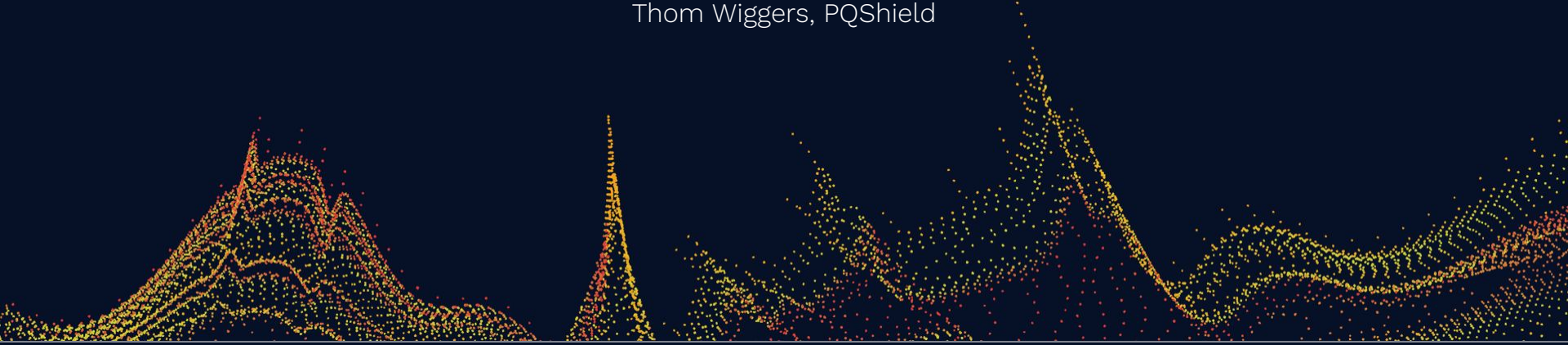
PQSHIELD

Hash-based Signatures: State and Backup Management

draft-wiggers-hbs-state-00

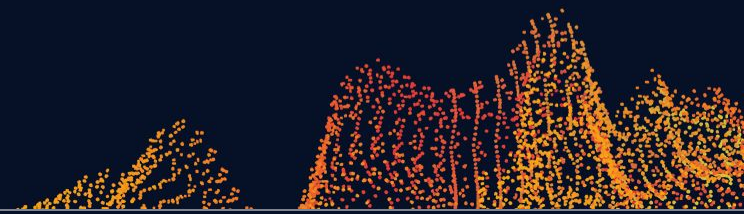
Thom Wiggers, PQShield



PQSHIELD

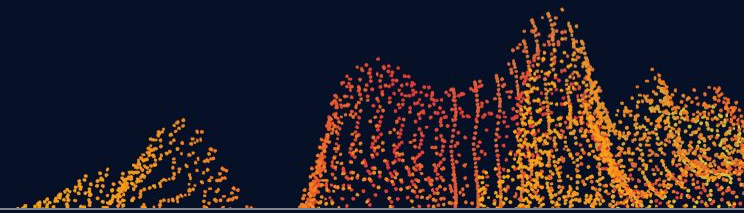
Guidance for state management (repeat from IETF119)

