Post-Quantum Diversity for DNSSEC: Routine Performance, Resilient Fallback

DNS OARC 45 October 7-8, 2025

Minh Hoang Tran – Virginia Tech
Joe Harvey – Verisign Labs
Burt Kaliski – Verisign Labs
Daniel McVicker – Verisign Labs
Benno Overeinder – NLnetLabs
Swapneel Sheth – Verisign Labs
Ondřej Surý – ISC & University of Ostrava

PQ DNSSEC Context

- DNSSEC currently uses digital signature algorithms 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.

Finding a Way – Alternatives and Considerations

Select a high-performance signature algorithm to ensure **routine performance** with a conservative signature algorithm for **resilient fallback**. Enables the potential for newer low-impact, algorithms while minimizing overall risk of adopting something newer and less proven.

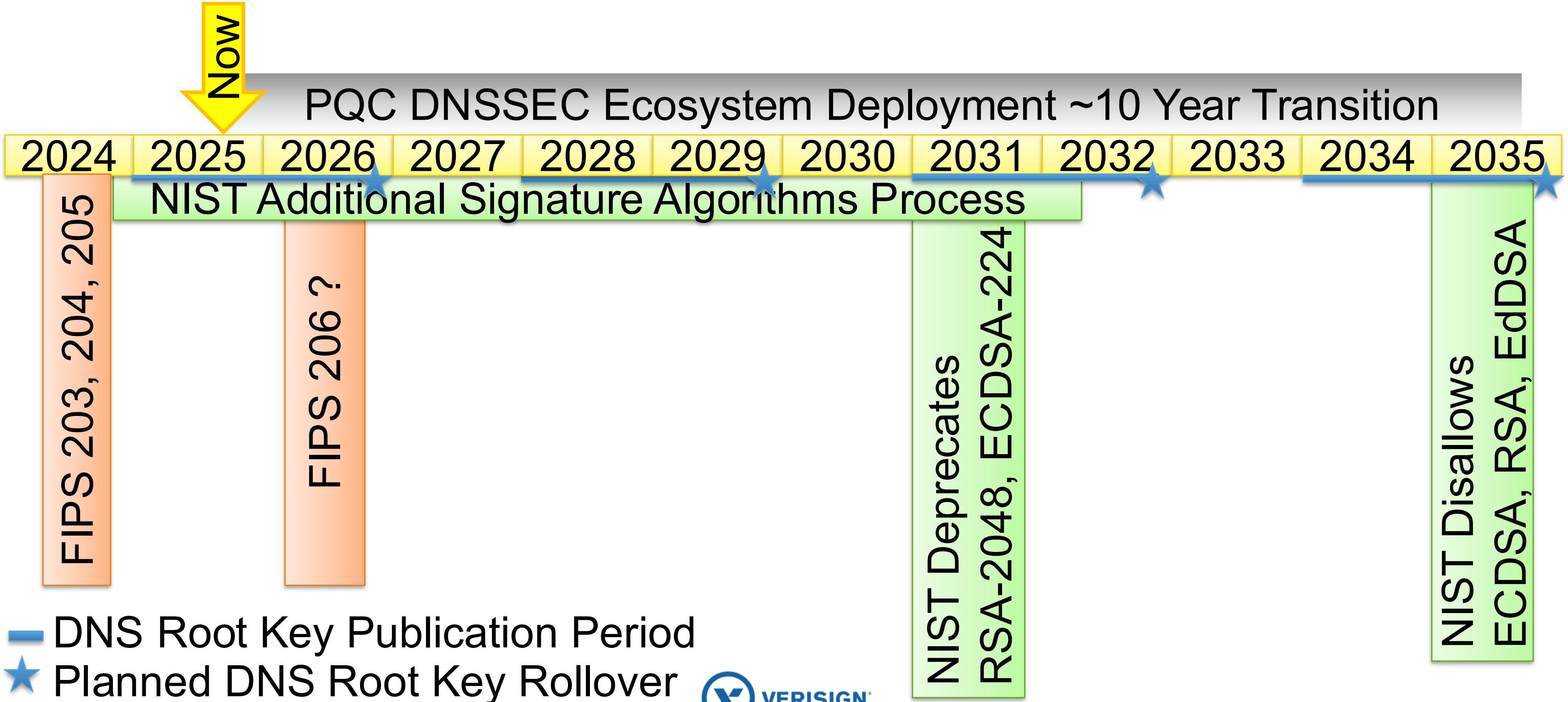
Routine Performance - Low-impact, drop-in algorithm used same way as traditional signature algorithms.

- No recommendation yet for low-impact, “drop-in” algorithm

Resilient Fallback - Conservatively designed algorithm unlikely ever to need to be replaced.

- Propose SLH-DSA as the choice for conservatively designed algorithm
- Open to considering ML-DSA as well as Falcon
- Open to HSS-LMS and XMSS^MT, while noting that state management introduces an operational risk
- Suggest allowing techniques like MTL mode that reduce the impact of the conservative algorithms.

