

$$u(q^{-1}v)^{(n-1)n^{-1}}\delta^{n-1} \equiv v(\delta+1)S \ (\%\ -1).$$

Transparently, the above equation is a true polynomial of degree $n-1$, which can not be solved via discrete logarithms. If Euler phi function $\varphi(M-1)$ is worked out, the attacker may seem to resort to the probabilistic algorithm in section 1.6 of reference [12]. However, the running time of this algorithm is at least $\max(O(2^{n-2}), O(\varphi(M-1)/(n-1)))$. When $n > 80$ or $\varphi(M-1)/(n-1) > 2^{80}$, seeking δ is infeasible at present. Furthermore, when $\langle u(q^{-1}v)^{(n-1)n^{-1}}, M-1 \rangle \nmid v$, doing division of polynomials is infeasible.

5.3 Faking a Signature through Known Signatures with a Public Key Being a Hardness

Given the file F and a signature (Q, U) on it, and assume that there exists another file F' with corresponding H' and $\hat{G}'$. Then, if an arbitrary (Q', U') satisfies

$$(\alpha Q'^{-1})^{Q'U'T}\alpha^{Q'^nT} \equiv (\hat{G}'^{Q'T}U'^{-1})^{U'S}\beta^{U'H'T}\gamma^{H'^nT} \ (\%\ M),$$

it is a signature fraud on F'.

The values of Q and U are allowed to be utilized separately.

If let $Q' = Q$, Q' does not necessarily satisfy $D \mid (\delta Q' - WH')$, and computing U' is intractable.

If let $U' = U$, no matter whether the preceding equation has solutions or not, seeking Q' is the super logarithm hardness.

If the two signatures (Q_1, U_1) and (Q_2, U_2) on the files F_1 and F_2 are obtained, due to $D \mid (\delta Q_1 - WH_1)$ and $D \mid (\delta Q_2 - WH_2)$, we see that $D \mid (\delta(Q_1 + Q_2) - W(H_1 + H_2))$. Let $Q' = Q_1 + Q_2$, $H' = H_1 + H_2$, then $D \mid (\delta Q' - WH')$. However, inferring F' from H' is intractable in terms of the properties of hash functions.

If several of the pair (Q, U) are known, because R involved in U is random, and there is not functional or statistical relation among different U, they are not helpful to solving the super logarithm problem.

Therefore, forging another signature via known signatures with a public key is the super logarithm hardness.

5.4 Faking a Signature through a Chosen-plaintext Being a Hardness

Due to $H = b_1 \ldots b_n$, $\hat{G} \equiv \prod_{i=1}^{n} C_i^{b_i} \ (\%\ M)$, there is not a linear expression relation between H and $\hat{G}$.

It is understood from the discriminant $(\alpha Q^{-1})^{QUT}\alpha^{Q^nT} \equiv (\hat{G}^{QT}U^{-1})^{US}\beta^{UHT}\gamma^{H^nT} \ (\%\ M)$ that

$$\hat{G}^{QUST}\beta^{UHT}\gamma^{H^nT} \equiv (\alpha Q^{-1})^{QUT}\alpha^{Q^nT}U^{US} \ (\%\ M).$$

Assume that Q and U are known and $\hat{G} = f(H)$. Clearly, it is harder than the DLP to seek H according to the above congruence. Also it is harder than the DLP to infer H from $\hat{G}$ (see sect. 4.2).

6 Conclusions

The REESSE1+ cryptosystem may be applied to data encryption and digital signature. If it is only applied to digital signature, the constraint $M > G$ at step 2 in the key generation algorithm may be removed.

If the constraint $M > G$ can not be removed, then when $n \geq 80$, the modulus M will be comparatively large. It will cause both increase in key lengths and, worse than all, decrease in algorithmic speeds. For example, $\log_2 M \geq 553$ (to select the first 80 primes as a coprime sequence make the sign '=' hold) when $n = 80$, $\log_2 M \geq 694$ when $n = 96$, $\log_2 M \geq 840$ when $n = 112$, and $\log_2 M \geq 990$ when $n = 128$. Thus, the theoretical algorithms of REESSE1+ must be optimized.

The basic idea of optimization is to make use of binary compact sequences. That is, the mapping is not regarded as one to one but as three items to two bits between a non-coprime sequence and a symmetric key. (or a plaintext block). The n-bit symmetric key is partitioned evenly into $n/2$ units in order, and every unit has 2 bits and 4 combinations, namely 00, 01, 10, and 11. Let 00 map to 1, and the others do respectively to the three consecutive items of $\{C_i\}$. After optimized, the bit-length of M will decrease to 384 when $n = 80$, and 544 when $n = 112$.

