From: Apon, Daniel C. (Fed)
To: Perlner, Ray A. (Fed); Moody, Dustin (Fed); Dang, Quynh H. (Fed); Robinson, Angela Y. (Fed); Smith-Tone, Daniel C. (Fed); internal-pqc
Subject: RE: 2nd Draft response to Kyber (A few minor editorial changes for grammar, plus a slight expansion of the conclusion)
Date: Monday, June 8, 2020 2:58:35 PM

We are a warm and friendly bunch =)

From: Apon, Daniel C. (Fed)
Sent: Monday, June 8, 2020 2:57 PM
To: Perlner, Ray A. (Fed) <ray.perlner@nist.gov>; Moody, Dustin (Fed) <dustin.moody@nist.gov>; Dang, Quynh H. (Fed) <quynh.dang@nist.gov>; Robinson, Angela Y. (Fed) <angela.robinson@nist.gov>; Smith-Tone, Daniel C. (Fed) <daniel.smith@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: RE: 2nd Draft response to Kyber (A few minor editorial changes for grammar, plus a slight expansion of the conclusion)

Replace ""As always, we encourage more feedback and community discussion." with
"As always, we warmly encourage more feedback and community discussion."

and

Replace "[..] Thanks again.

NIST pqc team" with

"[..]

Thank you,
NIST PQC Team"

From: Perlner, Ray A. (Fed) <ray.perlner@nist.gov>
Sent: Monday, June 8, 2020 2:55 PM
To: Moody, Dustin (Fed) <dustin.moody@nist.gov>; Dang, Quynh H. (Fed) <quynh.dang@nist.gov>; Robinson, Angela Y. (Fed) <angela.robinson@nist.gov>; Smith-Tone, Daniel C. (Fed) <daniel.smith@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: RE: 2nd Draft response to Kyber (A few minor editorial changes for grammar, plus a slight expansion of the conclusion)

Sure

How about the following at the end?

“As always, we encourage more feedback and community discussion. Thanks again.

NIST pqc team”

From: Moody, Dustin (Fed) <dustin.moody@nist.gov>
Sent: Monday, June 8, 2020 10:54 AM
To: Perlner, Ray A. (Fed) <ray.perlner@nist.gov>; Dang, Quynh H. (Fed) <quynh.dang@nist.gov>; Robinson, Angela Y. (Fed) <angela.robinson@nist.gov>; Smith-Tone, Daniel C. (Fed) <daniel.smith@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: Re: 2nd Draft response to Kyber (A few minor editorial changes for grammar, plus a slight expansion of the conclusion)

I like the response.

Should we add in a line at the end that we would appreciate feedback and hope to encourage more community discussion?

Dustin

From: Perlner, Ray A. (Fed) <ray.perlner@nist.gov>
Sent: Monday, June 8, 2020 10:14 AM
To: Dang, Quynh H. (Fed) <quynh.dang@nist.gov>; Robinson, Angela Y. (Fed) <angela.robinson@nist.gov>; Smith-Tone, Daniel C. (Fed) <daniel.smith@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: RE: 2nd Draft response to Kyber (A few minor editorial changes for grammar, plus a slight expansion of the conclusion)

Dear Kyber team (And Dan and whoever else is listening)
Thank you for these details regarding the Kyber team’s approach to security estimates.

The Kyber team asks:

“We are very curious to learn about the position that NIST takes on this and more generally on the importance of the gate-count metric for attacks that require access to large memories”

We would like to preface our response by noting that all 5 security categories are designed to be well beyond the reach of any current technology that could be employed to implement a computational attack. The reason we distinguish among security levels beyond what’s currently feasible is as a hedge against improvements in both technology and cryptanalysis. In order to be a good hedge against technology improvements, a model must be realistic, not just for current technology, but for future technology (otherwise we wouldn’t consider quantum attacks at all.) This means that we should be more convinced by hard physical limits than the particular limitations of current technology (although if something is difficult for current technology there often is a fundamental physical reason why, even if it's not obvious.) As a hedge against improvements in cryptanalysis, it's not 100% clear that a realistic model of computation, even for future technology, is optimal in providing that hedge. The more complicated a model of computation is, the harder it is to optimize an attack for that model, and the less certain we can be that we are truly measuring the best attacks in that model. As such, the gate model has the virtue of being simpler and easier to analyze than more realistic models.