Timeline Considerations



Operational Considerations

- Algorithm robustness and reliability
  - Currently have 3+ algorithms available (RSASHA256, ECDSAP256SHA256, ED25519).
    - Operators can pick which they prefer
    - Can use multiple for resiliency or switch to another at any time
  - PQC Options
    - Have not found one PQC algorithm that fits all criteria, let alone several
    - Need to have at least one conservative algorithm to use in case an algorithm is broken to allow for migration to new algorithms.

Especially concerning with zones like Root that publish their keys way in advance of using them to have a conservative fallback.

Proposed Diversity Strategy

- ✗ 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.

- ✓ Use a post-quantum diversity strategy for DNSSEC that involves at least one algorithm from two sets with complementary properties.
  - At least one conservatively designed algorithm
  - At least one low-impact drop-in algorithm

- ✓ Allow DNS operators choose which supported algorithm to use to sign a particular zone.

Routine Performance

PQC DNSSEC Research

New PQC for DNSSEC

Ondřej Surý – University of Ostrava/ISC Research

- NIST PQC Challenge
  - FALCON – lattice-based
- NIST Additional Digital Signing Scheme Challenge (Round 2)
  - HAWK – lattice-based
  - SQISign – isogeny-based
  - MAYO – multivariete-based
- ASIACRYPT 2023
  - Antrag – Espitau, Thomas, Thi Thu Quyen Nguyen, Chao Sun, Mehdi Tibouchi, and Alexandre Wallet. ‘Antrag: Annular NTRU Trapdoor Generation’, 2023. https://eprint.iacr.org/2023/1335.
    - Size-reduced variant of FALCON

https://theses.cz/auth/id/8p3ric/

New PQC for DNSSEC
Root Zone Signing and Validation

SIGNING (ROOT, 1 KSK, 1 ZSK, RAW)

ALGORITHM	MEAN	σ	SIGNATURE S/S	RAW SIZE
FALCON-512	4881.9 ms	26.8 ms	589	2891700
HAWK-256	195.5 ms	4.9 ms	62001	1727793
HAWK-512	261.0 ms	9.6 ms	49821	2582375
SQIsign	*54528.1 ms*	67.9 ms	51	1445334
MAYO	1086.6 ms	48.7 ms	2746	2301478
ANTRAG-512+	5339.6 ms	111.2 ms	546	2685056
RSA 2048	845.7 ms	3.0 ms	3980	1746936
ECDSAP256	218.1 ms	10.2 ms	44286	1211056
ED25519	240.6 ms	6.3 ms	47288	1210992
MTL	1501.6 ms	15.4 ms	5314	1604362

VALIDATION (ROOT, 1 KSK, 1 ZSK, RAW)

ALGORITHM	MEAN	σ
FALCON-512	403.7 ms	1.1 ms
HAWK-256	232.5 ms	1.4 ms
HAWK-512	359.4 ms	66.0 ms
SQIsign	22338.5 ms	35.0 ms
MAYO	995.8 ms	26.8 ms
ANTRAG-512	548.6 ms	1.4 ms
RSA 2048	250.2 ms	18.9 ms
ECDSAP256	610.0 ms	4.5 ms
ED25519	819.4 ms	4.5 ms

New PQC for DNSSEC
DNS Message Sizes

ALGORITHM	SOA	DNSKEY	NXDOMAIN	NODATA	Delegation
FALCON-512	797	3244	1520	1518	1023
HAWK-256	380	1237	686	684	606
HAWK-512	686	2691	1298	1296	912
SQIsign	279	366	484	482	505
MAYO	1108	3382	1096	1094	811
ANTRAG-512	723	2216	1372	1370	949
RSA 2048	387	864	700	698	613
ECDSAP256	195	280	316	314	421
ED25519	195	216	316	314	421

Doesn’t fit into 1232 bytes

Doesn’t fit into 1452 bytes

Large Public Keys are OK(ish)
Large Signature affects performance

- FALCON-512 – slow due to signature size, DNS switches to TCP
- HAWK-256 – promising, but cryptographically weaker algorithm
- **HAWK-512 – usable algorithm**
- SQIsign – small keys, small signatures, very slow due to computational complexity
- **MAYO – larger public key, but usable thanks to small signatures**
- **ANTRAG-512 – usable algorithm (outside NIST)**

New PQC for DNSSEC
Future Work

- Study of anomalous operations
  - Attacks on DNS resolvers
  - DDoS attacks using DNS
  - Configuration errors
- Study at multiple levels of the DNS hierarchy (TLD)
- Testing of other algorithms
  - Completely new algorithms (we should talk to NIST and cryptographers)
  - Algorithms with large public keys (like MAYO)

Resilient Fallback

PQC DNSSEC Research

Standardized Algorithms
Verisign Labs Research

Algorithm	PQC Algorithm	Time to Sign (seconds)	Time to Verify (seconds)	Signed Zone Size (MB)	Public Key Size (bytes)
RSA-2048	No	2.4	0.4	2.72	260
ECDSA	No	0.4	0.7	2.06	64
FL-DSA-512	Yes	1.5	0.6	4.36	897
ML-DSA-44	Yes	0.7	0.9	11.13	1312
SLH-DSA-SHA2-128	Yes	534.5	2.7	32.15	32
SLH-DSA-SHAKE-128	Yes	1058.4	3.1	32.22	32

