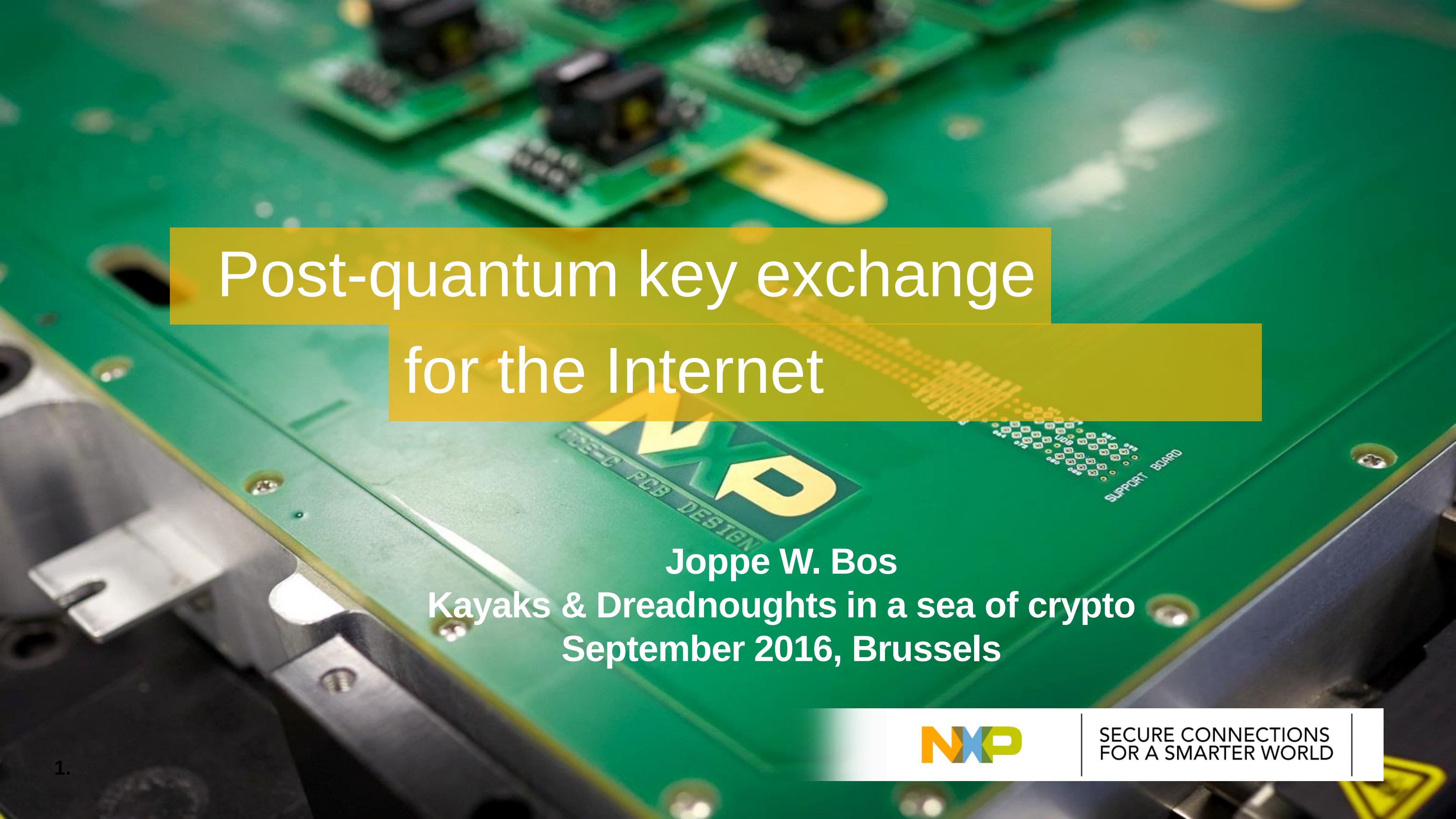


Post-quantum key exchange for the Internet

Joppe W. Bos

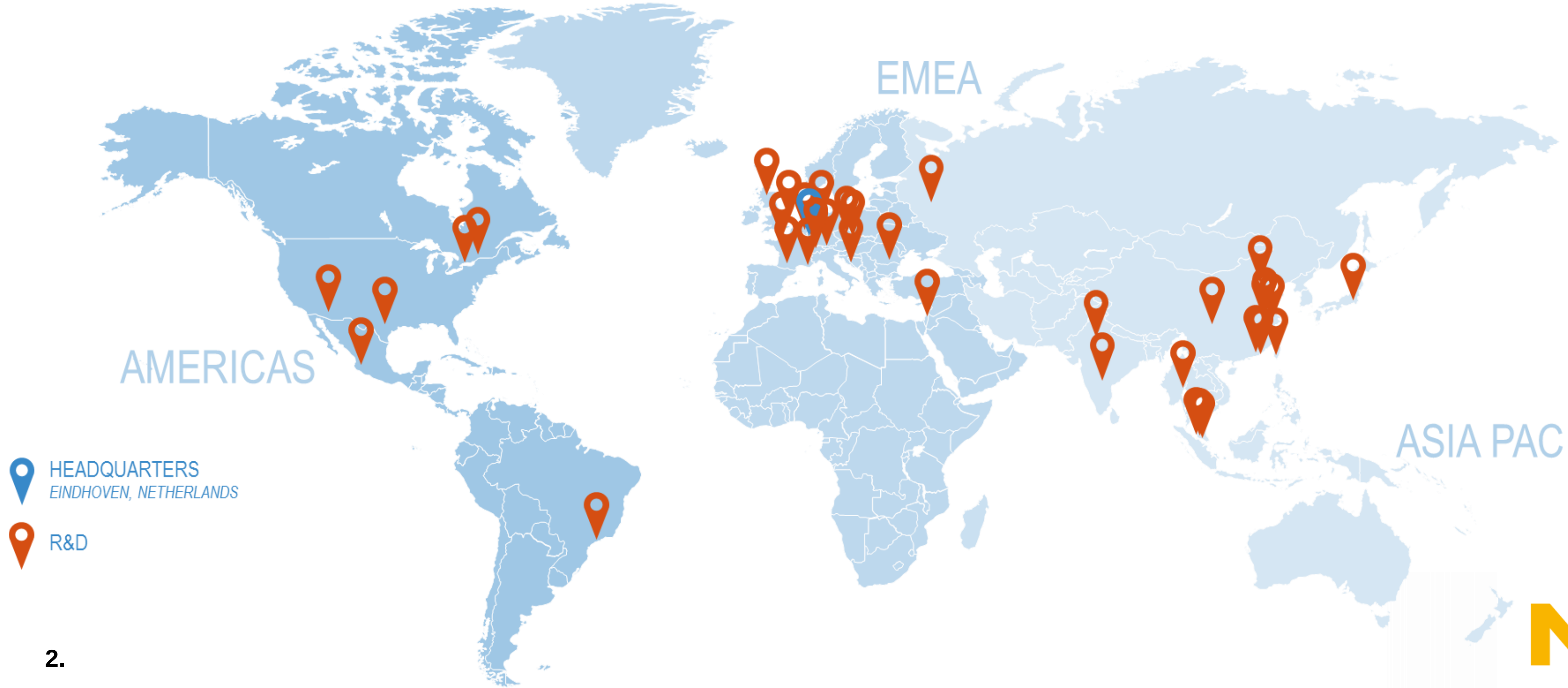
Kayaks & Dreadnoughts in a sea of crypto

September 2016, Brussels

NXP Semiconductors

Operations in > 35 countries, more than 130 facilities
≈ 45,000 employees

Research & Development
≈ 11,200 engineers in 23 countries



Acknowledgements

Collaborators

- Douglas Stebila
- Craig Costello and Michael Naehrig
- Léo Ducas
- Ilya Mironov and Ananth Raghunathan
- Valeria Nikolaenko

Support

Supported in part by the Commission of the European Communities through the Horizon 2020 program under project number 645622 (PQCRYPTO).

Slides reused from the talk Douglas gave at SAC 2016.

