Analysis of NTRUSign signature scheme

SungJun Min*, Go Yamamoto**, Kwangjo Kim*

*Information and Communications University (ICU)

**NTT, Information Sharing Platform Laboratories

Abstract

A new type of signature scheme, called NTRUSign, based on solving the approximately closest vector problem in an NTRU lattice was proposed in [7], [8]. 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.

I. Introduction

Recently, Hoffstein *et al.* introduced a new type of authentication and digital signature scheme called NTRUSign at CT-RSA'03 [7]. While traditional signature schemes are based on hard problem such as factoring problem 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. 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. In this paper, we show that this signature scheme does not contain one of important cryptographic properties that the signature scheme should guarantee, *non-malleability* and suggest a deterministic attack method how an attacker can generate new valid signatures from previous signed message.

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 and R-NSS have been proposed [6], [10]. A fast authentication and digital signature schemes called NSS, based on the same underlying hard problem and using key 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 in the preliminary cryptographic standard document EESS [18]. Although R-NSS was significantly stronger than the previous version(NSS), it was proved that key recovery attack could be mounted [4]. Later on, Hoffstein *et al.* proposed a new NTRU based signature scheme called NTRUSign using NTRU lattices. This paper 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 a second signature of the message from any message-signature pair. In this case, one cannot distinguish the second one 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 permits that the signature scheme cannot be used for all kinds of applications. For example, if one would like to apply to electronic cash, finding a second valid signature for a bill should be impossible.

Our Result In this paper, we show how a passive adversary who observes only a valid message-signature pair can generate another signature. The main idea of this forgery is to use specific polynomials of which norm value is zero. While this weakness might be overlooked for some applications, NTRUSign is not secure from the point of the non-malleability against known message attack. The notion of this security is well described in [16].

Organization The rest of this paper is organized as follows: In Section II, 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 will be briefly described. In Section III we show how an attacker can forge an additional signature for a message already signed by using some specific polynomials. Finally, we make a concluding remark in Section IV.

II. Description of NTRUSign Algorithm

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

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

For the sake of simplicity, we will use the same notation for the polynomial $a(x)$ and the vector $\boldsymbol{a}$. The product of two polynomials $\boldsymbol{a}$ and $\boldsymbol{b}$ in R is simply calculated by $\boldsymbol{a} * \boldsymbol{b} = \boldsymbol{c}$, 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 $R_q = \mathbf{Z}_q[x]/(x^{N-1})$, where the coefficients are reduced by modulo q, where q is typically a power of 2 *e.g.*, 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 *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 digital document being signed.
- The signature demonstrates an ability to solve a general closest vector problem in the lattice.

NTRUSign Algorithm

NTRUSign digital signature scheme works as follows:

1) System Parameters

① N: a (prime) dimension.

② q: a modulus, d_f, d_g: key size parameters.

③ *NormBound*: a bound parameter of verification.

2) 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 $f * G - g * F = q$.

3) 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 The signature on D is a vector $(s, t) \in L_h^{NT}$, which is very close to $m = (m_1, m_2)$.
$s \equiv f * B + F * b \pmod{q}$
$t \equiv g * B + G * b \pmod{q}$.

④ The polynomial s is the signature on the digital document D for the public key h.

4) Verification Step For a given signature s and document D, verifier should do the following:

① Hash the document D to recreate (m_1, m_2).

② With the signature s and public key h, compute the corresponding polynomial $t \equiv s * h \pmod{q}$. (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 it 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 $R = \mathbf{Z}[x]/(x^{N-1})$. Then the *centered norm* of $a(x)$ is defined by
$$\|a(x)\|^2 = \sum_{i=0}^{N-1}(a_i - \mu_a)^2 = \sum_{i=0}^{N-1} a_i^2 - \frac{1}{N}\left(\sum_{i=0}^{N-1} a_i\right)^2,$$
where $\mu_a = \frac{1}{N}\sum_{i=0}^{N-1} a_i$ 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, 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 signer is using his short basis $\{(f, g), (F, G)\}$ to find approximate solutions to the closest vector problem:

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

$$= \left\lfloor (m_1\ m_2)\begin{pmatrix} f & g \\ F & G \end{pmatrix}^{-1}\right\rceil \begin{pmatrix} f & g \\ F & G \end{pmatrix}$$

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 that provided by 1,024 bit RSA [8].