With that said, let's assume we are aiming for a realistic model of computation.

There are a number of ways that memory intensive attacks might incur costs beyond what's suggested by the basic gate model

First of all, there is the sheer cost of hardware. This is what seems to be alluded to by the Kyber team's observation that

"For a fun sense of scale: a micro-SD card has a 2^3.5 mm^2 footprint (if you stand it on the short end). A planar sheet of terabyte micro-SD cards the size of New York City (all five boroughs, 800 km^2 ~ 2^49.5 mm^2) would hold 2^89 bits."

This sounds quite impressive, but note that if we were to try to perform 2^143 bit operations with current hardware, we could for example invest in high-end bitcoin mining equipment. By our calculations, a planar array of high-end mining boxes could cover New York city 30 times over and still take 10000 years to perform 2^143 bit operations (Assuming you can dissipate all the heat involved somehow.) Moreover, this equipment would need to be powered by an array of solar cells (operating at 20% efficiency) covering all of North America. As such, we are unconvinced based on hardware costs alone that 2^89 bits of memory is enough to push an attack above level 1.

A second way memory could incur costs is latency. A number of submitters have pointed out that information cannot travel faster than the speed of light, and we are inclined to agree. However, some have gone further and suggested that memory must be arranged in a 2-dimensional fashion. We are unconvinced, as modern supercomputers tend to use a meaningfully 3-dimensional arrangement of components (although the components themselves tend to be 2-dimensional.) It's also worth noting that sending data long distance can be done at near light speed, (e.g. via fiber optics), but data travels somewhat slower over short distances in most technology we are aware of.

Finally, there is energy cost. Accessing far-away memory tends to cost more energy than nearby memory. One might in fact argue that what gate cost is really trying to measure is energy consumption. Some submitters have advocated modeling this by using a gate model of computation where only local nearest neighbor interactions are allowed. This however, seems potentially too pessimistic, because we would be modeling things like long distance fiber optic connections by a densely packed series of gates. It seems clear that sending a bit over a kilometer of fiber optic, while more expensive than sending a bit through a single gate is less expensive than sending a bit through a densely packed series of gates a kilometer long. There is also at least a logarithmic gate cost in the literal sense, since you need logarithmically many branch points to get data to and from the right memory address. For random access queries to extremely small amounts of data, the cost per bit gets multiplied by a logarithmic factor since you need to send the address of the data a good portion of the way, but the algorithms in question are generally accessing consecutive chunks of memory larger than the memory address, so we can probably only assume that random access queries have a log-memory-size cost per bit as opposed to a log^2-memory-size cost per bit, at least based on this particular consideration.

Overall, we think it's fairly easy to justify treating a random access query for b consecutive bits in a memory of size N as equivalent to a circuit with depth N^(1/3), and using log(N)(b+log(N)) gates. (Assuming that the random access query can't be replaced with a cheaper local nearest neighbor circuit.) This is almost certainly an underestimate, but it's somewhat difficult to justify treating memory as much more expensive than this, without making assumptions about future technology that might be wrong.

So what does that mean for NIST's decisions? We recognize that, given known attacks, lattice schemes like Kyber are the category most likely to have their security underestimated by the nonlocal gate model as compared to a more realistic memory model. However, without very rigorous analysis, it is a bit difficult to say by how much. In cases where we think the possible attack space is well explored, and the gate model cost of all known attacks can be shown to be very close to that of breaking AES or SHA at the appropriate level, and the attacks in question can be shown to need a lot of random access queries to a large memory, we're currently inclined to give submitters the benefit of the doubt that memory costs can cover the difference. To the extent any of those assumptions do not hold (e.g. if the gate cost isn't very close to what it should be ignoring memory costs) we're less inclined. We're planning on doing a more thorough internal review of this issue early in the third round. If we think the security of a parameter set falls short of what it should be, but we still like the scheme, we will most likely respond by asking the submitters to alter the parameters to increase the security margin, or to provide a higher security parameter set, but we would prefer not to have to do this.

