(ii) evidence submitted in support of allegations; (iii) publicly available information to value factors under 19 CFR 351.408(c) or to measure the adequacy of remuneration under 19 CFR 351.511(a)(2); (iv) evidence placed on the record by Commerce; and (v) evidence other than factual information described in (i)–(iv). These regulations require any party, when submitting factual information, to specify under which subsection of 19 CFR 351.102(b)(21) the information is being submitted and, if the information is submitted to rebut, clarify, or correct factual information already on the record, to provide an explanation identifying the information already on the record that the factual information seeks to rebut, clarify, or correct. The regulations, at 19 CFR 351.301, also provide specific time limits for such factual submissions based on the type of factual information being submitted. Please review the *Final Rule,*[12] available at *https://www.govinfo.gov/content/pkg/FR-2013-07-17/pdf/2013-17045.pdf,* prior to submitting factual information in this segment. Note that Commerce has amended certain of its requirements pertaining to the service of documents in 19 CFR 351.303(f).[13]

Any party submitting factual information in an AD or CVD proceeding must certify to the accuracy and completeness of that information using the formats provided at the end of the *Final Rule.*[14] Commerce intends to reject factual submissions in any proceeding segments if the submitting party does not comply with applicable certification requirements.

Extension of Time Limits Regulation

Parties may request an extension of time limits before a time limit established under Part 351 expires, or as otherwise specified by Commerce.[15] In general, an extension request will be considered untimely if it is filed after the time limit established under Part 351 expires. For submissions which are due from multiple parties simultaneously, an extension request will be considered untimely if it is filed after 10:00 a.m. on the due date. Examples include, but are not limited to: (1) case and rebuttal briefs, filed pursuant to 19 CFR 351.309; (2) factual information to value factors under 19 CFR 351.408(c), or to measure the adequacy of remuneration under 19 CFR 351.511(a)(2), filed pursuant to 19 CFR 351.301(c)(3) and rebuttal, clarification and correction filed pursuant to 19 CFR 351.301(c)(3)(iv); (3) comments concerning the selection of a surrogate country and surrogate values and rebuttal; (4) comments concerning CBP data; and (5) Q&V questionnaires. Under certain circumstances, Commerce may elect to specify a different time limit by which extension requests will be considered untimely for submissions which are due from multiple parties simultaneously. In such a case, Commerce will inform parties in the letter or memorandum setting forth the deadline (including a specified time) by which extension requests must be filed to be considered timely. This policy also requires that an extension request must be made in a separate, standalone submission, and clarifies the circumstances under which Commerce will grant untimely-filed requests for the extension of time limits. Please review the *Final Rule,* available at *https://www.gpo.gov/fdsys/pkg/FR-2013-09-20/html/2013-22853.htm,* prior to submitting factual information in these segments.

These initiations and this notice are in accordance with section 751(a) of the Act (19 U.S.C. 1675(a)) and 19 CFR 351.221(c)(1)(i).

Dated: August 8, 2024.

Scot Fullerton,

Acting Deputy Assistant Secretary for Antidumping and Countervailing Duty Operations.

[FR Doc. 2024–18103 Filed 8–13–24; 8:45 am]

BILLING CODE 3510–DS–P

[12] *See Certification of Factual Information To Import Administration During Antidumping and Countervailing Duty Proceedings,* 78 FR 42678 (July 17, 2013) (*Final Rule*); *see also* the frequently asked questions regarding the *Final Rule,* available at *https://enforcement.trade.gov/tlei/notices/factual_info_final_rule_FAQ_07172013.pdf.*

[13] *See Administrative Protective Order, Service, and Other Procedures in Antidumping and Countervailing Duty Proceedings; Final Rule,* 88 FR 67069 (September 29, 2023).

[14] *See* section 782(b) of the Act; *see also Final Rule;* and the frequently asked questions regarding the *Final Rule,* available at *https://enforcement.trade.gov/tlei/notices/factual_info_final_rule_FAQ_07172013.pdf.*

[15] *See* 19 CFR 351.302.

DEPARTMENT OF COMMERCE

National Institute of Standards and Technology

[Docket No. 240719–0201]

RIN 0693–XC131

Announcing Issuance of Federal Information Processing Standards (FIPS) FIPS 203, Module-Lattice-Based Key-Encapsulation Mechanism Standard, FIPS 204, Module-Lattice-Based Digital Signature Standard, and FIPS 205, Stateless Hash-Based Digital Signature Standard

AGENCY: National Institute of Standards and Technology (NIST), Commerce.

