

---

**From:** 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov>  
**Sent:** Thursday, July 7, 2022 7:50 AM  
**To:** pqc-forum  
**Subject:** [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

Dear NIST,

The current specification of CRYSTALS-Dilithium provides two versions. One deterministic and one randomized. I strongly think NIST should also standardize a hedged version where the seed is derived from a random string, a key, and the message. The deterministic version is according to the specification not recommended in scenarios where an adversary can mount side-channel attacks. The randomized version is (I assume) not recommended in scenarios where the PRNG cannot be fully trusted. A hedged version could likely be made to protect against both side-channel attacks and weak PRNGs. The Dual\_EC\_DRBG story has thought us the importance of not blindly trusting the PRNG. Putting minimal trust in the PRNG and trying to minimize the impact of a compromised PRNG is an essential part of following zero trust principles. Having hedged signature was one of the most requested features in the comments on FIPS 186-5 (Draft).

Best Regards,  
John Preuß Mattsson

--

You received this message because you are subscribed to the Google Groups "pqc-forum" group.

To unsubscribe from this group and stop receiving emails from it, send an email to [pqc-forum+unsubscribe@list.nist.gov](mailto:pqc-forum+unsubscribe@list.nist.gov).

To view this discussion on the web visit <https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/HE1PR0701MB30501549E67A2658CAE1D30889839%40HE1PR0701MB3050.eurprd07.prod.outlook.com>.

---

**From:** pqc-forum@list.nist.gov on behalf of Vadim Lyubashevsky <vadim1980@gmail.com>  
**Sent:** Friday, July 8, 2022 5:48 AM  
**To:** John Mattsson; pqc-forum  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

Hi John, all,

On Thu, 2022-07-07 at 11:50 +0000, 'John Mattsson' via pqc-forum wrote:

Dear NIST,

The current specification of CRYSTALS-Dilithium provides two versions. One deterministic and one randomized. I strongly think NIST should also standardize a hedged version where the seed is derived from a random string, a key, and the message.

The "hedged" version can simply replace the current randomized version which does not take the key and the message as inputs. Since the key is short and the message is already hashed anyway, including these two things in the seed creation will probably have a negligible performance effect.

If people think it's a good idea, it should be easy to incorporate and I suspect that it's better having just 2 versions of the algorithm instead of 3.

Best,  
Vadim

--

You received this message because you are subscribed to the Google Groups "pqc-forum" group.

To unsubscribe from this group and stop receiving emails from it, send an email to [pqc-forum+unsubscribe@list.nist.gov](mailto:pqc-forum+unsubscribe@list.nist.gov).

To view this discussion on the web visit <https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/638cab6bdea694438ce3cfe1e89621e0f898a689.camel%40gmail.com>.

---

**From:** pqc-forum@list.nist.gov on behalf of Hanno Böck <hanno@hboeck.de>  
**Sent:** Friday, July 8, 2022 6:37 AM  
**To:** pqc-forum  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

On Fri, 08 Jul 2022 11:47:30 +0200

Vadim Lyubashevsky <vadim1980@gmail.com> wrote:

> If people think it's a good idea, it should be easy to incorporate and  
> I suspect that it's better having just 2 versions of the algorithm  
> instead of 3.

Or just 1.

Please make one the default and don't spec several different versions of the possibly major crypto algorithm of the future internet. I think if we've learned one thing from past cryptography standards it's that excess flexibility is almost always bad.

Provide as few options as possible.

