EdDSA and Ed25519 for Transport Layer Security (TLS)
draft-josefsson-tls-eddsa-00

Abstract

This document introduces the public-key signature algorithm EdDSA for use in Transport Layer Security (TLS). With the previous NamedCurve and ECPointFormat assignments for the Curve25519 ECDHE key exchange mechanism, this enables use of Ed25519 in TLS. New Cipher Suites for EdDSA together with AES-GCM and ChaCha20-Poly1305 are introduced here. This is intended to work with any version of TLS and Datagram TLS.

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 November 21, 2015.

Copyright Notice

Copyright (c) 2015 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 (http://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 Simplified BSD License text as described in Section 4.e of
1. Introduction

TLS [RFC5246] and DTLS [RFC6347] support different key exchange algorithms and authentication mechanisms. In ECC in TLS [RFC4492], key exchange and authentication using ECC is specified, where the NamedCurve and ECPointFormat registries and associated TLS extensions are introduced. In [I-D.josefsson-tls-curve25519] support for ECDHE key exchange with the Curve25519 curve is added. That document introduce a new NamedCurve value for Curve25519, and a new ECPointFormat value to correspond to the public-key encoding. This document describe how to use EdDSA and Ed25519 [I-D.josefsson-eddsa-ed25519] as a new authentication mechanism in TLS, reusing the NamedCurve and ECPointFormat values already introduced for Curve25519, and finally specifying new Cipher Suites for Ed25519 with AES-GCM [RFC5288] and ChaCha20-Poly1305 [I-D.mavrogiannopoulos-chacha-tls].

1.1. 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].

2. The ECDHE_EDDSA Key Exchange Algorithm

Negotiation of the authentication mechanism is signalled by sending a SignatureAlgorithm value. Here we extend this structure for EdDSA.

```c
enum {
  eddsa(4)
} SignatureAlgorithm;
```

EdDSA is suitable for use with DTLS [RFC6347].

The new key exchange mechanism ECDHE_EDDSA provide forward secrecy. The key exchange mechanism work the same way ECDHE_ECDSA but with ECDSA replaced with EDDSA. Currently the only applicable curve is Curve25519.

3. Cipher Suites

The following Cipher Suite values are registered, using the ChaCha20/Poly1305 AEAD cipher described in [I-D.mavrogiannopoulos-chacha-tls] and the AES Galois Counter Mode (GCM) cipher. The AES-GCM cipher suites use authenticated encryption with additional data algorithms
AEAD_AES_128_GCM and AEAD_AES_256_GCM described in [RFC5116]. GCM is used as described in [RFC5288], but see also [RFC5289].

CipherSuite TLS_ECDHE_EDDSA_WITH_CHACHA20_POLY1305 = { 0xCC, 0xB0 }
CipherSuite TLS_ECDHE_EDDSA_WITH_AES_128_GCM_SHA256 = { 0xCC, 0xB1 }
CipherSuite TLS_ECDHE_EDDSA_WITH_AES_256_GCM_SHA256 = { 0xCC, 0xB2 }

The cipher suites are suitable for DTLS [RFC6347].

4. IANA Considerations

EdDSA should be registered in the Transport Layer Security (TLS) Parameters [IANA-TLS] registry under "SignatureAlgorithm" as follows.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
<th>DTLS-OK</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>eddsa</td>
<td>Y</td>
<td>This doc</td>
</tr>
</tbody>
</table>

The follow cipher suites should be registered in the TLS Parameters registry under "TLS Cipher Suite Registry" as follows. They should all be marked as DTLS-OK.

CipherSuite TLS_ECDHE_EDDSA_WITH_CHACHA20_POLY1305 = { 0xCC, 0xB0 }
CipherSuite TLS_ECDHE_EDDSA_WITH_AES_128_GCM_SHA256 = { 0xCC, 0xB1 }
CipherSuite TLS_ECDHE_EDDSA_WITH_AES_256_GCM_SHA256 = { 0xCC, 0xB2 }

5. Security Considerations


As with all cryptographic algorithms, the reader should stay informed about new research insights into the security of the algorithms involved.

While discussed in the EdDSA/Ed25519 specification and papers, we would like to stress the significant of secure implementation of EdDSA/Ed25519. For example, implementations ought to be constant-time to avoid certain attacks.
6. Acknowledgements

TBA.

7. References

7.1. Normative References


7.2. Informative References


Author’s Address

Simon Josefsson
SJD AB

Email: simon@josefsson.org