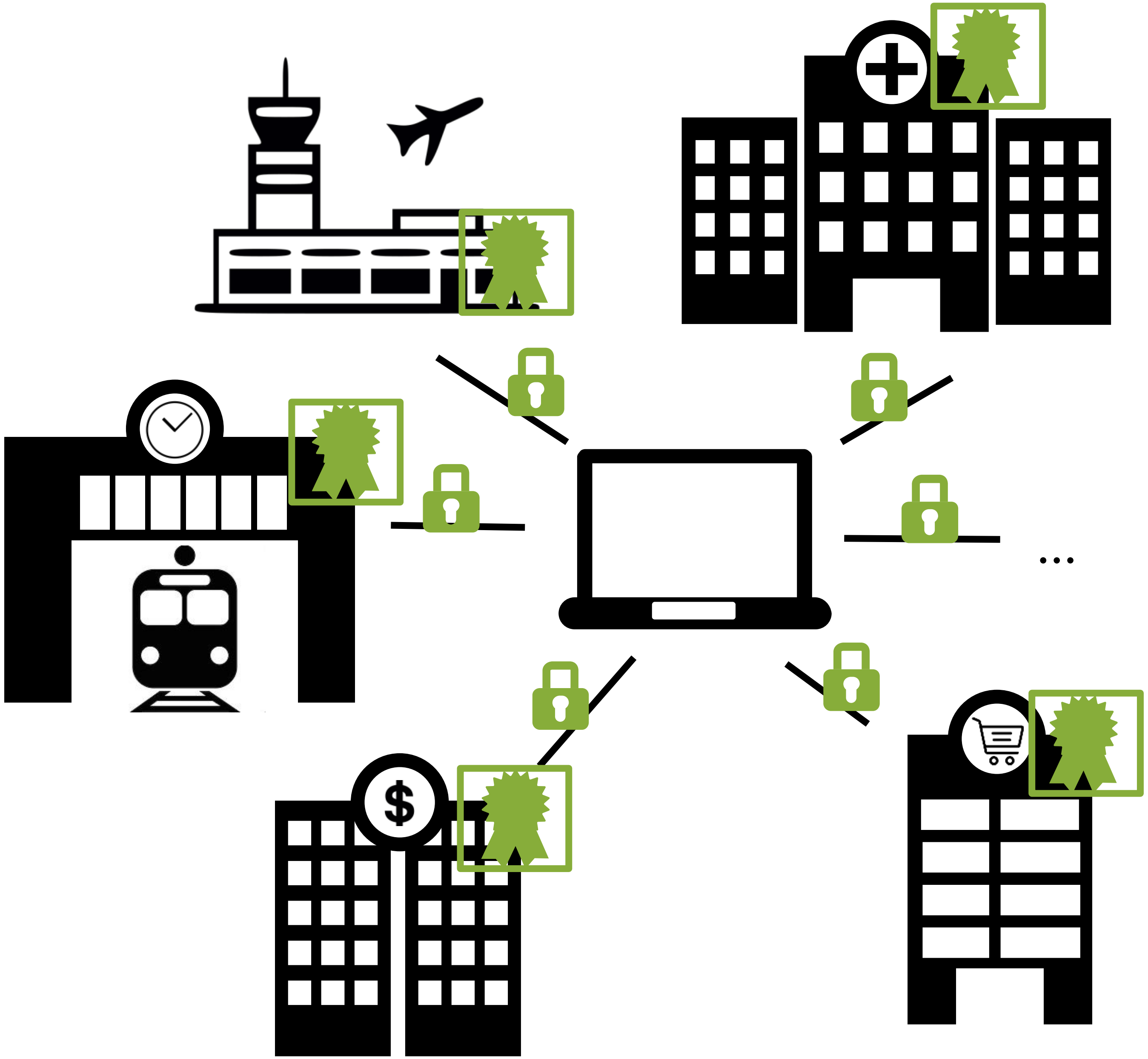
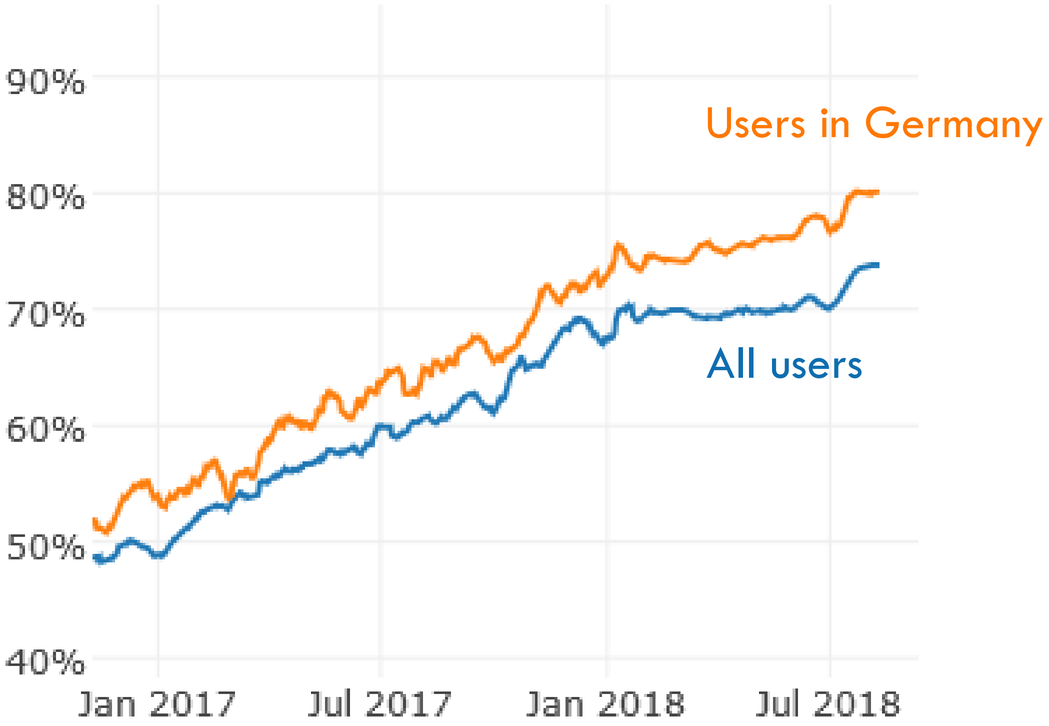
On the IND-CCA security of post-quantum public-key encryption schemes

