SIDN Labs

https://www.sidnlabs.nl

June 2, 2025

Peer-reviewed Publication

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

Authors: Caspar Schutijser, Ralph Koning, Elmer Lastdrager, and Cristian Hesselman

Venue: 2025 IEEE/IFIP Network Traffic Measurement and Analysis Conference (TMA 2025). Copenhagen, Denmark.

Conference dates: 10 – 13 June, 2025.

Citation:

- Caspar Schutijser, Ralph Koning, Elmer Lastdrager, and Cristian Hesselman. Evaluating Post-Quantum Cryptography in DNSSEC Signing for Top-Level Domain Operators. Proceedings of the 2025 IEEE/IFIP Network Traffic Measurement and Analysis Conference (TMA 2025), Copenhagen, Denmark, 10-13 June 2025.
- Bibtex:

```
@inproceedings{Lastdrager2020,
  author = {Caspar Schutijser and Ralph Koning and Elmer Lastdrager and Cristian Hesselman},
  title = {Evaluating Post-Quantum Cryptography in DNSSEC Signing for Top-Level Domain Operators}
  booktitle = {2025 IEEE/IFIP Network Traffic Measurement and Analysis Conference (TMA)},
  year = {2025},
  address = {Copenhagen, Denmark},
  publisher = {IFIP}
}
```

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

Caspar Schutijser*, Ralph Koning*†, Elmer Lastdrager* and Cristian Hesselman*‡
*SIDN Labs, Arnhem, The Netherlands
†University of Amsterdam, Amsterdam, The Netherlands
‡University of Twente, Enschede, The Netherlands

***Abstract*—Like all other digital systems, the Domain Name System (DNS) needs to remain secure, even when future quantum computers are built. This means that DNS operators need to understand the implications of replacing the currently deployed cryptographic algorithms of the DNS Security Extensions (DNSSEC) with post-quantum cryptography (PQC) ones, which are vastly different. We therefore empirically analyze the signing performance of promising PQC algorithms MAYO-2 and Falcon-512 from a DNS operator point of view, in terms of zone file size, signing time, and validation time, and compare them to currently deployed algorithms RSA-1280 and ECDSA-P256. In our experiments, we use a PQC-enhanced DNSSEC signer, the zone files of three country code Top-Level Domains (ccTLDs) of different sizes, and two different CPU models to measure the effects of CPU optimizations. We find that the DNSSEC signing performance of MAYO-2 is better than RSA-1280, while Falcon-512 performs similarly. The validation performance of MAYO-2 is better than ECDSA-P256 and comparable to RSA-1280, whereas Falcon-512 is 0.3 times slower than ECDSA-P256. These results suggest that DNSSEC signing with MAYO-2 and Falcon-512 is feasible for TLD operators. However, Falcon-512 generates larger signature size and MAYO-2 has larger public keys. These drawbacks should be studied further to assess their operational impact on the validation side, for example by studying the behavior of DNS resolvers with PQC algorithms.**

***Index Terms*—DNS, DNSSEC, post-quantum cryptography, measurements.**

I. Introduction

Quantum computers promise to solve certain problems much more efficiently than today's computers [1], such as designing drugs and understanding the properties of certain molecules [2]. However, they might also break currently deployed cryptographic algorithms in a matter of hours [3], allowing attackers to monitor and change messages. Even though it is presently unknown whether powerful enough quantum computers can be built at all, researchers are making considerable progress in realizing such quantum computers [1]. Therefore, we should consider the scenario in which currently deployed cryptographic algorithms need to be replaced with post-quantum ones [4]. Such so-called *post-quantum cryptography* (PQC) algorithms are currently being developed, and subsequently standardized, by the National Institute of Standards and Technology (NIST).

One of the affected Internet protocols is DNS Security Extensions (DNSSEC) [5], the security extension of the Domain Name System (DNS) [6]. The DNS is a globally and hierarchically decentralized and distributed system that sits at the core of the Internet. Its main function is to translate human-readable domain names (e.g., www.example.nl) to Internet Protocol (IP) addresses (e.g., `2001:db8:42::1`). DNSSEC adds cryptographic signatures to DNS, thereby providing authenticity and integrity guarantees for messages exchanged in the DNS. However, the cryptography currently present in DNSSEC (e.g., RSA-SHA-256) does not provide protection against future attacks with quantum computers. At the same time, we know it takes around 10 years for a new DNSSEC algorithm to reach a significant level of deployment [7], which means that adding PQC algoritms to DNSSEC is an urgent issue today.

The problem, however, is that DNS operators currently have limited insight in the operational consequences of introducing PQC algorithms for DNSSEC on a large scale. This is because the properties of many post-quantum cryptographic (PQC) algorithms are vastly different from currently deployed ones and misalign with some of the requirements of the DNS. For example, the PQC algorithm SLH-DSA [8] uses signatures of 7856 bytes in length, compared to 64 bytes for the pre-quantum ECDSA (algorithm 13 [9]). This is unlike the previous cryptographic algorithm change in DNSSEC, when operators switched from RSA (e.g., algorithm 8) to Elliptic Curve Cryptography (ECC), which reduced operational problems with DNSSEC [10]. Knowing the properties and performance of a cryptographic algorithm within DNSSEC is crucial to establish whether it can replace deployed algorithms, or whether it would break DNS.

This paper aims to empirically bridge this gap by measuring the impact of post-quantum algorithms on the DNSSEC signer setup of top-level domain (TLD) operators. We perform operations on three TLD zone files of different sizes (.nl, .se, and .nu) using an experimental signer setup that we enhanced with PQC support. We empirically compare the properties of a zone file signed with PQC algorithms MAYO-2 [11] and Falcon-512 [12] against current DNSSEC algorithms [13], namely the frequently used RSA-SHA-256 (DNSSEC algorithm 8) and ECDSA-P-256-SHA-256 (DNSSEC algorithm 13). In this study, we focus on the authoritative name server part of DNS operators only.

We make four contributions:

- We patch an open source PQC-enhanced DNSSEC signer to enable validation for use in our experiments (see Section III-D).

978-3-903176-74-4

- We contribute a patch to OQS-bind to add support for MAYO-2 (see Section III-D).
- We measure the performance of PQC algorithms on DNSSEC signer setups for DNS operators running large zone files, such as TLD operators (see Section IV).
- We provide insight in the possible operational impact of using PQC algorithms for DNS operations teams that run authoritative name servers (see Section V).

