Hash-based Signatures: State and Backup Management

draft-wiggers-hbs-state-00
Thom Wiggers, PQShield
Joint work

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

Network Working Group
Internet-Draft
Entitled 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.
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
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?
...then, after our drones take control of the cities, we will detonate the devices. California will break off from the mainland and drift out to sea!

How far out to sea? Will it put any of the cities in the UTC-9 time zone?

What? I don’t know.

One request: can we make sure this doesn’t happen during the daylight saving changeover?

You can tell when someone’s been a programmer for a while because they develop a deep-seated fear of time zone problems.

https://xkcd.com/1883/
EVENT #1 HAPPENED AT TIME T₁.
OKAY.

THEN EVENT #2 HAPPENED AT TIME T₂.
MHMM.

HOW WOULD YOU CALCULATE HOW MUCH TIME ELAPSED BETWEEN T₁ AND T₂?

NORMAL PERSON:
T₂ MINUS T₁.

ANYONE WHO'S WORKED ON DATETIME SYSTEMS:
IT IS IMPOSSIBLE TO KNOW AND A SIN TO ASK!

https://xkcd.com/2867/
Leap Year Software Glitch Closes Fuel Pumps Across New Zealand

By Matthew Brockett
29 februari 2024 at 00:46 CET
“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]
contributions very welcome