DNSOP Working Group
Internet-Draft
Intended status: Informational
Expires: 1 October 2026

A.M. Fregly
J. Harvey
B. Kaliski
D. Wessels
Verisign Labs

30 March 2026

Stateless Hash-Based Signatures in Merkle Tree Ladder Mode (SLH-DSA-MTL) for DNSSEC
draft-fregly-dnsop-slh-dsa-mtl-dnssec-06

Abstract

This document describes how to apply the Stateless Hash-Based Digital Signature Algorithm in Merkle Tree Ladder mode to the DNS Security Extensions. This combination is referred to as the SLH-DSA-MTL Signature scheme. This document describes how to specify SLH-DSA-MTL keys and signatures in DNSSEC. It uses both the SHA2 and SHAKE family of hash functions. This document also provides guidance for use of EDNS(0) in signature retrieval.

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 1 October 2026.

Copyright Notice

Table of Contents

1. Introduction

The Domain Name System Security Extensions (DNSSEC), which are broadly defined in [RFC4033], [RFC4034] and [RFC4035], use cryptographic keys and digital signatures to provide data origin authentication and data integrity in the DNS. Merkle Tree Ladder (MTL) Mode is a technique for using an underlying signature scheme to authenticate an evolving series of messages that is described in [I-D.harvey-cfrg-mtl-mode]. MTL Mode supports several underlying signature schemes such as the Stateless Hash-Based Digital Signature Algorithm (SLH-DSA). This document describes the application of MTL Mode to SLH-DSA as the SLH-DSA-MTL signature scheme for DNSSEC. Other combinations (such as ML-DSA) are possible and would be described in separate documents. SLH-DSA is described in the FIPS 205 standard [FIPS205]. As described herein, a DNSKEY resource record (RR) for an SLH-DSA-MTL key contains a SLH-DSA key. The SLH-DSA key is used for verifying signatures on Merkle tree ladders (MTLs). An RRSIG resource record for an SLH-DSA-MTL Signature contains a Merkle proof (authentication path) that is verifiable using a MTL, and optionally also contains the signed MTL.

The anticipation of quantum computers that can break the current signature algorithms led to NIST selecting post-quantum cryptographic (PQC) algorithms for standardization and developing specifications for the algorithms as NIST standards. These new algorithms are expected to replace classical digital signature algorithms (e.g., RSA and ECDSA) in IETF standards and to be widely implemented and deployed after that. NIST's proposed PQC algorithms have significantly larger signature sizes than RSA and ECDSA. The larger sizes may have a significant operational impact on DNSSEC. For example, the size of signed NSEC and NSEC3 responses may exceed UDP MTUs with this degrading the use of UDP as the prevalent DNSSEC transport. Larger signature sizes could also substantially increase memory requirements for in-memory zone databases used by authoritative name servers and for in-memory caches used by resolvers.

As described in [I-D.harvey-cfrg-mtl-mode], MTL mode is designed to reduce the size impact of PQC signature algorithms. For DNSSEC, the size impact reduction is achieved when signatures provided in RRSIG RRs are primarily comprised of "condensed signatures" (Merkle proofs / authentication paths) and are only occasionally comprised of "full signatures" that contain both a condensed signature and a signed MTL, where the signed ladder includes a signature using the underlying PQC signature algorithm. MTL mode reduces the memory requirements for PQC signatures as the signature data in the zone database or cache is primarily comprised of Merkle proofs and only occasionally of signed MTLs [CTRSAMTL].

SLH-DSA is a stateless hash-based PQC signature scheme selected by NIST for standardization [NISTSELECTIONS] in July 2022 and formally published as a standard in August 2024 [FIPS205]. This document specifies SLH-DSA for the initial application of MTL mode to DNSSEC based on three considerations: (1) SLH-DSA is also based on Merkle trees, and thus already has internal functions for computing leaf nodes and internal nodes; and (2) SLH-DSA has relatively large signature sizes and computational costs, and therefore can benefit significantly from the reductions offered by MTL mode; and (3) hash-based techniques are well understood and offer a conservative choice for long-term security relative to newer NIST selected signature schemes based on lattice-based cryptography. SLH-DSA is based on SPHINCS+ [SPHINCSPLUS], one of the submissions to NIST's PQC evaluation project [I-D.harvey-cfrg-mtl-mode] describes the combination of MTL mode with SLH-DSA.