--

Hanno Böck

---

**From:** Taylor R Campbell <campbell@mumble.net> on behalf of Taylor R Campbell <campbell+nist-pqc-forum@mumble.net>  
**Sent:** Friday, July 8, 2022 9:05 AM  
**To:** Vadim Lyubashevsky  
**Cc:** John Mattsson; pqc-forum  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

> Date: Fri, 08 Jul 2022 11:47:30 +0200  
> From: Vadim Lyubashevsky <vadim1980@gmail.com>  
>  
> On Thu, 2022-07-07 at 11:50 +0000, 'John Mattsson' via pqc-forum wrote:  
>> The current specification of CRYSTALS-Dilithium provides two  
>> versions. One deterministic and one randomized. I strongly think  
>> NIST should also standardize a hedged version where the seed is  
>> derived from a random string, a key, and the message.  
>  
> The "hedged" version can simply replace the current randomized version  
> which does not take the key and the message as inputs. Since the key  
> is short and the message is already hashed anyway, including these two  
> things in the seed creation will probably have a negligible  
> performance effect.  
>  
> If people think it's a good idea, it should be easy to incorporate and  
> I suspect that it's better having just 2 versions of the algorithm  
> instead of 3.

Don't have two or three versions -- have just one!

Signature creation should be defined to be a deterministic function of

1. secret key,
2. message, and
3. a randomization string.

Users can take advantage of this single standard function for many purposes:

- Users can cheaply test implementations as black boxes against standard known-answer test vectors to verify that they are incorporating all of the inputs.

Ignoring any one of the inputs in deriving rho' is invisible to verifiers, so interoperability tests will fail to detect such potentially security-destroying bugs -- even if the hard parts, the ring arithmetic and NTT, are formally verified and correct.

For randomized signatures, even if the RNG you feed into signature creation has a broken entropy source (like Sony PlayStation 3), the

implementation will still thwart cryptanalytic attacks by using a secret uniform random function of the message.

- Users can make deterministic signatures by setting the randomization string to something fixed in an application like the empty string.
- Users can cheaply mitigate fault attacks (or do anything else requiring randomized signatures) by feeding uniform random RNG output to signature creation as the randomization string. Everyone with access to a crypto API will have access to the RNG it would have used internally. No protocol changes are required for interoperability like transmitting an IV.
- Users can also `_detect_` fault attacks at somewhat higher cost: use a randomized signature as above, but pick the randomization string once and then run the signature creation function twice with the same inputs and verify whether the output is the same.

This keeps the specification simple -- `_one_` standard primitive signature creation function -- and cheaply enables defences against several different threat models: interoperable implementation bugs outside the hard parts, cryptanalytic attacks against predictable or reused per-signature secrets with a broken RNG, fault attacks.

--

You received this message because you are subscribed to the Google Groups "pqc-forum" group.

To unsubscribe from this group and stop receiving emails from it, send an email to [pqc-forum+unsubscribe@list.nist.gov](mailto:pqc-forum+unsubscribe@list.nist.gov).

To view this discussion on the web visit <https://groups.google.com/a/list.nist.gov/d/msgid/pqc-forum/20220708130519.80722609ED%40jupiter.mumble.net>.

---

**From:** pqc-forum@list.nist.gov on behalf of Vadim Lyubashevsky <vadim1980@gmail.com>  
**Sent:** Friday, July 8, 2022 9:12 AM  
**To:** Taylor R Campbell  
**Cc:** pqc-forum  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

Hi Taylor, all,

On Fri, 2022-07-08 at 13:03 +0000, Taylor R Campbell wrote:

Date: Fri, 08 Jul 2022 11:47:30 +0200

From: Vadim Lyubashevsky <[vadim1980@gmail.com](mailto:vadim1980@gmail.com)>

On Thu, 2022-07-07 at 11:50 +0000, 'John Mattsson' via pqc-forum wrote:

The current specification of CRYSTALS-Dilithium provides two versions. One deterministic and one randomized. I strongly think NIST should also standardize a hedged version where the seed is derived from a random string, a key, and the message.

The "hedged" version can simply replace the current randomized version which does not take the key and the message as inputs. Since the key is short and the message is already hashed anyway, including these two things in the seed creation will probably have a negligible performance effect.

If people think it's a good idea, it should be easy to incorporate and I suspect that it's better having just 2 versions of the algorithm instead of 3.

Don't have two or three versions -- have just one!