ACTION: Notice.

SUMMARY: This notice announces the Secretary of Commerce's approval of three Federal Information Processing Standards (FIPS): FIPS 203, Module-Lattice-Based Key-Encapsulation Mechanism Standard; FIPS 204, Module-Lattice-Based Digital Signature Standard; and FIPS 205, Stateless Hash-Based Digital Signature Standard. These standards specify key establishment and digital signature schemes that are designed to resist future attacks by quantum computers, which threaten the security of current standards. The three algorithms specified in these standards are each derived from different submissions in the NIST post-quantum cryptography standardization project (see *https://csrc.nist.gov/pqc-standardization*).

DATES: FIPS 203, FIPS 204, and FIPS 205 are effective on August 14, 2024.

ADDRESSES: FIPS 203, FIPS 204, and FIPS 205 are available electronically on the NIST Computer Security Resource Center website at *https://csrc.nist.gov.* Comments that were received on the proposed changes are published electronically at *https://www.regulations.gov* and the NIST post-quantum cryptography standardization project website at *https://csrc.nist.gov/pqc-standardization.*

FOR FURTHER INFORMATION CONTACT: Dr. Dustin Moody, National Institute of Standards and Technology, 100 Bureau Drive, Mail Stop 8930, Gaithersburg, MD 20899–8930, email: *Dustin.Moody@nist.gov, phone: (301) 975–8136.*

SUPPLEMENTARY INFORMATION: Over the past several years, there has been steady progress toward building quantum computers. The security of many commonly used public-key cryptosystems would be at risk if large-scale quantum computers were ever realized. In particular, this would

include key-establishment schemes and digital signatures that are based on integer factorization and discrete logarithms (both over finite fields and elliptic curves). As a result, in 2017, the National Institute of Standards and Technology (NIST) initiated a public process to select quantum-resistant public-key cryptographic algorithms for standardization. These quantum-resistant algorithms would augment the public-key cryptographic algorithms already contained in FIPS 186–5, Digital Signature Standard (DSS), as well as in NIST Special Publication (SP) 800–56A Revision 3, *Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography,* and SP 800–56B Revision 2, *Recommendation for Pair-Wise Key Establishment Using Integer Factorization Cryptography.*

NIST issued a public call for submissions to the Post-Quantum Cryptography (PQC) Standardization Process in December 2016. Prior to the November 2017 deadline, a total of 82 candidate algorithms were submitted. Shortly thereafter, the 69 candidates that met both the submission requirements and the minimum acceptability criteria were accepted into the first round of the standardization process. Submission packages for the first-round candidates were posted online for public review and comment.

After a year-long review of the candidates, NIST selected 26 algorithms to move on to the second round of evaluation in January 2019. These algorithms were viewed as the most promising candidates for eventual standardization and were selected based on both internal analysis and public feedback. During the second round of evaluation, there was continued evaluation of the analyses by NIST and the broader cryptographic community. After consideration of these analyses and other public input received throughout the evaluation process, NIST selected seven finalists and eight alternates to move on to the third round of evaluation in July 2020.

The third round of evaluation began in July 2020 and continued for approximately 18 months. During the third round of evaluation, there was a more thorough analysis of the theoretical and empirical evidence used to justify the security of the 15 candidates (*i.e.,* seven finalists and eight alternates). There was also careful benchmarking of their performance using optimized implementations on a variety of software and hardware platforms. Similar to conferences held during the first two rounds of evaluation, NIST also held the (virtual) Third NIST PQC Standardization Conference in June 2021. NIST summarized its decisions in a report at the end of each round, publishing NISTIR 8240 for the first round, NISTIR 8309 for the second round, and NISTIR 8413 for the third round. These reports are available at *https://csrc.nist.gov/publications/ir.*

After three rounds of evaluation and analysis, NIST selected four algorithms it will standardize as a result of the PQC Standardization Process. The public-key encapsulation mechanism selected was CRYSTALS–KYBER, along with three digital signature schemes: CRYSTALS–Dilithium, FALCON, and SPHINCS+. It is intended that these algorithms will be capable of protecting sensitive U.S. Government information well into the foreseeable future, including after the advent of quantum computers.