Through the optimization based on the binary compact sequence, the decryption time complexity of the REESSE1+ cryptosystem is reduced to $O(n^2)$ from $O(n^3)$. Through the integration of a coprime sequence and the lever function $\ell(.)$, we exchange the very slight cost of operations for the great security of the REESSE1+ cryptosystem.

Acknowledgment

The authors would like to thank Mulan Liu, Huanguo Zhang, Dengguo Feng, Lequan Min, Xiulin Zheng, Bingru Yang, Zhiying Wang, and Maozhi Xu for their important advice, suggestions, and corrections.

References

[1] R. L. Rivest, A. Shamir and L. M. Adleman, "A Method for Obtaining Digital Signatures and Public-key Cryptosystems," *Communications of the ACM*, 21(2), 1978, pp.120-126.

[2] T. ElGamal, "A Public-key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *IEEE Transactions on Information Theory*, v. 31, n. 4, 1985, pp. 469-472.

[3] A. J. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, London: CRC Press, 1997, ch. 2, 8.

[4] Oded Goldreich, *Foundations of Cryptography: Basic Tools*, Cambridge, UK: Cambridge University Press, 2001, pp. 31–35.

[5] I. F. Blake, G. Seroussi, and N. P. Smart, *Elliptic Curves in Cryptography*, Cambridge, UK: Cambridge University Press, 1999, pp. ch. 1, 6.

[6] R. C. Merkle and M. E. Hellman, "Hiding information and Signatures in Trapdoor Knapsacks," *IEEE Transactions on Information Theory*, v. 24, n. 5, 1978, pp. 525-530.

[7] A. Shamir, "A Polynomial Time Algorithm for Breaking the Basic Merkle-Hellman Cryptosystem," *Proceedings of the 23th IEEE Symposium on the Foundations of Computer Science*, 1982, pp. 145-152.

[8] Song Y. Yan, *Number Theory for Computing*, 2nd ed., Berlin: Springer-Verlag, 2002, ch. 1.

[9] Paul Garrett, *Making, Breaking Codes: An Introduction to Cryptology*, New Jersey: Prentice-Hall, 2001, ch. 12.

[10] Thomas W. Hungerford, *Algebra*, New York: Springer-Verlag, 1998, ch. 1–3.

[11] Kenneth H. Rosen, *Elementary Number Theory and Its Applications* (5th ed.), Boston: Addison-Wesley, 2005, ch. 12.

[12] Henri Cohen, *A Course in Computational Algebraic Number Theory*, Berlin: Springer-Verlag, 2000, ch. 1, 3.

* Refuting the Pseudo Attack on REESSE1+

Shenghui Su [1], and Shuwang Lü [2]

[1] School of Information Engineering, University of Science & Technology Beijing

Beijing 100083, P. R. China

sheenway@126.com

[2] School of Graduate, Chinese Academy of Sciences

Beijing 100039, P. R. China.

March 15 2007

Abstract: This paper illustrates that the condition relevant to theorem 1 in reference [8] does not satisfy necessity, and theorem 1 is only a conjecture tending to be untenable. The paper discusses that the converse proposition of fact 1.1 in [8] does not hold, namely the condition for $f(i) + f(j) = f(k)$ is only necessary but not sufficient, there is a logic error in deducing fact 1.2 causing fact 1.2, fact 1.3 and fact 4 not to hold, and property 1, 2, and 5 are meaningless. The paper argues that the combination of fact 1.1 and fact 1.2 or fact 4 is not sufficient for $f(i) + f(j) = f(k)$ by offering an example, and Alg.1 and Alg.2 based on such theory, facts and properties are wrong logically. For signature fraud, the paper points out that [8] misunderstands that T^{-1} and Q^{-1} mod $(M - 1)$ certainly exist, and modifies improper the verification algorithm. Therefore, the conclusion of [8] that REESSE1+ is not secure at all is fully incorrect.

Keywords: Public key cryptosystem; Security; Lever function; Continued fraction; Sufficient condition

1 Introduction

In 2001, we put forward the REESSE1 public-key encryption scheme [1]. In 2003, we proposed the REESSE1 public-key cryptosystem which is an extension of the old scheme and includes encryption and signature [2]. In May 2005, it was analyzed that the lever function is necessary and sufficient for the security of REESSE1 encryption [3]. In [3], the continued fraction analysis of the REESSE1 encryption was mentioned earlier than other published literatures. In Nov. 2006, an abbreviated version of the REESSE1+ public-key cryptosystem was submitted to eprint.iacr.org [4].

As REESSE1+ points out, $\Omega = \{i\delta \pmod{M-1} \mid i = 5, \ldots, n + 4\}$ is not unique, and users may select other Ω. For example, take $\Omega = \{n + 1, \ldots, n + \mathrm{n}\}$. Here, $\ell(i) + \ell(j) = \ell(k)$ does not hold forever.