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

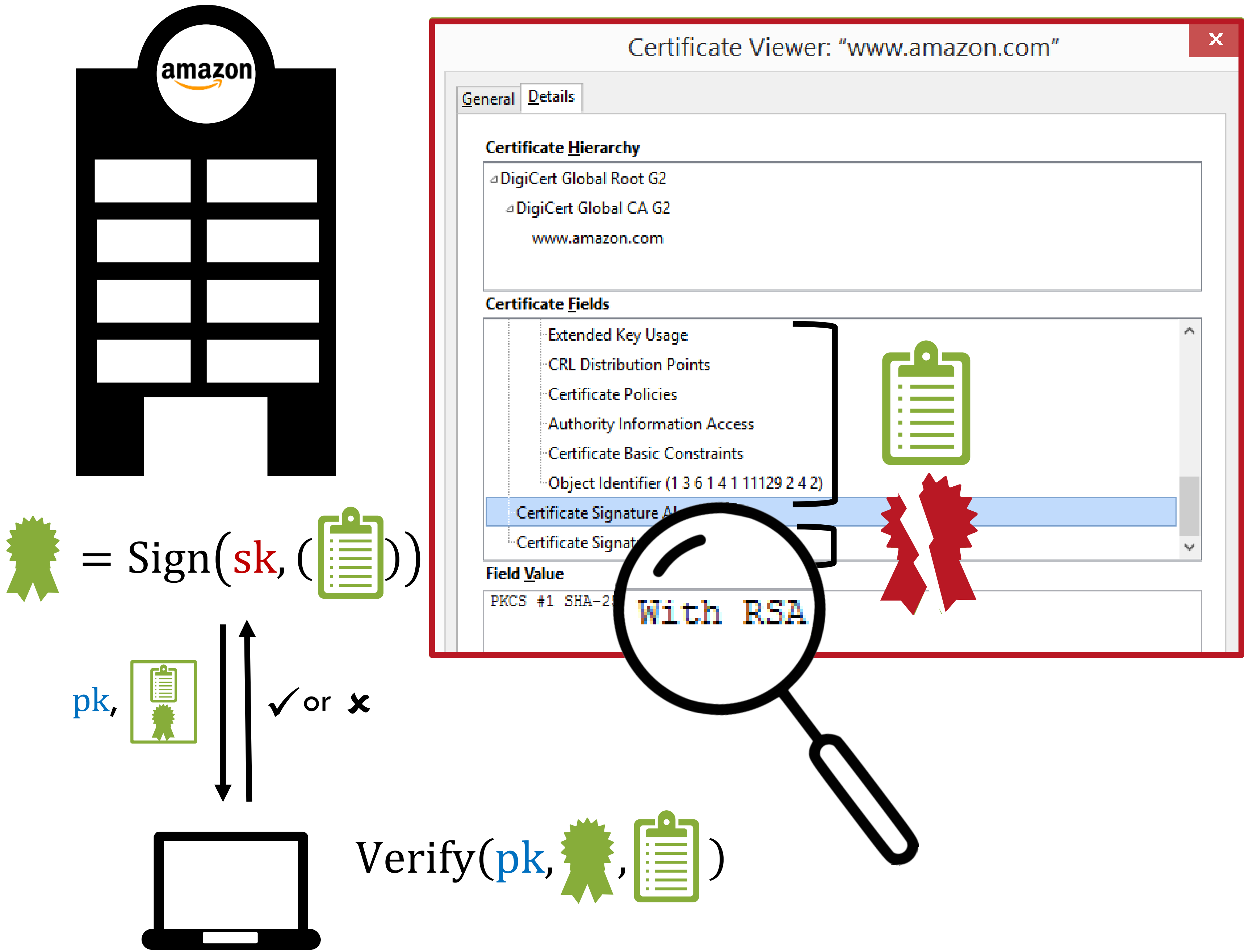
Nina Bindel

14/01/2020





Percentage of Firefox pageloads using TLS



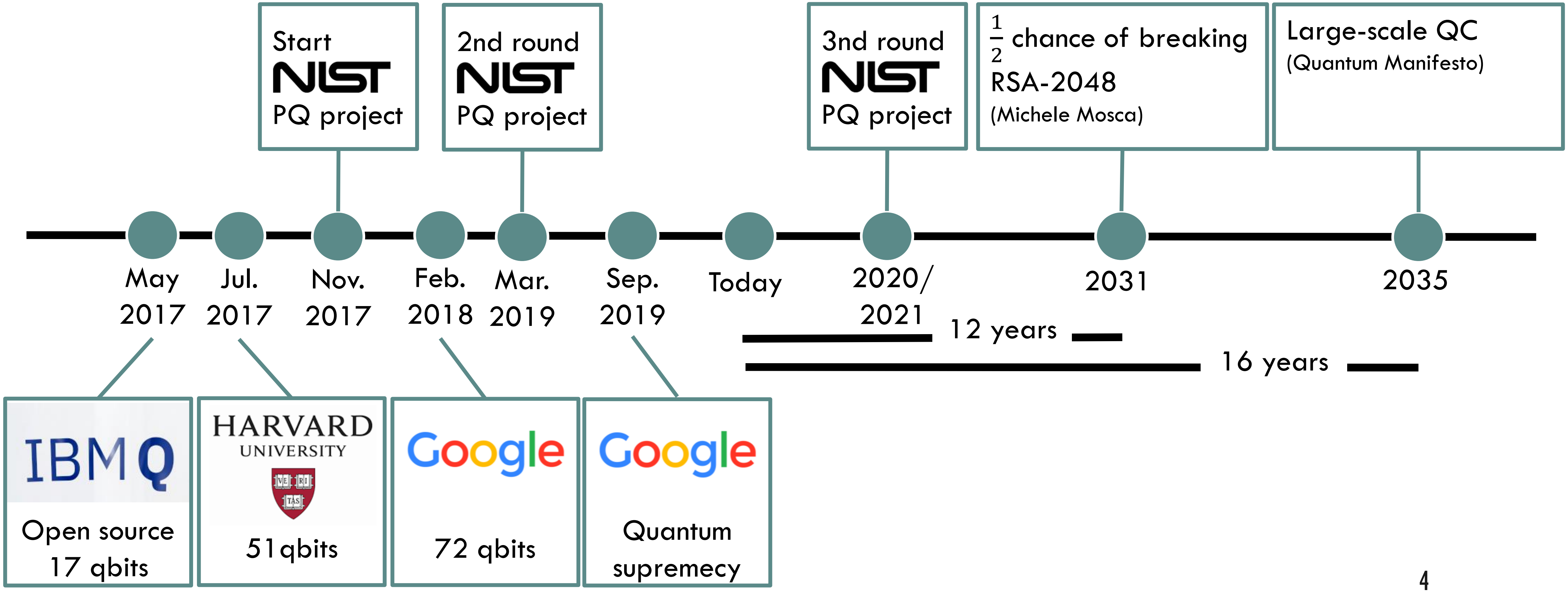
Shor's quantum algorithm [Shor97]:

⟹ Recover sk

⟹ Generate RSA- for any

Decrypt any RSA-

Quantum computing: State-of-the-art and estimations



Outline

- NIST standardization effort
- IND-CCA Security of PKEs/KEMs
- Decryption failures of PKEs/KEMs

