Weak property of malleability in NTRUSign

SungJun Min[1⋆], Go Yamamoto[2], and Kwangjo Kim[3]

[1] National Computerization Agency (NCA),
NCA Bldg, 77, Mugyo-Dong, Jung-Gu, Seoul, Korea,
sjmin@nca.or.kr
[2] NTT, Information Sharing Platform Laboratories,
1-1, Hikarinooka, Yokosuka, Kanagawa, Japan,
yamamo@isl.ntt.co.jp
[3] International Research center for Information Security (IRIS),
Information and Communications University (ICU),
119, Munjiro, Yuseong-Gu, Daejeon, 305-714, Korea,
kkj@icu.ac.kr

Abstract. A new type of signature scheme, called NTRUSign, based on solving the approximately closest vector problem in a NTRU lattice was proposed at CT-RSA'03. However no security proof against chosen messages attack has been made for this scheme. In this paper, we show that NTRUSign signature scheme contains the weakness of malleability. From this, one can derive new valid signatures from any previous message-signature pair which means that NTRUSign is not secure against strongly existential forgery. Finally, we propose a simple technique to avoid this flaw in NTRUSign scheme.

Keywords: NTRUSign, Digital Signature Scheme, Strong Existential Forgery, Malleability, Centered Norm

1 Introduction

Recently, Hoffstein *et al.* introduced a new type of authentication and digital signature scheme called NTRUSign [7] at CT-RSA'03. While traditional signature schemes are based on well-known hard problem such as factorization or discrete log problem, the hard problem underlying NTRUSign is to find the approximately shortest(or closest) vectors in a certain lattice, called NTRU lattice L_h^{NT}. In this scheme, the signer uses secret knowledge to find a point in the NTRU lattice close to the given point. He/She then exploits this approximate solution to the closest vector problem as his signature. One of the significant advantages is the fast operation: NTRU-based algorithms, for example, executes hundreds of times faster while providing the same security than competing algorithms such as RSA. In this paper, we claim that the NTRUSign signature scheme, however, does not contain one of important cryptographic properties

⋆ This work was done while the author was with Information and Communications University(ICU).

that the signature scheme should guarantee, *non-malleability*. We first suggest a deterministic attack method how an attacker can generate new valid signatures from the previously signed message. Next, we propose a simple technique to avoid this attack.

History of NTRUSign scheme Since the advent of NTRU encryption scheme based on a hard mathematical problem of finding short vectors in certain lattices in 1996, several related signature schemes such as NSS [10] and R-NSS [6] have been proposed. A fast authentication and digital signature scheme called NSS, based on the same underlying hard problem and using keys of the same form, was presented at Eurocrypt 2001 [10]. However, this scheme was broken by Mironov and Gentry *et al*, see [3, 12]. In their Eurocrypt presentation, the authors of NSS sketched a revised version of NSS (called R-NSS) and published it in the preliminary cryptographic standard document EESS [18]. Although it seemed that R-NSS was significantly stronger than the initial version(NSS), it was proved that the key recovery attack could be mounted by Gentry and Szydlo [4]. The source of these weaknesses about NSS and R-NSS was an incomplete linking of the NSS method with the approximate closest vector problem in the NTRU lattice. In other words, the weaknesses of NSS and R-NSS arose from the fact that the signer did not possess a complete basis of short vectors for the NTRU lattice L_h^{NT}. Later on, Hoffstein *et al.* proposed a new NTRU based signature scheme called NTRUSign. Unlike the old signature schemes, the link in NTRUSign between the signature and the underlying approximate closest vector problem is clear and direct: the signer must solve an "approximate CVP problem" in the lattice *i.e.,* produce a lattice point that is sufficiently close to a message digest point. This paper, however, describes a weakness in NTRUSign: from any given message-signature pair, one can derive many different signatures of the same message, thus it is *malleable*.

