From:	Moody, Dustin (Fed)
To:	Liu, Yi-Kai (Fed); Perlner, Ray A. (Fed); Jordan, Stephen P (Fed); Chen, Lily (Fed); (b) (6)
Cc:	Peralta, Rene C. (Fed); Bassham, Lawrence E. (Fed)
Subject:	RE: PQC call for papers v4
Date:	Thursday, April 7, 2016 9:35:43 AM
Attachments:	CFP v5.docx

Attached is the latest version we will use at our meeting today.

From: Liu, Yi-Kai (Fed)
Sent: Wednesday, April 06, 2016 8:18 PM
To: Perlner, Ray (Fed) <ray.perlner@nist.gov>; Jordan, Stephen P (Fed) <stephen.jordan@nist.gov>; Moody, Dustin (Fed) <dustin.moody@nist.gov>; Chen, Lily (Fed) <lily.chen@nist.gov>; Daniel Smith (b) (6)
Cc: Peralta, Rene (Fed) <rene.peralta@nist.gov>; Bassham, Lawrence E (Fed) <lawrence.bassham@nist.gov>
Subject: Re: PQC call for papers v4

Hi everyone,
I cleaned up section 2 -- see attached file. (Dustin: I was editing Ray's version from earlier in this email chain, and all of my changes were confined to section 2. If you have made any edits on your copy of the file, can you just take my section 2 and paste it into your file?)
I think the first half of the document is in decent shape, so tomorrow we can just focus on the second half.
Cheers,
--Yi-Kai

From: Perlner, Ray (Fed)
Sent: Thursday, March 31, 2016 10:51:04 AM
To: Jordan, Stephen P (Fed); Moody, Dustin (Fed); Liu, Yi-Kai (Fed); Chen, Lily (Fed); Daniel Smith
Cc: Peralta, Rene (Fed); Bassham, Lawrence E (Fed)
Subject: RE: PQC call for papers v4

Thanks for the comment, Stephen
I'm glad someone else is looking carefully at our proposed evaluation criteria. That said, I don't think we should be overly concerned with submitters doing incorrect or biased security analysis. The worst thing that would come of that is that they set their parameters incorrectly – something which I think is likely to be less fatal for the submissions in this process than it was in the SHA3 competition. If we like a submission but think the submitters set the parameters wrong, we should simply tell the submitters that we'd like them to tweak their parameters for the next round, and publicly state the same in the report. I'm also not convinced that counting elementary gates is any easier than the sort of analysis suggested by my text. Hopefully I am getting across the message that we would prefer an imprecise measurement of security in a realistic attack model to a precise measurement of security in an unrealistic attack model (which, by the way, is the opposite of the typical incentives when the primary goal is getting academic papers published, so I do think we need to be somewhat explicit to push the analysis in this direction.)
I think it's also important to emphasize that these security metrics are evaluation criteria, not instructions to the submitters, and so they primarily constrain how we analyze submissions. If we give a precise definition of security which does not include consideration of parallelism, relative cost of classical and quantum operations etc, then we have prevented ourselves from taking these

factors into account when we analyze submissions.
Cheers, Ray

From: Jordan, Stephen P (Fed)
Sent: Wednesday, March 30, 2016 10:24 PM
To: Perlner, Ray (Fed) <ray.perlner@nist.gov>; Moody, Dustin (Fed) <dustin.moody@nist.gov>; Liu, Yi-Kai (Fed) <yi-kai.liu@nist.gov>; Chen, Lily (Fed) <lily.chen@nist.gov>; Daniel Smith (b) (6)
Cc: Peralta, Rene (Fed) <rene.peralta@nist.gov>; Bassham, Lawrence E (Fed) <lawrence.bassham@nist.gov>
Subject: Re: PQC call for papers v4

I like the direction the security definition is heading, but my intuition is that we may wish to simplify it further. A danger is that different submitters may make incomparable security analyses. If we leave too much complexity people may make mistakes and if we leave wiggle room people will be likely to interpret things in a way that makes their own submission look more favorable, even if they are not doing it consciously. I'd be in favor of saying something totally simpleminded and mathematically well-defined like: "the best known quantum attack must use at least 2^80 elementary quantum gates" (where we replace 2^80 with a few different numbers for different security levels). If we worry that someone might discover a way to parallelize the quantum attacks I think it is better to compensate by replacing 2^80 with 2^90 (or something) rather than adding more complexity or malleability to the security definition. Furthermore, our assumptions about the relative cost of quantum vs classical operations can simply be baked into our choices of number bits of security for each rather than leaving this as an aspect of the security definition for the individual teams to decide for themselves.
Best regards,
Stephen

From: Perlner, Ray (Fed)
Sent: Wednesday, March 30, 2016 4:49 PM
To: Moody, Dustin (Fed); Liu, Yi-Kai (Fed); Chen, Lily (Fed); Jordan, Stephen P (Fed); Daniel Smith
Cc: Peralta, Rene (Fed); Bassham, Lawrence E (Fed)
Subject: RE: PQC call for papers v4

Here is my update. All changes are confined to section 4, except for one comment to section 3, pointing out that we cannot require submitted signature algorithms to take arbitrary-length messages, since SHA256 has a maximum input size.
I have offered two choices for section 4A.iv (a slightly modified version of what I wrote before and something more aligned with what I think Yi-Kai was looking for.) See which one you like better.
Thanks,
Ray

From: Moody, Dustin (Fed)
Sent: Wednesday, March 30, 2016 10:21 AM
To: Liu, Yi-Kai (Fed) <yi-kai.liu@nist.gov>; Chen, Lily (Fed) <lily.chen@nist.gov>; Perlner, Ray (Fed) <ray.perlner@nist.gov>; Jordan, Stephen P (Fed) <stephen.jordan@nist.gov>; Daniel Smith

(b) (6)

Cc: Peralta, Rene (Fed) <rene.peralta@nist.gov>; Bassham, Lawrence E (Fed) <lawrence.bassham@nist.gov>

Subject: Re: PQC call for papers v4

I've added my fixes. I've also made some other small revisions throughout the document, so if you haven't yet started, please use the attached version. If you have already started writing, maybe you can copy/paste your sections you've edited into this document. Thanks.

