White Paper

Quantum-Resistant Code Signing Secured by Hardware Security Modules

thalestct.com

THALES
Building a future we can all trust

Stateful Hash-Based Signature Schemes

Stateful hash-based signature (HBS) schemes are digital signature schemes believed to be resistant to the threat posed by a cryptographically relevant quantum computer. The National Institute of Standards and Technology (NIST) has standardized two stateful HBS schemes under SP 800-208: the Leighton-Micali Signature (LMS) system and the eXtended Merkle Signature Scheme (XMSS), including their multi-tree variants, the Hierarchical Signature System (HSS) and multi-tree XMSS. Stateful HBS schemes differ from other asymmetric signature schemes in that a HBS private key is comprised of a predefined set of one-time signature (OTS) private keys. "State" refers to the capacity to enforce single usage of each OTS private key across the lifespan of the HBS private key.

CNSA 2.0

In September 2022, NSA released the Commercial National Security Algorithm Suite 2.0 (CNSA 2.0), setting timelines for the adoption of quantum-resistant algorithms in national security systems. Under CNSA 2.0, vendors are encouraged to adopt stateful HBS schemes as defined in SP 800-208 immediately for all software and firmware code signing, with a requirement to support them by 2025.
NSA provided three justifications for preferring stateful HBS schemes now for code signing versus waiting for final standards for post-quantum asymmetric algorithms:

1. **Urgency.** Firmware for systems being deployed into the field now may continue to be in service well after the quantum threat becomes real.
2. **Standards.** NIST has already codified the standards for stateful HBS schemes under SP 800-208. Final standards for new post-quantum algorithms are not expected until 2024.
3. **Cryptanalysis.** HBS schemes have been extensively researched for their quantum resistance and the performance impacts of implementing HBS schemes are non-critical to the code signing use case.

Thales Luna Hardware Security Modules (HSMs) Support of LMS/HSS

LMS/HSS enables customers to begin their transition to quantum-resistant software and firmware signing. Thales has two separate LMS/HSS implementations that are both compliant with SP 800-208 and PKCS#11 v3:

Luna T-Series HSM Release 7.13.0

In June 2023, Thales TCT released support for LMS mechanisms, along with its multi-tree variant HSS, in all Luna T-Series hardware security modules (HSM) starting in version 7.13.0.

Luna 7 Functionality Modules

Thales has also released a Functionality Module (FM) that can be installed on existing Luna 7 HSMs without doing a firmware upgrade.

SP 800-208 and HSMs

SP 800-208 requires that all stateful HBS key generation and signature algorithms be implemented within a FIPS 140 certified HSM with Level 3 physical security. Furthermore, the HSM "shall not allow for the export of private keying material." In practice, in order to maintain conformity to the standards, private keys cannot be cloned to other HSMs to establish high availability and redundancy. Nor can backup and restore operations be performed to create offline cold storage backups of the keying material. These limitations are intended to ensure that the state of the OTS signature keys is always enforced and keys are never reused, which would introduce cryptologic vulnerabilities into the signature scheme.

For vendors making use of LMS/HSS to sign software or firmware, this means that the entire lifecycle of the stateful HBS private keys must be considered at instantiation of the signature scheme, and keying material be generated across a distributed cache of HSMs to ensure redundancy over that lifetime.

SP 800-208 describes two methods by which redundancy across multiple HSMs can be achieved:

1. **Enable endpoints to accept signatures from multiple independent root keys.** This allows the vendor to establish redundant roots across multiple (n) HSMs and tolerate the loss of n-1 HSMs over the lifetime of the signature scheme.
2. **Use a hierarchical tree of private keys distributed across HSMs.** This allows vendors to ensure redundancy and similarly tolerate the loss of n-1 HSMs, but endpoints need only be configured to accept signatures from a single root key.

