From: Keita Xagawa <xagawa.keita@lab.ntt.co.jp>
Sent: Tuesday, January 09, 2018 11:42 PM
To: pqc-comments
Cc: pqc-forum@list.nist.gov
Subject: OFFICIAL COMMENT: LEDAkem

Dear authors, deal all,

The current specification (and implementation) of LEDAkem seems to fail to achieve CCA security.
LEDAkem tries to construct an IND-CCA-secure KEM by applying the conversion in [30] to a OW-CPA-secure deterministic PKE.
The authors would not notice the chosen-ciphertext attacks in [A1] and [A3, Appendix K] against KEM/Hybrid PKE in [30].

LEDAkem
==========
* The public key is M in F_2^{p x n}.
* The encapsulation algorithm chooses e <- F_2^n with HW(e) = t, and outputs a ciphertext s = M e^T and a session key K = KDF(e).
* The decapsulation algorithm recovers e from s by using the secret key and outputs k_s = KDF(e). If s is invalid, the decapsulation algorithm returns a "pseudorandom" key k_s = KDF(s).

The footnote 1 of [30] suggests k_s = KDF(s), which is not pseudorandom.

Chosen-Ciphertext Attack against the current LEDAkem ========== The following CCA exists even if the scheme is perfectly correct. See [A1] and [A3, Apendix K].
For i = 0,...,n-1, let u_i be the i-th unit vector of dimension n.

* Assume that a ciphertext s = M e^T is given and assume that e[0] = 0.
* For i = 1, ..., n-1, we query s_i = s + M {u_0+u_i}^T and obtain the result.
* Set e[i] = 0 if k_s == KDF(s_i); else set e[i] = 1.
* Compute K = KDF(e)

If e[i] = 1, then HW(e + u_0 + u_i) = t. On the other hand, if e[i] = 0, then HW(e + u_0 + u_i) = t + 2 > t.
This breaks the onewayness of KEM.

Note
==========
If DFR is 0, it is easy to fix the problem.

* Persichetti's thesis suggests to use KDF(s'), where s' = L_{n_0-1}^{-1} s in the LEDAkem context.
* [A1] and [A2] suggests to use KDF(\pi(s)), where \pi is a random permutation. Notice that this \pi should be pseudorandom. Otherwise, one can still check if a ciphertext is valid or invalid by checking the answer is random or deterministic.
* [HHK17] and [SXY17] suggests to use Hash(secret-seed,s) (or KDF(secret-seed,s)).

[30]: Edoardo Persichetti:
"Secure and Anonymous Hybrid Encrytpion from Coding Theory" in PQCrypto 2013
[A1]: Pierre-Louis Cayrel, Cheikh Thiecoumba Gueye, El Hadji Modou Mboup, Ousmane Ndiaye, and Edoardo Persichetti:
"Efficient Implementation of Hybrid Encryption from Coding Theory" in C2SI 2017
[A2]: Edoardo Persichetti:
"Code-based Key Encapsulation from McEliece's Cryptosystem" in MACIS2017
[A3]: Daniel J. Bernstein, Chitchanok Chuengsatiansup, Tanja Lange, Christine van Vredendaal:
"NTRU prime: reducing attack surface at low cost" in SAC 2017.
(http://eprint.iacr.org/2016/461)

Regards,
Keita Xagawa

From: Gerardo Pelosi <gerardo.pelosi@polimi.it>
Sent: Wednesday, January 10, 2018 12:31 PM
To: pqc-forum@list.nist.gov
Subject: Re: [pqc-forum] OFFICIAL COMMENT: LEDAkem

Dear Keita,

thank you for the observations provided and the clarity in pointing them out.

In the LEDAkem supporting documentation Sect. 2.4, we claimed that LEDAkem provides the NIST required IND-CPA security and that it is possible to adapt it to achieve IND-CCA security employing a KDF to hide the case where a decoding failure (in your observation, induced by the
attacker) takes place.

We acknowledge that the KDF to provide such a guarantee requires the addition of a secret bitstring of some kind as a parameter to the KDF, as pointed out by Persichetti's thesis, [A1], [A2] (can you provide references for [HHK17] and [SXY17] to us privately?), and this is not pointed out in the specification and reference implementation.

While the presence of a secret bitstring has no impact on the IND-CPA security of the scheme in a case where ephemeral keys are used, it provides a useful resiliency feature in case of accidental key reuse, in addition to allowing to achieve IND-CCA.

We are thus planning to provide an up-to date specification document including this clarification, and adapt the implementation accordingly on our website as soon as possible.

Kind Regards,
--the LEDAkem team

Il 01/10/18 05:42, Keita Xagawa ha scritto:
> Dear authors, deal all,
>
> The current specification (and implementation) of LEDAkem seems to
> fail to achieve CCA security.
> LEDAkem tries to construct an IND-CCA-secure KEM by applying the
> conversion in [30] to a OW-CPA-secure deterministic PKE.
> The authors would not notice the chosen-ciphertext attacks in [A1] and
> [A3, Appendix K] against KEM/Hybrid PKE in [30].
>
> LEDAkem
> =========
> * The public key is M in F_2^{p x n}.
> * The encapsulation algorithm chooses e <- F_2^n with HW(e) = t, and
> outputs a ciphertext s = M e^T and a session key K = KDF(e).
> * The decapsulation algorithm recovers e from s by using the secret
> key and outputs k_s = KDF(e). If s is invalid, the decapsulation
> algorithm returns a "pseudorandom" key k_s = KDF(s).
>
> The footnote 1 of [30] suggests k_s = KDF(s), which is not pseudorandom.
>
>
> Chosen-Ciphertext Attack against the current LEDAkem ========= The