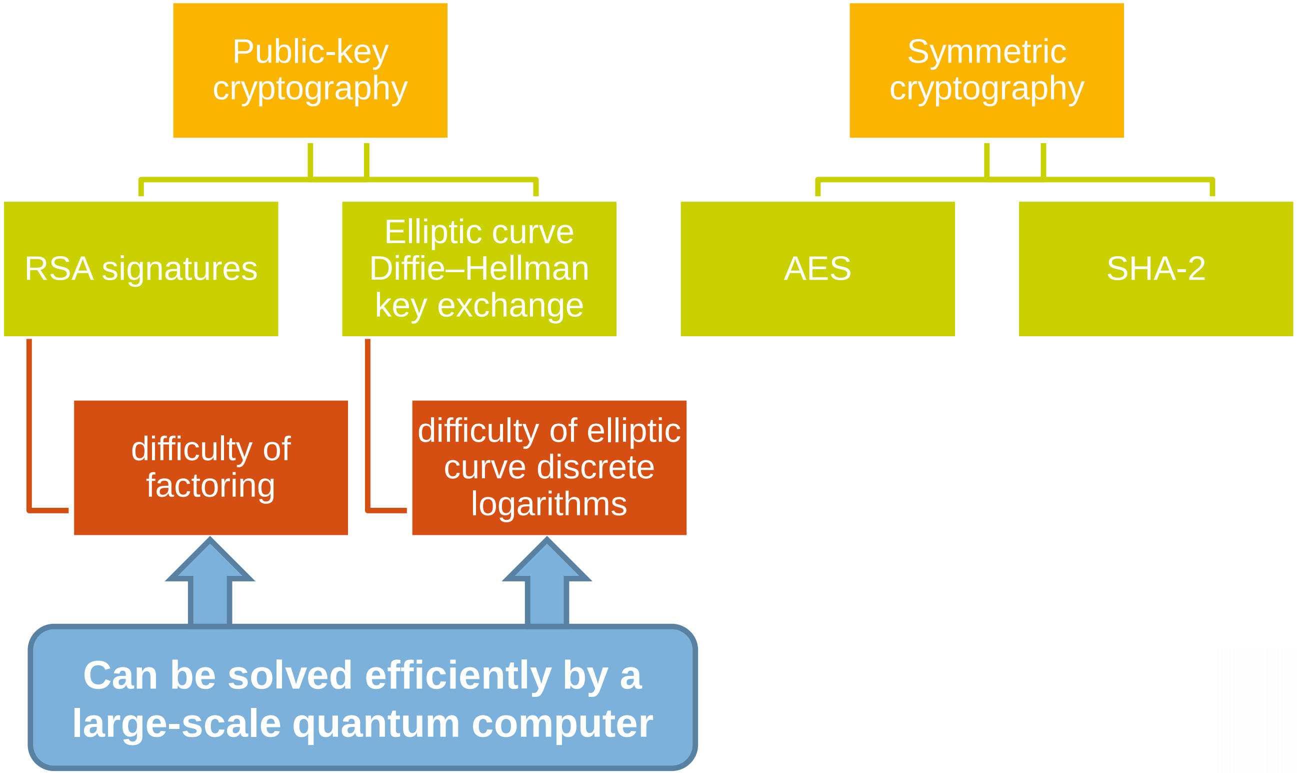
1. J. W. Bos, C. Costello, M. Naehrig, and Douglas Stebila. **Post-quantum key exchange for the TLS protocol from the ring learning with errors problem. In** IEEE Symposium on Security and Privacy – S&P, pp. 553-570, IEEE Computer Society, 2015.

2. J. W. Bos, C. Costello, L. Ducas, I. Mironov, M. Naehrig, V. Nikolaenko, A. Raghunathan and D. Stebila. **Frodo: Take off the ring! Practical, Quantum-Secure Key Exchange from LWE** in 23rd ACM Conference on Computer and Communications Security (CCS 2016).

MOTIVATION

Contemporary cryptography

`TLS-ECDHE-RSA-AES128-GCM-SHA256`



Public-key cryptography

Symmetric cryptography

RSA signatures

Elliptic curve Diffie–Hellman key exchange

AES

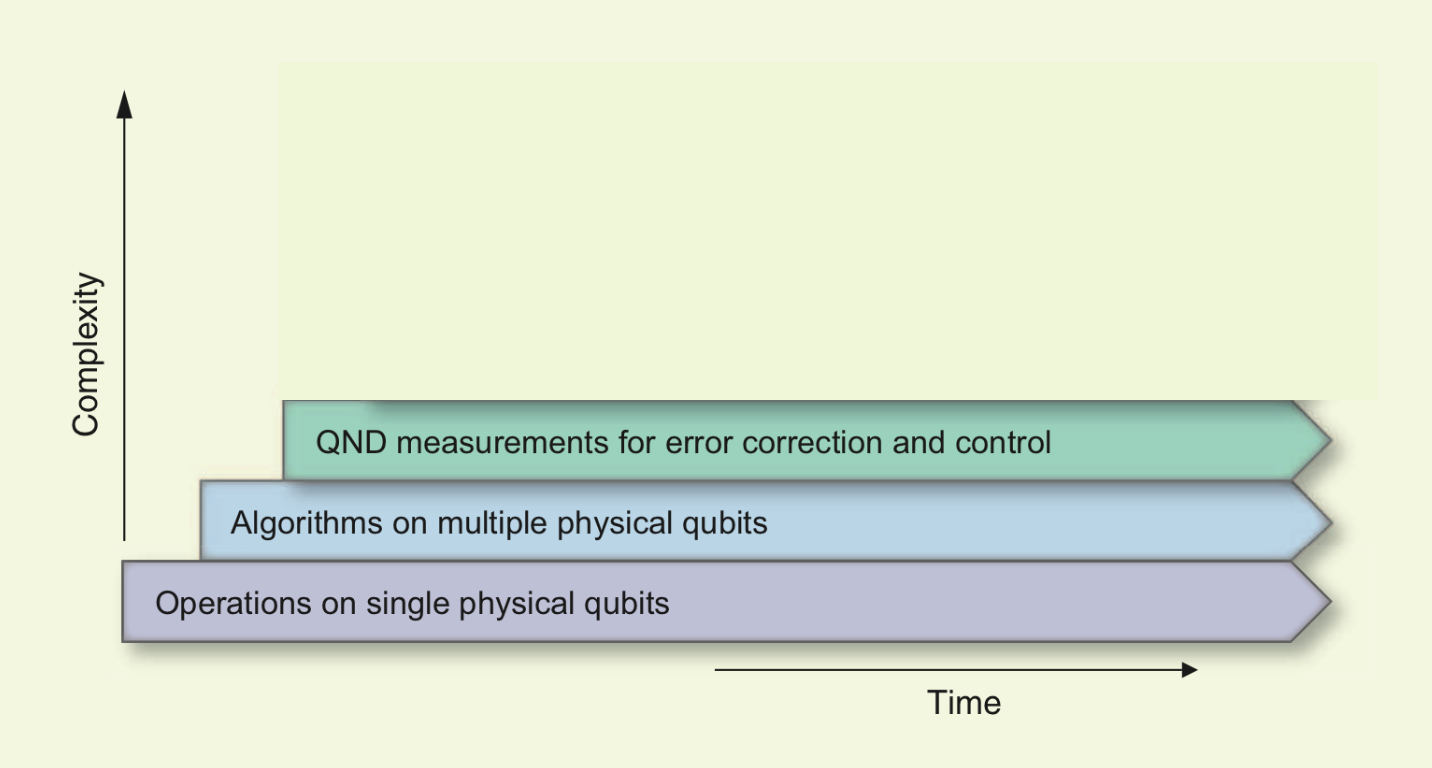
SHA-2

difficulty of factoring

difficulty of elliptic curve discrete logarithms

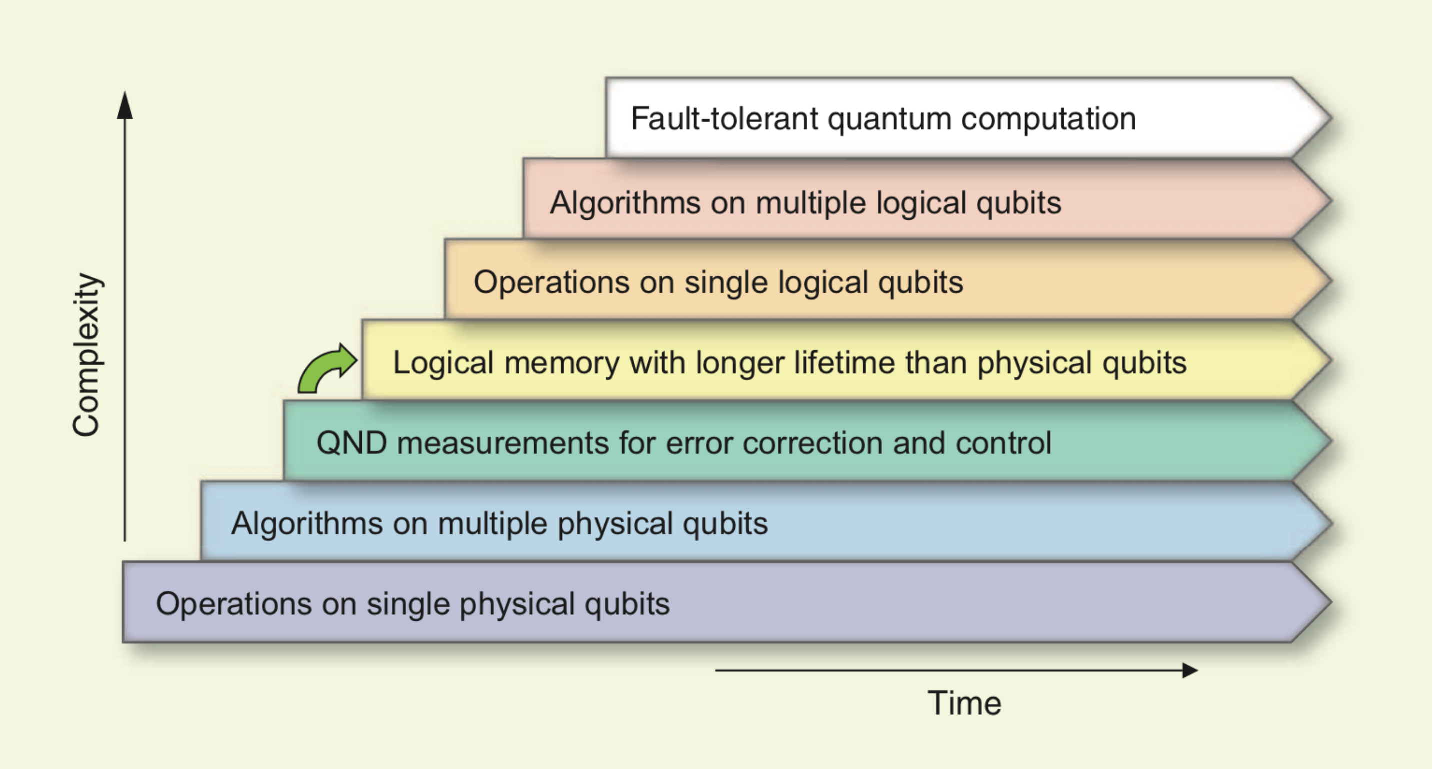
Can be solved efficiently by a large-scale quantum computer