Impact of malleability If a signature scheme is malleable, we can derive another signature of the message from any message-signature pair. In this case, we cannot distinguish it from the original one generated by who knows the secret key, which can be in practice regarded as a forgery. Although such a weakness does not allow the attacker to change the message string, this forgery shows that the signature scheme cannot be used for all kinds of applications. For example, if one would like to apply it to electronic cash, finding a second valid signature for a given bill should be impossible. Also, an entity receiving the message-signature pairs (m, s) and (m, s') such that $s \neq s'$ at the same time, neither s nor s' will be accepted as a valid signature for the message m by him. If a legitimate signer wants to assert s as his/her own signature for the message m, then he/she should exhibit his/her private key, which is a negative property.

Our Contributions In this paper, we show how a passive adversary observing only a valid message-signature pair can generate another signature on the same

message. The main idea of this forgery is to use specific polynomials of which norm value is zero. Although this weakness might be overlooked for some applications, NTRUSign is not secure in the non-malleability sense against known message attack. The notion of this security is well described in [16]. Finally we propose a simple technique to avoid our proposed attack.

Organization The rest of this paper is organized as follows: In Section 2, we briefly describe the NTRUSign signature scheme. We do not give all the technical and theoretical details for the functions used in the scheme. Only the general construction is described here.

In Section 3 we show how an attacker can forge an additional signature for a message previously signed using some specific polynomials, and then in Section 4, we introduce a simple method to avoid this weakness. Finally, we make concluding remarks in Section 5.

2 Description of NTRUSign Algorithm

In this section, we briefly describe NTRUSign digital signature scheme. As NTRU encryption scheme, basic operations take place in the quotient polynomial ring $R = \mathbb{Z}[x]/(x^N - 1)$, where N is the security parameter. A polynomial $a(x) \in R$ (shortly, a) can be presented by a vector $\mathbf{a}$ of its coefficients as follows:

$$\mathbf{a} = \sum_{i=0}^{N-1} a_i x^i = (a_0, a_1, \cdots, a_{N-1}).$$

For the sake of simplicity, we will use the same notation for the polynomial $a(x)$ and the vector $\mathbf{a}$. The product of two polynomials a and b in R is simply calculated by $a * b = c$, where the k-th coefficient c_k is

$$c_k = \sum_{i=0}^{k} a_i b_{k-i} \; + \sum_{i=k+1}^{N-1} a_i b_{N+k-i} = \sum_{i+j \equiv k \pmod N} a_i b_j.$$

In some steps, NTRUSign uses the quotient ring $R_q = \mathbb{Z}_q[x]/(x^N - 1)$, where the coefficients are reduced by modulo q, where q is typically a power of 2, for example 128. The multiplicative group of units in R_q is denoted by R_q^*. The inverse polynomial of $a \in R_q^*$ is denoted by a^{-1}. If a polynomial a has all coefficients chosen from the set $\{0, 1\}$, we call this a *binary* polynomial.

The security of NTRUSign scheme is based on the approximately closest vector problem in a certain lattice, called NTRU lattice. In this scheme, the signer can sign a message by demonstrating the ability to solve the approximately closest vector problem reasonably well for the point generated from a hashed message in a given space.

The basic idea is as follows: The signer's private key is a short basis for an NTRU lattice and his public key is a much longer basis for the same lattice. The signature on a digital document is a vector in the lattice with two properties:

- The signature is attached to the document being signed.
- The signature demonstrates an ability to solve a general closest vector problem in the lattice.

NTRUSign digital signature scheme works as follows:

System Parameters

1. N: a (prime) dimension.
2. q: a modulus.
3. d_f, d_g: key size parameters.
4. *NormBound*: a bound parameter of verification.

Key Generation

A signer creates his public key h and the corresponding private key $\{(f, g), (F, G)\}$ as follows:

1. Choose binary polynomials f and g with d_f 1's and d_g 1's, respectively.
2. Compute the public key $h \equiv f^{-1} * g \pmod{q}$.
3. Compute small polynomials (F, G) satisfying $f * G - g * F = q$.