FIPS 203 specifies a cryptographic scheme called Module-Lattice-Based Key-Encapsulation Mechanism, or ML–KEM, which is derived from the CRYSTALS–KYBER submission. A Key Encapsulation Mechanism (KEM) is a particular type of key establishment scheme which can be used to establish a shared secret key between two parties communicating over a public channel. Current NIST-approved key establishment schemes are specified in SP 800–56A, *Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm-Based Cryptography,* and in SP 800–56B, *Recommendation for Pair-Wise Key Establishment Schemes Using Integer Factorization Cryptography.*

FIPS 204 and 205 each specify digital signature schemes, which are used to detect unauthorized modifications to data and to authenticate the identity of the signatory. FIPS 204 specifies the Module-Lattice-Based Digital Signature Algorithm (ML–DSA), which is derived from the CRYSTALS-Dilithium submission. FIPS 205 specifies the Stateless Hash-Based Digital Signature Algorithm (SLH–DSA), which is derived from the SPHINCS+ submission. Current NIST-approved digital signature schemes are specified in FIPS 186–5, *Digital Signature Standard* and SP 800–208, *Recommendation for Stateful Hash-based Signature Schemes.* In the future, NIST intends to develop a FIPS specifying a digital signature algorithm derived from FALCON as an additional alternative to these standards.

NIST published a notice in the **Federal Register** (88 FR 57938) on August 24, 2023, requesting public comments on the drafts FIPS 203, FIPS 204, and FIPS 205. For FIPS 203, NIST received 43 sets of comments: three from U.S. federal agencies, one from a foreign government agency, five from private-sector organizations, and 34 from private academics and technologists. For FIPS 204, NIST received 37 sets of comments: two from U.S. federal agencies, one from a foreign government agency, five from private-sector organizations, and 29 from private academics and technologists. For FIPS 205, NIST received 23 sets of comments: two from U.S. federal agencies, two from a foreign government agency, four from private-sector organizations, and 15 from private academics and technologists. NIST addresses points made by multiple commenters as singular comments in the sections following.

The following is a summary and analysis of the comments received during the public comment period and NIST's responses to them, including the interests, concerns, recommendations, and issues considered in the development of FIPS 203, FIPS 204, and FIPS 205:

General Comments

Comment 1: Several commenters expressed interest in both general and negative test vectors, with one explicit request for tests dealing with input validation.

Response: While test vectors will not be included in the three PQC FIPS, test vectors will be available on NIST's website.

Comment 2: Commenters noted that the usage of SHAKE in the draft FIPS does not match the interfaces available in FIPS 202, *SHA–3 Standard: Permutation-Based Hash and Extendable-Output Functions,* specifically interfaces where a variable amount of output is requested.

Response: NIST agrees with the comment noting the inconsistency with FIPS 202. NIST is revising Special Publication 800–185, *SHA–3 Derived Functions: cSHAKE, KMAC, TupleHash, and ParallelHash,* to include a new application programming interface (API) to describe how to use SHAKE to generate pseudorandom bits in a streaming fashion. The text in the FIPS has been revised to accommodate this usage in the revised final version.

Comment 3: One comment requested guidance for transitioning to PQC algorithms to be included in the FIPS.

Response: NIST will provide additional transition guidance on the migration to PQC following the publication of the FIPS. This guidance will be published in forthcoming NIST guidelines and incorporated into future revisions of relevant NIST publications. All NIST guidelines are available on the NIST Computer Security Resource

Center website at *https://csrc.nist.gov/publications.*

Comment 4: One comment requested guidance for handling side-channels to be included.

Response: Detailed implementation guidance, including side-channel attack countermeasures, are outside the scope of the algorithm specifications described in FIPS 203, FIPS 204, and FIPS 205.

Comment 5: Several commenters suggested moving the description of the security categories in the Appendix to a different document or revising the descriptions. Another commenter expressed concerns about using the Advanced Encryption Standard (AES) as the security benchmark.

Response: The description of the security categories was removed from the document and will be included in a future revision to Special Publication 800–57 Part 1, *Recommendation for Key Management.* NIST notes that the text used in the description of the security categories mentioned a ''block cipher with 128-bit key (*e.g.,* AES–128).'' Thus, a generic block cipher (and not specifically AES) is what was described.

Comments on FIPS 203

Comment 6: To improve the efficiency of side-channel-resistant implementations, two commenters suggested modifying the structure of the shared secret key used in the case of failed checks during the ML–KEM Decaps function. One comment suggested hashing the ciphertext before use to shorten it, and another suggested inputting the ciphertext before the secret value.

