From: Martin R. Albrecht <martin.albrecht@royalholloway.ac.uk>
Sent: Tuesday, September 8, 2020 4:58 AM
To: pqc-comments
Cc: pqc-forum
Subject: ROUND 3 OFFICIAL COMMENT: SABER

Hi all,

Several people, Jan-Pieter D'Anvers, Dan Bernstein, Léo Ducas, Fre Vercauteren, have pointed out a potential bug in the LWE estimator https://bitbucket.org/malb/lwe-estimator/src/master/ when assessing the security of Saber.

That is, these two should return the same but didn't:

#+begin_src jupyter-python :kernel sagemath
load('https://bitbucket.org/malb/lwe-estimator/raw/HEAD/estimator.py')
n = 512
q = 8192
alpha_0 = alphaf(sqrt(10/4.0), q, sigma_is_stddev=True)
alpha_1 = alphaf(sqrt(21/4.0), q, sigma_is_stddev=True) print(primal_usvp(n, alpha_0, q, secret_distribution=alpha_1, m=n, reduction_cost_model=BKZ.ADPS16)) print(primal_usvp(n, alpha_1, q, secret_distribution=alpha_0, m=n, reduction_cost_model=BKZ.ADPS16)) #+end_src

#+RESULTS:
: rop: 2^111.8, red: 2^111.8, delta_0: 1.004104, beta: 383, d: 1004, m: 491
: rop: 2^118.0, red: 2^118.0, delta_0: 1.003955, beta: 404, d: 1022, m: 509

This was indeed due to a bug:

https://bitbucket.org/malb/lwe-estimator/commits/1c2a39d509ec91f30a098c58cada6016135e58f5
https://bitbucket.org/malb/lwe-estimator/commits/c6414bb92eaad7bcec4e572d2ae1279f2df1d3be

The output is now:

#+begin_src jupyter-python :kernel sagemath n = 512 q = 8192
alpha_0 = alphaf(sqrt(10/4.0), q, sigma_is_stddev=True)
alpha_1 = alphaf(sqrt(21/4.0), q, sigma_is_stddev=True)

print(primal_usvp(n, alpha_0, q, secret_distribution=alpha_1, m=n, reduction_cost_model=BKZ.ADPS16)) #+end_src

: Traceback (most recent call last)
: …
: NotImplementedError: secret size 0.000701 > error size 0.000484

#+begin_src jupyter-python :kernel sagemath print(primal_usvp(n, alpha_1, q, secret_distribution=alpha_0, m=n, reduction_cost_model=BKZ.ADPS16)) #+end_src

: rop: 2^118.0, red: 2^118.0, delta_0: 1.003955, beta: 404, d: 1022, m: 509

That is, the LWE esitmator – in agreement with scripts of Léo Ducas and Dan Bernstein – predicts that the primal uSVP attack requires block size 404 when n samples are available for LightSaber.

There is, however, still a (in this case minor) issue to be resolved:

https://bitbucket.org/malb/lwe-estimator/issues/46/support-small-secrets-that-are-larger-than

Cheers,
Martin

--

_pgp: https://keybase.io/martinralbrecht
_www: https://malb.io/

From: Fre <frederik.vercauteren@gmail.com>
Sent: Wednesday, September 9, 2020 6:07 AM
To: pqc-comments; pqc-forum
Subject: OFFICIAL COMMENT: SABER

Dear pq-cryptographers

We recently found an error in the security estimates of SABER as described in the round 2 document and also in the paper [1]. This error was discovered independently by Léo Ducas who ran his new leaky LWE-security estimator [2] on the SABER parameters, and was confirmed by the authors of [1].

The correct security levels are lower than those stated in the round 2 document, but do not affect the NIST levels as such. The numbers given below are expressed in coreSVP, not actual bits (the levels from the round 2 document are given in brackets):

LightSaber:
- classical: 118 (125)
- quantum: 107 (114)

Saber:
- classical: 189 (203)
- quantum: 172 (185)

FireSaber:
- classical: 260 (283)
- quantum: 236 (257)

The above numbers have now been confirmed by 3 independent implementations:
- the original LWE estimator [3]
- the leaky LWE-estimator [2]
- a script written and ran by Dan Bernstein

These all result in the same numbers so we are confident about their correctness.

We would like to thank the many researchers who have spent time and effort improving the security evaluation of SABER, with special thanks to (in alphabetical order) Martin Albrecht, Dan Bernstein, Léo Ducas, Rachel Player and Fernando Virdia.

The SABER team

[1] Martin R. Albrecht and Benjamin R. Curtis and Amit Deo and Alex Davidson and Rachel Player and Eamonn W. Postlethwaite and Fernando Virdia and Thomas Wunderer. Estimate all the {LWE, NTRU} schemes!, Cryptology ePrint Archive, Report 2018/331.

[2] https://github.com/lducas/leaky-LWE-Estimator

[3] https://bitbucket.org/malb/lwe-estimator/src/master/

From: 'Samuel Nascimento Pagliarini' via pqc-forum <pqc-forum@list.nist.gov>
Sent: Friday, September 17, 2021 9:58 AM
To: pqc-comments
Cc: pqc-forum
Subject: [pqc-forum] ROUND 3 OFFICIAL COMMENT: SABER

Dear PQC community,

I would like to bring a hardware implementation of SABER to your attention:
https://eprint.iacr.org/2021/1202
This work is a design space exploration for SABER in a 65nm ASIC technology. We have considered pipelining, memory arrangements, and logic sharing when evaluating several design candidates. All the nitty gritty details are there in the preprint.

Some details are more interesting for circuit designers than for cryptographers. So let me say that maybe the most remarkable thing about the winning design candidate is its frequency of operation: 1GHz (not that easy to achieve in 65nm, a technology that is commercially available for some ~15 years now). A significant effort was put into compromising area (not by much) for the sake of performance. The final design is still relatively compact.

While the results reported in the paper are from simulations, our team did the whole design of the winning candidate. An actual chip was sent for fabrication just a few days ago. We expect to receive fabricated parts from Taiwan by the end of the year. Then we can confirm the physical characteristics, power consumption for different operations, etc. We expect these to be very competitive.

Thanks,
Sam

--
Samuel Pagliarini
Professor, Department of Computer Systems Tallinn University of Technology (TalTech)

--

From: '赵运磊' via pqc-forum <pqc-forum@list.nist.gov>
Sent: Thursday, May 12, 2022 5:42 AM
To: pqc-comments
Cc: pqc-forum
Subject: [pqc-forum] ROUND 3 OFFICIAL COMMENT: SABER
Attachments: CNTR-Saber.JPG

Dear Saber team and dear all in PQC community:

Recently, we proposed compact NTRU based on RLWR, referred to CNTR. The paper is available from:
https://arxiv.org/abs/2205.05413

To our knowledge, CNTR has almost the smallest ciphertext size. Compared with Saber, it has smaller ciphertext size, stronger security, and lower error probabilities. By combining NTRU and RLWR, it could eliminate most of the existing patent threats. In addition, CNTR has flexible plaintext message space that is ${0,1}^{n/2}$ where n is the polynomial dimension, compared to the fixed message size of 256 bits of Saber. The comparison between CNTR and Saber is summarized in the attached table.

Any feedbacks and suggestions are appreciated from you.

All the best
Yunlei