Secure Shell Maintenance
Internet-Draft
Intended status: Standards Track
Expires: 22 April 2026

S. Josefsson
19 October 2025

Hybrid Ed25519 with ML-DSA-65 for Secure Shell (SSH)
draft-josefsson-ssh-ed25519mldsa65-01

Abstract

This document describes the use of Ed25519 with ML-DSA-65 as a hybrid digital signature in the Secure Shell (SSH) protocol.

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-josefsson-ssh-ed25519mldsa65/.

Discussion of this document takes place on the SSHM Working Group mailing list (mailto:ssh@ietf.org), which is archived at https://mailarchive.ietf.org/arch/browse/ssh/.

Source for this draft and an issue tracker can be found at https://gitlab.com/jas/ietf-ssh-ed25519mldsa65.

Status of This Memo

Table of Contents

1. Introduction

Secure Shell (SSH) [RFC4251] is a secure remote-login protocol. It provides for an extensible variety of public key algorithms for identifying servers and users to one another.

Ed25519 [RFC8032] is a digital signature system.

CRYSTALS-Kyber is a post-quantum digital signature system, standardized in [NIST.FIPS.204] as Module-Lattice-Based Digital Signature Standard (ML-DSA).

This document specify how Ed25519 and ML-DSA-65 may be used in SSH, using the hybrid signature scheme suggested in [DJB-HYBRID-SIGNATURE].

2. Conventions Used In This Document

The descriptions of key and signature formats use the notation introduced in [RFC4251], Section 3, and the string data type from [RFC4251], Section 5. Identifiers and terminology from [RFC8032] and [NIST.FIPS.204] are used throughout the document.

3. Requirements Language

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.

4. Public Key Algorithm

This document describes a public key algorithm for use with SSH, as per [RFC4253], Section 6.6. The name of the algorithm is "ssh-ed25519-ml-dsa-65". This algorithm only supports signing and not encryption.

Standard implementations of SSH SHOULD implement this signature algorithm.

5. Public Key Format

The "ssh-ed25519-ml-dsa-65" key format has the following encoding:

string "ssh-ed25519-ml-dsa-65"

string key

The content of ’key’ is the concatenation of the Ed25519 32-octet public key described in [RFC8032], Section 5.1.5, with the 1952-octet public key described in [NIST.FIPS.204], for the ML-DSA-65 algorithm. The resulting key length is therefor 1984.

6. Signature Algorithm

Signatures are generated according to the following procedure, based on [DJB-HYBRID-SIGNATURE].

The signed message is (s2,s1,r,h,m) where

```
m = the message being signed,
r = H(fresh randomness chosen during signing),
h = H(r,H(hybridpk),hybridsigname,appname,appcontext,m),
s1 = Ed25519 signature of (r,h),
s2 = ML-DSA-65 signature of (s1,r,h),
H = SHA3-256.
```

The ’hybridpk’ value is the public key from the previous section. Here the fresh randomness MUST be 16 bytes, and only to be used for the signature. The ’hybridsigname’ field is "Ed25519MLDSA65", and ’appname’ is ’SSH’ with ’appcontext’ being ’SSH-Ed25519MLDSA65". Strings are encoded using ASCII [RFC0020].

The signed message (s2,s1,r,h,m) is the concatenation of each value. The ML-DSA-65 signature ’s2’ is 3309 octets, the Ed25519 signature ’s1’ is 64 octets, ’r’ is 16 octets, ’h’ is 32 octets, therefor the signature size is 3421 octets plus the message itself.

This protocol always uses the ’pure’ version of ML-DSA (where ML-DSA signs the message), and not the ’prehashed’ variant (where ML-DSA signs a previously hashed message). The ML-DSA ’context’ input MUST be the string "ML-DSA-65-Ed25519-SSH" encoded in ASCII [RFC0020]. ML-DSA may be used in deterministic or hedged mode.

7. Signature Format

The "ssh-ed25519-ml-dsa-65" key format has the following encoding:

```
string "ssh-ed25519-ml-dsa-65"

string signature
```

The ’signature’ value is the signed message produced in accordance with the previous section.