Signing Step

A signer generates his signature s on the digital document D as follows:

1. Obtain the polynomials (m_1, m_2) mod q for the document D by using the public hash function.
2. Write
$$G * m_1 - F * m_2 = A + q * B,$$
$$-g * m_1 + f * m_2 = a + q * b,$$
where A and a have coefficients between $-q/2$ and $q/2$.
3. Compute polynomials s and t as
$$s \equiv f * B + F * b \pmod{q},$$
$$t \equiv g * B + G * b \pmod{q}.$$
Here, a vector $(s, t) \in L_h^{NT}$ is very close to $m = (m_1, m_2)$.
4. The polynomial s is the signature on the digital document D for the public key h.

Verification Step

For a given signature s and document D, a verifier should do the following:

1. Hash the document D to recreate (m_1, m_2) mod q.
2. Using the signature s and public key h, compute the corresponding polynomial
$$t \equiv s * h \pmod{q},$$
which becomes exactly the same as the polynomial $g * B + G * b \pmod{q}$. (Note that (s, t) is a point in the NTRU lattice L_h^{NT}.)

3. Compute the distance from (s,t) to (m_1,m_2) and verify that this value is smaller than the *NormBound* parameter. In other words, check that

$$\| s-m_1 \|^2 + \| t-m_2 \|^2 \leq NormBound^2,$$

where the norm($\| \cdot \|$) is a centered norm.

NTRUSign algorithm uses the centered norm concept instead of Euclidean norm in verification step to measure the size of an element $a \in R$.

Definition 1. *Let $a(x)$ be a polynomial in ring $R = \mathbb{Z}[x]/(x^N-1)$. Then the centered norm of $a(x)$ is defined by*

$$\| a(x) \|^2 = \sum_{i=0}^{N-1} (a_i - \mu_a)^2 = \sum_{i=0}^{N-1} {a_i}^2 - \frac{1}{N}(\sum_{i=0}^{N-1} a_i)^2,$$

where $\mu_a = \dfrac{1}{N}\displaystyle\sum_{i=0}^{N-1} a_i$ *is the average of the coefficients of $a(x)$.*

The centered norm of an n-tuple $(a_1, a_2, \cdots, a_n)$ with $a_1, a_2, \cdots, a_n \in R$ can be defined by this formula

$$(\| (a_1, a_2, \cdots, a_n) \|)^2 = \| a_1 \|^2 + \| a_2 \|^2 + \cdots + \| a_n \|^2 .$$

Note that the signature on D is a vector (s,t) in NTRU lattice L_h^{NT}, which is very close to m. To solve an approximately closest vector problem in the lattice, a signer uses a "short basis" defined as below:

Definition 2. *A basis $\{(f,g),(F,G)\}$ is called a short basis in L_h^{NT} if*

$$\| f \|, \| g \| = O(\sqrt{N}), \text{ and } \| F \|, \| G \| = O(N).$$

The signing process of NTRUSign may be explained by the following matrix equation, which shows that the role of a signer is to find approximate solution about the closest vector problem by using his short basis $\{(f,g),(F,G)\}$:

$$\begin{aligned}(s\ t) = (B\ b)\begin{pmatrix} f & g \\ F & G \end{pmatrix} &= \left\lfloor (m_1\ m_2)\begin{pmatrix} G/q & -g/q \\ -F/q & f/q \end{pmatrix}\right\rceil \begin{pmatrix} f & g \\ F & G \end{pmatrix} \\ &= \left\lfloor (m_1\ m_2)\begin{pmatrix} f & g \\ F & G \end{pmatrix}^{-1}\right\rceil \begin{pmatrix} f & g \\ F & G \end{pmatrix}\end{aligned}$$

A valid signature demonstrates that the signer knows a lattice point that is within *NormBound* of the message digest vector m. Clearly, the smaller that *NormBound* is set, the more difficult it will be for an attacker, without knowledge of the private key, to solve this problem. The designers recommend that the suggested parameters $(N, q, d_f, d_g, NormBound) = (251, 128, 73, 71, 300)$ offer security at least as strong as 1,024 bit RSA [8].

