An Algebraic Broadcast Attack against NTRU

Jintai Ding[1,2], Yanbin Pan[3], and Yingpu Deng[3]

[1] Chongqing University,
[2] Department of Mathematical Sciences, University of Cincinnati
[3] Key Laboratory of Mathematics Mechanization,
Academy of Mathematics and Systems Science, Chinese Academy of Sciences
jintai.ding@gmail.com, {panyanbin,dengyp}@amss.ac.cn

Abstract. In this paper, we propose an algebraic broadcast attack against NTRU, which recovers a single message encrypted multiple times using different NTRU public keys. Namely, when a message is broadcasted, under some reasonable assumptions, our attack can be completed in polynomial time and space. To the best of our knowledge, this is the first successful broadcast attack against NTRU.

Keywords: Broadcast attack, NTRU, lattice-based cryptosystems, LWE.

1 Introduction

The NTRU cryptosystem of Hoffstein, Pipher, Silverman [12] is one of the most practical public key schemes, which is an IEEE 1363.1 Standard [18]. It features reasonably short, easily created keys, high speed, and low memory requirements.

Coppersmith and Shamir [3] showed the security of NTRU is related to the hardness of certain lattice problems. Although there are many attacks, like the ciphertext-only attacks [3,19,15,13], no significant weakness has seemed to be yet found on NTRU encryption. The most effective attacks may have been the chosen-ciphertext attacks[14,8], most of which utilize the NTRU's decryption failures. When a single message is encrypted multiple times using a NTRU public key, Hoffstein and Silverman [10] proposed a multiple transmission attack.

Another type of attack, the broadcast attack, was first proposed by Hästad [9] in 1988. The attack enables an attacker to recover the single message sent by the sender to multiple recipients, who use the same type of cryptosystrems but with different keys, without requiring any knowledge of the recipients' secret keys. Another effective attack is the hybrid attack [13,17].

In 2009, Plantard and Susilo [22] first considered the broadcast attack against the lattice-based public key cryptosystems. They constructed many lattices that share the same short vector carrying the information of a single message. By intersecting these lattices, they then gave some heuristic attacks using the lattice reduction algorithm. However, they also showed that their attacks do not apply to NTRU, since the lattices in general do not share the same short target vector.

W. Susilo, Y. Mu, and J. Seberry (Eds.): ACISP 2012, LNCS 7372, pp. 124–137, 2012.

In this paper, we propose an algebraic broadcast attack against NTRU, which is based on the new algorithms in [5,6,7,1]. Instead of the lattice reduction algorithm, we present a new algebraic algorithm to complete the attack. The new algorithm involves nonlinearization and linearization, and can also be used to solve the learning with errors problem with bounded errors.

Since there are many variants of NTRU, we give our attacks against the main instantiations: NTRU-1998 [12], NTRU-2001 [11] (with an odd d_g whose definition can be found in Section 2.2) and NTRU-2005 [16]. Under some reasonable assumption, to complete the attacks, we need gather $O(N)$ (resp. $O(N^2)$) recipients' public keys and ciphertexts, and solve a set of $O(N^2)$ (resp. $O(N^3)$) linear equations with $O(N^2)$ (resp. $O(N^3)$) variables for NTRU-2001 (with an odd d_g) and NTRU-2005 (resp. NTRU-1998), where N is the main parameter of NTRU. The attacks are efficient since they may be completed in polynomial time and space, but these attacks become more difficult in practice as N increases.

The hybrid of these attacks and some other attacks, for example, the lattice reduction attack and the meet-in-the-middle attack, may lead better attacks. With XL [4,2] type of algorithms, we can further reduce the number of recipients, but increase the computation complexity. It remains an open problem to find an efficient broadcast attack with a constant number of recipients. Compared with the chosen-ciphertext attacks, our attacks don't need the decryption oracle or the decryption failures. Compared with the multiple transmission attack, our attacks allow that the recipients' public keys are different. Different public keys make attacking NTRU a hard task. Our broadcast attacks do not work for more sophisticated padding schemes of NTRU. We can also use our method to attack some other lattice-based cryptosystems which have a similar linear structure to NTRU and bounded random perturbations over $\mathbb{Z}_q$.

To attack NTRU-1998 and NTRU-2001 with an odd d_g, we have to extend the algorithm in [5,6,7,1] over finite field onto the ring $\mathbb{Z}_{2^k}$. Further more, we can derive N linear equations with fewer variables from every recipient instead of one linear equation with much more variables. Thus, we use much fewer recipients, and solve much fewer linear equations for much fewer variables to complete the attacks.

The paper is organized as follows. Section 2 gives some preliminaries. Section 3 describes our broadcast attacks. Section 4 presents the conclusion.

2 Preliminaries

Let $\mathbb{Z}$ be the integer ring, $\mathbb{Z}_q$ the residue class ring $\mathbb{Z}/q\mathbb{Z}$, $\mathbb{F}_q$ the finite fields $\mathbb{Z}_q$ when q is a prime. We use bold letters to denote vectors, in column notation. For a vector $\mathbf{v}$, we denote by $\mathbf{v}_i$ the i-th entry of $\mathbf{v}$.

2.1 The Learning with Errors Problem

The learning with errors (LWE) problem introduced by Regev [23] has many applications in constructing cryptosystems with security proofs.

An LWE problem has a parameter n, a prime modulus q, and an error probability distribution κ on the finite field $\mathbb{F}_q$. Let $\prod_{\mathbf{m},\kappa}$ on $\mathbb{F}_q$ be the probability distribution obtained by selecting an element $\mathbf{a}$ in $\mathbb{F}_q^n$ randomly and uniformly, choosing $r \in \mathbb{F}_q$ according to κ, and outputting $(\mathbf{a}, < \mathbf{a}, \mathbf{m} > +r)$, where $+$ is the addition that is performed in $\mathbb{F}_q$. We say that an algorithm solves LWE with modulus q and error distribution κ, if, for any $\mathbf{m}$ in $\mathbb{F}_q^n$, with an arbitrary number of independent samples from $\prod_{\mathbf{m},\kappa}$, it outputs $\mathbf{m}$ with high probability.

2.2 NTRU

We present a brief description of the NTRU-1998 cryptosystems. For more details see [12]. The NTRU-1998 cryptosystem depends on three integer parameters (N, p, q) and four sets $\mathcal{L}_f, \mathcal{L}_g, \mathcal{L}_r, \mathcal{L}_m$ of polynomials of degree $N-1$ with small integer coefficients. We choose p, q such that $\gcd(p, q) = 1$ and p is much smaller than q. Denote the ring $\mathbb{Z}[x]/(x^N - 1)$ by R and the multiplication in R by $*$. Every element in R can be represented as a polynomial. For example, $f \in R$ can be represented as $f = \sum_{i=0}^{N-1} f_i x^i$. We work over the ring R.

