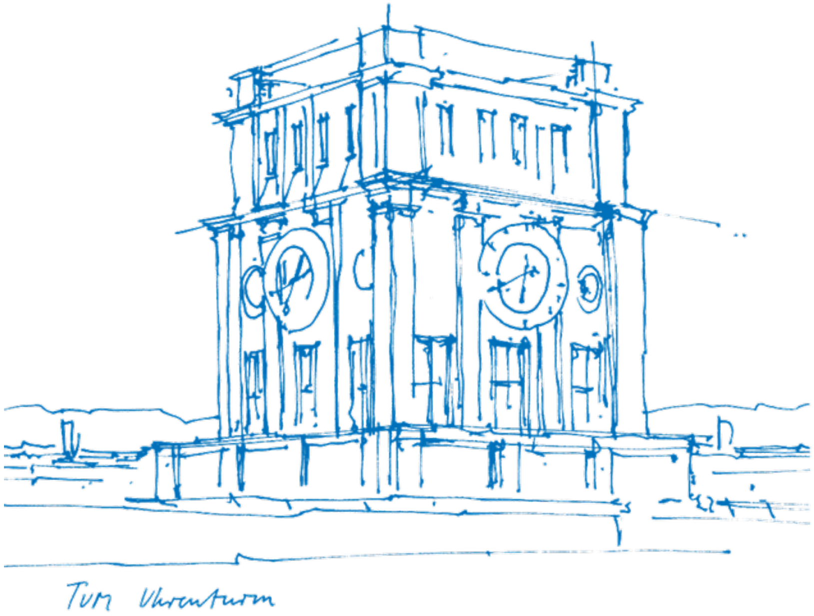
Verification of the $(1-\delta)$-Correctness Proof of Kyber

Katharina Kreuzer
Technische Universität München
Departement of Computer Science
Chair for Logic and Verification
Madrid, 24th October 2022



Isabelle

Isabelle is an interactive theorem prover used to formalize and verify mathematics and computer science.

A special ring. . .

$$R_q = \mathbb{Z}_q[x]/(x^n + 1)$$

q prime

n a power of 2

A special ring. . .

Using Isabelle theories for

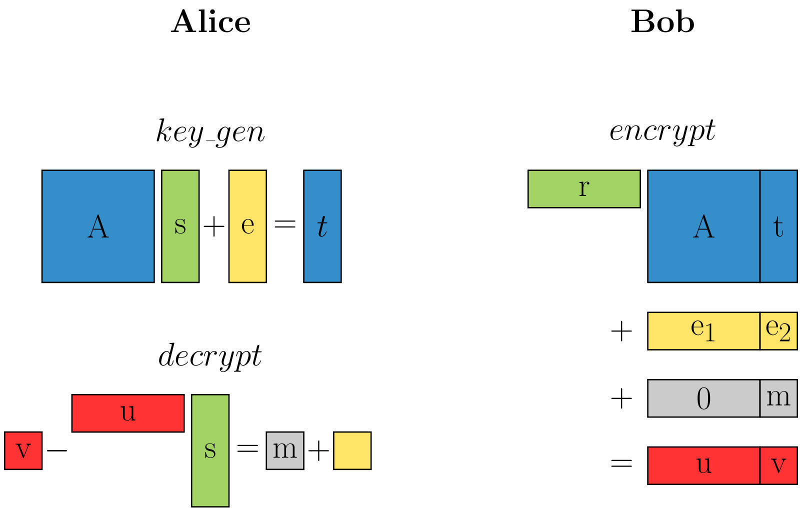
- finite fields
- polynomials
- equivalence relations
- quotient ring structure
- algebra
- analysis
- . . .

$$R_q = \mathbb{Z}_q[x]/(x^n + 1)$$

q prime

n a power of 2

Kyber - a post-quantum crypto system



public key private key errors

message (plaintext) encrypted message (ciphertext)