The remainder of this paper is organized as follows. Section II discusses related work. In Section III, we elaborate on the software, algorithms, and experimental set-up used for our measurements. Section IV shows the results of the measurements, the implications of which are further discussed in Section V. Finally, we present our conclusions and future work in Section VI.

II. Related Work

In 2020, Müller et al. [14] performed a theoretical analysis of the requirements for using post-quantum algorithms in DNSSEC. We use those requirements in our work to select the post-quantum algorithms that look promising for DNSSEC. Cryptographers are working on designing and evaluating post-quantum algorithms, work that is mainly supervised by the NIST competition [15]. NIST has standardized two post-quantum signature algorithms: ML-DSA and SLH-DSA [16]. Unfortunately, those two algorithms do not satisfy the requirements formulated by Müller et al. [14].

In DNSSEC, algorithms have been changed before. Van Rijswijk-Deij et al. [17] analyzed the operational cost of switching from RSA to ECC. On the one hand, ECC provided benefits compared to RSA, such as smaller signature sizes. However, ECC also imposes a higher CPU load on DNS resolvers since ECC signature validation takes longer than RSA signature validation. While their work shows that the increased cost in CPU usage was acceptable, it also shows the need for careful evaluations of new algorithms in DNSSEC.

Furthermore, Goertzen et al. [18] performed Internet measurements using RIPE ATLAS to see what issues arise when using PQC in DNSSEC. They deployed DNS zones with both traditional and PQC algorithms to investigate whether the bigger response sizes when using PQC (due to larger keys and signatures) causes DNS resolution to fail. Their work shows, among other things, that increased response sizes due to PQC indeed negatively impacts succesful DNS message delivery. Our work is complementary because we focus on a different aspect: the impact of using PQC algorithms on DNSSEC signing setups.

Verisign developed Merkle Tree Ladder (MTL) Mode and applied it to DNSSEC [19] to transition DNSSEC to PQC. Using MTL Mode for DNSSEC involves a change to the DNSSEC protocol. This differs from our work where we are investigating using post-quantum algorithms as a *drop-in replacement* for currently deployed algorithms. Our work does not require changes to the DNSSEC protocol.

III. Methodology

We are interested in the performance of DNSSEC signing operations in combination with PQC algorithms compared to the currently used algorithms.

A. DNSSEC Signing Performance Parameters

To determine the DNSSEC performance of each algorithm, we evaluate them on following three performance aspects: (1) zone size (results in Section IV-A); (2) zone signing time (results in Section IV-B); (3) zone validation time (results in Section IV-C).

Zone size and signing time are particularly relevant for TLD operators since they may need to sign large numbers of DNS records. Validation time influences user experience and the willingness of resolver operators to adopt DNSSEC validation, which means it is a concern to TLD operators as well. Furthermore, before publishing a new version of the zone, TLD operators may want to validate the zone they are about to publish.

For the signing and validation time we compare all algorithms to algorithm 13 (ECDSA-P-256-SHA-256). This is our *baseline*, because it is currently the state of the art when it comes to DNSSEC algorithms. For the zone sizes, we use the unsigned zone as baseline for our comparisons. Additionally, we also provide the absolute zone sizes.

Other performance parameters of PQC algorithms are out of scope of this study. For example, both public key size and signature size are relevant during key rollovers, but analysing this requires an in-depth analysis which does not fit within the scope of this study.

B. DNSSEC Signing Variables

There are two methods of signing: offline and online signing. With offline signing, the key management and signing is performed centrally. The entire zone file is signed and then transferred to the name servers so they can serve requests. Typically, the zone file gets signed periodically (e.g., once per hour).

With online signing, the key management and signing is performed by the name server(s), which implies that name server(s) need access to the signing keys. In practice, this means that the DNS answers can be signed when they are requested. This allows both for publishing changes faster and giving dynamic answers, but it does put an extra load on the name servers due to the extra cryptographic operations. Online signing can be used in a distributed approach where each edge node posesses the signing keys and signs answers [20]. Choosing between online and offline signing is a matter of trade-offs. For our experiments, we decided to use the offline approach since it results in clearer insights into the resulting size of the zones and into the time it takes to fully sign them. Furthermore, TLD operators typically use offline signing. For example, the zone file of .nl is signed offline and then distributed to all name server instances, who then serve the 4B DNS requests per day (as per May 2025).

Signing zone files does not only consist of signing existing records, DNSSEC also includes several approaches for proving that a record does not exist [21], namely:

- NSEC — Simplest proof of non-existence for a record.
- NSEC3 — Proof of non-existence that adds additional cryptographic operations to prevent zone enumeration attacks.
- NSEC3 opt-out — Similar to NSEC3; excludes unsigned delegated zones from being signed.

Research shows [22] that when DNSSEC is used, between NSEC (short for *Next Secure*) and NSEC3, NSEC3 is used most by TLD operators and generally by other authoritative DNS operators as well. All three approaches affect the number of signatures that end up in the resulting zone. Since these options vary per DNS operator and since the opt-out flag could compensate for slower signing speeds and large zone sizes, we include these variables into our measurements.

C. Algorithm Selection

We settled on Falcon with parameter set 512 (Falcon-512) and MAYO with parameter set 2 (MAYO-2). Falcon-512 is selected to be standardized by NIST as FN-DSA [23] whereas MAYO-2 is competing in the Round 2 for Additional Signatures call by NIST [24]. Compared to [14], this means we included Falcon-512 as a standard and picked MAYO-2 as a candidate from the Additional Signatures round. Table III summarizes the key characteristics of the algorithms we selected for this study.

To select these post-quantum algorithms, we applied the criteria of Müller et al. [14] to PQShield's list of post-quantum algorithms [25], filtering on a maximum public key size of 65,536 bytes (maximum DNS packet size) and a signature size of max 1,232 bytes (which is the "optimum DNS message size" chosen by the DNS operator community to avoid IP fragmentation and use of TCP[1]). We then excluded algorithms with known vulnerabilities. Next, we looked at the following selection criteria: (1) theoretical verification time, this is important for validating resolvers; (2) signature size, this influences the DNS answer packet length; and (3) library support for the algorithms, in our case liboqs [26] that integrates in OpenSSL through oqs-provider [27], allowing us to use DNS tooling that is deployed in production environments to sign and verify zone files.

