From:	Kevin Carrier <kevin.carrier@ensea.fr>
Sent:	Monday, July 31, 2023 9:50 AM
To:	pqc-comments
Cc:	pqc-forum; Jean-Pierre Tillich
Subject:	Round 1 (Additional Signatures) OFFICIAL COMMENT: SDitH
Attachments:	Peters_analysis_for_SDitH.pdf

Dear SDiTH, dear all,

We wish to report here that the security level of SDitH has been underestimated *at least* by about 1 to 9 bits for the parameters given in the specification document.

The reason why there is an improved attack:

A key recovery attack on SDitH is as hard as solving the d-split Syndrome Decoding problem (SD(d)). Unfortunately, the specification document of SDitH does not take into account that for the regime of parameters which is chosen, SD(d) has actually several solutions: between 31 and 748 depending on the parameters. The security analysis provided in Section 7.1 of the SDitH submission ignores this point.

The gain we report on the parameters is however less than this number of solutions because the complexity of SD(d) has been obtained in the specification document by a reduction of standard syndrome decoding (this corresponds to SD(1)) to SD(d)). This reduction is not tight in the case at hand, when there are several solutions. We wish to mention however that our complexity claims correspond to an actual attack adapting Christiane Peters' ISD [1] to the d-split problem where the complexity is computed with a formula making similar assumptions as the formula for the ISD cost given p.50 of the specification document. Full details are given here.

Results :

The complexity of solving SD(d) which is instrumental in estimating the security of SDitH is estimated apparently in the specification document of SDitH when d>1 by relying on (a) a lower bound on the complexity T(d) for solving SD(d) relying on Theorem 6.1 p.48 (b) an estimation of the complexity T(1) of the best algorithm for solving SD(1). Note that this lower bound is not tight when there is more than one solution to the SD(1) problem.

For each set of parameters of SDitH, we give
(i) the target bit security,
(ii) the estimated lower bound on T(d) following from Theorem 6.1 p.48 and the estimated cost for solving SD(1) (end of Section 7.1) in the specification document of SDitH,
(iii) the actual complexity of Peters' ISD adaptation to the SD(d) problem taking into account that there are multiple solutions.

SDitH_L1_gf256: (i) 143 bits, (ii) T(d)>=143.5 bits and T(1)=143.5 bits, (iii) 134.6 bits.
SDitH_L1_gf251: (i) 143 bits, (ii) T(d)>=143.4 bits and T(1)=143.4 bits, (iii) 133.9 bits.
SDitH_L3_gf256: (i) 207 bits, (ii) T(d)>=207.7 bits and T(1)=211.2 bits, (iii) 206.2 bits.
SDitH_L3_gf251: (i) 207 bits, (ii) T(d)>=207.6 bits and T(1)=211.1 bits, (iii) 205.0 bits.
SDitH_L5_gf256: (i) 272 bits, (ii) T(d)>=272.3 bits and T(1)=276.0 bits, (iii) 271.3 bits.
SDitH_L5_gf251: (i) 272 bits, (ii) T(d)>=272.3 bits and T(1)=276.0 bits, (iii) 269.8 bits.

Best regards,

Kevin Carrier and Jean-Pierre Tillich

[1] C. Peters. Information-set decoding for linear codes over Fq. In N. Sendrier, editor, The Third International Workshop on Post-Quantum Cryptography, PQCRYPTO 2010, pages 81–94. Springer, Heidelberg, 2010

From:	Matthieu Rivain <matthieu.rivain@cryptoexperts.com>
Sent:	Friday, August 4, 2023 11:38 AM
To:	Kevin Carrier
Cc:	pqc-comments; pqc-forum; SDitH Consortium
Subject:	Re: [pqc-forum] Round 1 (Additional Signatures) OFFICIAL COMMENT: SDitH

Dear Kevin and Jean-Pierre,

Thank you for your official comment. We acknowledge that our parameter generation script seems to be missing a factor that results in a misestimation of the complexity of traditional attacks (more on this below).

Two aspects are at play here:
1) There appears to be multiple solutions in our SD instances, which lower the complexity of the best attack;
2) SDitH relies on the d-split variant of Syndrome Decoding.

First of all, it is important to clarify that parameters for Category 1 are *not* based on the d-split variant. To be more precise, such parameters are instead based on the traditional Syndrome Decoding problem (which is the same as d-split with d=1).

Regarding 1):

To begin, we noted that your write-up is missing a component when computing the number of existing solutions; in equation (7), for instance, there should be a multiplicative factor of (q-1)^w. Without such a factor, your calculations yield in fact only 1 solution (or 2 if rounding up), and lead to the expected security target (e.g. 143 for category 1 parameters). Once this factor is included, we obtain a significant number of solutions (e.g. 460 for category 1 parameters). Could you please share the details of your calculations, confirming our intuition, and justifying the claimed "31 to 748" numbers?

