ETSI/IQC Quantum Safe Cryptography Conference 2026

State Management Challenges in Distributed XMSS/LMS Deployments

Presented by:

Jefferson Ricardini

Sr. Applied Scientist, Amazon

17/06/2026

The single-reuse failure mode

One index reuse. Not a class break, not a subtle flaw.

- Any observer of both signatures can **forge new signatures** without the private key.
- Classical key protection still applies. The new problem is counter duplication.

After 1st signature — SECURE ✓

Chain i: $x_i \to H \to H \to \cdots \to H \to y_i$

Attacker sees $\sigma_i = H^{(m_i)}(x_i)$

✅ Can hash forward: $\sigma_i \to H(\sigma_i) \to H(H(\sigma_i)) \to \cdots$
pushes digit up → checksum drops

❌ Checksum drop requires backward step → must invert hash → impossible

Checksum blocks ALL forgery directions.

After 2nd signature — FORGERY ✗

Two sigs on different messages m, m':
$\sigma_i = H^{(m_i)}(x_i)$ -> $\sigma'_i = H^{(m'_i)}(x_i)$

Attacker holds depth $\min(m_i, m'_i)$ per chain.

Since $m \neq m'$, some chains have $m_i < m'_i$
and others have $m_i > m'_i$.

→ Lower floors across different chains
→ Forge m^* where $m^*_i = \min(m_i, m'_i)$

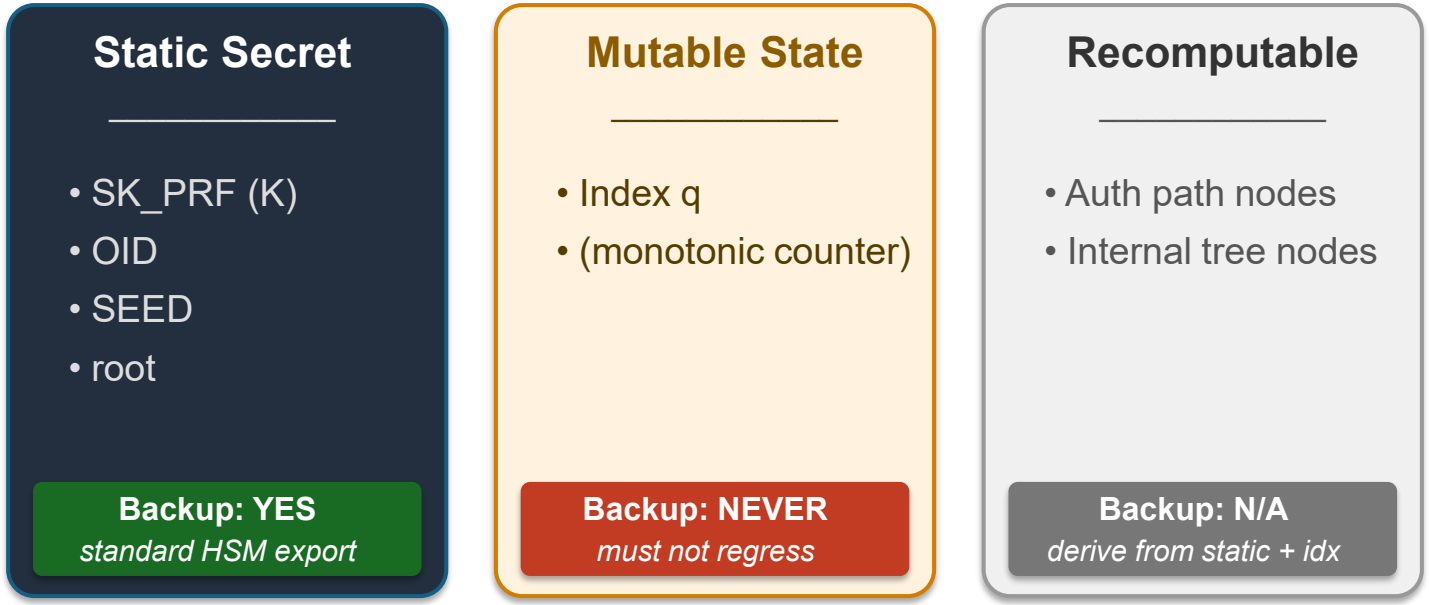
All digits reachable. Checksum of m^* also reachable. No hash inversion needed.

