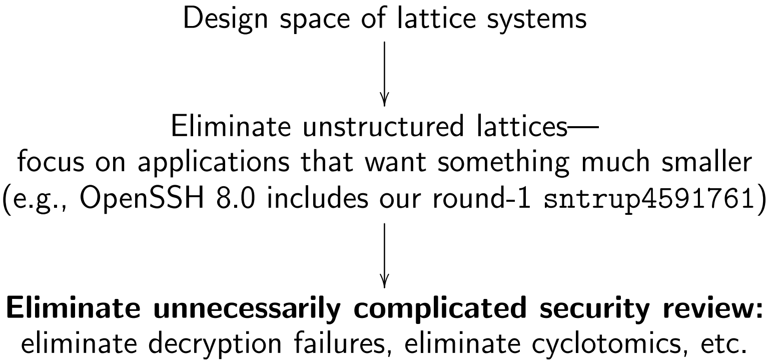
NTRU Prime: round 2

Daniel J. Bernstein, Chitchanok Chuengsatiansup,
Tanja Lange, and Christine van Vredendaal

https://ntruprime.cr.yp.to

24 August 2019

Same one-way functions; same core advantage

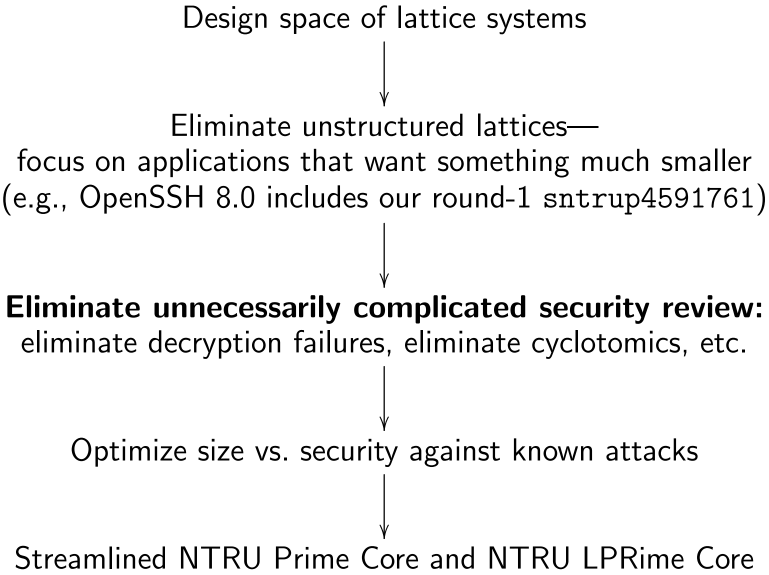


Design space of lattice systems

Eliminate unstructured lattices—
focus on applications that want something much smaller
(e.g., OpenSSH 8.0 includes our round-1 `sntrup4591761`)

Eliminate unnecessarily complicated security review:
eliminate decryption failures, eliminate cyclotomics, etc.

Same one-way functions; same core advantage



Design space of lattice systems

Eliminate unstructured lattices—
focus on applications that want something much smaller
(e.g., OpenSSH 8.0 includes our round-1 `sntrup4591761`)

Eliminate unnecessarily complicated security review:
eliminate decryption failures, eliminate cyclotomics, etc.

Optimize size vs. security against known attacks

Streamlined NTRU Prime Core and NTRU LPRime Core

Extra parameter sets; improved CCA conversion

Added smaller dim and larger dim to parameter sets:

- `sntrup653` and `ntrulpr653`. (New smaller dim.)
- `sntrup761` and `ntrulpr761`. (Same dim as round 1.)
- `sntrup857` and `ntrulpr857`. (New larger dim.)

Extra parameter sets; improved CCA conversion

Added smaller dim and larger dim to parameter sets:

- `sntrup653` and `ntrulpr653`. (New smaller dim.)
- `sntrup761` and `ntrulpr761`. (Same dim as round 1.)
- `sntrup857` and `ntrulpr857`. (New larger dim.)

Tweaks to CCA conversion:

- Implicit rejection as second layer of CCA defense beyond plaintext confirmation.
- More hashing, to enforce unique encodings of ciphertexts and public keys.
- New unified encoding mechanism. Shorter key/ciphertext strings than round-1 encoding.

Expanded documentation

§2: Reorganized and expanded algorithm specification, with modules matching the conceptual layers of the design.

§3: Modularized and generalized parameter specification.

Expanded documentation

§2: Reorganized and expanded algorithm specification, with modules matching the conceptual layers of the design.

§3: Modularized and generalized parameter specification.

§4: Extended design rationale: CCA changes in round 2; analysis of arguments for other one-way functions.

§9: Extended analysis of advantages and limitations.

Expanded documentation

§2: Reorganized and expanded algorithm specification, with modules matching the conceptual layers of the design.

§3: Modularized and generalized parameter specification.

§4: Extended design rationale: CCA changes in round 2; analysis of arguments for other one-way functions.

§9: Extended analysis of advantages and limitations.

§6: Expanded and updated analysis of known attacks. Many different security estimates computed by our new script: some to compare to "Estimate" page, some for improvements.

§7, §8: Expanded and updated analysis of expected strength.

§5: Expanded and updated performance analysis.

New software: more modular, faster

New test script `ntruprime.sage`. Same structure as spec. Covers all parameter sets. Also has `round1` option.

New `ref` C implementation. Same structure as test script.

New `factored` implementation. Portable C wrapper around modules with separate tests and optimizations.

New software: more modular, faster

New test script `ntruprime.sage`. Same structure as spec. Covers all parameter sets. Also has `round1` option.

New `ref` C implementation. Same structure as test script.

New `factored` implementation. Portable C wrapper around modules with separate tests and optimizations.

What round-1 submission said about `sntrup4591761` speed: 59456 enc, 97684 dec, >6000000 keygen. (`titan0` cycles.)

New software: more modular, faster

New test script `ntruprime.sage`. Same structure as spec. Covers all parameter sets. Also has `round1` option.

New `ref` C implementation. Same structure as test script.

New `factored` implementation. Portable C wrapper around modules with separate tests and optimizations.

What round-1 submission said about `sntrup4591761` speed: 59456 enc, 97684 dec, >6000000 keygen. (`titan0` cycles.)

Our current round-2 `sntrup761` speed: **55252 enc, 70464 dec, 946772 keygen.** (`titan0` cycles.)

Slowdown from extra hashing, but speedups in constant-time inversion (CHES 2019 Bernstein–Yang), sorting, mults, etc.

`ntrulpr761`: **77280 enc, 95316 dec, 47396 keygen.**