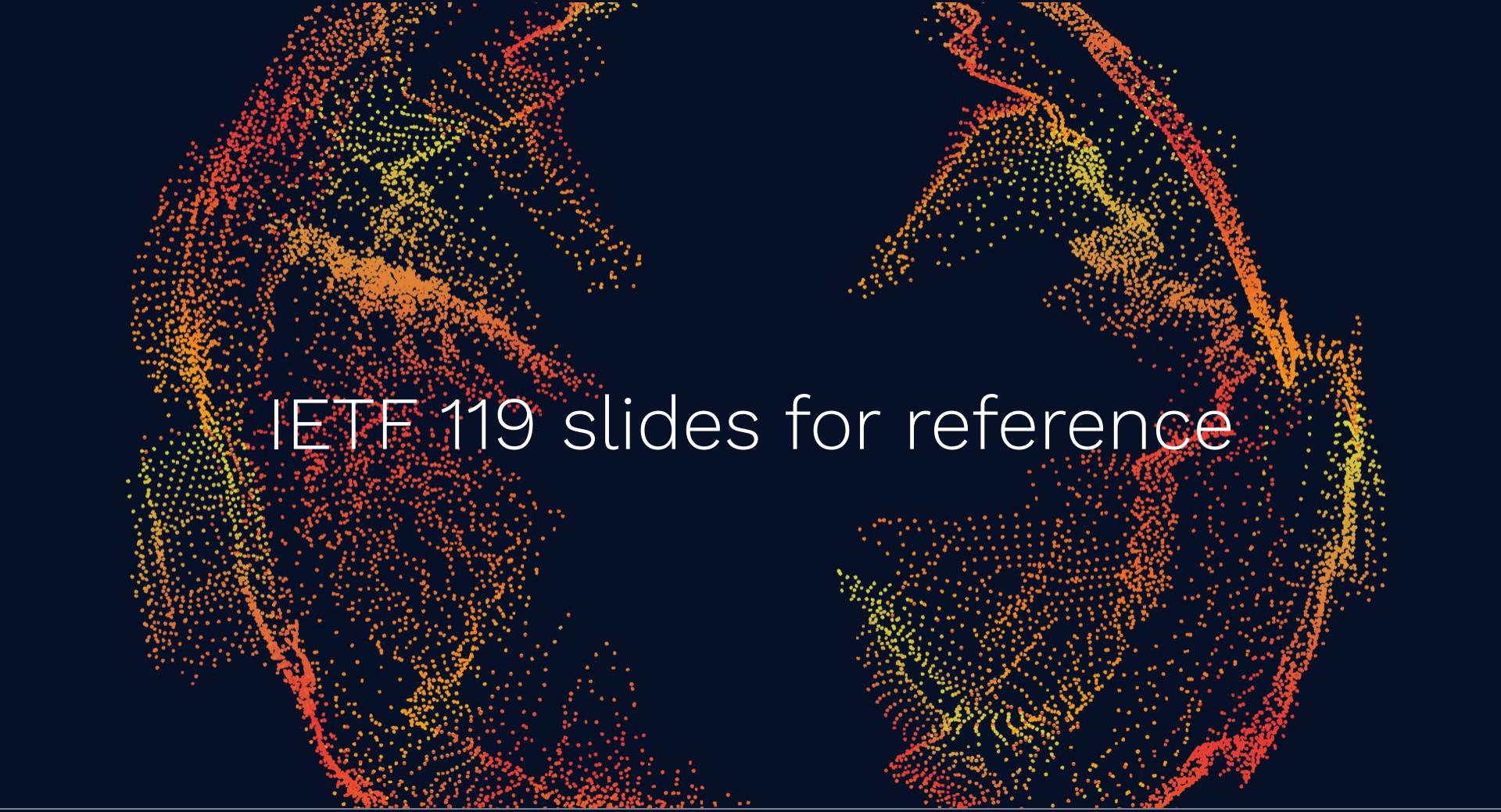
- Dealing with state is hard
- Dealing with state is scary
- “Thou MUST NOT use a key more than once” — but how?
  - You SHOULD use ~~SPHINCS+~~ SLH-DSA if possible
  - You SHOULD probably use an HSM
- How do you reliably deploy S-HBS schemes?
- And what about backups?



Discussion on mailing list

- People seem to agree this is worth documenting
- Possible additions to the current document:
  - When should S-HBS be considered in the first place? (Issue #2)
  - More approaches?