8. Verification Algorithm

Verification is done by invoking the verify functions for Ed25519 and ML-DSA-65 using the received values as follows, and taking the logical AND of their verification outputs.

```
Signed message is (s2,s1,r,h,m)
v1 = Ed25519 verification of s1 on message (r,h),
v2 = ML-DSA-65 verification of s2 on message (s1,r,h),
verify = v1 && v2
```

9. SSHFP DNS Resource Records

Usage and generation of the SSHFP DNS resource record is described in [RFC4255]. This section illustrates the generation of SSHFP resource records for Ed25519MLDSA65 keys, and this document also specifies the corresponding code point to "SSHFP RR Types for public key algorithms" in the "DNS SSHFP Resource Record Parameters" IANA registry [IANA-SSHFP].

The encoding of Ed25519MLDSA65 public keys is described in earlier sections.

The SSHFP Resource Record for the Ed25519MLDSA65 public key with SHA-256 fingerprint [NIST.FIPS.180] would, for example, be:

```
ssh.example.com IN SSHFP TBD1 2 ( a87f1b687ac0e57d2a081a2f2826723
                                  34d90ed316d2b818ca9580ea384d924
                                  01 )
```

Replace TBD1 with the value eventually allocated by IANA.

10. IANA Considerations

This document augments the Public Key Algorithm Names in [RFC4250], Section 4.11.3.

IANA is requested to add the following entry to "Public Key Algorithm Names" in the "Secure Shell (SSH) Protocol Parameters" registry [IANA-SSH]:

Public Key Algorithm Name	Reference
ssh-ed25519-ml-dsa-65	THIS-RFC

Table 1: SSH Public Key Code Points

IANA is requested to add the following entry to "SSHFP RR Types for public key algorithms" in the "DNS SSHFP Resource Record Parameters" registry [IANA-SSHFP]:

Value	Description	Reference
TBD1	SSH-ED25519-ML-DSA-65	THIS-RFC

Table 2: SSH DNS SSHFP RR Public Key Algorithm Types

11. Security Considerations

The security considerations in [RFC4251], Section 9 apply to all SSH implementations, including those using Ed25519MLDSA65.

The security considerations in [RFC8032] and [NIST.FIPS.204] apply to all uses of Ed25519 and ML-DSA-65, respectively, including those in SSH.

Verification of the hybrid signature may leak timing information that can be used to infer which of the Ed25519 or ML-DSA-65 verifications failed, if an implementation avoid to invoke one verification when the other one fails.

Ed25519MLDSA65 signatures are intended to be secure if SHA3-256 is secure and at least one of Ed25519 or ML-DSA-65 is secure.

Cryptographic algorithms and parameters are usually broken or weakened over time. Implementers and users need to continously re-evaluate that cryptographic algorithms continue to provide the expected level of security.

12. Acknowledgments

The text of [RFC8709] was used as a template for this document.

13. Test vectors

The following illustrate test vectors using file formats used by, for example, OpenSSH.

13.1. Private Key

Private key:

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAAAYwAAABtzc2gtc2xoLWRz
YS1zaGEyLTI1NmYAAABAPS6Ma/U7TKh4/I8HoTobiV+1sQnpkHZL7oztSTxgWoJYTDYEYdQpwty9
IJfwQDvK778DQCr9dx1gWb1HYDwYMAAAAQAS6f2dEun9nQAAABtzc2gtc2xoLWRzYS1zaGEyLTI1
NmYAAABAPS6Ma/U7TKh4/I8HoTobiV+1sQnpkHZL7oztSTxgWoJYTDYEYdQpwty9IJfwQDvK778D
QCr9dx1gWb1HYDwYMAAAAIBlB//OALih6/bAIOUGOGuaSKuK86IySusLX5xiqsPmJmE32DHKfIgg
mmvckaPbwnliYgL0mV/aAetfELu7XoqHPS6Ma/U7TKh4/I8HoTobiV+1sQnpkHZL7oztSTxgWoJY
TDYEYdQpwty9IJfwQDvK778DQCr9dx1gWb1HYDwYMAAAAAhqYXNAa2FrYQECAwQF
-----END OPENSSH PRIVATE KEY-----
```

13.2. Public-Key

Public key:

```
ssh-ed25519-ml-dsa-65 AAAAG3NzaC1zcGhpbmNzcGx1c0BvcGVuc3NoLmNvbQAAAEA9Loxr9TtMqHj8jwehOhu
JX7WxCemQdkvujO1JPGBaglhMNgRh1CnC3L0gl/BAO8rvvwNAKv13HWBZvUdgPBgw jas@kaka
```

13.3. Message

The namespace context string used is "my-namespace", and the message is (including final newline):