Dustin

From: Liu, Yi-Kai (Fed)

Sent: Tuesday, March 29, 2016 4:32 PM

To: Chen, Lily (Fed); Moody, Dustin (Fed); Perlner, Ray (Fed); Jordan, Stephen P (Fed); Daniel Smith

Cc: Peralta, Rene (Fed); Bassham, Lawrence E (Fed)

Subject: PQC call for papers v4

Hi everyone,

Here is an updated version of the call for papers, after our discussion this morning. I cleaned up my section. Could you all take turns revising your sections? If we can get this cleaned up by Friday afternoon, that would be great!

Thanks!

--Yi-Kai

Billing Code:

DEPARTMENT OF COMMERCE

National Institute of Standards and Technology

Docket No.:

Announcing Request for Proposals for Quantum-Resistant Cryptographic Algorithms

AGENCY: National Institute of Standards and Technology, Commerce.

ACTION: Notice and request for nominations for Quantum-Resistant Cryptographic Algorithms.

SUMMARY: This notice solicits nominations from any interested party for quantum-resistant cryptographic algorithms to be considered for new public key cryptographic standards that will be secure against quantum computation. It addresses the nomination requirements and the minimum acceptability requirements of a ''complete and proper'' algorithm submission. The evaluation criteria that will be used to appraise the submitted algorithms are also described.

Commented [SC1]: For the hash competition, we published an FRN just to discuss the evaluation criteria. When this was settled ten months later, we then issued an FRN to call for candidate nomination. I wonder if you want to do that as well.

DATES: Submission packages must be received by DATE. Further details are available in Section X.

ADDRESSES: Submission packages should be sent to: XXX, Information Technology Laboratory, Attention: Quantum-Resistant Cryptographic Algorithm Submissions, 100 Bureau Drive – Stop 8930, National Institute of Standards and Technology, Gaithersburg, MD 20899–8930.

FOR FURTHER INFORMATION CONTACT: For general information, send e-mail to XXX@nist.gov. For questions related to a specific submission package, contact XXX, National Institute of Standards and Technology, 100 Bureau Drive – Stop 8930, Gaithersburg, MD 20899–8930; telephone: 301–975–XXX or via fax at 301–975–8670, e-mail: XXX@nist.gov.

SUPPLEMENTARY INFORMATION: This notice contains the following sections:

Authority: This work is being initiated pursuant to NIST's responsibilities under the Federal Information Security Management Act (FISMA) of 2002, Public Law 107–347.

1. Background

In recent years, there has been a substantial amount of research on quantum computers – machines that exploit quantum mechanical phenomena to solve mathematical problems that are difficult or intractable for conventional computers. If large-scale quantum computers are ever built, they will compromise the security of many commonly used cryptographic algorithms.

In particular, quantum computers would completely break many public key cryptosystems, including RSA, DSA, and elliptic curve cryptosystems. These cryptosystems are used to implement digital signatures and key exchange, and they play a crucial role in ensuring the confidentiality and integrity of communications on the Internet and other networks.

Due to this concern, many researchers have begun to investigate post-quantum cryptography (also called quantum-resistant cryptography). The goal of this research is to develop cryptographic algorithms that would be secure against both quantum and classical computers. These algorithms could serve as replacements for our current public key cryptosystems, in the event that large-scale quantum computers become a reality.

At present, there are several candidate post-quantum cryptosystems which have been proposed, including lattice-based cryptosystems, code-based cryptosystems, multivariate cryptosystems, and hash-based signatures among others. However, further research is needed in order to gain more confidence in their security (particularly against quantum adversaries), and to improve their efficiency and performance.

NIST has decided that it is prudent to begin developing standards for post-quantum cryptography now. This is driven by two factors. First, there has been noticeable progress in the development of quantum computers, including theoretical techniques for quantum error correction and fault-tolerant quantum computation, and experimental demonstrations of physical qubits and entangling operations in architectures that have the potential to scale up to larger systems.

Second, it appears that a transition to post-quantum cryptography will not be painless, as there is unlikely to be a simple "drop-in" replacement for our current public key cryptographic algorithms. A significant effort will be required in order to develop, standardize, and deploy new post-quantum algorithms. In addition, this transition needs to take place well before any large-scale quantum computers are built, so that any information which is later compromised by quantum cryptanalysis is no longer sensitive when that compromise occurs. Therefore, it is desirable to plan for this transition early.

NIST is taking a number of steps with regard to standardizing post-quantum cryptography. First, as an interim solution, NIST allows the use of "hybrid modes," which combine a currently approved cryptographic algorithm with a post-quantum algorithm, in such a way that the combined system is at least as secure as the stronger of the two components. Such hybrid modes can be approved for use under existing NIST guidelines. In addition, NIST will work to ensure appropriate coordination with other standardization efforts (for instance, other efforts to standardize stateful hash-based signatures).

Most importantly, NIST is beginning a process to develop new post-quantum standards for public key encryption, key establishment, and digital signatures. In developing these standards, NIST has two main considerations. First, these cryptosystems should provide strong security against both classical and quantum computers (and combinations thereof). Second, these cryptosystems should be easy to deploy in existing applications and protocols, such as TLS, IPSec (IKE), and digital certificates. In particular, these cryptosystems will be used to replace existing NIST standards that are not secure against quantum computers, including FIPS 186 and SP 800-56 A/B.

Commented [MD2]: Can we somehow indicate we want public key encryption for key establishment? Maybe Lily's suggestion of using backslash: public key encryption/key establishment?

NIST will solicit proposals for post-quantum cryptosystems from the community, and it will solicit comments from the community as part of its evaluation process. NIST expects to perform multiple rounds of evaluation, over a period of 3-5 years. The goal of this process will be to select some number of acceptable candidate cryptosystems, which will then be developed into NIST standards.

NIST anticipates that the evaluation process for these post-quantum cryptosystems may be significantly more complex than the evaluation of the SHA-3 and AES candidates. One reason is that the requirements for public key encryption and digital signatures are more complicated. Another reason is that the current scientific understanding of the power of quantum computers is far from comprehensive. A final reason is that some of the candidate cryptosystems may have completely different design attributes and mathematical foundations, so that a direct comparison is simply impossible.

Due to these complexities, NIST believes that the post-quantum standards process should not be treated as a competition. Due to the uncertainties in the evaluation of the submissions, in some cases, it may not be possible to make a well-supported

judgement that one candidate is "better" than another. Rather, the goal of the process is to perform a thorough analysis of the submitted algorithms, in a manner which is open and transparent to the community. This will inform NIST's decision on the subsequent development of post-quantum standards.