When to Use Stateful HBS for Code Signing Support

The code signing requirements for each vendor's products will determine whether or not an LMS/HSS architecture is feasible. There are certain use cases such as resource constrained devices or massive code signing volumes that are not good matches for an LMS/HSS code signing solution. However, with proper upfront planning, most code signing requirements can be met using a set of LMS/HSS capable HSMs. The following Reference Architectures illustrate a solution for a hypothetical code signing use case. Although this is just one example, it is easy to see how the architecture can scale to meet a wide range of code signing needs.

Reference Architectures

Using a single, common set of firmware signing requirements, three code signing architectures are described:

- Basic Architecture
- High Assurance Architecture
- Offline Redundancy Architecture

The selection of which architecture a hardware or software vendor would select depends on additional variables such as risk tolerance, geographic separation requirements, and complexity. However, having three valid architectures to meet the same requirements illustrates the flexibility of LMS/HSS HSM deployment solutions.

Vendor Requirements

The following requirements from a hypothetical vendor requiring stateful HBS firmware signatures will be used to demonstrate an SP 800-208 compliant signature scheme:

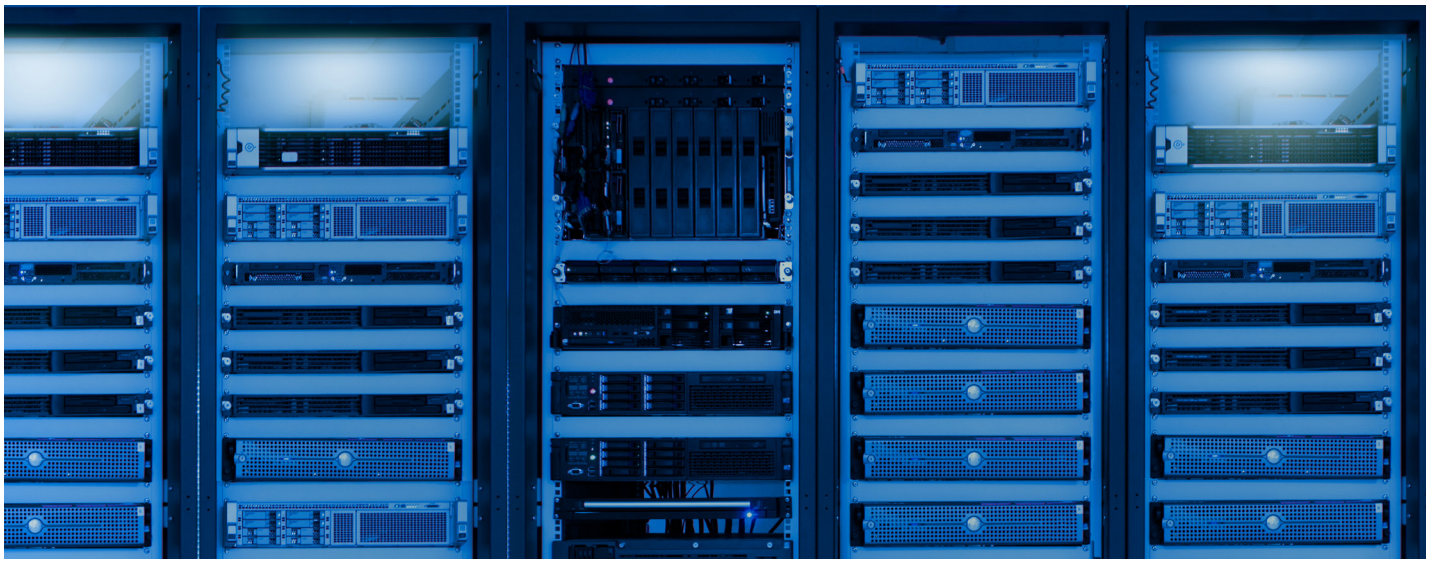
- The vendor has **four (4)** product lines
- Each product line must have an independent PKI root
- Each product has the capability to accept firmware signatures from only a single root
- Signature sizes cannot fluctuate over the lifespan of the products
- The vendor averages **one (1)** signature per workday per product
- Each product has a lifespan of **25 years**
- **Thus the number signatures required is 6,500 per product or 26,000 in total**
- The vendor has **four (4)** Thales Luna Network HSMs in two (2) regions to distribute keys across