3 Weakness in NTRUSign

In this section we describe that the NTRUSign is strong existential forgeable, sometimes this notion is called as malleable. Strong existential forgeability for a given signature scheme means that one can create a message-signature pair that has never been observed by the signer [16]. A different signature for a once legitimately signed message can be regarded as a forgery. In practice, this forgery shows that the NTRUSign scheme cannot be used for all kinds of applications. For example, in electronic cash system, finding another valid signature for a given bill should be impossible. Thus the application area of this scheme is limited, because a digital signature scheme is selected according to both its security level and the context of use.

Now we will describe how we can generate a valid signature different from a previous valid signature for a given message. Remind that NTRUSign signature scheme uses the centered norm concept in verification step. The centered norm has quasi-multiplicative property, that is, $\| a(x) * b(x) \| \approx \| a(x) \| * \| b(x) \|$ for random polynomials $a(x)$ and $b(x)$ in R, which was well discussed in [9]. The properties of the centered norm will be employed to induce a new signature from a given signature without knowing the private key.

The following lemma describes the centered norm properties:

Lemma 1. *Let R be a quotient polynomial ring $R = \mathbb{Z}[x]/(x^N - 1)$. Then*

(i) In $R_q = \mathbb{Z}_q[x]/(x^N - 1)$, there exist exactly q polynomials $\alpha(x)$ such that $\| \alpha(x) \| = 0$.
(ii) If $\| \alpha(x) \| = 0$, then $\| \alpha(x) * \beta(x) \| = 0$ for every polynomial $\beta(x) \in R$.

Proof. (i) It is obvious that $\alpha_0 = \alpha_1 = \cdots = \alpha_{N-1}$ for $\alpha_i \in (-q/2, q/2]$ if and only if $\sum_{i=0}^{N-1}(a_i - \mu_a)^2 = 0$ where $\mu_a = \frac{1}{N}\sum_{i=0}^{N-1} a_i$, namely $\| a(x) \| = 0$.

(ii) From the result of (i) we can know that all coefficients of α are the same, say $\alpha = (\alpha_0, \alpha_0, \cdots, \alpha_0)$. Then, clearly the k-th coefficient of $\alpha * \beta$ is $\sum_{i=0}^{N-1}(\alpha_0\beta_{k-i}) + \sum_{i=k+1}^{N-1}(\alpha_0\beta_{N+k-i}) = \alpha_0(\beta_0 + \cdots + \beta_k + \beta_{k+1} + \beta_{N-1}) = \alpha_0 * \beta$, and so are the other coefficients of $\alpha * \beta$ the same. Again by applying to (i), we complete the proof of this lemma. □

We call these q polynomials satisfying $\| \alpha(x) \| = 0$ *annihilating polynomial*. These annihilating polynomials makes the NTRUSign algorithm to be malleable.

Hoffstein *et al.* argued that forgery of a signature in NTRUSign is equivalent to solve an approximately closest vector problem in high dimension for the class of NTRU lattices. It seems to be true if we do not consider the stronger attack model. Historically, Goldwasser, Micali and Rivest [5] introduced the notion of existential forgery against chosen-message attack for public key signature scheme. This notion has become the *de facto* security definition for digital signature algorithm, against which all new signature algorithms are measured. In this scenario, an adversary with access to the public key of the scheme and to a signing oracle, should not be able to forge a valid signature for some new message or

for a message of his choice(existential forgery and selective forgery, respectively). An even stronger requirement called the non-malleability, or strong unforgeability, also forbids an adversary to forge an additional signature for a message which might already have been signed by the oracle [16]. We can see more detail security notions for digital signature scheme and the relation between them in [5, 14].