Building quantum computers


Complexity
QND measurements for error correction and control
Algorithms on multiple physical qubits
Operations on single physical qubits
Time

Devoret, Schoelkopf. *Science* 339:1169–1174, March 2013.

Building quantum computers


Fault-tolerant quantum computation
Algorithms on multiple logical qubits
Operations on single logical qubits
Logical memory with longer lifetime than physical qubits
QND measurements for error correction and control
Algorithms on multiple physical qubits
Operations on single physical qubits
Complexity
Time

Devoret, Schoelkopf. *Science* 339:1169–1174, March 2013.

When will a large-scale quantum computer be built?

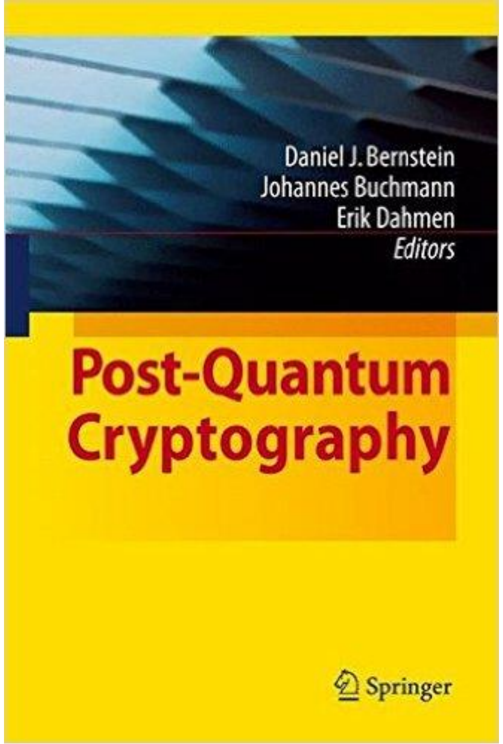
“I estimate a 1/7 chance of
breaking RSA-2048 by 2026
and a 1/2 chance by 2031.”

— Michele Mosca, November 2015
https://eprint.iacr.org/2015/1075

Post-quantum cryptography in academia

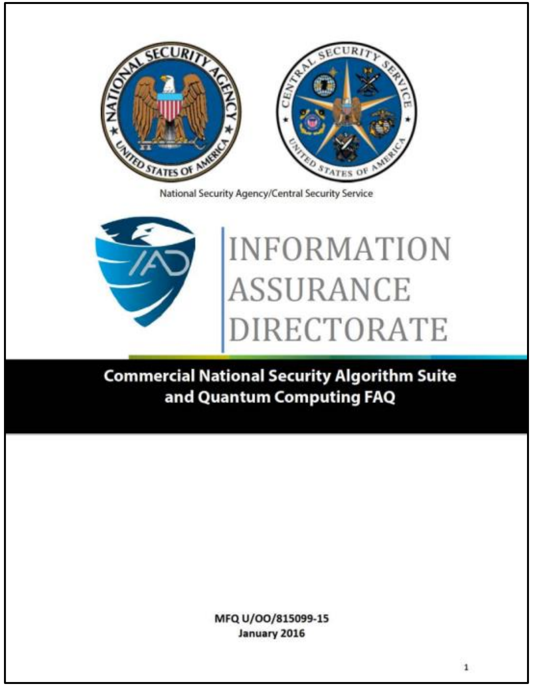
Conference series

- PQCrypto 2006
- PQCrypto 2008
- PQCrypto 2010
- PQCrypto 2011
- PQCrypto 2013
- PQCrypto 2014
- PQCrypto 2016



2009

Post-quantum cryptography in government



National Security Agency/Central Security Service

INFORMATION
ASSURANCE
DIRECTORATE

Commercial National Security Algorithm Suite and Quantum Computing FAQ

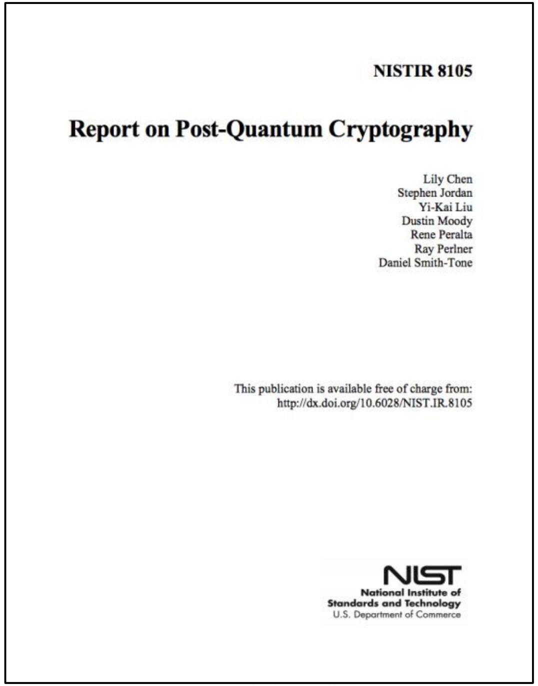
MFQ U/OO/815099-15
January 2016

1

Aug. 2015 (Jan. 2016)

“IAD will initiate a transition to quantum resistant algorithms in the not too distant future.”

– NSA Information Assurance Directorate, Aug. 2015



NISTIR 8105

Report on Post-Quantum Cryptography

Lily Chen
Stephen Jordan
Yi-Kai Liu
Dustin Moody
Rene Peralta
Ray Perlner
Daniel Smith-Tone

This publication is available free of charge from:
http://dx.doi.org/10.6028/NIST.IR.8105

NIST
National Institute of
Standards and Technology
U.S. Department of Commerce

Apr. 2016

NIST Post-quantum Crypto Project timeline

September 16, 2016	**Feedback on call for proposals**
Fall 2016	Formal call for proposals
November 2017	Deadline for submissions
Early 2018	Workshop – submitters’ presentations
3-5 years	Analysis phase
2 years later	Draft standards ready

http://www.nist.gov/pqcrypto

Post-quantum / quantum-safe crypto

No known exponential quantum speedup

Hash-based

- Merkle signatures
- Sphincs

Code-based

- McEliece

Multivariate

- multivariate quadratic

Lattice-based

- NTRU
- learning with errors
- ring-LWE

Isogenies

- supersingular elliptic curve isogenies

Lots of questions

- Design better post-quantum key exchange and signature schemes
- Improve classical and quantum attacks
- Pick parameter sizes
- Develop fast, secure implementations
- Integrate them into the existing infrastructure

This talk

- Two key exchange protocols from lattice-based problems
  - BCNS15: key exchange from the **ring** learning with errors problem
  - Frodo: key exchange from the learning with errors problem

Why key exchange?

Premise: large-scale quantum computers don’t exist right now, but we want to protect today’s communications against tomorrow’s adversary.

