Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 10, 2016

S. Josefsson
SJD AB

April 8, 2016

EdDSA, Ed25519, Ed448, Curve25519 and Curve448 for X.509 draft-ietf-curdle-pkix-00

Abstract

This document specify algorithm identifiers and ASN.1 encoding formats for EdDSA digital signatures, subject public keys, and a "named curve" object identifier, used in the Internet X.509 Public Key Infrastructure. Parameters for Ed25519, Ed25519ph, Ed448, Ed448ph, Curve25519 and Curve448 are defined.

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 http://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 October 10, 2016.

Table of Contents

1. Introduction 2
2. Requirements Terminology 3
3. Curve25519 and Curve448 Named Curve Identifier 3
4. Subject Public Key Information Fields 3
5. EdDSA Public Keys 4
6. Key Usage Bits 5
7. EdDSA Signatures 5
8. Private Key Format 6
9. Human Readable Algorithm Names 7
10. ASN.1 Module 7
11. Examples 8
11.1. Example Ed25519ph Public Key 8
11.2. Example Ed25519ph Certificate 8
11.3. Example Ed25519ph Private Key 10
12. Acknowledgements 10
13. IANA Considerations 11
14. Security Considerations 11
15. References 11
15.1. Normative References 11
15.2. Informative References 12
Author's Address 12

1. Introduction

In [RFC7748], the elliptic curves Curve25519 and Curve448 are described. They are designed with performance and security in mind. The curves may be used for Diffie-Hellman and Digital Signature operations. In [I-D.irtf-cfrg-eddsa] the elliptic curve signature system EdDSA is described and the recommended choice of curves Ed25519/Ed448 are chosen. For each curve, two modes are defined, the PureEdDSA mode without pre-hashing (Ed25519 and Ed448), and the HashEdDSA mode with pre-hashing (Ed25519ph and Ed448ph).

This RFC defines ASN.1 object identifiers for EdDSA for use in the Internet X.509 PKI [RFC5280], and parameters for Ed25519, Ed25519ph, Ed448 and Ed448ph. This document serves a similar role as [RFC3279] does for RSA (and more), [RFC4055] for RSA-OAEP/PSS, and [RFC5758] for SHA2-based (EC)DSA. This document also specify ASN.1 "named curve" object identifiers for Curve25519 and Curve448, similar to what is done in [RFC5639]. This allows the curves to be used and referenced in PKIX standards and software, in particular enabling re-

use of existing constructs already defined for ECDSA/ECDH but for the new curves. Similar to [RFC5639], this document does not describe the cryptographic algorithms to be used with the specified parameters nor their application in other standards.

2. Requirements Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Curve25519 and Curve448 Named Curve Identifier

Certificates conforming to [RFC5280] may convey a public key for any public key algorithm. The certificate indicates the algorithm through an algorithm identifier. This algorithm identifier is an OID and optionally associated parameters. Section 2.3.5 of [RFC3279] describe ECDSA/ECDH public keys, specifying the id-ecPublicKey OID. This OID has the associated EcpkParameters parameters structure, which contains the namedCurve CHOICE. Here we introduce two new OIDs for use in the namedCurve field.

```
id-Curve25519   OBJECT IDENTIFIER ::= { 1.3.101.110 }
id-Curve448     OBJECT IDENTIFIER ::= { 1.3.101.111 }
id-Curve25519ph OBJECT IDENTIFIER ::= { 1.3.101.112 }
id-Curve448ph   OBJECT IDENTIFIER ::= { 1.3.101.113 }
```

The OID id-Curve25519 refers to Curve25519. The OID id-Curve448 refers to Curve448. Both curves are described in [RFC7748]. The OIDs id-Curve25519ph and id-Curve448ph refers to Curve25519 and Curve448 when used with pre-hashing as Ed25519ph and Ed448ph described in [I-D.irtf-cfrg-eddsa].

The public key value encoded into the ECPoint value is the raw binary values described in [RFC7748].

4. Subject Public Key Information 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
}
```

The fields in SubjectPublicKeyInfo have the following meanings:

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

- subjectPublicKey is the EdDSA public key.

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

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

The fields in AlgorithmIdentifier have the following meanings:

- algorithm identifies the cryptographic algorithm with an object identifier. This is the EdDSA OID defined below.

- parameters, which are optional, are the associated parameters for the algorithm identifier in the algorithm field.

5. EdDSA Public Keys

Certificates conforming to [RFC5280] may convey a public key for any public key algorithm. The certificate indicates the algorithm through an algorithm identifier. This algorithm identifier is an OID and optionally associated parameters.

This section identify the OID and parameters for the EdDSA algorithm. Conforming CAs MUST use the identified OIDs when issuing certificates containing EdDSA public keys. Conforming applications supporting EdDSA MUST, at a minimum, recognize the OID identified in this section.

The id-EdDSAPublicKey OID is used for identifying EdDSA public keys.