```
Hello world!
```

13.4. Signature

Signature:

```
-----BEGIN SSH SIGNATURE-----
U1NIU0lHAAAAAQAAAGMAAAAbc3NoLXNwaGluY3NwbHVzQG9wZW5zc2guY29tAAAAQD0ujG
v1O0yoePyPB6E6G4lftbEJ6ZB2S+6M7Uk8YFqCWEw2BGHUKcLcvSCX8EA7yu+/A0Aq/Xcd
YFm9R2A8GDAAAAAMbXktbmFtZXNwYWNlAAAAAAAAAAZzaGE1MTIAAHSDAAAAG3NzaC1zcG
hpbmNzcGx1c0BvcGVuc3NoLmNvbQAAdGCZtK1w9NaIGAV9HcHArlgyCGRb/a+f8/EDt1bL
BHVvMQiGVR4guZ1g20dasKIxJznf8YqoYQeSXEktX7ukD+Go+icRJoTQj7n0RaKjaWz/aM
P1iKeNN1hhfyOMP9nCzUKSBOlcBe1IDnHTMZDuX7wUVTu4WTcd4WrTb5Qos+fxY2cBUM9p
QeUPm2WpwkqVjpd8e4bG5ku2q4Q3jCHsambOH5VqZI+khzQ5w3M+b1wMXfWVwEd8O7t++U
-----END SSH SIGNATURE-----
```

14. References

14.1. Normative References

[NIST.FIPS.180]
NIST, "Secure hash standard", NIST Federal Information Processing Standards Publications 180, DOI 10.6028/NIST.FIPS.180, May 1993, <https://nvlpubs.nist.gov/nistpubs/Legacy/FIPS/NIST.FIPS.180.pdf>.

[NIST.FIPS.204]
"*** BROKEN REFERENCE ***".

[RFC0020] Cerf, V., "ASCII format for network interchange", STD 80, RFC 20, DOI 10.17487/RFC0020, October 1969, <https://www.rfc-editor.org/info/rfc20>.

[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>.

[RFC4250] Lehtinen, S. and C. Lonvick, Ed., "The Secure Shell (SSH) Protocol Assigned Numbers", RFC 4250, DOI 10.17487/RFC4250, January 2006, <https://www.rfc-editor.org/info/rfc4250>.

[RFC4251] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Protocol Architecture", RFC 4251, DOI 10.17487/RFC4251, January 2006, <https://www.rfc-editor.org/info/rfc4251>.

[RFC4253] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Transport Layer Protocol", RFC 4253, DOI 10.17487/RFC4253, January 2006, <https://www.rfc-editor.org/info/rfc4253>.

[RFC4255] Schlyter, J. and W. Griffin, "Using DNS to Securely Publish Secure Shell (SSH) Key Fingerprints", RFC 4255, DOI 10.17487/RFC4255, January 2006, <https://www.rfc-editor.org/info/rfc4255>.

[RFC8032] Josefsson, S. and I. Liusvaara, "Edwards-Curve Digital Signature Algorithm (EdDSA)", RFC 8032, DOI 10.17487/RFC8032, January 2017, <https://www.rfc-editor.org/info/rfc8032>.

[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>.

14.2. Informative References

[DJB-HYBRID-SIGNATURE]
Bernstein, D., "How to construct a hybrid signature combiner?", March 2024, <https://mailarchive.ietf.org/arch/msg/cfrg/LdvasJBpseekZtQkQF1nuPPDH_s/>.

[IANA-SSH] IANA, "Secure Shell (SSH) Protocol Parameters", n.d., <https://www.iana.org/assignments/ssh-parameters/>.

[IANA-SSHFP] IANA, "DNS SSHFP Resource Record Parameters", n.d., <https://www.iana.org/assignments/dns-sshfp-rr-parameters/>.

[RFC8709] Harris, B. and L. Velvindron, "Ed25519 and Ed448 Public Key Algorithms for the Secure Shell (SSH) Protocol", RFC 8709, DOI 10.17487/RFC8709, February 2020, <https://www.rfc-editor.org/info/rfc8709>.

Author's Address

Simon Josefsson
Email: simon@josefsson.org