Basic Architecture

Since each product can only use a single root, these signature schemes must utilize a hierarchical tree of private keys distributed across the four (4) available HSMs. And because signature sizes cannot fluctuate over the lifespan of the products, the number of tiers in the hierarchy must be predetermined and set during instantiation of the signature scheme. Therefore, it is critical to calculate the total required capacity of private keys for the entire lifespan of a given signature scheme prior to generating any keying material.
For this architecture, the key hierarchy will consist of two (2) tiers: (1) a Root Key at the top for each product line (with LMS/HSS parameter Height=5), and (2) a set of Signing Keys at the bottom distributed across the HSMs (with LMS/HSS parameter Height=10). The Root Key will contain a set of 32 OTS private keys, which will be used to sign 32 Signing Keys divided among the available HSMs. Each Signing Key will contain a set of 1,024 OTS private keys. This creates a total capacity of 32,768 signatures per product or 131,072 in total. Compared to the requirement of 6,500 signatures per product, the signature capacity of 32,768 per product provides a fourfold surplus of signing keys that will be necessary to achieve the required redundancy once the signature scheme is deployed into the available HSMs.

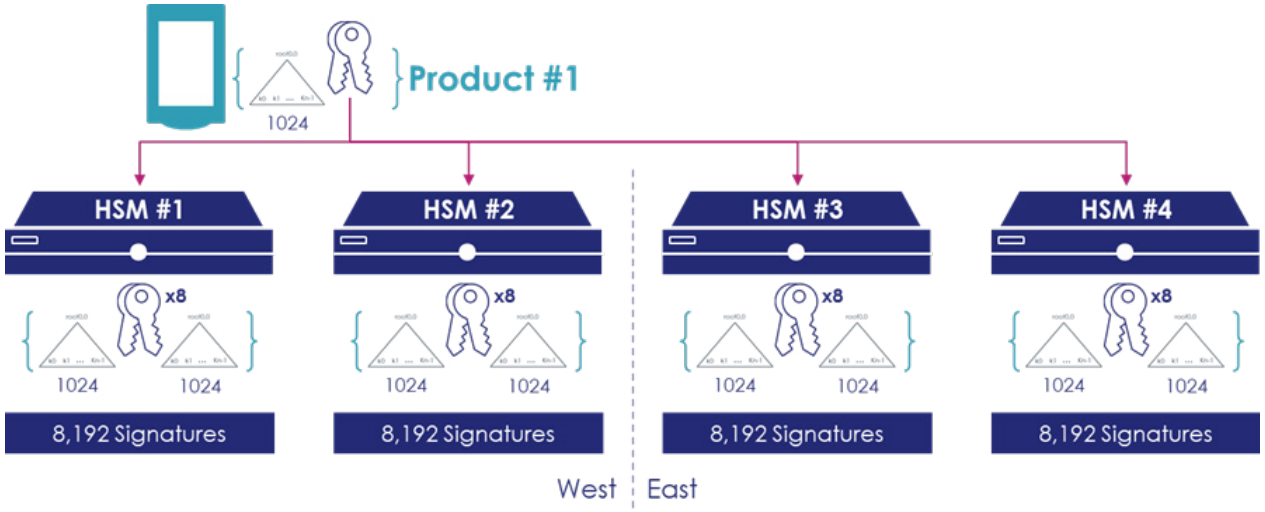
To deploy into the HSMs, the first step for each product is to generate the Root Key on one HSM. Then, generate eight (8) Signing Keys on all four HSMs, signing all 32 with the Root Key. This distribution enables load balancing across the available HSMs and ensures that there is sufficient capacity on each individual HSM to support the entire lifecycle of signatures. And because the Root Key's private key is no longer needed – having exhausted its OTS signature capacity – this signature scheme can tolerate the loss of any and all but one of the deployed HSMs.