If the above is confirmed, then the presence of such multiple solutions is due to an oversight in the calculation of the GV distance. In fact, in our parameter generation script, the factor present is (q-1)^(w-1), where it should have been (q-1)^w.

Upon confirmation, we will propose updated parameters to fix this issue. This should not greatly affect the sizes and performances of SDitH.

Regarding 2): We would like to stress that no security loss comes from using the d-split problem. Indeed, Theorem 6.1 gives a security reduction: breaking the d-split problem implies breaking a standard SD problem whose parameters are chosen to give the desired security level for the d-split instance (given the theorem bound). This might actually be over-conservative in terms of security.

Thanks again for your analysis and best regards,

The SDitH consortium

> On 1 Aug 2023, at 09:52, Kevin Carrier <kevin.carrier@ensea.fr> wrote:
>
> Dear SDiTH, dear all,

From: Kevin Carrier <kevin.carrier@ensea.fr>
Sent: Friday, August 4, 2023 1:59 PM
To: pqc-forum; SDitH Consortium
Cc: pqc-comments
Subject: Re: [pqc-forum] Round 1 (Additional Signatures) OFFICIAL COMMENT: SDitH
Attachments: Peters_analysis_for_SDitH.pdf

Dear SDitH consortium,

First, there is indeed a typo in Equations (7) and (8) of our pdf: the factor (q-1)^w is missing. Of course, we have taken this factor into consideration in the results that we give. A corrected version of our draft is available here or in the attached file.

Concerning Theorem 6.1, we do not questions the truth of it. We only say that the inequality given in this theorem is not tight when SD(d) has many solutions. The proof in [FJR22, Appendix A] is essentially based on the probability that an SD(1) problem becomes an SD(d) problem when the positions are randomly permuted. This probability is not binom(n/d,w/d)^d/binom(n,w) but something greater when there is many solutions.

That does not imply a security loss but only that the security is underestimated when using this theorem.

Best regards,

Kévin

> Le 4 août 2023 à 17:38, Matthieu Rivain <matthieu.rivain@cryptoexperts.com> a écrit :
>
> Dear Kevin and Jean-Pierre,
>
> Thank you for your official comment. We acknowledge that our parameter generation script seems to be missing a factor that results in a misestimation of the complexity of traditional attacks (more on this below).
>
> Two aspects are at play here:
> 1) There appears to be multiple solutions in our SD instances, which lower the complexity of the best attack;
> 2) SDitH relies on the d-split variant of Syndrome Decoding.
>
> First of all, it is important to clarify that parameters for Category 1 are *not* based on the d-split variant. To be more precise, such parameters are instead based on the traditional Syndrome Decoding problem (which is the same as d-split with d=1).
>
> Regarding 1):

From: Thibauld Feneuil <thibauld.feneuil@cryptoexperts.com>
Sent: Thursday, November 23, 2023 4:22 AM
To: pqc-comments
Cc: pqc-forum
Subject: Round 1 (Additional Signatures) OFFICIAL COMMENT: SDitH

Dear all,

The SDitH consortium is happy to announce that SDitH v1.1 has been released. This new version simply corresponds to SDitH v1.0 with updated parameters. We thank Kevin Carrier and Jean-Pierre Tillich for raising the issue about the previous parameter sets (non-unicity of the SD solution).

The new specifications and code are available on the SDitH website
https://sdith.org/
and on the GitHub repository
https://github.com/sdith/sdith

We also released the Python script we used to choose the parameter sets:
https://github.com/sdith/sdith-parameters

The new parameter sets have only a small impact on the signature performance. The signature sizes have increased by a factor of at most +3% (for all security levels). The impact of the running times of the hypercube variant is negligible. The running times of the threshold variant have increased by a factor of at most +38% when working on GF(256) and of at most +9% when working on GF(251). Let us stress that we did not optimize more the implementations compared to the NIST submission package.

Best regards,

The SDitH Consortium

From:	Kevin Carrier <kevin.carrier@ensea.fr>
Sent:	Wednesday, December 6, 2023 2:20 AM
To:	pqc-comments
Cc:	pqc-forum
Subject:	Round 1 (Additional Signatures) OFFICIAL COMMENT: SDitH

Dear SDitH consortium, dear all,

We would like to communicate on the security of SDitH
(i) as it was submitted to the NIST, which is called SDiTH v1.0, by the authors and of
(ii) SDiTH v1.1 when it was updated by the authors on November 23 and announced on the pqc forum by Thibauld Feneuil the same day (see https://sdith.org).