2. Requirements for Algorithm Submission Packages

Commented [MD3]: I removed "candidate" in most places in the document. A few places I thought the usage was fine to leave in.

Commented [YKL4]: But actually it is not just one algorithm, it is a collection of algorithms, e.g., Keygen, Encrypt, Decrypt. Can we use "cryptosystem" or "scheme" instead?

Algorithm nomination packages must be received by XXX. Submission packages received before XXX will be reviewed for completeness by NIST; the submitters will be notified of any deficiencies by XXX, allowing time for deficient packages to be amended by the submission deadline. No amendments to packages will be permitted after the submission deadline.

Due to the specific requirements of the submission package such as Intellectual Property Statements / Agreements / Disclosures as specified in section 2D, e-mail submissions will not be accepted for these statements or for the initial submission package. However, e-mail submissions of amendments to the initial submission package will be allowed prior to the submission deadline.

''Complete and proper'' submission packages received in response to this notice will be posted at http:// www.nist.gov/ for inspection. To be considered as a ''complete'' submission, packages must contain the following (as described in detail below):

- Cover Sheet.
- Algorithm Specifications and Supporting Documentation.
- Optical Media.
- Intellectual Property Statements/ Agreements/Disclosures.
- General Submission Requirements.

Each of these items is discussed in detail below.

2.A Cover Sheet

A cover sheet shall contain the following information:

- Name of the submitted algorithm.
- Principal submitter's name, e-mail address, telephone, fax, organization, and postal address.
- Name(s) of auxiliary submitter(s).
- Name of the algorithm inventor(s)/ developer(s).
- Name of the owner, if any, of the algorithm (normally expected to be the same as the submitter).
- Signature of the submitter.
- (optional) Backup point of contact (with telephone, fax, postal address, e- mail address).

2.B Algorithm Specifications and Supporting Documentation

2.B.1 A complete written specification of the algorithms shall be included, consisting of all necessary mathematical operations, equations, tables, diagrams, and parameters that are needed to implement the algorithms. The document shall include design rationale and an explanation for all the important design decisions that are made. It should also include 1) a survey of known work on the cryptosystem; 2) any applicable security analysis; 3) a precise security claim against quantum computation; and 4) a performance analysis.

Commented [MD(5]: Yi-Kai wants to remove?

In addition, the submission should specify several parameter sets which allow the selection of a range of possible security/performance tradeoffs as well as the construction of weakened versions of the submitted algorithm for analysis. In particular, the submitter should provide an analysis of how the security and performance of the algorithm depend on these parameter sets. Specific parameter sets may permit NIST to select a different performance/security tradeoff than originally specified by the submitter, in light of discovered attacks or other analysis, and in light of the alternative algorithms that are available. NIST will consult with the submitter of the algorithm if it plans to select that algorithm for standardization, but with a different parameter set than originally specified by the submitter.

A complete submission will include any necessary padding mechanisms and usage of approved primitives in order to achieve security. If the submitted algorithm cannot be used as a drop-in replacement for the algorithms and schemes specified in FIPS or NIST Special Publications, the point(s) of failure must be clearly indicated (and a potential compatibility construct offered?).

2.B.2 A statement of the algorithm's estimated computational efficiency and memory requirements for the ''NIST PQC Reference Platform'' (specified in section 5.B). (Efficiency estimates for other platforms may be included at the submitters' discretion.) These estimates shall each include the following information, at a minimum:

a. Description of the platform used to generate the estimate, in sufficient detail so that the estimates could be verified in the public evaluation process (e.g., for software running on a PC, include information about the processor, clock speed, memory, operating system, etc.). For hardware estimates, a gate count (or estimated gate count) should be included.

b. Speed estimate and memory requirements for the algorithm on the platform specified in section 5.B. At a minimum, the number of milliseconds required to perform each required operation (e.g., generate public and private keys, encrypt one message of length XXX, decrypt the resulting ciphertext, sign one message of length XXX, verify the resulting signature), and the size of all inputs and outputs (e.g., keys, ciphertexts, signatures).

Commented [YKL6]: FIXME: More details? Are milliseconds the right way to measure this?

Commented [MD(7]: This doesn't seem to fit in?

2.B.3 In addition, each submission package is required to include Known Answer Test (KAT) values, which can be used to determine the correctness of an implementation of the submitted algorithm. The KATs are individual input tuples that produce single output

values, e.g., an input tuple of a key and plaintext resulting in an output of the corresponding ciphertext. If the algorithm is randomized, the KAT should specify a fixed value for the random bits used by the algorithm, in order to force the algorithm to produce a fixed output value. Separate KATs should be provided to exercise different aspects of the algorithm, e.g., key generation, encryption, decryption, sign, verify, etc.

The KATs shall be included as specified below. All of these KAT values shall be submitted electronically, in separate files, on a CD–ROM or DVD as described in section 2.C.4.

Each file shall be clearly labeled with header information listing:

1. Algorithm name,
2. Test name,
3. Description of the test, and
4. Other parameters

Followed by a set of tuples where all values within the tuple shall be clearly labeled (*e.g.*, Plaintext, PublicKey, RandomBits, Ciphertext, etc.).

Commented [BLE(8]: I could generate a sample file and we could link to it.

All applicable KATs shall be included that can be used to exercise various features of the algorithm. A set of KATs shall be included for each security strength specified in section 4.A. Required KATs include:

i. . If the execution of the algorithm produces intermediate results that are informative (e.g., for debugging an implementation of the algorithm), then the submitter shall include known answers for those intermediate values for the computation for each of the required security strengths. Examples of providing such intermediate values are available at: *http://csrc.nist.gov/groups/ST/toolkit/index.html*.

ii. If tables are used in the algorithm, then a set of KAT vectors shall be included to exercise every table entry.

Note: The submitter is encouraged to include any other KATs that exercise different features of the algorithm (e.g., for permutation tables, padding scheme, etc.). The purposes of these tests shall be clearly described in the file containing the test values.

2.B.4 A statement of the expected strength (i.e., work factor) of the algorithm shall be included, along with any supporting rationale This statement shall include a description of which of the algorithm and parameter settings, specified by the submitter, the submitter is confident meet or exceed each of the security targets specified in section 4.A.iv, for at least one of the security models specified in section 4.A.ii and section 4.A.iii. If the submitter believes these settings exceed the relevant security target, the submitter shall give an estimate of how much the settings exceed the security target. Additionally the statement shall discuss the additional attack scenarios specified in section 4.A.v.