To evaluate how PQC algorithms affect DNSSEC signer setups, we compare them to widely used DNSSEC algorithms. According to Section 3.1 in [28], there are two algorithms that are mandatory-to-implement for both signing and validation: RSA-SHA-256 (algorithm 8) and ECDSA-P-256-SHA-256 (algorithm 13) and according to [29] they also happen to be the most deployed. In order to get a better overview of the different algorithms that are deployed by TLDs, we performed a measurement for all TLDs. We downloaded a list of all TLDs from IANA[2] on 20 February 2025. This gave us 1446 TLDs, of which 1356 (∼94%) TLDs contain as least one DNSKEY record. At the time of measurement, we observed 3417 published DNSKEYs, which is described in Table I. The DNSSEC algorithm names and numbers are as described by IANA [13]. We distinguish a Key Signing Key (KSK), which are used to sign Zone Signing Keys (ZSK). A ZSK is then used to sign the actual zone. Cases when TLDs use a Combined Signing Key (CSK) are listed under KSK, since that is how CSKs are published. For algorithm 8, we choose a key size of 1280 bits (RSA-1280). While RSA-1024 is deployed more than RSA-1280 amongst TLDs [30] (see Table II), the use of RSA-1024 is not recommended and we therefore we picked the currently tolerated smallest "next best option" as key size.

Number	Algorithm name	As KSK	As ZSK	Both
5	RSA/SHA-1	2	2	4
7	RSASHA1-NSEC3-SHA1	16	18	34
8	RSA/SHA-256	1278	1728	3006
10	RSA/SHA-512	32	32	64
13	ECDSA-P-256-SHA-256	149	158	307
14	ECDSA-P-384-SHA-384	1	1	2

TABLE I: Summary of observed DNSKEY records for all TLDs on 20 February 2025.

Additionally, we show the public key sizes of the algorithm 8 DNSKEYs in Table II.

RSA key size (bits)	As KSK	As ZSK	Both
1024	2	1183	1185
1280	0	351	351
2048	1256	194	1450
4096	20	0	20

TABLE II: Public key size of the RSA/SHA-256 (algorithm 8) DNSKEYs for TLDs on 20 February 2025.

D. Developing a PQC-Enabled Zone Signer

We found that PQC support for DNS software is limited to PQC modifications for PowerDNS [35, 36] and BIND 9 [37] with liboqs support, with BIND 9 providing tools for offline signing. Support for Falcon-512 and MAYO-2, however, is available through the liboqs library [26]. Towards finishing our study, a series of patches for BIND 9 was published [38] to support several PQC algorithms, but we were unable to include those in our study due to time restrictions.

Therefore, to conduct our experiments, we extended OQS-bind by developing patches for MAYO-2 to OQS-bind [37], so that we can use the BIND tools to sign and verify zone files.

All software was compiled with the defaults and compiler optimizations as provided by the software maintainer. We published our patches as open source[3].

E. Experimental DNSSEC Signing Setup with PQC

For analysis, we picked the .nl, .se, and .nu TLD zones, which are readily available us. We selected zones of different size so our results are relevant for a broad range of TLD operators. Table IV shows the characteristics of the zones we

[1]https://www.dnsflagday.net/2020/#message-size-considerations

[2]https://data.iana.org/TLD/tlds-alpha-by-domain.txt

[3]https://github.com/SIDN/OQS-bind/tree/sidnlabs-pqc

Algorithm	Private key size	Public key size	Sig. size
RSA-1280 [31, 32]	160[a]	160+variable[b]	160
ECDSA-P256 [34]	96	64	64
Falcon-512 [12]	1281[c]	897	666
MAYO-2 [11][d]	24	4912	186

[a] variable, but minimally the *computed* private exponent `d`.
[b] public modulus `n` is fixed length, but the chosen public exponent `e` length is variable, usually $e = 3$ or $e = 65537$ [33].
[c] Falcon's private key size is not documented [12], but can be found in its source code.
[d] Round 1 parameters.

TABLE III: Key characteristics of selected algorithms. Private key size, public key size and signature size are in bytes.

Zone	Size (bytes)	RRsets	NS RRsets	DS RRsets	Serial	Category
.nl	1,174,758,893	10,064,085	6,210,813	3,834,915	2024082916	Large
.se	268,483,576	2,267,055	1,411,206	840,878	2024082910	Medium
.nu	40,902,886	335,624	216,317	118,224	2024082910	Small

TABLE IV: Some key characteristics of the zones.

used in our measurements. The *DS RRsets* column denotes the number of domain names that have DNSSEC enabled. In percentages, this means .nl has 61.7% of its domain names signed, .se 59.6% and .nu 54.6%.

There are many different ways to operate DNSSEC signing in a realistic scenario. For example, TLD operators may use hardware security modules (HSMs) to do the signing, and such hardware may have dedicated hardware implementations to increase performance. Additionally, most validation operations will happen typically at resolvers. For this study, we assume a TLD operator that uses off-the-shelf server hardware to sign its zone files, and validates each zone file before publishing it. Therefore, we opt to use the same setup for both signing and validation.

We account for hardware acceleration based on CPU features, by means of the feature levels of x86-64 architecture provided by QEMU/KVM. At the moment of performing this study, there are four such feature levels: v1, v2, v3, and v4 [39]. The baseline is v1, which is old and we exclude it under the assumption that all processors that are currently in use have either v2 or higher. We include v2 and v3 in our experiments, since they are currently deployed, where v3 brings support for AVX and AVX2. The v4 level is the newest and supports AVX512, but the algorithms that we test do not have optimalisations for this feature level yet, so we do not include it in our experiments.

Our setup consists of a x86-64 virtual machine with 8 virtual CPU cores and 128 GiB of RAM running on top of a Intel(R) Xeon(R) Gold 5115 CPU @ 2.40GHz host system using KVM. The virtual machine is deployed with Ubuntu Linux 24.04 LTS. The experiments run inside a container using podman 4.9.3. The container includes OpenSSL 3.4.0, liboqs[4], and our patched OQS-bind that provides the tools for signing and verifying the zone.

[4] liboqs: commit 26f83d082c01b504140fdbebb6b8651ba4b6f02f, oqsprovider: commit 4db09a9dc540543ff0e22b2713757a7e90e1f0c6

F. Measuring the Signing and Validation Process

For each of the three zones, we calculate the time it takes to sign and to validate it using the algorithms from Section III-C by using the tools from Section III-D. Timings are measured in full seconds, due to the size of the zones we do not need sub-second measurement precision. Reading and parsing the zone files into memory consume a large portion of the signing and validation times in our setup.