This distribution enables load balancing across the available HSMs and ensures that there is sufficient capacity on each individual HSM to support the entire lifecycle of signatures. And because the Root Key's private key is no longer needed – having exhausted its OTS signature capacity – this signature scheme can tolerate the loss of any and all but one of the deployed HSMs.

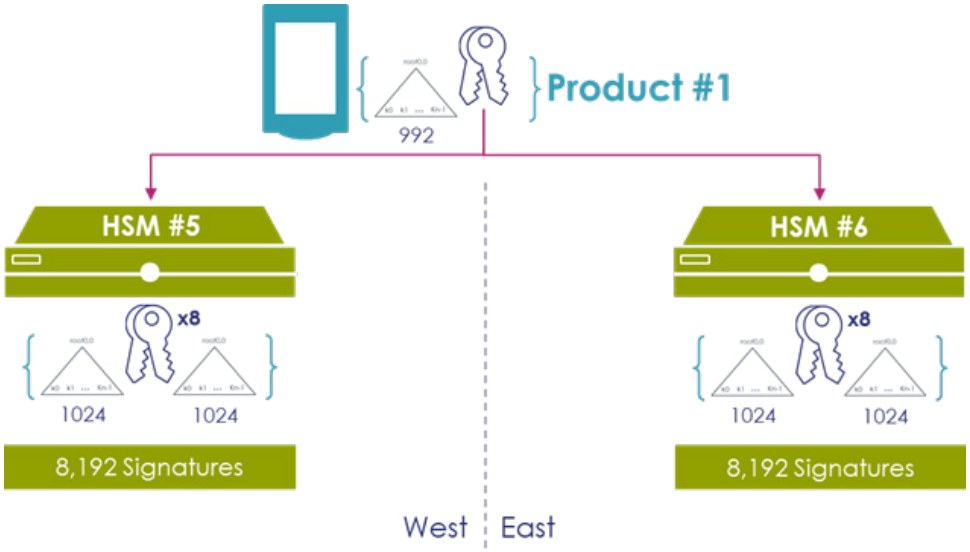


Higher Assurance Architecture

This signature scheme can be further enhanced with a higher level of assurance by introducing a Thales USB-based HSM. Instead of generating the Root Key for each product on one of the production Network HSMs, the vendor can generate the Root Key on a USB-based HSM.



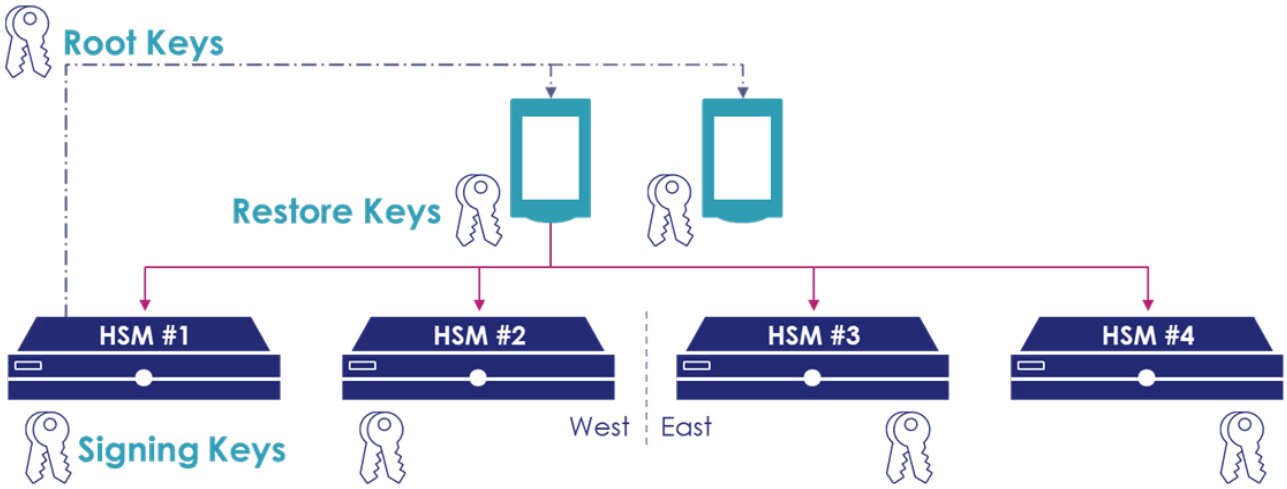
Here, the Root Key is generated with a set of 1,024 OTS private keys, while the same Signing Keys are generated on the Network HSMs. Then, the Tablet HSM can be taken offline and put into cold storage.



