

None
Internet-Draft
Intended status: Standards Track
Expires: 8 September 2022

S. Turner
sn3rd
P. Kampanakis
J. Massimo
AWS
B. Westerbaan
Cloudflare
7 March 2022

Algorithm Identifiers for NIST's PQC Algorithms for Use in the Internet
X.509 Public Key Infrastructure
[draft-turner-lamps-nist-pqc-kem-certificates-01](#)

Abstract

This document specifies algorithm identifiers and ASN.1 encoding format for the US NIST's PQC KEM (United States National Institute of Standards and Technology's Post Quantum Cryptography Key Encapsulation Mechanism) algorithms. The algorithms covered are Candidate TBD1. The encoding for public key and private key is also provided.

[EDNOTE: This draft is not expected to be finalized before the NIST PQC Project has standardized PQ algorithms. After NIST has standardized its first algorithms, this document will replace TBD, with the appropriate algorithms and parameters before proceeding to ratification. The algorithm Candidate TBD1 has been added as an example in this draft, to provide a more detailed illustration of the content - it by no means indicates its inclusion in the final version. This specification will use object identifiers for the new algorithms that are assigned by NIST, and will use placeholders until these are released.]

About This Document

This note is to be removed before publishing as an RFC.

Status information for this document may be found at
<https://datatracker.ietf.org/doc/draft-turner-lamps-nist-pqc-kem-certificates/>.

Discussion of this document takes place on the Limited Additional Mechanisms for PKIX and SMIME (lamps) Working Group mailing list (mailto:spasm@ietf.org), which is archived at
<https://mailarchive.ietf.org/arch/browse/spasm/>.

Internet-Draft

PQC KEM for Certificates

March 2022

Source for this draft and an issue tracker can be found at <https://github.com/seanturner/draft-turner-lamps-nist-pqc-kem-certificates>.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 8 September 2022.

Copyright Notice

Copyright (c) 2022 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the [Trust Legal Provisions](#) and are provided without warranty as described in the Revised BSD License.

Table of Contents

1.	Introduction	3
2.	Conventions and Definitions	3
3.	Algorithm Identifiers	3
4.	Candidate TBD1	4
5.	Subject Public Key Fields	4
6.	Key Usage Bits	5

7.	Private Key Format	5
8.	ASN.1 Module	6
9.	Security Considerations	6
10.	IANA Considerations	7
11.	References	7

11.1.	Normative References	7
11.2.	Informative References	7
	Acknowledgments	7
	Authors' Addresses	8

[1.](#) Introduction

The US NIST PQC Project has selected the Candidate TBD1 algorithms as winners of their PQC Project [[PQCProj](#)]. These algorithms are KEM algorithms. NIST has also defined object identifiers for these algorithms (TODO insert reference).

This document specifies the use of the Candidate TBD1 algorithms in X.509 public key certificates, see [[RFC5280](#)]. It also specifies private key encoding. An ASN.1 module is included for reference purposes.

These certificates could be used as Issuers in CMS where the public key is used to encapsulate a shared secret used to derive a symmetric key used to encrypt content in CMS [EDNOTE: Add reference [draft-perret-prat-lamps-cms-pq-kem](#)]. To be used in TLS, these certificates could only be used as end-entity identity certificates and would require significant updates to the protocol [EDNOTE: Add reference [draft-celi-wiggers-tls-authkem](#)].

[2.](#) Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [BCP 14](#) [[RFC2119](#)] [[RFC8174](#)] when, and only when, they appear in all capitals, as shown here.

[3.](#) Algorithm Identifiers

Certificates conforming to [[RFC5280](#)] can convey a public key for any

public key algorithm. The certificate indicates the algorithm through an algorithm identifier. An algorithm identifier consists of an object identifier and optional parameters.

The AlgorithmIdentifier type, which is included herein for convenience, is defined as follows:

```
AlgorithmIdentifier ::= SEQUENCE {  
    algorithm  OBJECT IDENTIFIER,  
    parameters ANY DEFINED BY algorithm OPTIONAL  
}
```

| NOTE: The above syntax is from [\[RFC5280\]](#) and matches the
| version used therein, i.e., the 1988 ASN.1 syntax. See
| [\[RFC5912\]](#) for ASN.1 compatible with the 2015 ASN.1 syntax.

The fields in AlgorithmIdentifier have the following meanings:

- * algorithm identifies the cryptographic algorithm with an object identifier. XXX such OIDs are defined in Sections [Section 4](#).
- * parameters, which are optional, are the associated parameters for the algorithm identifier in the algorithm field.

In this document, TODO (specify number) new OIDs for identifying the different algorithm and parameter pairs. For all of the object identifiers, the parameters MUST be absent.

It is possible to find systems that require the parameters to be present. This can be due to either a defect in the original 1997 syntax or a programming error where developers never got input where this was not true. The optimal solution is to fix these systems; where this is not possible, the problem needs to be restricted to that subsystem and not propagated to the Internet.

[4](#). Candidate TBD1

TODO insert object-identifiers

[5](#). Subject Public Key Fields

In the X.509 certificate, the subjectPublicKeyInfo field has the SubjectPublicKeyInfo type, which has the following ASN.1 syntax:

```
SubjectPublicKeyInfo ::= SEQUENCE {
    algorithm      AlgorithmIdentifier,
    subjectPublicKey BIT STRING
}
```

| NOTE: The above syntax is from [\[RFC5280\]](#) and matches the
| version used therein, i.e., the 1988 ASN.1 syntax. See
| [\[RFC5912\]](#) for ASN.1 compatible with the 2015 ASN.1 syntax.