Signature creation should be defined to be a deterministic function of

1. secret key,
2. message, and
3. a randomization string.

- Users can make deterministic signatures by setting the randomization string to something fixed in an application like the empty string.

This is exactly what the two versions of the algorithm would look like using the "deterministic" and "hedged" modes. If you think that this counts as just one version, then great!

Best,  
Vadim

--

You received this message because you are subscribed to the Google Groups "pqc-forum" group.

To unsubscribe from this group and stop receiving emails from it, send an email to [pqc-forum+unsubscribe@list.nist.gov](mailto:pqc-forum+unsubscribe@list.nist.gov).

---

**From:** pqc-forum@list.nist.gov on behalf of Blumenthal, Uri - 0553 - MITLL <uri@ll.mit.edu>  
**Sent:** Friday, July 8, 2022 11:35 AM  
**To:** Vadim Lyubashevsky  
**Cc:** Taylor R Campbell; pqc-forum  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

I like this proposal.

Thanks!

Regards,  
Uri

On Jul 8, 2022, at 09:13, Vadim Lyubashevsky <vadim1980@gmail.com> wrote:

Hi Taylor, all,

On Fri, 2022-07-08 at 13:03 +0000, Taylor R Campbell wrote:

Date: Fri, 08 Jul 2022 11:47:30 +0200

From: Vadim Lyubashevsky <[vadim1980@gmail.com](mailto:vadim1980@gmail.com)>

On Thu, 2022-07-07 at 11:50 +0000, 'John Mattsson' via pqc-forum wrote:

The current specification of CRYSTALS-Dilithium provides two versions. One deterministic and one randomized. I strongly think NIST should also standardize a hedged version where the seed is derived from a random string, a key, and the message.

The "hedged" version can simply replace the current randomized version which does not take the key and the message as inputs. Since the key is short and the message is already hashed anyway, including these two things in the seed creation will probably have a negligible performance effect.

If people think it's a good idea, it should be easy to incorporate and I suspect that it's better having just 2 versions of the algorithm instead of 3.

Don't have two or three versions -- have just one!

Signature creation should be defined to be a deterministic function of

1. secret key,
2. message, and
3. a randomization string.

- Users can make deterministic signatures by setting the randomization

---

**From:** 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov>  
**Sent:** Friday, July 8, 2022 12:30 PM  
**To:** Hanno Böck; pqc-forum  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

Vadim Lyubashevsky wrote:

> The "hedged" version can simply replace the current randomized version

I think that is a great idea.

Hanno Böck wrote:

>Please make one the default and don't spec several different versions  
>of the possibly major crypto algorithm of the future internet. I think  
>if we've learned one thing from past cryptography standards it's that  
>excess flexibility is almost always bad.

I think there are strong reasons to have a deterministic implementation option. That enables testing which might otherwise be impossible. If the signature algorithm is implemented in a black box like an HSM, any randomized version (also hedged) implies blind trust in the HSM vendor. A deterministic version allows the user to verify that the HSM follows the specification and does not leak the private key by using bad randomness (I don't know if that is the consequence in Dilithium, but it is in ECDSA). National states have in the past controlled cryptographic hardware manufacturers like the Swiss company Crypto AG and intentionally weakened the products. Putting minimal trust in the HSM manufacturer is an essential part of following zero trust principles.

Note that the "versions" that are discussed would be the same algorithm from a protocol perspective. The verifier stays the same. The "versions" are just implementation choices for the signer.

Cheers,  
John

---

**From:** 'Scott Fluhrer (sfluhrer)' via pqc-forum <pqc-forum@list.nist.gov>  
**Sent:** Friday, July 8, 2022 2:11 PM  
**To:** John Mattsson; Hanno Böck; pqc-forum  
**Subject:** RE: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

On the other hand, requiring only the 'hedged' version would mean that an implementation must do two passes over the message to be signed. This can be a practical issue; for example, if an HSM is signing a large message, it can't hold the entire message internally, which means it must be fed the message twice.

