sshm
Internet-Draft
Updates: 4253 (if approved)
Intended status: Standards Track
Expires: 18 October 2026

R. Petrov

16 April 2026

ML-DSA Public Key Algorithms for the Secure Shell (SSH) Protocol
draft-rpe-ssh-mldsa-03

Abstract

This document describes the use of the ML-DSA digital signature algorithms in the Secure Shell (SSH) protocol. Accordingly, this RFC updates RFC 4253.

Document and implementation details

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

The datatracker status page of the draft is draft-rpe-ssh-mldsa (https://datatracker.ietf.org/doc/draft-rpe-ssh-mldsa).

The source of this document is located at I-D ssh-mldsa (https://gitlab.com/secsh/pkixssh/-/blob/mldsa_demo/draft-rpe-ssh-mldsa.xml). Implementation could be found at PKIX-SSHMLDSA-DEMO (https://gitlab.com/secsh/pkixssh/-/tree/mldsa_demo) branch.

Discussion of this document takes place on the Secure Shell Maintenance (sshm)" (https://datatracker.ietf.org/group/sshm/about) mailing list (mailto:ssh@ietf.org) which is archived here (https://mailarchive.ietf.org/arch/browse/ssh/).

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 18 October 2026.

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.

This document describes the use of ML-DSA algorithms to be implemented by Secure Shells (SSH) and standardize the use of names *mldsa-44*, *mldsa-65*, and *mldsa-87*. These algorithms correspond to the Table 1. "ML-DSA parameter sets" defined in [FIPS-204] Section 4 "Parameter Sets".

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.

2.1. 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.

3. Public Key Algorithm

This document describes a public key algorithms for use with SSH, as per [RFC4253], Section 6.6. The name of the algorithms are *mldsa-44*, *mldsa-65*, and *mldsa-87*. These algorithms only supports signing and not encryption. Keys are generated according to the procedure described in [FIPS-204] Algorithm 1 step 5.

Standard implementations of SSH SHOULD implement *mldsa-65* public Key algorithm. It MAY implement *mldsa-44* and *mldsa-87* public key algorithms.

4. Public Key Format

* The *mldsa-44* key format has the following encoding:

  ```
  string  mldsa-44
  string  key
  ```

  Here, ’key’ is the 1312-octet public key encoded as is described in [FIPS-204] Algorithm 22.

* The *mldsa-65* key format has the following encoding:

  ```
  string  mldsa-65
  string  key
  ```

  Here, ’key’ is the 1952-octet public key encoded as is described in [FIPS-204] Algorithm 22.

* The *mldsa-87* key format has the following encoding:

  ```
  string  mldsa-87
  string  key
  ```

Here, 'key' is the 2592-octet public key encoded as is described in [FIPS-204] Algorithm 22.

5. Signature Algorithm

Signatures are generated according to the procedure described in [FIPS-204]. Signature generation should use normal signing process (Pure ML-DSA Signature Generation) with empty string as context parameter. The process is defined in [FIPS-204] Algorithm 2 step 10(sign) and Algorithm 3 step 5(verify).

6. Signature Format

* The *mldsa-44* public key algorithm has the following format for encoding the signature:

  ```
  string  mldsa-44
  string  signature
  ```

  Here, 'signature' is the 2420-octet signature produced in accordance with [FIPS-204] Algorithm 2.

* The *mldsa-65* public key algorithm has the following format for encoding the signature:

  ```
  string  mldsa-65
  string  signature
  ```

  Here, 'signature' is the 3309-octet signature produced in accordance with [FIPS-204] Algorithm 2.

* The *mldsa-87* public key algorithm has the following format for encoding the signature:

  ```
  string  mldsa-87
  string  signature
  ```

  Here, 'signature' is the 4627-octet signature produced in accordance with [FIPS-204] Algorithm 2.

7. Verification Algorithm

ML-DSA signatures are verified according to the procedure in [FIPS-204] Algorithm 3 step 5.

8. IANA Considerations

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

This document requests new entries to "Public Key Algorithm Names" in the "Secure Shell (SSH) Protocol Parameters" registry [IANA-SSH] according to the procedures in [RFC9519], Section 3:

Public Key Algorithm Name	Reference
mldsa-44	This document.
mldsa-65	This document.
mldsa-87	This document.

Table 1

9. Security Considerations

The security considerations in [RFC4251], Section 9 apply to all SSH implementations, including those using ML-DSA-44, ML-DSA-65, and ML-DSA-87. Also rules in [FIPS-204] Section 3.6 "Additional Requirements" apply as well.

10. References

10.1. Normative References

[FIPS-204] "Module-lattice-based digital signature standard", National Institute of Standards and Technology (U.S.), DOI 10.6028/nist.fips.204, August 2024, <https://csrc.nist.gov/pubs/fips/204/final>.

[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>.

[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>.

[RFC9519] Yee, P., "Update to the IANA SSH Protocol Parameters Registry Requirements", RFC 9519, DOI 10.17487/RFC9519, January 2024, <https://www.rfc-editor.org/info/rfc9519>.

10.2. Informative References

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

Acknowledgements

TBD

Author’s Address

Roumen Petrov
1750 Sofia
Bulgaria
Email: pkixssh@roumenpetrov.info
URI: https://roumenpetrov.info/secsh/