Lattice-based cryptography: Episode V: the ring strikes back

Daniel J. Bernstein
University of Illinois at Chicago

Crypto 1999 Nguyen: "At Crypto '97, Goldreich, Goldwasser and Halevi proposed a public-key cryptosystem based on the closest vector problem in a lattice, which is known to be NP-hard. We show that . . . the problem of decrypting ciphertexts can be reduced to a special closest vector problem which is much easier than the general problem. As an application, we solved four out of the five numerical challenges proposed on the Internet by the authors of the cryptosystem. At least two of those four challenges were conjectured to be intractable. We discuss ways to prevent the flaw, but conclude that, even modified, the scheme cannot provide sufficient security without being impractical.”

Fix would "probably need dimension $\geq$ 400" for security: "Public key $\approx$ 1.8 Mbytes".

Crypto 1998 Nguyen–Stern: "Provably secure" Ajtai–Dwork system breakable with 20MB keys.

Fix would "probably need dimension $\geq$ 400" for security: "Public key $\approx$ 1.8 Mbytes".

Crypto 1998 Nguyen–Stern: "Provably secure" Ajtai–Dwork system breakable with 20MB keys.

Compare to 1978 McEliece code-based cryptosystem: much more stable security story through dozens of attack papers. Typical parameters: 1MB key for $>2^{128}$ *post-quantum* security.

2017.05: Lattice student adds the following text to Wikipedia page "Lattice-based cryptography": "Lattice-based constructions are currently the primary candidates for post-quantum cryptography."

2017.05: Lattice student adds the following text to Wikipedia page "Lattice-based cryptography": "Lattice-based constructions are currently the primary candidates for post-quantum cryptography."

— [citation needed]

2017.05: Lattice student adds the following text to Wikipedia page “Lattice-based cryptography”: “Lattice-based constructions are currently the primary candidates for post-quantum cryptography.”

— [citation needed]

2016.07: Google rolls out large-scale experiment with post-quantum crypto between Chrome and some Google sites. **Uses lattice-based crypto.**

Google sent only a few KB
for public keys, ciphertexts.

How can lattice-based crypto
work within a few KB?
Combine two ingredients:

1. Do *not* take key sizes
large enough for theorems to
connect to "well-studied" SVP_γ.
See, e.g., 2016 Chatterjee–
Koblitz–Menezes–Sarkar.

Google sent only a few KB
for public keys, ciphertexts.

How can lattice-based crypto
work within a few KB?
Combine two ingredients:

1. Do *not* take key sizes
large enough for theorems to
connect to "well-studied" SVP_γ.
See, e.g., 2016 Chatterjee–Koblitz–Menezes–Sarkar.

2. **Use ideal lattices.**
Hope that the extra structure
doesn't damage security.

1996–1998 Hoffstein–Pipher–Silverman "NTRU":

Define R as the ring $\mathbf{Z}[x]/(x^{503} - 1)$.

Elements of R are polynomials $c_0 + c_1 x + c_2 x^2 + \cdots + c_{502} x^{502}$ with integer coefficients c_j.

To multiply in R:
multiply polynomials;
replace x^{503} with 1;
replace x^{504} with x; etc.
e.g.: $(x^{100} + x^{300})(x^{200} + 7x^{400})$
$= x^{300} + 8x^{500} + 7x^{700}$
$= 7x^{197} + x^{300} + 8x^{500}$ in R.

Define $q = 2048$.

Alice's public key: $A \in R$ with coefficients in $\{0, 1, \ldots, q - 1\}$.
This is $503 \cdot 11 = 5533$ bits.

Define $q = 2048$.

Alice's public key: $A \in R$ with coefficients in $\{0, 1, \ldots, q-1\}$. This is $503 \cdot 11 = 5533$ bits.

Bob generates random $b, c \in R$ *with small coefficients*: e.g., all coefficients in $\{-1, 0, 1\}$.

Define $q = 2048$.

Alice's public key: $A \in R$ with coefficients in $\{0, 1, \ldots, q - 1\}$. This is $503 \cdot 11 = 5533$ bits.

Bob generates random $b, c \in R$ *with small coefficients*: e.g., all coefficients in $\{-1, 0, 1\}$.

Bob computes $Ab + c \bmod q$: multiply A by b in R; add c; reduce each coefficient modulo q to the range $\{0, 1, \ldots, q - 1\}$.

Define $q = 2048$.

Alice's public key: $A \in R$ with coefficients in $\{0, 1, \ldots, q - 1\}$. This is $503 \cdot 11 = 5533$ bits.

Bob generates random $b, c \in R$ *with small coefficients*: e.g., all coefficients in $\{-1, 0, 1\}$.

Bob computes $Ab + c \bmod q$: multiply A by b in R; add c; reduce each coefficient modulo q to the range $\{0, 1, \ldots, q - 1\}$.

Bob sends $Ab + c \bmod q$. This is also 5533 bits.

“Quotient NTRU” (new name),
used in original NTRU design:

Alice generated $A = 3a/d$ in R/q
for small random a, d
(with suitable invertibility):
i.e., $dA - 3a \bmod q = 0$.

"Quotient NTRU" (new name), used in original NTRU design:

Alice generated $A = 3a/d$ in R/q for small random a, d (with suitable invertibility): i.e., $dA - 3a \bmod q = 0$.

Alice receives $C = Ab + c \bmod q$. Alice computes $dC \bmod q$, i.e., $3ab + dc \bmod q$.

