NTRU Cryptosystems Technical Report

Report # 004, Version 1
Title: A Meet-In-The-Middle Attack on an NTRU Private Key
Author: Joseph H. Silverman
Release Date: Tuesday, July 15, 1997

Abstract. In this report we describe a meet-in-the-middle attack on an NTRU private key. Hence if the private key is chosen from a sample space with 2^M elements, then the security level of the cryptosystem is $2^{M/2}$.

Acknowledgement. We would like to thank Andrew Odlyzko for showing us the following meet-in-the-middle attack.

We begin with some notation:

N, d	Integer parameters used to create an NTRU cryptosystem. To make the explanation clearer, we will assume N and d are even; the modifications for odd values are easy.
f	The private key, chosen consisting of d ones and $N-d$ zeros.
g	Used to form the public key, chosen with coefficients between 0 and $r-1$.
h	The public key $h \equiv f^{-1}g \pmod q$.
k, ℓ	Integers chosen by the attacker so that $(q/2^\ell)^k$ is larger than $\binom{N/2}{d/2}$ (say by factor of 100).

The idea is to search for f in the form $f_1 + f_2$, where f_1 and f_2 each have $d/2$ ones. The attacker performs the following steps:

Step 1. Choose at random $N/2$ of the N possible positions. We will assume that in the actual private key f, exactly $d/2$ of these $N/2$ positions have ones. The odds of this happening are approximately $\sqrt{d}$-to-1, so the algorithm will need to be executed approximately $\sqrt{d}$ times before succeeding.

To make our description clearer, we will relabel the positions so that we have chosen the first $N/2$ of them. Then we are looking for two vectors f_1 and f_2, each of length $N/2$ with $d/2$ ones and the rest zeros, so that their sum $f_1 + f_2$ has the property that $(f_1 + f_2)h \pmod q$ is small; more precisely, so that its coordinates all lie between 0 and $r-1$.

Step 2. Enumerate the vectors f_1. This takes $\binom{N/2}{d/2}$ steps and is done as follows. We put the f_1's into bins based on the first k coordinates of $f_1 h \pmod q$. To form the bins, we do the following. Break the interval $[0, q-1]$ up into subintervals of length 2^ℓ. We will write $\mathcal{I}$ for this set of subintervals. In other words, $\mathcal{I}$ is the set of intervals

$$I_j = [2^\ell(j-1), 2^\ell j - 1] \qquad \text{with } 1 \le j \le q/2^\ell.$$

(For convenince, we will assume that $2^\ell | q$.) A bin is then a k-tuple of intervals chosen from $\mathcal{I}$. If the first k coordinates of some $f_1 h \pmod q$ are $(a_1, \ldots, a_k)$, say, then we put f_1 into the bin $(I_1, \ldots, I_k)$ such that $a_i \in I_i$ for $1 \le i \le k$. We also note for future reference that the set of bins $\mathcal{I}^k$ for f_1's has order

$$\#\mathcal{I}^k = (q/2^\ell)^k.$$

Step 3. Enumerate the vectors f_2 and put them into bins, which also takes $\binom{N/2}{d/2}$ steps. We put the f_2's into bins based on the first k coordinates of $-f_2 h \pmod q$. However, we are going to make the f_2 bins a little bit larger than the f_1 bins. More precisely, we let $\mathcal{J}$ be the set of intervals

$$J_j = [2^\ell(j-1) - (r-1), 2^\ell j - 1] \qquad \text{with } 1 \le j \le q/2^\ell.$$

Notice that these intervals overlap, so some f_2's will go into more than one bin. In any case, the set of bins $\mathcal{J}^k$ for f_2's has order

$$\#\mathcal{J}^k = (q/2^\ell)^k.$$

Step 4. We now perform an inspection of the bins to see where they overlap. Note that if $(f_1 + f_2)h \pmod q$ has all of its coordinates between 0 and $r-1$, and if f_1 is in the bin

$$(I_{j_1}, I_{j_2}, \ldots, I_{j_k}),$$

then f_2 must be in the bin

$$(J_{j_1}, J_{j_2}, \ldots, J_{j_k}).$$

So assuming that the choice of positions in Step 1 was correct, we'll be able to match the correct f_1 with the corresponding f_2.

Analysis of the Algorithm
As indicated, the correct f_1 and its matching f_2 will be in corresponding bins, so we will find them. We also want to make sure that there aren't too many "false alarms." The key number here is the occupancy rate, which is the fraction of bins which have an f_1 (or f_2) in them. If this is reasonably small, say less than 1%, then 99% of the f_1's will be in bins whose corresponding f_2 bin is empty, so such f_1's may be immediately discarded. The remaining f_1's will have a non-empty corresponding f_2 bin, but almost always that f_2 bin will contain only one f_2, so all we need to do is check if that (f_1, f_2) pair has the property that $(f_1 + f_2)h \pmod q$ is small. Note that the occupancy rate is

$$\text{Occupancy Rate} \approx \frac{\binom{N/2}{d/2}}{\#\mathcal{J}^k} = \frac{\binom{N/2}{d/2}}{(q/2^\ell)^k}.$$

(We are ignoring the effect caused by f_2's which are put into more than one bin. If q is much larger than r, which we will assume, then this effect will be negligible.)

Next we observe that the number of steps in the algorithm will be on the order of

$$\sqrt{d}\left\{2\cdot\binom{N/2}{d/2}+\left(\frac{q}{2^\ell}\right)^k\right\},$$

where the $\sqrt{d}$ comes from choosing $N/2$ positions in Step 1, the two $\binom{N/2}{d/2}$'s come from the putting of the f_1's and f_2's into bins, and the $(q/2^\ell)^k$ comes from checking each pair of corresponding bins in $\mathcal{I}^k$ and $\mathcal{J}^k$ to see if they are simultaneously occupied. Since we will choose the occupancy rate to be fairly small, the time for the algorithm is on the order of

$$\text{Time}\approx\sqrt{d}\left(\frac{q}{2^\ell}\right)^k.$$

Finally we note the the storage space necessary for the bins is on the order of

$$\#\mathcal{I}^k+\#\mathcal{J}^k=2\left(\frac{q}{2^\ell}\right)^{2k}.$$

Sample Parameters
The following table gives some sample parameters which indicate how the meet-in-the-middle attack reduces the search time from $\binom{N}{d}$ to a small multiple of $\binom{N/2}{d/2}$.

N	d	q	k	ℓ	$\binom{N}{d}$	Occupancy Rate $\frac{\binom{N/2}{d/2}}{(q/2^\ell)^k}$	Time $\sqrt{d}\left(\frac{q}{2^\ell}\right)^k$	Storage $2\left(\frac{q}{2^\ell}\right)^k$
168	80	65536	15	10	$2^{163.70}$	$10^{-2.91}$	$2^{93.16}$	$2^{90.10}$
108	10	64	14	4	$2^{45.14}$	$10^{-1.93}$	$2^{29.66}$	$2^{28.10}$
264	18	64	25	4	$2^{91.44}$	$10^{-1.65}$	$2^{52.08}$	$2^{50.10}$

Comments and questions concerning this technical report should be addressed to

techsupport@ntru.com

Additional information concerning NTRU Cryptosystems and the NTRU Public Key Cryptosystem are available at

www.tiac.net/users/ntru