We already announced on the NIST forum on July 31 that the security of SDiTH was underestimated by around 1 to 9 bits depending on the parameters if we choose the same methodology as the authors to estimate the security. This was due to the fact that for the regime of parameters chosen by the authors there were multiple solutions to the decoding problem which were not taken into account. This point has been corrected in SDiTH v.1.1. Note that in SDiTH v.1.1 there is now a 4 to 5 bits margin between the security level asked by the NIST and the estimate of the authors of the best attack against their scheme. For instance in SDiTH, SDitH_L1_gf256 v1.1 for which 143 bits of security are mandatory the estimate of the best attack provided in the specifications has complexity >= 2^{147.7}. It should also be noted that the authors chose to ignore the cost of Gaussian elimination in their estimate of the best attack both for SDiTH v1.0 and SDiTH v.1.1.

In https://arxiv.org/abs/2312.02607, we have

(i) studied the security of SDiTH v1.0 and of SDiTH v1.1 with the help of a slight tweak on Stern's decoding algorithm. It consists in noticing that we can speed up decoding by considering a projective version of the decoding problem. This trick can be incorporated in standard decoding algorithms with a complexity gain of up to log q bits where q is the field size of the code which is decoded. The gain we obtain is not negligible when we work on large finite field as is the case for SDitH where q is either 251 or 256.
(ii) analyzed the effect of this tweak on Peters' improvements (and adaptation) on Stern's decoding [1,2] for codes defined over F_q.
The cost of Gaussian elimination is not negligible and we have taken this into account in our analysis.
(iii) concluded that on
a) SDiTH v.1.0 the combined effect of multiple solutions + new projective Stern/Peters decoder gives an attack which is between 9 and 14 bits below the NIST requirements.
b) SDiTH v.1.1 our attack is about 3 to 6 bits below the estimate of the best attack provided on https://sdith.org. Due

to the 4-5 bits security margin taken by the authors after preliminary results were announced on November 20 in our workshop on code based cryptography (see the slides on https://anses.hal.science/ETIS-ICI/hal-04311262v1) the estimate of our attack is sometimes above or below by about 1 bit on the security level asked by the NIST.

The following table summarizes the results we obtain. We give here

(i) the security level required by the NIST,
(ii) the security claimed by the SDitH consortium (and the corrected values when taking into account the number of solutions to the decoding problem),
(iii) the actual complexity of our "projective" Stern/Peters' decoder.

SDitH_L1_gf256 v1.0: (i) 143 bits, (ii) >143.5 bits (>135.3 bits), (iii) 129.2 bits.
SDitH_L1_gf251 v1.0: (i) 143 bits, (ii) >143.5 bits (>134.6 bits), (iii) 128.5 bits.
SDitH_L3_gf256 v1.0: (i) 207 bits, (ii) >207.7 bits (>202.4 bits), (iii) 199.3 bits.
SDitH_L3_gf251 v1.0: (i) 207 bits, (ii) >207.6 bits (>201.3 bits), (iii) 198.2 bits.
SDitH_L5_gf256 v1.0: (i) 272 bits, (ii) >272.4 bits (>267.4 bits), (iii) 264.3 bits.
SDitH_L5_gf251 v1.0: (i) 272 bits, (ii) >272.3 bits (>265.9 bits), (iii) 262.8 bits.

SDitH_L1_gf256 v1.1: (i) 143 bits, (ii) >147.7 bits, (iii) 141.5 bits.
SDitH_L1_gf251 v1.1: (i) 143 bits, (ii) >147.7 bits, (iii) 141.5 bits.
SDitH_L3_gf256 v1.1: (i) 207 bits, (ii) >211.1 bits, (iii) 207.9 bits.
SDitH_L3_gf251 v1.1: (i) 207 bits, (ii) >211.0 bits, (iii) 207.9 bits.
SDitH_L5_gf256 v1.1: (i) 272 bits, (ii) >276.3 bits, (iii) 273.3 bits.
SDitH_L5_gf251 v1.1: (i) 272 bits, (ii) >276.3 bits, (iii) 273.2 bits.

Best regards,

Kévin Carrier, Valérian Hatey and Jean-Pierre Tillich

References:

[1] Christiane Peters. "Information-set decoding for linear codes over Fq." In Post-Quantum Cryptography 2010, volume 6061 of LNCS, pages 81–94. Springer, 2010.
[2] Christiane Peters. Curves, Codes, and Cryptography. PhD thesis, Chapter 6, Tech- nische Universiteit Eindhoven, 2011.