From: Liu, Yi-Kai (Fed)
To: Peralta, Rene C. (Fed)
Subject: Re: A couple easy steps toward moving our stuff for post-quantum crypto
Date: Friday, July 15, 2016 7:49:57 PM

Hi Rene,

Sorry for not replying earlier, I was out of town. I think I subscribed to the forum too, but I haven't gotten any emails. The listserv says there aren't any messages: https://email.nist.gov/pipermail/pqc-forum/

Cheers,

--Yi-Kai

From: Peralta, Rene (Fed)
Sent: Wednesday, June 29, 2016 11:58:55 AM
To: Liu, Yi-Kai (Fed)
Subject: Re: A couple easy steps toward moving our stuff for post-quantum crypto

Hi Yi-Kai,

Is there any traffic in the forum. I registered in March, but I don't think I have gotten any emails.

Rene.

From: Liu, Yi-Kai (Fed)
Sent: Friday, June 17, 2016 1:28 PM
To: Dang, Quynh (Fed); Kelsey, John M. (Fed); internal-crypto
Subject: Re: A couple easy steps toward moving our stuff for post-quantum crypto

Hi everyone,

Just to add a bit to John's advice:

One of the big challenges with standardizing post-quantum cryptosystems is understanding the requirements of the higher-level protocols and applications that will use PQC. Obviously, protocol designers want everything to be faster and more compact. But that's not all going to happen. So given that PQC is likely going to have some kind of performance penalty, what kinds of tradeoffs should we make, to ensure that this change doesn't break anything too badly?

This motivates my suggestion: when thinking about how our current protocols can be adapted to use new public key algorithms, try to *document* any hard limits on key sizes and other performance characteristics. What, exactly, could post-quantum crypto do that would really ruin your day?

Then *communicate* this information to the NIST post-quantum crypto team. And we will try not to do that thing. There's a discussion forum and a contact email here:

http://csrc.nist.gov/groups/ST/post-quantum-crypto/
Post-Quantum crypto Project - NIST.gov<http://csrc.nist.gov/groups/ST/post-quantum-crypto/>
csrc.nist.gov

Post-Quantum crypto Project . In recent years, there has been a substantial amount of research on quantum computers – machines that exploit quantum mechanical ...

Cheers,

--Yi-Kai

From: Dang, Quynh (Fed)
Sent: Friday, June 17, 2016 8:50 AM
To: Kelsey, John M. (Fed); internal-crypto
Subject: Re: A couple easy steps toward moving our stuff for post-quantum crypto

Hi John,

(c) indicates to me that increasing key sizes of public key algorithms to deal with small quantum computers temporarily (short term solution for existing classical public key crypto). I think we are currently providing that option with RSA and DH. For ECC, increasing its key sizes are not that effective comparing to RSA and DH. I don't know what other classical public key algorithms that we are going to standardize in the future. For hash-based signatures, moving to bigger hashes helps increasing resistance to quantum attacks.

Regards,
Quynh.

From: John Kelsey <john.kelsey@nist.gov<mailto:john.kelsey@nist.gov>>
Date: Thursday, June 16, 2016 at 4:05 PM
To: internal-crypto <internal-crypto@nist.gov<mailto:internal-crypto@nist.gov>>
Subject: A couple easy steps toward moving our stuff for post-quantum crypto

Everyone,

I've been watching presentations about PQC today, and thinking about the transition time for post-quantum crypto. Thinking about this, it strikes me that we can do a few things to make the transition easier that don't require any new algorithms or anything.

I think in our current and future standards, we want to do a couple things:

a. Say that applications SHOULD use the larger security strength symmetric algorithms wherever possible, instead of simply the minimal 112-bit or 128-bit security levels.

b. Ensure that all the protocols and algorithms that we approve in the future at least can support 256-bit security level symmetric algorithms.

c. Wherever possible, ensure that protocols and such that we approve in the future that use public key algorithms *can* be adapted to much bigger sizes of key and message, and any other weird behavior that some PQ algorithms need (like stateful signatures or non-negligible error probabilities).

Comments?

--John