```
id-EdDSAPublicKey OBJECT IDENTIFIER ::= { 1 3 101 100 }
```

The id-EdDSAPublicKey OID is intended to be used in the algorithm field of a value of type AlgorithmIdentifier.

EdDSA public keys use the parameter field to specify the particular instantiation of EdDSA parameters. The parameters field have the ASN.1 type EdDSAParameters as follows.

```
EdDSAParameters ::= ENUMERATED { ed25519   (1),
                                 ed25519ph (2) }
                                 ed448     (3) }
                                 ed448ph   (4) }
```

The EdDSAParameters enumeration may be extended in the future.

The "ed25519" and "ed448" values correspond to the PureEdDSA variants, and the "ed25519ph" and "ed448ph" values correspond to the HashEdDSA variants, as discussed in [I-D.irtf-cfrg-eddsa].

The raw binary EdDSA public key is encoded directly in the subjectPublicKey BIT STRING object. Note that unlike some other schemes, there is no additional OCTET STRING encoding step.

6. Key Usage Bits

The intended application for the key MAY be indicated in the keyUsage certificate extension.

If the keyUsage extension is present in an end-entity certificate that conveys an EdDSA public key with the id-EdDSAPublicKey object identifier, then the keyUsage extension MUST contain one or both of the following values:

nonRepudiation; and
digitalSignature.

If the keyUsage extension is present in a certification authority

certificate that conveys an EdDSA public key with the id-EdDSAPublicKey object identifier, then the keyUsage extension MUST contain one or more of the following values:

nonRepudiation;
digitalSignature;
keyCertSign; and
cRLSign.

7. EdDSA Signatures

Certificates and CRLs conforming to [RFC5280] may be signed with any public key signature algorithm. The certificate or CRL indicates the algorithm through an algorithm identifier which appears in the signatureAlgorithm field within the Certificate or CertificateList. This algorithm identifier is an OID and has optionally associated parameters. For illustration the Certificate structure is reproduced here:

```
Certificate  ::=  SEQUENCE  {
     tbsCertificate       TBSCertificate,
     signatureAlgorithm   AlgorithmIdentifier,
     signatureValue       BIT STRING  }
```

Recall the definition of the AlgorithmIdentifier type:


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

This document identify an AlgorithmIdentifier OID for EdDSA signatures. No parameters are defined. The EdDSA parameters follow from the public-key parameters.

The data to be signed is prepared for EdDSA. Then, a private key operation is performed to generate the signature value. This value is the opaque value ENC(R) || ENC(S) described in section 3.3 of [I-D.irtf-cfrg-eddsa]. This signature value is then ASN.1 encoded as a BIT STRING and included in the Certificate or CertificateList in the signatureValue field.

The id-EdDSASignature OID is used for identifying EdDSA signatures.

```
id-EdDSASignature OBJECT IDENTIFIER ::= { 1 3 101 101 }
```

The id-EdDSASignature OID is intended to be used in the algorithm field of a value of type AlgorithmIdentifier. The parameters field MUST be absent. To further clarify how to encode the parameters field, due to historical misunderstandings in this area, it MUST NOT have an ASN.1 type NULL.

8. Private Key Format

In Elliptic Curve Private Key Structure [RFC5915] it is described how to encode elliptic curve private keys. Unfortunately, that format is specific to how traditional elliptic curve cryptography works so in order to re-use the format some details has to be modified for EdDSA, Curve25519 and Curve448. In particular, [RFC5915] assumes that EC private keys are unsigned integers, but for EdDSA, Curve25519 and Curve448 private keys are opaque binary byte sequences.

For illustration, the ASN.1 structure ECPrivateKey as defined by [RFC5915] is repeated here.

```
ECPrivateKey ::= SEQUENCE {
  version        INTEGER { ecPrivkeyVer1(1) } (ecPrivkeyVer1),
  privateKey     OCTET STRING,
  parameters [0] ECParameters {{ NamedCurve }} OPTIONAL,
  publicKey  [1] BIT STRING OPTIONAL
}
```

To encode a EdDSA, Curve25519 or Curve448 private key, the "privateKey" field will hold the raw binary private key rather than

any I2OSP-converted data as described by [RFC5915]. The ECParameters field "parameters" will hold one of the NamedCurve OIDs described earlier in this document. If present, the "publicKey" field will hold the raw public key. Note that these requirements normatively updates [RFC5915] on how these fields are interpreted, for the OIDs defined in this document.

9. Human Readable Algorithm Names

For the purpose of consistent cross-implementation naming this section establish human readable names for the algorithms specified in this document. Implementations SHOULD use these names when referring to the algorithms. If there is a strong reason to deviate from these names -- for example, if the implementation has a different naming convention and wants to maintain internal consistency -- it is encouraged to deviate as little as possible from the names given here.

Use the string "EdDSA" when referring to a public key or signature when the parameter set is not known or relevant.

When the EdDSAParameters value is known, use a more specific string. For the ed25519(1) value use the string "Ed25519". For the ed25519ph(2) value use the string "Ed25519ph". For ed448(3) use "Ed448". For ed448ph(4) use "Ed448ph".

10. ASN.1 Module

For reference purposes, the ASN.1 syntax is presented as an ASN.1 module here.


