



How Merkle Tree Ladders Help Address PQC DNSSEC Challenges

Swapneel Sheth, Joseph Harvey, and Andrew Kaizer

IETF DNSOP PQ DNSSEC Interim

October 2nd, 2026

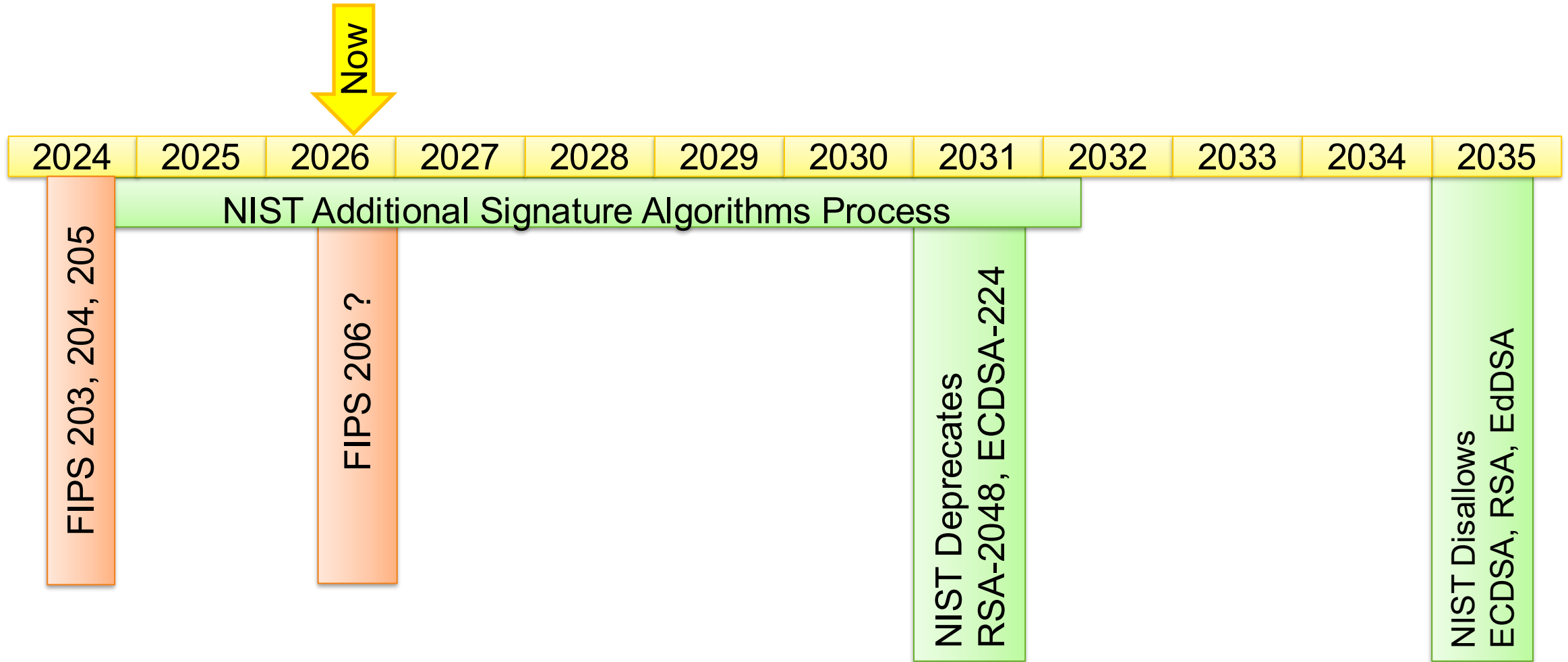
PQ DNSSEC Context

As presented at IETF 123

- DNSSEC currently uses digital signature algorithms (e.g., Algorithm 8 - RSASHA256 and Algorithm 13 - ECDSAP256SHA256) that are at risk of compromise by quantum computers.
- The PQC signature algorithms that are currently standardized (e.g., ML-DSA in FIPS 204 and SLH-DSA in FIPS 205) have large signature sizes relative to DNSSEC's constraints.
- NIST's "onramp" call for additional PQC signature algorithms intends to standardize algorithms with smaller signature sizes – but they likely will be based on newer cryptographic assumptions and likely won't be available in near future.