- Signatures still done with traditional primitives (RSA/ECDSA)
  - we only need authentication to be secure *now*
  - benefit: use existing RSA-based PKI
- Key agreement done with ring-LWE, LWE, …
  - Also consider “hybrid” ciphersuites that use post-quantum and traditional elliptic curve

LEARNING WITH ERROR PROBLEM

Solving systems of linear equations

$$\overset{\mathbb{Z}_{13}^{7\times 4}}{\begin{bmatrix} 4 & 1 & 11 & 10 \\ 5 & 5 & 9 & 5 \\ 3 & 9 & 0 & 10 \\ 1 & 3 & 3 & 2 \\ 12 & 7 & 3 & 4 \\ 6 & 5 & 11 & 4 \\ 3 & 3 & 5 & 0 \end{bmatrix}} \times \overset{\substack{\textbf{secret} \\ \mathbb{Z}_{13}^{4\times 1}}}{\begin{bmatrix} \\ \\ \\ \end{bmatrix}} = \overset{\mathbb{Z}_{13}^{7\times 1}}{\begin{bmatrix} 4 \\ 8 \\ 1 \\ 10 \\ 4 \\ 12 \\ 9 \end{bmatrix}}$$

Linear system problem: given blue, find red

Solving systems of linear equations

$$\overset{\mathbb{Z}_{13}^{7\times 4}}{\begin{bmatrix} 4 & 1 & 11 & 10 \\ 5 & 5 & 9 & 5 \\ 3 & 9 & 0 & 10 \\ 1 & 3 & 3 & 2 \\ 12 & 7 & 3 & 4 \\ 6 & 5 & 11 & 4 \\ 3 & 3 & 5 & 0 \end{bmatrix}} \times \overset{\substack{\textbf{secret} \\ \mathbb{Z}_{13}^{4\times 1}}}{\begin{bmatrix} 6 \\ 9 \\ 11 \\ 11 \end{bmatrix}} = \overset{\mathbb{Z}_{13}^{7\times 1}}{\begin{bmatrix} 4 \\ 8 \\ 1 \\ 10 \\ 4 \\ 12 \\ 9 \end{bmatrix}}$$

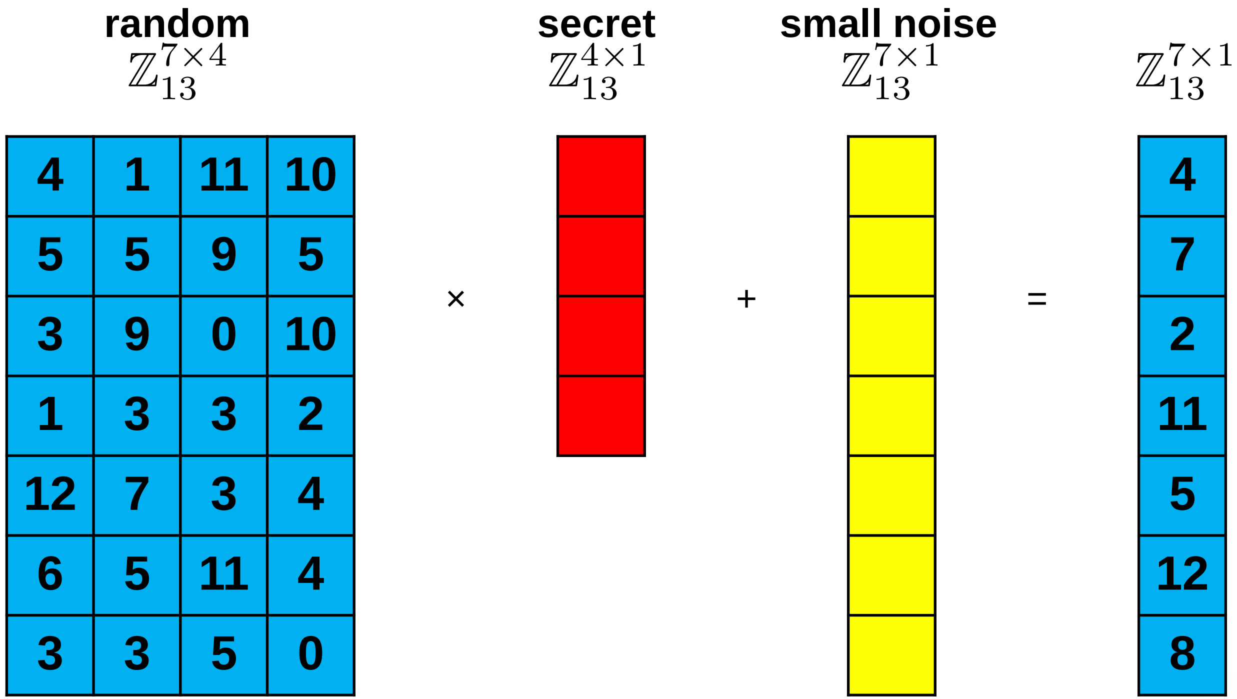
Easily solved using Gaussian elimination (Linear Algebra 101)

Linear system problem: given **blue**, find **red**

Learning with errors problem

$$\begin{array}{ccccccc}
\textbf{random} & & \textbf{secret} & & \textbf{small noise} & & \\
\mathbb{Z}_{13}^{7\times 4} & & \mathbb{Z}_{13}^{4\times 1} & & \mathbb{Z}_{13}^{7\times 1} & & \mathbb{Z}_{13}^{7\times 1} \\
\begin{bmatrix} 4 & 1 & 11 & 10 \\ 5 & 5 & 9 & 5 \\ 3 & 9 & 0 & 10 \\ 1 & 3 & 3 & 2 \\ 12 & 7 & 3 & 4 \\ 6 & 5 & 11 & 4 \\ 3 & 3 & 5 & 0 \end{bmatrix}
& \times &
\begin{bmatrix} 6 \\ 9 \\ 11 \\ 11 \end{bmatrix}
& + &
\begin{bmatrix} 0 \\ -1 \\ 1 \\ 1 \\ 1 \\ 0 \\ -1 \end{bmatrix}
& = &
\begin{bmatrix} 4 \\ 7 \\ 2 \\ 11 \\ 5 \\ 12 \\ 8 \end{bmatrix}
\end{array}$$

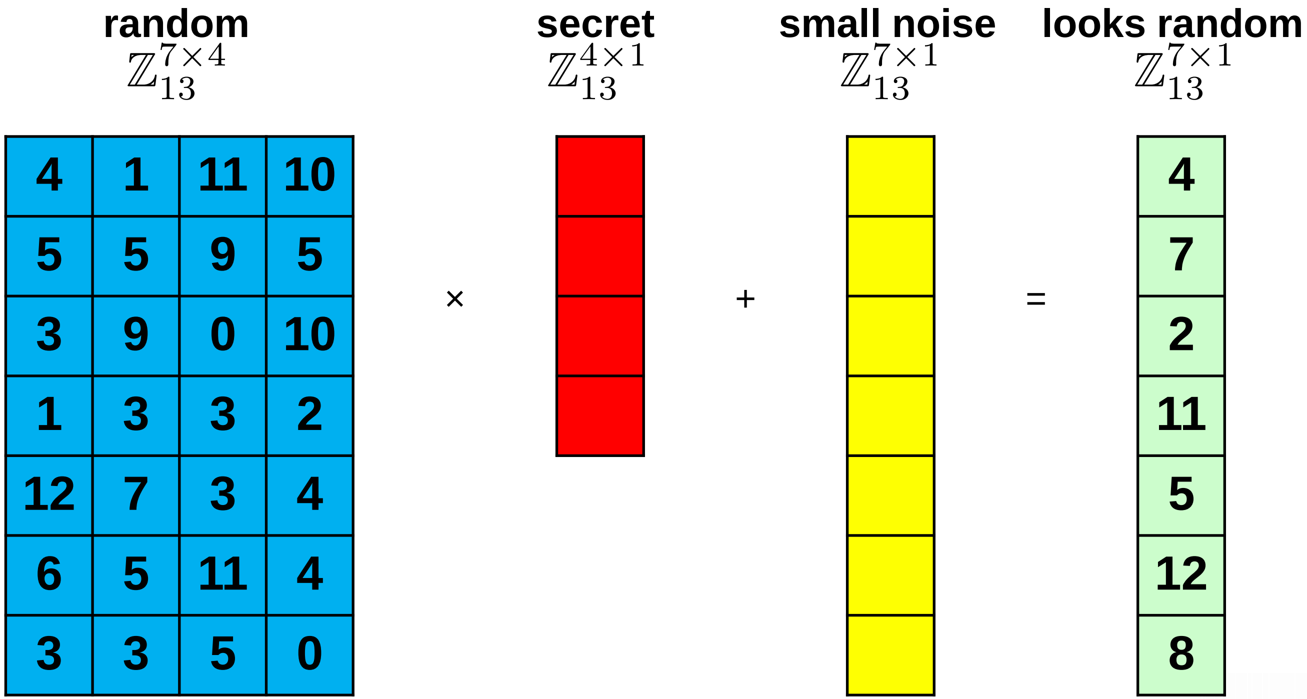
Learning with errors problem



$$\begin{array}{ccccccc} \textbf{random} & & \textbf{secret} & & \textbf{small noise} & & \\ \mathbb{Z}_{13}^{7\times 4} & & \mathbb{Z}_{13}^{4\times 1} & & \mathbb{Z}_{13}^{7\times 1} & & \mathbb{Z}_{13}^{7\times 1} \\ \begin{pmatrix} 4 & 1 & 11 & 10 \\ 5 & 5 & 9 & 5 \\ 3 & 9 & 0 & 10 \\ 1 & 3 & 3 & 2 \\ 12 & 7 & 3 & 4 \\ 6 & 5 & 11 & 4 \\ 3 & 3 & 5 & 0 \end{pmatrix} & \times & \begin{pmatrix} \, \\ \, \\ \, \\ \, \end{pmatrix} & + & \begin{pmatrix} \, \\ \, \\ \, \\ \, \\ \, \\ \, \\ \, \end{pmatrix} & = & \begin{pmatrix} 4 \\ 7 \\ 2 \\ 11 \\ 5 \\ 12 \\ 8 \end{pmatrix} \end{array}$$

