
From: ducas <L.Ducas@cwi.nl>
Sent: Thursday, July 30, 2020 9:49 AM
To: pqc-comments
Cc: pqc-forum
Subject: ROUND 3 OFFICIAL COMMENT: NTRU

Dear all,

following valuable comments and references from John Schanck, we have updated our report:

LWE with Side Information: Attacks and Concrete Security Estimation
Dana Dachman-Soled and Léo Ducas and Huijing Gong and Mélissa Rossi
<https://eprint.iacr.org/2020/292>

This updates now also considers the symmetries in the NTRU problem in Section 6.3, and discuss the (known) ways of exploiting it in a primal attack. In particular, we found that the technique of May and Silverman is in fact slightly counter-productive, if one accounts the accumulated probabilities of finding each rotation of the secret key.

The quantitative gain from this analysis remains low (e.g. from 379 bikz to 368 bikz for ntruhs2048509, improving the attack by about 3-4 bits).

This does ***not*** contradict the claims of the NTRU Specifications document, which only claimed 359 bikz because of conservative simplifications. We hope this clarify certain details of NTRU's cryptanalysis.

Best regards,

Dana, Léo, Huijing and Mélissa

From: Simone Dutto <simone.dutto@polito.it>
Sent: Friday, July 31, 2020 10:49 AM
To: pqc-comments
Cc: pqc-forum
Subject: ROUND 3 OFFICIAL COMMENT: NTRU
Attachments: fixed-sample.patch

Dear NTRU team,

while working with the C implementations found in the ZIP file attached to your submission (2nd round), we found a minor error in the code.

Specifically, in all files sample.c, line 86 should be

```
s[4*i+3] = (u[15*i+11] & 0xfc) + (u[15*i+12] << 8) + (u[15*i+13] << 16) + (u[15*i+14] << 24);
```

instead of

```
s[4*i+3] = (u[15*i+11] & 0xfc) + (u[15*i+12] << 8) + (u[15*i+13] << 15) + (u[15*i+14] << 24);
```

This is not a mandatory correction but, without it, the implemented sampling is not the one described in the documentation.

We saw that this correction is also necessary in the current version of the implementation at

<https://github.com/jschanck/ntru/blob/master/ref-common/sample.c>

Attached to this email there is the related github patch.

Hoping this comment will help, we thank you for your great work.

Best regards,
Simone Dutto

From: D. J. Bernstein <djb@cr.yp.to>
Sent: Friday, September 18, 2020 4:21 AM
To: pqc-comments
Cc: pqc-forum
Subject: ROUND 2 OFFICIAL COMMENT: NTRU
Attachments: signature.asc

I've posted a new paper "A discretization attack" identifying an NSA-exploitable weakness in some standardization processes:

<https://cr.yp.to/papers.html#categories>

The NISTPQC process has exactly this weakness. The paper uses NTRU vs. Kyber as a case study, showing the results of hypothetical pro-NTRU and pro-Kyber discretization attacks. The paper also identifies claims in NIST IR 8309 regarding NTRU that match the results of a hypothetical pro-Kyber discretization attack and that do not match the facts. I am therefore filing this OFFICIAL COMMENT to

- (1) dispute what NIST IR 8309 says regarding NTRU and
- (2) request transparency regarding the NISTPQC process so that the public can see whether a discretization attack was carried out.

Full details appear in the paper.

---Dan

P.S. My question "What exactly has NSA told NIST regarding NISTPQC, regarding security levels or otherwise?" (email dated 2 Aug 2020 11:50:26 +0200) remains unanswered.

From: 'Moody, Dustin (Fed)' via pqc-forum <pqc-forum@list.nist.gov>
Sent: Monday, September 28, 2020 5:17 PM
To: D. J. Bernstein
Cc: pqc-forum
Subject: [pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: NTRU

Dan,

In response to your two points in your official comment on NTRU.

1) You dispute what NISTIR 8309 says about NTRU

Reading your paper for further information, this seems to be disputing the sentences in our NTRU write-up that said "While NTRU is very efficient, it is not quite at the level of the highest-performing lattice schemes" and "NTRU has a small performance gap in comparison to KYBER and SABER". NTRU, Kyber, and SABER are all based on structured lattices and have very good performance. In our report, we wanted to highlight some of the differences between them. Thus the report noted: "In particular, NTRU has slower key generation than the schemes based on RLWE and MLWE." We did not do a detailed dive into all possible application scenarios. We agree that there are scenarios where NTRU could outperform Kyber or SABER. And note - we did select NTRU as a finalist along with Kyber and SABER.

2) You requested more transparency to know if NIST was subjected to a "discretization attack."

NIST certainly strives to run our PQC standardization process in an open and transparent way. We welcome suggestions to improve. However, we do not believe that a discretization attack took place, and don't feel we need to respond to claims of one. We looked at a variety of performance numbers when assessing the 2nd round candidates, and considered them from different viewpoints. When comparing similar candidates, the selections we make are always going to be subjective to some degree, and we understand not everybody will agree with them. These minor disagreements should not be interpreted as a failure of the NIST PQC process.

Dustin