Challenge

Find quantum hard problems

Construct schemes over these problems

With courtesy of Denis Butin and Johannes Buchmann

Quantum-hard problems -- NIST

Lattice-based

Learning **W**ith **E**rrors
Ring-LWE
Module LWE

Learning **W**ith **R**ounding
Module LWR

Sort Integer Solution
SelfTargetMSIS

NTRU problem
NTRU-SIS

Hash-based

PQ-DM-SPR
PQ-ITSR

Isogeny-based

Supersingular Isogeny DH

Multivariate

MQ
MinRank
IP

Code-based

Quasi-**c**yclic codeword finding

QC syndrome decoding

QC syndrome decoding with parity

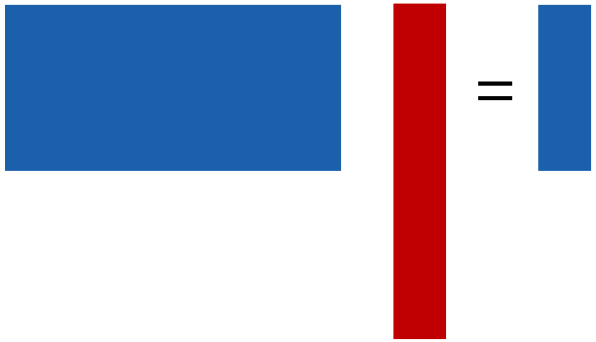
QC low-density-parity-check syndrome decoding

Ideal rank syndrome decoding

Ideal low-rank parity check distinguishing

Goppa code distinguishing

Short Integer Solution Problem



$$A \cdot s = b \bmod q$$

“short“ s

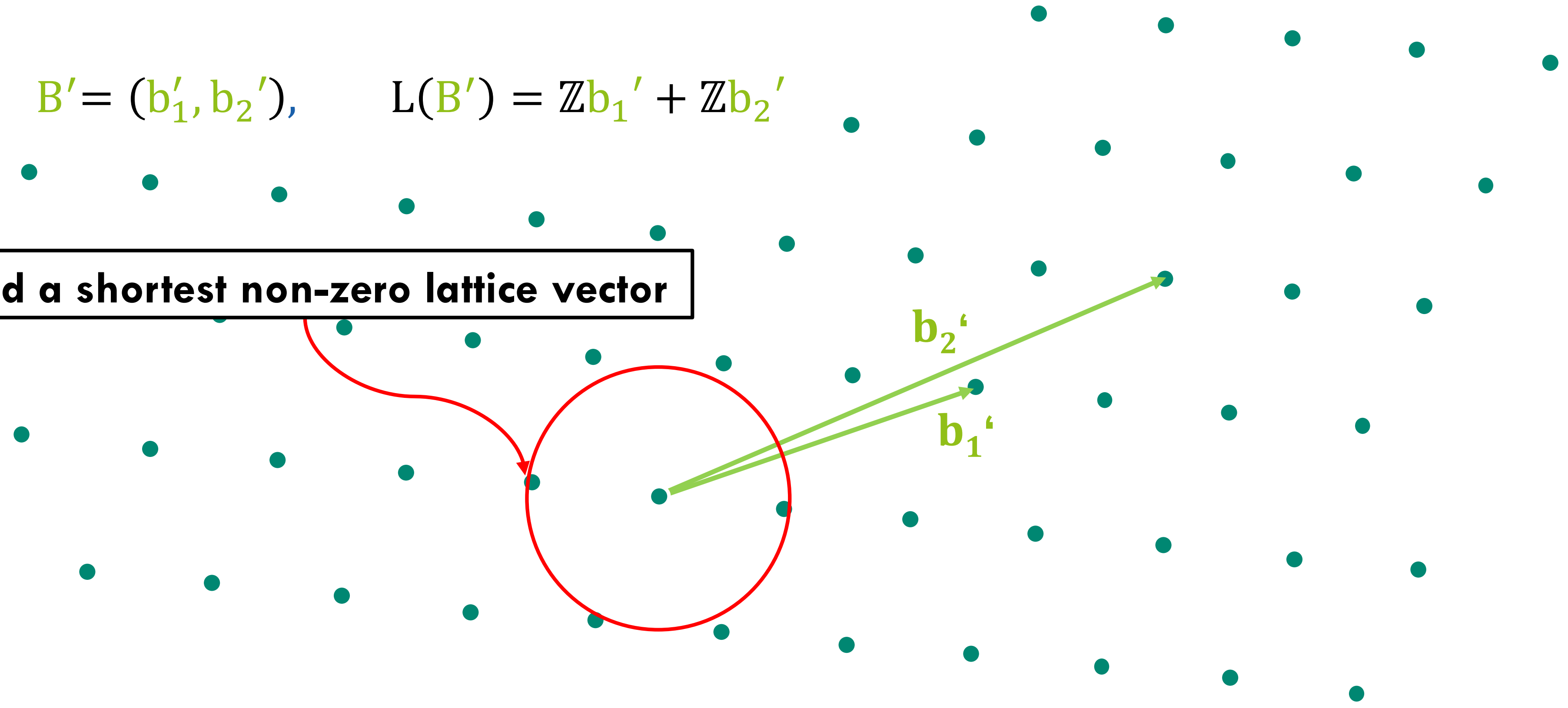
A defines lattice:

$$\Lambda_q(A) = \{\, z \in \mathbb{Z}^n : z = A^T s \bmod q,\ s \in \mathbb{Z}_q^m \,\}$$

To solve SIS, solve SVP

Shortest Vector Problem (SVP)

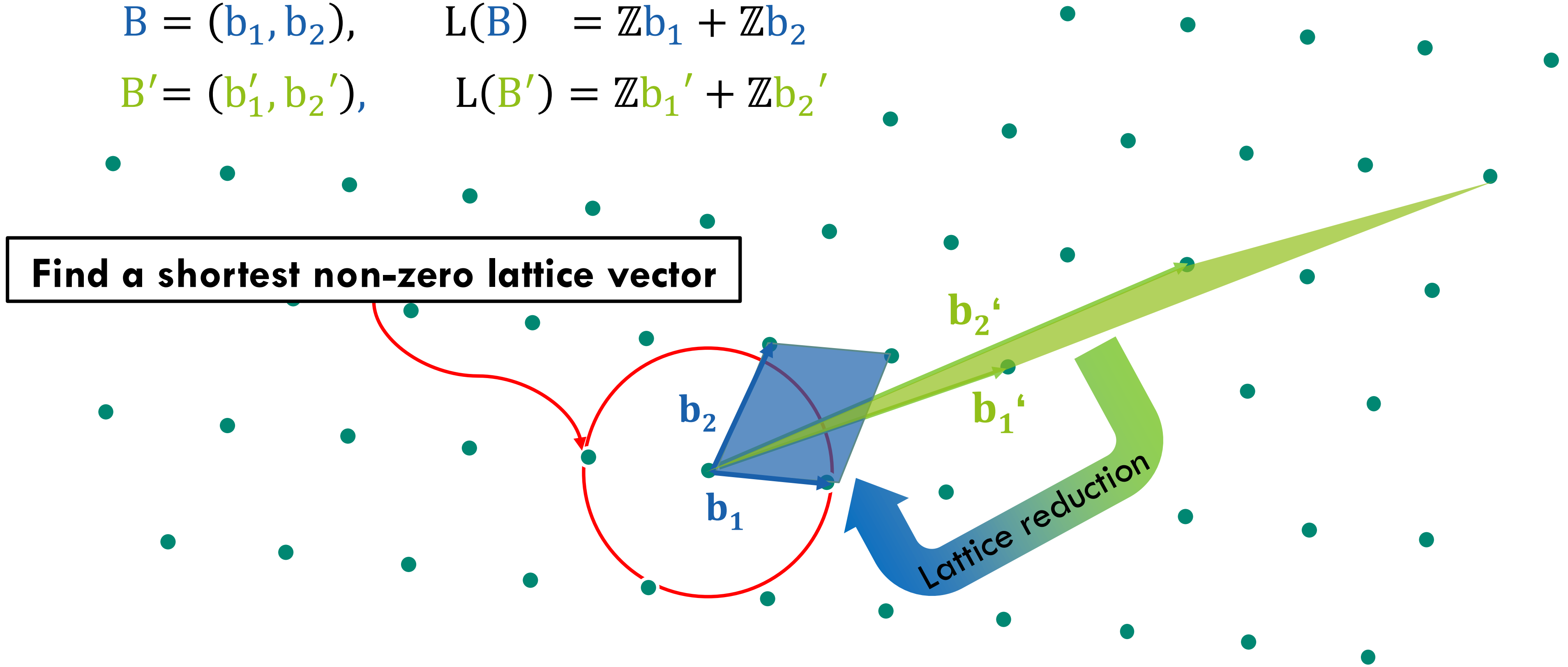
$B' = (b_1', b_2'), \qquad L(B') = \mathbb{Z}b_1' + \mathbb{Z}b_2'$



