From: Tanja Lange <tanja@hyperelliptic.org>
Sent: Friday, September 30, 2022 4:55 PM
To: pqc-comments
Cc: pqc-forum; authorcontact-mceliece-merged@box.cr.yp.to
Subject: ROUND 4 OFFICIAL COMMENT: Classic McEliece
Attachments: mods3.pdf

The attached document "modifications for round 4" specifies the Classic McEliece tweaks for round 4. We will provide updated documentation and software matching this.

Tanja, on behalf of the Classic McEliece team

From: Michael Lyons <mlyons3@gmu.edu>
Sent: Friday, September 30, 2022 5:46 PM
To: pqc-forum
Cc: Tanja Lange; pqc-forum; authorcontact-...@box.cr.yp.to; pqc-comments
Subject: Re: ROUND 4 OFFICIAL COMMENT: Classic McEliece

On page 2 in the sentence [SECDED means “single error correction, double error correction”.]
I believe the last word should be "detection".

Regards,
Mike Lyons
Cryptographic Engineering Research Group
George Mason University

From: pqc-forum@list.nist.gov on behalf of Tung Chou <blueprint@crypto.tw>
Sent: Friday, September 30, 2022 9:07 PM
To: Michael Lyons
Cc: pqc-forum; pqc-comments
Subject: Re: [pqc-forum] Re: ROUND 4 OFFICIAL COMMENT: Classic McEliece

Hi Mike,

You are right. Thank you for pointing this out.

Tung Chou

On Sat, 1 Oct 2022 at 05:45, Michael Lyons <mlyons3@gmu.edu> wrote:

> On page 2 in the sentence [SECDED means “single error correction, double error correction”.]
> I believe the last word should be "detection".
>
> Regards,
> Mike Lyons
> Cryptographic Engineering Research Group
> George Mason University

From: 'Moody, Dustin (Fed)' via pqc-forum <pqc-forum@list.nist.gov>
Sent: Monday, October 3, 2022 9:29 AM
To: Tanja Lange
Cc: pqc-forum; authorcontact-mceliece-merged@box.cr.yp.to
Subject: [pqc-forum] Re: ROUND 4 OFFICIAL COMMENT: Classic McEliece

Thanks Tanja (and team),

When do you think you can send us the updated specs and software?

Dustin

From: Tanja Lange <tanja@hyperelliptic.org>
Sent: Friday, September 30, 2022 4:54 PM
To: pqc-comments <pqc-comments@nist.gov>
Cc: pqc-forum <pqc-forum@list.nist.gov>; authorcontact-mceliece-merged@box.cr.yp.to <authorcontact-mceliece-merged@box.cr.yp.to>
Subject: ROUND 4 OFFICIAL COMMENT: Classic McEliece

The attached document "modifications for round 4" specifies the
Classic McEliece tweaks for round 4. We will provide updated
documentation and software matching this.

Tanja, on behalf of the Classic McEliece team

--

From: D. J. Bernstein <djb@cr.yp.to>
Sent: Tuesday, October 25, 2022 8:01 AM
To: pqc-comments
Cc: pqc-forum; authorcontact-mceliece-merged@box.cr.yp.to
Subject: ROUND 4 OFFICIAL COMMENT: Classic McEliece
Attachments: signature.asc

The round-4 Classic McEliece submission is available here:

https://classic.mceliece.org/nist/mceliece-20221023.tar.gz

As before, KATs have been split into a separate file:

https://classic.mceliece.org/nist/mceliece-kat-20221023.tar.gz

---D. J. Bernstein, on behalf of the Classic McEliece team

From: Wrenna Robson <wren.robson@gmail.com>
Sent: Tuesday, October 25, 2022 8:17 AM
To: pqc-forum; authorcontact-mceliece-merged@box.cr.yp.to; pqc-comments
Subject: Re: [pqc-forum] ROUND 4 OFFICIAL COMMENT: Classic McEliece

Thanks for this, Dan.

Obviously I've just had a quick glance, and will read in detail in the fullness of time, but I just want to say that I love the restructuring of the supporting documentation and the separation of content into the different documents for different purposes, and the rewriting and clarification of the content that I've seen already. It looks really great.

Best,

Wrenna

On Tue, 25 Oct 2022 at 13:01, D. J. Bernstein <djb@cr.yp.to> wrote:
>

From: D. J. Bernstein <djb@cr.yp.to>
Sent: Wednesday, July 24, 2024 2:24 PM
To: pqc-comments
Cc: pqc-forum
Subject: ROUND 4 OFFICIAL COMMENT: Classic McEliece
Attachments: signature.asc

Summary: This comment points out a mathematical explanation for _why_ Classic McEliece has much smaller ciphertexts than, e.g., Kyber.