You can, of course, get around this by hashing the message first (possibly external to the HSM), and then Dilithium signing the hash; however, that is obviously not transparent to the verifier; would it be appropriate to mandate that? The answer may be "yes"; I'm just pointing out the question...

---

**From:** 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov>  
**Sent:** Friday, July 8, 2022 12:30 PM  
**To:** Hanno Böck <hanno@hboeck.de>; pqc-forum@list.nist.gov  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

Vadim Lyubashevsky wrote:

> The "hedged" version can simply replace the current randomized version

I think that is a great idea.

Hanno Böck wrote:

>Please make one the default and don't spec several different versions  
>of the possibly major crypto algorithm of the future internet. I think  
>if we've learned one thing from past cryptography standards it's that  
>excess flexibility is almost always bad.

I think there are strong reasons to have a deterministic implementation option. That enables testing which might otherwise be impossible. If the signature algorithm is implemented in a black box like an HSM, any randomized version (also hedged) implies blind trust in the HSM vendor. A deterministic version allows the user to verify that the HSM follows the specification and does not leak the private key by using bad randomness (I don't know if that is the consequence in Dilithium, but it is in ECDSA). National states have in the past controlled cryptographic hardware manufacturers like the Swiss company Crypto AG and intentionally weakened the products. Putting minimal trust in the HSM manufacturer is an essential part of following zero trust principles.

Note that the "versions" that are discussed would be the same algorithm from a protocol perspective. The verifier stays the same. The "versions" are just implementation choices for the signer.

Cheers,  
John

---

**From:** pqc-forum@list.nist.gov on behalf of Blumenthal, Uri - 0553 - MITLL <uri@ll.mit.edu>  
**Sent:** Friday, July 8, 2022 2:29 PM  
**To:** Scott Fluhrer (sfluhrer); pqc-forum  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

> On the other hand, requiring only the 'hedged' version would mean that an implementation  
> must do two passes over the message to be signed.

I'm not sure it's true, considering that every sane email or document signer (that I'm aware of) signs the hash, rather than the document itself.

> This can be a practical issue; for example, if an HSM is signing a large message, it  
> can't hold the entire message internally, which means it must be fed the message twice.  
>  
> You can, of course, get around this by hashing the message first (possibly external to the HSM),  
> and then Dilithium signing the hash; however, that is obviously not transparent to the verifier;  
> would it be appropriate to mandate that? The answer may be "yes"; I'm just pointing out the question...

In my understanding, this has been de-facto standard for a long time. Thus, IMHO, it is perfectly appropriate to (explicitly) mandate it.

Thanks!

---

**From:** pqc-forum@list.nist.gov on behalf of Vadim Lyubashevsky  
<vadim.lyubash@gmail.com>  
**Sent:** Friday, July 8, 2022 2:37 PM  
**To:** Blumenthal, Uri - 0553 - MITLL  
**Cc:** Scott Fluhrer (sfluhrer); pqc-forum  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

On Fri, Jul 8, 2022 at 8:30 PM Blumenthal, Uri - 0553 - MITLL <uri@ll.mit.edu> wrote:

>  
>> On the other hand, requiring only the 'hedged' version would mean  
>> that an implementation  
>  
>> must do two passes over the message to be signed.  
>  
>  
>  
> I'm not sure it's true, considering that every sane email or document signer (that I'm aware of) signs the hash, rather than the document itself.

And even if it is not hashed, the Dilithium signing algorithm hashes the message right away and then never touches the actual message again. So I am not seeing why one would go over the message twice in any scenario. Just to be clear, the "hedged" mode would replace line 12 of Figure 4 in the dilithium spec with  $\rho = H(K \parallel \text{seed} \parallel \mu)$  where seed is either "" in the deterministic case or a random 512-bit string in the randomized one.