This initial version of the draft focuses on the code-points applicable to DNSKEY and RRSIG formulation and a proposed DNSSEC protocol change to support retrieval of MTL mode condensed signatures and MTL mode full signatures as described in Section 3, Section 9.4,

and Section 9.5 of [I-D.harvey-cfrg-mtl-mode]. Later versions may describe DNSSEC protocol and/or operational changes related to zone signing, zone composition, zone updates, zone transfer, name server processing, resolver signature processing, and resolver caching.

2. Conventions Used in This Document

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.

Double pipe characters, "||" are used in this document to indicate concatenation of the elements preceding and following the double pipe characters.

All numeric DNSKEY elements and RRSIG elements specified in this document are unsigned integers in network byte order (big endian order).

3. DNSKEY Resource Records

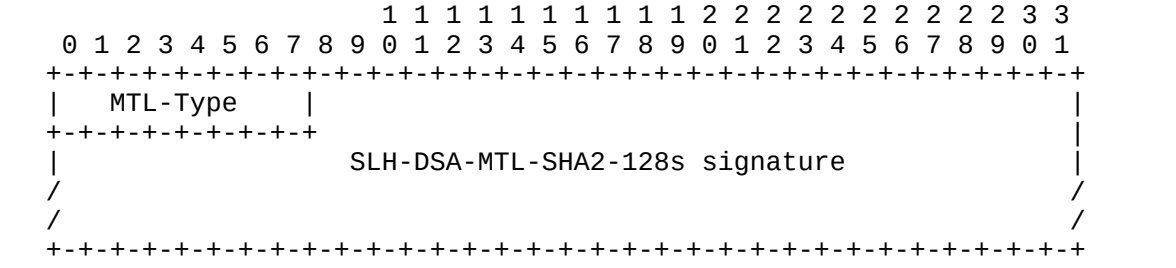
An SLHDSAMTLSHA2128S key consists of a 32-octet value, which is encoded into the Public Key field of a DNSKEY resource record as a simple bit string. SLHDSAMTLSHA2128S keys are generated as SLH-DSA keys using the SLH-DSA-SHA2-128s parameter set, as defined in 10.1 and 11 of [FIPS205].

An SLHDSAMTLSHAKE128S key consists of a 32-octet value, which is encoded into the Public Key field of a DNSKEY resource record as a simple bit string. SLHDSAMTLSHAKE128S keys are generated as SLH-DSA keys using the SLH-DSA-SHAKE-128s parameter set, as defined in 10.1 and 11 of [FIPS205].

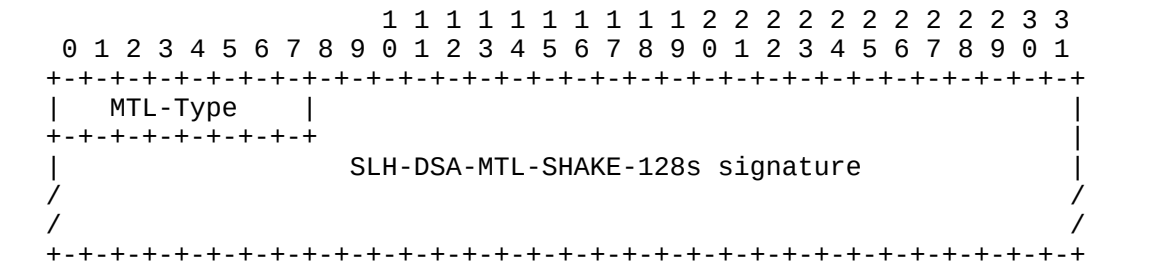
4. RRSIG Resource Records

MTL mode signatures are either full or condensed as described in [I-D.harvey-cfrg-mtl-mode]. SLHDSAMTLSHA2128S and SLHDSAMTLSHAKE128S signatures utilize a one-octet prefixed MTL-Type field to indicate whether the signature is condensed (0) or full (1).

