Round5: merge of Round2 and HILA5

Hayo Baan[3], Sauvik Bhattacharya[3], Jung Hee Cheon[6], Scott Fluhrer[2], Oscar Garcia-Morchon[3], Paul Gorissen[3], Thijs Laarhoven[7], Rachel Player[5], Ronald Rietman[3], Markku-Juhani O. Saarinen[4], Yongha Son[6], Ludo Tolhuizen[3], José Luis Torre-Arce[3], and Zhenfei Zhang[1]

[1]Algorand (US)
[2]Cisco (US)
[3]Philips (Netherlands)
[4]PQShield (UK)
[5]RHUL (UK)
[6]SNU (Republic of Korea)
[7]TU/e (Netherlands)

April 10, 2020

Round5

Round5 is a merger of the Round2 [2] and HILA5 [4] submissions. Basically, Round5 is Round2 combined with the error-correction code defined in HILA5.

This document summarizes which elements of Round2 and HILA5 are combined in Round5. It also describes the official comments on Round2, HILA5, and Round5 on the NIST PQC forum, and how they have been addressed. It further summarizes changes done over time.

1 2nd Round Submission

1.1 Changes in Round5 compared to Round2 and HILA5

- **Error correction:** Round5 incorporates an error correction code, called XEf and based on that of HILA5, into the INDCPA-PKE scheme defined in Round2. The goal is to achieve the same target failure probability as Round2, but using smaller configuration parameters that lead to better performance. Direct application of HILA5' error correction in Round2 does not work well, as the decoder is confronted with correlated errors. These correlations are caused by the usage of a prime cyclotomic polynomial as reduction polynomial. Securely applying HILA5's code to Round2 requires performing operations on $\mathbf{v}$ in the NTRU ring and using balanced secrets. These are not major changes since Round2 – internally – already performs all operations on $\mathbf{v}$ in the NTRU ring, and Round2's implementation also uses balanced secrets.
  The XEf error correction code avoids table look-ups and conditions altogether and it is, therefore, resistant to timing attacks.

- **Security targets:** Security targets in Round5 for NIST security levels I, III, and V are such that breaking Round5 requires a classical effort of at least 128, 192, and 256 bits, respectively. Similarly, the quantum-effort to break Round5 is bigger than $128 - \text{MAXDEPTH}$, $192 - \text{MAXDEPTH}$, and $256 - \text{MAXDEPTH}$, respectively. Round5 encapsulates 128-, 192-, and 256- bit long keys in NIST security levels I, III, and V.

- **Parameter sets:** Round5 defines 21 parameter sets.

  - Six ring-based parameter sets (three each for INDCPA-KEM and INDCCA-PKE, each for NIST security levels I, III, and V) with a code XEf capable of five error correction. Using XEf requires the reduction polynomial to be $\xi(x) = x^{n+1} - 1$ and that the sparse ternary secrets are balanced. These parameter sets are based on the merge of HILA5 with Round2 and show that the usage of error correction leads to the smallest public key and ciphertext sizes.
  - Six ring-based parameter sets (as above, three each for KEM and PKE, corresponding to NIST security levels I, III, and V) without error correction. These parameter choices can be considered more conservative than the previous ones, as they do not employ error correction and therefore are only based on the Round2 design that has received public review since its submission. However, since no error correction is applied, bandwidth requirements are around 33% higher than the previous parameters based on the merge of Round2 and HILA5 using error correction.
  - Six non-ring-based parameter sets (as above, three each for KEM and PKE, corresponding to NIST security levels I, III, and V) without

error correction. These parameter choices rely on same design choices as the original Round2 submission.

- Three application-tailored parameter sets.
  - A ring-based KEM parameter set addressing Internet of Things applications that achieves even smaller bandwidth (736 Bytes in total) at the price of lower security and higher failure probability.
  - A ring-based KEM NIST level 1 parameter set in which the encapsulated key is 192-bit long instead of just 128-bit long so that the difficulty of attacking the encapsulated key (by Grover) equals the difficulty of quantum lattice attack to Round5.
  - A non-ring-based PKE NIST Level III parameter set with a ciphertext size of only 988 Bytes, with very fast encryption and decryption, by taking $\overline{m} = 1$, at the cost of a larger public key. This configuration makes unstructured lattice configurations feasible in applications in which the public-key can remain static for a long time, e.g., email encryption.

- **Rounding constants:** In contrast to Round2, Round5 defines the rounding operation in terms of flooring and rounding constants. Round5 does so to make the INDCPA security proof work.

- **Power-of-two moduli:** All moduli in Round5 are powers of two. This allows for easy-to-implement modular arithmetic, and avoids the generation of random uniform noise otherwise required to guarantee uniform symbols in public keys and ciphertexts. Thus, Round5 does not provide support for NTT speed-ups that were applicable with both Round2 and HILA5.

- **Improved description:** Round5 specification is based on Round2 documentation to make it easier to identify changes with regard to the original submission. The specification is improved by including a broader security analysis and a more detailed technical specification. Moreover, it reports simulation results that support the independence assumption in the analysis of the failure probability in case that error correction is used.

