Securing Post-Quantum DNSSEC Against Fragmentation Mis-Association Threat

Manohar Raavi
Department of Computer Science
Kennesaw State University
mraavi3@kennesaw.edu

Simeon Wuthier
Department of Computer Science
University of Colorado, Colorado Springs
swuthier@uccs.edu

Sang-Yoon Chang
Department of Computer Science
University of Colorado, Colorado Springs
schang2@uccs.edu

Abstract*—Domain Name System Security Extensions (DNSSEC) uses public-key digital signatures to provide integrity and authentication for DNS query responses. The current standardized DNS for reliable UDP delivery limits DNS response (including the message, signature, and public key) to a maximum of 1232 bytes. Incorporating NIST's post-quantum digital signatures into the DNS protocol results in a response size that exceeds the limit set by the Ethernet standardization, making PQC incompatible with the current standardized DNS. To address the incompatibility and enable PQC to protect the authenticity against the quantum-equipped adversaries, previous research proposed fragmenting the DNSSEC messages. Fragmentation however exposes DNSSEC to Fragmentation Mis-Association threat, traditionally studied in the broader IP fragmentation contexts and not applicable in the current DNSSEC with classical/pre-quantum cipher (no fragmentation needed). We distinguish our work from the previous research incorporating PQC to DNSSEC to defend against the Fragmentation Mis-Association Threat by chaining the fragments and applying cryptographic commit-and-reveal. We also advance the previous research and further reduce the number of packet fragments, which can be particularly useful as the DNSSEC based on UDP is prone to packet transmission failure increasing the chance of the DNS response failure when sent in multiple fragments, by using blockchain to offload and enable the offline delivery of the public key. Our scheme thus even allows the Falcon-512 PQC cipher incorporation to forgo the fragmentation, in contrast to the previous research requiring fragmentation for Falcon-512; the other PQC ciphers, i.e., Dilithium ciphers and Falcon-1024, still require fragmentation in our scheme due to the standardized signature sizes. We implement our scheme and analyze the effectiveness and performances through experimentation.

I. Introduction

The Domain Name System (DNS) is a major foundation and enabler of the internet infrastructure and internet traffic routing that resolves user-friendly domain names into numerical IP addresses. To protect the authenticity and integrity of DNS, DNSSEC uses public-key digital signature, e.g., RSA and ECDSA. The security of digital signature algorithms relies on the underlying computational hard problems such as integer factorization for RSA and discrete logarithms for ECDSA. Shor's algorithm [1] assuming quantum-computing attacker can solve both integer factorization and discrete logarithm problems in polynomial time. Recent advancements in quantum computing [2], [3] and the availability of Shor's algorithm threaten the security of digital communications and develop a need to transition to quantum-resistant cryptography.

National Institute of Standards and Technology (NIST) started the post-quantum cryptography (PQC) standardization project that includes identifying new computational hard problems and developing quantum-resistant digital signature algorithms. In July 2022 [4], NIST announced the selection of three new digital signature algorithms for standardization including Dilithium [5], Falcon [6], and SPHINCS$^+$ [7].

Transitioning to the use of the PQC mitigates the quantum risks and it is a requirement for the United States Executive departments and agencies to migrate to PQC by 2035[1] [8]. All of the digital networking protocols, including DNSSEC, that require message integrity and source authentication are expected to transition to the use of post-quantum digital signature algorithms. Cryptographic algorithms are often designed to be modular in principle and PQC can be drop-in replacements with no standardized protocol changes in some digital security applications, including TLS [9], [10], [11], [12], QUIC [11], [12], SSH [9], and X.509 PKI [13].