Best,  
Vadim

---

**From:** 'Scott Fluhler (sfluhler)' via pqc-forum <pqc-forum@list.nist.gov>  
**Sent:** Friday, July 8, 2022 2:44 PM  
**To:** Vadim Lyubashevsky; Blumenthal, Uri - 0553 - MITLL  
**Cc:** pqc-forum  
**Subject:** RE: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

Hmmm, sorry, I believe I misunderstood what the 'hedged' proposal was -- nevermind...

-----Original Message-----

From: pqc-forum@list.nist.gov <pqc-forum@list.nist.gov> On Behalf Of Vadim Lyubashevsky  
Sent: Friday, July 8, 2022 2:37 PM  
To: Blumenthal, Uri - 0553 - MITLL <uri@ll.mit.edu>  
Cc: Scott Fluhler (sfluhler) <sfluhler@cisco.com>; pqc-forum@list.nist.gov  
Subject: Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

On Fri, Jul 8, 2022 at 8:30 PM Blumenthal, Uri - 0553 - MITLL <uri@ll.mit.edu> wrote:

>  
>> On the other hand, requiring only the 'hedged' version would mean  
>> that an implementation  
>  
>> must do two passes over the message to be signed.  
>  
>  
>  
> I'm not sure it's true, considering that every sane email or document signer (that I'm aware of) signs the hash, rather than the document itself.

And even if it is not hashed, the Dilithium signing algorithm hashes the message right away and then never touches the actual message again. So I am not seeing why one would go over the message twice in any scenario. Just to be clear, the "hedged" mode would replace line 12 of Figure 4 in the dilithium spec with  $\rho = H(K \parallel \text{seed} \parallel \mu)$  where seed is either "" in the deterministic case or a random 512-bit string in the randomized one.

Best,  
Vadim

>  
>

---

**From:** 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov>  
**Sent:** Friday, July 8, 2022 3:18 PM  
**To:** Vadim Lyubashevsky; Blumenthal, Uri - 0553 - MITLL  
**Cc:** Scott Fluhrer (sfluhrer); pqc-forum  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

>Just to be clear, the "hedged" mode would replace line  
>12 of Figure 4 in the dilithium spec with  $\rho = H(K \parallel \text{seed} \parallel \mu)$  where  
>seed is either "" in the deterministic case or a random 512-bit  
>string in the randomized one.

Another benefit with this construction is that it decreases the chance for implementation mistakes like the infamous PS3 bug where the software signing used ECDSA with a fixed number instead of a per-message random number.

John

---

**From:** pqc-forum@list.nist.gov <pqc-forum@list.nist.gov> on behalf of Vadim Lyubashevsky <vadim.lyubash@gmail.com>  
**Date:** Friday, 8 July 2022 at 20:38  
**To:** Blumenthal, Uri - 0553 - MITLL <uri@ll.mit.edu>  
**Cc:** Scott Fluhrer (sfluhrer) <sfluhrer@cisco.com>, pqc-forum@list.nist.gov <pqc-forum@list.nist.gov>  
**Subject:** Re: [pqc-forum] OFFICIAL COMMENT: CRYSTALS-Dilithium

On Fri, Jul 8, 2022 at 8:30 PM Blumenthal, Uri - 0553 - MITLL <uri@ll.mit.edu> wrote:

>  
>> On the other hand, requiring only the 'hedged' version would mean that an implementation  
>  
>> must do two passes over the message to be signed.  
>  
>  
>  
> I'm not sure it's true, considering that every sane email or document signer (that I'm aware of) signs the hash, rather than the document itself.

And even if it is not hashed, the Dilithium signing algorithm hashes the message right away and then never touches the actual message again. So I am not seeing why one would go over the message twice in any scenario. Just to be clear, the "hedged" mode would replace line 12 of Figure 4 in the dilithium spec with  $\rho = H(K \parallel \text{seed} \parallel \mu)$  where seed is either "" in the deterministic case or a random 512-bit string in the randomized one.

Best,  
Vadim