```
-- ASN.1 Module
```

```
safecurves-pkix-0 {1 3 101 120}

DEFINITIONS ::= BEGIN

id-EdDSAPublicKey OBJECT IDENTIFIER ::= { 1 3 101 100 }
id-EdDSASignature OBJECT IDENTIFIER ::= { 1 3 101 101 }

id-Curve25519   OBJECT IDENTIFIER ::= { 1.3.101.110 }
id-Curve448     OBJECT IDENTIFIER ::= { 1.3.101.111 }
id-Curve25519ph OBJECT IDENTIFIER ::= { 1.3.101.112 }
id-Curve448ph   OBJECT IDENTIFIER ::= { 1.3.101.113 }

EdDSAParameters ::= ENUMERATED { ed25519   (1),
                                 ed25519ph (2) }
                                 ed448     (3) }
                                 ed448ph   (4) }

END
```

11. Examples

This section contains illustrations of EdDSA public keys and certificates, illustrating parameter choices.

11.1. Example Ed25519ph Public Key

An example of a Ed25519ph public key:

```
     Public Key Information:
         Public Key Algorithm: EdDSA
         Algorithm Security Level: High
         Parameters: Ed25519ph

     Public Key Usage:

     Public Key ID: 9b1f5eeded043385e4f7bc623c5975b90bc8bb3b

     -----BEGIN PUBLIC KEY-----
     MC0wCAYDK2VkCgECAyEAGb9ECWmEzf6FQbrBZ9w7lshQhqowtrbLDFw4rXAxZuE=
     -----END PUBLIC KEY-----
```

11.2. Example Ed25519ph Certificate

An example of a PKIX certificate using Ed25519ph would be:

```
   X.509 Certificate Information:
```

```
        Version: 3
        Serial Number (hex): 5601474a2a8dc326
        Issuer: CN=Test Ed25519ph certificate
        Validity:
            Not Before: Tue Sep 22 12:19:24 UTC 2015
            Not After: Fri Dec 31 23:59:59 UTC 9999
        Subject: CN=Test Ed25519ph certificate
        Subject Public Key Algorithm: Ed25519ph
        Algorithm Security Level: High
        Extensions:
            Basic Constraints (critical):
                Certificate Authority (CA): FALSE
            Key Usage (critical):
                Digital signature.
            Subject Key Identifier (not critical):
                9b1f5eeded043385e4f7bc623c5975b90bc8bb3b
        Signature Algorithm: Ed25519ph
        Signature:
            be:9d:f8:b4:19:07:99:c9:04:12:21:e7:85:33:55:76
            b0:5f:29:70:77:bd:69:7a:a6:db:33:fe:c4:f5:3d:79
            d2:ba:77:6d:68:9b:a3:e9:53:bc:a6:56:54:3f:fa:f4
            1c:37:89:4e:c7:43:c0:3b:77:68:5d:98:f6:19:9d:05
    Other Information:
        SHA1 fingerprint:
            a3b75d83a56e127d0728ed8563233cadf943757e
        SHA256 fingerprint:
            cab1d7df29bdf82270d2192997c81f1b333dc37e670d7e88068fbe9dd747da3a
        Public Key ID:
            9b1f5eeded043385e4f7bc623c5975b90bc8bb3b
        Public key's random art:
            +---[Ed25519ph]---+
            |               .  |
            |              o ..|
            |               o.=|
            |           . .  +=|
            |        S  o .+oo|
            |         o  o.++o|
            |        o ...*.o.|
            |         o Eo.oo |
            |          ooo ..o|
            +-----------------+

    -----BEGIN CERTIFICATE-----
    MIIBUTCCAQKgAwIBAgIIVgFHSiqNwyYwBgYEK2VkATAqMSgwJgYDVQQDEx9UZXN0
    IEVkMjU1MTktU0hBNTEyIGNlcnRpZmljYXRlMCAXDTE1MDkyMjEyMTkyNFoYDzk5
    OTkxMjMxMjM1OTU5WjAqMSgwJgYDVQQDEx9UZXN0IEVkMjU1MTktU0hBNTEyIGNl
```