Unlike for some other digital security protocols, the PQC transition for DNSSEC requires a change in the DNS protocol; Section VI describes the research literature discovering the incompatibility of PQC with the current DNS in greater detail. DNS builds on UDP and has a maximum response size limit of 1232 bytes [14] to fit in one UDP packet. Any response larger than 1232 bytes needs fragmentation and UDP fragmentation is unreliable [15]. A TCP session needs to be established to deliver the DNS response bigger than the 1232 bytes limit. However, 10-11% of the DNS implementations do not support TCP and cannot handle fragmentation [14], [15]. From Table I, one can observe that delivering a response with the message, public key, and signature exceeds the DNS 1232 byte response size limit, necessitating fragmentation, for all of the post-quantum digital signature algorithms.

The DNSSEC fragmentation opens up a new vulnerability for Fragmentation Mis-Association Threat, while the current DNSSEC without post-quantum defense does not need

[1]The Internet and web are generally slow in adopting major foundational changes, for example, the IPv4 to IPv6 transition.

TABLE I: Digital signature key and signature sizes

Algorithm	Public-Key Size (B)	Signature Size (B)	Public Key + Signature (B)
RSA 2048	256	256	512
ECDSA - Secp256k1	64	48	112
Dilithium 2	1312	2420	3732
Dilithium 3	1952	3293	5245
Dilithium 5	2592	4595	7187
Falcon 512	897	666	1563
Falcon 1024	1793	1280	3073
SPHINCS$^+$-128s	64	7856	7920
SPHINCS$^+$-192s	96	16224	16320
SPHINCS$^+$-256s	128	29792	29920

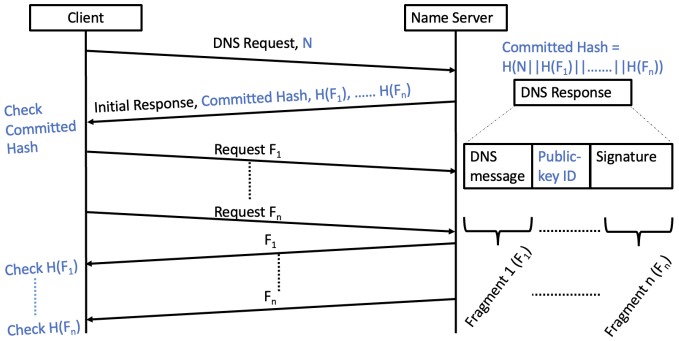

Fig. 1: DNS fragmentation to support post-quantum DNSSEC. State-of-the-art research is in black color, while our scheme additions are in blue color. Also, the state-of-the-art uses the public key itself instead of the public key ID in the DNS Response shown in the right

fragmentation. The Fragmentation Mis-Association Threat, in which the adversary injects a mal-crafted fragment to disrupt and disable the overall response, has traditionally been studied in the broader IP networking, e.g. [16], [17]. To the best of our knowledge, we are the first to study the threat in DNSSEC.

Our work is distinguishable from the previous state-of-the-art research in two ways. First, our scheme defends DNSSEC from Fragmentation Mis-Association attacks by using the commit and reveal technique and, second, it also offloads the public key distribution through blockchain for performance efficiency and reliability. Our scheme provides early detection of fragment spoofing by verifying the hashes of individual fragments. Our scheme also increases the attacker load in that the attacker needs to perform real-time hash computations to attempt to spoof the packets. We use an Ethereum blockchain as a trusted repository to deliver the public keys of reputable DNS servers before the DNS request is sent. This effectively reduces the DNS response to send only the message and signature.

Contributions We design and build a scheme for efficient fragmentation in DNSSEC for post-quantum authentication. Our scheme uses the commit and reveal technique to defend against Fragmentation Mis-Association Threats and verify the hashes of individual fragments to avoid any spoofing attempts. Our scheme uses an Ethereum blockchain, as a trusted public key repository, to offload the public keys from the DNSSEC and reduce the attack surface. We implement and experimentally compare our scheme with the state-of-the-art research for post-quantum DNSSEC.

Organization The rest of the paper is organized as follows: Section II provides the background on DNS and DNSSEC, NIST PQC standardization project, and the use of blockchain as a public key repository. Section III describes the threat model while Section IV provides our scheme design. In Section V, we analyze our scheme's performance. We discuss the related work in Section VI and conclude our work in Section VII.