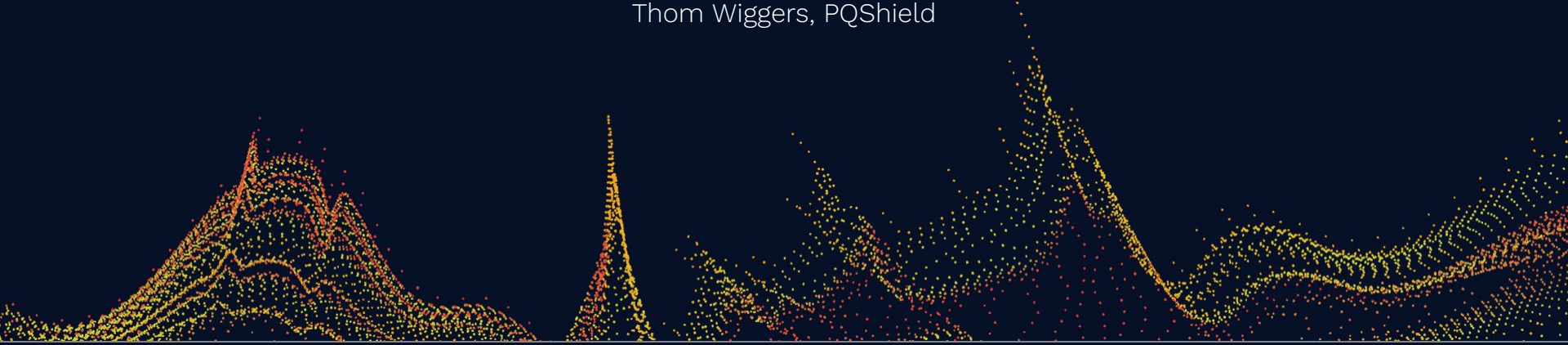
IETF 119 slides for reference

PQSHIELD

Hash-based Signatures: State and Backup Management

draft-wiggers-hbs-state-00

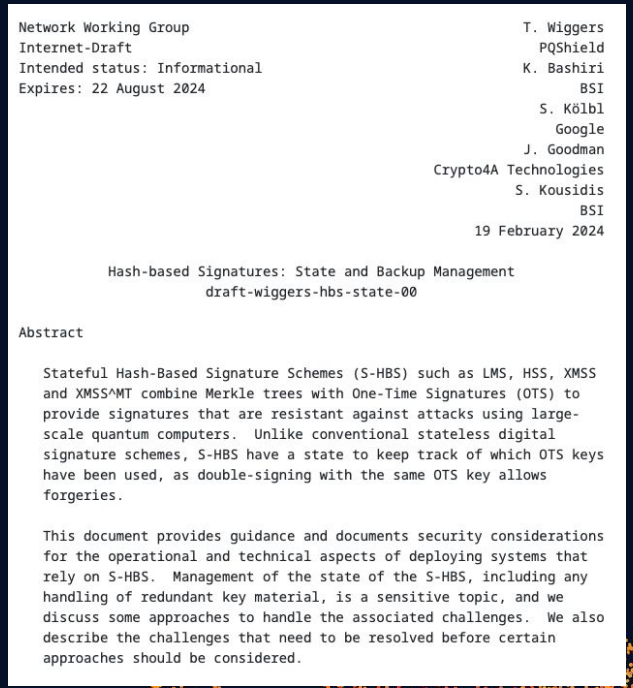
Thom Wiggers, PQShield



PQSHIELD

Joint work

Many thanks to Kaveh Bashiri, Stefan Kölbl, Jim Goodman, Stavros Kousidis, and Bruno Coulliard.



Network Working Group — T. Wiggers, PQShield
Internet-Draft — K. Bashiri, BSI
Intended status: Informational — S. Kölbl, Google
Expires: 22 August 2024 — J. Goodman, Crypto4A Technologies
S. Kousidis, BSI
19 February 2024

