From: Vadim Lyubashevsky <vadim.lyubash@gmail.com>
Sent: Sunday, April 14, 2019 4:59 AM
To: pqc-comments
Cc: pqc-forum@list.nist.gov; Peter Schwabe
Subject: ROUND 2 OFFICIAL COMMENT: qTESLA

Dear all,

The 2nd round submission of qTesla contains 3 different versions --

1. Essentially the version that was submitted to the 1st round (the Bai-Galbraith scheme with somewhat different parameters) which is now claimed secure in the ROM based on LWE and SIS (for which they incorrectly claimed security in the QROM based on LWE in the 1st round),
2. A version with much larger parameters which is currently claimed to be tightly secure based on LWE in the QROM,
3. A version of (1) that now also uses the main new idea from the original CRYSTALS-Dilithium submission to significantly reduce the size of the public key by letting the signer send a short hint as part of the signature in lieu of the verifier needing access to a large part of the public key. This scheme is claimed to be secure in the ROM based on LWE and SIS.

We (Peter Schwabe and I) give a complete break of (3) - in particular, a signing algorithm for arbitrary messages that doesn't require the use of the secret key and is faster than the qTesla signing algorithm (which uses the secret keys). Peter's code is available at https://cryptojedi.org/qTesla-attack.tar.bz2.

The attack is not due to a bug in the qTesla implementation or any fundamental issue in the hint-generation technique from CRYSTALS-Dilithium, but rather due to the authors of qTesla not understanding the fact that letting the signer send a hint is equivalent to letting him add an arbitrary vector of some length -- and so an adversary can do the same thing! In Dilithium, we controlled the length of this arbitrary vector (as well as the length of the signature component that it affects) and gave a security reduction from the LWE and SIS problems to the hardness of breaking the scheme. There is no such proof in qTesla. Furthermore, even for the version of qTesla without the hint (i.e. (1)), there is no reasoning in the paper as to why their parameters have the concrete hardness of SIS that they claim. One may notice that the lattice dimensions needed in qTesla are smaller than Dilithium's for the same security levels -- this is entirely due to the fact that qTesla ignores analyzing the actual hardness of the SIS problem.

Additionally, in light of the above, their statement in the change-log document that the technique for shortening the public key from (3) can be "easily extended" to their version of (2) is rather presumptuous. First, it's necessary to correctly understand the technique in order to "easily extend" it; and secondly, this was *already done* a year-and-a-half ago (before the NIST process started) in https://eprint.iacr.org/2017/916 where the scheme was called Dilithium-QROM.

So the state of affairs concerning qTesla at this point is that the parameters for (3) are completely broken, parameters for (1) have an unclear reasoning for their claimed security since the hardness of SIS is ignored, and if one wants the most compact version of (2), this was already done in https://eprint.iacr.org/2017/916

Best,

Vadim and Peter

From: Nina Bindel <nbindel@cdc.informatik.tu-darmstadt.de>
Sent: Saturday, April 27, 2019 5:59 AM
To: Vadim Lyubashevsky; pqc-comments
Cc: pqc-forum@list.nist.gov; Peter Schwabe
Subject: Re: [pqc-forum] ROUND 2 OFFICIAL COMMENT: qTESLA

Dear all,

we would like to thank Vadim and Peter for their analysis. We confirm that their attack in fact breaks a specific qTESLA variant that uses the “public key compression” idea from Dilithium (which we refer to as “public key splitting”). Specifically, the parameter sets that are affected are qTESLA-I-s, qTESLA-II-s, qTESLA-III-s, qTESLA-V-s and qTESLA-V-size-s. We remark that the seven remaining parameter sets, i.e., all the parameters that do not use the public key compression technique, are unaffected.

Accordingly, we have updated our submission package by removing the affected parameter sets indicated above. The updated package can be accessed on our webpage: https://qtesla.org/wp-content/uploads/2019/04/qTESLA_NIST_update_04.26.2019.zip

While we dearly appreciate that Vadim and Peter brought this mistake to our attention, we would like to clarify a few incorrect statements in their post about the remaining parameter sets and the original qTESLA scheme.