An SLHDSAMTLSHA2128S signature consists of a variable-length value, which is encoded into the Signature field of an RRSIG resource record as a simple bit string as the concatenation of the MTL-Type and a SLH-DSA-MTL-SHA2-128s signature as described in [I-D.harvey-cfrg-mtl-mode]:



An SLHDSAMTLSHAKE128S signature consists of a variable-length value, which is encoded into the Signature field of an RRSIG resource record as a simple bit string as the concatenation of the MTL-Type and a SLH-DSA-MTL-SHAKE-128s signature as described in [I-D.harvey-cfrg-mtl-mode]:



The signature and verification algorithms for both SLH-DSA-MTL-SHA2-128s and SLH-DSA-MTL-SHAKE-128s are described in 9.1 and 9.2 of [I-D.harvey-cfrg-mtl-mode]. The signature and verification algorithms for the underlying signature algorithms used for signing ladders in SLH-DSA-MTL-SHA2-128s and SLH-DSA-MTL-SHAKE-128s full signatures, SLH-DSA-SHA2-128s and SLH-DSA-SHAKE-128s respectively, are described in 10.2 and 10.3 of [FIPS205].

5. Algorithm Numbers for DS, DNSKEY, and RRSIG Resource Records

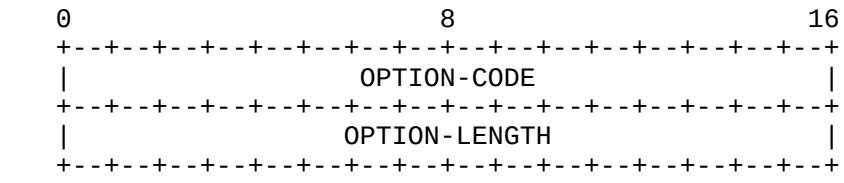
The algorithm number associated with the use of SLHDSAMTLSHA2128S in DS, DNSKEY, and RRSIG resource records is TBD. The algorithm number associated with the use of SLHDSAMTLSHAKE128S in DS, DNSKEY, and RRSIG resource records is TBD. This registration is fully defined in the IANA Considerations section.

6. The mtl-mode-full EDNS(0) Option

MTL mode signatures are either full or condensed. A MTL mode-aware client MAY request that signatures be returned in the full format by providing the mtl-mode-full EDNS(0) option in the OPT meta-RR of its query [RFC6891].

6.1. Option Format

The mtl-mode-full option is encoded as follows:



Where:

OPTION-CODE The EDNS0 option code assigned to mtl-mode-full, TBD.
OPTION-LENGTH Always zero.

6.2. Use By Responders

When a query includes the mtl-mode-full option, the response requirement depends on the number of RRSIG records in the response that were produced in MTL mode:

* If exactly one RRSIG record in the response was produced in MTL mode, then that RRSIG record MUST be returned in the full signature format.
* If more than one RRSIG record in the response was produced in MTL mode, then enough of these RRSIG records MUST be returned in the full signature format to ensure that every other RRSIG in the response that was produced in MTL mode can be verified.

When the mtl-mode-full option is not included, every signature in the response that was produced in MTL mode MUST be returned in the condensed signature format.

As described in 9.2 of [I-D.harvey-cfrg-mtl-mode], when a verifier receives a condensed signature, the verifier determines whether any of the MTLs it has previously verified includes a rung that is compatible with the authentication path in the condensed signature. If not, then the verifier requests a new signed ladder. Accordingly, a resolver SHOULD first query a name server without the mtl-mode-full option, and then, if needed, re-issue the query with the mtl-mode-full option. Since responses to queries with the mtl-mode-full option are expected to be large, it is RECOMMENDED that queries with the mtl-mode-full option be issued over transports (e.g., TCP, TLS, QUIC) that support large responses without truncation and/or fragmentation.