2.B.5 An analysis of the algorithm with respect to known attacks, and their results shall be included.

To prevent the existence of possible ‘‘trap-doors’’ in an algorithm, the submitter shall explain the provenance of any constants or tables used in the algorithm, with justification of why these were not chosen to make some attack easier.

The submitter shall provide a list of references to any published materials describing or analyzing the security of the submitted algorithm. The submission of copies of these materials (accompanied by a waiver of copyright or permission from the copyright holder for public evaluation purposes) is encouraged.

2.B.6 A statement that lists and describes the advantages and limitations of the algorithm shall be included. Such advantages and limitations may address the ability to: Implement the algorithm in various environments, including—but not limited to: 8-bit processors (e.g., smartcards), voice applications, satellite applications, or other environments where low power, constrained memory, or limited real-estate are factors. To demonstrate the efficiency of a hardware implementation of the algorithm, the submitter may include a specification of the algorithm in a nonproprietary Hardware Description Language (HDL).

2.C Optical Media

All electronic data shall be provided on a single CD-ROM or DVD labeled with the submitter’s name, and the algorithm name.

2.C.1 Implementations

Commented [BLE(9]: We can discuss if we want an “Additional Implementations” for additional code optimized for other platforms. YES

Two implementations are required in the submission package: a reference implementation and an optimized implementation. The goal of the reference implementation is to promote understanding of how the submitted algorithm may be implemented. Since this implementation is intended for reference purposes, clarity in programming is more important than efficiency. The reference implementation should include appropriate comments and clearly map to the algorithm description included in section 2.B.1 . The optimized implementation targeting the Intel x64 processor (a 64-bit implementation) is intended to demonstrate the performance of the algorithm. Both implementations shall consist of source code written in ANSI C.

Both implementations shall be capable of fully demonstrating the operation of the candidate algorithm. This includes support for all core features of the algorithm, e.g., key generation, public key validation, digital signature generation, digital signature validation.

A separate document specifying a set of cryptographic service calls, namely a cryptographic API, for the ANSI C implementations, will be made available at

<web_page>. Both the reference implementation and the optimized implementation shall adhere to the provided API. Separate source code for implementing the KATs shall also be included and shall adhere to the provided API.

Commented [BLE(10]: Insert appropriate web page for the project.

The reference implementation shall be provided in a directory labeled: \Reference_Implementation.

The optimized implementation shall be provided in a directory labeled: \Optimized_Implementation.

2.C.2 Known Answer Tests

The files on the CD–ROM or DVD shall contain all of the test values required under section 2.B.2 of this announcement. That section includes descriptions of the required tests, as well as a list of the values that must be provided.

The required format for the test vectors will be specified by NIST at *http://www.nist.gov/XXXX*.

The test values shall be provided in a directory labeled: \KAT.

2.C.3 Supporting Documentation

To facilitate the electronic distribution of submissions to all interested parties, copies of all written materials must also be submitted in electronic form in PDF. Submitters are encouraged to use the thumbnail and bookmark features, to have a clickable table of contents (if applicable), and to include other links within the PDF as appropriate.

This electronic version of the supporting documentation shall be provided in a directory \Supporting_Documentation

2.C.4 General Requirements for Optical Media

For the portions of the submissions that may be provided electronically, the information shall be provided on a single CD-ROM or DVD using the ISO 9660 format. This disc shall have the following structure:

- \README
- \Reference_Implementation
- \Optimized_Implementation
- \KAT
- \Supporting_Documentation

The "README" file shall list all files that are included on this disc with a brief description of each.

All optical media presented to NIST must be free of viruses or other malicious code. The submitted media will be scanned for the presence of such code. If malicious code is found, NIST will notify the submitter and ask that a clean version of the optical media be re-submitted.

2.D Intellectual Property Statements/ Agreements/Disclosures

Each submitted algorithm must be available worldwide on a royalty free basis during the period of the quantum-resistant algorithm search. In order to ensure this and minimize any intellectual property issues, the following series of signed statements are required for a submission to be considered complete: 1) Statement by the Submitter, 2) Statement by Patent (and Patent Application) Owner(s) (if applicable), and 3) Statement by Reference/Optimized Implementations' Owner(s). Note that for the last two statements, separate statements must be completed if multiple individuals are involved.

2.D.1 Statement by the Submitter

I, _____ (print submitter's full name) _____ do hereby declare that, to the best of my knowledge, the practice of the algorithm, reference implementation, and optimized implementations that I have submitted, known as ____ (print name of algorithm)____, may be covered by the following U.S. and/or foreign patents: _____ (describe and enumerate or state "none" if appropriate)_____ .

I do hereby declare that I am aware of no patent applications that may cover the practice of my submitted algorithm, reference implementation or optimized implementations. **– OR –** *I do hereby declare that the following pending patent applications may cover the practice of my submitted algorithm, reference implementation or optimized implementations: _____ (describe and enumerate) _____.*

I do hereby understand that my submitted algorithm might not be selected for standardization by NIST. I further understand that I will not receive financial compensation from the U.S. Government for my submission. I certify that, to the best of my knowledge, I have fully disclosed all patents and patent applications relating to my algorithm. I also understand that the U.S. Government may, during the course of the lifetime of the standard or during the public review process, modify the algorithm's specifications (e.g., to protect against a newly discovered vulnerability).

I understand that NIST will announce any selected algorithm(s) and proceed to publish the draft standards for public comment. Should my submission be selected for standardization, I hereby agree not to place any restrictions on the use of the algorithm, intending it to be available on a worldwide, non-exclusive, royalty-free basis.

I do hereby agree to provide the statements required by Sections 2.D.2 and 2.D.3, below, for any patent or patent application identified to cover the practice of my algorithm, reference implementation or optimized implementations and the right to use such implementations for the purposes of the evaluation process.

I understand that, during the quantum resistant algorithm evaluation process, NIST may remove my algorithm from consideration for standardization. If my algorithm (or the derived algorithm) is removed from consideration for standardization or withdrawn from consideration by the submitter, I understand that all rights, including use rights of the reference and optimized implementations, revert back to the submitter (and other owner[s], as appropriate).

