Falcon

What's next?

Pierre-Alain Fouque[1] Jeffrey Hoffstein[2] Paul Kirchner[1] Vadim Lyubashevsky[3] Thomas Pornin[4] Thomas Prest[5] Thomas Ricosset[6] Gregor Seiler[3] William Whyte[7] Zhenfei Zhang[8]

Technical Overview

A bird's eye view

Keygen(1^λ)

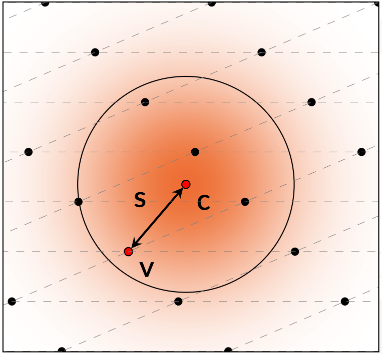
1. Gen. matrices $\mathbf{A}, \mathbf{B}$ s.t.:
   - $\mathbf{A}$ is pseudorandom
   - $\mathbf{B} \cdot \mathbf{A} = 0$
   - $\mathbf{B}$ has small coefficients
2. $\mathsf{pk} := \mathbf{A}, \mathsf{sk} := \mathbf{B}$

Sign$(\mathsf{msg}, \mathsf{sk} = \mathbf{B})$

1. Compute $\mathbf{c}$ such that $\mathbf{c} \cdot \mathbf{A} = H(\mathsf{msg})$
2. $\mathbf{v} \leftarrow$ vector in $\mathcal{L}(\mathbf{B})$, close to $\mathbf{c}$
3. $\mathsf{sig} := \mathbf{s} = (\mathbf{c} - \mathbf{v})$

Verify$(\mathsf{msg}, \mathsf{pk} = \mathbf{A}, \mathsf{sig} = \mathbf{s})$

Check ($\mathbf{s}$ short) & $(\mathbf{s} \cdot \mathbf{A} = H(\mathsf{msg}))$



Details omitted: salt the hash as $H(\mathsf{salt} \| \mathsf{msg})$, restart if $\mathbf{s}$ not short enough, etc.

Final tweaks (tentative)

- Updated encoding for signatures
  - Reduce signature sizes by about 20 bytes for Falcon-512
- BUFF transform [CDF+21]
  - Instead of $h = H(\mathbf{salt} \| \mathbf{msg})$, compute $h = H(H(\mathbf{pk}) \| \mathbf{salt} \| \mathbf{msg})$ and include h in $\mathbf{sig}$
  - Provides additional security properties
- Add the condition $\|\mathbf{s}\|_\infty \leq B_\infty$, with $B_\infty \approx 840$ (suggested by Yang Yu)
  - Forgery remains at least as hard
- Make the signing restart rate very small
  - Desirable for applications where worst-case running time matters.

Negligible impact on performance.

When to Deploy

Pros and cons

Pros

- Compact public key and signature sizes
- Very fast verification
- Signing is also fast, but less than Dilithium

Cons

- Key generation and signing require floating-point arithmetic (FPA)
  - Be mindful on devices with non-existent or variable-time floating-point units
  - Say goodbye to masking
- Key generation and signing are complex to implement
- Key generation is slow-ish

Mapping criteria to applications

Compact sizes

Verification speed

Worst-case running time

Verification memory

Vehicle-to-vehicle (V2V) communications

Compact sizes

Verification speed

Worst-case running time

Verification memory

V2V

Drive (Quantum) Safe! – Towards Post-Quantum Security for V2V Communications [BMTR22]

> “ Only signature schemes whose explicit certificate can be sent in five or less fragments can be used in the *True Hybrid* design. [...] Falcon is the only viable scheme. ”

TLS certificates

TLS

Compact sizes

Verification speed

Worst-case running time

Verification memory

Post-Quantum Authentication in TLS 1.3: A Performance Study [SKD20]

> " The PQ algorithms with the best performance for time-sensitive applications are Dilithium and Falcon. "

NIST's pleasant post-quantum surprise [Wes22] recommends:

- → Falcon for offline signature
- → Dilithium for handshake

Verification on embedded devices

Compact sizes

Verification speed

Worst-case running time

Verification memory

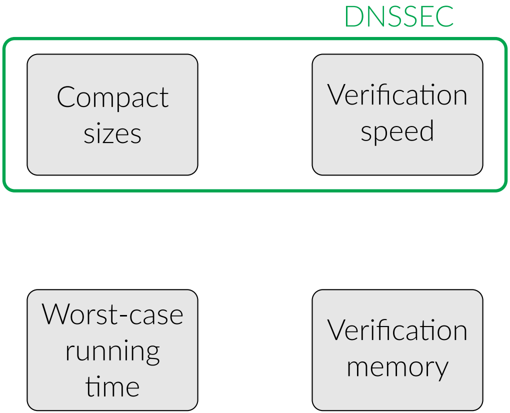
Embedded verif.