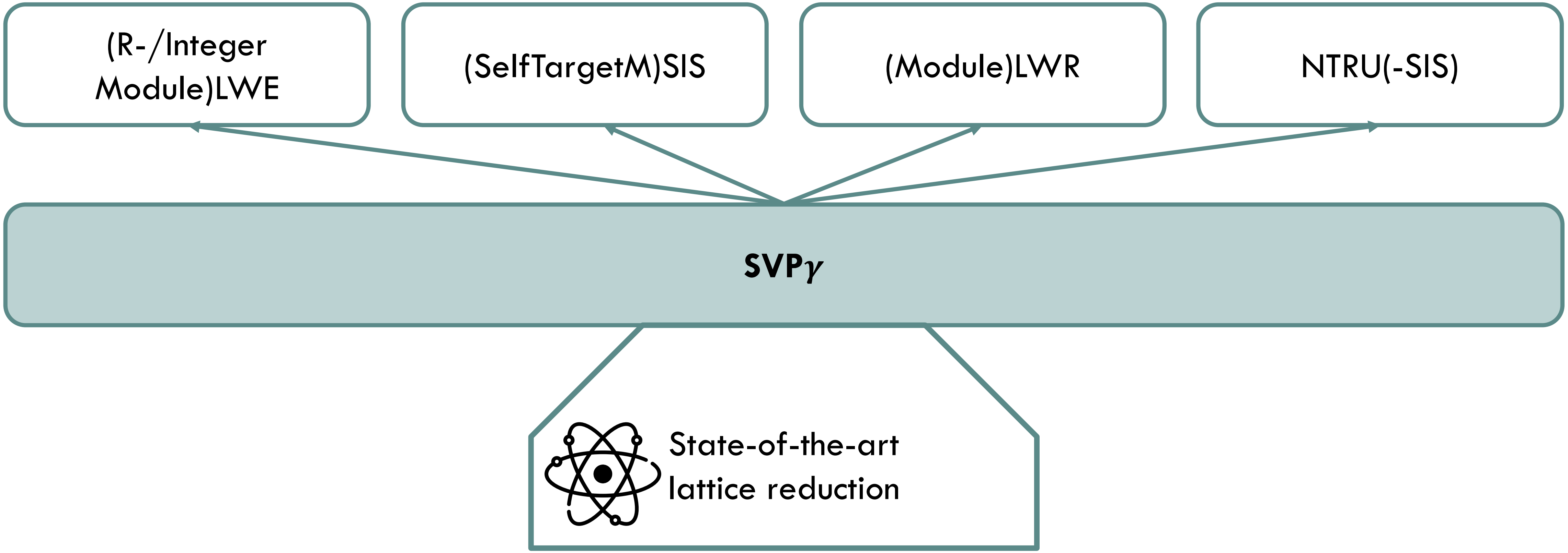
Solving the SVP

$B = (b_1, b_2), \qquad L(B) = \mathbb{Z}b_1 + \mathbb{Z}b_2$

$B' = (b_1', b_2'), \qquad L(B') = \mathbb{Z}b_1' + \mathbb{Z}b_2'$

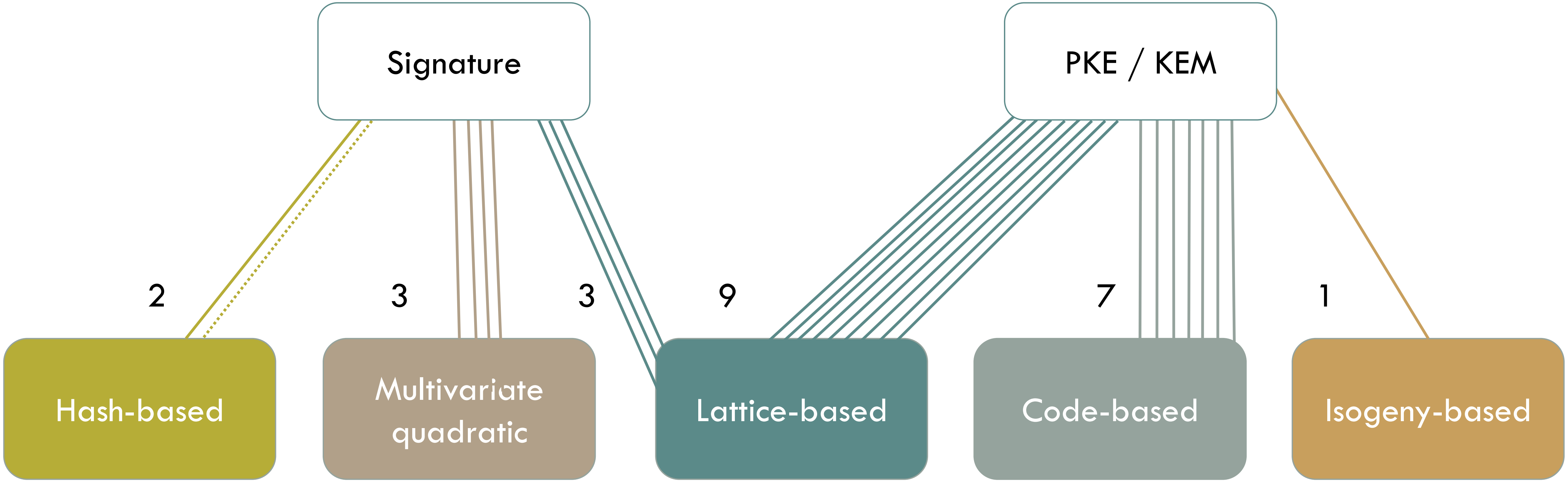


Lattice-based problems



NIST candidates

With courtesy of Denis Butin and Johannes Buchmann



Almost all use the Fujisaki-Okamoto transform

Outline

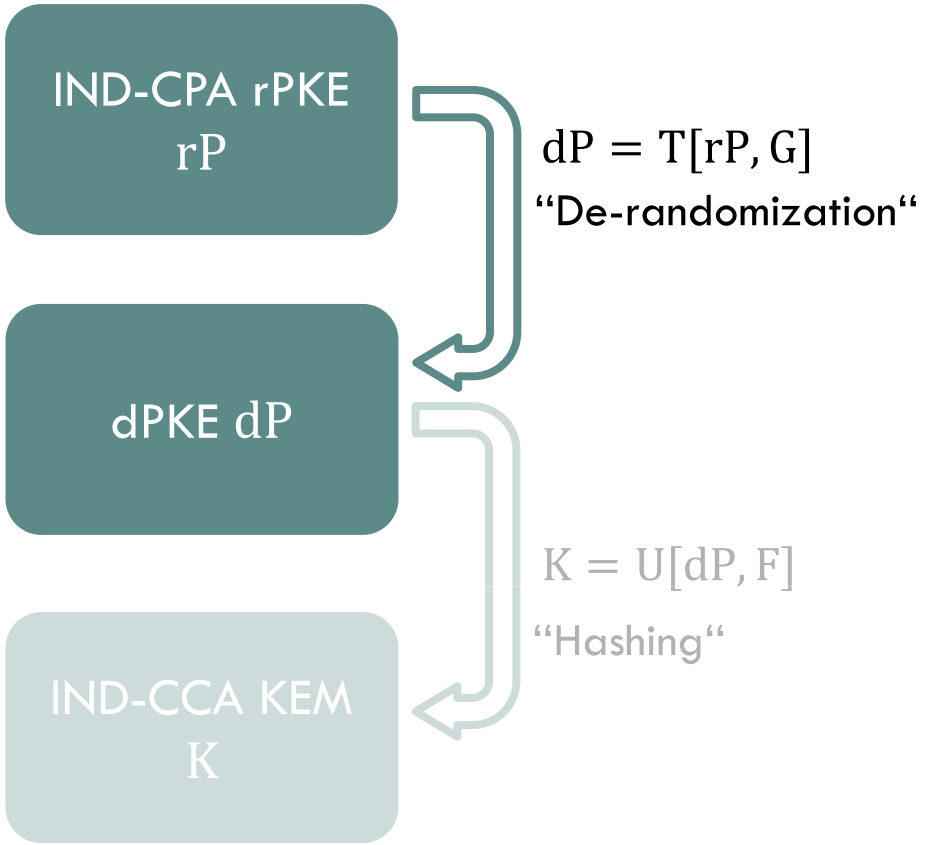
NIST standardization effort

IND-CCA Security of PKEs/KEMs

- Fujisaki-Okamoto Transform
- QROM
- IND-CCA reduction

Decryption failures of PKEs/KEMs

Fujisaki-Okamoto Transform [FO99,HHK17]



$\mathbf{Gen_{dP}}(\quad) = \quad \mathrm{Gen_{rP}}(\quad)$

$\mathbf{Encr_{dP}(pk, m)} = \quad \mathrm{Encr_{rP}(pk, m; G(m))}$

$\mathbf{Decr_{dP}(sk, c)} = \quad \mathrm{Decr_{rP}(sk, c)}$

Fujisaki-Okamoto transform [FO99,HHK17]

IND-CPA rPKE
rP

$\mathrm{dP} = \mathrm{T}[\mathrm{rP}, \mathrm{G}]$
“De-randomization“

dPKE dP

$\mathrm{K} = \mathrm{U}[\mathrm{dP}, \mathrm{F}]$
“Hashing“

IND-CCA KEM
K

Encaps(pk):
$\mathrm{m} \leftarrow_{\$} \mathrm{M}$
Encryption: $\mathrm{c} \leftarrow \mathrm{Encr}_{\mathrm{dP}}(\mathrm{pk}, \mathrm{m})$
Hashing: $\mathrm{k} \leftarrow \mathrm{H}(\mathrm{m}, \mathrm{c})$
return (k, c)

Decaps(sk, prfk, c):
Decryption: $\mathrm{m}' \leftarrow \mathrm{Decr}_{\mathrm{dP}}(\mathrm{sk}, \mathrm{c})$