In round 1 and round 2, we include two security statements in our submission:

Our 1st statement (Theorem 4) says that qTESLA is EUF-CMA secure in the classical ROM as long as R-LWE and R-SIS are hard (the corresponding security reduction is not tight).

Our 2nd statement (Theorem 5) says that qTESLA is EUF-CMA secure in the QROM as long as R-LWE is hard and under the assumption of a technical conjecture (the corresponding security reduction is tight).

Right after Theorem 4 and before Theorem 5, we clearly state that: “[…] In our opinion, this second theorem [Theorem 5] is much stronger since it guarantees security against adversaries that have quantum access to a quantum random oracle. Accordingly, we always refer to Theorem 5 when discussing the security of the scheme.”

Hence, the security of qTESLA’s parameters is *only* based on the hardness of the R-LWE problem, as we also explain in numerous parts of the document that are hard to miss. Consequently, we do not state or discuss the hardness of R-SIS in our submission and we do not see an “unclear reasoning” for our claimed security.

We note that at the very beginning of Section 5, we stated that the security of qTESLA is based on R-LWE as well as R-SIS. In particular, we said: “[…] To this end we first define the hardness assumptions qTESLA is based on. This includes the ring short integer solution (R-SIS) problem and the decisional ring learning with errors (decisional R-LWE) problem.” Since these introductory sentences might have led to the misunderstanding that we need to analyze the hardness of R-SIS during our security estimates, we reformulate the two sentences in our update.

Vadim and Peter further write in their post: “So the state of affairs concerning qTesla at this point is that the parameters for (3) are completely broken, parameters for (1) have an unclear reasoning for their claimed

security since the hardness of SIS is ignored, and if one wants the most compact version of (2), this was already done in https://eprint.iacr.org/2017/916."

This assessment of qTESLA is rather presumptuous, and we obviously disagree with it.

We already dismissed the supposedly "unclear reasoning" about the security. Regarding the last statement, it surprises us that https://eprint.iacr.org/2017/916 is mentioned. This paper, advertised as "the most compact version of (2)", does not present an actual implementation, and applies a completely different size/performance trade-off (e.g., the bitlength of the modulus q is let to grow much larger than 32 bits, and the use of the efficient NTT for polynomial multiplication is not possible). Without an actual implementation it is hard to predict performance, but arguably Dilithium-QROM is expected to be quite slow. Moreover, this scheme has not been submitted to the NIST process and therefore the comment is irrelevant.

To finish, we'd like to emphasize that we are always open to scientific discussions, improvements and corrections, as long as that they are done in a professional and respectful manner. We sincerely hope this to be the case as we move forward in the process.

Sincerely,

The qTESLA team

Am 14.04.19 um 10:59 schrieb Vadim Lyubashevsky:

> Dear all,
>
> The 2nd round submission of qTesla contains 3 different versions --
>
> 1. Essentially the version that was submitted to the 1st round (the Bai-Galbraith scheme with somewhat different parameters) which is now claimed secure in the ROM based on LWE and SIS (for which they incorrectly claimed security in the QROM based on LWE in the 1st round),
> 2. A version with much larger parameters which is currently claimed to be tightly secure based on LWE in the QROM,
> 3. A version of (1) that now also uses the main new idea from the original CRYSTALS-Dilithium submission to significantly reduce the size of the public key by letting the signer send a short hint as part of the signature in lieu of the verifier needing access to a large part of the public key. This scheme is claimed to be secure in the ROM based on LWE and SIS.
>
> We (Peter Schwabe and I) give a complete break of (3) - in particular, a signing algorithm for arbitrary messages that doesn't require the use of the secret key and is faster than the qTesla signing algorithm (which uses the secret keys). Peter's code is available at https://cryptojedi.org/qTesla-attack.tar.bz2.
>
> The attack is not due to a bug in the qTesla implementation or any fundamental issue in the hint-generation technique from CRYSTALS-Dilithium, but rather due to the authors of qTesla not understanding the fact that letting the signer send a hint is equivalent to letting him add an arbitrary vector of some length -- and so an adversary can do the same thing! In Dilithium, we controlled the length of this arbitrary vector (as well as the length of the signature component that it affects) and gave a security reduction from the LWE and SIS problems to the hardness of breaking the scheme. There is no such proof in qTesla. Furthermore, even for the version of qTesla without the hint (i.e. (1)), there is no reasoning in the paper as to why their parameters have the concrete hardness of SIS that they claim. One may notice that the lattice dimensions needed in qTesla are smaller than Dilithium's for the