Sample zone with 1500 Delegated Sub-Domains

Query/Response Performance

			Message Size			Query Time (10 samples)		
Protocol	Algorithm	Record	Query (bytes)	Response (bytes)	Truncated	Average (ms)	Median (ms)	Stdev (ms)
UDP	RSA-2048	NS	54	715		1.53	1.50	0.10
TCP	RSA-2048	NS	54	715		2.21	2.18	0.25
UDP	ECDSA	NS	56	527		1.59	1.58	0.08
TCP	ECDSA	NS	56	527		2.24	2.31	0.17
UDP	FL-DSA-512	NS	57	1120		1.54	1.52	0.06
TCP	FL-DSA-512	NS	57	1120		2.30	2.40	0.22
UDP	ML-DSA-44	NS	60	150	TRUE	0.82	0.84	0.07
TCP	ML-DSA-44	NS	60	2891		1.77	1.71	0.15
UDP	SLH-DSA-SHA2-128	NS	62	152	TRUE	0.82	0.81	0.05
TCP	SLH-DSA-SHA2-128	NS	62	8331		1.80	1.75	0.13
UDP	SLH-DSA-SHAKE-128	NS	64	154	TRUE	0.92	0.81	0.25
TCP	SLH-DSA-SHAKE-128	NS	64	8335		1.86	1.85	0.06

Queries are for a NS record using the network default MTU of 1232 bytes.

PQC DNSSEC NSEC/NSEC3

NSEC Responses

Protocol	Algorithm	Query Size	Response Size(1)	Truncated	Query Time (10 average)	RR Count(2)
UDP	ecdsa	53	579	FALSE	0.00073555	6
UDP	sqisign	55	843	FALSE	0.00072072	6
UDP	mayo-2	54	951	FALSE	0.00070024	6
UDP	rsa-2048	51	1143	FALSE	0.00073285	6
UDP	mayo-1	54	1206	TRUE	0.00057456	4
UDP	hawk	52	735	TRUE	0.00047271	2
UDP	fl-dsa	54	840	TRUE	0.00049796	2
UDP	ml-dsa	54	54	TRUE	0.00030167	0
UDP	slh-dsa-sha	59	59	TRUE	0.00029428	0
UDP	slh-dsa-shake	61	61	TRUE	0.00033879	0

NSEC3 Responses

Protocol	Algorithm	Query Size	Response Size(1)	Truncated	Query Time (10 average)	RR Count(2)
UDP	ecdsa	53	788	FALSE	0.00101929	8
UDP	sqisign	55	1134	FALSE	0.00095022	8
UDP	mayo-2	54	1000	TRUE	0.00074904	6
UDP	rsa-2048	51	1198	TRUE	0.00077493	6
UDP	mayo-1	54	1221	TRUE	0.00062342	4
UDP	hawk	52	734	TRUE	0.0004606	2
UDP	fl-dsa	54	836	TRUE	0.00053473	2
UDP	ml-dsa	54	54	TRUE	0.00041535	0
UDP	slh-dsa-sha	59	59	TRUE	0.00028985	0
UDP	slh-dsa-shake	61	61	TRUE	0.00029438	0

Key Metrics

Algorithm	Time to Sign (seconds)	Time to Verify (seconds)	Signed Zone Size (MB)	Public Key Size (bytes)
slh-dsa-sha2	564.1	2.5	30.79	32
slh-dsa-shake	1080	2.9	30.81	32
ecdsa	0.2	0.5	0.85	64
sqisign	165.4	6.5	1.19	65
rsa-2048	2	0.2	1.56	260
fl-dsa	1.2	0.4	3.13	897
hawk	0.2	0.4	2.72	1024
ml-dsa	0.4	0.7	9.9	1312
mayo-1	0.8	0.5	2.36	1420
mayo-2	0.5	0.3	1.33	4912

Tested on a zone with 1500 labels (with A records). A copy of the zone file is signed for each signature algorithm and then served via NSD.

(1) Response size is the size of the initial DNS response returned by NSD. For truncated responses, the full response would be larger than the 1232 MTU and indicates how much of the response fit before it was truncated.

(2) RR Count is the number of RR's included in the initial DNS response.

