Machine-checking post-quantum cryptography

Andreas Hülsing
Applied & Provable Security Group
Eindhoven University of Technology

How do you ensure that a cryptographic scheme is hard to break?

Traditional Answer: Cryptanalysis

Have many smart people try to break it.

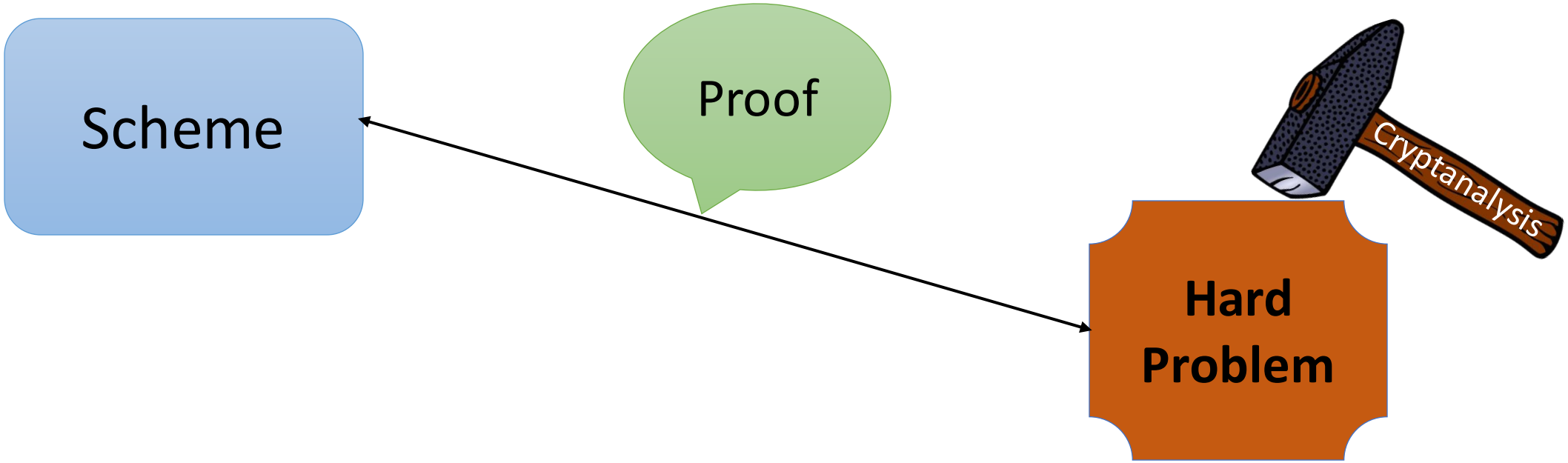
Does not scale!

- NIST: 64 candidates
- NIST signature on-ramp: 40 candidates
- KpqC: 16 candidates
- China, Russia, …

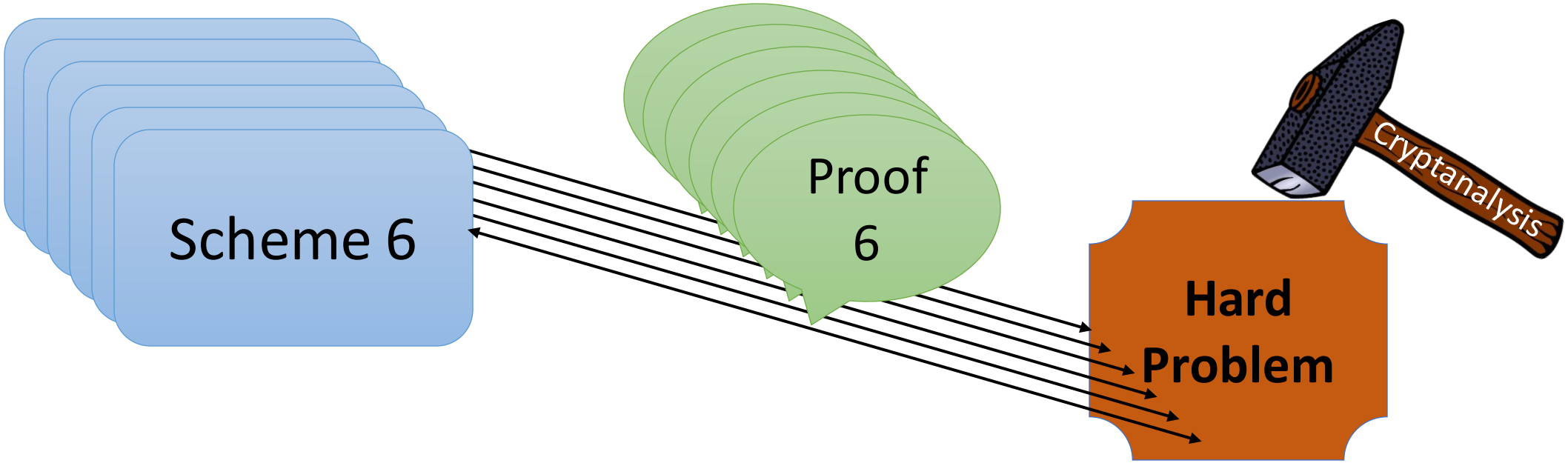
Who is supposed to cryptanalyze all of these?