Kyber Algorithms in Isabelle

```
definition key_gen where
key_gen dt A s e = compress_vec dt (A * s + e)

definition encrypt where
encrypt t A r e1 e2 dt du dv m =
  (compress_vec du (A^T * r + e1),
   compress_poly dv ((decompress_vec dt t)^T * r + e2 +
      to_module (round(q/2)) * bitstring_to_module m))

definition decrypt where
decrypt u v s du dv = compress_poly 1
  ((decompress_poly dv v) - s^T * (decompress_vec du u))
```

Kyber Algorithms in Isabelle

```
definition key_gen where
key_gen dt A s e = compress_vec dt (A * s + e)

definition encrypt where
encrypt t A r e1 e2 dt du dv m =
  (compress_vec du (A^T * r + e1),
   compress_poly dv ((decompress_vec dt t)^T * r + e2 +
      to_module (round(q/2)) * bitstring_to_module m))

definition decrypt where
decrypt u v s du dv = compress_poly 1
  ((decompress_poly dv v) - s^T * (decompress_vec du u))
```

Kyber Algorithms in Isabelle

```
definition key_gen where
key_gen dt A s e = compress_vec dt (A * s + e)

definition encrypt where
encrypt t A r e1 e2 dt du dv m =
  (compress_vec du (A^T * r + e1),
   compress_poly dv ((decompress_vec dt t)^T * r + e2 +
      to_module (round(q/2)) * bitstring_to_module m))

definition decrypt where
decrypt u v s du dv = compress_poly 1
  ((decompress_poly dv v) - s^T * (decompress_vec du u))
```

Compression and Decompression induce:

- smaller key sizes
- compression errors
- a problem in the $(1-\delta)$-correctness proof
- a problem in the IND-CPA security proof

$(1-\delta)$-correctness

$(1-\delta)$-correctness

A cryptographic scheme is $(1-\delta)$-correct if and only if for all messages m it holds:

$$\mathbb{P}[m = decrypt(sk, encrypt(pk, m)) \mid (sk, pk) \leftarrow key\_gen] \geq 1 - \delta$$

$(1-\delta)$-correctness

$(1-\delta)$-correctness

A cryptographic scheme is $(1-\delta)$-correct if and only if for all messages m it holds:

$$\mathbb{P}[m = decrypt(sk, encrypt(pk, m)) \mid (sk, pk) \leftarrow key\_gen] \geq 1 - \delta$$

Correctness of Kyber

Define $\delta := \mathbb{P}[\|e^T r + e_2 - s^T e_1 + e'\|_\infty \geq \lceil q/4 \rfloor]$. Then Kyber is $(1-\delta)$-correct.

Problem in $(1-\delta)$-correctness proof

compression error: $\|x - decompress_d(compress_d(x))\|_\infty$

Problem in $(1-\delta)$-correctness proof

compression error: $\|x - decompress_d(compress_d(x))\|_\infty$

$$\|x\|_\infty = \max_i \ |x_i \bmod^{\pm} q|$$

Problem in $(1-\delta)$-correctness proof

compression error: $||x - decompress_d(compress_d(x))||_\infty$

$$||x||_\infty = \max_i \; |x_i \mod^{\pm} q|$$

$$\hat{x} = x \mod^{\pm} q \quad \text{iff} \quad \hat{x} \in (-\tfrac{q}{2}, \tfrac{q}{2}] \text{ and } \hat{x} \equiv x \mod q$$

Problem in $(1-\delta)$-correctness proof

compression error: $\|x - decompress_d(compress_d(x))\|_\infty$

$$\|x\|_\infty = \max_i \ |x_i \mod^{\pm} q|$$

$$\hat{x} = x \mod^{\pm} q \quad \text{iff} \quad \hat{x} \in (-\tfrac{q}{2}, \tfrac{q}{2}] \text{ and } \hat{x} \equiv x \mod q$$

does not guarantee homogeneity (only $\|s \cdot x\|_\infty \leq |s| \cdot \|x\|_\infty$)

Problem in $(1-\delta)$-correctness proof

compression error: $\|x - decompress_d(compress_d(x))\|_\infty$

$$\|x\|_\infty = \max_i \; |x_i \mod^\pm q|$$

$$\hat{x} = x \mod^\pm q \quad \text{iff} \quad \hat{x} \in (-\tfrac{q}{2}, \tfrac{q}{2}] \text{ and } \hat{x} \equiv x \mod q$$

does not guarantee homogeneity (only $\|s \cdot x\|_\infty \leq |s| \cdot \|x\|_\infty$)

$\Rightarrow \|\cdot\|_\infty$ is not a norm $\lightning$

How to solve this problem?

