From: Kerman, Sara J. (Fed)
To: Moody, Dustin (Fed)
Subject: RE: revise the PQC FAQ
Date: Thursday, August 10, 2017 3:55:33 PM

Yes, I can still post the new ones….and I'll need to go back through the changes and figure out a way to organize them. Maybe like a comment template table or something.

From: Moody, Dustin (Fed)
Sent: Thursday, August 10, 2017 3:40 PM
To: Kerman, Sara J. (Fed) <sara.kerman@nist.gov>
Subject: Re: revise the PQC FAQ

I agree with you that we don't need to show every change. I like the approach of just putting a copy of the old Q+A on the archive document, along with the date it was moved to the archive.

If you think of some other way - that's fine, but I don't think we need to do anything too complex.

Dustin

From: Moody, Dustin (Fed)
Sent: Thursday, August 10, 2017 3:27:14 PM
To: Kerman, Sara J. (Fed)
Subject: Re: revise the PQC FAQ

Sounds good. Thanks.

(I assume you'll still post the new Q+A's even though we haven't figured out the archive)

From: Kerman, Sara J. (Fed)
Sent: Thursday, August 10, 2017 3:20:46 PM
To: Moody, Dustin (Fed)
Subject: RE: revise the PQC FAQ

I'd like to find a cleaner way to do this. It seems like the FAQs are getting edited/replaced often. Maybe the solution is to completely replace old with new and on the archive site, just state what date it was moved to archive. The archive file is going to start getting very confusing. For instance, Q15 & 16 (now) have already been edited twice.

Maybe there is another way to show changes on the historical page-maybe the way I started isn't the best. Let's think on this for a little before making the changes.

http://csrc.nist.gov/groups/ST/post-quantum-crypto/archive/faq-historical.pdf

From: Moody, Dustin (Fed)
Sent: Thursday, August 10, 2017 3:11 PM

To: Kerman, Sara J. (Fed) <sara.kerman@nist.gov>
Subject: revise the PQC FAQ

Sara,

Some more changes for the FAQ. We want to replace Q3 and Q4 with what's below. As far as what is changed, Q3 is almost entirely new except for the paragraph talking about NTL. Since even that paragraph was revised, it seems simplest just to archive the old A3 and put in the new A3, and say that the new answer expands or clarifies the old answer. As for A4, we moved the 2nd paragraph location, edited both of the old paragraphs, and added some new paragraphs. Do you think we want to show exactly what was changed? If so - I can do that - let me know.

We also want to change the first paragraph of A16 to what is below. All we're doing is adding a library, so I don't think we need to put the old version in the archive? (or should we)?

Thanks,
Dustin

Q3: What is the rationale for the NIST decision to limit both the required reference implementation and the required optimized implementation to ANSI C source code? Are there any exceptions that allow for the use of other versions of C, C++, or assembly optimizations?

A3: NIST understands that real-world cryptographic algorithm implementations will necessarily contain platform-specific optimizations. The two required implementations in the submission package are primarily intended to facilitate future analysis and development throughout the evaluation period, and as such, we require that both be written in a cross-platform manner. Additionally, the two required implementations need not be distinct. If a submitter does not see value in a separate cross-platform optimized implementation, they may simply note in their submission that the reference implementation is also the cross-platform optimized implementation.

Regarding the ANSI C requirement, submitters should note that key requirements are that the submission code should be written in a cross-platform manner and that the submission must contain build scripts or instructions for version 6.4.0 of the GNU Compiler. In particular, mandatory implementations written in C99 and C11 are both perfectly fine, as long as any necessary compiler directives are included as part of the build script(s).

Additionally, implementations that use NTL (see Question and Answer 16 for details on the use of third-party open source libraries) are necessarily allowed to be written in C++, although to ease portability to a pure C implementation via swapping NTL for C-based libraries, we ask that the original and new code in this submission be as ANSI C-like as possible, only using C++ functionality where absolutely required in order to interact with NTL.

Submitters may not write their own new and original assembly (including inline assembly) code or compiler intrinsics for either the mandatory referenced implementation or the mandatory optimized implementation but may use third party open-source libraries that themselves rely on assembly optimizations, subject to the constraints described in Question and Answer 16.

During the course of the evaluation process, NIST will be looking at performance data for the best-available implementations a variety of platforms. As such, we strongly encourage submitters to include optimized versions for major platforms- particularly x64, and 32-bit and 64-bit ARM architectures. However, we have made such submissions optional so as not to discourage submissions from teams that may have very strong algorithmic candidates, but have little experience in the area of platform optimization. For further questions on platform-specific optimizations and the role they will play in NIST's evaluation process, see Question and Answer 4.

Q4: Will NIST consider platforms other than the "NIST PQC Reference Platform" when evaluating submissions?

A4: The reference platform was defined in order to provide a common and ubiquitous platform to verify the execution of the code provided in the submissions.

The reference platform should be treated as a single core machine, but if an algorithm can make particular use of multiple cores or vector instructions, submitters are encouraged to provide additional implementations for these platforms.

In our evaluation process, NIST plans to include performance metrics from a variety of platforms, including: 64-bit "desktop/server class," 32-bit "mobile class," microcontrollers (32-, 16-, and where possible, 8-bit), as well as hardware platforms (e.g., FPGA). Submitters are strongly encouraged to provide additional implementations for these platforms, but to avoid discouraging submissions from teams with strong candidate algorithms but little experience in the area of platform-specific optimizations, NIST is making them optional as part of the submission itself.

NIST expects that as the evaluation process moves beyond the first round, we will see the wider cryptographic community (in particular, those skilled in platform-specific optimizations) provide optimized implementations for most submissions for a wide variety of platforms, as was the case in the SHA-3 competition. NIST plans to use such third-party optimized implementations and third-party benchmarking tools such as eBaCS/ SUPERCOP and Open Quantum Safe as part of its evaluation process.

A16: In both the mandatory reference implementation and the mandatory optimized implementation, submissions may use NTL Version 10.5.0 (http://www.shoup.net/ntl/download.html), GMP Version 6.1.2 (https://gmplib.org), the Keccak code package (https://github.com/gvanas/KeccakCodePackage), and OpenSSL Version 1.10f (https://www.openssl.org/source). Submitters may assume that these libraries are installed on the reference platform and do not need to provide them along with their submissions.