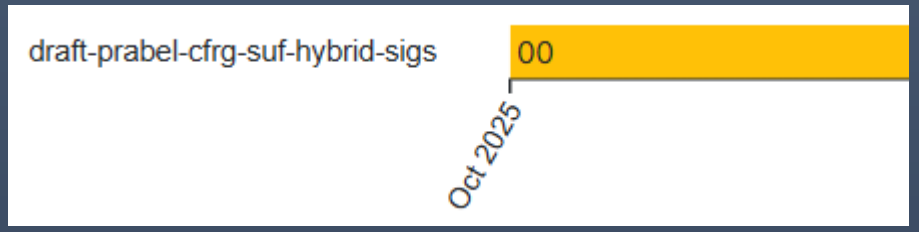
Hybrid Digital Signatures with Strong Unforgeability

draft-prabel-cfrg-suf-hybrid-sigs-00

L. Prabel, G. Wang, J. Janneck, T. Reddy, J. Preuß Mattsson



IETF 124, Montreal

Motivation & Problem

Why this draft?

EUF-CMA (*Existential Unforgeability under Chosen Message Attack):* no attacker can forge *any* valid signature for a ***new*** message.
SUF-CMA (*Strong Unforgeability under Chosen Message Attack*): no attacker can forge any new valid signature, even for a **previously signed** message.

- Current hybrid signature schemes (e.g., composite hybrids) in several IETF drafts are only **EUF-CMA secure.**
- However, some applications need **SUF-CMA security**: protecting against replay attacks, log poisoning, ensuring non-repudiation, …
- **Challenge**: How to achieve **SUF-CMA** even if one component is only **EUF-CMA** secure?

This draft defines two hybrid signature constructions offering SUF-CMA security under different conditions.

1st construction: black box

$KeyGen() \rightarrow pk, sk$

1. $(pk_1, sk_1) = Trad.KeyGen()$
2. $(pk_2, sk_2) = PQ.KeyGen()$
3. Output $pk = (pk_1 \,||\, pk_2)$ and $sk = (sk_1 \,||\, sk_2)$

$Sign(sk, m) \rightarrow s$

1. Compute $m' = Prefix \,||\, Label \,||\, len(ctx) \,||\, ctx \,||\, PH(m)$
2. $s_1 = Sign_1\,(sk_1, m')$
3. $s_2 = Sign_2(sk_2, m' \,||\, s_1)$
4. Output $s = (s_1 \,||\, s_2)$

$Verify(pk, m, s) \rightarrow \{0, 1\}$

1. Compute $m' = Prefix \,||\, Label \,||\, len(ctx) \,||\, ctx \,||\, PH(m)$
2. Compute $Verify_1(pk_1, m', s_1)$
3. Compute $Verify_2(pk_2, m' \,||\, s_1, s_2)$
4. Accept if both verifications succeed.

- **Black box.** The hybrid construction treats both signature schemes as black boxes, without depending on internal details of either scheme.
- **SUF-CMA security**. The 2nd signature binds to the 1st one, ensuring SUF-CMA security as long as the 2nd component is SUF-CMA secure.
- **Non-parallelizable**. The signing process is sequential (sign traditionally, then post-quantum).

2nd construction: non-black box

$Sign(m, sk) \rightarrow s$

1. Compute m′ = Prefix || Label || len(ctx) || ctx || pk_1 || pk_2 || PH(m)
2. Compute $(com, st) = ID.Com(sk_1)$
3. Compute $m'' = PH(1 \,||\, m'' \,||\, com)$
4. Compute $s_2 = \text{Sign}_2(sk_2 \,||\, \text{m}'')$
5. Compute $chl = PH(2 \,||\, s_2)$
6. Compute $rsp = \text{ID.Rsp}(\text{sk}_1, com, chl, st)$
7. $Return\ s = (rsp \,||\, s_2)$

$Verify(m, s, pk) \rightarrow \{0, 1\}$

1. Compute m′ = Prefix || Label || len(ctx) || ctx || pk_1 || pk_2 || PH(m)
2. Compute $chl = PH(2 \,||\, s_2)$
3. Compute $com = ID.ExtCom(pk_1, chl, rsp)$
4. Compute $m'' = PH(1 \,||\, m'' \,||\, com)$
5. Compute $Verify_2(pk_2, m'', s_2)$
6. Accept if verification succeeds.

- **Non-black box**. The 2nd component depends on the internal structure of the Fiat-Shamir scheme.
- **SUF-CMA security.** The hybrid construction remains SUF-CMA secure as long as one component is SUF-CMA secure.
- **Non-parallelizable**. The signing process is sequential (sign traditionally, then post-quantum).

Next Steps

Next planned changes

- Detailed analysis of **security**.
- Potential **refinements** of the construction.
- **Additional details** on the properties of each construction.
- **Clarification**, editorial modifications, …

Questions for CFRG

- Do you see a **need** for this hybrid construction in **real-world applications**?
- Do you have a preference for **black-box vs. non-black-box** constructions?
- What are the important **gaps** that need to be filled?
- Any **suggestions or concerns** regarding the proposed constructions?

All comments and suggestions are welcome

https://github.com/lucasprabel/draft-cfrg-suf-hybrid-sigs