

Guy Fedorkow
Juniper Networks
gfedorkow@juniper.net
Kevin Micciche
HPE Aruba Networking
Kevin.micciche@hpe.com
Paul Bottorff
HPE Aruba Networking
Paul.bottorff@hpe.com

June 23, 2025

Proposed Post Quantum Cryptography Amendment to 802.1AR-2018

This document outlines a proposed amendment to IEEE 802.1AR-2018 to add support for Post Quantum Cryptography (PQC).

For the most part, this change is simply to add the Object Identifiers (OIDs) for soon-to-be-standardized Module-Lattice-Based Digital Signature (ML-DSA, FIPS-204) in the ITU-T X.509 certificate format specified by IETF (currently draft-ietf-lamps-dilithium-certificates).

This addition has a side-effect; DevID is normally used for signing, but can be used for encryption (although that was not widely agreed to be a Good Idea). RSA can be used for both, but PQC can't. Rather than define "both", it would be safer to drop the Encryption capability.

The basic proposal is simply to recommend *adding* ML-DSA / Dilithium (and not subtracting anything)

- But not adding SPHINCS+ / SLH-DSA
- And not adding LMS/XMSS

ML-DSA is defined in three security strengths; we would recommend all three security strengths – ML-DSA-44, ML-DSA-65, ML-DSA-87

It may be desirable to add an informational note indicating that government agencies intend to require ML-DSA-87, in spite of the large key sizes. [see [Informative doc](https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/evaluation-criteria/security-(evaluation-criteria))
[https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/evaluation-criteria/security-\(evaluation-criteria\)](https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/evaluation-criteria/security-(evaluation-criteria))]

802.1AR-2018 mentions support for RSA and ECC in passing in several paragraphs. ML-DSA should be added to these lists.

The technical details for protocol support are given in Section 9, so this note reproduces the existing Section 9.1 for RSA, with no proposed changes, followed by a proposed new Section 9.4 to cover ML-DSA.

The 2018 version of 802.1AR also references Trusted Computing Group documents for provisioning DevID in TPM1.2. These sections should be updated to cross-reference TPM2.0 as well. A footnote to the existing TPM1.2 cross reference might note that TPM1.2 is not defined to extend to ML-DSA.

Additional notes:

Should we consider mitigation against algorithm downgrade attack? – [see notes from TCG Amsterdam Jun 2025 on this topic]

Existing RSA section from 802.1AR-2018

9.1 RSA-2048/SHA-256

9.1.1 Algorithms and parameters

RSASSA-PKCS1-v1.5 signature schemes are defined in RFC 8017. The sha256WithRSAEncryption (RFC 4055) algorithm is used with a 2048-bit key and the SHA-256 secure hash algorithm as specified in NIST FIPS 180-4.

9.1.2 Key generation

An RNG used by a DevID module to generate keys for this signature suite shall have sufficient entropy to generate keys with a security strength of at least 128 bits.

9.1.3 signatureAlgorithm

The signatureAlgorithm field (8.8) value conforms to the general ASN.1 structure specified by RFC 5280 4.1.1.2 with the algorithm object identifier sha256WithRSAEncryption specified in RFC 4055 Section 5 and RFC 8017 A.2.4:

```
pkcs-1 OBJECT IDENTIFIER ::= { iso(1) member-body(2) us(840)
  rsadsi(113549) pkcs(1) 1 }
sha256WithRSAEncryption OBJECT IDENTIFIER ::= { pkcs-1 11 }
```

and a parameters field of type NULL, as specified in RFC 4055 Section 6 and RFC 8017 A.2.4.

9.1.4 subjectPublicKeyInfo

The subjectPublicKeyInfo field (8.7) conforms to the general ASN.1 structure specified by RFC 5280 4.1 with the algorithm object identifier rsaEncryption:

```
rsaEncryption OBJECT IDENTIFIER ::= { pkcs-1 1 }
```

and a parameters field of type NULL, as specified in RFC 3279 2.3.1 and RFC 8017 A.2.4.

The subjectPublicKey BIT STRING encapsulates the DER encoded RSAPublicKey, as specified in RFC 3279 2.3.1:

```
RSAPublicKey ::= SEQUENCE {
    modulus INTEGER, -- n
    publicExponent INTEGER }-e
```

9.1.5 signatureValue

The `signatureValue` field (8.9) encodes the result of applying the signing algorithm as a `BIT STRING`.