Key Generation: Step 1. Choose $f \in \mathcal{L}_f, g \in \mathcal{L}_g$ such that there exist $F_q, F_p \in R$ satisfying $f * F_q = 1 \bmod q$ and $f * F_p = 1 \bmod p$. **Step 2.** Let $h = p * F_q * g \bmod q$.
Public Key: h, p, q.
Private Key: f, F_p.
Encryption: To encrypt a message $m \in \mathcal{L}_m$, first we choose an $r \in \mathcal{L}_r$ randomly, then compute the ciphertext: $c = h * r + m \bmod q$.
Decryption: First we compute $a = f * c \bmod q == pg * r + f * m \bmod q$, then we choose the coefficients of a in the interval from $-\frac{q}{2}$ to $\frac{q}{2}$. By the fact that all the coefficients of $pg * r + f * m$ may be in the interval from $-\frac{q}{2}$ to $\frac{q}{2}$, we almost get $a = pg * r + f * m$. Then we recover the message m by computing $m = F_p * a \bmod p$.

As pointed in [20], analyzing NTRU is a tricky task, since there are several variants of NTRU. We may use totally different ways to attack different variants of NTRU instead of a uniform one. We summarize the main instantiations of NTRU in the table below as in [20]:

Variant	q	p	$\mathcal{L}_f$	$\mathcal{L}_g$	$\mathcal{L}_m$	$\mathcal{L}_r$	F	Ref
NTRU-1998	$2^k \in [\frac{N}{2}, N]$	3	$L(d_f, d_f - 1)$	$L(d_g, d_g)$	L_m	$L(d_r, d_r)$	-	[12]
NTRU-2001	$2^k \in [\frac{N}{2}, N]$	$2 + x$	$1 + p * F$	$\mathcal{B}(d_g)$	$\mathcal{B}$	$\mathcal{B}(d_r)$	$\mathcal{B}(d_F)$	[11]
NTRU-2005	prime	2	$1 + p * F$	$\mathcal{B}(d_g)$	$\mathcal{B}$	$\mathcal{B}(d_r)$	$\mathcal{B}(d_F)$	[16]

where $L_m = \{m \in R : m$ has coefficients lying between $-\frac{1}{2}(p-1)$ and $\frac{1}{2}(p-1)\}$, $L(d_1, d_2) = \{F \in R : F$ has d_1 coefficients equal $1, d_2$ coefficients equal -1, the rest $0\}$, $\mathcal{B}$ denotes the set of all polynomials with binary coefficients, $\mathcal{B}(d) = \{F \in R : F$ has d coefficients equal 1, the rest $0\}$.

2.3 Transforming NTRU into Its Linear Form

In NTRU, a polynomial $f = \sum_{i=0}^{N-1} f_i x^i \in R$ can also be represented as a vector: $\mathbf{f} = (f_0, f_1, \cdots, f_{N-1})^T$. The multiplication of f and g can be represented as $\begin{pmatrix} f_0 & f_{N-1} & \cdots & f_1 \\ f_1 & f_0 & \cdots & f_2 \\ \vdots & \vdots & \ddots & \vdots \\ f_{N-1} & f_{N-2} & \cdots & f_0 \end{pmatrix} \begin{pmatrix} g_0 \\ g_1 \\ \vdots \\ g_{N-1} \end{pmatrix}$. The result is the vector corresponding to $f * g$ in R. Then, for the equation in R: $c = h * r + m \bmod q$, we have the linear form

$$\mathbf{c} = H\mathbf{r} + \mathbf{m} \bmod q, \tag{1}$$

where $H = \begin{pmatrix} h_0 & h_{N-1} & \cdots & h_1 \\ h_1 & h_0 & \cdots & h_2 \\ \vdots & \vdots & \ddots & \vdots \\ h_{N-1} & h_{N-2} & \cdots & h_0 \end{pmatrix}$.

Proposition 1. *For a uniformly random instance of NTRU-1998 (also NTRU-2001 with an odd d_g, NTRU-2005), for any message $\boldsymbol{m}$, ciphertext $\boldsymbol{c}$ and the corresponding random vector $\boldsymbol{r}$, we can find $\widehat{H} \in \mathbb{Z}_q^{N\times N}$ and $\boldsymbol{b} \in \mathbb{Z}_q$ with very high probability in polynomial time, without knowing $\boldsymbol{m}$ and $\boldsymbol{r}$, such that*

$$\widehat{H}\boldsymbol{m} + \boldsymbol{r} = \boldsymbol{b} \ mod\ q.$$

Proof. For NTRU-2001 with an odd d_g and NTRU-2005, H is invertible in $\mathbb{Z}_q^{N\times N}$ with very high probability. So we can get easily from (1): $H^{-1}\mathbf{m} + \mathbf{r} = H^{-1}\mathbf{c} \bmod q$. Let $\widehat{H} = H^{-1}$, $\mathbf{b} = H^{-1}\mathbf{c}$, then we have: $\widehat{H}\mathbf{m} + \mathbf{r} = \mathbf{b} \bmod q$.

However, For NTRU-2001 with an even d_g, H is not invertible. It seems that we need some extra restrictions on the other parameters, for example, d_r, to get a similar result as the proposition. That's why we exclude the case.

For NTRU-1998, notice that H is not invertible in $\mathbb{Z}_q^{N\times N}$ either. However, for any public key h in NTRU-1998, we can usually find a polynomial $h' \in R$ with overwhelming probability such that for any $r \in L(d_r, d_r)$: $h' * h * r = r \bmod q$.

We say h is pseudo-invertible as in [21].

As pointed in [21], we can find h' in polynomial time as follows. Since $R_q = \mathbb{Z}_q[x]/(x^N - 1)$ is isomorphic to $P_1 \times P_2$ where $P_1 = \mathbb{Z}_q[x]/(x - 1)$ and $P_2 = \mathbb{Z}_q[x]/(x^{N-1} + x^{N-2} + \cdots + 1)$, we have $\phi : R_q \to P_1 \times P_2$.

Since $h(1) = 0 \bmod q$, we have $\phi(h) = (0, \bar{h})$ where $\bar{h}$ denotes the reduction of h modulo $x^{N-1} + x^{N-2} + \cdots + 1$. With high probability, $\bar{h}$ is invertible in P_2. We denote its inverse in P_2 by $\widetilde{h}$. Considering the polynomial $h' = \phi^{-1}((1, \widetilde{h}))$ in R_q, it satisfies $h' * h * r = r \bmod q$ for $r \in L(d_r, d_r)$.

Let $H' = \begin{pmatrix} h'_0 & h'_{N-1} & \cdots & h'_1 \\ h'_1 & h'_0 & \cdots & h'_2 \\ \vdots & \vdots & \ddots & \vdots \\ h'_{N-1} & h'_{N-2} & \cdots & h'_0 \end{pmatrix}$, then we have: $H'\mathbf{m} + \mathbf{r} = H'\mathbf{c} \bmod q$, from (1). Similarly, let $\widehat{H} = H'$, $\mathbf{b} = H'\mathbf{c}$, then we have $\widehat{H}\mathbf{m} + \mathbf{r} = \mathbf{b} \bmod q$.