II. Background

A. Domain Name System (DNS) and DNSSEC

Initially, DNS standards were developed without any authentication or integrity protection and the response size was limited to 512 bytes. DNSSEC was introduced to address security concerns while Extension Mechanisms were introduced to increase the response size to 64 KB. However, DNS uses UDP at the transport layer and has a recommended response size of 1232 bytes to avoid unreliable UDP fragmentation [14]. For response sizes below 1232 bytes, a DNS query receives the entire response in one UDP packet, and for responses bigger than 1232 bytes, DNS fragmentation is needed where one query results in a response with multiple fragments. This attracts the attackers to target DNS for implementing amplification, spoofing, and DDoS attacks.

The state-of-the-art research scheme [18] for DNS Fragmentation, shown in black color of Figure 1, addresses the amplification attacks by requiring the client to request each fragment individually and keeping the response limited. After receiving the DNS request, the server sends an initial response with the resource record information, and the client then subsequently requests individual chunks of up to 1232 bytes to receive the entire DNS response. The client doesn't have a way of verifying these individual fragments till it receives all the fragments and does the signature verification. Any successful injection attack targeting one of the fragments would result in a failure of signature verification and the client needs to repeat the DNS request.

B. NIST's PQC Standardization

NIST PQC standardization process has been a multi-year process developing quantum-resistant cryptography [19]. After three rounds of analysis, NIST announced the selection of lattice-based (Dilithium and Falcon) and hash-based computational problems (SPHINCS$^+$) for the standardization of new digital signature algorithms. Each of these algorithms has different parameter sets, as listed in Table I, targeting different security categories. For more information on the parameter sets and their security strengths, we refer the readers to the

NIST PQC documentation [4]. Both lattice-based and hash-based algorithms have significantly bigger signature or public key sizes compared to their classical counterparts. NIST furthermore began a new search for more efficient digital signature algorithms [20]. Currently, there are 40 new candidates under analysis that are based on computational problems including code-based, isogeny-based, MPC-in-the-Head, multivariate, symmetric-based, and also more lattice-based [21].

Among the NIST-selected PQC ciphers, we focus on the lattice-based PQC ciphers of Dilithium and Falcon families rather than the hash-based SPHINCS$^+$ family for the following reasons. First, the lattice-based schemes of Dilithium and Falcon are better analyzed, for example, they were the finalists in the previous Round 3 of NIST's PQC cipher standardization process while SPHINCS$^+$ was not and has been included as an alternative candidate in Round 3 for diversity of the PQC cipher families. Second, SPHINCS$^+$ requires a significantly larger signature size than Dilithium and Falcon; as a result, SPHINCS$^+$-128s with the shortest key length/security strength in the SPHINCS$^+$ family requires greater fragmentations in DNSSEC than any other ciphers in the Dilithium or Falcon families, including those with greater security strength/brute-force resistance than SPHINCS$^+$-128s.

C. Blockchain as Public Key Repository

To address the challenges posed by the larger sizes of post-quantum digital signature algorithms in DNSSEC, an efficient storage solution is needed. The Ethereum blockchain, known for its decentralized architecture, presents a viable solution. Notably, there is an increasing recognition of Ethereum's capability to implement Public Key Infrastructures (PKIs) as smart contracts [22], [23], [24]. Ethereum's smart contracts ensure a secure storage of public keys. The blockchain preserves the authenticity and integrity of stored keys, safeguarding them against unauthorized changes. This method significantly reduces the DNS response size since the public key can be retrieved from the blockchain. Only the message and signature need transmission in the DNS response. Thus, leveraging the Ethereum blockchain aligns DNSSEC with contemporary security requirements, while maintaining integrity and reliability.