Signed:
Title:
Dated:
Place:

2.D.2 Statement by Patent (and Patent Application) Owner(s)

If there are any patents (or patent applications) identified by the submitter, including those held by the submitter, the following statement must be signed by each and every owner of the patent and patent applications above identified.

I, _____ (print full name) _____ , of _____(print full postal address)_____ , am the owner or authorized representative of the owner (print full name, if different than the signer) of the following patent(s) and or patent application(s): _____ (enumerate) _____ , and do hereby agree to grant to any interested party if the algorithm known as _____(print name of algorithm) _____ is selected for standardization, an irrevocable nonexclusive royalty-free license to practice the referenced algorithm, reference implementation or the optimized implementations. Furthermore, I agree to grant the same rights in any other patent application or patent granted to me or my company that may be necessary for the practice of the referenced algorithm, reference implementation, or the optimized implementations.

Signed:
Title:
Dated:
Place:

Note that the U.S. government may conduct research as may be appropriate to verify the availability of the submission on a royalty free basis worldwide.

2.D.3 Statement by Reference/Optimized Implementations' Owner(s)

The following must also be included:

I, _____ (print full name) _____ , am the owner of the submitted reference implementation and optimized implementations and hereby grant the U.S. Government and any interested party the right to use such implementations for the purposes of the quantum-resistant algorithm evaluation process, notwithstanding that the implementations may be copyrighted.

Signed:

Title:
Dated:
Place:

2.E General Submission Requirements

NIST welcomes both domestic and international submissions; however, in order to facilitate analysis and evaluation, it is required that the submission packages be in English. This requirement includes the cover sheet, algorithm specification and supporting documentation, source code, and intellectual property information. Any required information that is submitted in a language other than English shall render the submission package ''incomplete.'' Optional supporting materials (e.g., journal articles) in another language may be submitted.

Classified and/or proprietary submissions will not be accepted.

2.F Technical Contacts and Additional Information

For technical inquiries, send e-mail to XXX@nist.gov, or contact XXX, National Institute of Standards and Technology, 100 Bureau Drive—Stop XXX, Gaithersburg, MD 20899–XXX; telephone: 301–975–XXX or via fax at 301–975–8670, e-mail: XXX

3. Minimum Acceptability Requirements

Those packages that are deemed to be ''complete'' will be evaluated for the inclusion of a ''proper'' post-quantum public key algorithm. To be considered as a ''proper'' post-quantum public key algorithm submission (and continue further in the standardization process), submitted algorithms shall meet the following minimum acceptability requirements:

i. The algorithms shall be publicly disclosed and available worldwide without royalties or any intellectual property restrictions.
ii. The algorithms shall be implementable in a wide range of hardware and software platforms.
iii. The algorithms shall provide at least one of: public key encryption, digital signatures, or key exchange.
- Digital signatures: have to sign arbitrary-length messages
    - Key gen, sign, verify
- Public key encryption / key exchange: messages / keys of length at least 256 bits
    - Key gen, encrypt, decrypt, or whatever key exchange does

iv. Theoretical and empirical evidence shall be provided to justify security claims of meeting the target security levels.

A post-quantum public key algorithm submission package that is complete (as defined above) and whose algorithm meets the minimum acceptability requirements (as defined

Commented [PR(11]: Actually, this should probably be messages up to (somewhat less than) 2^64 bits. The maximum input size for SHA-256 is 2^64-1 bits, and we probably want to allow for the fact that, in Fiat-Shamir Constructions for example, the hash input needs to be strictly longer than the message.

Maybe we should just say up to 2^63 bits (or equivalently 2^60 bytes.)

Commented [YKL12]: FIXME – (Daniel)

immediately above) will be deemed to be a ''complete and proper'' submission. A submission that is deemed otherwise at the close of the submission period will receive no further consideration. Submissions that are ''complete and proper'' will be posted at XXX for public review.

4. Evaluation Criteria

NIST will form an internal selection panel composed of NIST employees to analyze the submitted algorithms; the evaluation process will be discussed in section 6. All of NIST's analysis results will be made publicly available.

Although NIST will be performing its own analyses of the submitted algorithms, NIST strongly encourages public evaluation and publication of the results. NIST will take into account its own analysis, as well as the public comments that are received in response to the posting of the "complete and proper" submissions, to make its decisions.

This is not a competition with NIST as judge. We see our role as managing a process of achieving community consensus in a transparent and timely manner. We do not expect to "pick a winner". Ideally, several algorithms will emerge as "good choices". We may pick more than one of these for standardization.

Commented [PR13]: May want to change this wording. I feel like we may not want to say this isn't a competition any more than we want to say it is.

Commented [CL(14]: We can specifically say what we expect.

4.A Security

The security provided by an algorithm is the most important factor in the evaluation. Algorithms will be judged on the following factors:

i. Applications of Public Key Cryptography

NIST intends to standardize quantum-resistant alternatives to its existing standards for digital signatures (FIPS 186) and key establishment (SP 800-56A, SP 800-56B). These standards are used in a wide variety of internet protocols, such as TLS, SSH, IPSec, and DNSsec. Algorithms will be evaluated by the security they provide in these applications, and in additional applications that may be brought up by NIST or the public during the evaluation process. Claimed applications will be evaluated for their practical importance if this evaluation is necessary for deciding which algorithms to standardize.

ii. Security Model for Encryption

One particularly important application of public key cryptography is key transport (i.e. public key encryption of a symmetric key.). NIST intends to standardize at least one algorithm which enables semantically secure encryption with respect to adaptive chosen ciphertext attack (This property is generally denoted IND-CCA2 security in academic literature.)

Submitted algorithms for encryption and key exchange will be evaluated based on how well they appear to provide this property, when used as specified by the submitter. For the purpose of estimating security levels, it may be assumed that the attacker has access to the decryptions of no more than 2^{64} chosen ciphertexts, which are taken to be classical bit strings rather than arbitrary quantum states, however attacks involving more ciphertexts may also be considered.

iii. Security Model for Digital Signatures

One particularly important application of public key cryptography is digital signatures. NIST intends to standardize at least one algorithm which enables existentially unforgeable digital signature with respect to adaptive chosen message attack (This property is generally denoted EUF-CMA security in academic literature.)