Hash-based Signatures: State and Backup Management
draft-wiggers-hbs-state-00

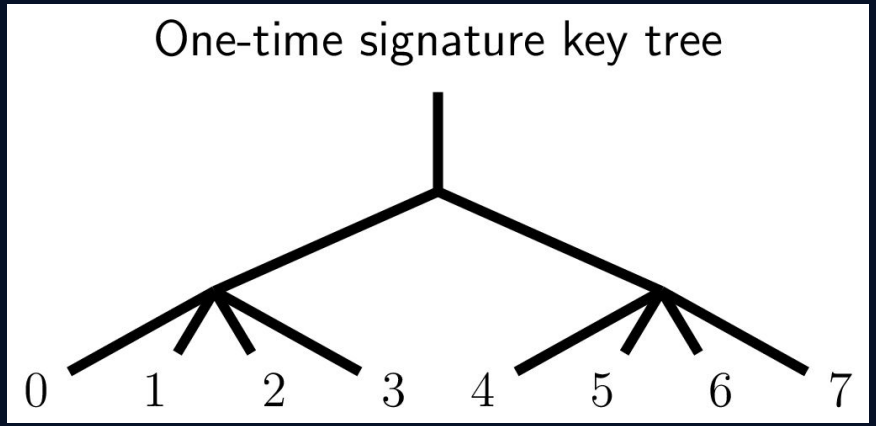
Abstract

Stateful Hash-Based Signature Schemes (S-HBS) such as LMS, HSS, XMSS and XMSS^MT combine Merkle trees with One-Time Signatures (OTS) to provide signatures that are resistant against attacks using large-scale quantum computers. Unlike conventional stateless digital signature schemes, S-HBS have a state to keep track of which OTS keys have been used, as double-signing with the same OTS key allows forgeries.

This document provides guidance and documents security considerations for the operational and technical aspects of deploying systems that rely on S-HBS. Management of the state of the S-HBS, including any handling of redundant key material, is a sensitive topic, and we discuss some approaches to handle the associated challenges. We also describe the challenges that need to be resolved before certain approaches should be considered.

Stateful hash-based signatures

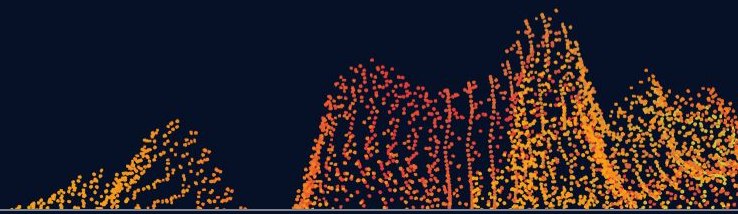
- A list of one-time use signature keys
- Key reuse ⇒ game over
- Keeping track of which signatures have been consumed is super important.
  - We call this the state



PQ SHIELD

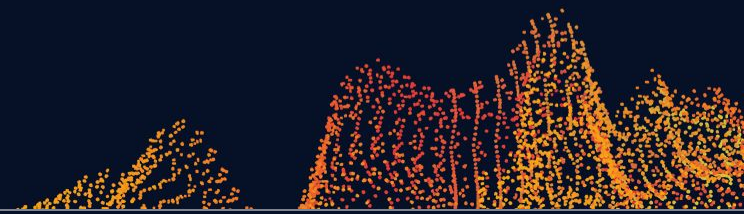
Guidance for state management

- Dealing with state is hard
- Dealing with state is scary
- “Thou MUST NOT use a key more than once” — but how?
  - You SHOULD use ~~SPHINCS⁺~~ SLH-DSA if possible
  - You SHOULD probably use an HSM
- How do you reliably deploy S-HBS schemes?
- And what about backups?



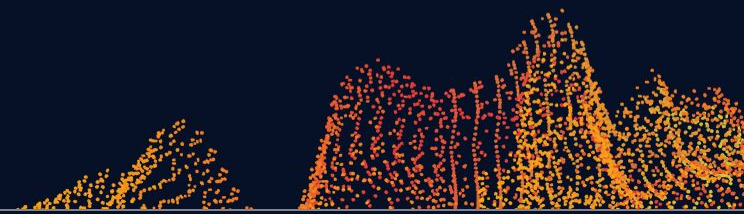
Contents of the draft

- Terminology
- Requirements
- Operational considerations
- Some potential solutions for state management and backups
- Evaluation of certain approaches for specific setups
- Both within and beyond SP800-208's key export ban