Response: While the suggestions may provide improvements in some cases, this version of Decaps has not been sufficiently evaluated by the community. Therefore, the current generation step was not modified.

Comment 7: Some commenters expressed concern about the input validation checks required for ML–KEM Encaps and Decaps. One commenter specifically requested removing the length check to support storing the key as a seed. Two commenters requested removing the modulus check on the value of public key. One commenter specifically mentioned the difficulty that some programming languages support returning exceptions.

Response: The input validation checks were included to provide an appropriate amount of assurance that the public key is of the correct type and format. Failure to obtain sufficient assurance can lead to security vulnerabilities. As a result, NIST made no changes based on these comments. NIST also notes that input validation should occur before Encaps and Decaps so there should be no need for returning exceptions.

Comment 8: A few commenters noted that the core algorithms within ML–KEM are difficult to test because they are non-deterministic.

Response: NIST reconfigured the ML–KEM functions to sample randomness before calling a more testing-friendly interface. The functions were split into a deterministic function that accepts randomness as input (to enable testing) and an externally callable function that generates the needed randomness.

Comment 9: Several commenters offered many suggestions to improve the readability of the document and correct small errors.

Response: NIST incorporated revisions throughout the FIPS to improve clarity and correctness. In particular, the code comments in ML–KEM pseudo-code were updated, along with adding footnotes to reference comments that were too lengthy to include. The mathematical symbols have also been reordered.

Comment 10: A few commenters requested further guidance on specific implementation details of ML–KEM (*e.g.,* handling bytes vs. bits, avoiding floating-point operations, and avoiding pass-by-reference).

Response: NIST added language elaborating on these topics. Specifically, NIST added explicit wording disallowing the usage of floating-point arithmetic, and added explicit input copying in cases where uncertainty with pass-by-references might cause confusion. In addition, the algorithms were revised to only use byte strings, with the exception of SHAKE. This is purely for syntactical conformance with FIPS 202, which specifies SHAKE. It is expected that most ML–KEM implementations will not implement conversions between bits and bytes when invoking hash functions or extendable output functions (XOFs).

Comment 11: A few commenters expressed security concerns over standardizing the parameter set ML–KEM–512 and requested its removal. Other commenters specifically disagreed with removing ML–KEM–512.

Response: NIST made no changes based on these comments. NIST is confident in the security of all the ML–KEM parameter sets and believes they will serve many different use cases and applications on a wide variety of computing platforms.

Comment 12: A few commenters expressed interest in guidance for constructing different ML–KEM parameter sets for different performance and security values.

Response: FIPS 203 specifies approved parameter sets to facilitate interoperability and validation of implementations.

Comment 13: Several commenters expressed interest in guidance on the secure usage of ML–KEM or KEMs in general. Some noted they expect such usage to be in a referenced SP 800–227 but noted that this document is currently unavailable. One comment requested guidance on applications of KEMs.

Response: NIST will provide guidance on the secure usage and applications of KEMs in the forthcoming SP 800–227, *Recommendations for Key Encapsulation Mechanisms.* A draft will be published for public comment on the NIST Computer Security Resource Center website in the second half of 2024.

Comment 14: A few commenters requested that the approved usage of ML–KEM shared secret keys be extended to the key derivation methods found in SP 800–56C, *Recommendation for Key-Derivation Methods in Key-Establishment Schemes.*

Response: NIST updated FIPS 203 to allow the key derivation methods in SP 800–56C to apply to keys generated as specified in FIPS 203 in place of shared secrets. When combining the key derivation with another key establishment or key exchange procedure, then the security of the combined procedure needs to be assessed on a case-by-case basis. More guidance will be provided in SP 800–227, *Recommendations for Key-Encapsulation Mechanisms,* a draft of which will be published for public comment on the NIST Computer Security Resource Center website in the second half of 2024.

Comment 15: One commenter recommended that NIST require the use of hybrid implementations, specifically using ML–KEM with another pre-quantum, standardized algorithm for security reasons.

Response: NIST is confident in the security of the standardized algorithms, which included a six-year public evaluation process and public review of the specifications in the draft FIPS publications. While NIST will not require that ML–KEM, or the other PQC algorithms, be used in hybrid schemes that incorporate a second standardized algorithm, it will ensure that its cryptographic standards support security protocols and applications that choose to implement hybrid approaches. Additional guidance relevant to hybrid approaches is being developed within NIST guidelines on key derivation methods and key

encapsulation mechanisms, as described in the response to Comment 14.