What about protocols?

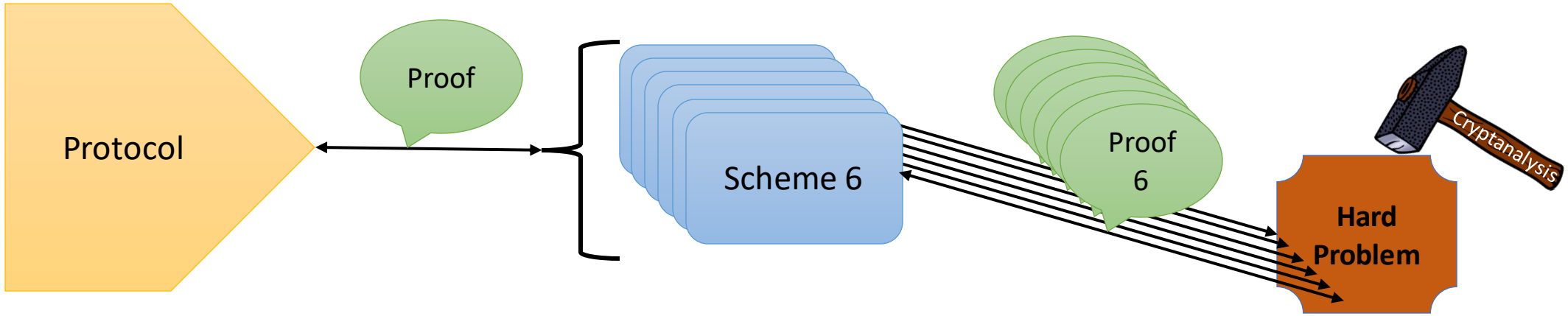
The Role of Security Proofs in Cryptography



The Role of Security Proofs in Cryptography



The Role of Security Proofs in Cryptography



Beautiful idea, but who checks the proofs?

- The reviewers?
  - Review load per reviewer at top tier IACR: 16+ papers, 30 pages main body, often 50+ pages with appendix
- The community?
  - eprint 2023:
    - **1703** papers, of which 512 tagged protocols, 264 tagged PKC (ignoring foundations, applications,…)
    - 2919 IACR members in 2023

Does that work?

Bugs in proofs / proof is wrong.

- XMSS & SPHINCS+:
  - Kudinov, Kiktenko, and Fedorov 2020: Bug in proof of tight security bound for SPHINCS+.
- Dilithium (and many other schemes):
  - Flaw in the HVZK proof step for Fiat-Shamir with aborts.
    [Barbosa, Barthe, Doczkal, Don, Fehr, Grégoire, Huang, Hülsing, Lee, and Wu. Fixing and Mechanizing the Security Proof of Fiat-Shamir with Aborts and Dilithium. CRYPTO 2023.]

All these are fixed now!

Does that work?

Bugs in instantiation / proof does not apply:

- XMSS & SPHINCS+:
  - Peickert 2018: Tight-security proof does not apply to instantiations.
  - Antonov 2022: SHA256 instantiation of SPHINCS+ does not achieve full conjectured security on required security properties.
- Kyber:
  - FO-transform used by Kyber is not the one with a security proof
  - Kyber round 1: Proof does not apply when using key compression

