Cryptanalysis of the Revised NSS Signature Scheme

Craig Gentry and Mike Szydlo

September 17, 2001

Abstract. In this note we address the modifications proposed to the NSS signature scheme to protect it against the attacks of Gentry, Jonsson, Stern, and Szydlo presented at the EUROCRYPT rump session. The NSS signature scheme itself (with suggested revisions) was presented at the same conference. These revisions include new key-gen, signing and verification procedures (full details appear in the EEES standard document). We contend that these new attacks render the revised scheme insecure. We exhibit a multi-stage attack with heuristic and polynomial time components to reveal the private key.

NSS Summary: The public key of NSS consists of a polynomial h of degree $N-1$, whose coefficients are integers modulo q. This public key is related to the three private key polynomials via the relation $(3f_1+u)h = (3g_1+u)$, where f_1, g_1, and u are small polynomials. The signing of the message m employs two small masking polynomials w_1 and w_2 to form the signature $s = (m+w_1)u^{-1}(mod 3) + w_2 f \pmod{128}$. There are many verification steps, the most significant of which seem to be that both $(s-m)/p \pmod{128}$ and $(sh-m)/p \pmod{128}$ are small polynomials.

Stage 1: The first stage of our attack finds unreduced multiples of the private key f in the ring $Z[x]/(1-x^n)$. The key idea is to first use the congruence $f*w_i = s_i \bmod 128$ with the estimate $f*w_i = m_i \bmod 3$ to obtain an estimate s_i' of $f*w_i$. Observing that $f*w_i*g*w_j - f*w_j*g*w_i = 0$, we may revise our estimates by systematically forcing our estimates $s_i'*t_j' - s_j'*t_i'$ to zero. We are able to correct signatures completely 90% of the time using only 4 signatures.

Stage 2: We show how to use unreduced signatures and a small dimension lattice (dimension $N/2$) to very quickly recover the product $f*f_{rev}$, which will be used in following attacks. The idea is to estimate $f*f_{rev}$ with an averaging attack and use this estimate to set up a closest vector problem in the lattice of multiples of $f*f_{rev}$ which are palindromes.

Stage 3: This attack uses both orthogonality ideas and congruence ideas. First, if a lattice has a target basis of v_i that is circulant and nearly orthogonal, then a bound on $|v*a|$ implies a bound on $|a|$. Next we exploit the structure of the cyclotomic integers - in $Z[x]/(1-x^p)$, $f^(p-1) = 1 \pmod{\text{p}}$ for $p = 1 \pmod{\text{N}}$. We find some α so that αf^{p-1} is nearly orthogonal. LLL will find a short multiple $\alpha f^{p-1} a$, and we use the congruence to find a exactly. It is not difficult to find f from a f^{p-1}. There are technical steps which avoid computing with the full polynomials f^{p-1}, whose coefficients are very big.

Remark We also survey the attacks available when one is working with the well studied cyclotomic integers. In particular, the standard GCD algorithm is a lattice problem in only dimension N, half of the dimension in the previous attacks. For some lower dimensional parameters (still $N < 150$) this already yields the private key.

Remark: We show how to use the ideal (f) combined with the ratio g/f to reduce the key recovery problem to a low density knapsack problem. Although theoretically interesting, such knapsack problems do still require lattices to solve.

Remark: Alternatively, we show how to use $f*f_{rev}$ to convert the a lattice problem to one for which the lattice has an *orthonormal basis*. This lattice is presented via its Gram matrix, rather than in coordinates. This Gram matrix has determinant one and contains only small, integral entries. We expect LLL to perform better on such orthogonal lattices.