Obviously, from Proposition 1, for $i = 0, \cdots N-1$, we get N linear equations: $\sum_{j=0}^{N-1} \widehat{H}_{i+1,j+1}\mathbf{m}_j + \mathbf{r}_i = \mathbf{b}_i \bmod q$.

3 A Broadcast Attack against NTRU

Suppose there is a sender and n recipients. All recipients use NTRU cryptosystems with the same parameters N, q, p but different public/private keys. The sender encrypts the single message m with each recipient's public key with independent $r \in \mathcal{L}_r$ respectively, and sends the n ciphertexts to corresponding recipients. The broadcast attack is to recover m with these n ciphertexts. More precisely, the attacker wants to recover m from the n equations:

$$h_{(i)} * r_{(i)} + m = c_{(i)} \bmod q$$

where $h_{(i)}$ is the i-th recipient's public key.

For each recipient, by Proposition 1, we have N linear equations. For each linear equation, taking $\sum_{j=0}^{N-1} \widehat{H}_{1,j+1}\mathbf{m}_j + \mathbf{r}_0 = \mathbf{b}_0 \bmod q$ as an example, let $\mathbf{a} = (\widehat{H}_{1,1}, \widehat{H}_{1,2}, \cdots, \widehat{H}_{1,N})$, we have a pair, $(\mathbf{a}, \mathbf{b}_0 =< \mathbf{a}, \mathbf{m} > +\mathbf{r}_0)$, which can be seen as a sample from an LWE oracle. So, recovering m from the n recipients is similar to solving an LWE problem for $\mathbf{m}$ from nN samples.

3.1 The Basic Algorithm for LWE with Bounded Errors

An LWE problem is with bounded errors if the error probability distribution is on a proper subset $ES = \{e_1, e_2, \cdots, e_D\}$ of $\mathbb{F}_q$ with fixed $D (D < q)$. The main steps of the algorithm for LWE with bounded errors are:

1. Nonlinearization. For any sample $(\mathbf{a}, < \mathbf{a}, \mathbf{m} > +r)$, let $b' =< \mathbf{a}, \mathbf{m} > +r$. We know that $\mathbf{m}$ satisfies: $\sum_{i=1}^{N} \mathbf{a}_i\mathbf{x}_i + r = b'$.
   So $\mathbf{m}$ also satisfies the corresponding nonlinear equation

$$\prod_{r_k \in ES} (\sum_{i=1}^{N} \mathbf{a}_i\mathbf{x}_i + r_k - b') = 0. \tag{2}$$

   Note here that D needs to be less than q, otherwise the equation above will be totally trivial, namely the so-called field equations: $\mathbf{x}_i^q = \mathbf{x}_i$.
2. Linearization. We will solve Equation (2) by linearization. Notice that the total degree of every monomial of $\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N$ in (2) is at most D. We assign each of such monomials a new variable $\mathbf{y}_i$. Let $Q(N, D) = \binom{N+D}{N}$. We know that the number of $\mathbf{y}_i$'s, denoted by d, is at most $Q(N, D) - 1$. What's more, we can assign $\mathbf{x}_i$ to be $\mathbf{y}_{d-N+i}$, then we transform (2) into a linear equation: $\sum_i \mathbf{c}_i\mathbf{y}_i = b$, where $\mathbf{c}_i$ is the corresponding coefficient of $\mathbf{y}_i$ in (2).

3. Solving. Since there are d $\mathbf{y}_i$'s, when we have enough linear equations from above to find a $d \times d$ nonsingular matrix L with good probability such that, $L\mathbf{y} = \mathbf{b}$, where $\mathbf{b}$ is a constant vector, we can find $\mathbf{y}$ by solving the set of linear equations. Then we can solve $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_N)^T$.

We next estimate the expectation of the number of samples in the algorithm under an optimistic assumption that $\mathbf{v}$ is chosen from $\mathbb{F}_q^d$ uniformly and independently.

Let X be the number of samples until we first have $|F| = d$, X_i be the number of samples until we first have $|F| = i$ starting from when we first have $|F| = i-1$, where $i = 1, 2, \cdots, d$. Obviously, $X = \sum_{i=1}^{d} X_i$.

When $|F| = i-1$, the probability we have a new $\mathbf{v}$ that can't be generated by the vectors in F is $1 - \frac{q^{i-1}}{q^d}$. Hence, the expectation of X_i is $\frac{1}{1-\frac{q^{i-1}}{q^d}} = \frac{q^d}{q^d - q^{i-1}}$. Then, $E(X) = \sum_{i=1}^{d} E(X_i) = \sum_{i=1}^{d} \frac{q^d}{q^d - q^{i-1}}$.

We just give an upper bound for $E(X)$ here. Since $\frac{q^d}{q^d - q^{i-1}} < \frac{q^d}{q^d - q^{d-1}} = \frac{q}{q-1}$, we have $E(X) < \frac{q}{q-1}d$.

We implemented the algorithm. When we make $d + O(N)$ queries, we have never failed to obtain an invertible L in all the extensive experiments (thousands and $q > 3$), hence to solve the LWE problem. So it is reasonable to believe the algorithm succeeds with very high probability.

3.2 The Broadcast Attack against NTRU-1998

For NTRU-1998, $ES = \{-1, 0, 1\}$ and $\mathbf{m} \in \{-1, 0, 1\}^N$. Since $q = 2^k$, notice that the algorithm above works over the finite field but not the ring $\mathbb{Z}_{2^k}$ and L is invertible over $\mathbb{Z}_{2^k}^{d\times d}$ if and only if L is invertible over $\mathbb{F}_2^{d\times d}$. So, we try to work over the finite field $\mathbb{F}_2$. A nature idea is to transform these equations into the ones over $\mathbb{F}_2$ by simply mapping each coefficient into $\mathbb{F}_2$. However, if we did so, we would get $ES = \{0, 1\}$ which is not a proper set of $\mathbb{F}_2$. So we have to involve some new ideas to solve the problem.

Since we will work over $\mathbb{F}_2$, we don't recover $\mathbf{m}$ directly, but to find another $\mathbf{m}' \in \mathbb{F}_2^N$ such that $\mathbf{m} = \mathbf{m}' \bmod 2$. We first show that this is enough for our attack.