- **All these are fixed now!**

Proof failure modes

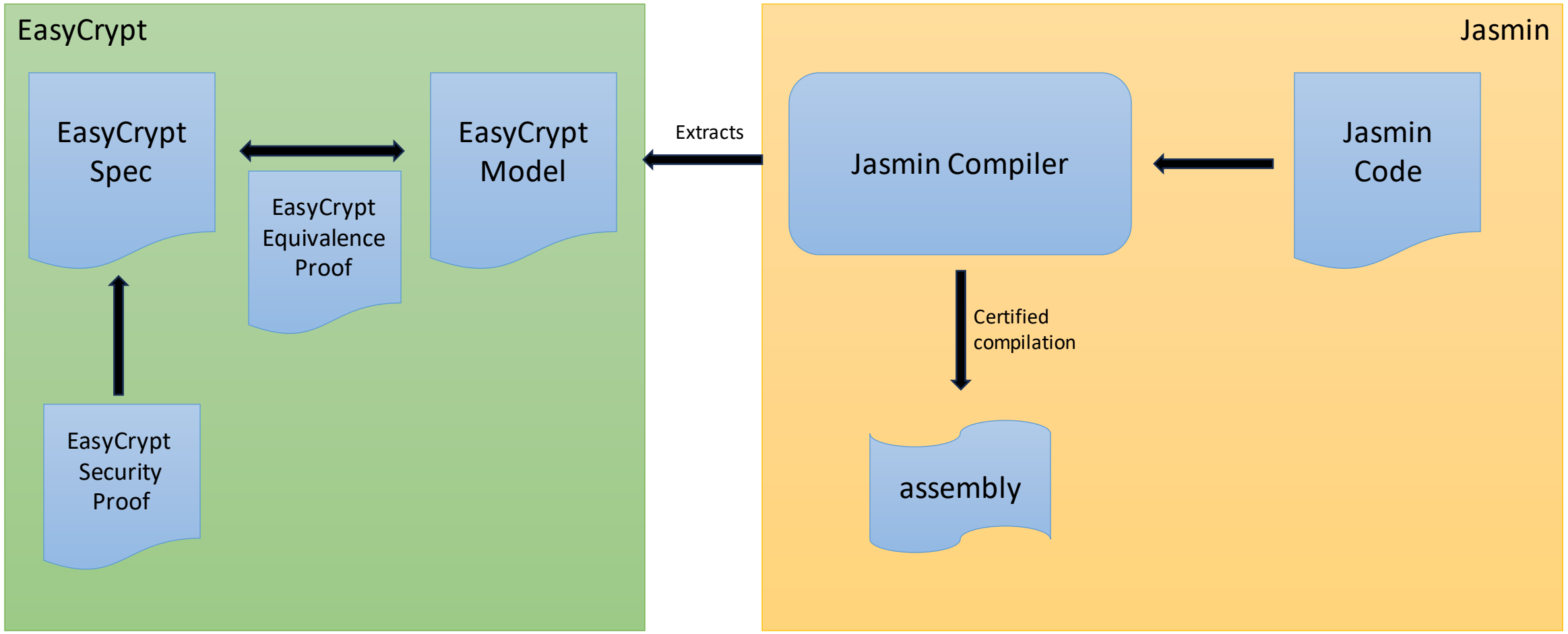
(Taken from Peter Schwabe)

- Proof is wrong
  - Theorem is correct
  - Theorem is also wrong
    - Scheme is still (possibly) secure
    - Scheme is efficiently broken
- Proof doesn’t apply to the scheme
  - Proof correct, but theorem “insufficient”
  - Example: attack hides in non-tightness
- Proof (and possibly theorem) too vague
- Theorem and proof correct, but not very useful
  - “A is secure if A is secure”

How to solve this?