1.2 PQC comments during the first round

- **Constant time sorting in Round2:** On December 27, 2017, Daniel J. Bernstein addressed the constant-time generation of ternary secrets. Round5 addresses this by not requiring sorting in the generation of ternary secrets and using simple rejection sampling. Rejection sampling is not constant-time, but it is not related to the secret itself.

- **INDCCA security in HILA5:** On December 28, 2017, Lorenz Panny pointed out an error in HILA5's description that claimed IND-CCA security. Round5 addresses this by using the Fujisaki - Okamoto transformation proposed in Round2.

- **INDCPA-PKE proof in Round2:** On January 12, 2018, Jan-Pieter D'Anvers pointed out that the IND-CPA security proof of Round2 should be corrected using rounding constants. Round5 addresses this issue in the way suggested by D'Anvers.

- **Security levels:** On January 13, 2018, Michael Hamburg pointed out that the Round2 security levels did not match NIST definition. Round5 uses correct NIST security levels.

- **Correlated errors in prime cyclotomic polynomial:** On August 4th, 2018, Léo Ducas pointed out potential issues in the independence assumption in the failure probability analysis of the initial Round5 description.

  In a subsequent comment on August 24, 2018, Michael Hamburg discussed the correlation of failures in the prime cyclotomic ring. He concluded that it does not affect the original Round2 design, but it frustrates the direct application of XEf on Round2. In the same comment, Hamburg also describes a ring switching trick, developed by himself and three members of the Round2 team, which addresses this issue and is used to securely apply Xef error correction to Round5. The Round5 specification also reports on simulation results that support the independence assumption in the analysis of the failure probability in case that error correction is used.

2 Round5 update during 2nd Round

2.1 Team extension

The Round5 team has been extended with new members who have contributed to Round5: Jung Hee Cheon and Yongha Son from SNU (Republic of Korea) and Paul Gorissen from Philips (Netherlands).

2.2 Round5 improvements and updates during the second round

The following three sections detail improvements and updates in the specification, implementation and scripts. Although the list is long, most of the improvements refer to better analysis or changes to enable better implementations. Parameters and core algorithms remain unchanged.

2.2.1 Updates in the specification

The following changes have been made in the specification.All section numbers refer to [1].

- Round5's internal IND-CCA KEM is exposed explicitly, since some applications might benefit of an IND-CCA KEM. This does not require any changes in parameters or change in the specification. However, since after this change Round5 has two available KEMs, 2nd Round parameter sets are renamed from R5N*_*{KEM/PKE}_*d into R5N*_*{CPA/CCA}_*d. An IND-CPA parameter set can be used in combination with Round5's IND-CPA KEM. Round5's IND-CCA KEM and PKE require IND-CCA parameter sets.

- The specification now includes the enumeration method used to obtain concrete enumeration-based security estimates.

- Section 2.7.6 on the hybrid attack now contains the state-of-the-art analysis due to Wunderer [5]. This has led to a slight increase/decrease in the concrete classical/quantum security estimates.

- Section 2.7.7, formerly called "Attacks against Sparse Secrets", contains a more detailed study of the guessing + dual attack. This has led to a slight reduction in the concrete security estimates.

- The analysis of the failure probability of Round5 in Section 2.8 now takes into account that secrets are balanced, i.e., contain equally many ones as minus ones. This results in slightly smaller failure probabilities.

- The concrete security number in the tables for the different parameter sets are those obtained from the tighter security and failure probability analysis in Section 2.7.6, 2.7.7 and 2.8.

- According to [3, Sec.4], lattice-based schemes in which the public parameter $\mathbf{A}$ is not sent over explicitly, but is obtained from a applying a function to a transmitted seed σ, can only claim INDCPA security in their core INDCPA secure public-key encryption scheme in the random oracle model. Thus, the INDCPA-proof for Round5 in Appendix A has been updated accordingly; specifically, the previous section Appendix A.1 has been removed. In the new section 2.7.10, we argue why it is unlikely that there exist attack avenues against (default) instantiation of the function for generating $\mathbf{A}$ from σ and for generating secrets.

- The new Section 2.9.4 describes the multiple options for side-channel countermeasures that are applicable to different types of platforms. These include countermeasures against implementation errors or attacks relying on malformed public parameters $\mathbf{B}$ and $\mathbf{U}$, and various countermeasures against timing attacks. Appendix B shows how a parameter pertaining to these implementations is computed.

- Section 2.9.8 includes updated performance numbers for the different side-channel countermeasures, AVX2-optimized code for ring and non-ring parameters, and performance numbers when a Hardware-Software co-design is applied.

- Section 2.11.4 contains an updated definition of the deterministic random bit generation and hash functions using TupleHash instead of cSHAKE. This allows a unified way to place different Round5 routines in different domains, i.e., it ensures domain separation. These routines are the hash functions H in r5_cpa_kem and H and G in r5_cca_kem, the generation of pseudorandom data to sample secret keys $\mathbf{R}$ and $\mathbf{S}$, and the generation of pseudo random data to sample public matrix $\mathbf{A}$ and to obtain the permutations required in $f_{n,d}^{(\tau=1,2)}$.

