FIPS 205 STATUS UPDATE

John Kelsey*
NIST and KU Leuven

* David Cooper did most of the work here, I'm just taking the credit.

National Institute of Standards and Technology
U.S. Department of Commerce

SPHINCS+ --> SLH-DSA

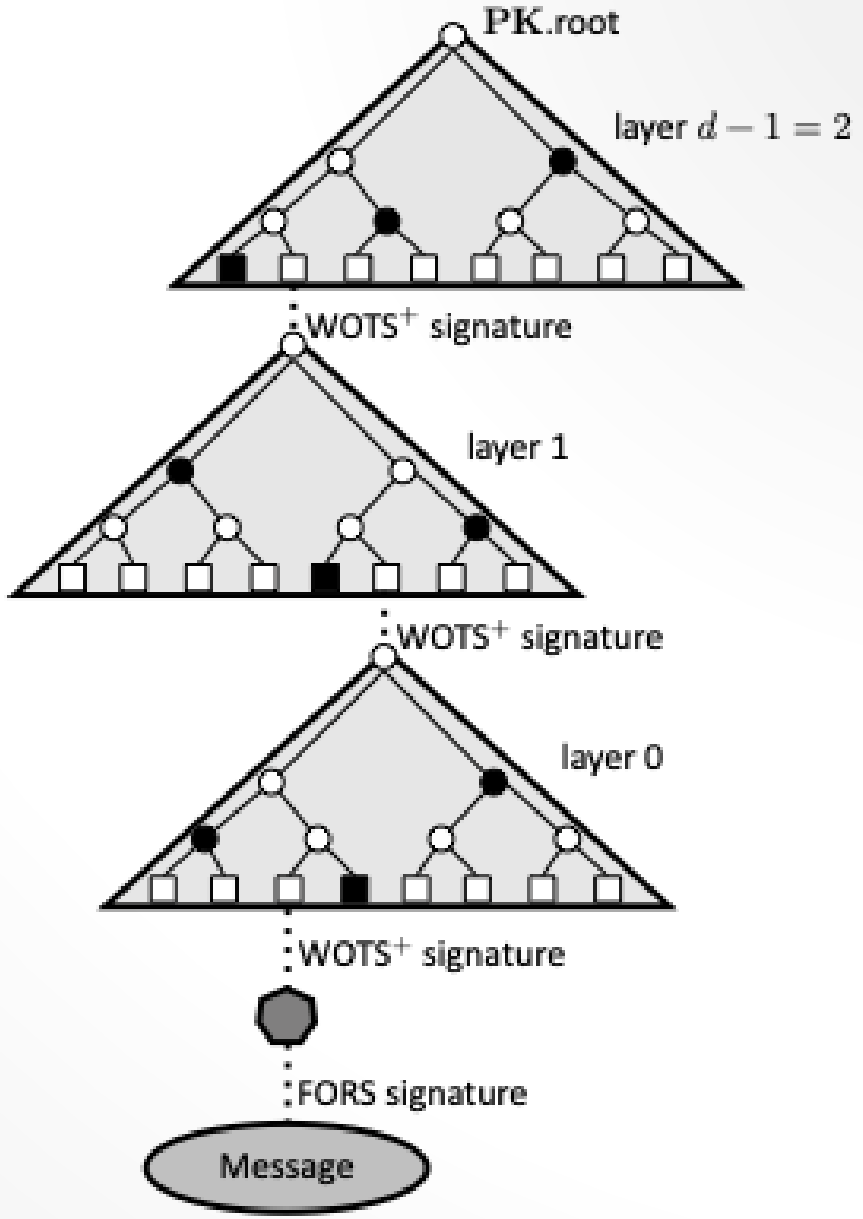
Stateless Hash-Based Signature

Security for up to 2^{64} signatures

Signatures are very big

- 8 KiB – 50 KiB

Signing slow, verifying faster



SUMMARY

- Most comments on FIPS 205 editorial.
  - Few technical changes proposed.
- Changes made to address editorial issues.
- Planned technical changes unlikely to require changes to cryptographic modules.

TOO MANY PARAMETER SETS?

SLH-DSA has 12 parameter sets

- Different security categories
- SHA256 vs SHAKE
- Slow/small sigs vs fast/big sigs

Many comments wanted fewer parameter sets

- Eliminate SHA-2 for categories 3 and 5
- Eliminate fast parameter sets
- Keep all 12 parameter sets

Given lack of consensus, NIST plans to leave all 12

SMALLER SPHINCS+

Some commenters requested adding smaller parameter sets

- Fewer signatures allowed (like 2^{20} or 2^{30} instead of 2^{64})
- Result: smaller and faster sigs

Planning to address this in a separate publication

- No change to SLH-DSA spec

PARAMETER SET CHANGES?

- One commenter asked why SLH-DSA-SHA2-{192,256}{s,f} use a combination of SHA-256 and SHA-512 rather than just SHA-512.
  - No changes planned. This decision was made by the SPHINCS+ team.

- One commenter proposed using tweaked versions of SHA-256 and SHA-512 to improve performance.
- One commenter proposed using TurboSHAKE256 instead of SHAKE256 to improve performance.
  - No changes planned.

DETERMINISTIC API FOR KNOWN-ANSWER TESTING

Cryptographic Algorithm Validation Program (CAVP) requires known answer testing.