Now we will show that one can easily generate a message-signature pair that has never been observed by the signer. To create additional valid signatures we use the following observations: Note that all coefficients of polynomials are reduced by modulo q.

Remark 1. Let α be an annihilating polynomial. Then $\| r + \alpha \| \approx \| r \|$ for randomly chosen polynomial $r \in R$.

If both "reduced form" and "not reduced form" of polynomial $r+\alpha$ are equal, then the centered norm values of $\| r \|$ and $\| r + \alpha \|$ are exactly the same. The differences between $\| r + \alpha \|$ and $\| r \|$ are caused from only the gap failure. The concepts of gapping and wrapping failure are presented in [15]. We have implemented the above remark with the suggested parameters 1,000 times for each α using Mathematica 4.2. It is clear that as the coefficients of annihilating polynomial gets smaller, the probability of having the same norm gets higher. When the coefficient of α is ± 1 or ± 2, our experiment shows that each probability which two centered norm values are exactly the same becomes 0.15 and 0.015 approximately. **Figure 1** describes the distribution of distances between $\| r + \alpha \|$ and $\| r \|$ for random polynomial $r \in R$, where the x-axis denotes the integer coefficient α_i of an annihilating polynomial and y-axis denotes the average distance between $\| r + \alpha \|$ and $\| r \|$ for random polynomial r.

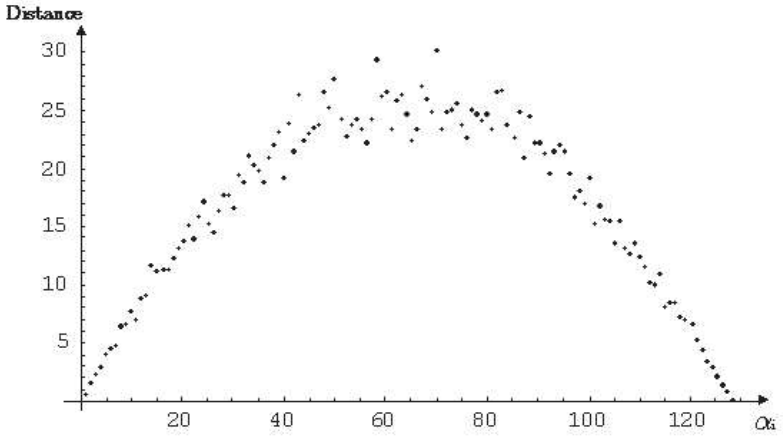

Fig. 1. Distance between $\| r + \alpha \|$ and $\| r \|$

We will see some results induced from the properties of an annihilating polynomial. For any polynomial $f = (f_0, f_1, \cdots, f_{N-1}) \in R$, $\mathcal{V}(f)$ denotes the sum of all coefficients of f modulus q, that is,

$$\mathcal{V}(f) = f(1) = \sum_{i=0}^{N-1} f_i \pmod{q} \in \mathbb{Z}_q. \tag{1}$$

For any $f \in R$, the product $f * \alpha$ can be presented by $\mathcal{V}(f)\alpha$, where α is an annihilating polynomial (See the proof of **Lemma 1**.).

From (1) it is trivial that $\mathcal{V}$ has the following properties:

Lemma 2. *Let f and g be two polynomials in R.*

(i) $\mathcal{V}(f)\mathcal{V}(g) \equiv \mathcal{V}(f * g) \pmod{q}$.
(ii) $\mathcal{V}(f^{-1}) \equiv \mathcal{V}(f)^{-1} \pmod{q}$ *if f has an inverse in R_q.*

Proof. By definition of $\mathcal{V}$, we have

$$\begin{aligned}\mathcal{V}(f)\mathcal{V}(g) &\equiv f(1)g(1) = (f * g)(1) \\ &\equiv \mathcal{V}(f * g) \pmod{q}.\end{aligned}$$