Suppose we have already known $\mathbf{m}' = \mathbf{m} \bmod 2$, we know exactly which entries of $\mathbf{m}$ are zero. Without loss of generality, we assume $\mathbf{m}_i = 0$ for $i = 0, 1, \cdots, t-1$. Then for every recipient, we eliminate the first t columns of the corresponding $\widehat{H}$, and denote the remaining $N \times (N-t)$ matrix by $\widehat{H}'$. We have

$$\widehat{H}'(\mathbf{m}_t, \cdots, \mathbf{m}_{N-1})^T + \mathbf{r} = b \bmod 2^k,$$

where $\mathbf{m}_i$ is either -1 or 1 and q is a power of 2. So

$$\widehat{H}'(\mathbf{m}_t + 1, \cdots, \mathbf{m}_{N-1} + 1)^T + \mathbf{r} = b + \widehat{H}'(1, \cdots, 1)^T \bmod 2^k,$$

where $\mathbf{m}_i + 1$ is either 0 or 2. Notice that $\mathbf{r}_i = 0$ if and only if the $i-$th entry of $b+\widehat{H}'(1,\cdots,1)^T$ is congruent to 0 modulo 2. We will get a set of $\mathbf{r}_i$'s which equal 0. For each of these $\mathbf{r}_i$'s, we get a linear equation: $\sum_{j=0}^{N-t-1} \widehat{H}'_{i+1,j+1}\mathbf{m}_{t+j} = \mathbf{b}_i$.

For each recipient, we can have $N-2d_r$ such linear equations. When collecting enough equations, we can obtain these $\mathbf{m}_i$'s.

We next show that how to find $\mathbf{m}'$ such that $\mathbf{m} = \mathbf{m}' \bmod 2$.

From (2), we know $\mathbf{m}$ satisfies $\prod_{r_k\in\{-1,0,1\}}(\sum_{i=1}^N \mathbf{a}_i\mathbf{x}_i + r_k - b') = 0 \bmod 2^k$, i.e.

$$\begin{aligned}&\sum_{i=1}^N \mathbf{a}_i^3\mathbf{x}_i^3 + 6\sum_{i<j<k}\mathbf{a}_i\mathbf{a}_j\mathbf{a}_k\mathbf{x}_i\mathbf{x}_j\mathbf{x}_k + 3\sum_{i\neq j}\mathbf{a}_i^2\mathbf{a}_j\mathbf{x}_i^2\mathbf{x}_j\\ &-6\sum_{i<j}b'\mathbf{a}_i\mathbf{a}_j\mathbf{x}_i\mathbf{x}_j - 3\sum_{i=1}^N b'\mathbf{a}_i^2\mathbf{x}_i^2 + \sum_{i=1}^N(3(b')^2\mathbf{a}_i - \mathbf{a}_i)\mathbf{x}_i\\ &= (b')^3 - b' \bmod 2^k.\end{aligned}$$

Since $\mathbf{x}_i^3 = \mathbf{x}_i$ for $\mathbf{x}_i \in \{-1,0,1\}$ and there exists $\bar{\mathbf{x}}_i \in \{0,1\}$ such that $\mathbf{x}_i^2 = \mathbf{x}_i + 2\bar{\mathbf{x}}_i$, substituting $\mathbf{x}_i$ for $\mathbf{x}_i^3$, and $\mathbf{x}_i + 2\bar{\mathbf{x}}_i$ for $\mathbf{x}_i^2$, we have

$$\begin{aligned}&6\sum_{i<j<k}\mathbf{a}_i\mathbf{a}_j\mathbf{a}_k\mathbf{x}_i\mathbf{x}_j\mathbf{x}_k + \sum_{i<j}(3\mathbf{a}_i^2\mathbf{a}_j + 3\mathbf{a}_j^2\mathbf{a}_i - 6b'\mathbf{a}_i\mathbf{a}_j)\mathbf{x}_i\mathbf{x}_j\\ &+6\sum_{i\neq j}\mathbf{a}_i^2\mathbf{a}_j\bar{\mathbf{x}}_i\mathbf{x}_j - 6\sum_{i=1}^N b'\mathbf{a}_i^2\bar{\mathbf{x}}_i + \sum_{i=1}^N(3(b')^2\mathbf{a}_i - \mathbf{a}_i + \mathbf{a}_i^3 - 3b'\mathbf{a}_i^2)\mathbf{x}_i\\ &= (b')^3 - b' \bmod 2^k.\end{aligned}$$

Since $t^3 = t \bmod 2$ and $t^2 = t \bmod 2$ for any integer t, obviously there is a positive integer $u \geq 1$ such that 2^u divides the greatest common divisor of all the coefficients but 2^{u+1} can not. If $u < k$, we have

$$\begin{aligned}&\frac{3}{2^{u-1}}\sum_{i<j<k}\mathbf{a}_i\mathbf{a}_j\mathbf{a}_k\mathbf{x}_i\mathbf{x}_j\mathbf{x}_k + \frac{1}{2^u}\sum_{i<j}(3\mathbf{a}_i^2\mathbf{a}_j + 3\mathbf{a}_j^2\mathbf{a}_i - 6b'\mathbf{a}_i\mathbf{a}_j)\mathbf{x}_i\mathbf{x}_j\\ &+\frac{3}{2^{u-1}}\sum_{i\neq j}\mathbf{a}_i^2\mathbf{a}_j\bar{\mathbf{x}}_i\mathbf{x}_j - \frac{3}{2^{u-1}}\sum_{i=1}^N b'\mathbf{a}_i^2\bar{\mathbf{x}}_i + \frac{1}{2^u}\sum_{i=1}^N(3(b')^2\mathbf{a}_i - \mathbf{a}_i + \mathbf{a}_i^3 - 3b'\mathbf{a}_i^2)\mathbf{x}_i\\ &= \frac{1}{2^u}((b')^3 - b') \bmod 2^{k-u}.\end{aligned}$$

As above, we assign each of the monomials a new variable $\mathbf{y}_i$(Notice that $\mathbf{y}_{d-N+i} = \mathbf{x}_i$ for $1 \leq i \leq N$), and transform it into a linear equation: $\sum_i \bar{\mathbf{c}}_i\mathbf{y}_i = \bar{b} \bmod 2^{k-u}$.

Let $\mathbf{c}_i = \bar{\mathbf{c}}_i \bmod 2$ and $b = \bar{b} \bmod 2$. We get a linear equation over $\mathbb{F}_2$: $\sum_i \mathbf{c}_i\mathbf{y}_i = b \bmod 2$.

Collecting enough equations in our experiments, we always get a matrix L, by performing Gaussian elimination, such that

$$L\mathbf{y} = \left(\begin{array}{c|c} 1\cdots & \\ \ddots & * \\ \hline \mathbf{0} & I_{N\times N}\end{array}\right)\left(\begin{array}{c}\mathbf{y}_1\\ \vdots\\ \hline \mathbf{x}\end{array}\right) = \mathbf{b}.$$

whereas L is not invertible, $\mathbf{x} = (\mathbf{x}_1, \cdots, \mathbf{x}_N)^T$. Nevertheless, it is enough for us to recover $\mathbf{m}'$ since we can solve for $\mathbf{x}$.