Private key vs state

Private key ≠ single secret — three parts.

- Threat model: **regression or duplication**, not disclosure.
- Protect the counter's monotonicity, not the counter.

XMSS Private Key



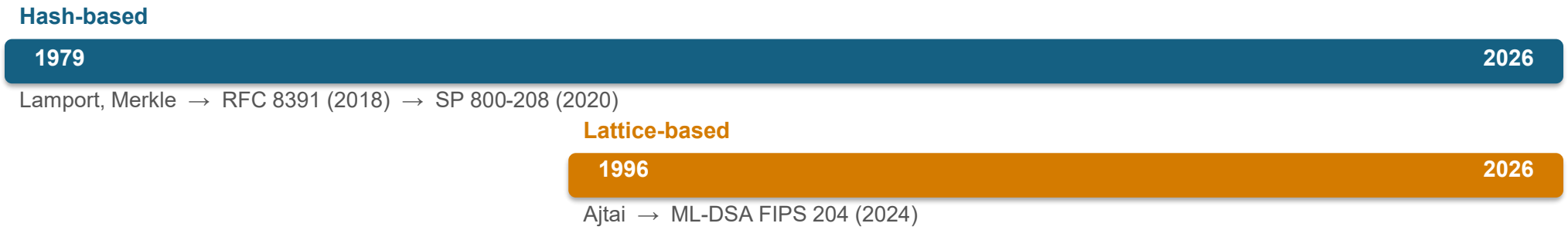
Why distributed breaks it

- **Sync latency**: volatile/persistent storage skew; crash between sign and commit.
- **Sync failure:** power loss or corruption between state write and signature release.
- **Cloning:** VM snapshot, forked process, restored backup, replicated HSM.

Restore from backup can be more dangerous than device loss.

Why stateful HBS, really

- Hash-based: minimal-assumption crypto, 47 years of cryptanalysis.



- ML-DSA standardized Oct 2024. Any SoC taped-out before that date ships with XMSS as its only PQC secure boot option. Those chips are deploying now.
- The obstacle: SP 800-208 §8.1 blocks key export, leaving no validated path to DR/HA. That gap is why the revision exists.

The solution space at a glance

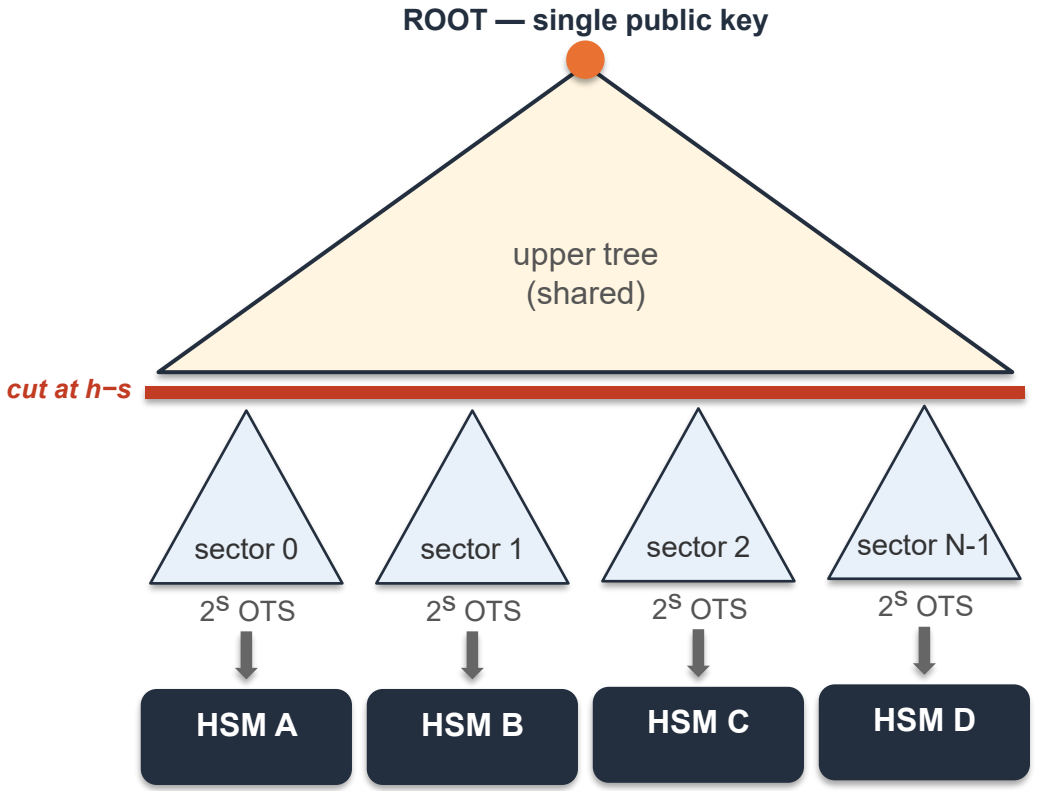
Nine approaches in draft-ietf-pquip-hbs-state-04, three natural families. Sectorization (highlighted) is the focus of this talk.