NIST PQC Timeline

As presented at IETF 123



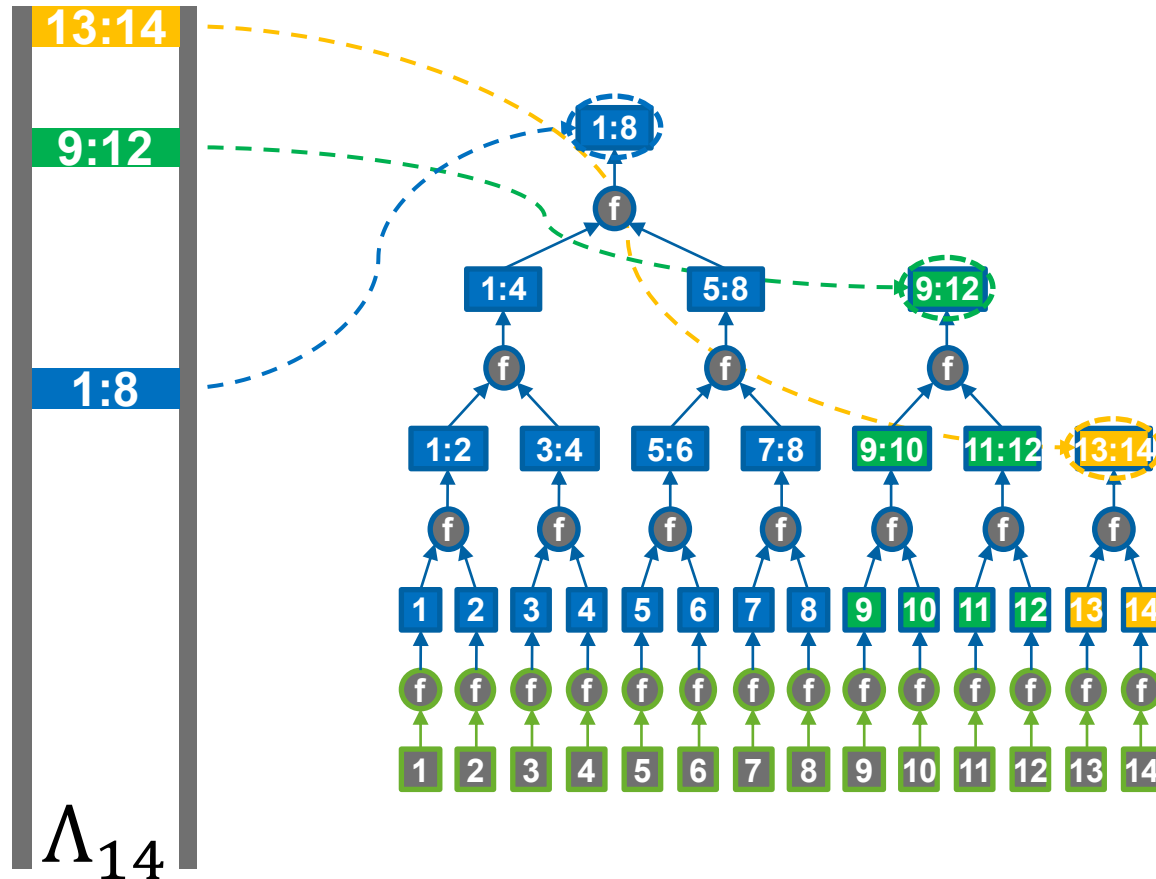
Proposed Diversity Strategy

As presented at IETF 123

-  **DO NOT wait** for NIST's onramp effort to conclude before starting to prepare, anticipating the availability of one or more additional signature algorithms more suitable for DNSSEC in terms of signature size.
-  **Find a way** to deploy the currently standardized PQC algorithms.
-  Follow post-quantum diversity strategy based on the consensus from the PQC DNSSEC side meetings (see [draft-sheth-pqc-dnssec-strategy](#))
 - At least **one conservatively designed algorithm**
 - At least **one low-impact drop-in algorithm**
-  DNS operators choose which supported algorithm to use to sign a particular zone.