if $\mathrm{m}' = \perp$: return $\mathrm{F}(\mathrm{prfk}, \mathrm{c})$
Re-encryption: if $\mathrm{Encr}_{\mathrm{dP}}(\mathrm{pk}, \mathrm{m}') \neq \mathrm{c}$: return $\mathrm{F}(\mathrm{prfk}, \mathrm{c})$
return $\mathrm{H}(\mathrm{m}', \mathrm{c})$

$\mathrm{k} \leftarrow \mathrm{H}(\mathrm{m}, \mathrm{c})$	$\mathrm{k} \leftarrow \mathrm{H}(\mathrm{m})$	Rejection
$\mathrm{U}^{\perp}$	$\mathrm{U}_{\mathrm{m}}^{\perp}$	$\perp$ - ‘’explicit’’
$\mathrm{U}^{\$}$	$\mathrm{U}_{\mathrm{m}}^{\$}$	$\$$ - ‘’implicit’’

Related work

$$\text{IND-CPA rPKE } \mathrm{rP} \xrightarrow{\mathrm{dP} = \mathrm{T}[\mathrm{rP}, \mathrm{G}]} \text{dPKE } \mathrm{dP} \xrightarrow{\mathrm{K} = \mathrm{U}[\mathrm{dP}, \mathrm{F}]} \text{IND-CCA KEM } \mathrm{K}$$

[HHK17]	$q_G\sqrt{\epsilon_{rP}} \geq \epsilon_{dP}$	$(q_{H'}+q_H)\sqrt{\epsilon_{dP}} \geq \epsilon_K$	$\boldsymbol{\epsilon_{rP} \geq \epsilon_K^4/q_{RO}^6}$
[SXY18, JZCWM18]			$\boldsymbol{\epsilon_{rP} \geq \epsilon_K^2/q_{RO}^2}$
[JZM19,HKSU18,…]			$\boldsymbol{\epsilon_{rP} \geq \epsilon_K^2/q_{RO}}$
[**B**HHHP19]	$d\epsilon_{rP} \geq \epsilon_{dP}$	$\sqrt{\epsilon_{dP}} \geq \epsilon_K$	$\boldsymbol{\epsilon_{rP} \geq \epsilon_K^2/q_{RO}}$

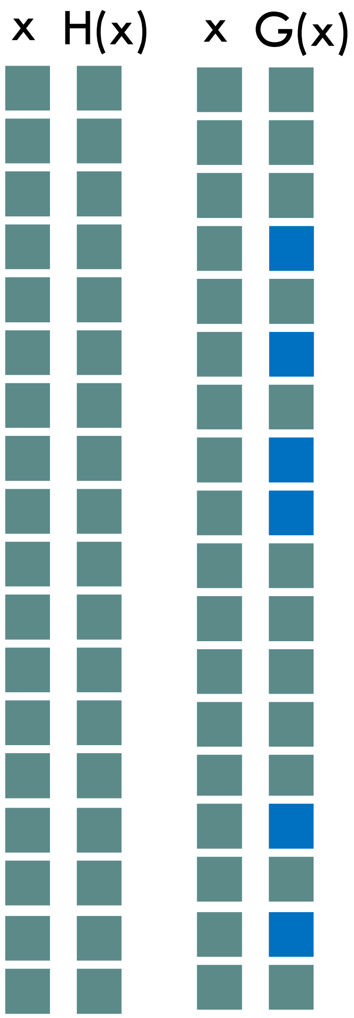
Random oracle vs. quantum random oracle

- Classical queries
- Queries and responses can be easily recorded
- Random oracle can be reprogrammed