- The algorithm to generate the secrets is slightly modified so that if a Round5 secret has more that one vector, then the pseudorandom data required for each of the secret vectors is independent of each other. This allows for interoperability independently of the side-channel countermeasures used in an implementation. The algorithm is described in 2.11.6, functions create_S_T and create_R_T.

- The create_A function in Section 2.11.6 has an updated definition of $f_{n,d}^{(\tau=0)}$ method to compute the public matrix $\mathbf{A}$ that allows for optimized implementations based on vector instructions. With this definition, Round5 non-ring variants using $f_{n,d}^{(\tau=0)}$ become between 3x and 5x faster. Round5 ring-variants become 2x faster.

2.2.2 Updates in implementations

The implementations are extended as follows:

- The $r5\_cca\_kem$ algorithm is accessible and it can be used with IND-CCA parameter sets.
- Renaming of the parameter sets to enable the previous change: R5N*_*KEM_*d is renamed into R5N*_*CPA_*d and R5N*_*PKE_*d into R5N*_*CCA_*d.
- Included a library (standalone) implementing SHA-3, SHA-3 Extendable-Output Functions, cSHAKE, and TupleHashXOF. Thus, $r5\_cpa\_kem$ and $r5\_cca\_kem$ algorithms can be used without external libraries. This library includes AVX2 optimized code.
- Renamed of the DRBG functions used to generate pseudorandom data that were used by multiple routines into different functions having different purposes such as the generation of pseudorandom data for A (AGeneration) or for the secret keys (SKGeneration). All these functions encapsulate TupleHash(XOF).
- Modification in the generation of Round5 secret keys containing multiple secret key vectors so that the pseudorandom data required to sample each of those vectors is independent of each other. This allows for interoperability in implementations using different types of side-channel countermeasures in the secret-key generation routine.
- Updated the definition of the matrix **A** generation routine to allow for optimized AVX2 implementations.
- In the optimized implementation, we have included the CM_CT option for a fully constant-time implementation in the generation of a ternary secret and ring multiplication routines.
- In the optimized implementation, we have included an AVX2 optimized ring multiplication routine.
- In the optimized implementation, we have included tests to check for malformed **B** and **U** parameters.
- General small improvements in the code.

2.2.3 Updates in scripts

- Extension of the scripts used for parameter search to perform a wider-search. Some better performing parameter sets have been found.
- Improvements in the routines used to compute the failure probability to take into account the usage of balanced secrets.
- Included new scripts to obtain tighter security estimates for the Guessing+Dual attack and the hybrid attack. Note that since these functions are rather slow, the parameter search script still uses the old hybrid code to do a first faster parameter search. Once candidates are found, the security is estimated by means of the new scripts.

- Included a script to obtain HMAX, the fixed number of calls to generate h different indexes with probability $1 - 2^{\kappa}$. This is required in the constant time implementation of the secret key generation.
- Included a script to obtain the thresholds to detect malformed parameters $\mathbf{B}$ and $\mathbf{U}$.

2.3 PQC Comments during the second round

On September 16, 2019, Mike Hamburg pointed out that the sampling of secrets is not constant-time, as well as its potential risk, especially in the re-encryption in the FO-transform. Round5 has addressed this comment by providing a constant-time implementation option which uses rejection sampling with a constant number HMAX of experiments, see Section 2.9.4. HMAX is chosen such that rejection sampling is successful with high probability (Appendix B). As only the h first successful samples are kept, the secret generated with the constant-time implementation equals that of a straightforward approach which stops sampling after h successes.

References

[1] Hayo Baan, Sauvik Bhattacharya, Jung Hee Cheon, Scott Fluhrer, Oscar Garcia-Morchon, Paul Gorissen, Thijs Laarhoven, Rachel Player, Ronald Rietman, Markku-Juhani O. Saarinen, Yongha Son, Ludo Tolhuizen, Jose Luis Torre-Arce, and Zhenfei Zhang. Round5: KEM and PKE based on (Ring) Learning with Rounding, April 2020.

[2] Hayo Baan, Sauvik Bhattacharya, Oscar Garcia-Morchon, Ronald Rietman, Ludo Tolhuizen, Jose-Luis Torre-Arce, and Zhenfei Zhang. Round2: KEM and PKE based on GLWR. Technical report, National Institute of Standards and Technology, November 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[3] Daniel J. Bernstein. Comparing prooof of security for lattice-based encryption. Cryptology ePrint Archive, Report 2019/691, 2019.

[4] Markku-Juhani O. Saarinen. HILA5: Key Encapsulation Mechanism (KEM) and Public Key Encryption Algorithm. Technical report, National Institute of Standards and Technology, November 2017. Available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

[5] Thomas Wunderer. *On the Security of Lattice-Based Cryptography Against Lattice Reduction and Hybrid Attacks.* PhD thesis, Technische Universität, Darmstadt, 2018.