We performed the measurements for each combination of experimental variables: TLD, algorithm, signing method (e.g., NSEC). This results in 3 (TLDs) $\times$ 4 (algorithms) $\times$ 4 (signing methods) $=$ 48 experiments. To rule out possible external influences, we run each experiment 16 times. Then, in presenting our results, we compute the mean value and the standard error of these 16 runs.

Since the sample size in our study of 16 is too low to assume a normal distribution of the samples, we analyzed our results using statistical tests that do not assume a normal distribution. We checked for all experimental conditions whether the samples (as ratios) were independent by using a Kruskal-Wallis H-test, and subsequently performed a pairwise Dunn's test to find significant differences ($p < 0.05$), with the p-adjustment of Benjamini–Hochberg procedure. We performed the same tests to test for significant differences ($p < 0.05$) for signing and validation performance in absolute numbers, but then compared results within a single zone only, due to the differences in zone file size.

We plot the increase ratios of all algorithms compared to a baseline (algorithm 13). For brievity and after a statistical test for similarity using a Kruskal-Wallis H-test, we decide to omit the plots for .nu and .se since they show similar ratios as .nl.

For the zone size measurements, we do include all zones, and we simply look at the size (in bytes) of the signed zones.

IV. Results

We show the results of our measurements in three parts: (1) the zone size, which relates to the number of signatures and signature size; (2) the signing time, which relates to the signing performance of algorithms; and (3) the validation time, which relates to the signature verification performance of algorithms.

A. Zone Size

Fig. 1 shows the sizes of the .nl, .se and .nu zone files when the zones are unsigned and when they are signed using the algorithms from Section III-C. The unsigned zones are the smallest, since they lack any DNSSEC signatures. For the signed zones, the zones signed with algorithm 13 (ECDSA-P256) are generally the smallest, then follows MAYO-2, then algorithm 8 (RSA-1280), and finally Falcon-512. For each of the algorithms, we see that the zones using NSEC3 are the largest, when using NSEC the zone is slightly smaller and when using NSEC3 opt-out, the zone is even smaller.

B. Zone Signing Time

When looking at the algorithm signing performance, we show the mean duration in seconds, as well as the standard error,

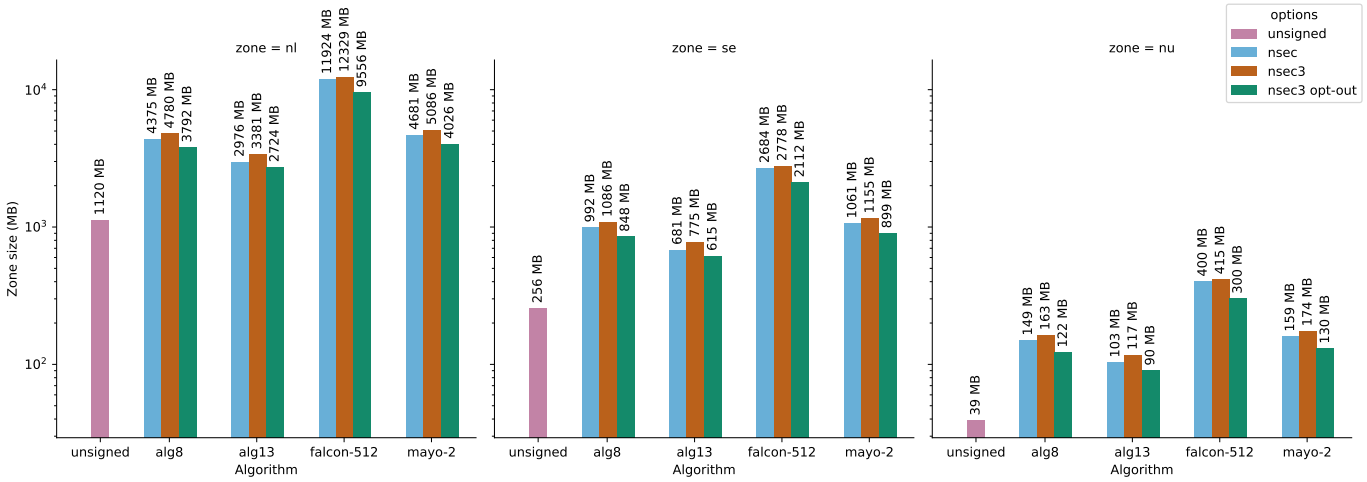

Fig. 1: The size of the .nl, .se and .nu zone files when the zone is unsigned, signed with "traditional" algorithms and signed with PQC algorithms. Note that the y-axis is logarithmic.

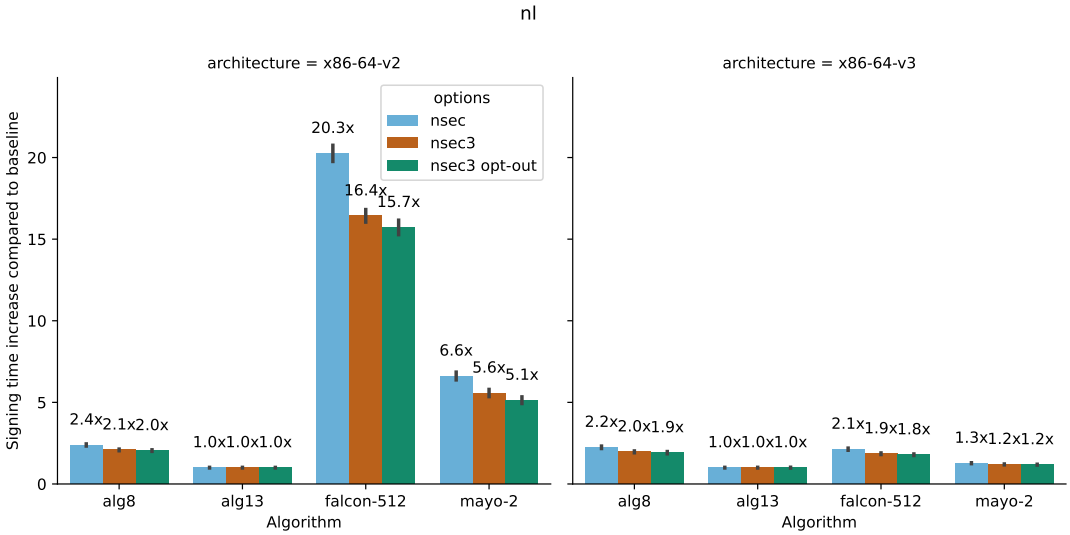

Fig. 2: Increase in signing time for .nl zone compared to the baseline algorithm (number 13, ECDSA-P256). The absolute numbers are shown in Tables V, VI and VII.

of the 16 measurements and for each experimental condition in Table V (for .nl), Table VI (for .se) and Table VII (for .nu). To ease comparing these numbers, Fig. 2 shows for .nl how long it takes to sign the zone compared to the baseline algorithm, algorithm 13 (ECDSA-P256). We compared the ratios for .nl to the other TLDs. For validation, we found only one result where rations of the TLDs differ significantly: MAYO-2 with NSEC. For signing, we found 5 cases where ratios between TLDs differ significantly, namely for RSA-1280 (NSEC, NSEC3, and NSEC3 with opt-out) with a maximum of 0.2 difference and Falcon-512 (NSEC3 and NSEC3 with opt-out) with a maximum difference of 1.2 between the ratios the different TLDs. All other combinations of ratios did not differ in a statistically significant way.