7. Implementation Considerations

Signing RRSets in batches rather than as individual messages can leverage MTL mode to reduce the number of public/private key signing operations performed with the underlying signature algorithm. This results in reducing the average computational overhead per message signed. This practice can also reduce the load on a hardware security module. Batches also benefit the verifier by reducing the number of full signatures required for validation because multiple RRSIGs can be verified by the ladder covering the batch. The appropriate batch size will depend on the properties of the zone and the requirements of the zone operator. Batch size needs to be considered carefully to ensure that new signatures are available in a timely manner while still gaining the benefits of batch signing [MTL-ENDURANCE].

8. Examples

Examples with DNS tools are described in Section 10

9. IANA Considerations

This document updates the IANA registry for DNSSEC "Domain Name System Security (DNSSEC) Algorithm Numbers" located at https://www.iana.org/assignments/dns-sec-alg-numbers/dns-sec-alg-numbers.xhtml (https://www.iana.org/assignments/dns-sec-alg-numbers/dns-sec-alg-numbers.xhtml). The following entries are requested to be added to the registry subject to the Number update:

SLH-DSA-MTL-SHA2-128s

Number	TBD
Description	SLH-DSA-MTL-SHA2-128s
Mnemonic	SLHDSAMTLSHA2128S
Zone Signing	Y
Trans. Sec.	*
Reference	This specification

SLH-DSA-MTL-SHAKE-128s

Number	TBD
Description	SLH-DSA-MTL-SHAKE-128s
Mnemonic	SLHDSAMTLSHAKE128S
Zone Signing	Y
Trans. Sec.	*
Reference	This specification

* There has been no determination of standardization of the use of these algorithms with Transaction Security.

10. Implementation Status

NOTE: Please remove this section and the reference to RFC 7942 prior to publication as an RFC.

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in RFC 7942. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to RFC 7942, "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit".

For testing purposes, SLH-DSA-MTL-DNSSEC has been implemented in the following DNS open-source applications:

* LDNS for key generation, zone signing, and zone verification with MTL mode: https://github.com/Verisign/mtl-mode-ldns (https://github.com/Verisign/mtl-mode-ldns)
* NSD authoritative name server with MTL mode: https://github.com/verisign/mtl-mode-nsd (https://github.com/verisign/mtl-mode-nsd)
* Unbound recursive resolver with MTL mode: https://github.com/Verisign/mtl-mode-unbound (https://github.com/Verisign/mtl-mode-unbound)

These implementations depend on the reference implementation of MTL mode which is available in C. The MTL library can be found at https://github.com/Verisign/MTL (https://github.com/Verisign/MTL).

11. Security Considerations

The security considerations of [FIPS205] and [I-D.harvey-cfrg-mtl-mode] are inherited in the usage of SLH-DSA-MTL in DNSSEC.

SLH-DSA-MTL-SHA2-128s and SLH-DSA-MTL-SHAKE-128s are intended to operate at around the 128-bit security level against classical attacks and the 64-bit level against quantum attacks, consistent with NIST’s security level I.

A private key used for a DNSSEC zone MUST NOT be used for any other purpose than for that zone. Otherwise, cross-protocol or cross-application attacks are possible.

12. Acknowledgements

The authors would like to acknowledge the following individuals for their contributions to the development of this document: Scott Hollenbeck, Swapneel Sheth. This I-D has drawn from helpful examples of document structure and specification text from various DNSSEC algorithm RFCs. The authors express their gratitude to the authors of those RFCs for their contributions.

13. References

13.1. Normative References

[FIPS205] National Institute of Standards and Technology (NIST), "Stateless Hash-Based Digital Signature Standard", FIPS PUB 205, DOI 10.6028/NIST.FIPS.205, 13 August 2024, <https://doi.org/10.6028/NIST.FIPS.205>.