Context: The primary motivation for this KEM is security, but a frequent observation is that this KEM has much smaller ciphertexts than any of the alternatives. An interesting consequence is that the minimum network traffic for frequently used long-term post-quantum public keys (e.g., server identity keys) is achieved by this KEM. Various post-quantum deployments, such as Rosenpass, would lose efficiency if they switched the long-term keys from this KEM to alternatives.

The underlying cryptanalytic facts are that, for any particular ciphertext size, the ranking of costs of known attacks is as follows:

* Fastest: attacking ciphertexts or keys for Kyber, NTRU, etc.
* Much slower: attacking McEliece ciphertexts.
* Slowest: attacking McEliece keys.

People often wonder whether this means that the McEliece system hasn't been studied: i.e., whether further study would eliminate the McEliece ciphertext-size advantage. Normally this question is answered with a pointer to the long history of McEliece attack papers. I'm filing this comment to highlight a different answer.

I've recently given a talk with a unified description of code/lattice cryptosystems, pinpointing what's different about the McEliece system:

https://cr.yp.to/talks/2024.07.17/slides-djb-20240717-mceliece-4x3.pdf

Attacks against all of these cryptosystems can be viewed as finding a lattice vector close to a target point. Barak similarly commented in

https://eprint.iacr.org/2017/365

on the "family" of "coding/lattice" systems, and claimed that "all the known lattice-based public-key encryption schemes can be broken using oracle access to an O(√n) approximation algorithm for the lattice closest vector problem". A closer look shows, however, that it's easy to write down systems where that level of approximation doesn't break the system. For McEliece, the gap is only polylogarithmic: the distance d from the target to the lattice is so high that exponentially many lattice points are within distance d*polylog of the target.

The fundamental reason for this difference is that, for any particular ciphertext size, keygen and dec use a quantitatively much more powerful decoder for McEliece than for the alternatives. Simple attacks that exploit the low distances allowed by the alternative decoders are faster than state-of-the-art attacks against the much larger distances allowed by the Goppa decoder used by McEliece.

It's also worth noting that the structure of the alternative decoders is the basic reason that the weakness of the alternatives is shared between ciphertexts and keys. In principle the alternatives can strengthen ther keys (see, e.g., https://eprint.iacr.org/2013/004), but this costs size (https://link.springer.com/chapter/10.1007/978-3-319-11659-4_2), making ciphertexts even easier to attack for any particular ciphertext size.

---D. J. Bernstein

From: pqc-forum@list.nist.gov on behalf of Bobby McGee <janewaykilledtuvix@gmail.com>
Sent: Thursday, July 25, 2024 3:07 PM
To: pqc-forum
Cc: D. J. Bernstein; pqc-forum; pqc-comments; pqc-comments
Subject: [pqc-forum] Re: ROUND 4 OFFICIAL COMMENT: Classic McEliece

So

- McEliece: "maximal" error with fancy algebraic decoder,
- Kyber: simple modulation and rounding ("compress/decompress").

Maybe this question is naive or ignorant, but isn't there some way to improve LWE schemes by, e.g., first encoding a message for error-correction, then using larger noise, or maybe replacing some part of the "noisy Elgamal"-style scheme with something inspired by coding theory? I think I asked this somewhere else and was told that this had been considered and there wasn't any good trade-off, but I don't remember.

On Wednesday, July 24, 2024 at 12:24:37 PM UTC-6 D. J. Bernstein wrote:

> Summary: This comment points out a mathematical explanation for _why_
> Classic McEliece has much smaller ciphertexts than, e.g., Kyber.
>
> Context: The primary motivation for this KEM is security, but a frequent
> observation is that this KEM has much smaller ciphertexts than any of
> the alternatives. An interesting consequence is that the minimum network
> traffic for frequently used long-term post-quantum public keys (e.g.,
> server identity keys) is achieved by this KEM. Various post-quantum
> deployments, such as Rosenpass, would lose efficiency if they switched
> the long-term keys from this KEM to alternatives.
>
> The underlying cryptanalytic facts are that, for any particular
> ciphertext size, the ranking of costs of known attacks is as follows:
>
> * Fastest: attacking ciphertexts or keys for Kyber, NTRU, etc.
> * Much slower: attacking McEliece ciphertexts.
> * Slowest: attacking McEliece keys.
>
> People often wonder whether this means that the McEliece system hasn't
> been studied: i.e., whether further study would eliminate the McEliece
> ciphertext-size advantage. Normally this question is answered with a
> pointer to the long history of McEliece attack papers. I'm filing this
> comment to highlight a different answer.
>
> I've recently given a talk with a unified description of code/lattice
> cryptosystems, pinpointing what's different about the McEliece system:
>
> https://cr.yp.to/talks/2024.07.17/slides-djb-20240717-mceliece-4x3.pdf
>
> Attacks against all of these cryptosystems can be viewed as finding a
> lattice vector close to a target point. Barak similarly commented in
>
> https://eprint.iacr.org/2017/365
>
> on the "family" of "coding/lattice" systems, and claimed that "all the
> known lattice-based public-key encryption schemes can be broken using
> oracle access to an O(√n) approximation algorithm for the lattice
> closest vector problem". A closer look shows, however, that it's easy
> to write down systems where that level of approximation doesn't break
> the system. For McEliece, the gap is only polylogarithmic: the distance
> d from the target to the lattice is so high that exponentially many
> lattice points are within distance d*polylog of the target.
>
> The fundamental reason for this difference is that, for any particular
> ciphertext size, keygen and dec use a quantitatively much more powerful
> decoder for McEliece than for the alternatives. Simple attacks that
> exploit the low distances allowed by the alternative decoders are faster

From: Mike Hamburg <mike@shiftleft.org>
Sent: Thursday, July 25, 2024 5:27 PM
To: Bobby McGee
Cc: pqc-forum; D. J. Bernstein; pqc-comments
Subject: Re: [pqc-forum] ROUND 4 OFFICIAL COMMENT: Classic McEliece

Hi Bobby,

It's been tried, but as far as I know, the decoders are not nearly as powerful as McEliece's Goppa decoder.

In the KEM standardization process, at least LAC, Hila5 (later was merged into Round5) and ThreeBears were using some form of error correction, and at least early versions of NewHope did as well. This gave them slightly smaller parameters at a cost in complexity.

You could of course push things farther. As an extreme example, I've tried various combinations of RLWR and somewhat more powerful error correction (E8+large-field LDPC) under the "Glowstick" family of toy schemes. These are complete toys, and do not attempt to achieve negligible decryption failure, CCA security, or side-channel resistance: the idea is just to explore how small you can make the public key and ciphertext by using a stronger code (still not as strong a code as Goppa, but also it's a soft-decision code). In this case you can achieve 32 + 243 + 323 bytes for the nonce + public key + ciphertext for "doubtfully 128-bit" parameters, or 32 + 307 + 403 bytes for "more probably 128+-bit" parameters, etc.

A more skilled coding practitioner could probably go even farther with this.

These schemes do not approach McEliece's ciphertext size, but of course the public key is much smaller.

Regards,
-- Mike

> On Jul 25, 2024, at 9:06 PM, Bobby McGee <janewaykilledtuvix@gmail.com> wrote:
>
> So
>
> - McEliece: "maximal" error with fancy algebraic decoder,
> - Kyber: simple modulation and rounding ("compress/decompress").
>
> Maybe this question is naive or ignorant, but isn't there some way to improve LWE schemes by, e.g., first encoding a message for error-correction, then using larger noise, or maybe replacing some part of the "noisy Elgamal"-style scheme with something inspired by coding theory? I think I asked this somewhere else and was told that this had been considered and there wasn't any good trade-off, but I don't remember.
>
> On Wednesday, July 24, 2024 at 12:24:37 PM UTC-6 D. J. Bernstein wrote:
>
> Summary: This comment points out a mathematical explanation for _why_ Classic McEliece has much smaller ciphertexts than, e.g., Kyber.
>
> Context: The primary motivation for this KEM is security, but a frequent observation is that this KEM has much smaller ciphertexts than any of the alternatives. An interesting consequence is that the minimum network traffic for frequently used long-term post-quantum public keys (e.g., server identity keys) is achieved by this KEM. Various post-quantum deployments, such as Rosenpass, would lose efficiency if they switched the long-term keys from this KEM to alternatives.
>
> The underlying cryptanalytic facts are that, for any particular ciphertext size, the ranking of costs of known attacks is as follows:
>
> * Fastest: attacking ciphertexts or keys for Kyber, NTRU, etc.
> * Much slower: attacking McEliece ciphertexts.
> * Slowest: attacking McEliece keys.
>
> People often wonder whether this means that the McEliece system hasn't been studied: i.e., whether further study would eliminate the McEliece ciphertext-size advantage. Normally this question is answered with a

From: 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov>
Sent: Saturday, October 26, 2024 3:11 PM
To: pqc-forum
Subject: [pqc-forum] Round 4 (Code-based KEMs) OFFICIAL COMMENT

We strongly think NIST should standardize Classic McEliece, which has properties that makes it the best choice in many different applications. We are planning to use Classic McEliece.