- Queries in superposition
- Queries and responses are much harder to record [Zha19]
- Much harder to respond adaptevely/reprogramm oracle

Possible but leads to less tight bounds

Unruh‘s one-way to hiding (O2H) lemma



$\mathrm{S} = \mathrm{G}^{-1}(\blacksquare)$, A^{H} quantum oracle algorithm, q queries of depth $\mathrm{d} \leq \mathrm{q}$

If $|\Pr[\mathrm{Ev}: \mathrm{A}^{\mathrm{H}}(\mathrm{z})] - \Pr[\mathrm{Ev}: \mathrm{A}^{\mathrm{G}}(\mathrm{z})]| = \delta > 0$, A asked some $\mathrm{x} \in \mathrm{S}$

Behavior can be observed by B

$\mathrm{B} \rightarrow \mathrm{x}$ with probability ϵ

O2H variant	#S	Sim. must know	Bound
Original [Unr15]	Arbitrary	H or G	$\delta \leq 2d\sqrt{\epsilon}$
Semi-classical [AHU19]	Arbitrary	(G or H) and S	$\delta \leq 2\sqrt{d\epsilon}$
Double-sided [this work]	1	H and G	$\delta \leq 2\sqrt{\epsilon}$

OW-CPA dPKE to IND-CCA KEM

IND-CCA experiment:

$H \leftarrow \mathcal{H}$

$(\mathrm{sk}, \mathrm{pk}) \leftarrow \mathrm{KeyGen}()$

$m^* \leftarrow_\$ M$

$c^* \leftarrow \mathrm{Encrypt}(\mathrm{pk}, m^*)$

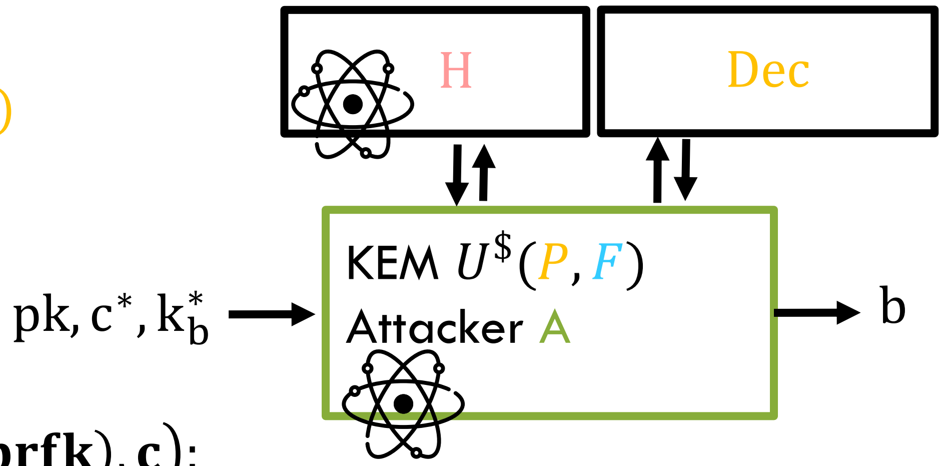
$k_0^* \leftarrow H(m^*, c^*)$

$k_1^* \leftarrow_\$ K$

$b \leftarrow_\$ \{0,1\}$

Given:

H Hash | F PRF | $P = T[P', G]$ dPKE



Construct:

$$\mathrm{Adv}_{U^\$(P,F)}^{\mathrm{IND-CCA}}(A) \leq 2\sqrt{\mathrm{Adv}_P^{\mathrm{OW-CPA}}(B_1)} + 2\mathrm{Adv}_F^{\mathrm{PRF}}(B_3) + f(\delta)$$

Oracle $\mathbf{Dec}((\mathbf{sk}, \mathbf{pk}, \mathbf{prfk}), \mathbf{c})$:

if $c = c^*$: return $\perp$

$m' \leftarrow \mathrm{Decrypt}(\mathrm{sk}, c)$

if $\mathrm{Encrypt}(\mathrm{pk}, m') = c$: return $k' \leftarrow H(m, c)$

return $k' \leftarrow F(\mathrm{prfk}, c)$

Outline

NIST standardization effort

IND-CCA Security of PKEs/KEMs

Decryption failures of PKEs/KEMs
- Attack
- Impact on NIST submissions

Example: Frodo

$\mathbb{Z}_{16} = \{-7, \dots, 0, \dots, 8\}$

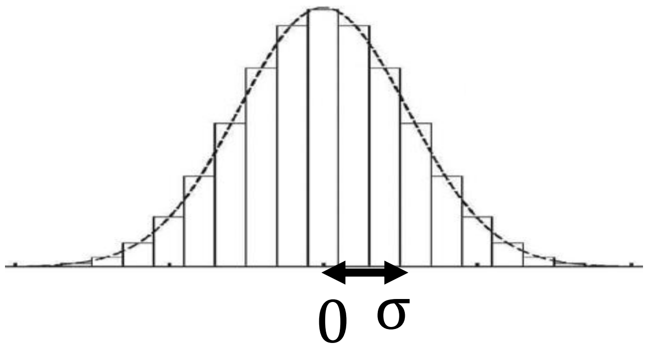
Key generation:

$A = 5$

sk $\quad S = 1, E = 2$

pk $\quad B = AS + E \bmod 16$

$= 7$



Encryption: $\mathrm{m} = 1$

$S' = -2, E' = 1, E'' = 2$

$C_1 = S'A + E' \bmod 16 \quad = 7$

$C_2 = (S'B + E'' \bmod 16) + \mathrm{Encode}(\mathrm{m})$

$= \underbrace{4 \qquad\qquad\qquad}_{\leq \frac{q}{8} = 2} + 1 \cdot \frac{q}{4}$

$= 8$

Correctness requirement

Decryption:

$M = C_2 - C_1 S \bmod 16 \quad = 1$

$\mathrm{m}' = \mathrm{Decode}(M)$

$= \left\lfloor 1 \cdot \frac{4}{q} \right\rceil \ mod\ 4 = 0$

$\neq \mathrm{m} = 1$

Correctness definition

Correctness experiment $\mathrm{COR}_{\mathrm{P}}^{\mathrm{A}}$**:**

$(\mathrm{sk}, \mathrm{pk}) \leftarrow \mathrm{KeyGen}()$
$\mathrm{m}^* \leftarrow \mathrm{A}(\mathrm{sk}, \mathrm{pk})$
$\mathrm{c}^* \leftarrow \mathrm{Encrypt}(\mathrm{pk}, \mathrm{m}^*)$
$\mathrm{return}\ [\mathrm{Decrypt}(\mathrm{sk}, \mathrm{c}) = \mathrm{m}]$

Theorem [HHK17]:

If rP is δ-correct
then $\mathrm{T}[\mathrm{rP}, \mathrm{G}]$ is $\delta_1(\mathrm{q_G}) = (\delta \cdot \mathrm{q_G})$-correct.