Comment 16: Several commenters noted that the indexing used for generating the *A* matrix in FIPS 203 swapped the indexing used in the CRYSTALS-Kyber specification.

Response 17: NIST changed the matrix indexing to match the CRYSTALS-Kyber specification.

Comment 18: Many commenters requested alternatives to and replacements for the XOFs and hash functions used inside ML–KEM, including Ascon and SHA–2. Several commenters requested no replacements be made for these functions.

Response: NIST made no changes based on these comments. Introducing alternative functions would create several new options which could hinder interoperability and adoption.

Comment 19: Several commenters requested NIST reintroduce a step in the ML–KEM Encaps function from the third-round specification of Kyber. This step hashed the randomness received from the system before its use in Encaps. They noted this step provided defense in depth and helped ensure security in the case of imperfect random bit generator (RBG) output. One commenter supported its removal.

Response: NIST did not reintroduce the hash of randomness into the Encaps. Hashing the RBG output within ML–KEM is an ineffective countermeasure against a faulty RBG. For example, other cryptographic algorithms or applications on a device could leak information from a faulty RBG regardless of countermeasures implemented within ML–KEM. This issue is addressed in FIPS 203 by requiring ML–KEM to be used only alongside an approved RBG as specified in SP 800–90A, *Recommendation for Random Number Generation using Deterministic Random Bit Generators.*

Comment 20: Some commenters requested NIST restore a step in the ML–KEM Encaps function from the previous version. This step updated the shared secret key with a hash of the ciphertext. Some motivation expressed include reverting a late-stage change and providing better security properties when ML–KEM is used in future applications.

Response: The indistinguishability under adaptive chosen ciphertext attack (IND–CCA2) security of ML–KEM does not require the ciphertext to be hashed during ML–KEM Encaps. NIST notes that this change was publicly proposed by the CRYSTALS-Kyber team in December 2022 and was followed by extensive public discussion. Due to no known security benefits for reverting to this step, NIST did not add back the hash of the ciphertext.

Comment 21: Many commenters expressed interest in allowance and guidance for storing a small seed string in place of the larger keys for ML–KEM, using the seed string to regenerate keys on demand during operation.

Response: NIST revised FIPS 203 to clarify that keys can be regenerated from saved seed values.

Comment 22: A few commenters raised issues related to the requirement for the Decaps function to implicitly reject on a protocol failure.

Response: Implicit rejection within ML–KEM has been maintained for consistency with the CRYSTALS–KYBER submission.

Comment 23: One commenter requested that the seed size be increased to 40 bytes for ML–KEM key generation to avoid multi-target attacks.

Response: NIST maintained the seed length as provided in the draft FIPS 203, noting that the seed of 32 bytes is sufficient to protect against multitarget attacks at security categories 1 and 3. Hardening category 5 would require further changes to ML–KEM beyond the extended seed. In addition, single target security should be sufficient for any plausible scenario (at category 5).

Comment 24: One comment noted outdated decapsulation failure rates for ML–KEM.

Response: NIST revised the decapsulation failure rates provided in FIPS 203 based on technical updates from the submitters of the CRYSTALS–KYBER algorithm. The rates changed by at most one order of magnitude for each parameter set.

Comment 25: One comment requested the addition of a table of precomputed values for ''zetas'' in number theory transform (NTT) computations, similar to ML–DSA.

Response: NIST added the relevant table of precomputed values as requested.

Comments on FIPS 204

Comment 26: Many comments about minor edits, including errors, inconsistencies, presentation, confusion, potential for confusion, and requests for clarification.

Response: NIST made several minor editorial changes to improve clarity where requested. In particular, ExpandMask and the call to SampleInBall in Sign have been slightly revised. Specifically, the input to SampleInBall in Sign is now the whole commitment hash value instead of just the first 256 bits. As for ExpandMask, the call to ''IntegerToBits'' is replaced with ''IntegerToBytes'', and the variable v is now always assigned bytes from the beginning of hash output. The Verify algorithm has also been simplified following a suggestion to remove the check for the Hamming weight of the hint vector.

Comment 27: One comment requested swapping the order of tr and M as inputs to a hash function in the Sign algorithm in order to simplify implementations of some digital signature APIs.