In May 2005, [5] set forth that the REESSE1 signature scheme is insecure, which is right. In July 2005, [6] thought rashly that the REESSE1 encryption scheme is insecure, which is wrong, and was refuted thoroughly in [7]. Furthermore, [7] pointed out that the analysis of REESSE1 by continued fractions is not the original idea of [6]. In Dec. 2006, [8] thought indiscreetly that the REESSE1+ public-key cryptosystem is insecure, which is a flubdub and gulf.

Firstly, theorem 1 in [8] is only a conjecture tending to be untenable.

Secondly, the converse proposition of fact 1.1 does not hold, namely the condition for $f(i) + f(j) = f(k)$ is only necessary, but not sufficient.

Thirdly, fact 1.2 and 1.3 both do not hold. Even if fact 1.2, namely Eq.(3) holds, the combination of fact 1.1 and 1.2 is not sufficient for $f(i) + f(j) = f(k)$.

Because fact 1.2, namely Eq.(3) does not hold, fact 4 on page 5 does not hold naturally, that is, $\Delta = (M / (2\prod_{i=n-2}^{m}\mathrm{Prime}[i]))^{1/2}$ is meaningless.

Because the combination of fact 1.1 and either fact1.2 or fact 4 is not sufficient for $f(i) + f(j) = f(k)$, property 1, 2, and 5 are meaningless.

* This appendix can be accessed separately at http://arxiv.org/abs/0704.0492.

Further, Alg.1 and Alg.2 based on such theory, facts and properties are wrong logically.

Moreover, the case of $\{i + \delta \pmod{M-1} \mid i = 5, \ldots, n+4, \delta \geq n-4\}$ with $\ell(i) + \ell(j) \neq \ell(k)$ for all i, j and k is not analyzed at all.

Fourthly, the inverse T^{-1} of T mod $(M-1)$ does not exist, and Q^{-1} mod $(M-1)$ does not necessarily exist.

Fifthly, the identity verification algorithm is modified.

To sum up, the cryptanalysis in [8] is a pseudo attack on the REESSE1+ cryptosystem.

Why is it such? The radical reason is that [8] neglects the indeterminacy of the lever function $\ell(.)$ which is mentioned in [4]:

(i) If W is not a generator, there is $W^{\ell(i)} \equiv W^{\ell(i)+d} \pmod{M}$ with order $d < M-1$. When $\ell(i) + \ell(j) = \ell(k)$, we see that $\ell(i) + d + \ell(j) + d \neq \ell(k) + d$.

(ii) When $\ell(i) + \ell(j) \neq \ell(k)$, there exist $C_i \equiv A_i' W'^{\ell'(i)}$, $C_j \equiv A_j' W'^{\ell'(j)}$, and $C_k \equiv A_k' W'^{\ell'(k)} \pmod{M}$ such that $\ell'(i) + \ell'(j) = \ell'(k)$ with $A_k' \leq P_\eta$, an upper prime allowed.

2 Theorem 1 in [8] Being Only a Conjecture Tending to Be Untenable

Theorem 1 in [8] is retailed as follows:

Theorem 1 *[5] Let α be a* **real** *number, and let r/s be a rational with* $\gcd(r, s)=1$ *and* $|\alpha - r/s| < 1/2s^2$. *Then r / s is a convergent of the continued fraction expansion of α.*

Here, reference *[5]* is namely reference [9] in this paper.

2.1 The Condition in Theorem 1 Not Satisfying Necessity

A counterexample is taken.

For example, let $r / s = 2 / 13$, and then $1 / 2s^2 = 1 / (2 \times 13^2) = 0.002958579882$.

Let $\alpha = 2039 / 13001$, and $2039 / 13001 - 2 / 13 = 0.002987935839 > 0.002958579882 = 1 / (2 \times 13^2)$.

On the other hand, the continued fraction of $2039 / 13001$ is $1 / (6 + (1 / (2 + 1 / (1 + \ldots 1 / 3))))$.

Thus, $2/13$ is a convergent of the continued fraction of $2039 / 13001$, which illustrates $|\alpha - r / s| < 1 / 2s^2$ is not a necessary condition for r / s to be a convergent of the continued fraction of α.

2.2 Theorem 1 Being Unproven in [8] and Unfound in [9]

In [8], theorem 1, that is, the sufficiency of the condition $|\alpha - r / s| < 1 / 2s^2$ is unproven. For displaying the proof, [8] guides readers to page 460 of [9]. However, in [9], The theorem equivalent to theorem 1 is not found from cover to cover.

In [9], accurately speaking, on page 486 of [9], we find out only the theorem below.