Submitted algorithms for digital signature will be evaluated based on how well they appear to provide this property, when used as specified by the submitter. For the purpose of estimating security levels, it may be assumed that the attacker has access to signatures for no more than 2^{64} chosen messages, which are taken to be classical bit strings rather than arbitrary quantum states, however attacks involving more messages may also be considered.

iv. Measuring Bits of Security against Quantum Cryptanalysis

Submitters are asked to provide parameter sets that meet or exceed each of five security targets:

1) 128 bits classical security / 64 bits quantum security
2) 128 bits classical security/ 80 bits quantum security
3) 192 bits classical security/ 96 bits quantum security
4) 192 bits classical security / 128 bits quantum security
5) 256 bits classical security/ 128 bits quantum security

In specifying these security targets, the intent is that parameter sets meeting security targets 1, 3, and 5 will remain secure as long as brute-force attacks against AES 128, AES 192, and AES 256, respectively, remain infeasible. Likewise, parameter sets meeting security targets 2 and 4 should remain secure, roughly as long as brute-force collision attacks against SHA 256/ SHA3-256 and SHA 384/SHA3-384, respectively, remain infeasible.

NIST recognizes that there is some uncertainty regarding the best way to measure the complexity of cryptanalytic attacks, especially those involving quantum computers.

One ambiguity present in such measurements is the unit of work used to measure the attacker's attack complexity (does 128-bits of security refer to an attack

requiring the same amount of computation as 2^{128} AES operations, or 2^{128} simple binary operations, like AND, XOR, and NOT?) To resolve this ambiguity NIST recommends simply defining AES128 to have 128 bits of classical security and 64 bits of quantum security, assuming that there are no attacks on AES which are significantly cheaper than brute force search.

A second ambiguity is the question of how to evaluate the complexity of parallel attacks. When performed serially, a quantum search for a 2*s* bit key, using Grover's algorithm, has the same complexity as a classical search for an *s* bit key. However, Grover's algorithm parallelizes significantly more poorly than classical search. As a result, in the realistic scenario where the attacker performs many operations in parallel, classical search for an *s*-bit key has a significantly lower complexity than quantum search for a 2*s*-bit key. NIST, therefore recommends that submitters claim *s* bits of quantum security, only if all quantum attacks on the cryptosystem remain more expensive than a quantum attack on a block cipher with a 2*s*-bit key, even when parallelism is taken into account. Ideally, the submitted parameter sets should meet or exceed the quantum security of a block cipher with a 2*s*-bit key for any degree of parallelism, but NIST recognizes that extremely serial or extremely parallel attacks (e.g. those that have a time depth or space complexity exceeding 2^{100}) may be of minimal practical importance.

It should also be noted that the above definition often has the effect of assigning less quantum security than classical security to an algorithm, even in the absence of a practical quantum speedup. For example, a quantum computer would offer little, if any, advantage to an attacker attempting to find collisions in a 256 bit hash function. Nonetheless, the above definition would still assign something like 80 rather than 128 bits of quantum security, simply based on the fact that classical parallel collision search uses parallel computation more efficiently than would be expected for a quantum algorithm of the same serial complexity.

Finally, there is a third area of ambiguity in assessing quantum security. Mathematically, classical attacks may be treated as a special case of quantum attacks. However, it is very likely that classical operations will remain significantly cheaper to implement than explicitly quantum operations, due to the need for error correction and special purpose hardware. The question then arises as to how much this discrepancy should be taken into account. NIST acknowledges that this is a difficult question, however, as the quantum security targets are meant as a safeguard against the "optimistic" scenario, where quantum computing is relatively cheap and ubiquitous, submitters should err towards a small discrepancy, when estimating quantum security.

Commented [PR(15]: If the rest of the team prefers, the commented text may replace the remainder of section 4A.iv. I have left the previous text (slightly modified) in place for comparison.

The NIST team's initial thoughts are as follows:

Commented [YKL16]: Discuss this in some other venue, not the CFP?

Dustin: If not here, maybe the PQC-forum?

Define this more abstractly: as secure as AES, don't give a formula?

The defining case for *s* bits of quantum security is taken to be a key search for a 2*s* bit key. The most cost effective way to do this using a quantum computer is

probably to divide the key space into p segments, each of which would be searched for the correct key using a parallel instance of Grover's algorithm. This would then suggest that s bits of quantum security should be defined as follows:

An algorithm has s bits of quantum security if (for ~~all (?) values~~ any realistic value of p), an attacker with quantum computational resources (i.e. memory and processing units) proportional to p requires time proportional to $2^s/(p^{1/2})$ to violate the algorithm's security model.

Constants of proportionality would be set so that AES 128 has 64 bits of quantum security. Ideally, the submitted parameter sets should meet or exceed the above definition for any value of p, but NIST recognizes that extremely serial or extremely parallel attacks (e.g. those that have a time depth or space complexity exceeding 2^{100}) may be of minimal practical importance.

It should also be noted that the above definition often has the effect of assigning less quantum security than classical security to an algorithm, even in the absence of a practical quantum speedup. For example, a quantum computer would offer little, if any, advantage to an attacker attempting to find collisions in a 256 bit hash function. Nonetheless, the above definition would still assign something like 80 rather than 128 bits of quantum security, simply based on the fact that classical parallel collision search uses parallel computation more efficiently than would be expected for a quantum algorithm of the same serial complexity.

Finally, there is an additional area of ambiguity in assessing quantum security. Mathematically, classical attacks may be treated as a special case of quantum attacks. However, it is very likely that classical operations will remain significantly cheaper to implement than explicitly quantum operations, due to the need for error correction and special purpose hardware. The question then arises as to how much this discrepancy should be taken into account. NIST acknowledges that this is a difficult question, however, as the quantum security targets are meant as a safeguard against the "optimistic" scenario, where quantum computing is relatively cheap and ubiquitous, submitters should err towards a small discrepancy, when estimating quantum security.

v. Additional Attack Scenarios

While the previously listed security definitions cover many of the attack scenarios which will be used in the evaluation of the submitted algorithms, there are several other properties which would be desirable:

One such property, is perfect forward secrecy. While this property can be obtained through the use of standard encryption and signature functionalities, the cost of doing so may be prohibitive in some cases. In particular, public key encryption algorithms with a slow key generation procedure, such as RSA, are typically

Commented [YKL17]: Subsets?

Commented [YKL18]: Space?

Commented [YKL19]: See quantum time-space tradeoffs, Zhandry's quantum query lower bounds?

considered unsuitable for perfect forward secrecy. This is a case where there is significant interaction between the cost, and the practical security, of an algorithm.