Ⅲ. 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 a second valid signature for a 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 usage. 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 in verification step. The properties of the centered norm will be employed to induce a new signature from a given signature. The following lemma describes the centered norm properties.

Lemma 3 Let R be a quotient polynomial ring $\mathbf{Z}[x]/(x^{N-1})$. Then

① For randomly chosen polynomials $a(x)$ and $b(x)$ in R, $\|a(x)*b(x)\| \approx \|a(x)\| * \|b(x)\|$, that is, the centered norm is quasi- multiplicative.

② In R_q, there exist exactly q polynomials $\alpha(x)$ such that $\|\alpha(x)\| = 0$.

③ If $\|\alpha(x)\| = 0$, then $\|\alpha(x)*\beta(x)\| = 0$ for every polynomial $\beta(x) \in R$.

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

Hoffstein *et al.* argued that forgery of a signature in NTRUSign is equivalent to the ability 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 introduced the notion of existential forgery against chosen-message attacks for public key signature scheme [5]. This notion has become the *de facto* security requirement for all the digital signature algorithms. 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 restricts an adversary to forge an additional signature for a message which might 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 *Remark* and *Lemma*. Remind that all coefficients of polynomials are reduced by modulo q.

Remark 4 Let α be an annihilating

polynomial. Then $\| a+\alpha \| \approx \| a \|$ for randomly chosen polynomial $a \in R$.

If both "reduced form" and "not reduced form" of polynomial $a+\alpha$ are equal, then the centered norm values of $\| a \|$ and $\| a+\alpha \|$ are exactly the same. The differences between $\| a+\alpha \|$ and $\| a \|$ 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 by 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.

We will see some results induced from the properties of an annihilating polynomial. For any polynomial $f=(f_0, f_1, \cdots, f_{N-1}) \in R$, $\nu(f)$ denotes the sum of all coefficients of f modulus q, that is, $\nu(f) = \sum_{i=0}^{N-1} f_i \pmod q \in Z_q$. For any $f \in R$, the product $f*\alpha$ can be presented by $\nu(f)\alpha$, where α is an annihilating polynomial.

Lemma 5 Let f and g be two polynomials in R. Then

① $\nu(f)\nu(g) \equiv \nu(f*g) \bmod q$.

② $\nu(f^{-1}) \equiv \nu(f)^{-1} \bmod q$ if f has an inverse in R_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 $\alpha_0 \nu(f)^{-1}\nu(g) \pmod q$ is still a small integer, 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 5*.

Remark 6 In the suggested parameters given in [8], one has $\nu(f)=-55$ and $\nu(g)=-57$ for $(d_f, d_g)=(73,71)$. In this case one can choose $\alpha=8\sum_{i=0}^{N-1} x^i$ so that $h*\alpha=-8\sum_{i=0}^{N-1} x^i$. For a given signature s generated under these parameters, $(s', t')=(s+8\sum_{i=0}^{N-1} x^i, t-8\sum_{i=0}^{N-1} x^i)$ 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, replacing (s, t) with $(s+8\sum_{i=0}^{N-1} x^i, t-8\sum_{i=0}^{N-1} x^i)$ always makes the left hand side of the inequality,

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

to be still the same. 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 $Z_{128}[x]/(x^{251}-1)$ 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

$$\sigma = \sqrt{NormBound^2/N} \approx 18.9.$$

For two sets $P' = \{s \in P \mid \| s \| < 300^2\}$ and $P'' = \{s \in P' \mid \| s+8\sum_{i=0}^{N-1} x^i \| < 300^2\}$, we obtained the result that the set P' consists of 20,650 distinct elements and that P' and P'' coincide exactly. From this, we can get the following theorem:

Theorem 7 NTRUSign scheme is malleable. In other words, it is not secure against strongly existential forgery.

Although the NTRUSign signature scheme is deterministic, several valid signatures are associated to the same message. For example, given valid signature (s, t) we can always get second valid signature

$$(s0, t0) = (s+8\sum_{i=0}^{N-1} x^i, t-8\sum_{i=0}^{N-1} x^i).$$

This property allows an adversary to find an additional signature for a message of his choice, previously signed by the oracle without solving the hard closest vector problems. This attack represents a failure of the strong unforgeability security, thus malleability.

Ⅳ. Concluding Remarks

In this paper we have described a weakness of the NTRUSign digital signature scheme that can cause significant problems in some real applications if one is not aware of it. We have 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 raise a new problem to find a method that makes the NTRUSign non-malleable in the near future.

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. Mirinov, "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.