Standardized Algorithms - Summary

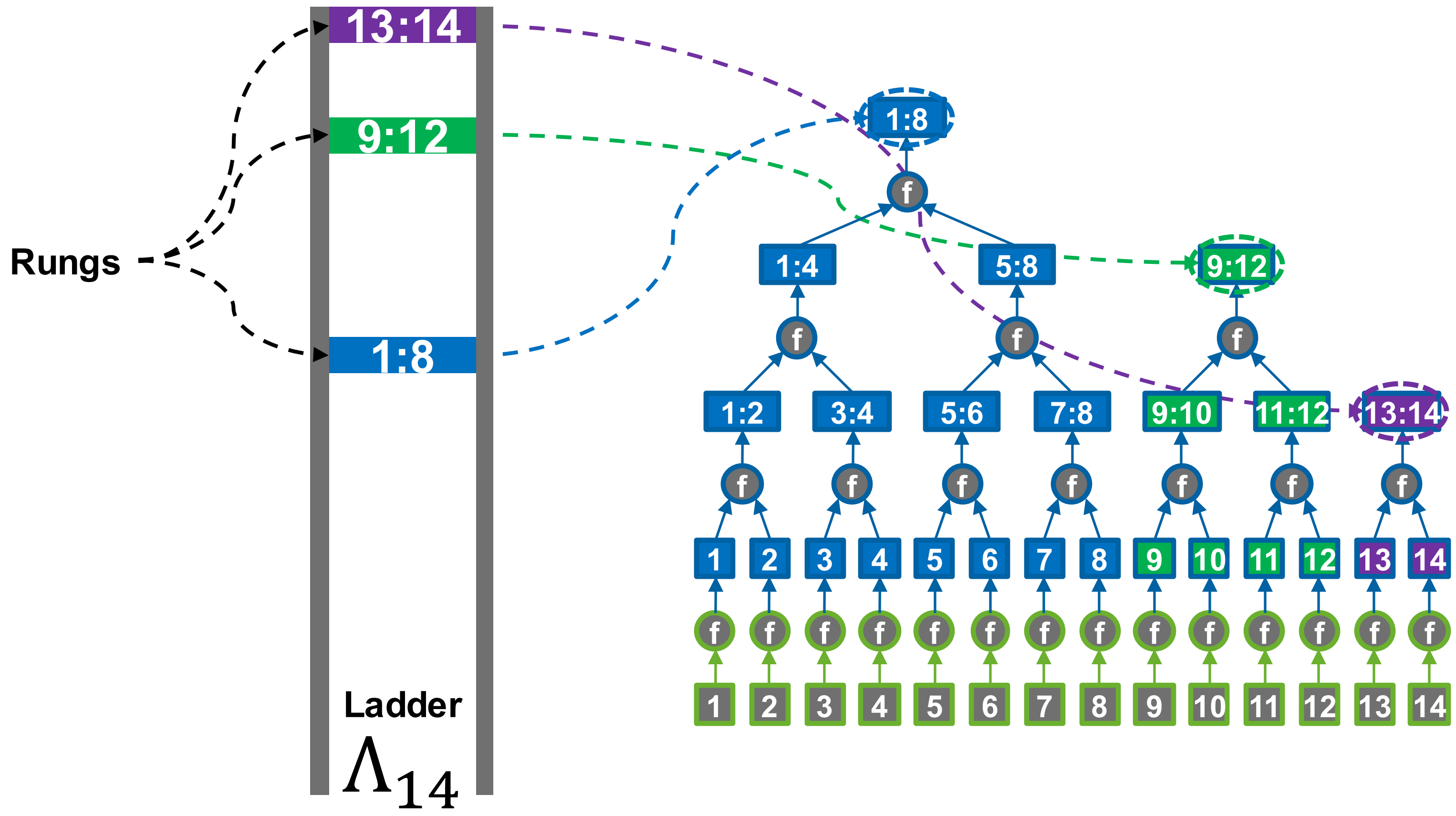
- None of the current NIST standard algorithms are close enough to RSA and ECDSA to provide an easy transition path.
  - ML-DSA is popular right now for WebPKI, although the signature size and public key size mean that it will not work over UDP for DNS
  - FN-DSA-512 is the closest, however it does not work for cases like non-existence (NSEC or NSEC3)
- Still some research to do to make denial of existence (NSEC and NSEC3) work with PQC algorithms.

Modes of Operation

PQC DNSSEC Research

What is MTL Mode?

MTL mode is a method for reducing a signature scheme's operational impact on an expanding message series.



- Rather than signing individual messages, MTL mode 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
- Useful for signature series that sign multiple things at one time. (DNSSEC, OCSP, etc.)

Impact of MTL Mode Signatures on DNSSEC

NLNETLABS

UNIVERSITY OF AMSTERDAM

RQ1. What is the impact of the MTL mode of SLH-DSA in DNSSEC?

Sub-questions:

RQ2. How does it affect the signature and key size

RQ3. How do sizes and signing and verification performance compare to other PQC algorithms in the context of DNSSEC?

RQ4. How do sizes and performance compare to ECDSAP256?
(for comparison with University of Ostrava/ISC research)

Impact of Merkle Tree Ladder (MTL) Mode Signatures on DNSSEC

Jannik Peters
Security and Network Engineering
University of Amsterdam
jannik.peters@os3.nl

***Abstract*—Quantum computing is expected to threaten current cryptography, especially the algorithms used in many Internet protocols. Quantum-resilient algorithms, colloquially referred to as Post-Quantum Cryptography (PQC), are under active development and standardization. Many of these new algorithms have** the Domain Name System (DNS), have certain limitations that impose requirements on signature and key size, and signing and verification performance on the Post-Quantum Cryptography (PQC) algorithms usable for the Domain Name System

Signature and Key Sizes (RQ2)

Table I
Size of the algorithms' public key and signature in bytes.