Changed internal functions to take randomness as input

- slh_keygen_internal(SK.seed, SK.prf, PK.seed)
- slh_sign_internal(M, SK) or slh_sign_internal(M, SK, opt_rand)
- slh_verify_internal(M, SIG, PK)

Requirements

- Testing: Known input for randomness -> known answers
- Production: Randomness comes from RNG in module

PRE-HASH

Several comments received about including a pre-hash option:

- Require pre-hash for all signatures
- Specify domain separation and/or different OIDs for pure and pre-hash
- Don't allow pre-hash in the FIPS; pre-hash can be implemented at the application level where needed.

NIST plans to specify both pure and pre-hash signatures

- Domain separation between pure and pre-hash versions
- Incorporate OID of external hash
- External hash output must be at least 2x security strength
- See the discussion Friday afternoon for more detail

NOTE: Plan is to allow pre-hash for all PQ signatures

WRAPUP

Final Document out Summer 2024 (we hope!)

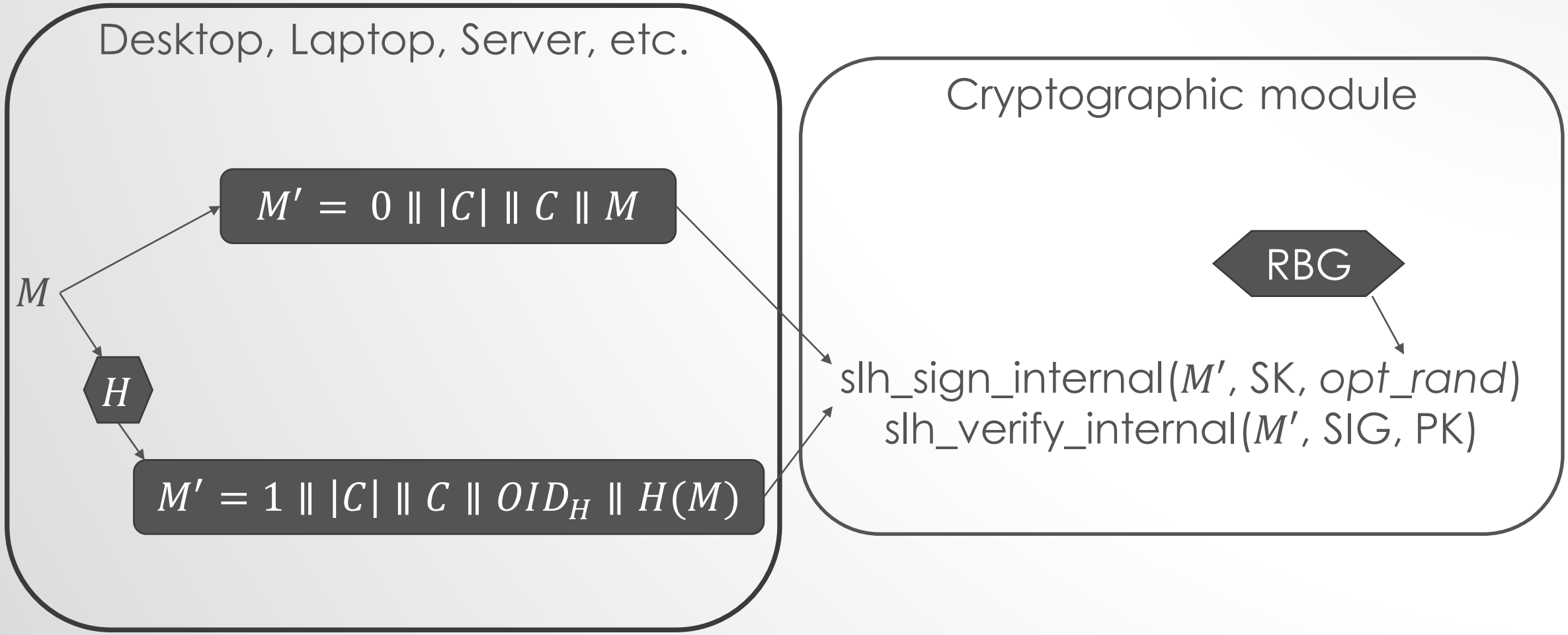
- No major changes to algorithm
- Lots of parameter sets (12)
- Only the 2^{64} signature version
  - Smaller versions coming soon
- New functions to allow known-answer testing
- Mechanisms for handling pre-hashing (see discussion Friday!)

Questions?

EXTRA SLIDES

PRE-HASH

Example separation of functionality:



PRE-HASH

- When defining OIDs, NIST plans to limit the number of options for pre-hash function (e.g., one per parameter set) in order to avoid combinatorial explosion.
  - OIDs will be on CSRC web site, not in the FIPS.
- Initial idea for SLH-DSA pre-hash OIDs:
  - SLH-DSA-SHA2-{128,192,256}{s,f}-with-SHA512-prehash
  - SLH-DSA-SHAKE-128{s,f}-with-SHAKE128-prehash
  - SLH-DSA-SHAKE-{192,256}{s,f}-with-SHAKE256-prehash
  - Prefer SHA-512 over SHA-256 for SLH-DSA-SHA2-128{s,f}… since SHA-512 is faster on many platforms.
- This topic will be discussed further tomorrow afternoon in the pre-hash panel.