- Classic McEliece is the most conservative KEM and Classic McEliece category 5 is the best option for protecting various other keys (ML-KEM, ML-DSA, SLH-DSA, FN-DSA, LMS, XMSS, etc.) in transit and storage. Classic McEliece occupies a role similar to SLH-DSA, providing a very conservative security assurance.

- The small ciphertexts and good performance makes Classic McEliece the best choice for many applications of static encapsulation keys of which there are many (WireGuard, S/MIME, IMSI encryption, File encryption, Noise, EDHOC, etc.). For many such applications, key generation time is not important, and the public key can be provisioned out-of-band. When the public key is provisioned in-band, Classic McEliece has the best performance after a few hundred encapsulations. For static encapsulation use cases where ML-KEM provides the best performance, Classic McEliece is the best backup algorithm. The memory requirement can be kept low by streaming the key.

We think NIST should standardize mceliece348864 (category 1), mceliece460896 (category 3), and one of mceliece6688128, mceliece6960119, and mceliece8192128 (category 5). 261 kB and 524 kB encapsulation keys can be used where 1 MB public keys cannot.

In addition, we think NIST should standardize one of BIKE and HQC. BIKE and HQC are the best backup algorithms to ML-KEM for ephemeral encapsulation keys. Additionally, ML-KEM+BIKE and ML-KEM+HQC hybrids seems like more conservative choices than FrodoKEM while also providing better performance. We are currently not planning to use BIKE or HQC, but we would like to see a standardized backup algorithm for ML-KEM in case attacks are found. Such a backup algorithms should have a different construction than ML-KEM. This practice of implementing independent cryptographic backup algorithms has long been a guiding principle in the telecom industry.

Cheers,
John Preuß Mattsson
Expert Cryptographic Algorithms and Security Protocols, Ericsson
--

From: pqc-forum@list.nist.gov on behalf of Crick Waters <crick@patero.io>
Sent: Tuesday, October 29, 2024 12:23 PM
To: pqc-forum
Cc: John Mattsson
Subject: [pqc-forum] Re: Round 4 (Code-based KEMs) OFFICIAL COMMENT

Patero concurs and has implemented the same. We support the standardization of Classic McEliece as recommended.
Regards,
Crick
Crick Waters
CEO Patero

On Saturday, October 26, 2024 at 12:11:13 PM UTC-7 John Mattsson wrote:

> We strongly think NIST should standardize Classic McEliece, which has properties that makes it the best choice in many different applications. We are planning to use Classic McEliece.
>
> - Classic McEliece is the most conservative KEM and Classic McEliece category 5 is the best option for protecting various other keys (ML-KEM, ML-DSA, SLH-DSA, FN-DSA, LMS, XMSS, etc.) in transit and storage. Classic McEliece occupies a role similar to SLH-DSA, providing a very conservative security assurance.
>
> - The small ciphertexts and good performance makes Classic McEliece the best choice for many applications of static encapsulation keys of which there are many (WireGuard, S/MIME, IMSI encryption, File encryption, Noise, EDHOC, etc.). For many such applications, key generation time is not important, and the public key can be provisioned out-of-band. When the public key is provisioned in-band, Classic McEliece has the best performance after a few hundred encapsulations. For static encapsulation use cases where ML-KEM provides the best performance, Classic McEliece is the best backup algorithm. The memory requirement can be kept low by streaming the key.
>
> We think NIST should standardize mceliece348864 (category 1), mceliece460896 (category 3), and one of mceliece6688128, mceliece6960119, and mceliece8192128 (category 5). 261 kB and 524 kB encapsulation keys can be used where 1 MB public keys cannot.
>
> In addition, we think NIST should standardize one of BIKE and HQC. BIKE and HQC are the best backup algorithms to ML-KEM for ephemeral encapsulation keys. Additionally, ML-KEM+BIKE and ML-KEM+HQC hybrids seems like more conservative choices than FrodoKEM while also providing better performance. We are currently not planning to use BIKE or HQC, but we would like to see a standardized backup algorithm for ML-KEM in case attacks are found. Such a backup algorithms should have a different construction than ML-KEM. This practice of implementing independent cryptographic backup algorithms has long been a guiding principle in the telecom industry.
>
> Cheers,
>
> John Preuß Mattsson
>
> Expert Cryptographic Algorithms and Security Protocols, Ericsson