Algorithm	Public Key	Signature Size
ECDSAP256SHA256	64	64
SLH-DSA-MTL-SHA2-128s	32	40–500†
SLH-DSA-MTL-SHAKE-128s	32	40–500†
SLH-DSA-SHA2-128s	32	7856
SLH-DSA-SHAKE-128s	32	7856

† **Condensed** signature for a zone with 1 000 000 000 RRsets: max 504B

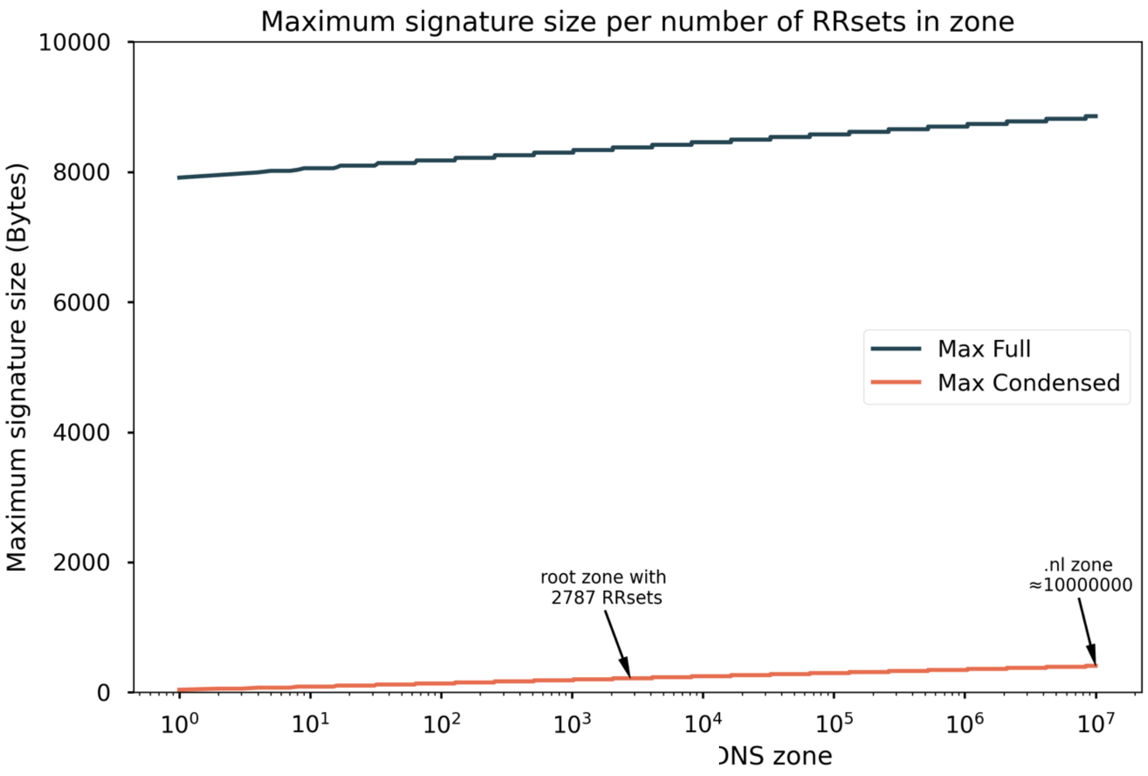


Table II
DNS Message Size (root zone with 1 KSK and 1 ZSK)

Algorithm	SOA	DNSKEY	NXDOMAIN	NODATA	Delegation
ECDSAP256SHA256	197	280	319	316	333
SLH-DSA-MTL-SHA2-128s	8366	8089	8641	8638	486
SLH-DSA-MTL-SHAKE-128s	8366	8089	8641	8638	486
SLH-DSA-SHA2-128s†	7989	8072	15903	15900	8125
SLH-DSA-SHAKE-128s†	7989	8072	15903	15900	8125

† Estimated message sizes, calculated by hand.

Signing & Verification Performance (RQ3)

Table III
Signing time in milliseconds.

Algorithm	Mean	σ
ECDSAP256SHA256	358.17	8.64
SLH-DSA-MTL-SHA2-128s	598.52	10.03
SLH-DSA-MTL-SHAKE-128s	902.96	5.79
SLH-DSA-SHA2-128s	398793.40	776.08
SLH-DSA-SHAKE-128s	807239.86	762.97

Table IV
Verification time in milliseconds.

Algorithm	Mean	σ
ECDSAP256SHA256	550.19	4.52
SLH-DSA-MTL-SHA2-128s	598.06	8.40
SLH-DSA-MTL-SHAKE-128s	600.89	7.63
SLH-DSA-SHA2-128s	2968.52	54.94
SLH-DSA-SHAKE-128s	3476.51	22.02

Comparing Against ECDSA256P
interpolating with O. Surý's results (RQ4)

Table V
Signing time in milliseconds incl. [6] adjusted by performance ratio.