Base Documents

This chart summarizes the external documents referenced for RSA:

IETF RSA spec in 802.1AR	Name of Spec	Replacement for ML-DSA
RFC 3279	Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile	Are both of these replaced by Internet X.509 Public Key Infrastructure Algorithm Identifiers for the Module-Lattice-Based Digital Signature Algorithm (ML-DSA) https://datatracker.ietf.org/doc/draft-ietf-lamps-dilithium-certificates/
RFC 4055 (updates RFC 3279)	Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile	
RFC 8017	PKCS #1: RSA Cryptography Specifications Version 2.2	
RFC 5280	Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile	Only mentions RSA and 'DSA' as Examples, not as normative
NIST FIPS 180-4	SHA-256	No change; it's not relevant in the ML-DSA section.

Proposed new 802.1AR Section

9.4 ML-DSA

9.4.1 Algorithms and parameters

Module-Lattice Digital Signature Algorithm (ML-DSA) signature schemes are defined in draft-ietf-lamps-dilithium-certificates. The ML-DSA algorithm can be configured with three security strengths, ML-DSA-44, ML-DSA-65, ML-DSA-87, corresponding to security strengths of Categories 2, 3 and 5 respectively, as defined by NIST [https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/evaluation-criteria/security-\(evaluation-criteria\)](https://csrc.nist.gov/projects/post-quantum-cryptography/post-quantum-cryptography-standardization/evaluation-criteria/security-(evaluation-criteria))

9.4.2 Key generation

An RNG used by a DevID module to generate keys for this signature suite shall have sufficient entropy to generate keys with a security strength of at least 256 bits, as required by FIPS-204 [paragraph xx.x].

9.4.3 signatureAlgorithm

The signatureAlgorithm field (8.8) value conforms to the general ASN.1 structure specified by RFC 5280 4.1.1.2 with the algorithm object identifiers id-ml-dsa-44, id-ml-dsa-65, id-ml-dsa-87 specified in draft-ietf-lamps-dilithium-certificates:

```
id-ml-dsa-44 OBJECT IDENTIFIER ::= { joint-iso-itu-t(2)
    country(16) us(840) organization(1) gov(101) csor(3)
    nistAlgorithm(4) sigAlgs(3) id-ml-dsa-44(17) }
```

```
id-ml-dsa-65 OBJECT IDENTIFIER ::= { joint-iso-itu-t(2)
    country(16) us(840) organization(1) gov(101) csor(3)
    nistAlgorithm(4) sigAlgs(3) id-ml-dsa-65(18) }
```

```
id-ml-dsa-87 OBJECT IDENTIFIER ::= { joint-iso-itu-t(2)
    country(16) us(840) organization(1) gov(101) csor(3)
    nistAlgorithm(4) sigAlgs(3) id-ml-dsa-87(19) }
```

and a parameters field of type NULL, as specified in draft-ietf-lamps-dilithium-certificates .

9.4.4 subjectPublicKeyInfo

The subjectPublicKeyInfo field (8.7) conforms to the general ASN.1 structure specified by RFC 5280 4.1 with the algorithm object identifier id-ml-dsa-87:

```
id-ml-dsa-87 OBJECT IDENTIFIER ::= { sigAlgs 19 }
```

and a parameters field of type NULL, as specified in draft-ietf-lamps-dilithium-certificates.

The subjectPublicKey BIT STRING encapsulates the Raw Byte String ML-DSA-44-PublicKey, ML-DSA-44-PublicKey, ML-DSA-44-PublicKey, as specified in FIPS-204:

```
pk-ml-dsa-44 PUBLIC-KEY ::= {
    IDENTIFIER id-ml-dsa-44
    -- KEY no ASN.1 wrapping --
    CERT-KEY-USAGE
    { digitalSignature, nonRepudiation, keyCertSign, cRLSign }
    PRIVATE-KEY ML-DSA-44-PrivateKey } -- defined in Section 6
```

```
pk-ml-dsa-65 PUBLIC-KEY ::= {
    IDENTIFIER id-ml-dsa-65
    -- KEY no ASN.1 wrapping --
    CERT-KEY-USAGE
    { digitalSignature, nonRepudiation, keyCertSign, cRLSign }
    PRIVATE-KEY ML-DSA-65-PrivateKey } -- defined in Section 6
```

```
pk-ml-dsa-87 PUBLIC-KEY ::= {  
  IDENTIFIER id-ml-dsa-87  
  -- KEY no ASN.1 wrapping --  
  CERT-KEY-USAGE  
  { digitalSignature, nonRepudiation, keyCertSign, cRLSign }  
  PRIVATE-KEY ML-DSA-87-PrivateKey } -- defined in Section 6
```

```
ML-DSA-44-PublicKey ::= OCTET STRING (SIZE (1312))
```

```
ML-DSA-65-PublicKey ::= OCTET STRING (SIZE (1952))
```

```
ML-DSA-87-PublicKey ::= OCTET STRING (SIZE (2592))
```

9.4.5 signatureValue

The `signatureValue` field (8.9) encodes the result of applying the signing algorithm as a `BIT STRING`.