Technical Overview
A bird’s eye view

### Keygen($1^λ$)

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

### Sign(msg, sk $B$)

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

### Verify(msg, pk $A$, sig $s$)

Check ($s$ short) & ($s \cdot A = H(msg)$)

Details omitted: salt the hash as $H(\text{salt} || \text{msg})$, restart if $s$ not short enough, etc.
When to Deploy
Pros and cons

Pros
- Compact sizes
- Very fast verification
- Signing is also fast, but less than Dilithium

Cons
- Keygen and signing require floating-point arithmetic
- Keygen and signing are complex to implement
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

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

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
- Verification memory
- Worst-case running time

**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.”

“ [...] 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
- ...

<table>
<thead>
<tr>
<th>TLS</th>
<th>DNSSEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compact sizes</td>
<td>Verification speed</td>
</tr>
<tr>
<td>Worst-case running time</td>
<td>Verification memory</td>
</tr>
</tbody>
</table>

**V2V**

**Embedded verif.**
Towards FN-DSA
Keygen and signing require **floating-point arithmetic (FPA)**

- Makes validation (i.e. KATs) difficult
- Be mindful on devices with non-existent or variable-time floating-point units
- Say goodbye to masking

How do we mitigate that?

- **Key generation**: use fixed-point arithmetic as in Hawk
- **Signing**: potential solution is to use Antrag
Antrag is a modified key generation algorithm proposed by Espitau et al., *Antrag: Annular NTRU Trapdoor Generation*, ASIACRYPT 2023 [ENS+23].

**Pros**
- Gives “better quality” trapdoors
- Make signing simpler (fast Fourier sampler → hybrid sampler)
- FPA becomes easier to analyze and possibly remove

**Cons**
- Very recent, too early for standardisation
- Full security implications to be determined

See Quyen’s talk tomorrow!
Smaller suggested tweaks for FN-DSA

BUFF transform \([CDF^+21]\)

- Instead of \(h = H(\text{salt}||\text{msg})\), compute \(h = H(H(\text{pk})||\text{salt}||\text{msg})\) and include \(h\) in \(\text{sig}\)
- Possibly better solution: use the lighter PS-3 transform \([PS05]\) like HAWK
- Provides additional security properties

Add the condition \(|s|_\infty \leq B_\infty\), 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.
Thank You!

https://falcon-sign.info/


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.

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

Moritz Müller, Jins de Jong, Maran van Heesch, Benno Overeinder, and Roland van Rijswijk-Deij.

Thomas Pornin and Julien P. Stern.
Digital signatures do not guarantee exclusive ownership.

Dimitrios Sikeridis, Panos Kampanakis, and Michael Devetsikiotis.

Bas Westerbaan.
Nist’s pleasant post-quantum surprise.
The Cloudflare Blog, July 2022.