Motivation

• There is much discussion about the advent of quantum computers

• Timeline for practical QCs that break current public key algorithms unsure, estimations vary from 15 to 50 years

• Post-Quantum Cryptographic algorithms are seeing a lot of development (e.g. NIST standardisation effort, which is in the final phase)

• There is momentum to start deployment of PQC algorithms (i.e. expect requirements for PQC support to appear in a government tender near you some time soon)
Motivation

• But: DNSSEC signatures have an effective zero-year shelf-life

• So why care about PQC for DNSSEC?

• Answer: standardisation, implementation and transition cycle is long (can easily be 10+ years, cf. e.g. Elliptic Curve algorithms)

• Challenges with “new” PQC algorithms: long-term security, unfavourable parameters for use in DNSSEC

• We believe we need a “safe fallback” that is standardised
Stateful HBS 101

• First proposed by Ralph Merkle, **constructed using Merkle trees**

• Stateful hash-based signature schemes are **considered to have very strong security** (if a secure cryptographic hash function is used; they essentially inherit their security properties from the hash function)

• Workings are **well-understood**, very unlikely to encounter “cryptoanalytical surprises” that suddenly **break** the **security** of HBS schemes

• Remain **secure in the face of** powerful **quantum computers**

• Note: stateless HBS also exist, e.g. SPHINCS+ being considered by NIST
Signature Composition: <OTS Signature on Data><Authentication Path>

<Authentication Path>: consists of Merkle tree node hashes that are used as an input along with companion sibling nodes to calculate parent node hashes

Example for signature on RRset by OTS Public Key:

<OTS Signature on data> = Signature on data (RRset') created using OTS Private Key' corresponding to OTS Public Key'

<OTS Public Key> = OTS Public Key'

<Authentication Path> = Leaf^2 | Right Child
Limitations

• Can create a **finite number of signatures with a signing key**, private key consists of a collection of one-time signing (OTS) keys

• **Essential to keep state** (re-use of the same OTS breaks security!)
  —> challenge for online signers and distributed setups*

• **Signatures are (very) large** (≥ 2.5 kB) - public keys are small (~70 bytes)
  —> requires EDNS0, and arguably TCP transport

• Therefore: **not the preferred option** for DNSSEC, **but a safe fallback**
  —> given timelines, we argue standardising a safe fallback now is needed
Side step: online/multi-signer

Hierarchical Tree Structures for Stateful Hash-Based Signature Schemes

- OTS Private Keys for levels above the bottom level sign roots of child trees
- Bottom level OTS Private Keys sign data
Draft status

• Draft proposes how to use stateful HBS schemes in DNSSEC

• Three HBS algorithms included in the draft:
  • HSS/LMS [RFC8554]
  • XMSS [RFC8391]
  • XMSS^MT [RFC8391]

• First “complete” draft —> interest in adoption?

• NLnet Labs have done a proof-of-concept implementation in Unbound
Follow-up work

• Considering a draft on implementation considerations of stateful HBS in DNSSEC
  • Interoperability across implementations
  • Trade-offs of hierarchical trees
  • Parameter choices
  • Transport considerations
  • ...?
Thoughts, questions, comments?

Our thanks go out to the following people who contributed to the development of the draft:
Dave Blacka, Jim Goodman, James Gould, Joseph Harvey, Scott Hollenbeck,
Russ Housley, Burt Kaliski, Swapneel Sheth, Sean Turner, Duane Wessels