Algorithm	Mean	σ	Sigs/s
ECDSAP256SHA256	358.17	8.64	7781
SLH-DSA-MTL-SHA2-128s	598.52	10.03	4656
SLH-DSA-MTL-SHAKE-128s	902.96	5.79	3087
FALCON-512	8017.19	22.7	103
HAWK-256	321.06	4.15	10894
HAWK-512	428.62	8.13	8754
SQIsign	89547.59	57.52	9
MAYO	1784.45	41.25	482
ANTRAG-512	8768.84	94.19	96

[6] University of Ostrava/ISC Research, O. Surý, "PQC FOR DNSSEC."

What is the impact of the MTL mode of SLH-DSA in DNSSEC? (RQ1)

- Data responses benefit from small signatures
- Performance on par with current signature algorithms

Future work

- Introduce EDNS(0) option: Stored Ladder Version
  - Currently SOA (and DNSKEY) RRs have the full signature
  - The authoritative can send a condensed signature for all SOA (and DNSKEY) RRs in responses if the ladder is up-to-date.

MTL Mode Zone Signing
Virginia Tech

- All-at-once
  - Best for MTL mode, all condensed signatures share a single ladder
  - *Impractical for large zones (TLDs), reduces zone's responsiveness*
- One-by-one
  - Worst for MTL mode, each condensed signature has its own ladder
    - Nameserver: (extremely) large zone size
    - Resolver: more frequent fetching of full signatures
- Batched
  - Tradeoff between batch size and full signature count
  - Condensed signatures within a batch share the same ladder
  - More batches: more responsive, lower signing spike, but more full signatures
  - Fewer batches: less responsive, higher signing spike, but fewer full signatures

COM Zone Signing Strategy

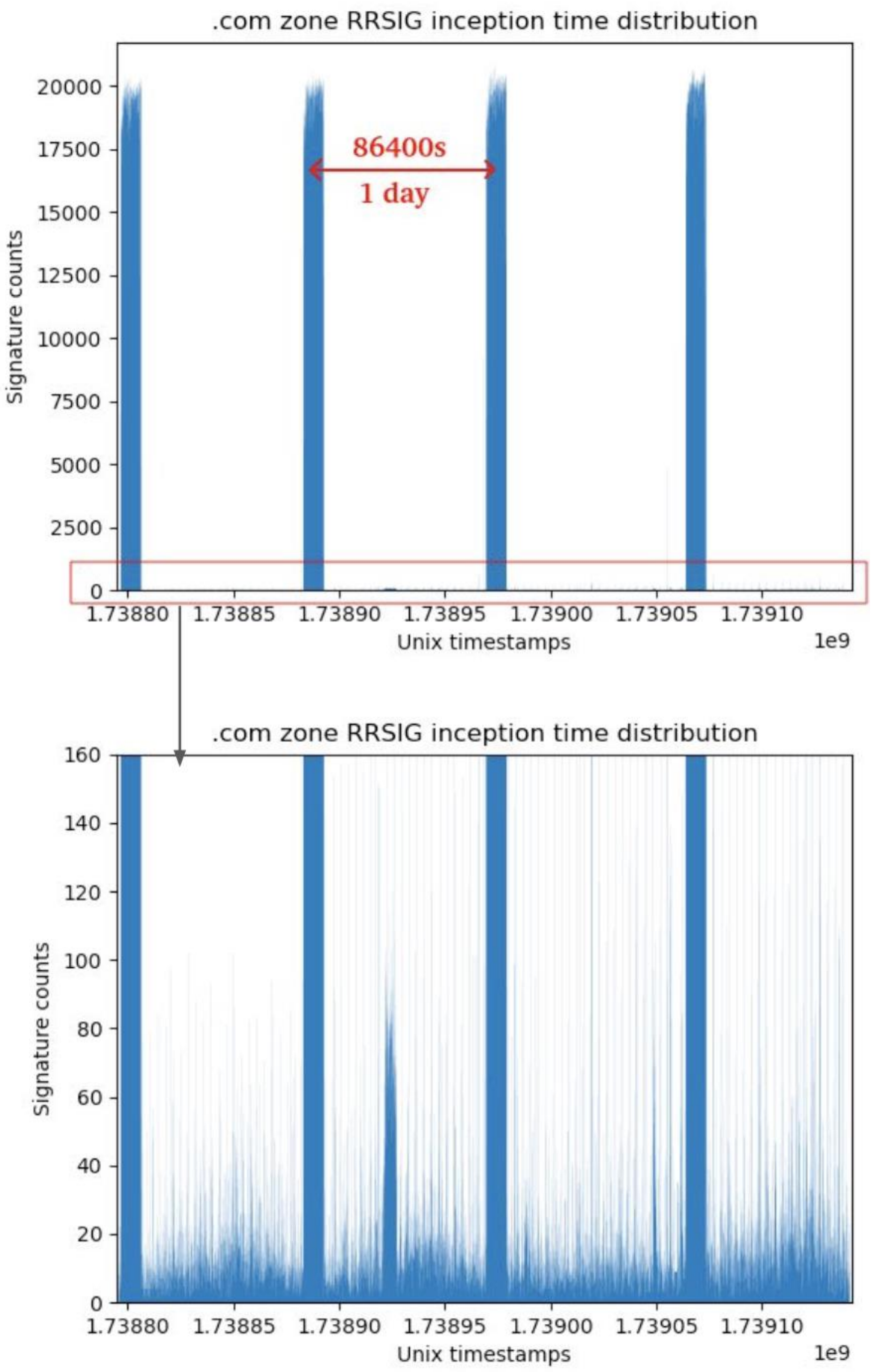
Zone data obtained from CZDS.