```
cnRpZmljYXRlMC0wCAYDK2VkCgECAyEAGb9ECWmEzf6FQbrBZ9w7lshQhqowtrbL
DFw4rXAxZuGjQDA+MAwGA1UdEwEB/wQCMAAwDwYDVR0PAQH/BAUDAweAADAdBgNV
```



```
HQ4EFgQUmx9e7e0EM4Xk97xiPFl1uQvIuzswBgYEK2VkAQNBAL6d+LQZB5nJBBIh
54UzVXawXylwd71peqbbM/7E9T150rp3bWibo+lTvKZWVD/69Bw3iU7HQ8A7d2hd
mPYZnQU=
-----END CERTIFICATE-----
```

11.3. Example Ed25519ph Private Key

An example of a Ed25519ph private key:

```
Public Key Info:
    Public Key Algorithm: EdDSA
    Key Security Level: High

parameters:    Ed25519ph
private key:
    d4:ee:72:db:f9:13:58:4a:d5:b6:d8:f1:f7:69:f8:ad
    3a:fe:7c:28:cb:f1:d4:fb:e0:97:a8:8f:44:75:58:42

x:
    19:bf:44:09:69:84:cd:fe:85:41:ba:c1:67:dc:3b:96
    c8:50:86:aa:30:b6:b6:cb:0c:5c:38:ad:70:31:66:e1


Public Key ID: 9B:1F:5E:ED:ED:04:33:85:E4:F7:BC:62:3C:59:75:B9:0B:C8:BB:3
Public key's random art:
+---[Ed25519ph]---+
|               .  |
|              o ..|
|               o.=|
|           . .  +=|
|        S  o .+oo|
|         o  o.++o|
|        o ...*.o.|
|         o Eo.oo |
|          ooo ..o|
+-----------------+

-----BEGIN EDDSA PRIVATE KEY-----
MCUKAQEEINTuctv5E1hK1bbY8fdp+K06/nwoy/HU++CXqI9EdVhC
```

```
-----END EdDSA PRIVATE KEY-----
```

12. Acknowledgements

Text and/or inspiration were drawn from [RFC5280], [RFC3279], [RFC4055], [RFC5480], and [RFC5639].

Several people suggested the utility of specifying NamedCurve OIDs for encoding Curve25519/Curve448 public keys into PKIX certificates. The editor of this document cannot take credit for this idea.

The following people discussed the document and provided feedback: Klaus Hartke, Ilari Liusvaara, Erwann Abalea, Rick Andrews, Rob Stradling, James Manger, Nikos Mavrogiannopoulos, Russ Housley, Jim Schaad.

A big thank you to Symantec for kindly donating the OIDs used in this draft.

13. IANA Considerations

None.

14. Security Considerations

The security considerations of [RFC5280], [RFC7748], and [I-D.irtf-cfrg-eddsa] apply accordingly.

A common misconception may be that a Ed25519 public key can be used to create Ed25519ph signatures, or vice versa. This leads to cross-key attacks, and is not permitted.

15. References

15.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[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, May 2008.

[RFC5480] Turner, S., Brown, D., Yiu, K., Housley, R., and T. Polk, "Elliptic Curve Cryptography Subject Public Key Information", RFC 5480, March 2009.

[RFC5915] Turner, S. and D. Brown, "Elliptic Curve Private Key Structure", RFC 5915, DOI 10.17487/RFC5915, June 2010, <http://www.rfc-editor.org/info/rfc5915>.

[RFC7748] Langley, A., Hamburg, M., and S. Turner, "Elliptic Curves for Security", RFC 7748, DOI 10.17487/RFC7748, January 2016, <http://www.rfc-editor.org/info/rfc7748>.

[I-D.irtf-cfrg-eddsa] Josefsson, S. and I. Liusvaara, "Edwards-curve Digital Signature Algorithm (EdDSA)", draft-irtf-cfrg-eddsa-00 (work in progress), October 2015.

15.2. Informative References

[RFC3279] Bassham, L., Polk, W., and R. Housley, "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile", RFC 3279, April 2002.

[RFC4055] Schaad, J., Kaliski, B., and R. Housley, "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 4055, June 2005.

[RFC5639] Lochter, M. and J. Merkle, "Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation", RFC 5639, March 2010.

[RFC5758] Dang, Q., Santesson, S., Moriarty, K., Brown, D., and T. Polk, "Internet X.509 Public Key Infrastructure: Additional Algorithms and Identifiers for DSA and ECDSA",

RFC 5758, January 2010.

Author's Address

Simon Josefsson
SJD AB

Email: simon@josefsson.org