PQC TSS and PQC TPM

Fraunhofer SIT

a prototype

Andreas Fuchs, 19th October 2018

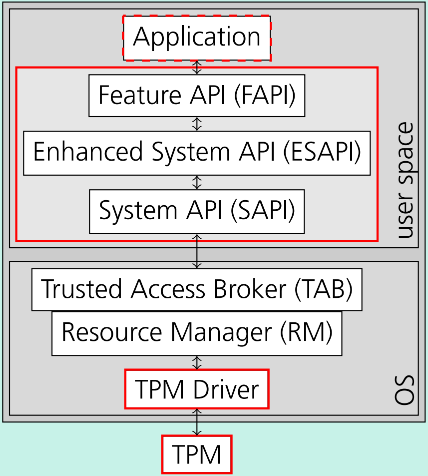
Introduction

- Due to the thread of quantum computers, we expect that asymmetric cryptography will transition to Post-Quantum Cryptography in the next ten years.
- PQC-schemes tend to have larger resource requirements than RSA, DH and ECC.
- In particular for resource restricted embedded systems, PQC might be hard to implement efficiently.
- TPMs have highly restricted resources.

⇒ Investigate the usability of PQC for TPMs.

Introduction

Communication between Application and TPM:



Hash-based Signature Schemes

Introduction

Properties:

- Hash functions as only building block.
- Well understood, high security guarantees.
- Limited number of signatures per public key!
- Some schemes need to maintain a state!

Examples:

- Stateful:
  - LMS, XMSS**XMSS**
- State-less:
  - SPHINCS, SPHINCS$^+$

Code-based Encryption Schemes

Introduction

Properties:

- Use *error correcting codes* for cryptography.
- Studied since 1978, security depends on code family.
- Conservative schemes require large keys!
- Decoding errors may enable attacks (for some code choices)!

Codes for the McEliece/Niederreiter system:

- binary Goppa
- GRS, Reed-Muller, BCH
- LDPC, QC-MDPC**QC-MDPC**

Lattice-based Encryption Schemes

Introduction

Properties:

- Use hard *lattice problems* for cryptography.
- Plenty of security proofs.
- Choice of parameters not yet well understood!
- Very promising, efficient schemes.

Examples:

- KEX: New Hope,
- KEM: NTRU, qTESLA, Kyber**Kyber**

Post-Quantum TPM

Approach

Simulation:

- Extend an existing TPM simulator by adding PQC schemes.
- Test functionality.

Prototype:

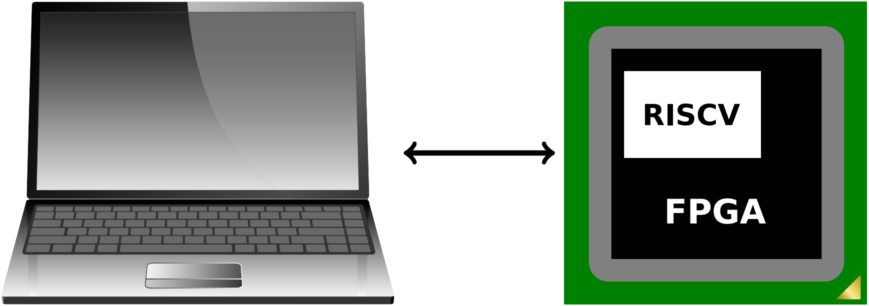
- Transfer the TPM simulator to an embedded RISC-V processor.
- Measure performance and memory demand.

Optimization (ongoing work):

- Optimize TPM “simulator” software.
- Provide hardware accelerators for PQC primitives.

Post-Quantum TPM

Demonstration



Post-Quantum TPM

Performance

Scheme	Key Generation		Encryption		Decryption	
	Cycles	Time	Cycles	Time	Cycles	Time
Kyber	35.7×10^6	0.715 s	44.5×10^6	0.891 s	9.36×10^6	0.187 s
QcBits	231×10^6	4.63 s	8.34×10^6	0.167 s	167×10^6	3.34 s

Scheme	Key Generation		Verification		Signing	
$h = 10$	Cycles	Time	Cycles	Time	Cycles	Time
XMSS	209×10^9	4190 s	130×10^6	2.60 s	209×10^9	4190 s
XMSS HW*	311×10^6	6.22 s	589×10^3	0.0118 s	1.77×10^6	0.0354 s

*estimation based on experiments

time at 50 MHz

Limitations of the TPM 2.0 Specification

Standard TPM Parameters

IO Buffer Size:

The default maximum size of the IO buffer is 4096 Bytes.
(This limitation is vendor-specific and not fixed in the specification.)

The default buffer size allows the following parameters:

- XMSS (SHA256):
  - Tree height: $24 \Rightarrow 2^{24} = 16,777,216$ signatures.
  - **Limitation: computing time (key gen and sign).**
  - **NVRAM of TPM is perfect for storing state!**
  - NVRAM size limits number of keys.
    $\Rightarrow$ Increase NVRAM size if more keys are required.
- QC-MDPC:
  - Buffer size fine for 80-bit and 128-bit security parameters.
  - Data structures for 256-bit security parameters too large.
    $\Rightarrow$ Double IO buffer size.

Fraunhofer SIT

Limitations of the TPM 2.0 Specification

Limitations of the Specification

Additional Commands for XMSS:

Optimized tree traversal algorithms (for signing) require to cache inner tree nodes in order to avoid recomputing the entire tree for each signature.

Solutions:

- Store caching data in NVRAM.
  Limited resource!
- Use pseudo-persistent storage outside the TPM.
  $\Rightarrow$ Requires additional commands to send and retrieve cache data.
  XMSS state (next leaf index) remains in NVRAM.
  Data on inner tree nodes is pseudo-persistently cached.
  Drop outdated caching data!

Conclusion

Take away:

- The TPM 2.0 specification is sufficiently agile for PQ crypto.
- Some limits on computation and communication need to be lifted.
- Some additional commands are required for efficiency.
- Hash-based signature schemes may be enabled by firmware updates.
  $\Rightarrow$ No need for new hardware.
- Fast and efficient lattice-, code-, or $\mathcal{MQ}$-based implementations require
  new crypto accelerators. $\Rightarrow$ New hardware required.

Thank you!

Kontakt Information

Andreas Fuchs
Ruben Niederhagen

Cyber-Physical Systems Security

Fraunhofer Institute for
Secure Information Technology

Addresse: Rheinstraße 75
64295 Darmstadt
Germany
Internet: www.sit.fraunhofer.de/en/pqc-tpm/

Telefon: +49 6151 869-228
Fax: +49 6151 869-224
E-Mail: andreas.fuchs@sit.fraunhofer.de

Image Sources

Title Page:

©IBM Research, CC BY-ND 2.0

`https://creativecommons.org/licenses/by-nd/2.0/`

Clip Art (slide 7):

Public Domain

`https://creativecommons.org/publicdomain/zero/1.0/`