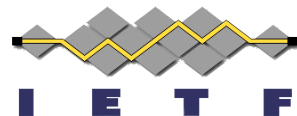


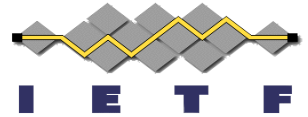
Adapting HSMs for Post-Quantum Cryptography



[draft-reddy-pquip-pqc-hsm-00](#)

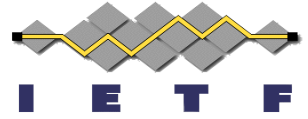
Tirumaleswar Reddy, Dan Wing, Ben Salter
IETF 122, Bangkok

Introduction



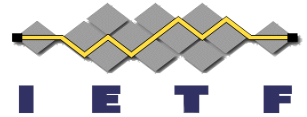
- Hardware Security Modules (HSMs) are widely used for secure key management.
- Transitioning to PQC introduces challenges for constrained HSMs.
 - MCUs integrate cryptographic hardware directly into the System-on-Chip.
 - External Security Chips for Cryptographic Operations.
- Constrained HSMs face unique limitations: storage, memory, computational power, and performance.

Scope of Draft



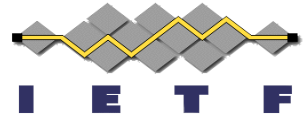
- Optimize PQC integration in constrained HSMs
- Optimal PQC signature algorithms
- PQC impacts to HSM firmware updates and backup

Seed-Based Key Generation



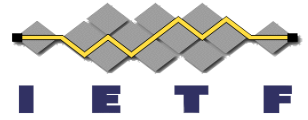
- PQC algorithms often require large keys. Constrained HSMs have limited storage capacity for key material.
- Seed-based key generation is a promising approach but has trade-offs.
- Key Idea: **Generate private keys deterministically from a seed.**
 - Reduces storage overhead (store only the seed in flash, not fully expanded private key).
 - ML-DSA-87: seed (32 bytes) and private key (4896 bytes)
 - Good: Enhances scalability in HSMs with limited memory.
 - Bad: Increases computational cost for key derivation.
 - ML-DSA key generation is more computationally efficient than SLH-DSA.

One-Way Nature of Key Derivation



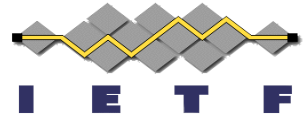
- Reconstructing the seed from the expanded key is infeasible.

Ephemeral Key Handling



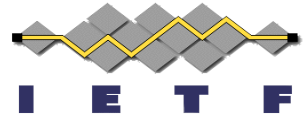
- Many protocols like TLS and IPSec require ephemeral key generation.
- Generate ephemeral key-pairs on-demand from an ephemeral seed stored temporarily within the cryptographic module.
- Enforce immediate seed erasure after the key-pair is generated and the cryptographic operation is completed.
- Delete private key after shared secret is generated.
- Prevent key reuse across different algorithm suites or sessions.

Optimizing performance of Signature Algorithms



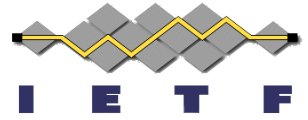
- Problem: Transmitting full messages to HSMs increases overhead.
- **Solution:** Send only a message digest instead of the full message.
 - Reduces communication overhead.
 - Applicable to ML-DSA, SLH-DSA, and future PQC algorithms.
 - ExternalMu-ML-DSA (ietf-lamps-dilithium-certificates)
 - Pre-hashing done in a software module.
 - Only pre-hashed message (mu) sent to HSM for signing.

Impact on HSM Firmware Updates & Backup



- PQC adoption will require
 - HSM Firmware updates to support new cryptographic algorithms.
- HSM will have to adopt PQC signature schemes for code signing.
- Recommended post-quantum algorithms:
 - **HSS-LMS, SLH-DSA, XMSS and ML-DSA**
 - **ML-DSA**: lattice-based and may be vulnerable to future attacks.
 - SLH-DSA might be better choice than ML-DSA for firmware authentication, particularly when long-term cryptographic stability is essential.

Next Steps



- Comments and suggestions are welcome