What's more, we can use the strategy below to get an invertible L with very high probability. Notice that $u = 1$ holds for very high probability. We just use those samples in which $u = 1$. So we have

$$\begin{aligned} &\sum_{i<j<k} \mathbf{a}_i\mathbf{a}_j\mathbf{a}_k\mathbf{x}_i\mathbf{x}_j\mathbf{x}_k + \frac{1}{2}\sum_{i<j}(3\mathbf{a}_i^2\mathbf{a}_j + 3\mathbf{a}_j^2\mathbf{a}_i - 6b'\mathbf{a}_i\mathbf{a}_j)\mathbf{x}_i\mathbf{x}_j \\ &+ \sum_{i\neq j} \mathbf{a}_i\mathbf{a}_j\bar{\mathbf{x}}_i\mathbf{x}_j - \sum_{i=1}^{N} b'\mathbf{a}_i\bar{\mathbf{x}}_i + \frac{1}{2}\sum_{i=1}^{N}(3(b')^2\mathbf{a}_i - \mathbf{a}_i + \mathbf{a}_i^3 - 3b'\mathbf{a}_i^2)\mathbf{x}_i \qquad (3) \\ &= \frac{1}{2}((b')^3 - b') \bmod 2. \end{aligned}$$

Denote by $\mathbf{y}_{(i,j)}$ the new variable corresponding to $\bar{\mathbf{x}}_i\mathbf{x}_j$, by $\mathbf{y}_{\{i,j,s\}}$ the new variable corresponding to $\mathbf{x}_i\mathbf{x}_j\mathbf{x}_s$ and by $\mathbf{c}_{(i,j)}$, $\mathbf{c}_{\{i,j,s\}}$ their coefficients in the equation above. We claim that

Proposition 2. *Denote by S the set $\{i_1, i_2, i_3\}$ where $1 \le i_k \le N$, we have:* **1)**. $\boldsymbol{c}_{(i,j)} = \boldsymbol{c}_{(j,i)}$ *mod* 2; **2)**. $\boldsymbol{c}_{(i,j)} = \sum_{\{i,j\}\subset S} \boldsymbol{c}_S$ *mod* 2.

Proof. The first one is obvious. It remains to prove the second one. Denote by $\widehat{h}(x)$ the polynomial corresponding to any row of the matrix $\widehat{H}$. Since $\widehat{h}(x)$ is invertible over the ring $\mathbb{Z}_{2^k}[x]/(x^N - 1)$, we have $\widehat{h}(1) = 1 \bmod 2$. Hence,

$$\sum_{i=1}^{N} \mathbf{a}_i = \widehat{h}(1) = 1 \bmod 2. \qquad (4)$$

Finally, we have $\mathbf{c}_{(i,j)} = \mathbf{a}_i\mathbf{a}_j \sum_{s=1}^{N} \mathbf{a}_s = \sum_{\{i,j\}\subset S} \mathbf{c}_S + \mathbf{a}_i^2\mathbf{a}_j + \mathbf{a}_i\mathbf{a}_j^2 = \sum_{\{i,j\}\subset S} \mathbf{c}_S + \mathbf{a}_i\mathbf{a}_j + \mathbf{a}_i\mathbf{a}_j = \sum_{\{i,j\}\subset S} \mathbf{c}_S \bmod 2$.

We involve some new variables $\bar{\mathbf{y}}_{\{i,j,s\}} = \mathbf{y}_{\{i,j,s\}} + \sum_{\{i,j\}\subset S}(\mathbf{y}_{(i,j)} + \mathbf{y}_{(j,i)})$.

Since $\mathbf{c}_{(i,j)}\mathbf{y}_{(i,j)} = \sum_{\{i,j\}\subset S} \mathbf{c}_S\mathbf{y}_{(i,j)}$ and $\mathbf{c}_{(j,i)}\mathbf{y}_{(j,i)} = \sum_{\{i,j\}\subset S} \mathbf{c}_S\mathbf{y}_{(j,i)}$, we have

$$\begin{aligned} \sum_{i<j<k} \mathbf{c}_{\{i,j,s\}}\bar{\mathbf{y}}_{\{i,j,s\}} &= \sum_{i<j<k} \mathbf{a}_i\mathbf{a}_j\mathbf{a}_k\mathbf{x}_i\mathbf{x}_j\mathbf{x}_k + \sum_{i\neq j} \mathbf{a}_i\mathbf{a}_j\bar{\mathbf{x}}_i\mathbf{x}_j \bmod 2 \\ &= \sum_{i<j<k} \mathbf{c}_{\{i,j,s\}}\mathbf{y}_{\{i,j,s\}} + \sum_{i\neq j} \mathbf{c}_{(i,j)}\mathbf{y}_{(i,j)} \bmod 2. \end{aligned}$$

Using the new variables $\bar{\mathbf{y}}_{\{i,j,s\}}$ in Equation (3), we can eliminate the old variables $\mathbf{y}_{\{i,j,s\}}$ and $\mathbf{y}_{(i,j)} (i \neq j)$. Now the number of the variables is $d = \binom{N}{3} + \binom{N}{2} + 2N = O(N^3)$.

If we have enough such linear equations such that we can find a $d \times d$ non-singular matrix L satisfying $L\mathbf{y} = \mathbf{b}$, we can find $\mathbf{y}$ by solving the set of linear equations over $\mathbb{F}_2$. In our experiments, we can always find an invertible L. So we can find an $\mathbf{m}'$ such that $\mathbf{m} = \mathbf{m}' \bmod 2$.

Remark 3. Notice that from $\sum_i \bar{\mathbf{c}}_i \mathbf{y}_i = \bar{b} \bmod 2^{k-u}$, we can also get the linear equation: $\sum_i \mathbf{c}_i \mathbf{y}_i = b \bmod 4$, when $k - u > 1$ if we let $\mathbf{c}_i = \bar{\mathbf{c}}_i \bmod 4$ and $b = \bar{b} \bmod 4$. If we have enough linear equations, and use the similar strategy above, we may recover $\mathbf{m}$ directly.

3.3 The Broadcast Attack against NTRU-2001 with an Odd d_g

For NTRU-2001 with an odd d_g, $ES = \{0, 1\}$ and $\mathbf{m} \in \{0, 1\}^N$. As the section above, we can not use the algorithm for LWE directly. We also propose a new algorithm below.