Another case where security and performance interact is resistance to side channel attack. Attacks which can be made resistant to side channel attack at minimal cost are more desirable than those whose performance is severely hampered by any attempt to resist side channel attacks.

A third desirable property is resistance to multi-key attacks. Ideally an attacker should not gain an advantage by attacking multiple keys at once, whether the attacker's goal is to compromise a single key pair, or to compromise a large number of keys.

A final desirable, although ill defined, property is resistance to misuse. Algorithms should ideally not fail catastrophically due to isolated coding errors, random number generator malfunctions, nonce reuse etc.

vi. Evaluations Relating to Attack Resistance

Algorithms will be evaluated against attacks or observations that may threaten existing or proposed applications, or demonstrate some fundamental flaw in the design.

Claimed attacks will be evaluated for their practicality and for their impact on applications. Attacks that violate the security of an existing FIPS or NIST Special Publication's use of public key cryptography will be given more weight than attacks that violate the security of other applications; and attacks on rare or obscure applications may be given relatively little weight.

Algorithms will be evaluated not only for their resistance against previously known attacks, but also for their resistance against attacks discovered during the evaluation process, and for their likelihood of resistance against future attacks.

vii. Other Consideration Factors

In addition to the evaluation factors mentioned above, the quality of the security arguments/proofs, the clarity of the documentation of the algorithm, the quality of the analysis on the algorithm performed by the submitters, the continuity of the algorithm's design with previously analyzed constructions, the simplicity of the algorithm, and the confidence of NIST and the cryptographic community in the algorithm's long-term security may all be considered.

4.B Cost

As the cost of a public key cryptosystem can be measured on many different dimensions, NIST will continually seek public input regarding which performance

metrics and which applications are most important. If there are important applications which require radically different performance tradeoffs, NIST may need to standardize more than one algorithm to meet these diverse needs.

i. Public Key, Ciphertext, and Signature Size

Algorithms will be evaluated based on the sizes of public keys, ciphertexts, and signatures that they produce. All of these may be important for bandwidth constrained applications or in internet protocols that have a limited packet size. The importance of public key size may vary depending on the application: If applications can cache public keys, or otherwise avoid transmitting them frequently, the size of the public key may be of lesser importance. In contrast, applications that seek to obtain perfect forward secrecy by transmitting a new public key at the beginning of every session are likely to benefit greatly from algorithms that use relatively small public keys.

ii. Computational Efficiency of Public and Private Key Operations

Algorithms will also be evaluated based on the computational efficiency of the public key (encryption and signature verification) and private key (decryption and signing) operations. The computational cost of these operations will be evaluated both in hardware and software. The computational cost of both public and private key operations is likely to be important for almost all operations, but some applications may be more sensitive to one or the other (e.g. signing or decryption operations may be done by a computationally constrained device like a smartcard, or alternatively, a server dealing with a high volume of traffic may need to spend a significant fraction of its computational resources verifying client signatures.)

iii. Computational Efficiency of Key Generation

Algorithms will also be evaluated based on the computational efficiency of their key generation operations, where applicable. As noted in section 4.c (v), the most common scenario where key generation time is important is when a public key encryption algorithm is used to provide perfect forward secrecy. Nonetheless, it is possible that key generation times may also be important for digital signature algorithms in some applications.

iv. Decryption Failures

Some public key encryption algorithms, even when correctly implemented, will occasionally produce ciphertexts that cannot be decrypted. For most applications it is important that such decryption failures be rare or absent. While applications can always obtain an acceptably low decryption failure rate by encrypting the same ciphertext multiple times, this type of solution has its own performance costs.

Commented [PR20]: Need to decide what constitutes an acceptably low decryption failure rate? 2^-64?, 2^-80? 2^-128? Let the submitter decide?

4.C Algorithm and Implementation Characteristics

i. Flexibility

Assuming good overall security and performance, algorithms with greater flexibility will meet the needs of more users than less flexible algorithms, and therefore, are preferable.

Some examples of "flexibility" may include (but are not limited to) the following:

a. The algorithm can be modified to provide additional functionalities that extend beyond the minimum requirements of public key encryption or digital signatures. (e.g. optimized or implicitly authenticated key exchange etc.)
b. It is straightforward to customize the algorithm's parameters to meet a range of security targets and performance goals.
c. The algorithm can be implemented securely and efficiently on a wide variety of platforms, including constrained environments, such as smart cards.
d. Implementations of the algorithm can be parallelized to achieve higher performance efficiency.

ii. Simplicity

The submitted algorithm will be judged according to its relative design simplicity.

5. Plans for the Evaluation Process

NIST plans to form an internal selection panel composed of NIST employees for the technical evaluations of the submitted algorithms. This panel will analyze the submitted algorithms, review public comments that are received in response to the posting of the ''complete and proper'' submissions, and all presentations, discussions and technical papers presented at the Candidate Conferences, as well as other pertinent papers and presentations made at other cryptographic research conferences and workshops. NIST will issue a report after each Candidate Conference, make (any) final selections and document the technical rationale for any such selections in a final report, similar to what NIST did for the selection of AES and SHA-3. The following is an overview of the envisioned submission review process.

Commented [MD(21]: Do we have a better name we can use? We removed candidate throughout the document.

5.A Overview

Following the close of the call for algorithm submission packages, NIST will review the received packages to determine which are ''complete and proper,'' as described in sections 2 and 3 of this notice. NIST will post all ''complete and proper'' submissions at http://XXX for public inspection. To help inform the public, a Candidate Conference will be held at the start of the public comment process to allow submitters to publicly explain and answer questions regarding their submissions.

The initial phase of evaluation will consist of approximately twelve to eighteen months of public review of the submitted algorithms. During this initial review period, NIST intends to evaluate the submitted algorithms as outlined in Section 5.B. NIST will review the public evaluations of the submitted algorithms' cryptographic strength and weaknesses, and will use these to narrow the candidate pool for more careful study and analysis. If an algorithm is not included in the narrowed pool, then it does not mean the algorithm is removed for consideration for standardization, unless expressly stated by NIST.

Commented [MD22]: Does this need more explanation? Or is it okay?

Because of limited resources, and also to avoid moving evaluation targets (i.e., modifying the submitted algorithms undergoing public review), NIST will NOT accept modifications to the submitted algorithms during this initial phase of evaluation.