From: Vadim Lyubashevsky <vadim.lyubash@gmail.com>
Sent: Saturday, April 27, 2019 10:04 AM
To: Nina Bindel
Cc: pqc-comments; pqc-forum@list.nist.gov; Peter Schwabe
Subject: Re: [pqc-forum] ROUND 2 OFFICIAL COMMENT: qTESLA

Dear all,

> In round 1 and round 2, we include two security statements in our submission:
>
> Our 1st statement (Theorem 4) says that qTESLA is EUF-CMA secure in the classical ROM as long as R-LWE and R-SIS are hard (the corresponding security reduction is not tight).
>
> Our 2nd statement (Theorem 5) says that qTESLA is EUF-CMA secure in the QROM as long as R-LWE is hard and under the assumption of a technical conjecture (the corresponding security reduction is tight).
>
> Right after Theorem 4 and before Theorem 5, we clearly state that: "[…] In our opinion, this second theorem [Theorem 5] is much stronger since it guarantees security against adversaries that have quantum access to a quantum random oracle. Accordingly, we always refer to Theorem 5 when discussing the security of the scheme."
>
> Hence, the security of qTESLA's parameters is *only* based on the hardness of the R-LWE problem, as we also explain in numerous parts of the document that are hard to miss. Consequently, we do not state or discuss the hardness of R-SIS in our submission and we do not see an "unclear reasoning" for our claimed security.

When I skimmed this statement, I assumed that Theorem 5 (i.e. the claim that qTESLA is based on only LWE) was only claimed to be applicable to the parameters in Table 6 (in my original e-mail I referred to this as parameter set 2). But if you are now saying that Theorem 5 also applies to the parameters in Table 5 (parameter set 1 from my original email), then this is false -- even assuming Conjecture 6 is true (I have no problem with Conjecture 6, btw). What I am claiming is that with a Ring-SIS oracle one can break the scheme with the parameters in Table 5 - note I am not giving any concrete attack, except pointing out that one really needs to care about the hardness of Ring-SIS, and not only of Ring-LWE as you claim. I will then also point out as to what I think could be the mistake in the proof (which is not given) that the scheme is only based on Ring-LWE.

Let's, for the sake of simplicity, assume that q=2^r and the hash function H(w,m) only uses the k highest bits of w. In the case of qTESLA, k=1 or 2 (let's use k=2 here). In other words, for a bit-decomposition of w=w_{r-1}w_{r-2} … w_0, we have H(w,m) = H(w_{r-1}w_{r-2},m).

So here is the attack that uses an inhomogeneous Ring-SIS oracle. Choose w=0. Choose any message m and compute c=H(w,m). Now use the Ring-SIS oracle to find polynomials z_1,z_2 that are small (|z_1|<B and |z_2|<2^{r-2}) such that az_1-z_2 = tc+w. My signature of m is (z_1,c). Observe that this will pass the verification step because verification checks that |z_1|<B and that c=H(az_1-tc,m). Observe that az_1-tc = w+z_2, and by the way I defined w and the fact that z_2 does not have any 1's in the 2 high-order bits, means that the high-order bits of w+z_2 are 00 -- the same as the high-order bits of w (i.e. z_2 does not change the 2 high-order bits of w). And therefore verification passes because H(az_1-tc,m)=H(w+z_2,m)=H(w,m)=c.

(As a simplification, I also ignored the fact that in the real scheme, one takes elements to be centralized around 0, rather than just positive ones, but the above can be easily adapted to that case.)

If I had to guess, I would say that the mistake you made somewhere in the proof is forgetting that a z_2 not affecting the high-order bits of w can exist, and so you only tried to solve for a short z_1 such that az_1=tc+w, and concluded that

such a z_1 is information-theoretically unlikely to exist. But additionally allowing for a z_2 makes the problem a computational one based on Ring-SIS rather than an information-theoretic one. In short, the Bai-Galbraith trick of reducing the signature size gets rid of the z_2 from the signature, but it's still implicitly there and the adversary can make use of it.