III. Threat Model and Fragmentation Mis-association Threat

A. Threat Model

We consider an adversary that has compromised the network and has both the monitoring capability (passive) and the injection and spoofing capabilities (active) to compromise the authenticity and integrity of DNSSEC. To consider the worse-case threat, an attacker can monitor and capture the communication packets in Figure 1 for its passive capability. For its active capabilities, the attacker can inject packets and spoof as other entities. These capabilities enable the fragmentation misassociation threat. The adversary also has the capability to inject newly crafted packets for sending DNSSEC packets with bogus public keys (and the signatures and the rest of the packets computed using those keys).

We focus on the DNSSEC integrity and authenticity and therefore do not consider an adversary against the networking availability. The attacker cannot disable legitimate communications and the packets from the client reach the DNS server and vice versa. The adversary injections can delay or incur additional computations but do not completely disable legitimate communications.

B. Fragmentation Mis-Association Threat

An attacker passively monitoring the DNS query and response can quickly learn from the initial response that DNSSEC with PQC needs fragmentation for integrity and authentication of the response. Fragmentation is required due to the bigger signature sizes of the PQC digital signature algorithms. An attacker can inject/spoof a tiny fragment to the client which will result in the wrong reassembly of fragments and packet loss [16]. To inject/spoof and target fragment mis-association, an attacker needs to know only the IP value which a passive attacker can learn. A Fragmentation Mis-Association Threat can be defined as the potential danger of an attacker injecting/spoofing one tiny fragment to cause wrong reassembly, packet loss, and validation failure. An attacker who targets DNSSEC Fragmentation Mis-Association Threat can spoof one tiny fragment to the client which after reassembly will cause signature verification failure. This results in DNS query resolution failure and the client not being able to communicate with internet services.

IV. Our Scheme

A. Commit and Reveal Application

We implement the commit and reveal technique in delivering the DNS response. On receiving a DNS request, the server creates a DNS response with the DNS message and generates the signature. For responses bigger than 1232 bytes, the server calculates the number of fragments needed to be delivered. Based on the post-quantum digital signature algorithm used and its signature size, the number of fragments varies from 1 to n. We use F_n to represent the n^{th} fragment. The server now calculates the hashes for each of the fragments as H(F_1), H(F_2), ..., and H(F_n). After obtaining all the hashes for the fragments, the server commits to the response for the client's request by generating a committed hash as H(N$\|$H(F_1)$\|$H(F_2)$\|$.....$\|$H(F_n)) using the nonce 'N' and individual fragment hashes. As shown in Figure 1 (our scheme additions in blue color), the server's initial response delivery includes these committed hash and individual fragment hashes.

B. Query Response Delivery

The client on receiving the initial response calculates the committed hash using the nonce 'N' it used in the DNS request and the hashes of the individual fragments it obtained from the initial response. If the calculated hash matches with the

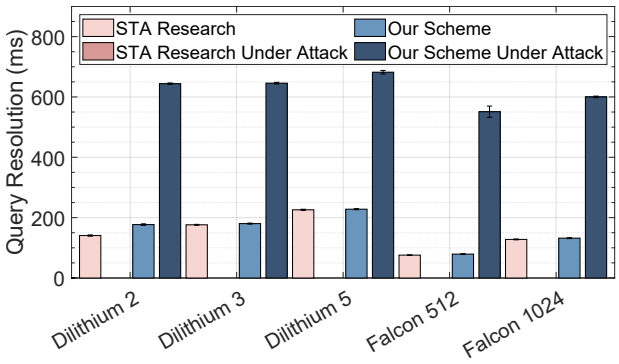

Fig. 2: DNS query resolution times for the state-of-the-art (STA) research and our scheme, including the STA under attack case where the query never resolves (thus the bars not drawn)

committed hash it received, the client stores all the individual fragment hashes. The client then requests the individual fragments and checks the hash of the received fragment with the stored hashes. For example, after receiving fragment F_1, the client computes H(F_1) and compares it to the stored hash of F_1. Only if they match, the fragment is accepted. As shown in Figure 1, the server responds to each of the client's fragment request to send one fragment at a time, and each fragment needs a successful hash verification before the client accepts the overall DNS resolution response, defending against Fragmentation Mis-Association Threat.