P is **δ-correct** if $\Pr\left[\mathrm{COR}_{\mathrm{P}}^{\mathrm{A}}\right] \leq \delta$

$\Leftrightarrow \Pr[\mathrm{Decrypt}(\mathrm{c}, \mathrm{sk}) \neq m : c \leftarrow Encrypt(m, pk), (pk, sk) \leftarrow Gen()] \leq \delta$ **if no depency on $\mathbf{m}$**

“One-shot correctness“

Reality check: Frodo

2.2.7 Correctness of IND-CPA PKE

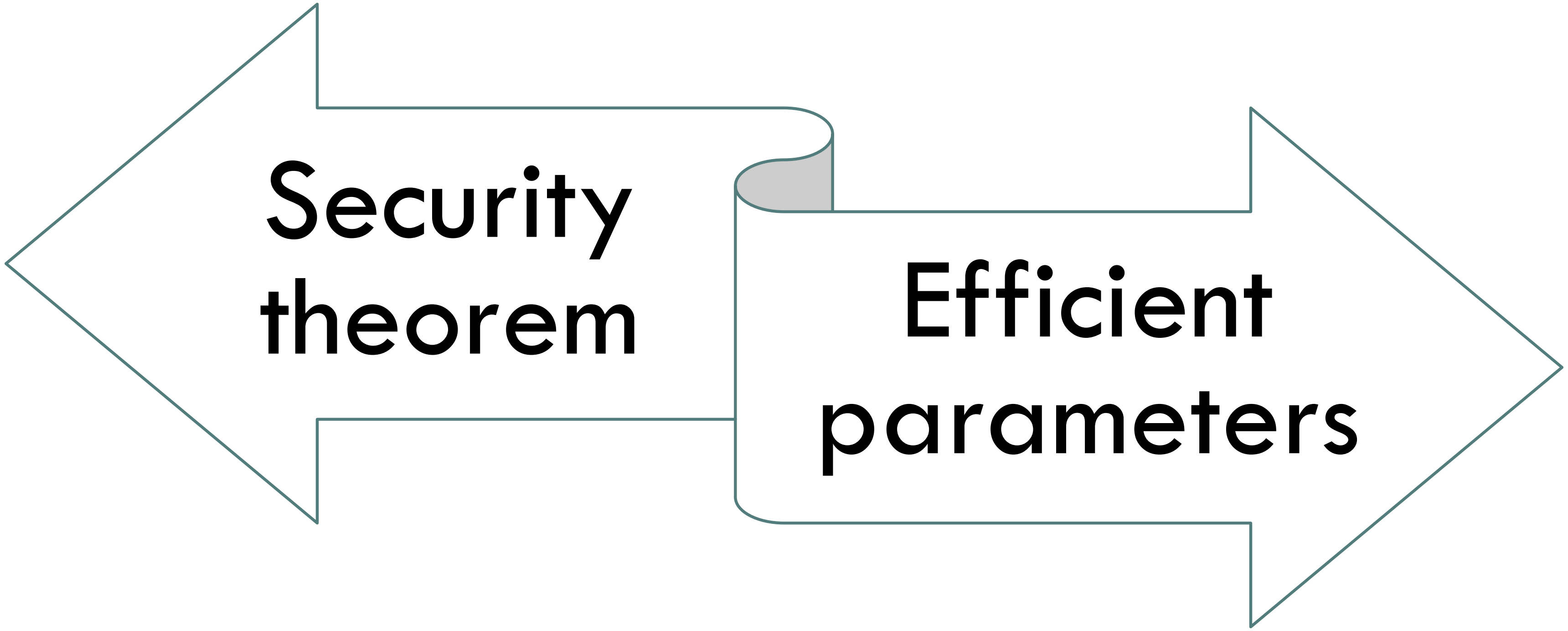
The next lemma states bounds on the size of errors that can be handled by the decoding algorithm.

Lemma 2.18. *Let $q = 2^D$, $B \leq D$. Then* $\mathrm{dc}(\mathrm{ec}(k) + e) = k$ *for any $k, e \in \mathbb{Z}$ such that $0 \leq k < 2^B$ and $-q/2^{B+1} \leq e < q/2^{B+1}$.*

Proof. This follows directly from the fact that $\mathrm{dc}(\mathrm{ec}(k) + e) = \lfloor k + e2^B/q \rceil \bmod 2^B$. □

2.2.10 Correctness of IND-CCA KEM

The failure probability δ of FrodoKEM is the same as the failure probability of the underlying FrodoPKE as computed in Section 2.2.7.


Security theorem
Efficient parameters

Alternative: State-of-the-art failure attacks

Recall:

$$C_1 = S'A + E' \bmod 16$$

$$C_2 = V + \mathrm{Encode}(m)$$

Failure boosting attack [DVV19,GJN19] = Method to find C_2 with large V

Security:
Assume adative adversary

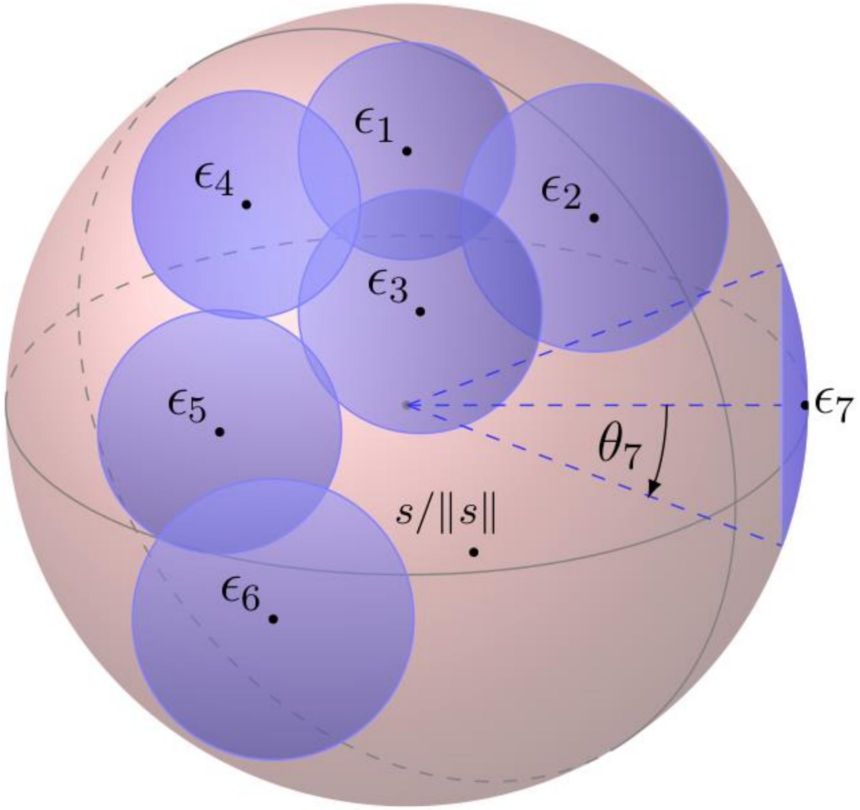
Correctness:
Assume **non**-adative adversary

If A breaks correctness, A can break security.