Computational LWE problem: given **blue**, find **red**

Decision learning with errors problem



$$\begin{array}{ccccccc} \text{random} & & \text{secret} & & \text{small noise} & & \text{looks random} \\ \mathbb{Z}_{13}^{7\times 4} & & \mathbb{Z}_{13}^{4\times 1} & & \mathbb{Z}_{13}^{7\times 1} & & \mathbb{Z}_{13}^{7\times 1} \\ \begin{bmatrix} 4 & 1 & 11 & 10 \\ 5 & 5 & 9 & 5 \\ 3 & 9 & 0 & 10 \\ 1 & 3 & 3 & 2 \\ 12 & 7 & 3 & 4 \\ 6 & 5 & 11 & 4 \\ 3 & 3 & 5 & 0 \end{bmatrix} & \times & \begin{bmatrix} \\ \\ \\ \end{bmatrix} & + & \begin{bmatrix} \\ \\ \\ \\ \\ \\ \end{bmatrix} & = & \begin{bmatrix} 4 \\ 7 \\ 2 \\ 11 \\ 5 \\ 12 \\ 8 \end{bmatrix} \end{array}$$

Decision LWE problem: given **blue**, distinguish **green** from random

Toy example versus real-world example

$\mathbb{Z}_{13}^{7\times 4}$

4	1	11	10
5	5	9	5
3	9	0	10
1	3	3	2
12	7	3	4
6	5	11	4
3	3	5	0

$\mathbb{Z}_{2^{15}}^{752\times 8}$

8

752

2738	3842	3345	2979	…
2896	595	3607		
377	1575			
2760				
…				

752 × 8 × 15 bits = **11 KiB**

Ring learning with errors problem

random

$\mathbb{Z}_{13}^{7\times 4}$

4	1	11	10
10	4	1	11
11	10	4	1
1	11	10	4
4	1	11	10
10	4	1	11
11	10	4	1

Each row is the cyclic shift of the row above

Ring learning with errors problem

random

$\mathbb{Z}_{13}^{7\times 4}$

4	1	11	10
3	4	1	11
2	3	4	1
12	2	3	4
9	12	2	3
10	9	12	2
11	10	9	12

Each row is the cyclic shift of the row above

…

with a special wrapping rule: *x* wraps to –*x* mod 13.

Ring learning with errors problem

random
$\mathbb{Z}_{13}^{7\times 4}$

4	**1**	**11**	**10**

Each row is the cyclic
shift of the row above
…
with a special wrapping rule:
x wraps to –*x* mod 13 (→ $\mathbb{Z}_{13}[x]/\langle x^4+1\rangle$)

So I only need to tell you the first row.