For the x86-64-v2 CPU type and NSEC, we observe that Falcon-512 and MAYO-2 are about 20.3 and 6.6 times slower than the baseline of algorithm 13 (ECDSA-P256), respectively. In comparison, algorithm 8 (RSA-1280) is only 2.4 times slower than the baseline. However, on the more modern x86-64-v3 CPU type, the PQC algorithms perform much better and are aligned with the currently deployed algorithms. Furthermore, on x86-64-v3, no algorithm needs more than 4 times longer than the baseline to sign the zone files.

For the x86-64-v2 CPU type, we find that Falcon-512 is the slowest, and that MAYO-2 is slower than both RSA-1280 ECDSA-P256, yet faster than Falcon-512. Additionally, RSA-1280 is significantly slower than ECDSA-P256. Post-hoc testing using Dunn's shows that these differences were all significant ($p < 0.05$).

For the more modern x86-64-v3 CPU type, we see no

		Sign			Verify		
		NSEC	NSEC3	NSEC3 opt-out	NSEC	NSEC3	NSEC3 opt-out
x86-64-v2	RSA-1280 (alg 8)	1036 (±30.9)	1147 (±30.0)	999 (±23.1)	896 (±12.8)	1018 (±12.6)	816 (±8.8)
	ECDSA-P256 (alg 13)	458 (±9.2)	566 (±11.4)	470 (±7.9)	1600 (±22.9)	1716 (±23.6)	1314 (±16.0)
	Falcon-512	9216 (±233.3)	9394 (±225.5)	7342 (±212.6)	1966 (±29.2)	2062 (±33.9)	1608 (±26.2)
	MAYO-2	2958 (±112.3)	3094 (±129.5)	2272 (±97.8)	9925 (±94.5)	9815 (±90.6)	7582 (±69.4)
x86-64-v3	RSA-1280 (alg 8)	1046 (±37.1)	1166 (±35.3)	932 (±36.4)	970 (±12.1)	1063 (±13.5)	829 (±10.6)
	ECDSA-P256 (alg 13)	479 (±9.8)	595 (±13.3)	494 (±13.1)	1616 (±25.6)	1710 (±26.0)	1333 (±22.2)
	Falcon-512	986 (±32.5)	1102 (±31.2)	880 (±24.9)	2062 (±46.5)	2272 (±39.8)	1753 (±26.1)
	MAYO-2	613 (±14.2)	727 (±17.4)	604 (±14.1)	1064 (±17.7)	1164 (±19.4)	927 (±15.5)

TABLE V: The mean duration (± standard error) in seconds for signing and verifying the .nl zone file.

		Sign			Verify		
		NSEC	NSEC3	NSEC3 opt-out	NSEC	NSEC3	NSEC3 opt-out
x86-64-v2	RSA-1280 (alg 8)	256 (±6.7)	272 (±6.2)	200 (±4.8)	197 (±2.7)	214 (±3.4)	168 (±2.8)
	ECDSA-P256 (alg 13)	101 (±1.3)	124 (±2.1)	101 (±1.8)	358 (±6.4)	377 (±5.5)	287 (±5.3)
	Falcon-512	2080 (±64.6)	2208 (±58.1)	1638 (±52.5)	445 (±8.3)	466 (±7.9)	355 (±6.0)
	MAYO-2	642 (±37.4)	684 (±31.3)	506 (±27.2)	2204 (±20.3)	2164 (±27.3)	1674 (±20.0)
x86-64-v3	RSA-1280 (alg 8)	244 (±11.8)	260 (±10.5)	202 (±7.0)	206 (±4.9)	234 (±3.9)	181 (±2.7)
	ECDSA-P256 (alg 13)	104 (±2.2)	128 (±3.4)	104 (±2.5)	368 (±4.6)	376 (±5.9)	294 (±5.0)
	Falcon-512	236 (±6.6)	286 (±7.3)	200 (±5.7)	453 (±8.0)	476 (±8.8)	364 (±6.7)
	MAYO-2	130 (±3.7)	158 (±4.3)	128 (±3.5)	234 (±3.4)	252 (±3.5)	194 (±3.5)

TABLE VI: The mean duration (± standard error) in seconds for signing and verifying the .se zone file.

		Sign			Verify		
		NSEC	NSEC3	NSEC3 opt-out	NSEC	NSEC3	NSEC3 opt-out
x86-64-v2	RSA-1280 (alg 8)	36 (±1.4)	38 (±1.5)	29 (±1.0)	29 (±0.4)	32 (±0.6)	24 (±0.4)
	ECDSA-P256 (alg 13)	15 (±0.2)	18 (±0.3)	14 (±0.3)	53 (±0.9)	56 (±0.7)	41 (±0.7)
	Falcon-512	312 (±10.0)	326 (±11.7)	222 (±9.7)	66 (±1.2)	70 (±1.2)	50 (±0.9)
	MAYO-2	96 (±5.2)	101 (±4.3)	74 (±3.1)	314 (±3.5)	322 (±4.0)	236 (±2.9)
x86-64-v3	RSA-1280 (alg 8)	36 (±1.8)	40 (±1.8)	30 (±1.3)	30 (±0.8)	33 (±1.0)	25 (±0.6)
	ECDSA-P256 (alg 13)	16 (±0.3)	19 (±0.4)	15 (±0.3)	54 (±1.2)	56 (±1.2)	41 (±0.7)
	Falcon-512	34 (±1.0)	40 (±1.1)	29 (±0.8)	66 (±1.3)	70 (±1.3)	52 (±1.0)
	MAYO-2	21 (±0.6)	24 (±0.7)	20 (±0.5)	35 (±0.6)	37 (±0.7)	28 (±0.5)

TABLE VII: The mean duration (± standard error) in seconds for signing and verifying the .nu zone file.

difference between using for signing Falcon-512 and signing with RSA-1280. They are both approximately 2 times slower than ECDSA-P256 and MAYO-2 who also show similar performance (although Fig. 2 shows that MAYO-2 is a little slower, this difference is not significant).