MTL : Reducing the Operational Impact

As presented at 4th NIST PQC Conference



- Rather than signing individual messages, MTL signs Merkle Tree Ladders
- Messages are authenticated with Merkle proofs relative to ladders
- Ladders provide backward compatibility since they can verify Merkle proofs constructed relative to future ladders too

MTL Benefits for DNSSEC Summary

- Conserve space
 - Each DNS RRset now has a small Merkle proof
- Reduce computation
 - Only need 1 large PQ signature for the MTL
- Preserve bandwidth
 - Once a client has the large PQ signature, they only need to request Merkle proofs
- More energy and resource efficient to sign, store and serve than plain per-records signing

MTL Open-Source Implementations

Reference Open-Source	Link
MTL reference library	https://github.com/verisign/MTL
MTL LDNS library	https://github.com/verisign/mtl-mode-ldns
NSD [authoritative name server]	https://github.com/verisign/mtl-mode-nsd
Unbound [recursive resolver]	https://github.com/verisign/mtl-mode-unbound
MTL Demonstration App [Jupyter Notebook]	https://github.com/verisign/MTL/tree/v1.3.1/examples/demo

MTL Efforts at IETF

	117	118	119	120	121	122	123	124	125	126
Drafts	CFRG-MTL	DNSOP-SLHDSA-MTL		MTL Considerations				Post-Quantum Cryptography Strategy for DNSSEC		DNSOP-MLDSA-MTL
Presentations	MTL@CFRG	MTL @HOT RFC		MTL@CFRG	MTL@PQC DNSSEC side meeting	MTL@PQC DNSSEC side meeting	PQC DNSSEC strategy @PQC DNSSEC side meeting	PQC DNSSEC WG Home @DNSOP		ML-DNSA-MTL @HOT RFC
Hackathons		MTLLIB Introduction		SLH-DNSA-MTL DNS POC	NSD& Unbound MTL POC	Worldwide POC metrics collection and EDNS test	MTL test in NSD, Unbound, Bind, and CoreDNS	PQC denial of existence		MTL transport independent benefits
	2023		2024			2025			2026	

Diverse group involved: ISC, NLnet Labs, Universidad Carlos III de Madrid, University of Ostrava, University of Twente, Virginia Tech

Produced: open-source contributions, presentations, research papers, and demonstrations

Latest MTL Drafts

- draft-kaizer-dnsop-ml-dsa-mtl-dnssec-02
 - Base version of MTL that minimizes implementation complexity
 - Removed EDNS(0) overhead that existed in previous drafts
 - Each RRSIG is now self-contained providing parity with existing DNSSEC algorithms
 - Continues to provide zone size benefits for authoritative operators
- draft-hdong-dnsop-ml-dsa-mtl-dnssec-sigtag-ext-00
 - EDNS(0) option to communicate known signatures to servers
 - Allows for bandwidth, cache storage benefits to those who opt-in

IPR: <https://datatracker.ietf.org/ipr/search/?submit=draft&id=draft-kaizer-dnsop-ml-dsa-mtl-dnssec> <https://datatracker.ietf.org/ipr/search/?submit=draft&id=draft-hdong-dnsop-ml-dsa-mtl-dnssec-sigtag-ext>

MTL as Two-Stage Improvement to ML-DSA-44

- Base MTL draft has been streamlined to ease implementation and focus on authoritative resource savings while functioning as a drop-in comparable to ML-DSA
- Extension draft provides opt-in benefits for bandwidth, recursive resource savings

	Drop-in DNSSEC Algorithm	Authoritative Resource Savings	Bandwidth Savings	Recursive Resource Savings
ML-DSA-44 (draft-westerbaan-dnssec-mldsa)			 ~38x more!! All TCP	
ML-DSA-44 with Base MTL (draft-kaizer-dnsop-ml-dsa-mtl-dnssec)			 ~40x more!! All TCP	
+ Sig Tag extension (draft-hdong-dnsop-ml-dsa-mtl-dnssec-sigtag-ext)	N/A	N/A	 Some UDP	

MTL Benefits for DNSSEC Summary

- Conserve space
 - Each DNS RRset now has a small Merkle proof
- Reduce computation
 - Only need 1 large PQ signature for the MTL
- Preserve bandwidth
 - Once a client has the large PQ signature, they only need to request Merkle proofs
- More energy and resource efficient to sign, store and serve than plain per-records signing

Call for Action

- Support adoption of MTL for DNSSEC
 - Base specification streamlines MTL, providing parity with DNSSEC algorithms like ML-DSA-44 where each RRSIG is self-contained
 - SigTag specification provides bandwidth and caching benefits
 - Results presented at IETF Hackathons & PQ DNSSEC side meetings have shown tangible benefits from MTL
- Adoption would fulfill the strategy goal of a conservative algorithm with reduced impact for PQ DNSSEC



© 2026 VeriSign, Inc. All rights reserved. VERISIGN and other trademarks, service marks, and designs are registered or unregistered trademarks of VeriSign, Inc. and its subsidiaries in the United States and in foreign countries. All other trademarks are property of their respective owners.