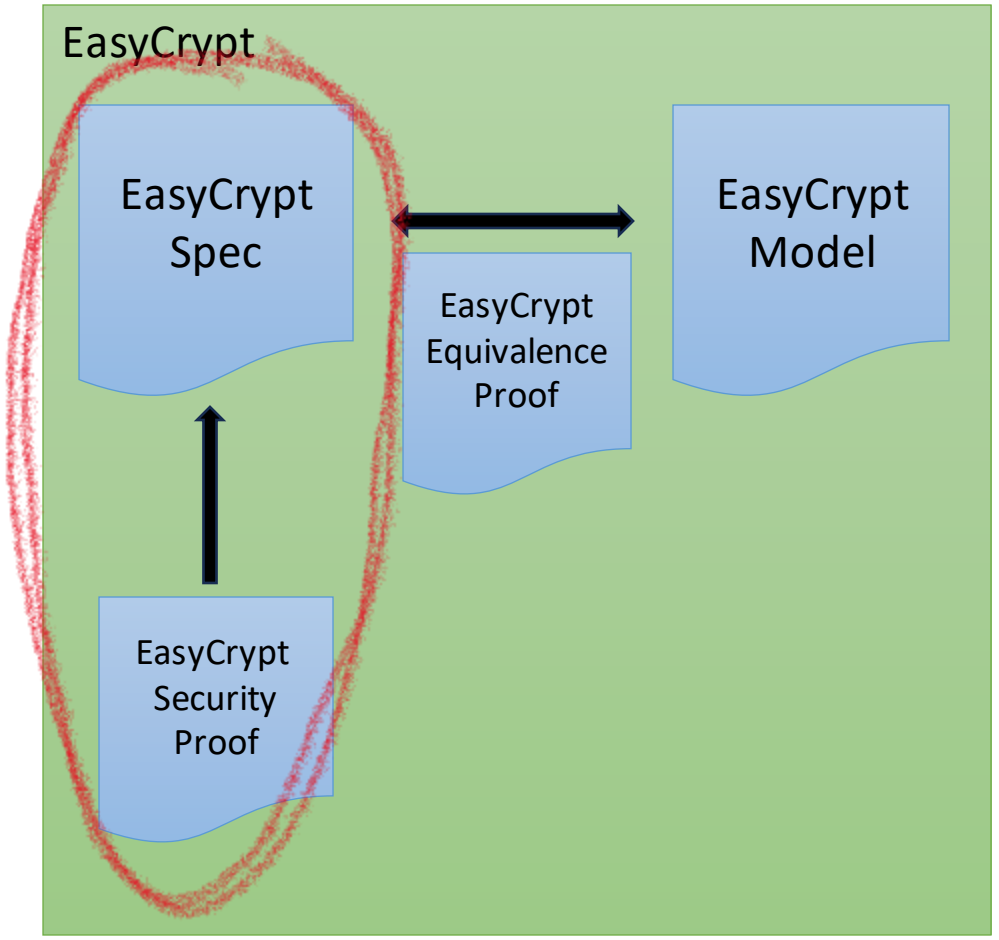
- Effort to formally verify crypto
- Goal: verified PQC ready for deployment
- Three main projects:
  - EasyCrypt proof assistant
  - Jasmin programming language
  - Libjade (PQ-)crypto library
- Core community of ≈ 30–40 people
- Discussion forum with >180 people

The toolchain



Results (Security proofs)

- Barbosa, Barthe, Fan, Grégoire, Hung, Katz, Strub, Wu, and Zhou. EasyPQC: Verifying Post-Quantum Cryptography. ACM CCS 2021
- Hülsing, Meijers, and Strub. Formal Verification of Saber's Public-Key Encryption Scheme in EasyCrypt. CRYPTO 2022
- Barbosa, Barthe, Doczkal, Don, Fehr, Grégoire, Huang, Hülsing, Lee, and Wu. Fixing and Mechanizing the Security Proof of Fiat-Shamir with Aborts and Dilithium. CRYPTO 2023
- Barbosa, Dupressoir, Grégoire, Hülsing, Meijers, and Strub. Machine-Checked Security for XMSS as in RFC 8391 and SPHINCS+. CRYPTO 2023