Theorem 12.19 If α is an **irrational** number and if r / s is a rational number in lowest terms, where r and s are integers with $s > 0$ such that $|\alpha - r / s| < 1 / (2s^2)$, then r / s is a convergent of the simple continued fraction expansion of α.

Here, 'r / s is in lowest terms' means $\gcd(r, s) = 1$.

Notice that α is an irrational number which can be uniquely expressed by an **infinite** simple continued fraction, but not a rational number which can be expressed by a **finite** simple continued fraction, to say nothing of being a real number. It is well known that real numbers consists of rational numbers and irrational number.

If when α is a rational number, theorem 12.19 also holds, then a similar theorem should be declared side-by-side in [9].

Therefore, theorem 1 in [8] is at most only a conjecture which is very likely false.

3 Combination of Fact 1.1 and 1.2 Being Insufficient for $\ell(i) + \ell(j) = \ell(k)$

3.1 The Converse Proposition of Fact 1.1 Not Holding

Fact 1.1 in [8] is retailed as follows:

Fact 1.1 If $f(i) + f(j) = f(k)$, there exists a q_u such that $q_u = A_k$ in $\{p_0/q_0, p_1/q_1, \ldots, p_t/q_t\}$, the convergent sequence of continued fraction expansion of Z / M with $Z \equiv C_i\, C_j\, C_k^{-1} \bmod M$.

Notice that $f(i) + f(j) = f(k)$, namely $\ell(i) + \ell(j) = \ell(k)$.

Due to

$$Z / M = l / A_k + A_i\, A_j / (M\, A_k) \tag{1}$$

and $M > \prod_{h=1}^{n} A_h$ and $A_h \geq 2$, we have

$$Z / M - l / A_k = A_i\, A_j / (M\, A_k) < A_i\, A_j / (A_k \prod_{h=1}^{n} A_h) < 1 / (2\, A_k^2).$$

The right of the above formula is a very small number, and thus l / A_k may be regarded as a convergence of the continued fraction of Z / M.

Hence, Fact 1.1 should be correct in most cases. Attention please. An example which may illustrate that l/A_k is not a convergence of the continued fraction of Z / M when $\ell(i) + \ell(j) = \ell(k)$ can be found out.

Let p_v / q_v be the rational determined by integers $[a_0; a_1, a_2, \ldots, a_v]$ with

$$p_v / q_v = a_0 + 1 / (a_1 + 1 / (a_2 + \ldots + 1 / (a_{v-1} + 1 / a_v))). \tag{2}$$

Let $Z / M = [0; a_1, a_2, \ldots, a_t]$. Assigned l / A_k to p_u / q_u.

Attention please. Not fact 1.1, but the converse proposition of fact 1.1 is the reasoning base of Alg.1.

In what follows, by offering a counterexample, we will prove that the converse proposition of fact 1.1 does not hold, that is, the condition $Z / M - l / A_k < 1 / (2\, A_k^2)$ is not sufficient for $\ell(i) + \ell(j) = \ell(k)$.

For convenience in computing, let $n = 6$, $\{A_i\} = \{11, 10, 3, 7, 17, 13\}$, $\delta = 1$, and $M = 510931$.

Arbitrarily select $W = 17797$, $\ell(1) = 9$, $\ell(2) = 6$, $\ell(3) = 10$, $\ell(4) = 5$, $\ell(5) = 7$, and $\ell(6) = 8$.

From $C_i \equiv A_i W^{\ell(i)} \pmod M$, we obtain

$\{C_i\} = \{113101, 79182, 175066, 433093, 501150, 389033\}$,

and its inverse sequence

$\{C_i^{-1}\} = \{266775, 236469, 435654, 149312, 434038, 425203\}$.

Randomly select $i = 1, j = 3$, and $k = 5$. In this case, $\ell(5) = 7 \neq \ell(1) + \ell(3) = 9 + 10$. Compute

$$Z \equiv C_1\, C_3\, C_5^{-1} \equiv 113101 \times 175066 \times 434038 \equiv 186640 \pmod{510931}.$$

Presume that W in $C_1\, C_3$ is just neutralized by W^{-1} in C_5^{-1}, then

$$186640 \equiv A_1\, A_3\, A_5^{-1} \pmod{510931}.$$

According to Eq.(1),

$$186640 / 510931 - l / A_5 = A_1\, A_3 / (510931\, A_5).$$

By the Euclidean algorithm, $a_1, a_2, a_3, \ldots$ are found out, and thus the continued fraction of

$$186640 / 510931 = 1 / (2 + 1 / (1 + 1 / (2 + 1 / (1 + 1 / (4 + \ldots + 1 / 3))))).$$

Heuristically let

$$l / A_5 = 1 / (2 + 1 / (1 + 1 / (2 + 1 / 1))) = 4 / 11,$$

which indicates that probably $A_5 = 11$. On this occasion,

$$186640 / 510931 - 4 / 11 = 0.0016575801 < 1 / (2 A_5^2) = 1 / (2 \times 11^2) = 0.0041322314,$$

which satisfies theorem 1 so-called in [8]. By fact 1.1, $A_5 = 11$ less than the maximal number in $\{A_i\}$ is deduced out. This is in direct contradiction to factual $A_5 = 17$.