Response: After reviewing the digital signature APIs referenced by the commenter, NIST determined that the potential to simplify API implementations through the proposed change was not significant enough to justify a deviation from the CRYSTALS–Dilithium specification as evaluated in the third round and which would have required additional changes to implementations of the draft standard.

Comment 28: A few commenters mentioned the mixed usage of bit strings and byte strings and noted that this could cause confusion.

Response: The revised FIPS 204 uses byte strings as input and output to/from most of the algorithms, with the limited exceptions that follow. The input to and output from SHAKE are bit strings. This is consistent with FIPS 202, which specifies SHAKE. Input to BitsToInteger and output from IntegerToBits are also bit strings. It is expected that most implementations will not implement conversions between bits and bytes. Moreover, the revised FIPS 204 has defined the hash functions H and G to be byte-oriented versions of SHAKE256 and SHAKE128 respectively, so that bit strings are only used in the pseudocode where necessary to deal with an input message that is not a whole number of bytes.

Comment 29: Several commenters requested clarification on the ''pre-hash'' version of ML–DSA (*i.e.,* when the message that is signed by ML–DSA may be the digest of the content that is to be protected by the signature).

Response: NIST proposed a more fully specified pre-hash version on the PQC mailing list, and also held a panel on this topic at the fifth NIST PQC Standardization Conference. Using the feedback obtained, FIPS 204 contains a revised specification of pre-hashing. It also now specifies domain separation to ensure that pre-hashed messages can be distinguished from non-pre-hashed messages. FIPS 204 specifies that the signature identifier should indicate whether or not the message was pre-hashed.

Comment 30: One commenter requested the usage of SHAKE in ML–DSA be changed to the hash function

SHA2. Another commenter suggested allowing ASCON as an alternative to SHAKE.

Response: NIST made no changes based on these comments, as neither using ASCON nor SHA2 was studied during the three rounds of evaluation of the NIST standardization process.

Comment 31: A few commenters asked for clarification regarding storing the seed used to regenerate the public and private keys, as an alternative to storing the keys.

Response: FIPS 204 was revised to clarify that seeds may be stored and used to regenerate public and private keys.

Comment 32: One commenter proposed changing the length of the seed used for key generation to at least 40 bytes to protect against multi-target attacks.

Response: NIST did not modify the seed length, as the current 32-byte seed is sufficient to protect against multitarget attacks at security categories 2 and 3. The seed was at 32-bytes to maintain consistency with ML–KEM, where getting multi-target security at category 5 would require further changes beyond a longer seed. At category 5, single-target security should be sufficient for any plausible use case.

Comment 33: Some commenters asked for a randomized version of ML–DSA where the sampling of *y* is done using randomness directly from an RBG.

Response: FIPS 204 was not revised to include a randomized version. While a randomized version could enable more efficient side-channel protections, it would be difficult to validate implementations for conformance.

Comment 34: One commenter requested that a reduced round version of SHAKE be allowed for use in FIPS 204.

Response: A reduced round version of SHAKE (*e.g.,* TurboSHAKE) is not currently specified in a FIPS. While such a change would provide a performance improvement, this was not evaluated during the three rounds of evaluation in the NIST PQC standardization process.

Comments on FIPS 205

Comment 35: A few commenters suggested reducing the number of parameter sets that are specified in the standard.

Response: NIST made no changes based on these comments. While there was some support for reducing the number of parameter sets, others believed that all twelve parameter sets should remain. In addition, among those who recommended reducing the number of parameter sets, there was no consensus as to which parameter sets should be removed.

Comment 36: One commenter requested that additional parameter sets be specified that have smaller signature sizes, but for which the number of signatures that can safely be generated is less than 2^{64}.

Response: NIST intends to propose such parameter sets in a separate Special Publication.

Comment 37: Several commenters requested clarification on the ''pre-hash'' version of SLH–DSA (*i.e.,* when the message that is signed by SLH–DSA may be the digest of the content that is to be protected by the signature).

Response: NIST proposed a more fully specified pre-hash version on the PQC mailing list, and also held a panel on this topic at the fifth NIST PQC Standardization Conference. Using the feedback provided, FIPS 205 contains a revised specification of pre-hashing. It also now specifies domain separation to ensure that pre-hashed messages can be distinguished from non-pre-hashed messages. FIPS 205 specifies that the signature identifier should indicate whether or not the message was pre-hashed.