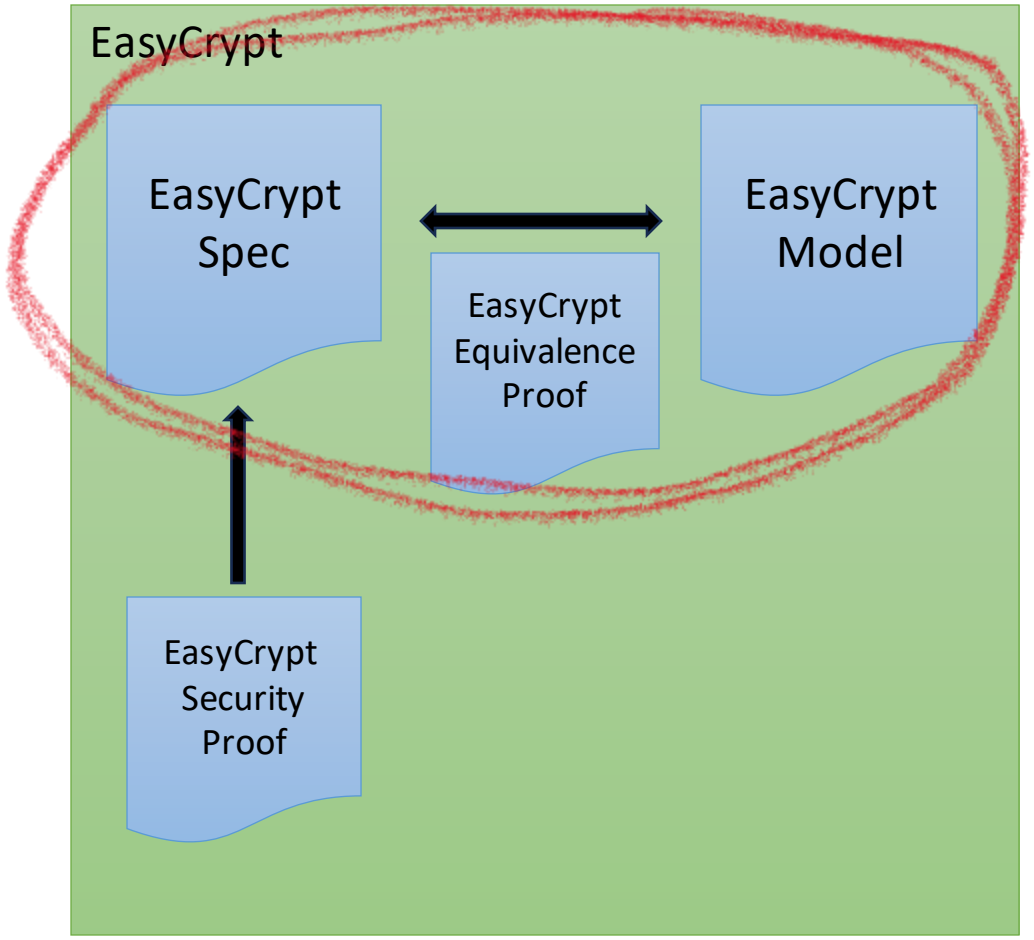


Impact

- Proof is wrong
  - Theorem is correct
  - Theorem is also wrong
    - Scheme is still (possibly) secure
    - Scheme is efficiently broken
- Proof doesn't apply to the scheme
  - Proof correct, but theorem "insufficient"
  - Example: attack hides in non-tightness
- Proof (and possibly theorem) too vague
- Theorem and proof correct, but not very useful
  - "A is secure if A is secure"

Results

- Almeida, Barbosa, Barthe, Grégoire, Laporte, Léchenet, Oliveira, Pacheco, Quaresma, Schwabe, Séré, Strub. Formally verifying Kyber Part I: Implementation Correctness.
  TCHES, 2023



Impact

- Proof is wrong
  - Theorem is correct
  - Theorem is also wrong
    - Scheme is still (possibly) secure
    - Scheme is efficiently broken
- Proof doesn't apply to the scheme
  - Proof correct, but theorem "insufficient"
  - Example: attack hides in non-tightness
- Proof (and possibly theorem) too vague
- Theorem and proof correct, but not very useful
  - "A is secure if A is secure"

Impact

- Proof is wrong
  - Theorem is correct
  - Theorem is also wrong
    - Scheme is still (possibly) secure
    - Scheme is efficiently broken
- Proof doesn't apply to the scheme
  - Proof correct, but theorem "insufficient"
  - Example: attack hides in non-tightness
- Proof (and possibly theorem) too vague
- Theorem and proof correct, but not very useful
  - "A is secure if A is secure"

Manual effort!

What I did not talk about

- Implementation security (Jasmin part)
  - Side-Channel Attack Resistance
  - Speculative Execution Attack Mitigation
  - Memory Safety
  - …
  - See CHES 2023 invited talk by Peter Schwabe https://youtu.be/7uIabAwB92M?si=gdGWEwXlz9XGZUhm&t=944
- Other tools
  - Barbosa, Barthe, Bhargavan, Blanchet, Cremers, Liao, Parno. SoK: Computer-Aided Cryptography. S&P '21 https://eprint.iacr.org/2019/1393

Why does NIST not require machine-checked proofs for the signature round?

Results are great but

- Full workflow for Kyber took more than 3 years of many, many people! (Still not fully published!)
- Tools are "Expert Tools"
- New proofs often need help of tool developer
- Little automation
- Little integration with higher level tools (e.g., for protocols)

Summary

- We have the tools, we can achieve great results
- Verifying proofs is still research
- Usability still needs improvement
- There are many different tools for different use-cases

- We are working on a fully verified PQC library!
- Join the Formosa project (https://formosa-crypto.org/)