From: 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov>
Sent: Sunday, November 10, 2024 3:03 AM
To: pqc-forum
Subject: [pqc-forum] Re: Round 4 (Code-based KEMs) OFFICIAL COMMENT

Hi,

There seems to be substantial interest in using FrodoKEM+ECC from European governments as it is seen as a conservative choice. My thought was that ML-KEM+BIKE+ECC and ML-KEM+HQC+ECC seem like more conservative choices than FrodoKEM+ECC while also providing significantly better performance. What is conservative is a matter of opinion, but my thinking would be that a theoretical attack breaking all lattice-based crypto (ML-KEM, FrodoKEM) is more likely than attacks breaking both structured lattices and QC-MDPC.

At CFRG last week there was a comment that this was an *"Interesting thought indeed"*. Based on that I thought I could share the excel sheets I made. I don't think I have seen any public discussion of Lattice+Code+ECC hybrids before. Ericsson does currently not have any concrete plans to use FrodoKEM or Lattice+Code hybrids, but we would like a QC-MDPC KEM standardized and implemented as a backup algorithm to ML-KEM for ephemeral encapsulation. This does not mean we doubt ML-KEM, we would also like to have a NIST approved backup algorithm for AES for crypto agility.

Table 1.KEM Public key and ciphertext sizes in bytes. Total size is public key size plus ciphertext size which is a relevant measure when KEMs are used for ephemeral key exchange is protocols like TLS 1.3 and IKEv2.

Name	Category	Public key	Ciphertext	Total
ML-KEM-512	1	800	768	1568
BIKE-L1	1	1541	1573	3114
ML-KEM-512+BIKE-L1	1	2341	2341	4682
HQC-128	1	2249	4481	6730
ML-KEM-512+HQC-128	1	3049	5249	8298
FrodoKEM-640	1	9616	9720	19336
ML-KEM-768	3	1184	1088	2272
BIKE-L3	3	3083	3115	6198
ML-KEM-768+BIKE-L3	3	4267	4203	8470
HQC-192	3	4522	9026	13548
ML-KEM-768+HQC-192	3	5706	10114	15820
FrodoKEM-976	3	15632	15744	31376
ML-KEM-1024	5	1568	1568	3136
BIKE-L5	5	5122	5154	10276
ML-KEM-1024+BIKE-L5	5	6690	6722	13412
HQC-256	5	7245	14469	21714
ML-KEM-1024+HQC-256	5	8813	16037	24850
FrodoKEM-1344	5	21520	21632	43152

Table 2. KEM performance in cycles (50%) on 2023 AMD Ryzen 7 7700. Performance number from eBACS: ECRYPT Benchmarking of Cryptographic Systems by Bernstein and Lange (editors) https://bench.cr.yp.to/results-kem/amd64-hertz.html. Total is cycles for key gen + encapsulation + decapsulation.

Name	Category	Key Gen	Encapsulation	Decapsulation	Total
ML-KEM-512 (kyber512)	1	15420	24443	18693	58556
HQC-128 (hqc128round4)	1	61311	170433	283249	514993
ML-KEM-512+HQC-128	1	76731	194876	301942	573549
BIKE-L1 (bikel1)	1	459202	83286	1069392	1611880
ML-KEM-512+BIKE-L1	1	474622	107729	1088085	1670436
FrodoKEM-640 (frodokem640shake)	1	2084314	2265633	2222733	6572680
ML-KEM-768 (kyber768)	3	26537	36373	27911	90821
HQC-192 (hqc192round4)	3	145927	388479	616013	1150419
ML-KEM-768+HQC-192	3	172464	424852	643924	1241240
BIKE-L3 (bikel3)	3	1276234	177463	3365184	4818881
ML-KEM-768+BIKE-L3	3	1302771	213836	3393095	4909702
FrodoKEM-976 (frodokem976shake)	3	4272608	4592978	4483035	13348621
ML-KEM-1024 (kyber1024)	5	34305	48320	38330	120955
HQC-256 (hqc256round4)	5	295441	733761	1192579	2221781
ML-KEM-1024+HQC-256	5	329746	782081	1230909	2342736
BIKE-L5	5	?	?	?	?
ML-KEM-1024+BIKE-L5	5	?	?	?	?
FrodoKEM-1344 (frodokem1344shake)	5	7309062	7857621	7702139	22868822