Ring learning with errors problem

$\mathbb{Z}_{13}[x]/\langle x^4 + 1\rangle$

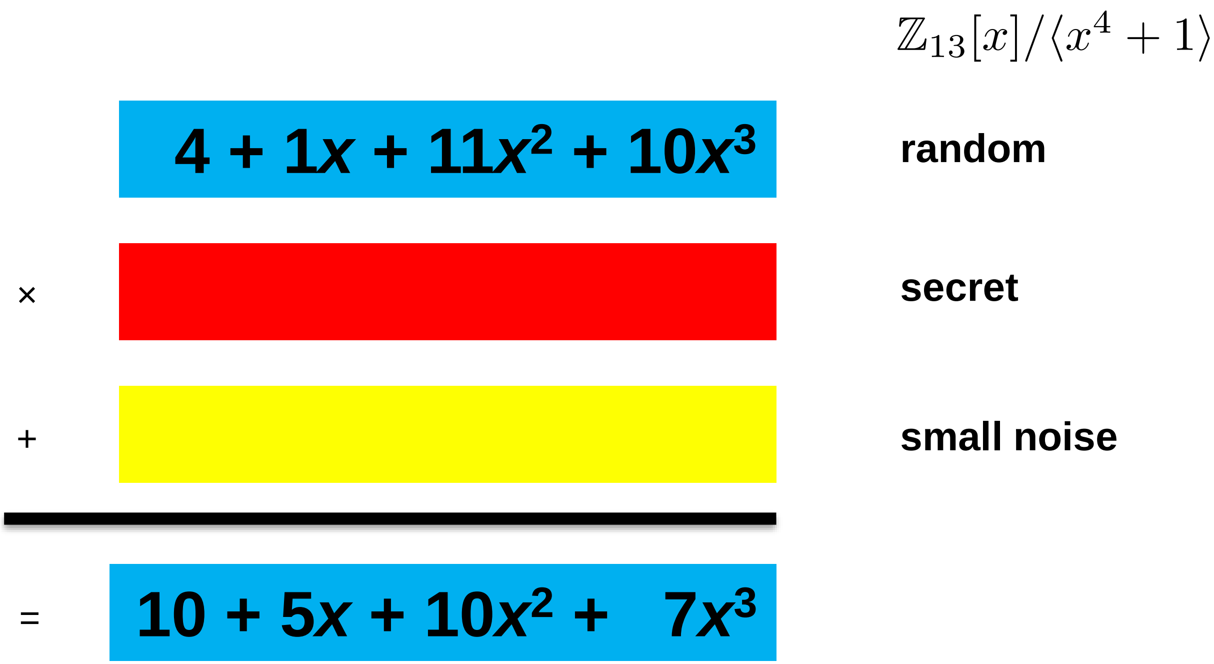
$4 + 1x + 11x^2 + 10x^3$ **random**

$\times\quad 6 + 9x + 11x^2 + 11x^3$ **secret**

$+\quad 0 - 1x + 1x^2 + 1x^3$ **small noise**

$=\quad 10 + 5x + 10x^2 + 7x^3$

Ring learning with errors problem



$\mathbb{Z}_{13}[x]/\langle x^4 + 1\rangle$

$4 + 1x + 11x^2 + 10x^3$ **random**

× **secret**

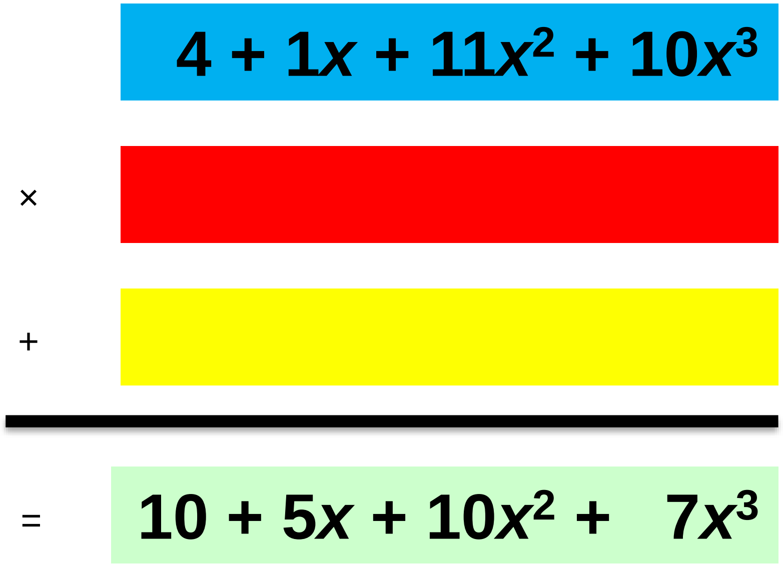
+ **small noise**

= $10 + 5x + 10x^2 + 7x^3$

Computational ring-LWE problem: given **blue**, find **red**

Decision ring learning with errors problem

$\mathbb{Z}_{13}[x]/\langle x^4 + 1\rangle$



$4 + 1x + 11x^2 + 10x^3$ random

× secret

+ small noise

= $10 + 5x + 10x^2 + 7x^3$ looks random

Decision ring-LWE problem: given **blue**, distinguish **green** from random

Decision ring learning with errors problem with small secrets

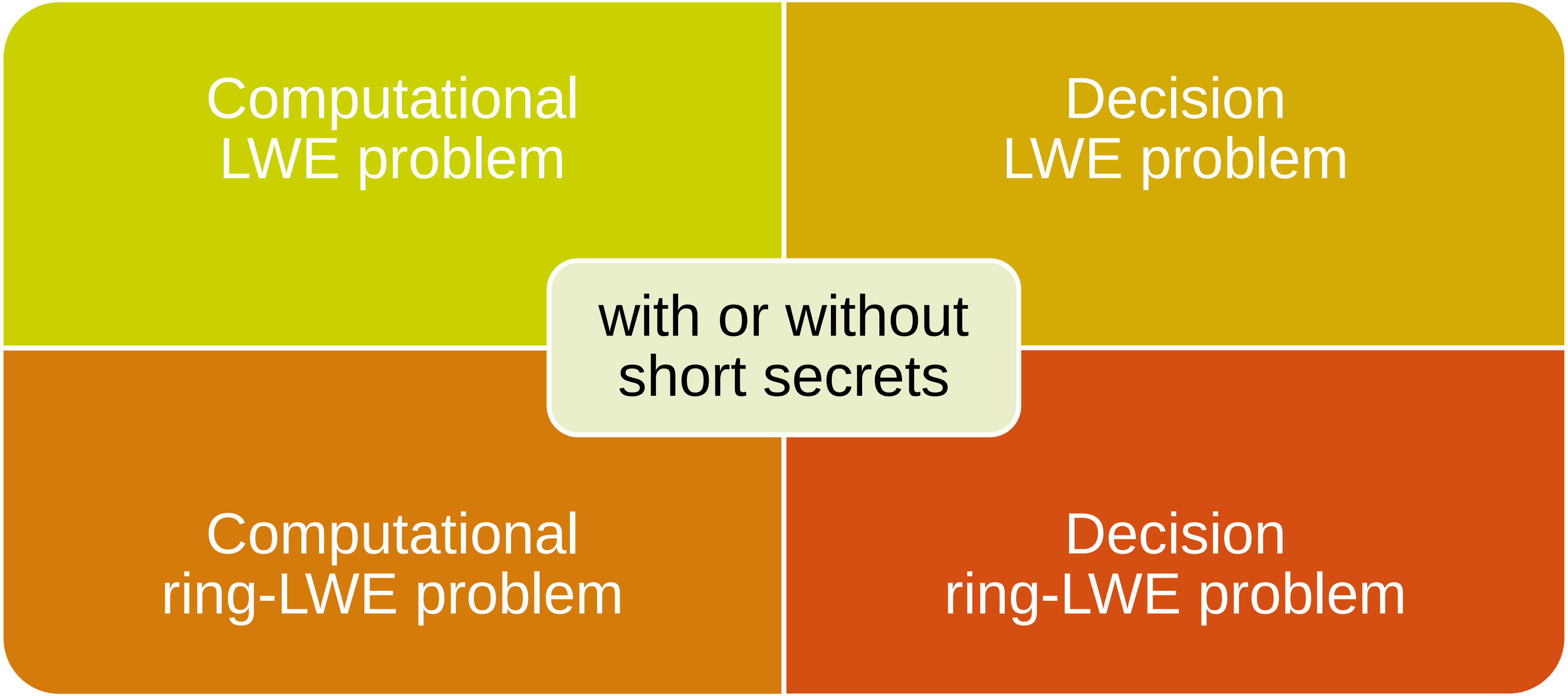
$$\mathbb{Z}_{13}[x]/\langle x^4 + 1\rangle$$

	$4 + 1x + 11x^2 + 10x^3$	random
×	$1 + 0x - 1x^2 + 2x^3$	small secret
+		small noise
=	$10 + 5x + 10x^2 + 7x^3$	looks random

Decision ring-LWE problem: given blue, distinguish green from random

Problems

[Reg05] Regev, *STOC 2005*; *J. ACM 2009*.



Computational LWE problem

Decision LWE problem

with or without short secrets

Computational ring-LWE problem

Decision ring-LWE problem

[LPR10] Lyubashevsky, Peikert, Regev. *EUROCRYPT 2010.*

KEY AGREEMENT FROM RING-LWE

Decision ring learning with errors problem with short secrets

Definition. Let n be a power of 2, q be a prime, and $R_q = \mathbb{Z}_q[X]/(X^n+1)$ be the ring of polynomials in X with integer coefficients modulo q and polynomial reduction modulo X^n+1. Let χ be a distribution over R_q.

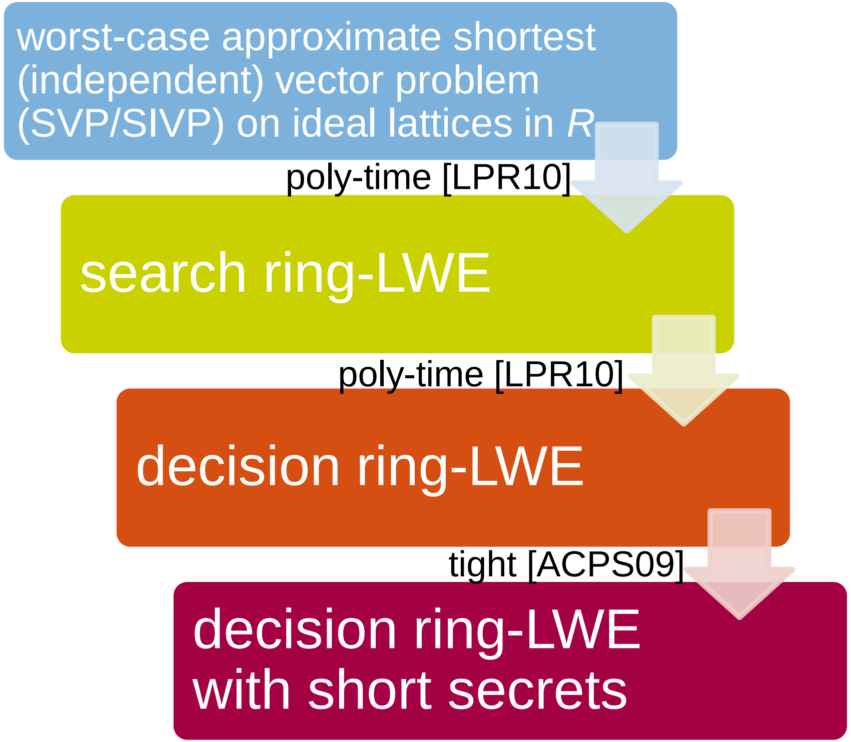
Let $s \overset{\$}{\leftarrow} \chi$.

Define:

- $O_{\chi,s}$: Sample $a \overset{\$}{\leftarrow} \mathcal{U}(R_q)$, $e \overset{\$}{\leftarrow} \chi$; return $(a, as+e)$.
- U: Sample $(a, b') \overset{\$}{\leftarrow} \mathcal{U}(R_q \times R_q)$; return (a, b').

The *decision R-LWE problem with short secrets* for n, q, χ is to distinguish $O_{\chi,s}$ from U.

Hardness of decision ring-LWE



worst-case approximate shortest (independent) vector problem (SVP/SIVP) on ideal lattices in *R*

poly-time [LPR10]

search ring-LWE

poly-time [LPR10]

decision ring-LWE

tight [ACPS09]

decision ring-LWE with short secrets

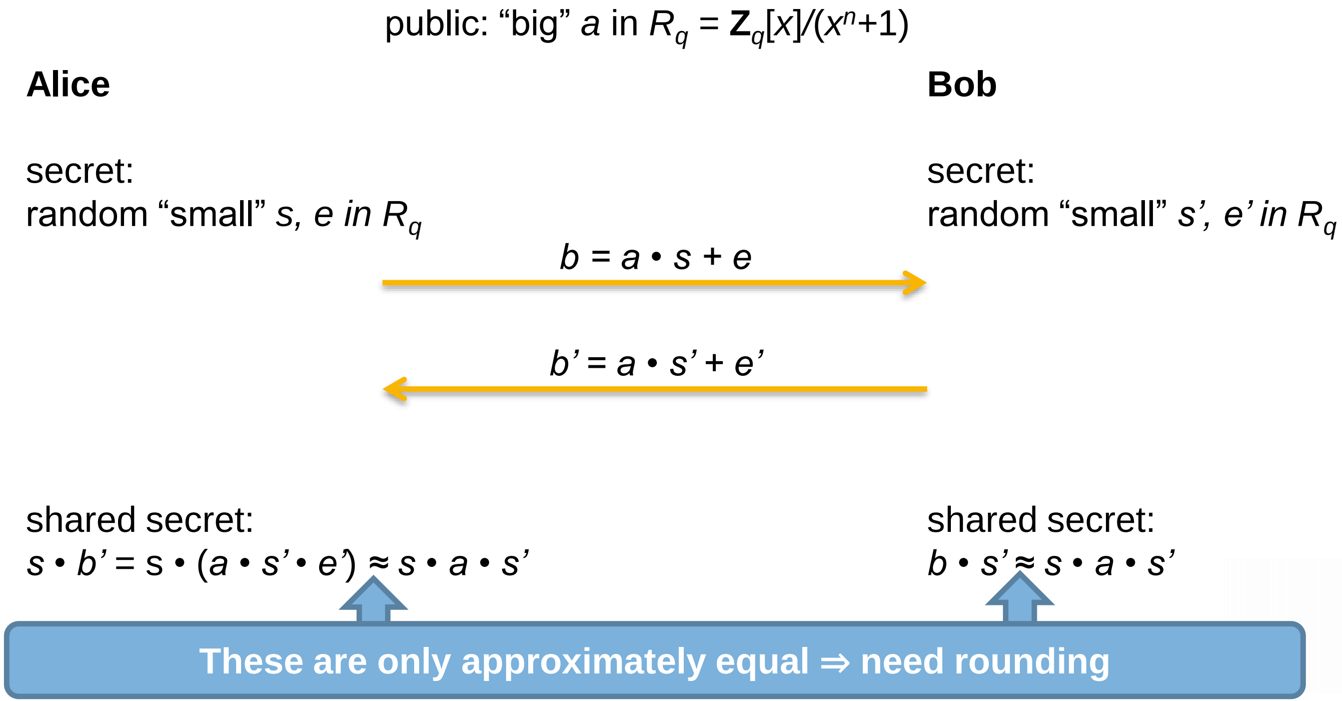
<u>Practice:</u>

- Assume the best way to solve DRLWE is to solve LWE.
- Assume solving LWE involves a lattice reduction problem.
- Estimate parameters based on runtime of lattice reduction algorithms e.g. [APS15]
- (Ignore non-tightness.) [CKMS16]

[LPR10] Lyubashevsky, Peikert, Regev. *EUROCRYPT 2010.*
[ACPS15] Applebaum, Cash, Peikert, Sahai. *CRYPTO 2009*.
[CKMS16] Chatterjee, Koblitz, Menezes, Sarkar. ePrint 2016/360.

