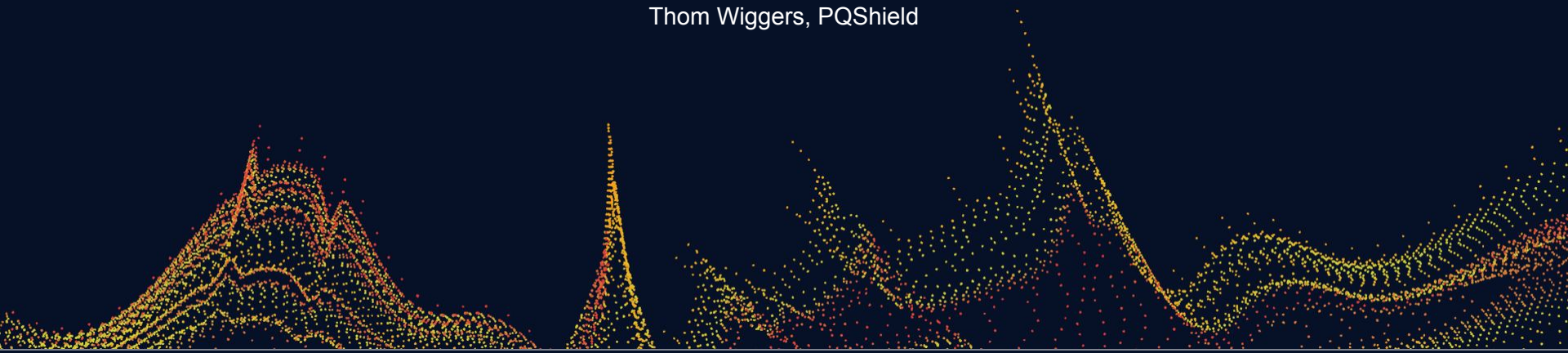
PQSHIELD

Hash-based Signatures: State and Backup Management

An IETF draft to give guidance and handholds to those designing and operating S-HBS-based systems

Thom Wiggers, PQShield



Network Working Group
Internet-Draft
Intended status: Informational
Expires: 22 August 2024

T. Wiggers
PQShield
K. Bashiri
BSI
S. Kölbl
Google
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.

Documenting 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?

§3 — Operational considerations

[...] Hence, archival procedures used for traditional trust infrastructures MUST be amended/redesigned to be used as viable options. [...]

[...] any resilient state management system SHOULD also provide some means to verify the integrity of these long lived backups [...]

[...] ensure the operators know how to execute the necessary recovery procedure(s). [...]

§4 — Requirements of state management

An incomplete list of some of the things that will trip you up

- Hard drive caches
- Virtual Machine Cloning
- Glitches

Using dedicated cryptographic hardware is RECOMMENDED to enforce these requirements, ensure correct behavior and handle the complexity of state management. In particular, this enables implementing rollback resistant counters which can be difficult to achieve in a software-only fashion.

§5-6 — Potential solutions

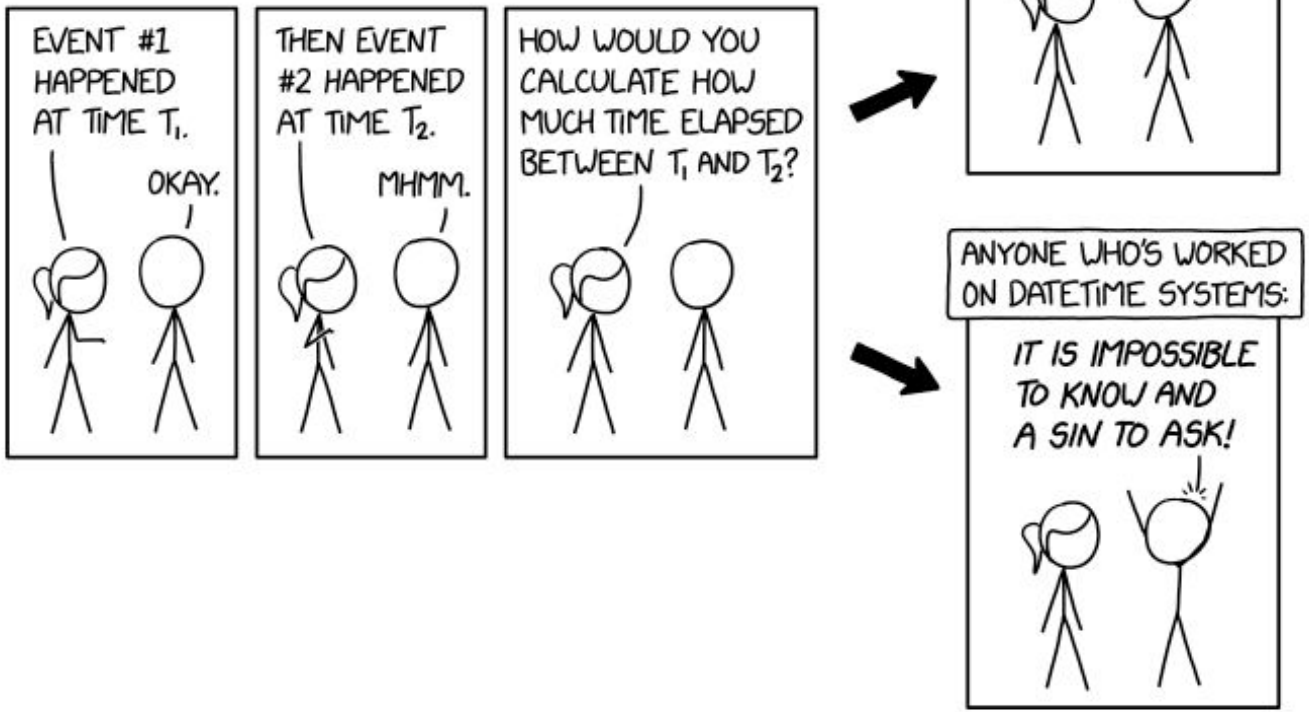
- We list some of the potential approaches, as well as challenges, for dealing with state
- §6 includes things that go beyond what SP800-208 allows
  - What should you pay attention to when you export your secret keys?
- Includes some commonly suggested approaches, including:
  - Approaches discussed in SP800-208
  - Splitting parts of keys between signing devices
  - Transferring between signing devices while making sure to delete from old devices
  - Pre-assigning one-time keys to to-be-signed messages (e.g., software version numbers)
  - Basing which key to use on a clock

§5-6 — Potentia

- We list some of the pot …s, for dealing with state
- §6 includes things tha
  - What should you pay at
- Includes some commo
  - Approaches discussed
  - Splitting parts of keys b
  - Pre-assigning one-time … numbers)
  - Basing which key to us

§5-6 -

- We li ith state
- §6 in
  -
- Inclu
  -
  -
  -
  -



§5-6

- We l
- §6 i
  -
- Incl
  -
  -
  -
  -

state

Make this document the best it can be

Let us know your thoughts, reviews, ideas and submit additional relevant XKCDs at:

- https://datatracker.ietf.org/doc/draft-wiggers-hbs-state/
- https://github.com/hbs-guidance/draft-hbs-state

We will also present at IETF 119 in the PQUIP (post-quantum usage in protocols) meeting.