C. Zone Validation Time

The zone validation measurements, which measure the algorithm verification performance, are shown in the same tables together with the signing results: Table V for .nl, Table VI for .se and Table VII for .nu. Furthermore, we show the ratio relative to the baseline for .nl in Fig. 3. We compared .nl to the other TLDs using a pairwise significance test, but there were no differences between the ratios, so we omit displaying figures for the other TLDs.

Across the TLDs, we see that MAYO-2 is significantly faster on x86-64-v3 compared to x86-64-v2, whereas the other algorithms are similar. ECDSA-P256 and Falcon-512 are equally fast regardless of CPU type. On x86-64-v2, RSA-1280 is the fastest and MAYO-2 is the slowest (being up to 6.1 times slower than the baseline), and each algorithm differs significantly from the others. On x86-64-v3, RSA-1280 is again fastest for verification, with MAYO-2 as runner-up (MAYO-2's improvement over x86-64-v2 is significant). Both RSA-1280 and MAYO-2 are faster than ECDSA-P256 and Falcon-512.

V. Discussion

Firstly, we looked at zone file sizes. When translating this to operational consequences, this means that using the MAYO-2 PQC algorithm provides properties that operators already have operational experience with in terms of zone file sizes. However, for Falcon-512, operators need to quadruple the disk space expectations and available memory in the authoritative name servers. This is important because zone files are often loaded into memory for fast serving.

When analyzing the zone signing time results, a couple of aspects stand out. ECDSA-P256 (algorithm 13) is faster on x86-64-v3 compared to x86-64-v2. DNSSEC algorithm 13 consists of a signing algorithm (ECDSA) and a hashing algorithm (SHA-256) [9]. Between x86-64-v2 and x86-64-v3, OpenSSL

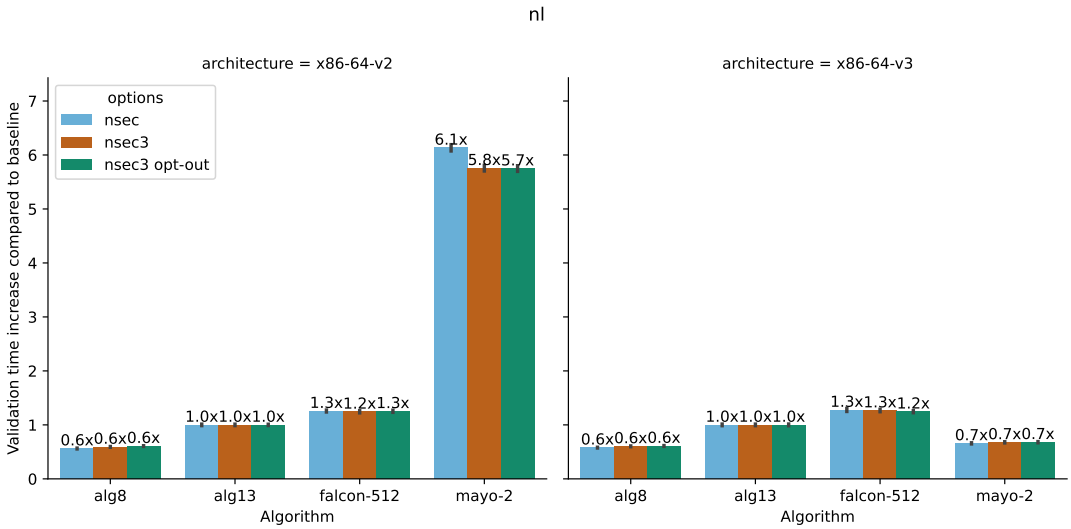

Fig. 3: Increase in validation time for .nl zone compared to the baseline algorithm (number 13, ECDSA-P256). The absolute numbers are shown in Tables V, VI and VII.

provides the same ECDSA implementation, but for SHA-256, an AVX2-accelerated version of SHA-256 is available. Support for the AVX2 instruction set extension that is present in x86-64-v3 and not present in x86-64-v2, which explains the performance improvement between x86-64-v2 and x86-64-v3. The performance difference between x86-64-v2 and x86-64-v3 for MAYO-2 and Falcon-512 is explained similarly. MAYO-2 and Falcon-512 both provide a "generic" implementation that is supposed to run on any CPU, as well as an implementation that accelerates performance by using AVX2. For MAYO-2 and especially Falcon-512, those optimized implementations provide a significant speed increase. That means that for all the PQC algorithms that we tested, signing times are not worse than currently deployed algorithms, provided that the optimized implementations of the signing algorithms can be used. We recommend TLD operators to make sure their signing infrastructure, when running on server hardware, uses CPUs that support at least x86-64-v3. Since we expect that it will take a few years for PQC algorithms to be fully adopted, we expected x86-64-v3 to be -even more- widely supported by the time PQC algorithms are used in-the-wild.

Validating an entire zone file is a common operation for operators to make sure the served zone file is correct. Therefore, any changes in the time it takes to validate the zone file are relevant for operators. The other common usage of DNSSEC signature validations that happen are at resolvers who validate answers from authoritative name servers. However, we do not include the impact on resolvers in this study, since there are many factors that affect the impact, such as DNS request patterns and caching.

The explanation for differences in performance between CPU architectures is similar to the zone signing times, but there are also exceptions. The most important one is that verification of both RSA-1280 and MAYO-2 is faster than the baseline ECDSA-P256. Furthermore, Falcon-512 is slower than the baseline. For MAYO-2, we see an order of magnitude difference between its performance on x86-64-v2 and x86-64-v3, which again can be attributed to MAYO's support for AVX2 on x86-64-v3.

Going back to the operational impact for operators, the most important observation from these numbers is that both investigated PQC algorithms are comparable to, or faster than, the baseline algorithm in terms of validation. Similarly, for signing, Falcon-512 is comparable to RSA-1280, while the performance of MAYO-2 is faster than both, but slower than the baseline ECSA-P256. This means that for TLD operators, the performance impact on both signing and validation is limited, since both PQC algorithms are of comparable performance to the currently most deployed algorithms, provided that hardware acceleration is used (x86-64-v3). The potential drawback of both PQC algorithms is in the larger signature sizes (particularly Falcon-512) and larger public key size (particularly MAYO-2).