Thus the condition $Z / M - l / A_k < 1 / (2 A_k^2)$ is not sufficient, namely the converse proposition of fact 1.1 does not hold.

3.2 Fact 1.2 and 1.3 Both Not Holding

Fact 1.2 in [8] is retailed as follows:

Fact 1.2 There is sharp increase from q_u to q_{u+1} since $q_{u+1} \geq (A_k M / (2A_i A_j))^{1/2}$.

The derivation of fact 1.2 in [8] is retailed as follows:

Let l / A_k be the u-th convergent, i.e., $q_u = A_k$ and $p_u = l$, i.e., $p_u / q_u = l / A_k$. Then we know that

$$|Z / M - p_{u+1} / q_{u+1}| < A_i A_j / (A_k M) = 1 / (2 ((A_k M / (2A_i A_j))^{1/2})^2). \quad (2')$$

According to Theorem 1 and convergence of sequence $\{p_0 / q_0, p_1 / q_1, \ldots, p_t / q_t\}$, we obtain that

$$q_{u+1} \geq (A_k M / (2A_i A_j))^{1/2} = A_k (M / (2A_i A_j A_k))^{1/2}. \quad (3)$$

Is the above derivation right?

According to the definition of a finite continued fraction, (2') clearly holds.

Assume that theorem 1 is correct. Then, there is

$$|Z / M - p_{u+1} / q_{u+1}| < 1 / (2 q_{u+1}^2). \quad (3')$$

In terms of (2') and (3'), we have either

$$|Z / M - p_{u+1} / q_{u+1}| < 1 / (2 q_{u+1}^2) < 1 / (2 ((A_k M / (2A_i A_j))^{1/2})^2), \quad (4)$$

or

$$|Z / M - p_{u+1} / q_{u+1}| < 1 / (2 ((A_k M / (2A_i A_j))^{1/2})^2) < 1 / (2 q_{u+1}^2). \quad (5)$$

If (4) holds, there exists $q_{u+1} \geq A_k (M / (2A_i A_j A_k))^{1/2}$ with $q_{u+1} \geq A_k = q_u$.

If (5) holds, there exist $A_k (M / (2A_i A_j A_k))^{1/2} \geq q_{u+1}$. In this case, $q_{u+1} \geq A_k = q_u$ is still possible.

Therefore, $q_{u+1} \geq A_k (M / (2A_i A_j A_k))^{1/2}$, namely fact 1.2 does not hold, which indicates that there is a logic error in the derivation of (3).

Fact 1.3 in [8] is retailed as follows:

Fact 1.3 Due to fact 1.2, there is also a sharp increase from a_u to a_{u+1}, since $q_{v+1} = a_{v+1} q_v + q_{v-1}$ for $v = 1, 3, \ldots, t$. Here a_vs are items of Z / M determined by Eq.(2).

Obviously, because fact 1.2 does not hold, fact 1.3 does also not hold.

Because fact 1.2, namely Eq.(3) does not hold, fact 4 does not hold naturally, that is, $\Delta = (M / (2\prod_{i=n-2}^{m} \text{Prime}[i]))^{1/2}$ is meaningless.

Observe an example once more.

Suppose that the bit-length of a plaintext block is 8, and two bits of a block correspond to three items of a coprime sequence $\{A_i\}$, namely the encryption algorithm is optimized through compact binary sequences.

Apparently, the length of $\{A_i\}$ is $3 \times (8 / 2) = 12$.

Let $\{A_i\} = \{23, 11, 17, 41, 29, 26, 15, 19, 37, 31, 7, 43\}$, $\delta = 1$, and $M = 2022169$.

Randomly select $W = 1507351$, $\ell(1) = 6$, $\ell(2) = 14$, $\ell(3) = 9$, $\ell(4) = 11$, $\ell(5) = 12$, $\ell(6) = 10$, $\ell(7) = 8$, $\ell(8) = 16$, $\ell(9) = 5$, $\ell(10) = 13$, $\ell(11) = 15$, and $\ell(12) = 7$.

From $C_i \equiv A_i W^{\ell(i)} \pmod M$ obtain

$\{C_i\} = \{\{572402, 1930240, 374715\}, \{25128, 265158, 350520\}, \{1674837, 1231458, 1448214\}, \{110225, 1198155, 757620\}\}$, and

$\{C_i^{-1}\} = \{\{1422427, 1728992, 1733243\}, \{988793, 1040888, 93176\}, \{1591882, 1096007, 508027,\} \{1283676, 1682978, 956584\}\}$.