"Quotient NTRU" (new name),
used in original NTRU design:

Alice generated $A = 3a/d$ in R/q
for small random a, d
(with suitable invertibility):
i.e., $dA - 3a \bmod q = 0$.

Alice receives $C = Ab + c \bmod q$.
Alice computes $dC \bmod q$,
i.e., $3ab + dc \bmod q$.

Alice reconstructs $3ab + dc$,
using smallness of a, b, d, c.
Alice computes dc,
deduces c, deduces b.

“Product NTRU” (new name),
2010 Lyubashevsky–Peikert–Regev:

Everyone knows random $G \in R$.
Alice generated $A = aG + d \bmod q$
for small random a, d.

"Product NTRU" (new name),
2010 Lyubashevsky–Peikert–Regev:

Everyone knows random $G \in R$.
Alice generated $A = aG + d \bmod q$ for small random a, d.

Bob sends $B = Gb + e \bmod q$ and $C = m + Ab + c \bmod q$ where b, c, e are small and each coefficient of m is 0 or $q/2$.

“Product NTRU” (new name),
2010 Lyubashevsky–Peikert–Regev:

Everyone knows random $G \in R$.
Alice generated $A = aG + d \bmod q$
for small random a, d.

Bob sends $B = Gb + e \bmod q$
and $C = m + Ab + c \bmod q$
where b, c, e are small and each
coefficient of m is 0 or $q/2$.

Alice computes $C - aB \bmod q$,
i.e., $m + db + c - ae \bmod q$.
Alice reconstructs m,
using smallness of d, b, c, a, e.

Lattice view: Define L as the set of pairs $(v, w) \in R \times R$ such that $vG - w \bmod q = 0$.

Lattice view: Define L as the set of pairs $(v, w) \in R \times R$ such that $vG - w \bmod q = 0$.

e.g. $(a, A - d) \in L$.
$(0, A)$ is close to a lattice point.

Try to find close lattice point.
Breaks both Product NTRU and Quotient NTRU.

Lattice view: Define L as the set of pairs $(v, w) \in R \times R$ such that $vG - w \bmod q = 0$.

e.g. $(a, A - d) \in L$.
$(0, A)$ is close to a lattice point.

Try to find close lattice point.
Breaks both Product NTRU and Quotient NTRU.

Try to exploit reuse of b for faster Product NTRU attack.
("Ring-LWE": arbitrary reuse.)

Try to exploit $A = 3a/d$ structure for faster Quotient NTRU attack.

2013 Lyubashevsky–Peikert–Regev: "All of the algebraic and algorithmic tools (including quantum computation) that we employ . . . can also be brought to bear against SVP and other problems on ideal lattices. Yet despite considerable effort, no significant progress in attacking these problems has been made. The best-known algorithms for ideal lattices perform essentially no better than their generic counterparts, both in theory and in practice."

Many more NTRU variants
(often not crediting NTRU).

Fully homomorphic encryption:
STOC 2009 Gentry
"Fully homomorphic encryption
using ideal lattices".
PKC 2010 Smart–Vercauteren.
Eurocrypt 2011 Gentry–Halevi.
etc.

Multilinear maps: e.g.,
Eurocrypt 2013 Garg–Gentry–
Halevi "Candidate multilinear
maps from ideal lattices".

STOC 2009 Gentry system is **broken** by quantum algorithms for typical "cyclotomic rings".

STOC 2009 Gentry system is **broken** by quantum algorithms for typical "cyclotomic rings".

First stage in attack:
SODA 2016 Biasse–Song
fast quantum algorithm to compute $gR \mapsto ug$ with $u \in R^*$.

Builds upon STOC 2014 Eisenträger–Hallgren–Kitaev–Song quantum $R \mapsto R^*$ algorithm.

STOC 2009 Gentry system is **broken** by quantum algorithms for typical "cyclotomic rings".

First stage in attack: SODA 2016 Biasse–Song fast quantum algorithm to compute $gR \mapsto ug$ with $u \in R^*$.

Builds upon STOC 2014 Eisenträger–Hallgren–Kitaev–Song quantum $R \mapsto R^*$ algorithm.

Older pre-quantum algorithms take subexponential time.

Second stage of attack: 2014.10 Campbell–Groves–Shepherd fast pre-quantum algorithm for typical cyclotomic ring to compute $ug \mapsto$ short g.

Second stage of attack: 2014.10 Campbell–Groves–Shepherd fast pre-quantum algorithm for typical cyclotomic ring to compute $ug \mapsto$ short g.

Eurocrypt 2017 Cramer–Ducas–Wesolowski extension of CGS: for typical cyclotomic ring, find fairly short element of *any* ideal.

Second stage of attack: 2014.10 Campbell–Groves–Shepherd fast pre-quantum algorithm for typical cyclotomic ring to compute $ug \mapsto$ short g.

Eurocrypt 2017 Cramer–Ducas–Wesolowski extension of CGS: for typical cyclotomic ring, find fairly short element of *any* ideal.

These attacks exploit structure of cyclotomic rings. Rescue system by switching to another ring?

2014.02 Bernstein: pre-quantum attack strategy; subexponential time for many choices of ring.

Eurocrypt 2017 Bauch–Bernstein–de Valence–Lange–van Vredendaal: quasipolynomial-time pre-quantum attack for "multiquadratic rings".

2016 Bernstein–Chuengsatiansup–Lange–van Vredendaal "NTRU Prime": use prime degree, large Galois group, inert modulus; reduce attack surface at low cost.