Is it possible to gain secret information from adaptively queried successfull decryptions?

Yes!

Bindel and Schanck, IACR eprint 2019/1392



Recall:

$\mathrm{sk} = \mathrm{S}, \mathrm{E}$

$\mathrm{C}_1 = \mathrm{S}'\mathrm{A} + \mathrm{E}' \bmod 16$

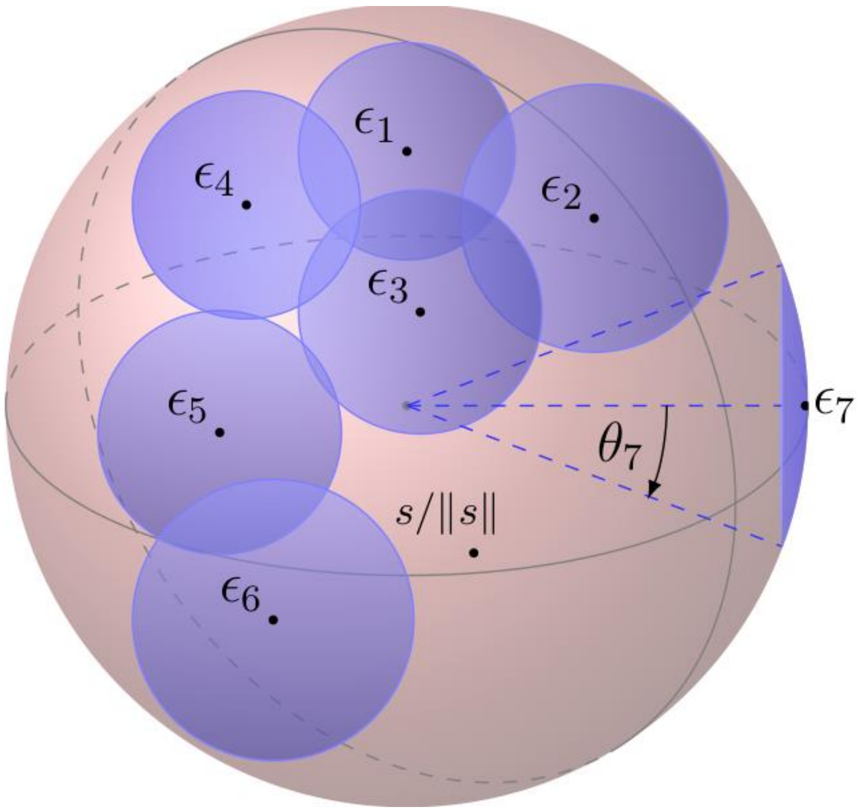
$\mathrm{C}_2 = V + \mathrm{Encode}(\mathrm{m})$

$\epsilon_i = \epsilon_i(S', E')$ randomness

Adversary learns from succesfull decryptions:

- S is not in blue region
- To trigger decryption error with higher probability, choose ϵ_8 in red region

Efficacy of a query set



$\mathrm{E} = \{\epsilon_1, \ldots, \epsilon_7, \ldots\}$

Efficacy of E = fraction of the sphere covered by caps

$$= \frac{\text{blue area}}{\text{red area}}$$

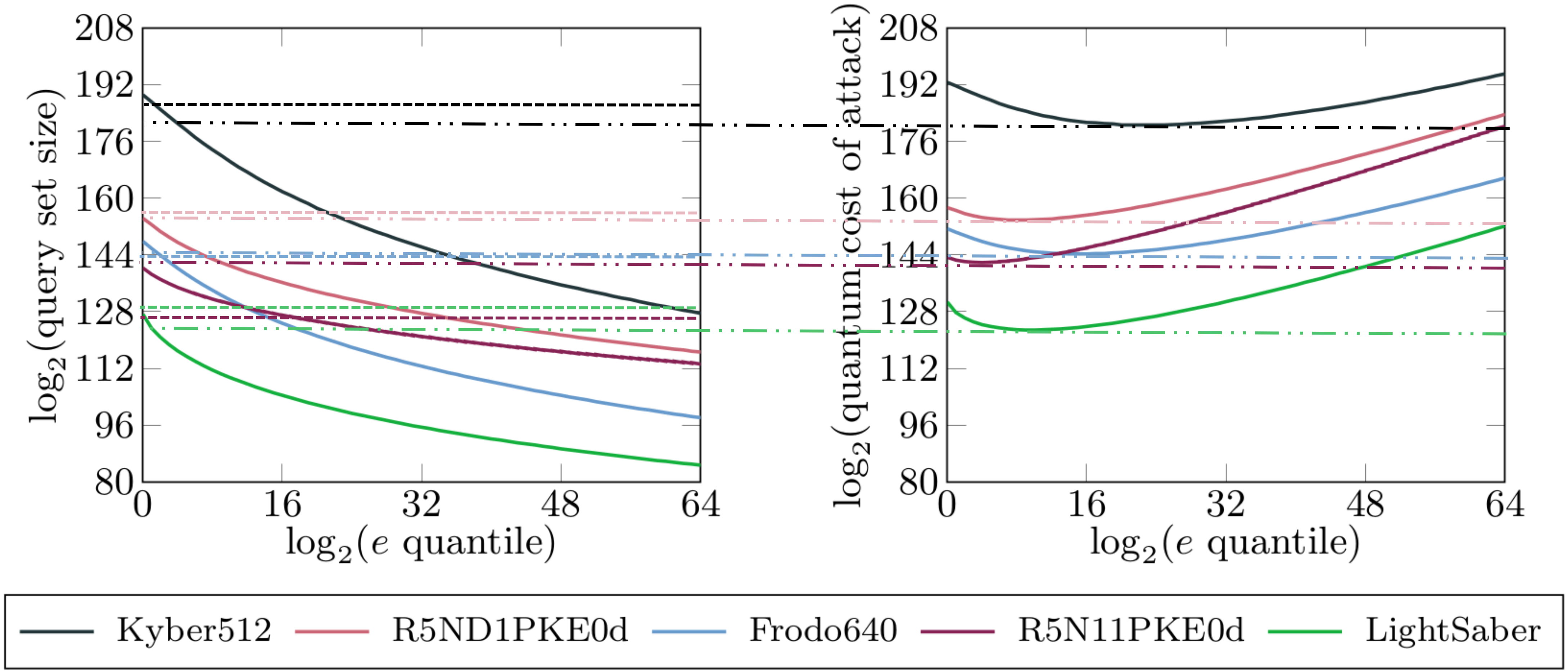
Intelligent adversary:

Efficacy ⬆ and $\#\mathrm{E}$ ⬇

Cost of adversary:

- Cost of generating efficient query set $\leq 2^{64}$
- Cost of asking queries: $\leq 2^{64}$ (NIST CfS)

Impact on NIST submissions



Conclusion

- Don‘t ask for revision of parameters
- Show one-shot correctness not reliable
- Open question about “right“ correctness def

  (more discussion in IACR eprint 2019/1392)

UNIVERSITY OF WATERLOO | IQC Institute for Quantum Computing

THANKS