Partition the signing space	Manage state differently	Change the scheme
Sectorization ← focus	Key/state transfer	Key rotation
Distributed multi-tree	Pre-assigned states	Variable-length chains
Multiple public keys	Time-based	
	Interval-based	

Sectorization — the idea

Cut the tree at h − s. Lower half becomes 2^(h−s) sectors.

- Each sector: own PRF seed, disjoint counter range, own auth path.
- Different HSMs sign in parallel. Same public key.
- Verifier changes: **none**.



N = 2^(h−s) sectors · disjoint counter ranges · verifier unchanged

Sectorization — tradeoffs

Nothing here is free. Size sectors to your actual failure and scaling model.

Wins

- Parallel signing
- No cross-HSM sync
- Isolated blast radius
- Single public key

Costs

- Counter waste on HSM loss
- Upfront provisioning ceremony
- Tree-height inflation
- Fixed sector size per key

Honest Limits

- AUTH formats still negotiated
- No interop vectors yet
- Uneven vendor support
- Some HSMs can't export K

Standards convergence & next steps

IETF pquip (catalog) + NIST 800-208 (normative) converging on sectorization. CNSA 2.0 required for new NSS acquisitions from Jan 1, 2027.

1. Algorithm

LMS SHA-256/192 or XMSS single-tree

NSA preference: LMS

Multi-tree variants: out

Tree height = device-lifetime sig budget + sectorization waste

>10k signatures → use ML-DSA

2. Architecture

Design every interface around per-sector provisioning today

Abstraction layer:
counter range, sector ID, AUTH block

Drop in certified hardware when it arrives

3. Procurement

CNSSP 15: CNSA 2.0 for new NSS acquisitions from Jan 1, 2027

Ask vendors:

- 208-revision roadmap
- AUTH protocols (live + offline)
- Per-sector import story

No certified stateful HBS HSM today

Takeaway

1. If your signing is distributed, stop synchronizing counters — partition them.
2. Sectorization is the direction. **Build architecture for it now**; drop in certified hardware when it arrives.
3. No validated path to HA for stateful HBS today. Start vendor conversations now: CNSA 2.0 requires compliance by **Jan 1, 2027**.

THANKS — HAPPY TO TAKE QUESTIONS

Sectorization — the import block

Sectorization needs a portable format: how does one HSM hand a sector to another?

Per-block PRF derivation isolates sectors: top-level compromise does not retro-expose delegated blocks.

Transfer block

`version`	4 B
`type`	4 B
`CPK`	variable — full XMSS/LMS public key
`PATH`	variable — Merkle path sector → root
`start`	4 B — counter lower bound
`end`	4 B — counter upper bound
`AUTH`	variable — authorization data

Per-block PRF key

`K[block] = PRF(K,domain-sep||CPK||PATH||start|| end)`

AUTH1 variants

Interactive: target HSM issues R, source binds transfer to R

Offline: R = timestamp/one-shot nonce at target

Source HSM tracks transfers — no block handed out twice · verifier unchanged