NIST pqc team

From: Dang, Quynh H. (Fed) <quynh.dang@nist.gov>
Sent: Saturday, June 6, 2020 8:34 AM
To: Robinson, Angela Y. (Fed) <angela.robinson@nist.gov>; Smith-Tone, Daniel C. (Fed) <daniel.smith@nist.gov>; Perlner, Ray A. (Fed) <ray.perlner@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: Re: Draft response to Kyber

Hi all,

I like Ray's conclusion at the end.

But, I am not sure I like " the attacks in question can be shown to need a lot of random access queries to a large memory, we're inclined to give submitters the benefit of the doubt that memory costs can cover the difference." because we did not say that in our call for proposal. I guess many algorithms could have been designed differently to improve their performances if their authors knew that up front.

Also, the immediate question is what difference and what memory cost are comparable ?

For the phrase: " If we think the security of a parameter set falls short of what it should be" , how short for each security level is acceptable ?

I guess the former can have a good answer. I am afraid that the latter is hard to answer.

Quynh.

From: Robinson, Angela Y. (Fed) <angela.robinson@nist.gov>
Sent: Friday, June 5, 2020 6:57 PM
To: Smith-Tone, Daniel C. (Fed) <daniel.smith@nist.gov>; Perlner, Ray A. (Fed) <ray.perlner@nist.gov>; internal-pqc <internal-pqc@nist.gov>
Subject: RE: Draft response to Kyber

The only distracting line I caught was: "The more complicated a model of computation is, the harder it is to optimize an attack for that model, and the less certain we can be we are truly measuring…" Maybe insert a "that": "…and the less certain we can be [that] we are truly measuring…"

I'm assuming the typos can wait. I didn't want to go through them all now since the content may change.

Angela

Sent from Mail for Windows 10

From: Smith-Tone, Daniel C. (Fed)
Sent: Friday, June 5, 2020 6:17 PM
To: Perlner, Ray A. (Fed); internal-pqc
Subject: RE: Draft response to Kyber

I like it. Spotted a grammar mistake somewhere but I forgot where now. Someone else will spot it, because it is distracting.

From: Perlner, Ray A. (Fed) <ray.perlner@nist.gov>
Sent: Friday, June 5, 2020 6:05 PM
To: internal-pqc <internal-pqc@nist.gov>
Subject: Draft response to Kyber

Dear Kyber team (And Dan and whoever else is listening)
Thank you for these details regarding the Kyber team's approach to security estimates.

The Kyber team asks:

"We are very curious to learn about the position that NIST takes on this and more generally on the importance of the gate-count metric for attacks that require access to large memories"

We would like to preface our response by noting that all 5 security categories are designed to be well beyond the reach of any current technology that could be employed to implement a computational attack. The reason we distinguish among security levels beyond what's currently feasible is as a hedge against improvements in both technology and cryptanalysis. In order to be a good hedge against technology improvements, a model must be realistic, not just for current technology, but for future technology (otherwise we wouldn't consider quantum attacks at all.) This means that we should be more convinced by hard physical limits, than the particular limitations of current technology (although if something is difficult for current technology there often is a fundamental physical reason why, even if it's not obvious.) As a hedge against improvements in cryptanalysis, it's not 100% clear that a realistic model of computation, even for future technology, is optimal in providing that hedge. The more complicated a model of computation is, the harder it is to optimize an attack for that model, and the less certain we can be we are truly measuring the best attacks in that model. As such, the gate model has the virtue of being simpler and easier to analyze than more realistic models.