Basic ring-LWE-DH key agreement (unauthenticated)

- Reformulation of Peikert’s ring-LWE KEM (*PQCrypto 2014*)



public: “big” a in $R_q = \mathbf{Z}_q[x]/(x^n+1)$

Alice

secret:
random “small” s, e *in* R_q

Bob

secret:
random “small” s', e' *in* R_q

$b = a \bullet s + e$

$b' = a \bullet s' + e'$

shared secret:
$s \bullet b' = s \bullet (a \bullet s' \bullet e') \approx s \bullet a \bullet s'$

shared secret:
$b \bullet s' \approx s \bullet a \bullet s'$

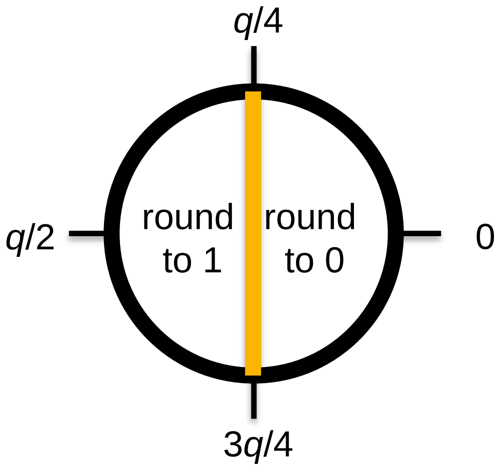
These are only approximately equal ⇒ need rounding

Rounding

- Each coefficient of the polynomial is an integer modulo *q*
- Treat each coefficient independently

Basic rounding

- Round either to 0 or *q*/2
- Treat *q*/2 as 1

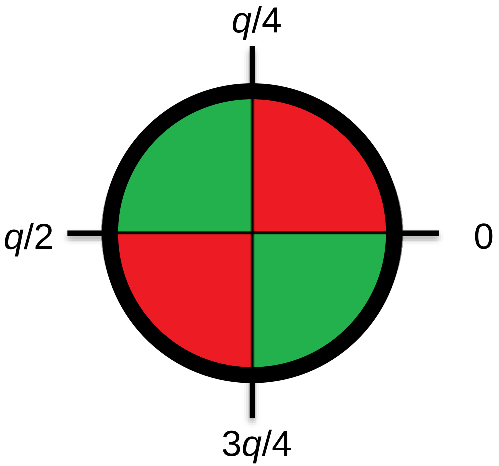


This works most of the time: prob. failure 2^{-10}.

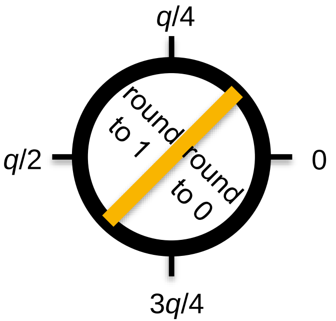
Not good enough: we need exact key agreement.

Better rounding (Peikert)

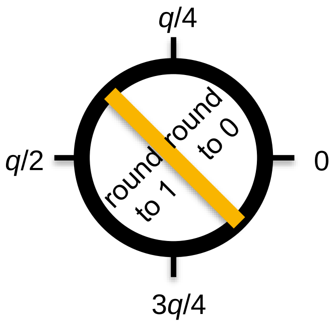
Bob says which of two regions the value is in: 🟩 or 🟥



If 🟩



If 🟥



Prob. Failure is less than 2^{-128}

Security not affected: revealing 🟩 or 🟥 leaks no information

Exact ring-LWE-DH key agreement (unauthenticated)

- Reformulation of Peikert's R-LWE KEM (*PQCrypto 2014*)

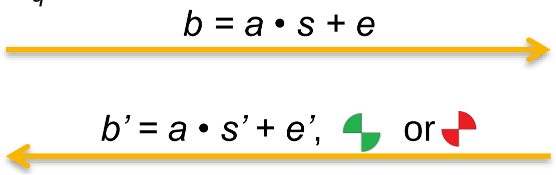
public: "big" a in $R_q = \mathbf{Z}_q[x]/(x^n+1)$

Alice

secret:
random "small" s, e in R_q

Bob

secret:
random "small" s', e' in R_q



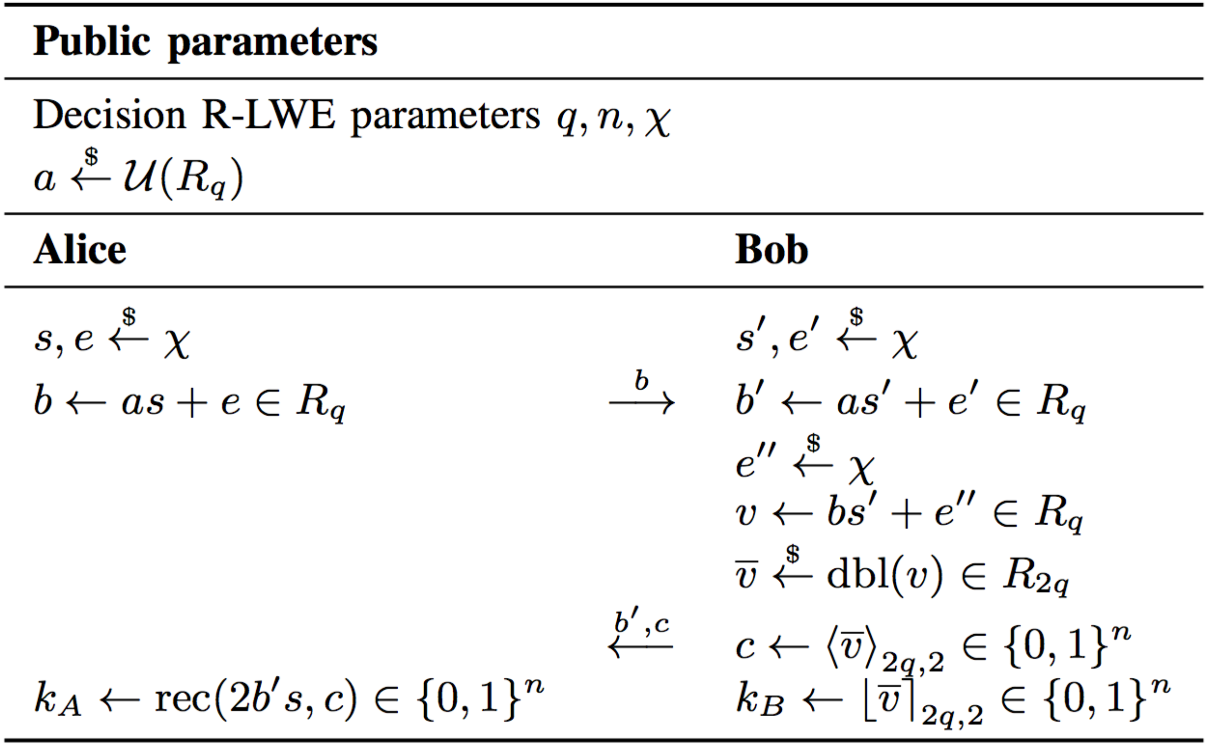
$b = a \bullet s + e$

$b' = a \bullet s' + e'$, 🟩 or 🟥

shared secret:
round($s \bullet b'$)

shared secret:
round($b \bullet s'$)

Ring-LWE-DH key agreement



Public parameters

Decision R-LWE parameters q, n, χ

$a \overset{\$}{\leftarrow} \mathcal{U}(R_q)$

Alice		**Bob**
$s, e \overset{\$}{\leftarrow} \chi$		$s', e' \overset{\$}{\leftarrow} \chi$
$b \leftarrow as + e \in R_q$	$\xrightarrow{b}$	$b' \leftarrow as' + e' \in R_q$
		$e'' \overset{\$}{\leftarrow} \chi$
		$v \leftarrow bs' + e'' \in R_q$
		$\overline{v} \overset{\$}{\leftarrow} \mathrm{dbl}(v) \in R_{2q}$
	$\xleftarrow{b', c}$	$c \leftarrow \langle \overline{v} \rangle_{2q,2} \in \{0,1\}^n$
$k_A \leftarrow \mathrm{rec}(2b's, c) \in \{0,1\}^n$		$k_B \leftarrow \lfloor \overline{v} \rceil_{2q,2} \in \{0,1\}^n$

Secure if decision ring learning with errors problem is hard.

