Security of Hedged Fiat–Shamir Signatures under Fault Attacks

Eurocrypt 2020
ePrint https://ia.cr/2019/956

Diego F. Aranha\textsuperscript{1} Claudio Orlandi\textsuperscript{1}
Akira Takahashi\textsuperscript{1} Greg Zaverucha\textsuperscript{2}

May 14, 2020

\textsuperscript{1}Aarhus University, Denmark
\textsuperscript{2}Microsoft Research, United States
Goal of Our Work

• Formally analyze the fault-resilience of existing Fiat–Shamir signatures
  • Provable security methodology.
  • Motivated by actual fault attacks on concrete schemes.
1. Randomized signature: $r \leftarrow \text{RNG}(\cdot)$

- Nonces don’t need to be uniform: low-quality RNG or counter should suffice.
- Randomness $r$ doesn’t repeat on the same message.

*To what extent are hedged FS signatures secure against fault attacks?*
1. Randomized signature: \( r \leftarrow \text{RNG}(\cdot) \) 😞 Risk of randomness bias!

- Nonces don’t need to be uniform: low-quality RNG or counter should suffice.
- Randomness \( r \) doesn’t repeat on the same message.

To what extent are hedged FS signatures secure against fault attacks?
How to Protect Fiat–Shamir from Randomness Failure?

1. Randomized signature: \( r \leftarrow \text{RNG}(\cdot) \) 😞 Risk of randomness bias!

2. Deterministic signature: \( r \leftarrow H(sk, m) \)

- Nonces don’t need to be uniform: low-quality RNG or counter should suffice.
- Randomness \( r \) doesn’t repeat on the same message.

To what extent are hedged FS signatures secure against fault attacks?
How to Protect Fiat–Shamir from Randomness Failure?

1. Randomized signature: \( r \leftarrow \text{RNG}(\cdot) \)  
   - Risk of randomness bias!

2. Deterministic signature: \( r \leftarrow H(sk, m) \)  
   - Vulnerable to fault attacks!

- Nonces don’t need to be uniform: low-quality RNG or counter should suffice.
- Randomness \( r \) doesn’t repeat on the same message.

To what extent are hedged FS signatures secure against fault attacks?
How to Protect Fiat–Shamir from Randomness Failure?

1. Randomized signature: \( r \leftarrow \text{RNG}(\cdot) \)  🙁 Risk of randomness bias!
2. Deterministic signature: \( r \leftarrow H(sk, m) \)  😞 Vulnerable to fault attacks!
3. Hedged signature: \( r \leftarrow H(sk, m, nonce) \)  😊 Seems secure?

- Nonces don’t need to be uniform: low-quality RNG or counter should suffice.
- Randomness \( r \) doesn’t repeat on the same message.

*To what extent are hedged FS signatures secure against fault attacks?*
Contributions

• Formal attacker model and security notions to capture the corrupted nonces and previous fault attacks.
• Proved that hedged FS schemes in general are secure against single-bit fault attacks on many intermediate wire values in the signing algorithm.
  + Negative results for a few wires.
• Application to concrete instantiations.
  • XEdDSA: Hedged variant of EdDSA used in Signal
  • Picnic2: NIST PQC competition round 2 candidate
Overview of Our Results

If $A$ doesn’t query the same $(m, n)$ pair more than once
- ✓ secure against single-bit flip/stuck-at faults.
- ✗ insecure against single-bit flip/stuck-at faults.
- ★ security only holds for signatures from subset-revealing ID (e.g., Picnic).
- ▲ security only holds for signatures from input-delayed ID (e.g., XEdDSA).
If $\mathcal{A}$ doesn’t query the same $(m, n)$ pair more than once

- ✓ secure against single-bit flip/stuck-at faults.
- ✗ insecure against single-bit flip/stuck-at faults.
- ★ security only holds for signatures from subset-revealing ID (e.g., Picnic).
- ▲ security only holds for signatures from input-delayed ID (e.g., XEdDSA).
Overview of Our Results

If $\mathcal{A}$ doesn’t query the same $(m, n)$ pair more than once

- ✓ secure against single-bit flip/stuck-at faults.
- ✗ insecure against single-bit flip/stuck-at faults.
- ★ security only holds for signatures from subset-revealing ID (e.g., Picnic).
- ▲ security only holds for signatures from input-delayed ID (e.g., XEdDSA).
Conclusion

- Hedged FS is provably more resilient than the randomized/deterministic FS!
  - Negative results show where practitioners pay the most attention.
- Open questions
  - Extension to more advanced fault attacker model.
    - Multi-bit/position faults. Partially handled by Fischlin and Günther (CT-RSA’20) for generic signatures.
    - Fault within Com, Resp or public parameters.
    - Model for instruction skipping faults.
    - Fault + QROM.
  - Lattice signatures from FS with aborts.

Thank you!

More details at https://ia.cr/2019/956