With that said, let's assume we are aiming for a realistic model of computation.

There are a number of ways that memory intensive attacks might incur costs beyond what's suggested by the basic gate model

First of all, there is the sheer cost of hardware. This is what seems to be alluded to by the Kyber team's observation that

"For a fun sense of scale: a micro-SD card has a 2^3.5 mm^2 footprint (if you stand it on the short end). A planar sheet of terabyte micro-SD cards the size of New York City (all five boroughs, 800 km^2 ~ 2^49.5 mm^2) would hold 2^89 bits."

This sounds quite impressive, but note that if we were to try to perform 2^143 bit operations with current hardware, we could for example invest in high end bitcoin mining equipment. By our calculations, a planar array of high end mining boxes could cover New York city 30 times over and still take 10000 years to perform 2^143 bit operations (Assuming you can dissipate all the heat involved somehow.) Moreover, this equipment would need to be powered by an array of solar cells (operating a 20% efficiency) covering all of North America. As such, we are unconvinced based on hardware costs alone that 2^89 bits of memory is enough to push an attack above level 1.

A second way memory could incur costs is latency. A number of submitters have pointed out that information cannot travel faster than the speed of light, and we are inclined to agree. However, some have gone further and suggested that memory must be arranged in a 2 dimensional fashion. We are unconvinced, as modern supercomputers tend to use a meaningfully 3 dimensional arrangement of components (although the components themselves tend to be 2 dimensional.) It's also worth noting that sending data long distance can be done at near light speed, (e.g. via fiber optics), but data travels somewhat slower over short distances in most technology we are aware of.

Finally, there is energy cost. Accessing far-away memory tends to cost more energy than nearby memory. One might in fact argue that what gate cost is really trying to measure is energy consumption. Some submitters have advocated modeling this by using a gate model of computation where only local nearest neighbor interactions are allowed. This however, seems potentially too pessimistic, because we would be modeling things like long distance fiber optic connections by a densely packed series of gates. It seems clear that sending a bit over a kilometer of fiber optic, while more expensive than sending a bit through a single gate is less expensive than sending a bit through a densely packed series of gates a kilometer long. There is also at least a logarithmic gate cost in the literal sense, since you need logarithmically many branch points to get data to and from the right memory address. For random access queries to extremely small amounts of data, the cost per bit gets multiplied by a logarithmic factor since you need to send the address of the data a good portion of the way, but the algorithms in question are generally accessing consecutive chunks of memory larger than the memory address, so we can probably only assume that random access queries have a log-memory-size cost per bit as opposed to a log^2-memory-size cost per bit, at least based on this particular consideration.

Overall, we think it's fairly easy to justify treating a random access query for b consecutive bits in a memory of size N as equivalent to a circuit with depth N^(1/3), and using log(N)(b+log(N)) gates. (Assuming that the random access query can't be replaced with a cheaper local nearest neighbor circuit.) This is almost certainly an underestimate, but it's somewhat difficult to justify treating memory as much more expensive than this, without making assumptions about future technology that might be wrong.

So what does that mean for NIST's decisions? We recognize that, given known attacks, lattice schemes like Kyber are most likely to have their security underestimated by the gate model as compared to a more realistic memory model. However, without very rigorous analysis, it is a bit difficult to say by how much. In cases where we think the possible attack space is well explored, and the gate model cost of all known attacks can be shown to be very close to that of breaking AES or SHA at the appropriate level, and the attacks in question can be shown to need a lot of random access queries to a large memory, we're inclined to give submitters the benefit of the doubt that memory costs can cover the difference. To the extent any of those assumptions do not hold (e.g. if the gate cost isn't very close to what it should be ignoring memory costs) we're less inclined. If we think the security of a parameter set falls short of what it should be, but we still like the scheme, we will most likely respond by asking the submitters to alter the parameters to increase the security margin, or to provide a higher security parameter set.

NIST pqc team