For informational and planning purposes, near the end of the initial public evaluation process, NIST intends to hold another Candidate Conference. Its purpose will be to publicly discuss the submitted algorithms, and to provide NIST with information for narrowing the field of algorithms to be focused on.

NIST plans to narrow the field of algorithms for further study, based upon its own analysis, public comments, and all other available information. It is envisioned that this narrowing will be done primarily on security, efficiency, and intellectual property considerations. NIST will issue a report describing its findings.

Before the start of a second evaluation period, the submitters of the algorithms will have the option of providing updated optimized implementations for use during the next phase of evaluation. During the course of the initial evaluations, it is conceivable that some small deficiencies may be identified in even some of the most promising submissions. Therefore, for the second round of evaluations, small modifications to the submitted algorithms will be permitted for either security or efficiency purposes. Submitters may submit minor changes (no substantial redesigns), along with a supporting explanation/ justification that must be received by NIST prior to the beginning of the second evaluation period. (Submitters will be notified by NIST of the exact deadline.) NIST will determine whether or not the proposed modification would significantly affect the design of the algorithm, requiring a major re-evaluation; if such is the case, the modification will not be accepted. If modifications are submitted, new reference and optimized implementations and written descriptions shall also be provided by the announced deadline. This will allow a thorough public review of the modified algorithms during the entire course of the second evaluation phase.

Note: All proposed changes must be proposed by the submitter; no proposed changes (to the algorithm or implementations) will be accepted from a third party.

The second round of evaluation will consist of approximately twelve to eighteen months of public review, with a focus on a narrowed pool of candidate algorithms. During the public review, NIST will similarly evaluate these algorithms as outlined in the next

section. After the end of the public review period, NIST intends to hold another Candidate Conference. (The exact date is to be scheduled.)

Following the third Candidate Conference, NIST will prepare a summary report, which may select algorithm(s) for possible standardization, and/or may determine that another phase of evaluation is needed. This third evaluation process would be similarly structured as the previous two evaluation periods. Any selected algorithm(s) for standardization will be incorporated into draft standards, which will be made available for public comment.

When evaluating algorithms, NIST will make every effort to obtain public input and will encourage the review of the submitted algorithms by outside organizations; however, the final decision as to which (if any) algorithm(s) will be selected for standardization is the responsibility of NIST.

It should be noted that this schedule for the evaluation process is somewhat tentative, depending upon the type, quantity, and quality of the submissions. Specific conference dates and public comment periods will be announced at appropriate times in the future. NIST estimates some algorithms could be selected for standardization after three to five years. However, due to developments in the field, this could change.

5.B Technical Evaluation

NIST will invite public comments on all complete and proper submissions. The analysis done by NIST during the initial phase(s) of evaluation is intended, at a minimum, to be performed as follows:

i. *Correctness check:* The KAT values included with the submission will be used to test the correctness of the reference and optimized implementations, once they are compiled. (It is more likely that NIST will perform this check of the reference code—and possibly the optimized code as well—even before accepting the submission package as ‘‘complete and proper.’’)

ii. *Efficiency testing:* Using the submitted optimized implementations, NIST intends to perform various computational efficiency tests. This could include, for example, the time required for key generation, encryption, decryption, digital signing, signature verification, or key establishment, as well as the size of keys, ciphertext, and signatures.

iii. *Other testing:* Other features of the submitted algorithms may be examined by NIST.

Platform and Compilers

The above tests will initially be performed by NIST on the

NIST Reference Platform: Intel x64 running Windows or Linux and supporting the GCC compiler.

At a minimum, NIST intends to perform an efficiency analysis on the reference platform; however, NIST invites the public to conduct similar tests and compare results on additional platforms (e.g., 8-bit processors, Digital Signal Processors, dedicated CMOS, etc.). NIST may also perform efficiency testing using additional platforms.

NIST welcomes comments regarding the efficiency of the submitted algorithms when implemented in hardware. During the second evaluation period, NIST may specify some of the algorithms using a Hardware Description Language, to compare the estimated hardware efficiency of the submitted algorithms.

Note: If the submitter chooses to submit updated optimized implementations prior to the beginning of the second round of evaluation, then some of the tests performed may be performed again using the new optimized implementations. This will be done to obtain updated measurements.

Note**:** Any changes to the intended platform/compiler will be noted on *http://XXX*

5.C Initial Planning for the First Candidate Conference

An open public conference will be held shortly after the end of the submission period, at which the submitter of each complete and proper submission package will be invited to publicly discuss and explain their submitted algorithm. The documentation for these algorithms will be made available at the Conference. Details of the conference will be posted at XXX.

For conference and resource allocation planning purposes, it would be appreciated if those planning to submit algorithms could notify the individuals listed in the **FOR FURTHER INFORMATION CONTACT** section as soon as possible.

6. Miscellaneous

This section is intended to address some of the questions/comments raised in the review of the draft evaluation criteria.

- NIST intends to develop a validation program for algorithm conformance testing, with the goal of having testing available by the time the final standards are published.
- Quantum security models…
- Submissions of hybrid modes are not in the purview of the post-quantum standardization process and will be rejected without consideration. Hybrid modes can be approved for use under existing NIST guidelines.

Commented [MD(23]: Add details – like co-location with PQCrypto?

Commented [CL(24]: Are we going to address the comments here?

Dustin: Let's remove this section. Anything we want to keep should be included somewhere else.

Commented [MD25]: Move where?

Commented [d26]: What exactly do we want to say here? Move to Ray's section

Commented [YKL27]: Move to minimum acceptability requirements?

- The use of complicated primitives such as block ciphers within a submitted algorithm should be restricted to NIST approved primitives. New such constructions requiring independent analysis will not be considered.
- Submitters of sufficiently similar algorithms may be asked to merge submissions. The submission of similar algorithms with distinct parameters and/or analyses may delay the public evaluation process and may well raise public questions as to the submitters' levels of confidence in the submissions.

Commented [YKL28]: Move to section 2.B?

Commented [YKL29]: Move to section 2

Note: Exportability decisions regarding submissions and, eventually, products implementing any selected algorithm(s) will be made by the appropriate U.S. Government regulatory authorities. NIST is a non-regulatory agency of the U.S. Department of Commerce.

Commented [MD30]: Anyone know if we need to leave this in?

Appreciation

NIST extends its appreciation to all submitters and those providing public comments during the quantum resistant algorithm evaluation process.

Dated: xxx