C. Signature Validation

Our scheme offloads the public key delivery from the DNS response by maintaining an Ethereum blockchain on the client. The blockchain acts as a trusted repository for the public keys of the deployed DNS servers that are publicly accessible. However, writing to the blockchain is permissioned, and, while primarily the DNS root servers have this authority, other trusted entities could also be granted this capability.

Each of the public keys is associated with an ID (public key ID) that the DNS response refers to indicate the public key of the server. As shown in Figure 1, our scheme uses a public key ID in DNS responses. The size of the public key ID is directly proportional to the number of DNS servers. For our experiment, we use public key IDs of 8 bytes or 64 bits accommodating up to 2^{64} servers. The client can use the public key ID from the DNS response to identify the public key from the blockchain and use it for signature verification. Signature verification confirms the integrity and authenticity of the DNS message and the resolved IP information is used only after a successful signature verification.

V. PQC Algorithm Performance Evaluation

A. Experimentation

We implement the post-quantum digital signature algorithms using Open-Quantum-Safe liboqs [25] and use a Redis server as a database. We host a server on the Google Cloud Platform to serve as a DNS server. Both the client and the server are loaded with liboqs. We collect results for DNS connections on the client end for 100 samples. We evaluate our scheme's response (including the attack scenario) and compare it with the state-of-the-art (STA) scheme. The cryptographic algorithms we test include classical, such as RSA and ECDSA, as well as NIST standardized post-quantum families: Dilithium and Falcon.

TABLE II: Number of fragments needed.

Algorithm	**Number of Fragments (n)**	
	State-of-the-art	Our Scheme
RSA 2048	1	-
ECDSA	1	-
Dilithium 2	4	2
Dilithium 3	5	3
Dilithium 5	6	4
Falcon 512	2	1
Falcon 1024	3	2

B. DNS Query Resolution Performance

We analyze the performance using query resolution time which is the time duration at the client since sending the DNS request to receiving and successfully verifying the signature of the response. Figure 2 plots the query resolution time of the state-of-the-art scheme and our scheme. For under attack scenario, the Fragmentation Mis-Association Threat is being targeted by the attacker by blindly spoofing fragments to the client. As we discussed before, spoofed fragments received and reassembled at the client will result in signature verification failure and hence the query could not be resolved. For this reason, the plot doesn't have a bar indicating that the query could never be resolved for the STA scheme. In contrast, our scheme can perform DNS query resolution even under attack. Under attack, the query resolution takes 4.18 and 3.92 times longer on average compared to STA and our scheme under no attack, respectively. This is due to the client processing fragments from both the attacker and the server to check against the stored hash value. The impact of the attacker is clearly seen and cannot be completely removed as the client needs to process the fragment before it can make a decision to accept or reject it. As a result, the query resolution time is most impacted by the networking latency from requiring additional round trips. For the scenario without attack, our scheme introduces a slight overhead in terms of query resolution time due to the additional hash computation required in the commit-and-reveal application. For example, our scheme is 2.12 milliseconds or 0.94% slower compared to the STA scheme when using Dilithium 5 and 4.24 milliseconds or 3.31% slower when using Falcon 1024. Although query resolution time can be influenced by latency during attacks, the variations highlighted here arise from computational overhead.

Table II shows the number of fragments required to deliver a DNSSEC response when using classical and post-quantum

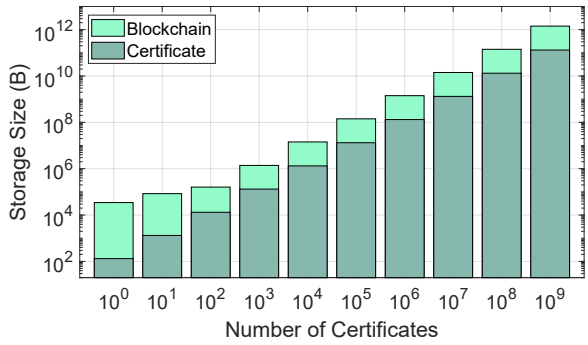

Fig. 3: Blockchain size overhead, including certificate, when varying the number of certificates stored in the blockchain.