Comment 38: One commenter suggested replacing SHA–256 and SHA–512 with $\text{SHA–256}^{\text{SLH–DSA}}$ and $\text{SHA–512}^{\text{SLH–DSA}}$. The new functions would be the same as the current ones, with the exception that the padded message would be reordered so that all constant information (including padding) would be processed before the variable information. Another commenter suggested replacing SHAKE256 with TurboSHAKE256, which reduces the number of rounds in the permutation function from 24 to 12.

Response: No changes were made based on these comments. While the proposed changes might provide some performance improvement, the improvements were not considered significant enough to justify considering the use of cryptographic primitives that are not currently specified in NIST standards.

Comment 39: Several commenters requested clarifications to the text describing addresses.

Response: FIPS 205 was revised to include tables showing the member functions for manipulating addresses, supplementing the text and the figures illustrating each of the address types. Information about compressed addresses was also revised.

Summary of Changes to FIPS 203, FIPS 204, and FIPS 205

The following is a summary of the changes made to FIPS 203, FIPS 204, and FIPS 205.

In FIPS 203, FIPS 204, and FIPS 205, the main functions now each call an internal derandomized function in order to facilitate validation of implementations of these algorithms. In FIPS 203, this applies to KeyGen, Encaps and Decaps. In FIPS 204 and FIPS 205, this applies to KeyGen, Sign, and Verify. In addition, to offer misuse resistance against the possibility that keys for different parameter sets might be expanded from the same seed, domain separation was added to the key generation routine.

In FIPS 203 and FIPS 204, a new API is now used to invoke functions from the SHAKE family. This API allows for appropriately streaming pseudorandom bytes from a SHAKE XOF in situations where no a-priori bound is known on the total number of needed bytes. This API will also be described and used in the next revision of NIST SP 800–185, *SHA–3 Derived Functions.*

Language was added in FIPS 203 and FIPS 204 to clarify that some intermediate values can be stored in order to speed up certain computations. Some of these stored values can be computed from the public key, and thus do not require any special safeguards, while others require the same protections as the private key.

In FIPS 203 and FIPS 204, it is now permitted to terminate certain rejection sampling loops once a required minimum number of attempts is made.

The differences between CRYSTAL–KYBER and ML–KEM are now described in Appendix C of FIPS 203.

Based on comments submitted on draft FIPS 203, domain separation was added to the key generation routine to prevent the misuse of keys generated to target one security level from being used for a different security level when saving a key as a seed.

The draft of FIPS 203 had inadvertently changed the order in which the algorithms generated the entries of the matrix A in the public key of ML–KEM. This was changed back in the final specification of ML–KEM in FIPS 203 to match CRYSTALS–KYBER as specified in the third round of the PQC Standardization Process.

The differences between CRYSTAL-Dilithium and ML–DSA are now described in Appendix D of FIPS 204. Based on comments that were submitted on the draft version, in the final version of ML–DSA, as specified in FIPS 204, the malformed input check, which had

been omitted from draft FIPS 204, was restored to the hint unpacking algorithm. Additionally, rather than using just the first 256 bits of the commitment hash, ~c, as the input to SampleInBall, the full commitment hash is used. Also, ExpandMask is modified to take output bits from the beginning rather than at an offset.

Based on comments that were submitted on draft FIPS 204, more details were provided for the pre-hash version, HashML–DSA. These modifications include domain separation for the cases in which the message is signed directly and cases in which a digest of the message is signed. The changes were made by modifying the inputs to the internal signing and verification functions.

The differences between SPHINCS+ specification and SLH–DSA are described in Appendix A of FIPS 205. Based on comments that were submitted on draft FIPS 205, the SLH–DSA signature generation and verification functions were modified to include domain separation cases in which the message is signed directly and cases in which a digest of the message is signed. The changes were made by modifying the inputs to the signing and verification functions.

Authority: 40 U.S.C. 11331(f), 15 U.S.C. 278g–3.

Alicia Chambers,

NIST Executive Secretariat.

[FR Doc. 2024–17956 Filed 8–13–24; 8:45 am]

BILLING CODE 3510–13–P

DEPARTMENT OF COMMERCE

National Oceanic and Atmospheric Administration

[RTID 0648–XE180]

Takes of Marine Mammals Incidental to Specified Activities; Taking Marine Mammals Incidental To Ferndale Refinery Dock Maintenance and Pile Replacement Activities in Ferndale, Washington

AGENCY: National Marine Fisheries Service (NMFS), National Oceanic and Atmospheric Administration (NOAA), Commerce.

