Overview of some results by Eindhoven University of Technology

Sven Schäge

Eindhoven University of Technology

13 July 2023

Approach for each submission

- Study the submission documents.
- Identify the mathematically hard problem they base their security on.
- Analyze the submission in three ways:
  - Estimate the security of the underlying mathematical problem.
  - Attempt to cryptanalyze the way the hardness of the system is linked to the hardness of the underlying mathematical assumption.
  - Check the security reduction. This can serve to focus the cryptanalysis work.
- For "surviving" submissions
  - Evaluate efficiency in terms of sizes (public and private keys and ciphertexts or signatures, respectively), speed, and bandwidth.
  - Check the security reduction (same as above, but now for tightness and as sanity check).

This is an ongoing process, this talk gives an overview of where we got so far.

IPCC

- IPCC is based on graphs and key recovery requires finding a perfect dominating set in a 3-regular graph. This looks like a hard problem.
- We (Daniel J. Bernstein, Jolijn Cottaar, and Tanja Lange) found attack to extract message from ciphertext.
- Attack announced 23 Dec 2022 at KpqC forum.
- Encryption partitions message into summands m_i, each assigned to a graph position. Encryption uses the graph properties to obfuscate these shares.
- Main issue is that the resulting ciphertexts are very sparse and that the summands leak by frequency analysis in 90% of all cases
- The authors have acknowledged the attack.

NTRU+

- Lattice-based scheme using $R = \mathbf{Z}[x]/(x^n - x^{n/2} + 1)$ and R/q and $R/3$ with $\gcd(q, 3) = 1$.
- Choosing $f = 3f' + 1$ gives faster decapsulation but larger q.
- Uses centered binomial distribution, hence variable-weight errors.
- We checked for
  - evaluation-at-1 attack,
  - generic attacks (sieving, combinatorial attacks, golden collision attack),
  - reaction attacks

  Our estimates do not always match those stated in the submission but we found no attack.
- We noticed that the FO transform was split into two stated but missed the CCA attack that Joohee Lee found.
- We can confirm the attack and are waiting for details of the fix.
- Internally, Hövelmanns had expressed concerns about the FO transform, we will check new proof and implementations, if any.

Other lattice-based KEMS

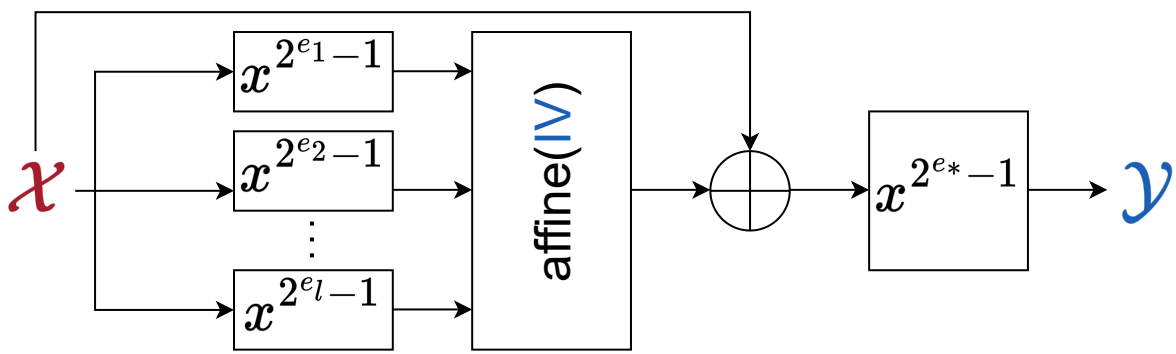
- SMAUG and TiGER are MLWE/MLWR RLWR/RLWE-based systems with a narrow error distribution.
- Both TiGER and SMAUG fit into the template of existing, provably secure lattice-based schemes (as initiated by Regev) and
  - offer new trade-offs and
  - benefit from more recent developments in the field
  - While aiming at high efficiency overall, the optimization strategies of the two algorithms are somewhat distinct: while TiGER concentrates on having both pk and ct sizes below a practical threshold (motivated by IKEv2 payload size), SMAUG specifically optimizes for very small secrets.
- Bernstein pointed out 28 December on KpqC forum that the combinatorial attack by May (Crypto 21) applies that exploits sparse secrets.
- We're analyzing old and new parameters.
- Since the schemes are lattice-based as well, we also consider the same set of attacks as for NTRU+.

Code-based KEMS

- For REDOG – see the separate talk.
- Layered ROLLO-I got analyzed by Chee, Jeong, Lee, Ryu in April.
- PALOMA is close to the NIST submission Classic McEliece in using binary Goppa codes, but has some core differences:
  - The Goppa polynomial g is chosen to split completely over $\mathbf{F}_{2^m}$ (instead of being irreducible).
  - Support and the t roots of g need to share $\mathbf{F}_{2^m}$, hence m is larger.
  - The authors generalized Patterson decoding to this case. The result looks correct but is slower than Berlekam-Massey.
  - This choice of g limits the keyspace, but it is still very large.
  - PALOMA's secret key is a lot larger and operations are slower.
  - We are still hunting for algebraic attacks using that g factors.
  - Encap/decap are slower and larger than for Classic McEliece.
  - Proof analysis is ongoing; efficiency analysis suggests some changes.

AIMer

- AIMer designed using MPC-IN-THE-HEAD design paradigm close to the NIST submission Picnic.
- Security of the signature scheme relies merely on underlying one-way function.
- Security model restricts attacker to O(1) data.
- Algebraic attacks: The security only suggests **upper**-bounds on the complexity of attacks.
- The experimental results suggests the randomness is worse that a linear congruential generator used in C progamming language.
- Not out of ideas, but no attack yet.



FIBS

- FIBS is based on SPHINCS+ but using a isogeny-based hash function based on the Charles-Goren–Lauter (CGL) hash function instead of SHA2 or SHAKE.
- This choice is very slow, more than 10000 slower than SPHINCS+ with SHA2 or SHAKE.
- Sizes for SPHINCS+ are independent of the used hash and therefore they are the same for FIBS.
- FIBS builds on the flawed proof stated in the first-round submission of SPHINCS+, Asiacrypt 2022 has fixed proof but additional requirements on the used hash functions (among others undetectability and decisional-second-preimage-resistance).
- Further research is needed to understand if the used isogeny-based hash function CGL guarantees these properties.

Other signatures

- Enhanced pqsigRM: See talk 17 April by Alex Pellegrini.
- GCK Signature: attacked by Kim, Ryu, and Lee; authors promised some updates, so we postponed.
- MQ-Sign: broken by Aulbach, Samardjiska, Trimoska and by Kematsu, Jo, Yasuda. Parameters for UOV still look OK.
- Peregrine attacked by Espitau, Lin, Suzuki, Tibouchi, Yu, Zhang. Waiting for updated parameters.