Let

$$Z \equiv (C_4 C_{12}) (C_6^{-1} C_7^{-1}) \equiv (25128 \times 757620) (93176 \times 1591882)$$
$$\equiv 776394 \times 1123251 \equiv 689616 \pmod{2022169}.$$

Then, $689616 / 2022169 - l / (A_6 A_7) = (A_4 A_{12}) / (2022169 A_6 A_7)$.

Further, the continued fraction of $689616 / 2022169$ is

$$1 / (2 + 1 / (1 + 1 / (13 + 1 / (1 + (1 / (3 + 1 / (2 + 1 / (2 + 1 / (2 + 1 / (97 + 4 / 9)))))))))).$$

Heuristically let

$$l / (A_6 A_7) = 1 / (2 + 1 / (1 + 1/ (13 + 1/ (1 + (1/ (3 + 1 / 2))))) = 133 / 390,$$

which indicates that probably $A_6 A_7 = 390$. Because the discriminant

$$689616 / 2022169 – 133 / 390 = 2.235477262\text{e-}6 < 1 / (2 \times 390^2) = 3.287310979\text{e-}6,$$

satisfies theorem 1 so-called in [8], $A_6 A_7 = 390$ is deduced out.

The integer 390 may be factorized into the pairs (2, 195), (3, 130), (5, 78), (6, 65), (10, 39), (13, 30), or (15, 26), where the elements of (10, 39), (13, 30), and (15, 26) are less than maximal number in $\{A_i\}$. Thus, true $(A_6, A_7) = (26, 15)$ is included in 6 potential cases. Here, $a_u = 2$ and also $a_{u+1} = 2$, and there is no sharp increase from a_u to a_{u+1}.

Additionally, this example also illustrates that when one attempts to reason the fit factors of the product $(A_{k1} A_{k2})$ by $\ell(i) + \ell(j) = \ell(k_1) + \ell(k_2)$ with every $\ell(x) \in \Omega = \{n + 1, \ldots, 2n\}$, much more indeterminacy is increased.

3.3 Combination of Fact 1.1 and 1.2 Being Insufficient for $\ell(i) + \ell(j) = \ell(k)$

Even if fact 1.3 were right, a counterexample would be given easily.

Still let $n = 6$, $\{A_i\} = \{11, 10, 3, 7, 17, 13\}$, $\delta = 1$, and $M = 510931$.

Arbitrarily select $W = 17797$, $\ell(1) = 9$, $\ell(2) = 6$, $\ell(3) = 10$, $\ell(4) = 5$, $\ell(5) = 7$, and $\ell(6) = 8$.

From $C_i \equiv A_i W^{\ell(i)} \pmod M$, we obtain $\{C_i\} = \{113101, 79182, 175066, 433093, 501150, 389033\}$, and its inverse sequence $\{C_i^{-1}\} = \{266775, 236469, 435654, 149312, 434038, 425203\}$.

Randomly select $i = 1, j = 3$, and $k = 6$. In this case, $\ell(6) = 8 \neq \ell(1) + \ell(3) = 9 + 10$. Compute

$$Z \equiv C_1 C_3 C_6^{-1} \equiv 113101 \times 175066 \times 425203 \equiv 425865 \pmod{510931}.$$

Presume that W in $C_1 C_3$ is just neutralized by W^{-1} in C_6^{-1}, then

$$425865 \equiv A_1 A_3 A_6^{-1} \pmod{510931}.$$

According to Alg.1 in [8],

$$425865 / 510931 – l / A_6 = A_1 A_3 / (510931 A_6).$$

Compute the continued fraction of186640 / 510931 being

$$1 / (1 + 1 / (5 + 1 / (159 + 1 / 535))).$$

Heuristically let

$$l / A_6 = 1 / (1 + 1 / 5) = 5 / 6,$$

which indicates that probably $A_6 = 6$. Further, judge that

$$425865 / 510931 – 5 / 6 = 0.000174518 < 1 / (2 \times 6^2) = 0.0138889,$$

satisfies theorem 1 so-called in [8].

Let $q_u = A_k = A_6 = 6$, and $p_{u+1} / q_{u+1} = 1 / (1 + 1 / (5 + 1 / 159)) = 796 / 955$.

In addition, $A_k (M / (2A_i A_j A_k))^{1/2} = 6 (510931 / (2 \times 11 \times 3 \times 6))^{1/2} = 6 \times 35.9197 = 215.5186$.

Thus, $q_{u+1} = 955 > A_k (M / (2A_i A_j A_k))^{1/2} = 216$ satisfies Eq. (3), and $a_{u+1} = 159 > a_u = 5$ satisfies fact 1.3.