Obviously $\mathcal{V}(f^{-1})\mathcal{V}(f) \equiv \mathcal{V}(f^{-1} * f) \equiv \mathcal{V}(1) \equiv 1 \pmod{q}$, hence $\mathcal{V}(f^{-1}) \equiv \mathcal{V}(f)^{-1} \pmod{q}$. □

Assume that one chooses two polynomial pair (f, g), where f has an inverse in R_q. If there exists somewhat small integer $\alpha_0 \in (-q/2, q/2]$ satisfying $\alpha_0\mathcal{V}(f)^{-1}\mathcal{V}(g) \pmod{q}$ is also small, then we can know that both polynomial $\alpha = (\alpha_0, \alpha_0, \cdots, \alpha_0)$ and $(f^{-1} * g) * \alpha$ are annihilating polynomials with somewhat small coefficients from **Lemma 2**.

Remark 2. In the suggested parameters $(d_f, d_g) = (73, 71)$ given in [8], one has $\mathcal{V}(f) = -55$ and $\mathcal{V}(g) = -57$. In this case one can choose $\alpha = 8\sum_{i=0}^{N-1} x^i$ so that

$$\begin{aligned}h * \alpha \pmod{q} = \mathcal{V}(h)\alpha &= \mathcal{V}(f^{-1} * g) * \alpha \\ &= \mathcal{V}(f)^{-1}\mathcal{V}(g) * \alpha \\ &= -8\sum_{i=0}^{N-1} x^i.\end{aligned}$$

For a given signature $(s, t) \in L_h^{NT}$ generated under the suggested parameters, we take $s' = s + \alpha \pmod{q}$, where $\alpha = 8\sum_{i=0}^{N-1} x^i$. Then the corresponding signature pair t' is

$$\begin{aligned}t' = s' * h \pmod{q} &= s * h + \alpha * h \pmod{q} \\ &= t - 8\sum_{i=0}^{N-1} x^i \pmod{q}.\end{aligned}$$

At this time, we can expect that both $\| s - m_1 \|$ and $\| t - m_2 \|$ are small. Moreover, it is plausible that the small number of their coefficients are out of the range $(-64 + 8, 64 - 8]$. Form these reasons, the new lattice point $(s', t') = (s + 8\sum_{i=0}^{N-1} x^i, t - 8\sum_{i=0}^{N-1} x^i)$ will be another valid signature with high probability. Simply speaking, if one has $s - m_1$ without any coefficients greater than 56 and $t - m_2$ without any coefficients less than -55, then one can have the following equation exactly:

$$\begin{aligned} \| s' - m_1 \|^2 + \| t' - m_2 \|^2 &= \| s - m_1 \|^2 + \| t - m_2 \|^2 \\ &\leq NormBound^2, \end{aligned}$$

which means that (s', t') is always another valid signature.

A numerical experimental result shows that one has much more chance to succeed in the proposed attack: we examine a set P that consists of 128,000 elements from $\mathbb{Z}_{128}[x]/(x^{251} - 1)$ generated in such a way that all coefficients are randomly chosen from normal distribution with uniformly chosen means $\mu \in (-64, 64]$ and a fixed standard deviation $\sigma = \sqrt{\mathrm{NormBound}^2/N} \approx 18.9$. For two sets

$$P' = \{s \in P \mid \| s \|^2 < 300^2\} \text{ and } P'' = \{s \in P' \mid \| s + 8\sum_{i=0}^{N-1} x^i \|^2 < 300^2\},$$

we obtained the result that the set P' consists of 20,650 distinct elements and that P' and P'' coincide exactly.