digital signature algorithms. Our scheme is not applicable to classical RSA and ECDSA algorithms as they fit in the 1232-byte payload limit. Offloading the public key to a blockchain in our scheme reduces the number of fragments required by a minimum of 33.3% and a maximum of 50% in DNS query resolution. For example, both Dilithium 2 and Falcon 512 need 50% less number of packets in our scheme compared to STA.

C. Ethereum Blockchain

As part of our scheme's approach to public key management using Ethereum, Figure 3 presents the Ethereum blockchain size in correlation with the number of stored certificates. For a singular certificate (10^0), the blockchain size is around 34,680 bytes, accounting for standard blockchain overheads. Notably, the size of an individual certificate is a fixed size, irrespective of the total number stored. This indicates that the overhead associated with the Ethereum blockchain is comparatively small, particularly when fewer certificates are stored. For instance, with 10^3 certificates totaling 132,000 bytes, the blockchain's overall size is approximately 1.39 megabytes. The data suggests that the additional overhead, arising from Ethereum-specific elements such as block and transaction hashes or timestamps, is minimal. Ethereum's storage efficiency makes it a compelling option for DNSSEC implementations.

VI. Related Work

Post-quantum digital signature has been studied broadly in digital networking, including those focusing on the overheads of the PQC cipher algorithms themselves (e.g., [26], [27]) and on the integrations and applications to popular digital security protocols, such as TLS (e.g., [9], [10], [11], [12]), QUIC (e.g., [11], [12]), SSH (e.g., [9]), and X.509 PKI (e.g., [13]). However, this section focuses on the previous research studying PQC in DNSSEC and treats them in greater details. In addition, Section VI-C describes the fragmentation security issues because of our research contribution focus to advance the defense against the vulnerabilities and threats from fragmentation.

A. PQC for DNSSEC Requiring Fragmentation

Müller et al. [14] studied the compatibility of Post-Quantum Cryptography (PQC) with DNS and DNSSEC. Their study identified the requirements of DNSSEC as small signature sizes and fast validation times. Their findings show that DNSSEC would need protocol-level changes to fully support PQC.

The increased signature size in PQC introduced fragmentation challenges. To address this, Goertzen et al. [15] developed A Resource Record Fragmentation (ARRF) method, which segmented DNS resource records at the application layer. When tested with specific algorithms like Falcon and Dilithium, the method led to reduced resolution times and data transmission. Goertzen's master's thesis [18] provided a detailed analysis of the ARRF mechanism and its practical benefits. Our work is built directly upon the foundation set by the ARRF method.

B. Alternative PQC Integration Strategies for DNSSEC

Shrishak et al. [28] introduced the idea of cipher-suite negotiation in DNSSEC. This approach allowed servers and resolvers to dynamically choose the best cryptographic algorithm, aiming to ease the transition to post-quantum signatures and reduce packet fragmentation issues.

Muller et al. [14] first proposed the idea of transmitting the public key out-of-band through an HTTP web server. Beernink's thesis [29], built on the out-of-band key transfer idea, proposes a key exchange method for PQC in DNSSEC. Using an HTTP server for the out-of-band causes an additional communication overhead in establishing the connection and fetching the public key from a web server during the query resolution. Our scheme, in contrast, maintains a public key blockchain locally on the client making the query resolution more efficient as it removes the additional communication overhead during the query resolution.

C. Fragmentation Vulnerability and Threats

Fragmentation vulnerability, threats and security have been studied in general networking beyond DNSSEC. Fragmentation opens up the scope for cache poisoning [30] and Fragmentation Mis-Association [31], [16] attacks. In, [30], Heftrig et al. used fragmentation to perform downgrading/cache poisoning attacks on DNSSEC. Their approach involves injecting a malicious fragment with the modified cipher algorithm that the resolvers don't support and skipping signature validation. In, [16], Gilad et al. demonstrated that an attacker-spoofed fragment can result in a wrong reassembly and lead to packet loss. Their results show that a blind attacker who does spoofing only can cause up to 94% packet loss.