By fact 1.1 and 1.3, $A_6 = 6$ is deduced out. However, this is in direct contradiction to factual $A_6 = 13$, and thus the combination of fact 1.1 and 1.2 is not sufficient for $\ell(i) + \ell(j) = \ell(k)$.

Because the combination of fact 1.1 and either fact1.2 or fact 4 is not sufficient for $f(i) + f(j) = f(k)$, property 1, 2, and 5 are meaningless. Further, running results of Alg.1 and 2 from an arbitrary public key sequence $\{C_1, \ldots, C_n\}$ will be incorrect.

On the basis of the above examples, we can infer that due to the randomness of W and $\ell(i)$, A_i will be evaluated to every potential value in $\{A_1, \ldots, A_n\}$ by fact 1.1 and 1.3. Therefore, the time complexity of such a attack will get to at least $O(n!)$, where $n \geq 80$ and $n! = n(n – 1) \ldots 1$.

4 The Example in [8] Illustrate Nothing about Breaking

We have no time to verify the correctness of the example in [8], but a suitable example should be able to be constructed since

(i) The lever value $\{\ell(1), \ldots, \ell(n)\}$ is known in advance;

(ii) The coprime sequence $\{A_1, \ldots, A_n\}$ may be designed elaborately in advance;

(iii) The condition $Z / M - l / A_k < 1 / (2 A_k^2)$ in fact 1.1 satisfies necessity in most cases;

(iv) The condition $q_{u+1} \geq (A_k M / (2A_i A_j))^{1/2}$ in fact 1.2 satisfies necessity in some cases.

However, as the above refution indicates, a suitable example does not illustrates that the original $\{A_1, \ldots, A_n\}$ can be extracted accurately from the public key $\{C_1, \ldots, C_n\}$ when $\{\ell(1), \ldots, \ell(n)\}$ and $\{A_1, \ldots, A_n\}$ are unknown in advance.

The example in [8] is neither legible nor repeatable in short time, and the proportion of n to $\log_2 M$ is not proper, which contravenes the optimization principle for the modulus M in the REESSE1+ system. An obvious fact is that if M is too large, the length of a public key will increase rapidly. Therefore, M should be as small as possible while it meets $M > \prod_{i=1}^{n} A_i$ and other constraints. Selection of some elements of the sequence $\{A_i\}$ in [8] also contravenes the optimization principle.

The intent for [8] to select such a large M that n is out of proportion to $\log_2 M$ seems to want to increase the necessity of the conditions in fact 1.1 and 1,2 for $\ell(i) + \ell(j) = \ell(k)$. However, it can not increase the sufficiency of the conditions in fact 1.1 and 1,2.

The table below displays the relevant parameters in the optimized REESSE1+ cryptosystem through binary compact sequences.

len n Block (bits)	$\tilde{n} = 3n / 2$	$n / 2$	max index η	Max prime P_η	Num Cps	$\log_2 M$ (bits)	len Pvtk (bits)	len Pubk (bits)	len Ciph (bits)
80	120	40	150	863	$> A_{150}^{120}$	384	3312	47232	384
96	144	48	159	937	$> A_{159}^{144}$	464	3984	68208	464
112	168	56	167	991	$> A_{156}^{142}$	544	4656	93024	544
128	192	64	197	1201	$> A_{197}^{192}$	640	5376	124800	640

There is no problem on storing a public key in modern IC cards. Under some extreme circumstances, the bit-length of a public key may be shortened through a compression algorithm.

5 Attack on the Signature Being an Eisegesis

5.1 T^{-1} mod $(M - 1)$ Not Existing and Q^{-1} Not Necessarily Existing

[8] deduces out $U \equiv ((Q/H)^{1/S} \hat{\boldsymbol{G}} (GW)^{-1} \delta^{\delta(\delta+1)-1/S})^{QT} \pmod{M}$, which is right.

Further, [8] gives $(GW)^{-1} \delta^{\delta(\delta+1)-1/S} \equiv ((Q/H)^{-S-1} \hat{\boldsymbol{G}}^{-1}) U^{(QT)-1} \pmod{M}$, which is wrong since T^{-1} mod $(M - 1)$ with $T \mid (M - 1)$ does not exist, and neither does Q^{-1} mod $(M - 1)$ exist when $\gcd(Q, M - 1) > 1$. In the signature algorithm, it is easy to let $\gcd(Q, M - 1) > 1$.

Let $x \equiv (GW)^{-1} \delta^{\delta(\delta+1)-1/S} \pmod{M}$.

Therefore, the trivial solution to $x^{QT} \equiv U((Q/H)^{1/S} \hat{\boldsymbol{G}})^{-QT} \pmod{M}$ does not exist when $\gcd(T, (M - 1) / T) > 1$.