We implemented the full NTRUSign signature scheme as described in [8] and [17] with suggested parameters using GNU MP version 4.1.2. Our experiment illustrates that the proposed forgery s' almost always succeeds for given message document D and a valid signature s. **Table 1** depicts the approximate probability that new pair $(s', t') = (s + \alpha, t + h * \alpha) \pmod q$ would be another signature for a given valid signature (s, t). In **Table 1**, note that α_i denotes the coefficient of an annihilating polynomial α and two sets A and B mean as follows:

$$A = \{(s, t) \in L_h^{NT} \mid (s, t) \text{ is a valid signature for given message } m\}$$

and

$$B = \{(s', t') \in L_h^{NT} \mid (s', t') \text{ is a valid forged signature for given message } m\},$$

respectively.

Remark 3. The EESS#1 standard introduces the centering method in the computation of centered norm [17, 18]. This centering method means that if the center of t not reduced modulo q is near to $\frac{q}{2}$ or $-\frac{q}{2}$, then the coefficients of t are properly shifted before being reduced modulo q. Because this centering method removes any effect of wrapping, if we use this method, then our analysis always holds.

α_i	*Success Prob(B|A)*
1	0.836
2	0.644
⋮	⋮
7	0.707
8	0.889
9	0.852
⋮	⋮
63	0.167
64	0.165

Table 1. Approximate forgery probability when $N = 251, q = 128$

4 Repairing NTRUSign

In this section we present a simple way in order to avoid the weakness in the NTRUSign signature scheme. The strategy for repairing NTRUSign is to make the signing transformation one-to-one corresponding on a given secret key. It can be achieved by adding an annihilating polynomial in the signing step. Our idea is to make the most significant coefficient (*i.e.,* the coefficient of x^{N-1}) of the signature s obtained from the original NTRUSign to be zero. If the distance between the new signature s' computed by this process and the given point is not as close as to the expected distance (*i.e., NormBound*), then we simply add the annihilating polynomial $\sum_{i=0}^{N-1} x^i$ to the signature s' until it becomes to a valid signature.

The repaired version of NTRUSign scheme is as follows:

Signing Signer generates his signature s' on the digital document D

INPUT: private key $\{(f, g), (F, G)\}$ and hashed message (m_1, m_2)
OUTPUT: valid signature s'

1. Obtain the signature s from the original NTRUSign.
2. Set $s' \longleftarrow s - s_{N-1} \sum_{i=0}^{N-1} x^i \pmod q$.
3. While $\| s' - m_1 \|^2 + \| t' - m_2 \|^2 > NormBound^2$ do the following:
   3.1. Set $s' \longleftarrow s' + \sum_{i=0}^{N-1} x^i \pmod q$.
4. Return(s').

Verifying Receiver verifies the signature s'

INPUT: signature s' and sender's public key h

OUTPUT: "Accept" or "Reject"

1. Compute $t' = s' * h \pmod{q}$.
2. If $\| s' - m_1 \|^2 + \| t' - m_2 \|^2 > NormBound^2$, then return("Reject").
3. While $s'_{N-1} \neq 0$:
   3.1. Set $s' \longleftarrow s' - \sum_{i=0}^{N-1} x^i \pmod{q}$.
   3.2. If $\| s' - m_1 \|^2 + \| t' - m_2 \|^2 \leq NormBound^2$, then return("Reject").
4. Return("Accept").

It is obvious that our modification does not degenerate the security of the original NTRUSign scheme. Actually two problems based on original NTRUSign and repaired NTRUSign are computationally equivalent. Although our proposed attack cannot be applied for repaired NTRUSign anymore, we do not know whether the repaired version of NTRUSign is non-malleable. It is an open problem to prove that the repaired NTRUSign is non-malleable signature scheme.

5 Concluding Remarks

In this paper we described a weakness of NTRUSign digital signature scheme that can cause significant problems in some real applications if one is not aware of it. We showed that NTRUSign signature scheme is not secure in terms of strongly existential forgeable, thus it is malleable. This notion allows an adversary to find new signatures for a message of his choice, given a signature for this message. This forgery requires a specific polynomial with small coefficient satisfying its norm value equal to zero. Even if this forgery does not admit an adversary to change the message, NTRUSign scheme cannot be used for all applications. We also proposed a simple technique to repair the scheme.