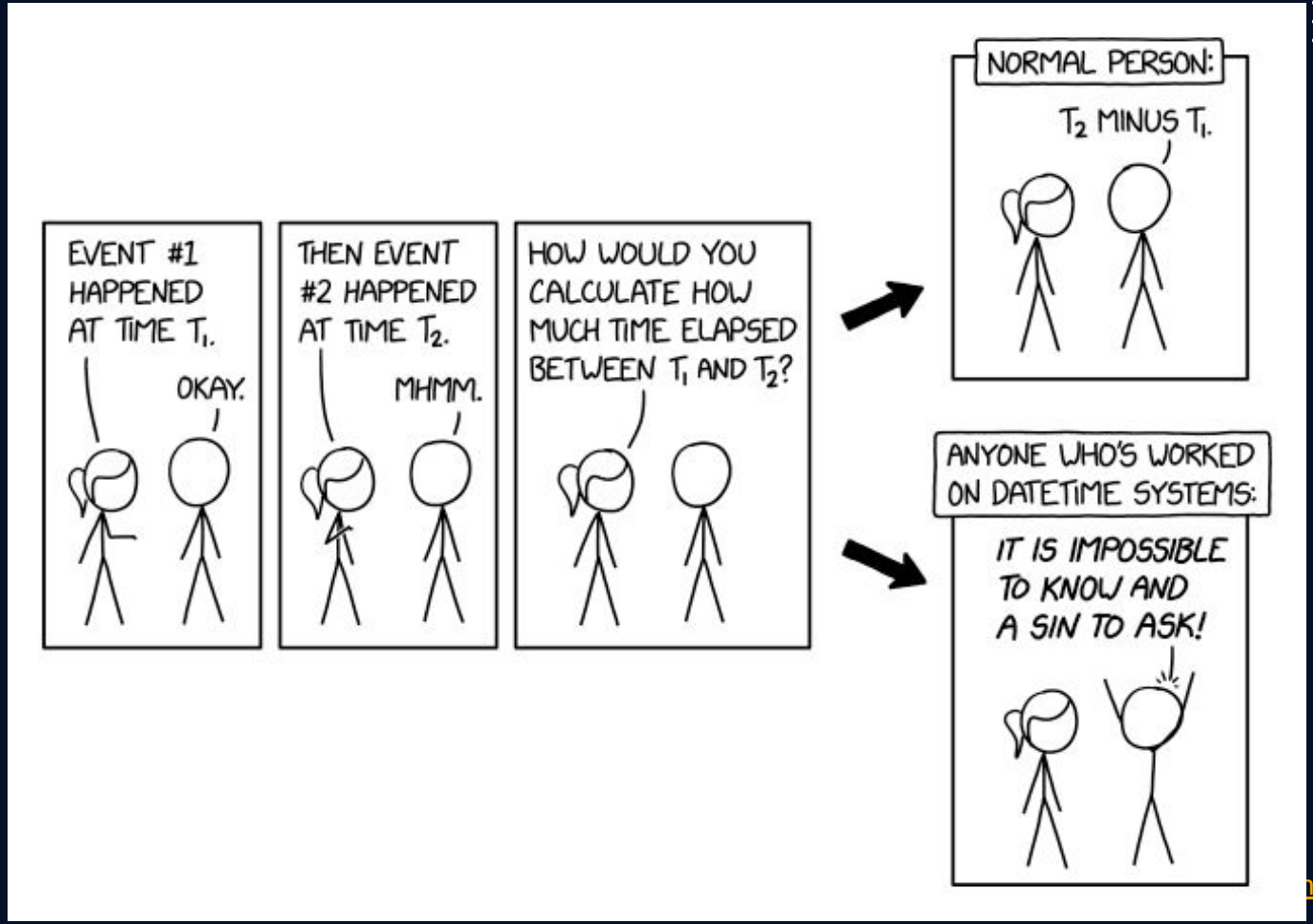


Example: “Can’t we just use time?”

- Divide time into epochs
- Only allow one signature per epoch
- Clock makes sure that you know which states are okay?
- Profit?



https://xkcd.com/1883/



https://xkcd.com/2867/

“Our HSM vendor told us to please not roll out any configuration changes tomorrow”

-- My friend who works at a major bank,
28 february 2024

5.8. Time-based State Management

[…]

Any time-based approach has a very strict reliance on accurate time-keeping and synchronization of clocks. In particular, we identify that at least the following engineering-related challenges need to be considered:

[16 BCP14 keywords follow]