From: Gerardo Pelosi <gerardo.pelosi@polimi.it>
Sent: Monday, October 01, 2018 5:17 AM
To: pqc-comments
Cc: pqc-forum@list.nist.gov
Subject: OFFICIAL COMMENT: LEDAkem and LEDApkc
Attachments: official_comment.pdf

Dear all,

we prepared a document as an official comment on LEDAkem and LEDApkc (attached to this message and available at the url reported below) in which:

** We provide our quantification of the quantum and classic computational effort levels we considered as the computational requirements to break AES. We relied on classical circuit design estimates for the classical computing complexity, and on the work by Grassl et al. at PQCrypto 2016 for the quantum computing complexity.

** We delineate an automatic procedure to find an optimal set of parameters for LEDAkem/LEDApkc matching the aforementioned security margin. We pair this procedure with the release of the sources of a parameter computation tool for LEDAkem/LEDApkc. The sources are hereby placed in the public domain, and we welcome contributions and suggestions.

** Our approach to the estimation of the computational effort required to perform Information Set Decoding attacks was to evaluate their complexity in the finite regime (as opposed to employing asymptotic bounds), and to perform an exhaustive search in the parameter space of the algorithms.
We report that, concerning the parameter sizes of the LEDA cryptosystems, the advantages offered by more recent ISD algorithms over the proposal by Stern in 1988 are lower than a factor of 2^4.

** We report new running time and key size figures for the optimal parameter sets, showing a x3.5--x6.8 speedup on the reference implementation and a ~x2 key size reduction w.r.t. the submission parameters.

** We propose a novel technique allowing us to design a set of QC-LDPC code parameters for use in LEDAkem/LEDApkc deriving an upper bound to the code DFR in closed-form. This allows to include a bound on the DFR as a parameter design criterion.

** We report sample sets of parameters targeting an upper bound for the DFR of 2^-64 for long term keys in LEDApkc.

The full document is both attached to this message, and hosted at https://www.ledacrypt.org/archives/official_comment.pdf

The public domain software implementation of the automated procedure for the design of tight and optimal sets of parameters for the LEDA cryptosystems is available at https://github.com/LEDAcrypt/LEDAtools
At the same address, we also provide a software tool to compute the complexity of the considered ISD attacks, given a set of code parameters.

The header files containing the revised parameter sets which tightly match the security requirements are both hosted at https://www.ledacrypt.org/archives/new_parameter_headerfiles.zip, and available on our github repositories, tagged as version 1.1.0 of the codebase.

The revised codebase also includes the appropriate modifications to cope with an artificially higher number of errors being inserted in the message (i.e. a check for the number of errors has been added) and the modifications suggested in the previous official comments.

Best regards,
--LEDA team

From: Marco Baldi <m.baldi@univpm.it>
Sent: Thursday, November 15, 2018 4:50 PM
To: pqc-forum@list.nist.gov
Subject: [pqc-forum] LEDAkem and LEDApkc merger announcement

Dear NIST and pqc-forum subscribers,

with this message we want to announce the merger of the LEDAkem and LEDApkc proposals.

* Concerning the similarity of the submissions, we note that the LEDAkem and LEDApkc primitives are based on equivalent mathematical trapdoors, i.e., the syndrome and codeword decoding problems for quasi-cyclic codes.
LEDAkem's key encapsulation exploits a Niederreiter formulation that allows fast operations when randomly generated messages (like keys) are conveyed.
LEDApkc's public key encryption scheme is directly derived from LEDAkem, but exploiting a McEliece formulation for the purpose of natively allowing more information to be encrypted in one go and employing the IND-CCA2 (assuming no decryption failures) gamma-construction proposed by Kobara and Imai.

* Concerning the parameter sets proposed, we note that LEDAkem and LEDApkc can be employed with the same parameter sets (as per the original submission) if a lifetime for the private key of 1/(DFR) decryptions satisfies the practical LEDApkc application scenario.
If this is not the case, our analysis (reported in our official comment 2018-10-01) on the upper bounds of the DFR provided by the QC-LDPC codes employed in the LEDA cryptosystems allows to scale the parameters to the desired DFR.
While the original submission of LEDAkem and LEDApkc pointed to a common set of parameters, if admitted to the second round we will provide differentiated parameter sets allowing to achieve lower DFRs for LEDApkc, where the DFR figure has some security impacts (for instance, when considering reaction attacks).
No changes, besides the ones due to the tighter parametrization proposed in our official comment are expected to the LEDAkem parameters.

* We acknowledge the fact that we are merging a KEM and a PKE scheme: this is explicitly allowed by the NIST merger guidelines. The schemes will retain, in the merged proposal, their corresponding security guarantees, as there will be no difference in their algorithmic description.

* We will provide, by the requested deadline, new signed IP statements for the merged submission as per NIST's request. We will carry on in our current line of proposing a patent-free cryptosystem and public domain licensed code base to avoid any issue to whomever may be willing to research on, implement or adopt the LEDA cryptosystems.

* We confirm that we will provide a merged submission package and the corresponding documentation if admitted to the second round, as per NIST's request. We foresee a simpler description of the merged submission in a single specification document and a simplification of the code base, which currently shares a non negligible amount of components, reducing the code footprint.

The LEDAkem and LEDApkc team