Due to stipulating $T \geq 2^n$ in the key generation algorithm, the time complexity of finding out a random solution to $x^{QT} \equiv U((Q/H)^{1/S} \hat{\boldsymbol{G}})^{-QT}$ is at least $\max(O(2^{n-1}), O(M/(QT)))$ through the probabilistic algorithm [10].

If a solution to $x^{QT} \equiv U((Q/H)^{1/S} \hat{\boldsymbol{G}})^{-QT}$ is found through the discrete logarithm method, the probability that the solution is just equal to $(GW)^{-1} \delta^{\delta(\delta+1)-1/S} \pmod{M}$ is at most $1 / 2^n$.

If let $x \equiv ((GW)^{-1} \delta^{\delta(\delta+1)-1/S})^T \pmod{M}$, then $x^Q \equiv U((Q/H)^{1/S} \hat{\boldsymbol{G}})^{-QT} \pmod{M}$. When $\gcd(Q, M - 1) > 5$ and $M / Q > 2^n$, seeking a solution to $x^Q \equiv U((Q/H)^{1/S} \hat{\boldsymbol{G}})^{-QT}$ is also at least the discrete logarithm problem.

5.2 Identity Verification Algorithm Being Modified

[8] modifies the identity verification algorithm, which is not allowed since the verification program is written by the verification algorithm of the REESSE1+ cryptosystem.

When a user try to verify an identical signature on a file more than one time, the verification program will think that such behavior is a fraud ant terminate it right away.

6 Conclusion

The above refution shows the cryptanalysis in [8] is a pseudo attack on the REESSE1+ cryptosystem, the relevant conditions does not meets sufficiency, and there are logic errors in the deduction. Hence, the conclusion of [8] that REESSE1+ is not secure at all is completely incorrect.

Especially, it deviates from normal academic criticism that so-called 'theorem 1' which is unproven in [8] and unfound in the reference is willfully declared to be a theorem.

For greater security, in practical applications, it is suggested on the shortcut that users move the parameter H in $Q \equiv (R\,G_0)^S H\delta \pmod{M}$ into $D \mid (\delta Q - W)$ becoming $D \mid (\delta Q - WH)$. Correspondingly, let λ $S \equiv ((\delta+1)U + \sum_{i=0}^{n-1} (\delta Q)^{n-1-i} (WH)^i)(\delta Q - WH) \pmod{M-1}$, and in the verification algorithm, judge whether $(\alpha Q^{-1})^{QUT} \alpha^{Q^n T} \equiv (\hat{G}^{QT} U^{-1})^{US} \beta^{UHT} \gamma^{H^n T} \pmod{M}$ or not.

Rome is not build in one day, and likewise Rome can not be demolish in one day.

References

[1] Shenghui Su, *The REESSE1 Public-key Encryption Algorithms*, Int. C1: H04L 9/14, ZL01110163.6, Chinese Patent, Apr. 2001.

[2] Shenghui Su, *The REESSE1 Public-key Cryptosystem*, Computer Engineering & Science, Chinese, v25(5), 2003, pp.13-16.

[3] Shenghui Su, Yixian Yang and Bingru Yang, *The Necessity and Sufficiency Analysis of the Lever Function in the REESSE1 Encryption Scheme*, Acta Electronica Sinica, Chinese, v34(10), 2006, pp.1892-1895. (Received May 13, 2005)

[4] Shenghui Su and Shuwang Lü, *The REESSE1+ Public-key Cryptosystem*, http://eprint.iacr.org/2006/420.pdf.

[5] Shengli Liu, Fangguo Zhang and Kefei Chen, *Cryptanalysis of REESSE1 Digital Signature Algorithm*, CCICS 2005, Xi'an, China, May 2005.

[6] Shengli Liu, Fangguo Zhang and Kefei Chen, *Cryptanalysis of REESSE1 Public Encryption Cryptosystem*, Information Security, Chinese, no.7, 2005, pp.121-124.

[7] Shenghui Su, *Refuting the Pseudo Attack on the REESSE1 Public-key Algorithms for Encryption*, Computer Engineering and Applications, Chinese, v42(20), 2006, pp.129-133.

[8] Shengli Liu and Fangguo Zhang, *Cryptanalysis of REESSE1+ Public Key Cryptosystem*, http://eprint.iacr.org/ 2006/ 480.pdf, Dec. 22, 2006.

[9] Kenneth H. Rosen, *Elementary Number Theory and Its Applications* (5th ed.), Boston: Addison-Wesley, 2005, ch. 12.

[10] Henri Cohen, *A Course in Computational Algebraic Number Theory*, Berlin: Springer-Verlag, 2000, ch. 1, 3.

Remark: This paper was sent to the authors of [8] via email on March 6, 2007.