I apologize for any potential errors in the tables above or if I chose the wrong algorithm variants from eBACS. I could not find performance numbers for BIKE-L5 on eBACS. FrodoKEM with SHAKE seemed like the fairest comparison to the other SHAKE based algorithms. I am a bit skeptical to the use of AES as a KDF https://csrc.nist.gov/csrc/media/Projects/crypto-publication-review-project/documents/initial-comments/sp800-38b-initial-public-comments-2024.pdf

The Excel documents are available here.
https://github.com/emanjon/KEMs

Cheers,
John Preuß Mattsson

From: pqc-forum@list.nist.gov on behalf of Christopher J Peikert <cpeikert@alum.mit.edu>
Sent: Wednesday, November 13, 2024 10:55 PM
To: John Mattsson
Cc: pqc-forum
Subject: Re: [pqc-forum] Re: Round 4 (Code-based KEMs) OFFICIAL COMMENT

On Sun, Nov 10, 2024 at 3:03 AM 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov> wrote:

> There seems to be substantial interest in using FrodoKEM+ECC from European governments as it is seen as a conservative choice. My thought was that ML-KEM+BIKE+ECC and ML-KEM+HQC+ECC seem like more conservative choices than FrodoKEM+ECC while also providing significantly better performance. What is conservative is a matter of opinion, but my thinking would be that a theoretical attack breaking all lattice-based crypto (ML-KEM, FrodoKEM) is more likely than attacks breaking both structured lattices and QC-MDPC.

Could you explain your thinking behind this conclusion?

Both "structured" (module) lattices and quasi-cyclic (QC) codes have many "cyclic symmetries" that are not present in the "unstructured" lattices that FrodoKEM uses. Hypothetically, these cyclic symmetries could someday lead to a break that does not apply to unstructured lattices or codes.

I'd say we can only speculate (not draw any useful conclusions) as to which dimension is more likely to yield a serious attack: lattices vs codes, or structured vs unstructured.

Sincerely yours in cryptography,
Chris

From: 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov>
Sent: Thursday, November 14, 2024 12:51 AM
To: Christopher J Peikert
Cc: pqc-forum
Subject: Re: [pqc-forum] Re: Round 4 (Code-based KEMs) OFFICIAL COMMENT

Hi Chris,

It was not meant as a conclusion. As you write it is very much a matter of speculation or "educated guesses". I am happy that my post lead to a discussion. Ignoring attacks on specific algorithms (which are also important), I think there are four high-level probabilities:

- A. Probability of attacks on all crypto using binary Goppa codes
- B. Probability of attacks on all crypto using QC-MDPC codes
- C. Probability of attacks on all crypto using unstructured lattices
- D. Probability of attacks on all crypto using structured lattices.

I think most people agree that A < B and C < D. None of the probabilities are independent but most people probably agree that "A and B" and "C and D" are much more dependant than the other pairs.

Is B * D < C? I don't know.

Any discussion is bound to be similar to previous discussion regarding the security of RSA, FFDH, ECC on random curves like Brainpool, and ECC on curves with special forms like P-256 and Curve25519, which I think there are still different opinions about.

Cheers,
John

From: Christopher J Peikert <cpeikert@alum.mit.edu>
Date: Thursday, 14 November 2024 at 04:55
To: John Mattsson <john.mattsson@ericsson.com>
Cc: pqc-forum <pqc-forum@list.nist.gov>
Subject: Re: [pqc-forum] Re: Round 4 (Code-based KEMs) OFFICIAL COMMENT

On Sun, Nov 10, 2024 at 3:03 AM 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov> wrote:

> There seems to be substantial interest in using FrodoKEM+ECC from European governments as it is seen as a conservative choice. My thought was that ML-KEM+BIKE+ECC and ML-KEM+HQC+ECC seem like more conservative choices than FrodoKEM+ECC while also providing significantly better performance. What is conservative is a matter of opinion, but my thinking would be that a theoretical attack breaking all lattice-based crypto (ML-KEM, FrodoKEM) is more likely than attacks breaking both structured lattices and QC-MDPC.

Could you explain your thinking behind this conclusion?

Both "structured" (module) lattices and quasi-cyclic (QC) codes have many "cyclic symmetries" that are not present in the "unstructured" lattices that FrodoKEM uses. Hypothetically, these cyclic symmetries could someday lead to a break that does not apply to unstructured lattices or codes.