First, from Equation (2), we have $(\sum_{i=1}^N \mathbf{a}_i \mathbf{x}_i - b')(\sum_{i=1}^N \mathbf{a}_i \mathbf{x}_i - b' + 1) = 0 \bmod 2^k$, i.e.

$$\sum_{i=1}^N \mathbf{a}_i^2 \mathbf{x}_i^2 + 2 \sum_{i<j} \mathbf{a}_i \mathbf{a}_j \mathbf{x}_i \mathbf{x}_j + \sum_{i=1}^N (\mathbf{a}_i - 2b' \mathbf{a}_i) \mathbf{x}_i = -(b')^2 + b' \bmod 2^k.$$

Since $\mathbf{x}_i^2 = \mathbf{x}_i$ holds for $\mathbf{x}_i \in \{0, 1\}$, we have

$$2 \sum_{i<j} \mathbf{a}_i \mathbf{a}_j \mathbf{x}_i \mathbf{x}_j + \sum_{i=1}^N ((\mathbf{a}_i^2 + \mathbf{a}_i) - 2b' \mathbf{a}_i) \mathbf{x}_i = -(b')^2 + b' \bmod 2^k.$$

Since $\mathbf{a}_i^2 = \mathbf{a}_i \bmod 2$, there is obviously an integer $u \geq 1$ such that 2^u divides the greatest common divisor of all the coefficients but 2^{u+1} can not. We have

$$\frac{1}{2^{u-1}} \left(\sum_{i<j} \mathbf{a}_i \mathbf{a}_j \mathbf{x}_i \mathbf{x}_j + \sum_{i=1}^N \left(\frac{\mathbf{a}_i^2 + \mathbf{a}_i}{2} - b' \mathbf{a}_i \right) \mathbf{x}_i \right) = \frac{-(b')^2 + b'}{2^u} \bmod 2^{k-u}.$$

If $u < k$, we assign each of the monomials a new variable $\mathbf{y}_i$, and transform it into a linear equation: $\sum_i \bar{\mathbf{c}}_i \mathbf{y}_i = \bar{b} \bmod 2^{k-1-u}$.

Let $\mathbf{c}_i = \bar{\mathbf{c}}_i \bmod 2$ and $b = \bar{b} \bmod 2$. So we get a linear equation over $\mathbb{F}_2$: $\sum_i \mathbf{c}_i \mathbf{y}_i = b \bmod 2$.

As in Section 3.2, when collecting enough equations in our experiments, we always get a matrix L by performing Gaussian elimination, such that

$$L\mathbf{y} = \left(\begin{array}{c|c} 1 \cdots & \\ \ddots & * \\ \hline \mathbf{0} & I_{N \times N} \end{array} \right) \left(\begin{array}{c} \mathbf{y}_1 \\ \vdots \\ \hline \mathbf{x} \end{array} \right) = \mathbf{b}.$$

whereas L is not invertible. Nevertheless, it is enough for us to recover $\mathbf{m}$.

What's more, we can also use the strategy below to get an invertible L with very high probability. Notice that $u = 1$ holds for very high probability. We only use those samples in which $u = 1$. So we have

$$\sum_{i<j} \mathbf{a}_i \mathbf{a}_j \mathbf{x}_i \mathbf{x}_j + \sum_{i=1}^N \left(\frac{\mathbf{a}_i^2 + \mathbf{a}_i}{2} - b' \mathbf{a}_i \right) \mathbf{x}_i = \frac{-(b')^2 + b'}{2} \bmod 2. \tag{5}$$

Denote by $\mathbf{y}_{(i,j)}$ the variable corresponding to $\mathbf{x}_i\mathbf{x}_j$ $(i < j)$ and by $\mathbf{c}_{(i,j)}$ its coefficient, namely $\mathbf{a}_i\mathbf{a}_j$. We claim that

Proposition 4. *For* $1 \le i \le N-1$, $\boldsymbol{c}_{(i,N)} = \sum_{k=1}^{i-1} \boldsymbol{c}_{(k,i)} + \sum_{k=i+1}^{N-1} \boldsymbol{c}_{(i,k)}$ *mod* 2.

Proof. By Equation (4), we know that $\sum_{k=1}^{N} \mathbf{a}_k = 1 \bmod 2$. So

$$\begin{aligned}
\sum_{k=1}^{i-1} \mathbf{c}_{(k,i)} + \sum_{k=i+1}^{N-1} \mathbf{c}_{(i,k)} &= \sum_{k=1}^{i-1} \mathbf{a}_k\mathbf{a}_i + \sum_{k=i+1}^{N-1} \mathbf{a}_i\mathbf{a}_k && \bmod 2 \\
&= \sum_{k=1}^{i-1} \mathbf{a}_k\mathbf{a}_i + \mathbf{a}_i(1 - \sum_{k=1}^{i-1} \mathbf{a}_k - \mathbf{a}_i - \mathbf{a}_N) && \bmod 2 \\
&= \sum_{k=1}^{i-1} \mathbf{a}_k\mathbf{a}_i - \sum_{k=1}^{i-1} \mathbf{a}_k\mathbf{a}_i + \mathbf{a}_i - \mathbf{a}_i^2 - \mathbf{a}_i\mathbf{a}_N && \bmod 2 \\
&= \mathbf{a}_i\mathbf{a}_N && \bmod 2 \\
&= \mathbf{c}_{(i,N)} && \bmod 2
\end{aligned}$$

We involve some new variables: $\bar{\mathbf{y}}_{(i,j)} = \mathbf{y}_{(i,j)} + \mathbf{y}_{(i,N)} + \mathbf{y}_{(j,N)}$, where $1 \le i \le N-2$, $i < j \le N-1$. Since for $1 \le i \le N-1$, $\mathbf{c}_{(i,N)}\mathbf{y}_{(i,N)} = \sum_{j=1}^{i-1} \mathbf{c}_{(j,i)}\mathbf{y}_{(i,N)} + \sum_{j=i+1}^{N-1} \mathbf{c}_{(i,j)}\mathbf{y}_{(i,N)} \bmod 2$, we have

$$\sum_{i<j<N} \mathbf{c}_{(i,j)}\bar{\mathbf{y}}_{(i,j)} = \sum_{i<j\le N} \mathbf{a}_i\mathbf{a}_j\mathbf{x}_i\mathbf{x}_j \bmod 2 = \sum_{i<j\le N} \mathbf{c}_{(i,j)}\mathbf{y}_{(i,j)} \bmod 2.$$

Using the new variables $\bar{\mathbf{y}}_{(i,j)}$ $(i < j < N)$ in Equation (5), we can eliminate the old variables $\mathbf{y}_{(i,j)}$ $(i < j \le N)$. Now the number of the variables is $d = \binom{N}{2} + 1 = O(N^2)$.

If we have enough such linear equations such that we can find a $d \times d$ nonsingular matrix L satisfying $L\mathbf{y} = \mathbf{b}$, we can find $\mathbf{y}$ by solving the set of linear equations over $\mathbb{F}_2$. In our experiments, we can always find an invertible L. So we can find $\mathbf{m}$.

3.4 The Broadcast Attack against NTRU-2005

For NTRU-2005, $ES = \{0, 1\}$ and $\mathbf{m} \in \{0, 1\}^N$. Since q is a prime, so we can use the algorithm for LWE directly. To decrease the number of the variables, we can also use the fact $\mathbf{x}_i^2 = \mathbf{x}_i$ holds for $\mathbf{x}_i \in \{0, 1\}$. So we use the following equations

$$2\sum_{i<j} \mathbf{a}_i\mathbf{a}_j\mathbf{x}_i\mathbf{x}_j + \sum_{i=1}^{N}((\mathbf{a}_i^2 + \mathbf{a}_i) - 2b'\mathbf{a}_i)\mathbf{x}_i = -(b')^2 + b' \bmod q.$$

If we have enough linear equations such that we can find a $d \times d$ nonsingular matrix L satisfying $L\mathbf{y} = \mathbf{b}$, where $d = \binom{N}{2} + N = O(N^2)$, we can find $\mathbf{y}$ by solving the set of linear equations over $\mathbb{F}_q$.