References

1. H. Cohen, *A course in computational algebraic number theory*, GTM 138, Springer-Verlag, 1993.
2. L. Granboulan, "How to repair ESIGN", SCN'02, LNCS, Vol.2576, Springer-Verlag, pp.234-240, 2003.
3. C. Gentry, J. Jonsson, J. Stern, and M. Szydlo, "Cryptanalysis of the NTRU Signature Scheme (NSS) from Eurocrypt '01" Advances in Cryptology-Asiacrypt '01, LNCS, Vol.2248, Springer-Verlag, pp.123-131, 2001.
4. C. Gentry and M. Szydlo, "Cryptanalysis of the Revised NTRU Signature Scheme", Advances in Cryptology-Eurocrypt '02, LNCS, Vol.2332, pp.299-320, Springer-Verlag, 2002.
5. S. Goldwasser, S. Micali, and R. Rivest, "A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks", SIAM Journal of Computing, pp.281-308, 1998.
6. J. Hoffstein, J. Pipher, and J. Silverman, "Enhanced Encoding and Verification Methods for the NTRU Signature Scheme", NTRU Technical Note #017, 2001. Available from http://www.ntru.com.

7. J. Hoffstein, N. Graham, J. Pipher, J. Silverman, and W. Whyte, "NTRUSign: Digital Signatures Using the NTRU Lattice Preliminary Draft 2", Available from http://www.ntru.com.
8. J. Hoffstein, N. Graham, J. Pipher, J. Silverman, and W. Whyte, "NTRUSign: Digital Signatures Using the NTRU Lattice", CT-RSA'03, LNCS, Vol.2612, Springer-Verlag, pp.122-140, 2003.
9. J. Hoffstein, J. Pipher, and J. Silverman, "NTRU: A Ring-Based Public Key Cryptosystem", in Algorithmic Number Theory (ANTS III), LNCS, Vol.1423, Springer-Verlag, pp.267-288, 1998.
10. J. Hoffstein, J. Pipher, and J. Silverman, "NSS: An NTRU Lattice-Based Signature Scheme", Advanced in Cryptology-Eurocrypt '01, LNCS, Vol.2045, Springer-Verlag, pp.123-137, 2001.
11. A. Joux and G. Martinet, "Some Weaknesses in Quartz Signature Scheme", NESSIE public reports, NES/DOC/ENS/WP5/026/1, 2003.
12. I. Mironov, "A Note on Cryptanalysis of the Preliminary Version of the NTRU Signature Scheme", IACR preprint server, Available from http://eprint.iacr.org/2001/005/.
13. T. Okamoto, E. Fujisaki, and H. Morita, "TSH-ESIGN: Efficient Digital Signature Scheme Using Trisection Size Hash (Submission to P1363a)", 1998.
14. D. Pointcheval and J. Stern, "Security Proofs for Signature Schemes", Advances in Cryptology-Proceedings of Eurocrypt '96, LNCS, Vol.1070, Springer-Verlag, pp.387-398, 1996.
15. J. Silverman, "Wraps, Gaps and Lattice Constants" NTRU Technical Report #011, 2001, Available from http://www.ntru.com.
16. J. Stern, D. Pointcheval, J. Lee, and N. Smart, "Flaws in Applying Proof Methodologies to Signature Schemes", Advances in Cryptology-Crypto'02, LNCS, Vol.2442, Springer-Verlag, pp.93-110, 2002.
17. Consortium for Efficient Embedded Security. Efficient Embedded Security Standard (EESS)#1: Implementation Aspects of NTRUEncrypt and NTRUSign. Available from http:// www.ceesstandards.org.
18. Consortium for Efficient Embedded Security. Efficient Embedded Security Standard (EESS)#1: Draft 2.0. Previously on http://www. ceesstandards.org.