In this configuration, the signature scheme can tolerate the loss of all production Network HSMs as well as accept the introduction of new Thales HSMs later in the products' lifecycles. Upon hardware failure or decommissioning, new Network HSMs can be brought online, the USB-based HSM brought out of storage, and the spare OTS private keys on the Root Key used to sign newly generated Signing Keys.

Offline Redundancy Architecture

By adding a third tier to the LMS key hierarchy, the signature scheme can replicate a traditional offline backup storage solution familiar to users of the Luna HSM family. Root Keys with sufficient signing capacity can be used to sign a series of "Restore Keys" generated on Luna USB-based HSMs. With the final Signing Keys signed by the Restore Keys. Once the scheme is established, the private Root Key is no longer needed and the USB-based HSMs containing the Restore Keys can be taken offline and stored for later recovery. Although only two USB-based HSMs with Restore Keys are shown in the following diagram, the actual number could be much larger to accommodate the vendor's HSM redundancy requirements.



This architecture can tolerate the loss of all production Network HSMs and all but one USB-based HSM. It also allows for the introduction new Thales HSMs later in the products' lifecycles. Upon hardware failure or decommissioning, new Network HSMs can be brought online, one of the USB-based HSMs brought out of storage, and the spare OTS private keys on a Restore Key used to sign newly generated Signing Keys.

Code Signing Performance

While LMS signatures are considerably larger than signatures generated with legacy RSA keys, they remain smaller than the highest security level ML-DSA signatures and significantly smaller than the stateless hash-based signature algorithm SPHINCS+ (see table). In use cases such as application software signing, the signature size has little impact on the solution. However, for memory-constrained devices such as IOT devices, smartcards, or embedded hardware that are validating firmware signatures, the signature size may be a factor.

	LMS (2-tier) L1,H10,W8	LMS (3-tier) L1,H10,W8	ML-DSA Level 5	SPHINCS+ 256 bits	RSA 4096 bits
Signature Size	2,912 bytes	4,368 bytes	4,595 bytes	49,856 bytes	512 bytes
Private Key Size	324 bytes	324 bytes	4,864 bytes	128 bytes	512 bytes
Public Key Size	172 bytes	172 bytes	2,592 bytes	64 bytes	512 bytes

In general, the generation of keys in an LMS/HSS tree takes significantly longer than other algorithms. However, for LMS/HSS the vendor would typically be generating a larger number of LMS/HSS keys to last the lifetime of the product during a "key ceremony" operation. The key generation process is performed on an infrequent basis and is thus usually not a major factor in selection of code signing architecture design. Performance of signature verification is heavily dependent on the characteristics of the device performing the verification. For example, a vendor of an embedded hardware device with a time budget for code verification during hardware boot-up would be concerned with signature verification performance. Alternatively, a software vendor performing a software update on a server class computer would likely have little concern regarding signature verification performance.

About Thales Trusted Cyber Technologies

Thales Trusted Cyber Technologies, a business area of Thales Defense & Security, Inc., is a trusted, U.S. provider of cybersecurity solutions dedicated to U.S. Government. We protect the government's most vital data from the core to the cloud to the edge with a unified approach to data protection. Our solutions reduce the risks associated with the most critical attack vectors and address the government's most stringent encryption, key management, and access control requirements.

For more information, visit www.thalestct.com

THALES

Building a future we can all trust

Contact us

For all office locations and contact information, please visit
thalestct.com/contact-us

thalestct.com