Something that should be pointed out is that if the modulus q is very large with respect to the allowed sizes of z_1 and z_2, then one can prove information-theoretically that there is no solution to az_1-z_2 = tc+w (possibly with Conjecture 6), and this is how one sets the parameters for the scheme based on just LWE (e.g. [AFLT12], Dilithium-QROM and possibly the scheme with parameters in Table 6). But if one takes more efficient parameters (as in Table 5), then z_1 and z_2 do exist, and therefore you need to set parameters to make sure that this Ring-SIS problem is hard. Incidentally, this is the exact same intuition that we used to break qTESLA with the hint, except there, the value of z_2 that we could have chosen was completely unrestricted due to the hint being large. So one really, really, needs to show security based on Ring-SIS.

> This assessment of qTESLA is rather presumptuous, and we obviously disagree with it.
>
> We already dismissed the supposedly “unclear reasoning” about the security. Regarding the last statement, it surprises us that https://eprint.iacr.org/2017/916 is mentioned. This paper, advertised as “the most compact version of (2)“, does not present an actual implementation, and applies a completely different size/performance trade-off (e.g., the bitlength of the modulus q is let to grow much larger than 32 bits, and the use of the efficient NTT for polynomial multiplication is not possible). Without an actual implementation it is hard to predict performance, but arguably Dilithium-QROM is expected to be quite slow. Moreover, this scheme has not been submitted to the NIST process and therefore the comment is irrelevant.

Just because the scheme was not submitted to NIST doesn't mean that it doesn't exist. The technique for public key reduction does not care about whether NTT is used or what the modulus is -- and it was already applied to schemes that are secure in the QROM in the Dilithium-QROM paper. All I was pointing out is that the statement that the technique "can be easily extended" should have mentioned that it was already done in a different paper.

In short, I still claim everything about the security of qTESLA as in my previous email, and adding the fact that the statement between Theorems 4 and 5 is wrong when applied to the parameters in Table 5.

-Vadim

[AFLT12]: Michel Abdalla, Pierre-Alain Fouque, Vadim Lyubashevsky, Mehdi Tibouchi: "Tightly-Secure Signatures from Lossy Identification Schemes." EUROCRYPT 2012

> .
>
> The qTESLA team

Am 14.04.19 um 10:59 schrieb Vadim Lyubashevsky:

> Dear all,
>
> The 2nd round submission of qTesla contains 3 different versions --

From: Nina Bindel <nbindel@cdc.informatik.tu-darmstadt.de>
Sent: Tuesday, August 20, 2019 7:52 AM
To: pqc-comments
Cc: pqc-forum
Subject: ROUND 2 OFFICIAL COMMENT: qTESLA
Attachments: signature.asc

Dear all,

On a post to the forum on 04/27, Vadim Lyubashevsky pointed out a security gap in the analysis of the heuristic parameter sets of qTESLA regarding their SIS hardness.

We confirmed the issue and took the decision of dropping the 2nd round heuristic parameters of qTESLA. This includes qTESLA-I, qTESLA-II, qTESLA-III, qTESLA-V and qTESLA-V-size. While we have derived parameters that achieve the necessary SIS bit-hardness, we observe that Dilithium already provides a similar scheme with a similar performance footprint. Therefore, we opt for only keeping the provably-secure parameter sets (i.e, qTESLA-p-I and qTESLA-p-III) which have a larger differentiator.

We have updated the spec accordingly.

The new spec is available here:
https://qtesla.org/wp-content/uploads/2019/08/qTESLA_round2_08.19.2019.pdf

The complete updated package is here:
https://qtesla.org/wp-content/uploads/2019/08/qTESLA_NIST_update_08.19.2019.zip

The most recent version of the code is here:
https://github.com/qtesla/qTesla

We also made a few minor bug fixes to the implementation and some optimizations to the CDT-based Gaussian sampler.

Again, special thanks to Vadim for pointing out the issue.

Regards,
The qTESLA team