3.5 Analysis of the Attacks

Some Observations. Notice that we can obtain N linear equations from every recipient. If we assume that d linearly independent linear equations can be found from $O(d/N)$ recipients with very high probability, it is easy to show that we need gather $O(N)$ (resp. $O(N^2)$) recipients's information and solve a set of $O(N^2)$ (resp. $O(N^3)$) linear equations to complete the attack against NTRU-2001 with an odd d_g and NTRU-2005 (resp. NTRU-1998). With ordinary Gaussian elimination and multiplication, we have the following result.

Variant	N	d	Recipients	Space	Time
NTRU-1998	N	$O(N^3/6)$	$O(N^2/6)$	$O(N^6/36)$	$O(N^9/768)$
NTRU-2001	N	$O(N^2/2)$	$O(N/2)$	$O(N^4/4)$	$O(N^6/24)$
NTRU-2005	N	$O(N^2/2)$	$O(N/2)$	$O(N^4 \log N/4)$	$O(N^6 \log^2 N/24)$

It remains to discuss the assumption. We have done many experiments and find that d linearly independent linear equations can be found from $O(d/N)$ recipients with probability very close to one. So the broadcast attacks succeed with overwhelming probability. In the appendix, we give the concrete estimates for the attack complexities for the cases used in IEEE 1363.1 Standard [18] based on NTRU 1998, regardless of its padding. Next we give our experimental results.

Experimental Results. All experiments were performed on a Windows XP system with a 3.20 GHz Pentium 4 processor and 2 GByte RAM using Shoup's NTL library version 5.4.1 [24].

We implemented the three instantiations of NTRU and the successful attacks against them, where we find a L invertible. For NTRU-1998 and NTRU-2001, we adopted the algorithms for $u = 1$. In our experiments, we always obtained an invertible L with the recipients whose number is just a little more than or equal to d/N. Some results are listed below:

Variant	N	q	p	d_f	d_g	d_r	d	Recipients	Rank(L)	Result
NTRU-1998	47	32	3	7	8	5	17390	373	17390	Success
NTRU-2001	167	128	$2+x$	60	19	18	13862	88	13862	Success
NTRU-2005	107	97	2	25	24	25	5778	56	5778	Success

All the experiment finish within minutes.

3.6 Improving the Attack

We have some methods to improve the attack.

With the Known Bits. If we know some bits, either the message bits or the random bits, we can also improve the attack.

- If we know some bits of $\mathbf{m}$, for example, $\mathbf{m}_0, \mathbf{m}_2, \cdots, \mathbf{m}_{t-1}$, then we can eliminate those monomials containing at least one of these known bits.
- If we know some bits of $\mathbf{r}$, for example, $\mathbf{r}_0, \mathbf{r}_2, \cdots, \mathbf{r}_{t-1}$, then for $i = 0, \cdots, t-1$, we have t linear equations: $\sum_{j=0}^{N-1} \widehat{H}_{i+1,j+1}\mathbf{m}_j + \mathbf{r}_i - \mathbf{b}_i = 0 \bmod q$. Since

these equations are linear independent, we can represent some t $\mathbf{m}_i$'s by the other $N-t$ variables, hence also eliminate those monomials containing at least one of these t $\mathbf{m}_i$'s.

However, how can we obtain these bits? We next show that "guessing" is a good idea for NTRU as pointed in [19].

For any vector $\mathbf{v} = (\mathbf{v}_0, \mathbf{v}_1, \cdots, \mathbf{v}_{N-1})^T$, we denote by $\mathbf{v}^{(r)}$ its r-cycle:

$$\mathbf{v}^{(r)} = \begin{cases} \mathbf{v}, & r = 0; \\ (\mathbf{v}_{N-r}, \mathbf{v}_{N-r+1}, \cdots, \mathbf{v}_{N-1}, \mathbf{v}_0, \mathbf{v}_1, \cdots, \mathbf{v}_{N-r-1})^T, & r \in \{1, \cdots, N-1\}. \end{cases}$$

For NTRU, since $H\mathbf{r} + \mathbf{m} = \mathbf{c} \bmod q$, we have $H\mathbf{r}^{(i)} + \mathbf{m}^{(i)} = \mathbf{c}^{(i)} \bmod q$ for $i = 0, \cdots, N-1$. We take $\mathbf{r}$ as an example to show how we guess some of its bits. For example, we guess $\mathbf{r}_0 = 0, \mathbf{r}_2 = 0, \cdots, \mathbf{r}_{t-1} = 0$. Then, if there is an i-cycle of $\mathbf{r}$ such that, for j=0,...,t-1, $\mathbf{r}_j^{(i)} = \mathbf{r}_j$, we can use the corresponding equation $H\mathbf{r}^{(i)} + \mathbf{m}^{(i)} = \mathbf{c}^{(i)} \bmod q$ instead of the origin one for any recipient to continue the attack. Of course, we don't know what i is. However, we can commit the attack for every $i \in \{0, 1, \cdots, N-1\}$, then check whether we get the correct message.

By [19], we know that the probability that there is an i-cycle satisfying our need is approximately equal to $1-(1-\prod_{j=0}^{d_r-1}(1-\frac{t}{N-j}))^N$ for $\mathcal{L}(r) = \mathcal{B}(d_r)$, which is very close to 1 for small t. For more analysis see [19].

With the Clue from the Parameters. We can easily get: $h(1)r(1) + m(1) = c(1) \bmod q$, when we take $h(x), r(x), m(x)$ and $c(x)$ as polynomials. Since we know $h(x), c(x)$, and by $\mathcal{L}_r$ we also know that $r(1) = 0$ in NTRU-1998 and $r(1) = d_r$ in NTRU-2001 and NTRU-2005, we have $\sum_{j=0}^{N-1} \mathbf{m}_j + h(1)r(1) - c(1) = 0 \bmod q$. Then we can eliminate a variable. For example, since $\mathbf{m}_0 = h(1)r(1) - c(1) - \sum_{j=1}^{N-1} \mathbf{m}_j \bmod q$, we can substitute $\mathbf{m}_0$ in every equation.

Use the XL or the Mutant XL Algorithms. Our attack can also be further improved with XL and MutantXL algorithms[2,4], or similar Gröbner basis type of algorithms. For instance, in case of NTRU 2001, if we do not have enough recipients to obtain enough ciphertext to generate enough quadratic equations in our attack, we could use the idea of the XL type of algorithm by multiplying the derived equations by monomial of degree one or higher, such that we may solve our system of equations at a higher degree. In this case, our attack could work with much fewer recipients but higher computation cost. The details of this part of work will be left to a subsequent paper.