RRSIGs' inception times visualized in graph.

Conclusion: Verisign signs COM in 4 batches:

- 1 batch a day
- 25% of zone per batch
- (Comparatively) small number of unbatched RRSIGs

=> MTL-mode's advantages in batch signing can mesh well with some existing zone signing workloads.

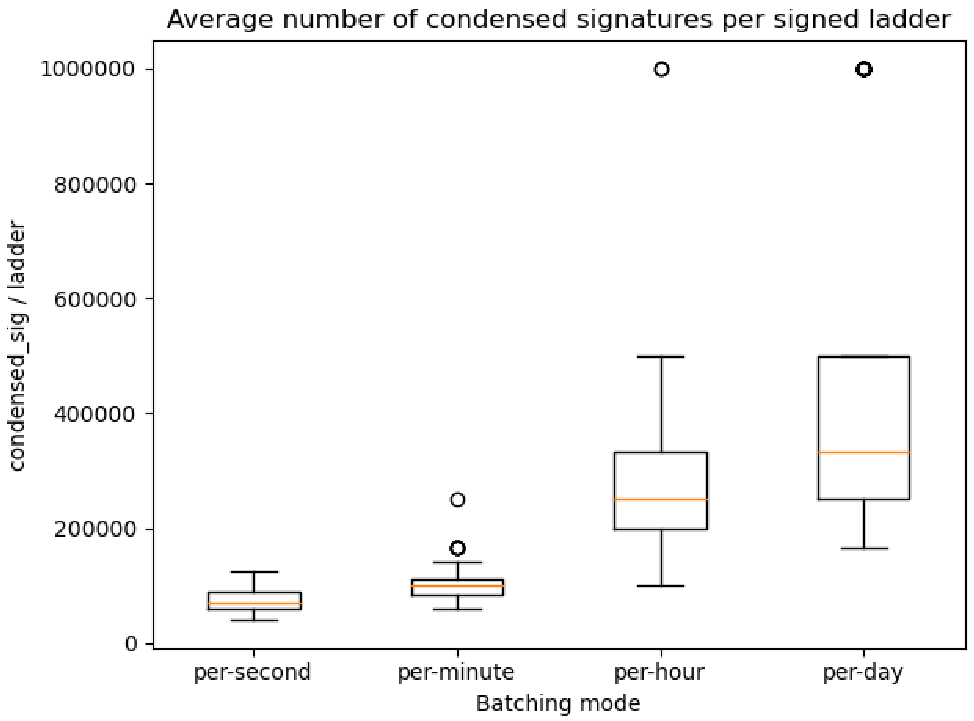


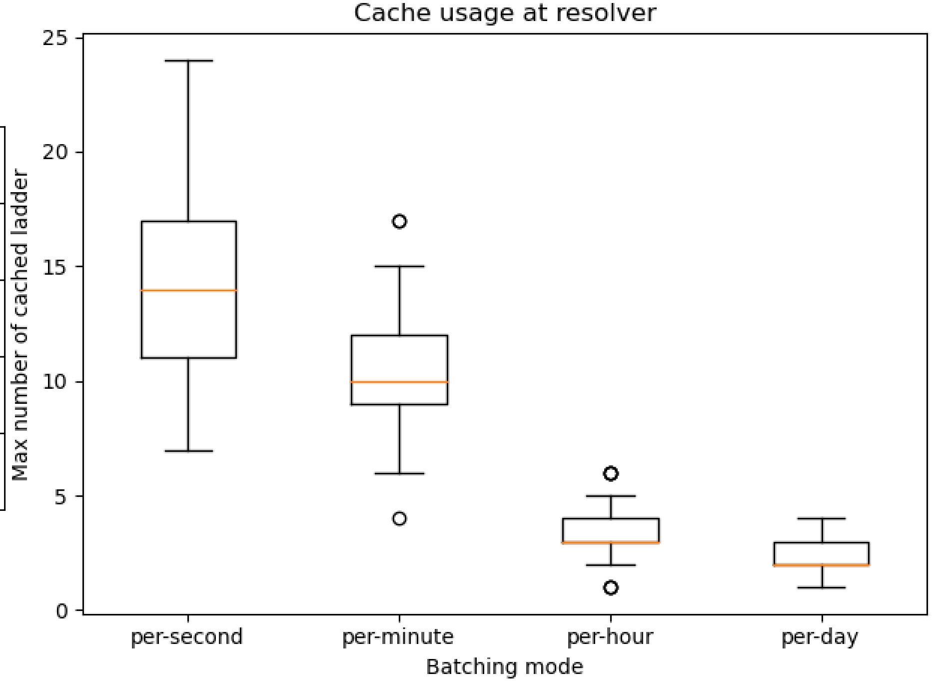
Ladder Endurance

Evaluation of signing strategy on ladder endurance:

- Uniform random querying of COM zone
- Records MTL-batch-signed at different intervals
  - Original RRSIG used as proxy for original signing time
- “Ladder endurance” metrics
  - Average number of condensed signature verified before resolver fetches new full signature with new signed ladder
  - Max number of signed ladder kept in resolver cache
- Values averaged over 100 runs per experiment

Batch signing interval	Avg $sigs_{condensed}$/ladder	Max ladders cached
Per-second	74,254	14.06
Per-minute	101,204	10.46
Per-hour	279,230	3.46
Per-day	471,000	2.23





MTL Mode Zone Signing - Summary

MTL-mode signing zones in larger batches:

- Increases condensed signatures served per ladder
- Reduces number of full ladders cached at ladder => Reduces load (bandwidth, cache use, …) on resolver
- Amortization in large zones can mitigate the cost of full signatures

Batch signing considerations:

- Larger batches per signing operation increases signing load, less ability to “spread out” signing cost of a large zone
- Registries may have SLA obligations on zone responsiveness:
  - Mandatory maximum time limit between domain registration and resolution availability
  - Between-batch zone changes would need to be signed separately, with separate signed ladders

PQC DNSSEC Considerations

Conclusions

- Need continued community participation in PQC DNSSEC discussion, in particular around low-impact drop-in algorithms.
- We should aim for a goal where any standardized PQC signature algorithm can be integrated into DNSSEC in principle.
  - *Perhaps combined with a mode of operation that mitigates its operational impact such as MTL Mode*
- PQC DNSSEC should support a conservatively designed algorithm and a low-impact, drop-in algorithm.
- With NIST deadlines looming for current DNSSEC algorithms, action is needed to ensure the DNS community has time to migrate to PQC.

Appendix

Current Community Efforts (IETF)

PQ DNSSEC Research Side Meetings (https://wiki.ietf.org/en/group/pq-dnssec)

- Evaluating PQC (Falcon and Mayo) in DNSSEC Signing for TLD Operators
- Impact of Merkle Tree Ladder (MTL) Mode Signatures on DNSSEC
- A post-quantum cryptography strategy for DNSSEC
- Randomized simulation of SLH-DSA-MTL's impact on reducing PQ-DNSSEC signature sizes
- PQ DNSSEC with MTL Mode (Verisign) - Metrics and Observations
- Feasibility of the new Post Quantum Cryptography for DNSSEC
- Field study on mitigating the costs of Post-Quantum DNSSEC with Merkle Trees
- PQ DNSSEC with MTL Mode
- A testbed to evaluate post-quantum cryptography in DNSSEC

Hackathons

- 123 – PQC DNSSEC Implementation
- 122 – PQC for DNSSEC
- 122 – PQC DNSSEC Metrics with MTL Mode
- 121 - Experiments with MTL Mode in DNS Resolvers
- 120 - Stateless Hash-Based Signatures in Merkle Tree Ladder Mode (SLH-DSA-MTL) for DNSSEC
- 118 - MTL Mode Experiments

Documents

- Stateful Hash-based Signatures for DNSSEC
- Merkle Tree Ladder (MTL) Mode Signatures
- Stateless Hash-Based Signatures in Merkle Tree Ladder Mode (SLH-DSA-MTL) for DNSSEC
- Impact of Merkle Tree Ladder (MTL) Mode Signatures on DNSSEC

Current Community Efforts (cont.…)

ICANN 70 Workshop

- The Impact of Post-Quantum Cryptography on DNSSEC

PQ Net Workshop

- The Challenges in Using PQC for DNSSEC

ACM SIGCOMM

- Retrofitting post-quantum cryptography in internet protocols: a case study of DNSSEC

SPACE

- Post-quantum DNSSEC over UDP via QNAME-Based Fragmentation

IEEE

- Securing Post-Quantum DNSSEC Against Fragmentation Mis-Association Threat

Real World Crypto Conference

- Field Experiments on Post-Quantum DNSSEC

Network Traffic Measurement and Analysis Conference

- Evaluating Post-Quantum Cryptography in DNSSEC Signing for Top-Level Domain Operators

Masters Thesis

- Beernink, G.J. - Taking the Quantum Leap: Preparing DNSSEC for Post Quantum Cryptography
- Gortzen, J. - Enabling Post-Quantum Signatures in DNSSEC: One ARRF at a time
- Surý, O. - Feasibility of the new Post Quantum Cryptography for DNSSEC