VII. Conclusion

Transitioning to the use of post-quantum digital signature algorithms for integrity and authentication in DNS induces fragmentation. This work studies the vulnerabilities induced by fragmentation and the resulting Fragmentation Mis-Association Threat against DNS/DNSSEC. We develop a scheme to defend

and protect from Fragmentation Mis-Association attacks using the commit-and-reveal technique. Our scheme also reduces the Fragmentation Mis-Association Threat surface by offloading the public key to a blockchain which in turn reduces the number the fragments needed to transmit and verify the DNSSEC response. Using our scheme reduces the number of fragments needed to deliver the DNSSEC response by at least 33% and by a maximum of 50%. For example, a DNSSEC response with Falcon-512 fits in one fragment while the state-of-the-art research needs two fragments.

ACKNOWLEDGMENT

This material is based upon work supported by the National Science Foundation under Grant No. 1922410. The authors would also like to thank the anonymous reviewers for their helpful feedback and comments.

REFERENCES

[1] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM review*, vol. 41, no. 2, pp. 303–332, 1999.

[2] "Ibm quantum real quantum computers. right at your fingertips." https://quantum-computing.ibm.com/, Last accessed 11 Oct 2023.

[3] "Google quantum computing hardware," https://quantumai.google/hardware, Last accessed 11 Oct 2023.

[4] G. Alagic, D. Apon, D. Cooper, Q. Dang, T. Dang, J. Kelsey, J. Lichtinger, C. Miller, D. Moody, R. Peralta *et al.*, "Status report on the third round of the nist post-quantum cryptography standardization process," *US Department of Commerce, NIST*, 2022.

[5] L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlé, "Crystals-dilithium: A lattice-based digital signature scheme," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 238–268, 2018.

[6] P.-A. Fouque, J. Hoffstein, P. Kirchner, V. Lyubashevsky, T. Pornin, T. Prest, T. Ricosset, G. Seiler, W. Whyte, and Z. Zhang, "Falcon: Fast-fourier lattice-based compact signatures over ntru," *Submission to the NIST's post-quantum cryptography standardization process*, 2018.

[7] D. J. Bernstein, A. Hülsing, S. Kölbl, R. Niederhagen, J. Rijneveld, and P. Schwabe, "The sphincs+ signature framework," in *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*, 2019, pp. 2129–2146.

[8] "Executive office of the president: Migrating to post-quantum cryptography," https://www.whitehouse.gov/wp-content/uploads/2022/11/M-23-02-M-Memo-on-Migrating-to-Post-Quantum-Cryptography.pdf, Last accessed 11 Oct 2023.

[9] D. Sikeridis, P. Kampanakis, and M. Devetsikiotis, "Assessing the overhead of post-quantum cryptography in tls 1.3 and ssh," in *Proceedings of the 16th International Conference on Emerging Networking EXperiments and Technologies*, ser. CoNEXT '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 149–156. [Online]. Available: https://doi.org/10.1145/3386367.3431305

[10] R. Döring and M. Geitz, "Post-quantum cryptography in use: Empirical analysis of the tls handshake performance," in *NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium*, 2022, pp. 1–5.

[11] M. Raavi, S. Wuthier, P. Chandramouli, X. Zhou, and S.-Y. Chang, "Quic protocol with post-quantum authentication," in *International Conference on Information Security*. Springer International Publishing Cham, 2022, pp. 84–91.

[12] M. Raavi, S. Wuthier, X. Zhou, and S.-Y. Chang, "Post-quantum quic protocol in cloud networking," in *2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*. IEEE, 2023, pp. 573–578.