4 Conclusion

In this paper, we present an algebraic broadcast attack against NTRU. Under a reasonable assumption, the attack can be completed in polynomial time and space. The experiments show that the attack succeed with probability very close

to one. This is the first efficient successful broadcast attack against NTRU. However, it is still an open problem to find a more efficient broadcast attack with just a constant number of recipients.

Acknowledgments. The work of J.D. was supported by **Charles Phelps Taft Foundation** and the NNSF of China under the grant #60973131. The work of Y.D. and Y.P. was supported by the NNSF of China (Grants Nos. 11071285, 61121062) and 973 Project (2011CB302401).

References

1. Arora, S., Ge, R.: New Algorithm for Learning in Presence of Errors, `http://www.cs.princeton.edu/~rongge/LPSN.pdf`
2. Buchmann, J., Cabarcas, D., Ding, J., Mohamed, M.S.E.: Flexible Partial Enlargement to Accelerate Gröbner Basis Computation over $\mathbb{F}_2$. In: Bernstein, D.J., Lange, T. (eds.) AFRICACRYPT 2010. LNCS, vol. 6055, pp. 69–81. Springer, Heidelberg (2010)
3. Coppersmith, D., Shamir, A.: Lattice Attacks on NTRU. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 52–61. Springer, Heidelberg (1997)
4. Courtois, N.T., Klimov, A.B., Patarin, J., Shamir, A.: Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 392–407. Springer, Heidelberg (2000)
5. Ding, J.: Solving LWE problem with bounded errors in polynomial time. Cryptology ePrint Archive, Report 2010/558 (2010)
6. Ding, J.: Fast Algorithm to solve a family of SIS problem with l_∞ norm. Cryptology ePrint Archive, Report 2010/581 (2010)
7. Ding, J.: Algebraic solvers for certain lattice-related problems. In: 2011 IEEE Information Theory Workshop (ITW), pp. 405–409. IEEE Conference Publications (2011)
8. Gama, N., Nguyên, P.Q.: New Chosen-Ciphertext Attacks on NTRU. In: Okamoto, T., Wang, X. (eds.) PKC 2007. LNCS, vol. 4450, pp. 89–106. Springer, Heidelberg (2007)
9. Hästad, J.: Solving simultaneous modular equations of low degree. SIAM J. Comput. 17, 336–341 (1988)
10. Hoffstein, J., Silverman, J.H.: Implementation Notes for NTRU PKCS Multiple Transmissions, Report #6, NTRU Technical Reports, `http://www.securityinnovation.com/cryptolab/pdf/NTRUTech006.pdf`
11. Hoffstein, J., Silverman, J.H.: Optimizations for NTRU. Technical report, NTRU Cryptosystems (June 2000), `http://citeseer.ist.psu.edu/693057.html`
12. Hoffstein, J., Pipher, J., Silverman, J.H.: NTRU: A Ring-Based Public Key Cryptosystem. In: Buhler, J.P. (ed.) ANTS 1998. LNCS, vol. 1423, pp. 267–288. Springer, Heidelberg (1998)
13. Howgrave-Graham, N.: A Hybrid Lattice-Reduction and Meet-in-the-Middle Attack Against NTRU. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 150–169. Springer, Heidelberg (2007)
14. Howgrave-Graham, N., Nguyên, P.Q., Pointcheval, D., Proos, J., Silverman, J.H., Singer, A., Whyte, W.: The Impact of Decryption Failures on the Security of NTRU Encryption. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 226–246. Springer, Heidelberg (2003)

15. Howgrave-Graham, N., Silverman, J.H., Whyte, W.: A Meet-In-The-Meddle Attack on an NTRU Private Key. Technical Report, http://www.ntru.com/cryptolab/technotes.htm#004
16. Howgrave-Graham, N., Silverman, J.H., Whyte, W.: Choosing Parameter Sets for NTRUEncrypt with NAEP and SVES-3. Technical Report, NTRU Cryptosystems (2005)
17. Hirschhorn, P.S., Hoffstein, J., Howgrave-Graham, N., Whyte, W.: Choosing NTRUEncrypt Parameters in Light of Combined Lattice Reduction and MITM Approaches. In: Abdalla, M., Pointcheval, D., Fouque, P.-A., Vergnaud, D. (eds.) ACNS 2009. LNCS, vol. 5536, pp. 437–455. Springer, Heidelberg (2009)
18. IEEE. P1363.1 Public-Key Cryptographic Techniques Based on Hard Problems over Lattices. IEEE (June 2003), http://grouper.ieee.org/groups/1363/lattPK/index.html
19. May, A., Silverman, J.H.: Dimension Reduction Methods for Convolution Modular Lattices. In: Silverman, J.H. (ed.) CaLC 2001. LNCS, vol. 2146, pp. 110–125. Springer, Heidelberg (2001)
20. Mol, P., Yung, M.: Recovering NTRU Secret Key from Inversion Oracles. In: Cramer, R. (ed.) PKC 2008. LNCS, vol. 4939, pp. 18–36. Springer, Heidelberg (2008)
21. Nguyên, P.Q., Pointcheval, D.: Analysis and Improvements of NTRU Encryption Paddings. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 210–225. Springer, Heidelberg (2002)
22. Plantard, T., Susilo, W.: Broadcast Attacks against Lattice-Based Cryptosystems. In: Abdalla, M., Pointcheval, D., Fouque, P.-A., Vergnaud, D. (eds.) ACNS 2009. LNCS, vol. 5536, pp. 456–472. Springer, Heidelberg (2009)
23. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Johnson, D.S., Feige, U. (eds.) Proc. of 37th STOC, pp. 84–93. ACM (2005)
24. Shoup, V.: NTL: A library for doing number theory, http://www.shoup.net/ntl/

A Some Results for IEEE 1363.1 Standard

Since IEEE 1363.1 Standard is based on NTRU-1998, we give our estimation for the cost of our broadcast attack on the schemes without the padding.

Parameter Set	N	d	Recipients(at least)	Space(bits $\approx$)	Time ($\approx$)
ees401ep1	401	10747600	26802	2^{47}	2^{69}
ees541ep1	541	26391100	48782	2^{50}	2^{73}
ees659ep1	659	47699700	72382	2^{52}	2^{74}
ees449ep1	449	15087300	33602	2^{48}	2^{70}
ees613ep1	613	38392200	62630	2^{51}	2^{73}
ees761ep1	761	73453200	96522	2^{53}	2^{77}
ees677ep1	677	51716000	76390	2^{52}	2^{76}
ees887ep1	887	116312000	131130	2^{54}	2^{79}
ees1087ep1	1087	214063000	196930	2^{56}	2^{82}
ees1087ep2	1087	214063000	196930	2^{56}	2^{82}
ees1171ep1	1171	267623000	228542	2^{56}	2^{83}
ees1499ep1	1499	561378000	374502	2^{59}	2^{85}