The drawbacks of PQC algorithms leads to some potential deployment challenges. Firstly, if DNS answers become larger due to larger public keys (MAYO-2) or larger signatures (Falcon-512), there will be an operational impact other than discussed in this study, on both the resolvers and the authoritative name servers. For example, in terms of increased network traffic, or server load for resolvers and authoritative name servers. This requires further study, for example by simulating a real-world workload and measuring the effects of larger DNS messages (which means more use of TCP) on DNS resolvers and on the network. Another deployment challenge is the lack of urgency for DNSSEC. Since there are currently no practical quantum computers able to break existing cryptography, many people do not see any urgency for DNSSEC to switch to PQC. In particular, since DNSSEC signatures are relatively short-lived (in the order of days or weeks), a potential quantum computer

would need to be very powerful to be able to crack existing algorithms before TLD operators rotate their keys and generate new signatures. A possible short-term mitigation against a less powerful quantum computer could be to generate a new Zone Signing Key (ZSK) every 1-2 weeks, and shorten the lifetime of DNSSEC signatures. Whereas this is merely a short-term patch, it does indicate that the necessity of using PQC in DNSSEC has a lower in priority than, for example, encryption, where there is a store-now-decrypt-later threat *right now*. A counter-argument against rotating ZSKs is that this may shift the attack vector to the Key Signing Key (KSK), which are less often rotated. Finally, any change in DNSSEC may trigger people who oppose DNSSEC to advocate for disabling DNSSEC as being too complex, or not needed. In particular for TLDs where the adoption of DNSSEC is low [29], a challenge to the mere existence of DNSSEC could potentially delay deployment of PQC DNSSEC.

There will be tradeoffs when it comes to chosing which PQC algorithm to deploy for DNSSEC. These will depend on, at least, zone size, number of signed domains, and role of the zone in the DNS hierarchy. To give an example, the root zone is relatively small and very often used, but also arguably the most-cached entry in DNS resolvers. The root having a slower validation time would therefore have less impact on DNS resolvers. Further study should be conducted on these tradeoffs.

A limitation of this work is that the performance results for the PQC algorithms are likely to change in the future. The PQC algorithms are not standardized yet and as such may undergo changes, for instance in response to cryptanalysis. Furthermore, the implementations of the PQC algorithms are likely to still be optimized, for instance by using specialized CPU instructions. As a result, the time it takes to for a signing or verification operation may increase or decrease. It is even possible that key or signature sizes will be adapted. Therefore, it we recommend to measure again both closer to and after standardization.

VI. Conclusions and Future Work

In this work, we present an initial look at one particular aspect of using new algorithms in DNSSEC: namely, the impact on the *signer setup* of TLD operators. With the caveat that the performance of the PQC algorithms is still likely to change, our results show that the currently unstandardized MAYO-2 performs comparably to algorithm 13 (ECDSA-P256) from the perspective of a DNSSEC zone signer setup, which most DNS operators have extensive operational experience with. Furthermore, MAYO-2 has much larger public keys, which is a drawback which requires sharing public keys (DNSKEY) over TCP instead of the default UDP. We also showed that the soon-to-be-standardized Falcon-512 algorithm is comparable to RSA-1280 (algorithm 8) in terms of needed computational power, but zones signed with Falcon-512 are considerably bigger than zones signed with algorithm 8, which requires authoritative name servers to be provisioned appropriately in terms of disk space and memory usage. For DNS operators, this could mean longer zone signing times (also for the incremental deltas), and more resource usage due to larger zone files and increased computational requirements.

When using modern hardware that has CPU support for AVX2, the impact of PQC is rather limited in terms of signing and validation performance. We see no problems adopting MAYO-2 and Falcon-512 in terms of performance. The drawbacks in terms of larger key size (MAYO-2) and larger signatures (Falcon-512) requires further study to assess the practical impact.

For future work, we would like to analyze the behavior of DNS resolvers when using PQC different algorithms, as well as to evaluate the performance of other (non-standardized) PQC algorithms. We would also like to evaluate the impact of PQC algorithms on authoritative name servers under production loads. For example, we want to analyze how often resolvers have to resort to a TCP connection when DNS answers are signed with PQC algorithms due to the responses not fitting in a single UDP packet.

References

1. Z. Montague, "The race to save our secrets from the computers of the future," *The New York Times*, 2023. [Online]. Available: https://www.nytimes.com/2023/10/22/us/politics/quantum-computing-encryption.html
2. R. de Wolf, "The potential impact of quantum computers on society," *Ethics and Information Technology*, vol. 19, no. 4, pp. 271–276, Sep. 2017. [Online]. Available: http://dx.doi.org/10.1007/s10676-017-9439-z
3. E. Grumbling and M. Horowitz, Eds., *Quantum Computing: Progress and Prospects*. National Academies Press, Mar. 2019.
4. D. J. Bernstein and T. Lange, "Post-quantum cryptography," *Nature*, vol. 549, no. 7671, pp. 188–194, Sep. 2017.
5. S. Rose, M. Larson, D. Massey, R. Austein, and R. Arends, "DNS Security Introduction and Requirements," RFC 4033, Mar. 2005.
6. P. Mockapetris, "Domain names - concepts and facilities," RFC 1034, Nov. 1987.
7. M. Müller, W. Toorop, T. Chung, J. Jansen, and R. van Rijswijk-Deij, "The reality of algorithm agility: Studying the dnssec algorithm life-cycle," in *Proceedings of the ACM Internet Measurement Conference*, ser. IMC 2020, vol. 18. ACM, Oct. 2020, pp. 295–308.
8. National Institute of Standards and Technology, *Stateless Hash-Based Digital Signature Standard*, Aug. 2024. [Online]. Available: http://dx.doi.org/10.6028/NIST.FIPS.205
9. P. E. Hoffman and W. Wijngaards, "Elliptic Curve Digital Signature Algorithm (DSA) for DNSSEC," RFC 6605, Apr. 2012.
10. R. van Rijswijk-Deij, A. Sperotto, and A. Pras, "Making the Case for Elliptic Curves in DNSSEC," *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 5, pp. 13–19, Sep. 2015. [Online]. Available: http://dx.doi.org/10.1145/2831347.2831350