[13] M. Raavi, P. Chandramouli, S. Wuthier, X. Zhou, and S.-Y. Chang, "Performance characterization of post-quantum digital certificates," in *2021 International Conference on Computer Communications and Networks (ICCCN)*. IEEE, 2021, pp. 1–9.

[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, 2020.

[15] J. Goertzen and D. Stebila, "Post-quantum signatures in dnssec via request-based fragmentation," in *International Conference on Post-Quantum Cryptography*. Springer, 2023, pp. 535–564.

[16] Y. Gilad and A. Herzberg, "Fragmentation considered vulnerable: blindly intercepting and discarding fragments," in *5th USENIX Workshop on Offensive Technologies (WOOT 11)*, 2011.

[17] ——, "Fragmentation considered vulnerable," *ACM Transactions on Information and System Security (TISSEC)*, vol. 15, no. 4, pp. 1–31, 2013.

[18] J. Goertzen, "Enabling post-quantum signatures in dnssec: One arrf at a time," Master's thesis, University of Waterloo, 2022.

[19] "Workshops and timeline," https://csrc.nist.gov/projects/post-quantum-cryptography/workshops-and-timeline, Last accessed 11 Oct 2023.

[20] "Call for additional digital signature schemes for the post-quantum cryptography standardization process," https://csrc.nist.gov/csrc/media/Projects/pqc-dig-sig/documents/call-for-proposals-dig-sig-sept-2022.pdf, Last accessed 11 Oct 2023.

[21] "Post-quantum cryptography: Round 1 additional signatures," https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures, Last accessed 11 Oct 2023.

[22] S. Huh, S. Cho, and S. Kim, "Managing iot devices using blockchain platform," in *2017 19th international conference on advanced communication technology (ICACT)*. IEEE, 2017, pp. 464–467.

[23] E. Kfoury and D. Khoury, "Distributed public key infrastructure and psk exchange based on blockchain technology," in *2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*. IEEE, 2018, pp. 1116–1120.

[24] S. T. Mehedi, A. A. M. Shamim, and M. B. A. Miah, "Blockchain-based security management of iot infrastructure with ethereum transactions," *Iran Journal of Computer Science*, vol. 2, no. 3, pp. 189–195, 2019.

[25] D. Stebila and M. Mosca, "Post-quantum key exchange for the internet and the open quantum safe project," in *International Conference on Selected Areas in Cryptography*. Springer, 2016, pp. 14–37.

[26] M. Raavi, S. Wuthier, P. Chandramouli, Y. Balytskyi, X. Zhou, and S.-Y. Chang, "Security comparisons and performance analyses of post-quantum signature algorithms," in *International Conference on Applied Cryptography and Network Security*. Springer International Publishing Cham, 2021, pp. 424–447.

[27] M. Rautell, O.-M. Latvala, V. Vallivaara, and K. Halunen, "Applying a cryptographic metric to post-quantum lattice-based signature algorithms," in *Proceedings of the 17th International Conference on Availability, Reliability and Security*, ser. ARES '22. New York, NY, USA: Association for Computing Machinery, 2022. [Online]. Available: https://doi-org.libproxy.uccs.edu/10.1145/3538969.3544438

[28] K. Shrishak and H. Shulman, "Negotiating pqc for dnssec," in *2021 51st Annual IEEE/IFIP International Conference on Dependable Systems and Networks-Supplemental Volume (DSN-S)*. IEEE, 2021, pp. 9–10.

[29] G. Beernink, "Taking the quantum leap: Preparing dnssec for post quantum cryptography," Master's thesis, University of Twente, 2022.

[30] E. Heftrig, H. Shulman, and M. Waidner, "Poster: Off-path dnssec downgrade attacks," in *Proceedings of the ACM SIGCOMM 2023 Conference*, 2023, pp. 1120–1122.

[31] M. Mathis, B. Chandler, and J. Heffner, "IPv4 Reassembly Errors at High Data Rates," RFC 4963, Jul. 2007. [Online]. Available: https://www.rfc-editor.org/info/rfc4963