[I-D.harvey-cfrg-mtl-mode] Harvey, J., Kaliski, B., Fregly, A., Sheth, S., and D. McVicker, "Merkle Tree Ladder (MTL) Mode Signatures", Work in Progress, Internet-Draft, draft-harvey-cfrg-mtl-mode-09, 24 March 2026, <https://datatracker.ietf.org/doc/html/draft-harvey-cfrg-mtl-mode-09>.

[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>.

[RFC4033] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "DNS Security Introduction and Requirements", RFC 4033, DOI 10.17487/RFC4033, March 2005, <https://www.rfc-editor.org/info/rfc4033>.

[RFC4034] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Resource Records for the DNS Security Extensions", RFC 4034, DOI 10.17487/RFC4034, March 2005, <https://www.rfc-editor.org/info/rfc4034>.

[RFC4035] Arends, R., Austein, R., Larson, M., Massey, D., and S. Rose, "Protocol Modifications for the DNS Security Extensions", RFC 4035, DOI 10.17487/RFC4035, March 2005, <https://www.rfc-editor.org/info/rfc4035>.

[RFC6891] Damas, J., Graff, M., and P. Vixie, "Extension Mechanisms for DNS (EDNS(0))", STD 75, RFC 6891, DOI 10.17487/RFC6891, April 2013, <https://www.rfc-editor.org/info/rfc6891>.

[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>.

13.2. Informative References

[CTRSAMTL] Kaliski, B., Fregly, A.M., Harvey, J., and S. Sheth, "Merkle Tree Ladder Mode: Reducing the Size Impact of NIST PQC Signature Algorithms in Practice", 2023.

[MTL-ENDURANCE] Tran, M. and T. Chung, "Randomized Evaluation of SLH-DSA-MTL's Impact on Reducing PQ-DNSSEC Signature Sizes", 18 March 2025, <https://github.com/IQTF/pq-dnssec-materials/raw/refs/heads/main/IETF122/Tran_Randomized_evaluation_of_SLH-DSA-MTL's_impact_on_reducing_PQ-DNSSEC_signature_sizes.pdf>.

[NISTSELECTIONS] National Institute of Standards and Technology (NIST), "Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process", June 2022, <https://nvlpubs.nist.gov/nistpubs/ir/2022/NIST.IR.8413-upd1.pdf>.

[SPHINCSPLUS]
Bernstein, D., Huelsing, A., Koelbl, S., Niederhagen, R., Rijneveld, J., and P. Schwabe, "The SPHINCS+ Signature Framework", Cryptology ePrint Archive, Report 2019/1086, 2019, <https://eprint.iacr.org/2019/1086.pdf>.

Appendix A. Change Log

00: Initial draft of the document.
01: Update expiration of document
02: Add appendix with example of MTL Mode signatures in DNSSEC
03: Update draft to align with FIPS-205
04: Updated the implementation status section with links to known implementations
05: Added the implementation considerations section on batch signing
06: Update to reference the v09 version of MTL Mode and note that other underlying signatures are possible but would be specified in other documents.

Authors' Addresses

A. Fregly
Verisign Labs
12061 Bluemont Way
Reston, VA 20190
United States of America
Email: afregly@verisign.com
URI: https://www.verisignlabs.com/

J. Harvey
Verisign Labs
12061 Bluemont Way
Reston, VA 20190
United States of America
Email: jsharvey@verisign.com
URI: https://www.verisignlabs.com/

B. Kaliski
Verisign Labs
12061 Bluemont Way
Reston, VA 20190
United States of America
Email: bkaliski@verisign.com
URI: https://www.verisignlabs.com/

D. Wessels
Verisign Labs
12061 Bluemont Way
Reston, VA 20190
United States of America
Email: dwessels@verisign.com
URI: https://www.verisignlabs.com/