11. W. Beullens, F. Campos, S. Celi, B. Hess, and M. Kannwischer, "MAYO specification," Sep 2023. [Online]. Available: https://pqmayo.org/assets/specs/mayo.pdf
12. Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Prest, Thomas Ricosset, Gregor Seiler, William Whyte, and Zhenfei Zhang, "Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU," Oct 2020. [Online]. Available: https://falcon-sign.info/falcon.pdf
13. Internet Assigned Numbers Authority, "Domain Name System Security (DNSSEC) Algorithm Numbers," accessed March 7, 2025. [Online]. Available: https://www.iana.org/assignments/dns-sec-alg-numbers/dns-sec-alg-numbers.xhtml
14. M. Müller, J. de Jong, M. van Heesch, B. Overeinder, and R. van Rijswijk-Deij, "Retrofitting post-quantum cryptography in internet protocols: a case study of DNSSEC," *ACM SIGCOMM Computer Communication Review*, vol. 50, no. 4, pp. 49–57, Oct. 2020.
15. NIST, "Post-quantum cryptography," 2024. [Online]. Available: https://csrc.nist.gov/projects/post-quantum-cryptography
16. ——, "Announcing Approval of Three Federal Information Processing Standards (FIPS) for Post-Quantum Cryptography," 2024. [Online]. Available: https://csrc.nist.gov/News/2024/postquantum-cryptography-fips-approved
17. R. van Rijswijk-Deij, K. Hageman, A. Sperotto, and A. Pras, "The Performance Impact of Elliptic Curve Cryptography on DNSSEC Validation," *IEEE/ACM Transactions on Networking*, vol. 25, no. 2, pp. 738–750, Apr. 2017. [Online]. Available: http://dx.doi.org/10.1109/TNET.2016.2605767
18. J. Goertzen, D. Joseph, and P. Thomassen, "More PQC in PowerDNS: A DNSSEC field study," *PowerDNS blog*, 2024. [Online]. Available: https://blog.powerdns.com/2024/07/15/more-pqc-in-powerdns-a-dnssec-field-study
19. A. Fregly, J. Harvey, B. S. Kaliski Jr, and S. Sheth, "Merkle tree ladder mode: reducing the size impact of nist pqc signature algorithms in practice," in *Cryptographers' Track at the RSA Conference*. Springer, 2023, pp. 415–441.
20. Ó. Guðmundsson, "DNSSEC Done Right," Jan 2015. [Online]. Available: https://blog.cloudflare.com/dnssec-done-right/
21. R. M. Gieben and M. Mekking, "Authenticated Denial of Existence in the DNS," RFC 7129, Feb. 2014. [Online]. Available: https://www.rfc-editor.org/info/rfc7129
22. C. Daniluk, "Zeros are heroes: NSEC3 parameter settings in the wild," 2025. [Online]. Available: https://blog.apnic.net/2024/12/12/zeros-are-heroes-nsec3-parameter-settings-in-the-wild/
23. NIST, "NIST Releases First 3 Finalized Post-Quantum Encryption Standards," *NIST*, Aug. 2024, last Modified: 2024-08-26T13:03-04:00. [Online]. Available: https://www.nist.gov/news-events/news/2024/08/nist-releases-first-3-finalized-post-quantum-encryption-standards
24. I. T. L. Computer Security Division, "Round 2 Additional Signatures - Post-Quantum Cryptography: Digital Signature Schemes | CSRC | CSRC," Mar. 2025. [Online]. Available: https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures
25. PQShield research team, "Post-Quantum signatures zoo," Oct 2024. [Online]. Available: https://pqshield.github.io/nist-sigs-zoo/
26. Open Quantum Safe, "liboqs," Mar 2025. [Online]. Available: https://openquantumsafe.org/liboqs/
27. ——, "oqs-provider," Mar 2025. [Online]. Available: https://github.com/open-quantum-safe/oqs-provider
28. P. Wouters and O. Surý, "Algorithm Implementation Requirements and Usage Guidance for DNSSEC," Internet Engineering Task Force, Request for Comments RFC 8624, Jun. 2019, num Pages: 11. [Online]. Available: https://datatracker.ietf.org/doc/rfc8624
29. R. Lamb, "DNSSEC Deployment Report," Oct 2025. [Online]. Available: https://rick.eng.br/dnssecstat/
30. Viktor Dukhovni, "DNSSEC algorithms for TLDs (and everyone else)," 2022. [Online]. Available: https://blog.apnic.net/2022/03/24/dnssec-algorithms-for-tlds-and-everyone-else/
31. J. Jonsson and B. Kaliski, "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1," RFC 3447, Feb. 2003. [Online]. Available: https://www.rfc-editor.org/info/rfc3447
32. D. Eastlake 3rd, "RSA/SHA-1 SIGs and RSA KEYs in the Domain Name System (DNS)," RFC 3110, May 2001. [Online]. Available: https://www.rfc-editor.org/info/rfc3110
33. D. Boneh and H. Shacham, "Fast variants of rsa," *CryptoBytes*, vol. 5, no. 1, pp. 1–9, 2002.
34. D. J. Bernstein and T. L. (editors), "eBACS: ECRYPT Benchmarking of Cryptographic Systems," accessed February 19, 2025. [Online]. Available: https://bench.cr.yp.to/results-sign.html
35. SIDN Labs, "pqc-auth-powerdns," Mar 2025. [Online]. Available: https://github.com/SIDN/pqc-auth-powerdns
36. Nils Wisiol, "pdns." [Online]. Available: https://github.com/nils-wisiol/pdns
37. OQS-Bind, "OQS-Bind," Mar 2025. [Online]. Available: https://github.com/desec-io/OQS-bind
38. O. Sury, "PQC for DNSSEC," Mar 2025. [Online]. Available: https://github.com/IQTF/pq-dnssec-materials/blob/refs/heads/main/IETF122/Sur%C3%BD_PQC_for_DNSSEC.pdf
39. "x86 psABIs / x86-64 psABI · GitLab," Jun. 2024. [Online]. Available: https://gitlab.com/x86-psABIs/x86-64-ABI

Ethical Considerations

TLD zone files potentially contain personal information, such as names of individuals that registered their own name as domain name. Therefore, we treat all zone files as personal information and use necessary measures accordingly. All

authors have access to the (non-public) .nl zone file as part of their job, working for the operator of .nl. For the .se and .nu TLDs: both are distributed publicly by their operator The Swedish Internet Foundation under a Creative Commons Attribution 4.0 International license.

ACKNOWLEDGEMENTS

This research is part of the SHARQS project, which received funding from the Dutch Research Council (NWO). We would like to thank Rian Hagebeuk for her helpful suggestions.