FPGA Energy Consumption of Post-Quantum Cryptography [BKG22]

> “ For signature verification, Falcon provides the lowest energy consumption, highest throughput, and lowest transmission size [compared to Dilithium and SPHINCS+]. ”

Verifying Post-Quantum Signatures in 8 kB of RAM [GHK+21]

> “ On Cortex-M3, [Falcon’s] overall memory footprint is about 6.5 kB. ”



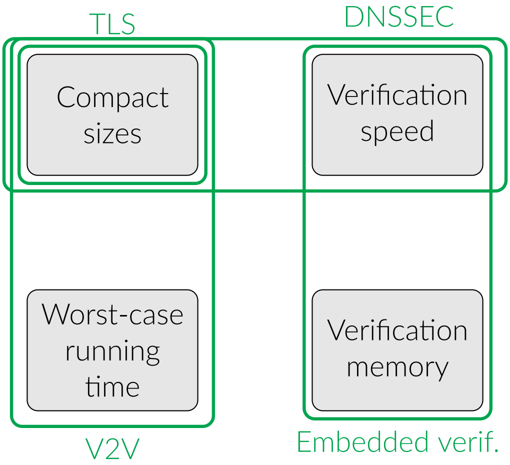
Retrofitting Post-Quantum Cryptography in Internet Protocols: A Case Study of DNSSEC [MdJvH+20]

> “ [...] the performance of Falcon-512 is closest to the current algorithms and meets the requirements of DNSSEC. ”

Post-Quantum Signatures in DNSSEC via Request-Based Fragmentation [GS22]

> “ [...] Falcon-512 may be the most suitable option currently available to be standardized for DNSSEC. ”

Summary



Suitable applications:

- V2V
- TLS certificates
- Verification on embedded devices
- DNSSEC
- …

What's next?

Specification

- **NIST draft standard:** 2023-2024?
- IETF draft?

Design evolution

- *SOLMAE* [KTW+22] [Korean PQC submission]

> " [SOLMAE] uses the same simple, fast, parallelizable signing algorithm as Mitaka [...]. However, by leveraging a novel key generation algorithm [...], SOLMAE achieves the same high security and short key and signature sizes as Falcon. "

Suggestion are welcome!

Specification

- → **NIST draft standard:** 2023-2024?
- → IETF draft?

Design evolution

- → *SOLMAE* [KTW+22] [Korean PQC submission]

> " [SOLMAE] uses the same simple, fast, parallelizable signing algorithm as Mitaka [...]. However, by leveraging a novel key generation algorithm [...], SOLMAE achieves the same high security and short key and signature sizes as Falcon. "

Suggestion are welcome!

PS: feel free to grab a physical copy of our white paper ☺
"*The First NIST Post-Quantum Cryptographic Standards*"

Thank You!

https://falcon-sign.info/

Luke Beckwith, Jens-Peter Kaps, and Kris Gaj.
Fpga energy consumption of post-quantum cryptography.
In *Fourth PQC Standardization Conference*, 2022.
https://csrc.nist.gov/Events/2022/fourth-pqc-standardization-conference.

Nina Bindel, Sarah McCarthy, Geoff Twardokus, and Hanif Rahbari.
Drive (quantum) safe! – Towards post-quantum security for V2V communications.
Cryptology ePrint Archive, Report 2022/483, 2022.
https://eprint.iacr.org/2022/483.

Cas Cremers, Samed Düzlü, Rune Fiedler, Marc Fischlin, and Christian Janson.
BUFFing signature schemes beyond unforgeability and the case of post-quantum signatures.
In *2021 IEEE Symposium on Security and Privacy*, pages 1696–1714. IEEE Computer Society Press, May 2021.

Ruben Gonzalez, Andreas Hülsing, Matthias J. Kannwischer, Juliane Krämer, Tanja Lange, Marc Stöttinger, Elisabeth Waitz, Thom Wiggers, and Bo-Yin Yang.
Verifying post-quantum signatures in 8 kB of RAM.
In Jung Hee Cheon and Jean-Pierre Tillich, editors, *Post-Quantum Cryptography - 12th International Workshop, PQCrypto 2021*, pages 215–233. Springer, Heidelberg, 2021.

Jason Goertzen and Douglas Stebila.
Post-quantum signatures in dnssec via request-based fragmentation, November 2022.

Kwangjo Kim, Mehdi Tibouchi, Alexandre Wallet, Thomas Espitau, Akira Takahashi, Yang Yu, and Sylvain Guilley.
Solmae.
Korean PQC competition - Round 1 submission, 2022.
https://kpqc.or.kr/competition.html.

Moritz Müller, Jins de Jong, Maran van Heesch, Benno Overeinder, and Roland van Rijswijk-Deij.
Retrofitting post-quantum cryptography in internet protocols: A case study of dnssec.
volume 50, page 49–57, New York, NY, USA, oct 2020. Association for Computing Machinery.

Dimitrios Sikeridis, Panos Kampanakis, and Michael Devetsikiotis.
Post-quantum authentication in TLS 1.3: A performance study.
In *NDSS 2020*. The Internet Society, February 2020.

Bas Westerbaan.
Nist's pleasant post-quantum surprise.
The Cloudflare Blog, July 2022.
https://blog.cloudflare.com/nist-post-quantum-surprise/.