The fields in SubjectPublicKeyInfo have the following meanings:

- * algorithm is the algorithm identifier and parameters for the public key (see above).

- * subjectPublicKey contains the byte stream of the public key. The algorithms defined in this document always encode the public key as TODO pick format e.g., exact multiple of 8 bits?.

The following is an example of a TBD public key encoded using the textual encoding defined in [\[RFC7468\]](#).

```
-----BEGIN PUBLIC KEY-----
TODO insert example public key
-----END PUBLIC KEY-----
```

[6.](#) Key Usage Bits

The intended application for the key is indicated in the keyUsage certificate extension; see [Section 4.2.1.3 of \[RFC5280\]](#).

If the keyUsage extension is present in a certificate that indicates Candidate TBD1 in SubjectPublicKeyInfo, then the following MUST be present:

keyEncipherment;

[7.](#) Private Key Format

"Asymmetric Key Packages" [[RFC5958](#)] describes how to encode a private key in a structure that both identifies what algorithm the private key is for and allows for the public key and additional attributes about the key to be included as well. For illustration, the ASN.1 structure `OneAsymmetricKey` is replicated below. The algorithm-specific details of how a private key is encoded are left for the document describing the algorithm itself.

```
OneAsymmetricKey ::= SEQUENCE {  
    version                Version,  
    privateKeyAlgorithm    PrivateKeyAlgorithmIdentifier,  
    privateKey             PrivateKey,  
    attributes             [0] IMPLICIT Attributes OPTIONAL,  
    ...,  
    [[2: publicKey        [1] IMPLICIT PublicKey OPTIONAL ]],  
    ...  
}
```

```
PrivateKey ::= OCTET STRING
```

```
PublicKey ::= BIT STRING
```

```
| NOTE: The above syntax is from [RFC5958] and matches the  
| version used therein, i.e., the 2002 ASN.1 syntax. The syntax  
| used therein is compatible with the 2015 ASN.1 syntax.
```

For the keys defined in this document, the private key is always an opaque byte sequence. The ASN.1 type `PqckemPrivateKey` is defined in this document to hold the byte sequence. Thus, when encoding a `OneAsymmetricKey` object, the private key is wrapped in a `PqckemPrivateKey` object and wrapped by the OCTET STRING of the "privateKey" field.

```
PqckemPrivateKey ::= OCTET STRING
```

The following is an example of a TBD private key encoded using the textual encoding defined in [[RFC7468](#)].

```
-----BEGIN PRIVATE KEY-----
TODO iser example private key
-----END PRIVATE KEY-----
```

The following example, in addition to encoding the TBD private key, has an attribute included as well as the public key. As with the prior example, the textual encoding defined in [[RFC7468](#)] is used.

```
-----BEGIN PRIVATE KEY-----
TODO insert example private key with attribute
-----END PRIVATE KEY-----
```

```
| NOTE: There exist some private key import functions that have
| not implemented the new ASN.1 structure OneAsymmetricKey that
| is defined in [RFC5958]. This means that they will not accept
| a private key structure that contains the public key field.
| This means a balancing act needs to be done between being able
| to do a consistency check on the key pair and widest ability to
| import the key.
```

[8.](#) ASN.1 Module

TODO ASN.1 Module

[9.](#) Security Considerations

The Security Considerations section of [[RFC5280](#)] applies to this specification as well.

[EDNOTE: Discuss side-channels for Candidate TBD1.]

[10.](#) IANA Considerations

This document will have some IANA actions.

[11.](#) References

[11.1.](#) Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC5280] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., and W. Polk, "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", [RFC 5280](#), DOI 10.17487/RFC5280, May 2008, <<https://www.rfc-editor.org/info/rfc5280>>.
- [RFC5912] Hoffman, P. and J. Schaad, "New ASN.1 Modules for the Public Key Infrastructure Using X.509 (PKIX)", [RFC 5912](#), DOI 10.17487/RFC5912, June 2010, <<https://www.rfc-editor.org/info/rfc5912>>.
- [RFC5958] Turner, S., "Asymmetric Key Packages", [RFC 5958](#), DOI 10.17487/RFC5958, August 2010, <<https://www.rfc-editor.org/info/rfc5958>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in [RFC 2119](#) Key Words", [BCP 14](#), [RFC 8174](#), DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.

11.2. Informative References

- [PQCProj] National Insititue of Standards and Technology, "Post-Quantum Cryptography Project", 20 December 2016, <<https://csrc.nist.gov/projects/post-quantum-cryptography>>.
- [RFC7468] Josefsson, S. and S. Leonard, "Textual Encodings of PKIX, PKCS, and CMS Structures", [RFC 7468](#), DOI 10.17487/RFC7468, April 2015, <<https://www.rfc-editor.org/info/rfc7468>>.

Acknowledgments

TODO acknowledge.

Sean Turner
sn3rd

Email: sean@sn3rd.com

Panos Kampanakis
AWS

Email: kpanos@amazon.com

Jake Massimo
AWS

Email: jakemas@amazon.com

Bas Westerbaan
Cloudflare

Email: bas@westerbaan.name