I'd say we can only speculate (not draw any useful conclusions) as to which dimension is more likely to yield a serious attack: lattices vs codes, or structured vs unstructured.

Sincerely yours in cryptography,
Chris

From: pqc-forum@list.nist.gov on behalf of Watson Ladd <watsonbladd@gmail.com>
Sent: Thursday, November 14, 2024 1:14 AM
To: Christopher J Peikert
Cc: John Mattsson; pqc-forum
Subject: Re: [pqc-forum] Re: Round 4 (Code-based KEMs) OFFICIAL COMMENT

On Wed, Nov 13, 2024, 7:55 PM Christopher J Peikert <cpeikert@alum.mit.edu> wrote:

> On Sun, Nov 10, 2024 at 3:03 AM 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov> wrote:
>
> > There seems to be substantial interest in using FrodoKEM+ECC from European governments as it is seen as a conservative choice. My thought was that ML-KEM+BIKE+ECC and ML-KEM+HQC+ECC seem like more conservative choices than FrodoKEM+ECC while also providing significantly better performance. What is conservative is a matter of opinion, but my thinking would be that a theoretical attack breaking all lattice-based crypto (ML-KEM, FrodoKEM) is more likely than attacks breaking both structured lattices and QC-MDPC.
>
> Could you explain your thinking behind this conclusion?
>
> Both "structured" (module) lattices and quasi-cyclic (QC) codes have many "cyclic symmetries" that are not present in the "unstructured" lattices that FrodoKEM uses. Hypothetically, these cyclic symmetries could someday lead to a break that does not apply to unstructured lattices or codes.
>
> I'd say we can only speculate (not draw any useful conclusions) as to which dimension is more likely to yield a serious attack: lattices vs codes, or structured vs unstructured.

Having two structured medium modulus systems vs one using a tiny modulus and one a medium one seems to put eggs in small baskets (also related) vs the other.

Codes also can introduce much more noise for the same ciphertext size.

> Sincerely yours in cryptography,
> Chris

From: pqc-forum@list.nist.gov on behalf of D. J. Bernstein <djb@cr.yp.to>
Sent: Friday, November 15, 2024 9:41 AM
To: pqc-forum
Subject: Re: [pqc-forum] Re: Round 4 (Code-based KEMs) OFFICIAL COMMENT
Attachments: signature.asc

Christopher J Peikert writes:
> Both "structured" (module) lattices and quasi-cyclic (QC) codes have
> many "cyclic symmetries" that are not present in the "unstructured"
> lattices that FrodoKEM uses. Hypothetically, these cyclic symmetries
> could someday lead to a break that does not apply to unstructured lattices or codes.

This isn't hypothetical. As one of many examples, BIKE issued a security patch in response to an attack from last year. That attack---

https://link.springer.com/chapter/10.1007/978-3-031-38548-3_3

---exploits decryption failures produced by, e.g., the polynomial p = 1+x+x^2+x^3+x^4+x^5+x^159+x^314+... having a big pileup of early terms in the products p, px, px^2, px^3, px^4, px^5 where a similar MDPC system avoiding polynomials would have lower probability of decryption failures.

https://epubs.siam.org/doi/abs/10.1137/1.9781611974331.ch64 is a lattice example, breaking Gentry's STOC 2009 FHE system for cyclotomics. All of the followup attacks listed in

https://ntruprime.cr.yp.to/latticerisks-20211031.pdf#subsection.1.1.2

are also exploiting the polynomial structure visible in public keys.

Real attack examples like this justify asking whether FrodoKEM would do a better job than BIKE or HQC as a backup for Kyber. On the other hand, there are also many other attack avenues. For example:

* FrodoKEM says it's an "instantiation and implementation" of https://eprint.iacr.org/2010/613---but that paper proposed dimension just 256 for security "about" 2^150 ("at least" 2^128). There have been many improvements in lattice attacks since then.

* In 2023, FrodoKEM issued a security patch to try to block the attack from https://cr.yp.to/papers/lprrr-20221114.pdf#frodooo. The underlying design flaw was triggered by performance pressure that arose from FrodoKEM ciphertexts actually containing multiple ciphertexts internally---which in turn is forced by the way that FrodoKEM avoids polynomials.

https://eprint.iacr.org/2023/947 presents a theorem saying that a particular lattice system provides 2^128 QROM IND-CCA2 security if there aren't further advances in general lattice attacks; but that system is much bigger than FrodoKEM-1344. It's entirely possible for FrodoKEM to have further weaknesses that aren't shared by other lattice systems.