Add property

$$q \equiv 1 \mod 4$$

How to solve this problem?

Add property

$$q \equiv 1 \mod 4$$

$\Rightarrow$ last step of proof is valid even though $\|\cdot\|_\infty$ is not a norm

How to solve this problem?

Add property

$$q \equiv 1 \mod 4$$

$\Rightarrow$ last step of proof is valid even though $\|\cdot\|_\infty$ is not a norm

$\Rightarrow$ not noticed since chosen parameters fulfil this property

$$7681 \equiv 1 \mod 4 \quad \text{and} \quad 3329 \equiv 1 \mod 4$$

Number Theoretic Transform in Kyber

- Requires $2n$-th root of unity ψ where $n = 2^{n'} = 256$

Number Theoretic Transform in Kyber

- Requires $2n$-th root of unity ψ where $n = 2^{n'} = 256$
- Number Theoretic Transform for $f = \sum_{k=0}^{n-1} f_k x^k \in R_q$ defined as

$$NTT(f)_k = \sum_{j=0}^{n-1} f_j \; \psi^{j(2k+1)}$$

Number Theoretic Transform in Kyber

- Requires $2n$-th root of unity ψ where $n = 2^{n'} = 256$
- Number Theoretic Transform for $f = \sum_{k=0}^{n-1} f_k x^k \in R_q$ defined as

$$NTT(f)_k = \sum_{j=0}^{n-1} f_j \, \psi^{j(2k+1)}$$

- using $q \equiv 1 \mod n$, i.e., $7681 = 30 \cdot 256 + 1$ and $3329 = 13 \cdot 256 + 1$

Number Theoretic Transform in Kyber

- Requires $2n$-th root of unity ψ where $n = 2^{n'} = 256$
- Number Theoretic Transform for $f = \sum_{k=0}^{n-1} f_k x^k \in R_q$ defined as

$$NTT(f)_k = \sum_{j=0}^{n-1} f_j \, \psi^{j(2k+1)}$$

- using $q \equiv 1 \mod n$, i.e., $7681 = 30 \cdot 256 + 1$ and $3329 = 13 \cdot 256 + 1$
- implies additional property $q \equiv 1 \mod 4$

Number Theoretic Transform in Kyber

- Requires $2n$-th root of unity ψ where $n = 2^{n'} = 256$
- Number Theoretic Transform for $f = \Sigma_{k=0}^{n-1} f_k x^k \in R_q$ defined as

$$NTT(f)_k = \sum_{j=0}^{n-1} f_j \; \psi^{j(2k+1)}$$

- using $q \equiv 1 \mod n$, i.e., $7681 = 30 \cdot 256 + 1$ and $3329 = 13 \cdot 256 + 1$
- implies additional property $q \equiv 1 \mod 4$

✓ $(1 - \delta)$-correctness proof is valid

Conclusion

Formalization and verification in Isabelle:

- Kyber PKE algorithms for key generation, encryption and decryption
- $(1-\delta)$-correctness proof of Kyber
- based on "CRYSTALS-Kyber: a CCA-secure module-lattice-based KEM" by Bos et al.

Conclusion

Formalization and verification in Isabelle:

- Kyber PKE algorithms for key generation, encryption and decryption
- $(1-\delta)$-correctness proof of Kyber
- based on "CRYSTALS-Kyber: a CCA-secure module-lattice-based KEM" by Bos et al.

Problems during formalization:

- $\|\cdot\|_\infty$ not a norm
- Proof was corrected using additional assumption $q \equiv 1 \mod 4$
- $q \equiv 1 \mod 4$ given by NTT properties

Thank you for your attention!

Question: How does the IND-CPA proof for Kyber PKE with compression/decompression work?

Thank you for your attention!

Question: How does the IND-CPA proof for Kyber PKE with compression/decompression work?

Questions?