ACTION: Notice; issuance of an incidental harassment authorization.

SUMMARY: In accordance with the regulations implementing the Marine Mammal Protection Act (MMPA) as amended, notification is hereby given that NMFS has issued an incidental harassment authorization (IHA) to Phillips 66 Co. to incidentally harass marine mammals during construction activities associated with a dock replacement project in Ferndale, Washington.

DATES: This authorization is effective from August 1 through July 31, 2025.

ADDRESSES: Electronic copies of the application and supporting documents, as well as a list of the references cited in this document, may be obtained online at: *https://www.fisheries.noaa.gov/action/incidental-take-authorization-phillips-66-cos-ferndale-refinery-dock-maintenance-and-pile*. In case of problems accessing these documents, please call the contact listed below.

FOR FURTHER INFORMATION CONTACT: Jennifer Gatzke, Office of Protected Resources, NMFS, (301) 427–8401.

SUPPLEMENTARY INFORMATION:

Background

The MMPA prohibits the ''take'' of marine mammals, with certain exceptions. Sections 101(a)(5)(A) and (D) of the MMPA (16 U.S.C. 1361 *et seq.*) direct the Secretary of Commerce (as delegated to NMFS) to allow, upon request, the incidental, but not intentional, taking of small numbers of marine mammals by U.S. citizens who engage in a specified activity (other than commercial fishing) within a specified geographical region if certain findings are made and either regulations are proposed or, if the taking is limited to harassment, a notice of a proposed IHA is provided to the public for review.

Authorization for incidental takings shall be granted if NMFS finds that the taking will have a negligible impact on the species or stock(s) and will not have an unmitigable adverse impact on the availability of the species or stock(s) for taking for subsistence uses (where relevant). Further, NMFS must prescribe the permissible methods of taking and other ''means of effecting the least practicable adverse impact'' on the affected species or stocks and their habitat, paying particular attention to rookeries, mating grounds, and areas of similar significance, and on the availability of the species or stocks for taking for certain subsistence uses (referred to in shorthand as ''mitigation''); and requirements pertaining to the monitoring and reporting of the takings. The definitions of all applicable MMPA statutory terms cited above are included in the relevant sections below.

Summary of Request

On February 29, 2024 we received a request from Phillips 66 for an IHA to take marine mammals incidental to Ferndale Refinery Dock Maintenance and Pile Replacement Activities in Ferndale, Washington. Following NMFS' review of the application, Phillips 66 submitted revised versions on May 16 and May 20, 2024. The application was deemed adequate and complete on May 21, 2024. Phillips 66 has requested authorization of take by Level B harassment for harbor seal, California sea lion, Steller sea lion and harbor porpoise. Neither Phillips 66 nor NMFS expect serious injury or mortality to result from this activity and, therefore, an IHA is appropriate. There are no changes from the proposed authorization to the final authorization.

Description of the Specified Activity

Phillips 66 is planning to modernize the existing timber loading dock (on the southeastern shoreline of the Strait of Georgia in Ferndale, Washington) and replace it with a stronger structure that meets current industry best practices. The activity includes installation of steel piles by vibratory driving, and pile removal using an underwater chainsaw or cutting torch.

In-water pile installation construction will occur for 35 days, which will occur intermittently through approximately October 31, 2024. Take of marine mammals is anticipated to occur due to vibratory pile installation. Removal of all piles is expected to take up to 66 days for underwater pile cutting with a chainsaw. Take of marine mammals is not anticipated to occur due to pile removal.

This IHA is valid for a period of 1 year from the date of issuance. Due to in-water work timing restrictions to protect Endangered Species Act (ESA)-listed salmonids, all planned in-water construction in this area is limited to a work window beginning August 1 and ending February 1. However, since the Strait of Georgia is a very large water body with a long fetch, calm in-water work conditions are typically only available from August to the end of October. Pile removal processes are less dependent on good weather, and this portion of the project may occur from approximately August 1 to February 1. Therefore, Phillips 66 expects that in-water pile installation construction work will occur through October 31, 2024. Pile driving is anticipated to take up to 35 days to complete. Work may occur on nonconsecutive days due to weather and other project needs. Pile driving will be completed intermittently throughout daylight hours.

A detailed description of the planned dock maintenance and pile replacement project is provided in the **Federal Register** notice for the proposed IHA (89