I do think that FrodoKEM-1344 has lower risk overall than Kyber-1024, and _some_ of the reasons that I'm concerned about Kyber are also reasons to be concerned about BIKE and HQC. But I also wouldn't be surprised by further FrodoKEM attacks that don't apply to Kyber, or by further FrodoKEM+Kyber attacks that don't apply to BIKE or to HQC.

> I'd say we can only speculate (not draw any useful conclusions) as to
> which dimension is more likely to yield a serious attack: lattices vs
> codes, or structured vs unstructured.

There's extensive literature on quantifying risk. See, e.g.,

https://www.cambridge.org/nl/universitypress/subjects/statistics-probability/optimization-or-and-risk/probabilistic-risk-analysis-foundations-and-methods

for an introduction to the topic.

Risk analysis includes comparisons of models to the available data. As a starting point, https://cr.yp.to/papers.html#qrcsp quantifies currently known failure rates of round-1 submissions to NISTPQC. Out of the 69 submissions, 48% are already

known to have failed, so we aren't in an information vacuum here. Furthermore, the attacks against NISTPQC are part of a broader literature successfully attacking many more problems, and it's certainly possible to collect more data regarding that.

But it's important for the question to be clear. Does "serious attack"
mean something other than breaking the minimum security level allowed in NISTPQC? Which dividing line is under consideration between "lattices"
and "codes"? (See https://cr.yp.to/talks.html#2024.07.17.) Et cetera.

---D. J. Bernstein

From: 'John Mattsson' via pqc-forum <pqc-forum@list.nist.gov>
Sent: Saturday, November 16, 2024 4:11 AM
To: Watson Ladd; Christopher J Peikert; D. J. Bernstein
Cc: pqc-forum
Subject: Re: [pqc-forum] Re: Round 4 (Code-based KEMs) OFFICIAL COMMENT

D. J. Bernstein wrote:
"I do think that FrodoKEM-1344 has lower risk overall than Kyber-1024, and _some_ of the reasons that I'm concerned about Kyber are also reasons to be concerned about BIKE and HQC. But I also wouldn't be surprised by further FrodoKEM attacks that don't apply to Kyber, or by further FrodoKEM+Kyber attacks that don't apply to BIKE or to HQC."

Watson Ladd wrote:
"Having two structured medium modulus systems vs one using a tiny modulus and one a medium one seems to put eggs in small baskets (also related) vs the other."

Christopher J Peikert wrote:
"Both "structured" (module) lattices and quasi-cyclic (QC) codes have many "cyclic symmetries" that are not present in the "unstructured" lattices that FrodoKEM uses. Hypothetically, these cyclic symmetries could someday lead to a break that does not apply to unstructured lattices or codes."

Does anybody have any speculation/educated guess/risk analysis regarding how likely/unlikely it would be with a single attack that applies to both structured lattices (i.e., ML-KEM) and structured codes (BIKE/HQC)?

While we cannot draw any certain conclusions, a lot of us need to make decisions on what to implement very soon. Some of these decisions are already taken. Given that ML-KEM will be the default in most cases, it would be good with more discussion if FrodoKEM, BIKE, or HQC is the best backup algorithm to ML-KEM for ephemeral key exchange.

Note that in many industrial use cases, high performance is very important for usage, and usage is of course essential for practical security of systems. TLS 1.3 (2018) tragically specifies and recommends psk_ke without asymmetric keying, allows reuse of ephemeral private keys in violation with SP 800-56A, and removes the mechanism to do asymmetric rekeying inside a connection. This despite X25519 having extremely good performance. Ericsson would like to see frequent rekeying with ephemeral keys based on data and time as a strong general recommendation in the future.

Cheers,
John
--

From: Loganaden Velvindron <loganaden@gmail.com>
Sent: Sunday, January 12, 2025 12:36 PM
To: pqc-comments
Subject: ROUND 4 OFFICIAL COMMENT: Classic McEliece

On behalf of the cyberstorm.mu,

Classic McEliece has a different design compared to Lattice based structures such as ML-KEM or NTRU.
Classic McEliece has useful applications such as VPNs such as Rosenpass and MulladVPN.
We would suggest that Classic McEliece be standardized for those niche applications such as VPNs.

We would also suggest a backup candidate for ML-KEM. We would hope for an NTRU or NTRU variant such as NTRU Prime be (re)considered for a round 5 or as an "alternative" candidate.

Loganaden Velvindron
(cyberstorm.mu)