Parameters

160-bit classical security,
80-bit quantum security

- $n = 1024$
- $q = 2^{32}–1$
- χ = discrete Gaussian with parameter sigma = 8/sqrt(2π)

- Failure: 2^{-128}

- Total communication: 8.1 KiB

Implementation aspect 1: Polynomial arithmetic

Polynomial multiplication in $R_q = \mathbb{Z}_q[X]/(X^{2^{10}} + 1)$ done with Nussbaumer's FFT

H. J. Nussbaumer. Fast polynomial transform algorithms for digital convolution. Acoustics, Speech and Signal Processing, IEEE Transactions on, 1980

Decompose $R = \mathbb{Z}[X]/(X^n + 1)$ into two extensions.
Let $n = 2^k = s \cdot r$ such that $s \mid r$. Then

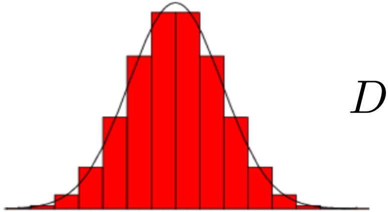
$$R \cong S = T[X]/(X^s - Z), \text{ where } T = \mathbb{Z}[Z]/(Z^r + 1)$$

Note: $Z^{r/s}$ is an s^{th} root of -1 in T and $X^s = Z$ in S.

Allow to compute the DFT symbolically in T.

Implementation aspect 2:

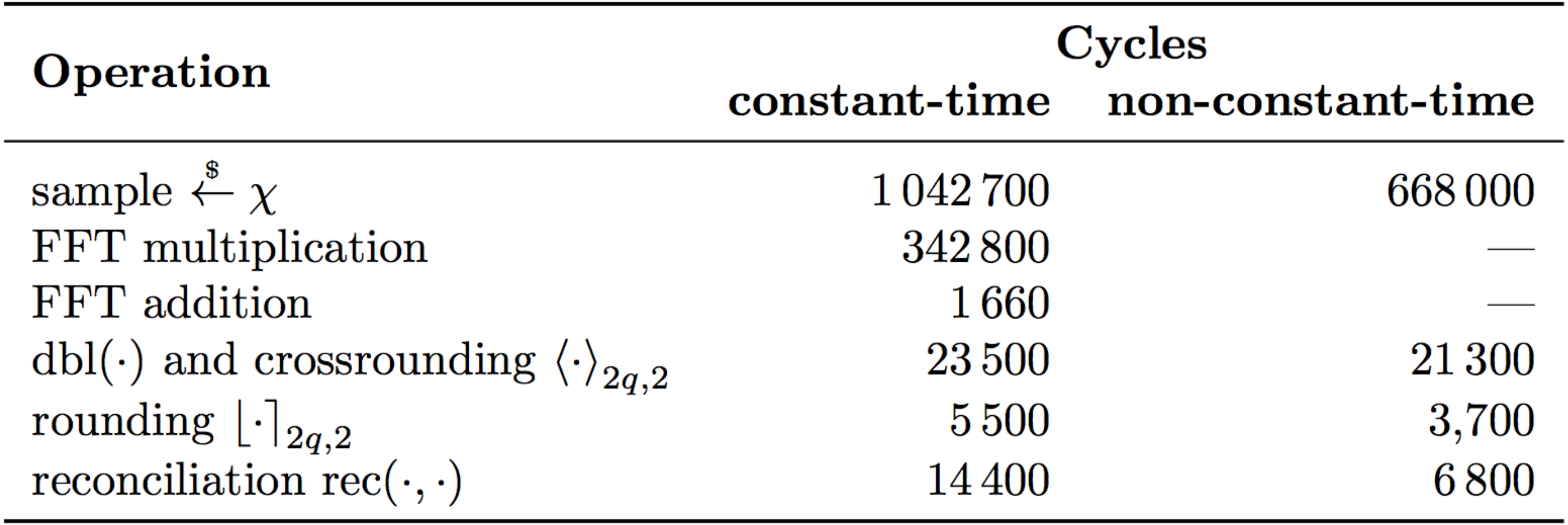
Sampling discrete Gaussians



$$D_{\mathbb{Z},\sigma}(x) = \frac{1}{S} e^{-\frac{x^2}{2\sigma^2}} \qquad \text{for } x \in \mathbb{Z}, \sigma \approx 3.2, S = 8$$

- Security proofs require “small” elements sampled within statistical distance 2^{-128} of the true discrete Gaussian
- We use inversion sampling: precompute table of cumulative probabilities
  - For us: 52 elements, size = 1248 bytes
- Sampling each coefficient requires six 192-bit integer comparisons and there are 1024 coefficients
  - 51 table entries and 1024 coefficients $\approx 52k$ comparisons for constant time

Sampling is expensive



Operation	Cycles	
	constant-time	non-constant-time
sample $\overset{\$}{\leftarrow} \chi$	1 042 700	668 000
FFT multiplication	342 800	—
FFT addition	1 660	—
dbl$(\cdot)$ and crossrounding $\langle \cdot \rangle_{2q,2}$	23 500	21 300
rounding $\lfloor \cdot \rceil_{2q,2}$	5 500	3,700
reconciliation rec$(\cdot, \cdot)$	14 400	6 800

“NewHope”

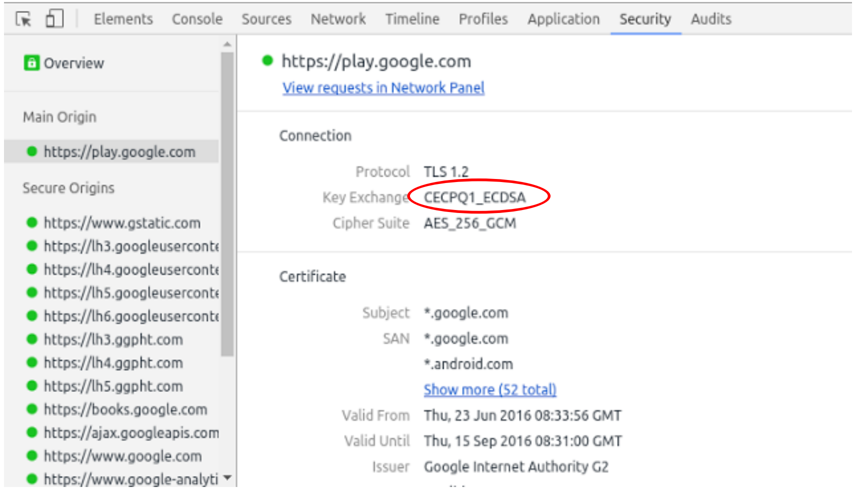
Alkim, Ducas, Pöppelman, Schwabe.
USENIX Security 2016

- New parameters
- Different error distribution
- Improved performance
- Pseudorandomly generated parameters
- Further performance improvements by others [GS16,LN16,…]

Google Security Blog

Experimenting with Post-Quantum Cryptography

July 7, 2016



Main Origin

- https://play.google.com

Secure Origins

- https://www.gstatic.com
- https://lh3.googleuserconte
- https://lh4.googleuserconte
- https://lh5.googleuserconte
- https://lh6.googleuserconte
- https://lh3.ggpht.com
- https://lh4.ggpht.com
- https://lh5.ggpht.com
- https://books.google.com
- https://ajax.googleapis.com
- https://www.google.com
- https://www.google-analyti

https://play.google.com

Connection

Protocol	TLS 1.2
Key Exchange	CECPQ1_ECDSA
Cipher Suite	AES_256_GCM

Certificate

Subject	*.google.com
SAN	*.google.com *.android.com
Valid From	Thu, 23 Jun 2016 08:33:56 GMT
Valid Until	Thu, 15 Sep 2016 08:31:00 GMT
Issuer	Google Internet Authority G2

[GS16] Gueron, Schlieker. ePrint 2016/467.
[LN16] Longa, Naehrig. ePrint 2016/504.

https://security.googleblog.com/2016/07/experimenting-with-post-quantum.html

KEY AGREEMENT FROM LWE

Bos, Costello, Ducas, Mironov, Naehrig, Nikolaenko, Raghunathan, Stebila.
Frodo: Take off the ring! Practical, quantum-safe key exchange from LWE.
ACM Conference on Computer and Communications Security (CCS) 2016.

See: https://eprint.iacr.org/2016/659

Decision learning with errors problem with short secrets

Definition. Let $n, q \in \mathbb{N}$. Let χ be a distribution over $\mathbb{Z}$.

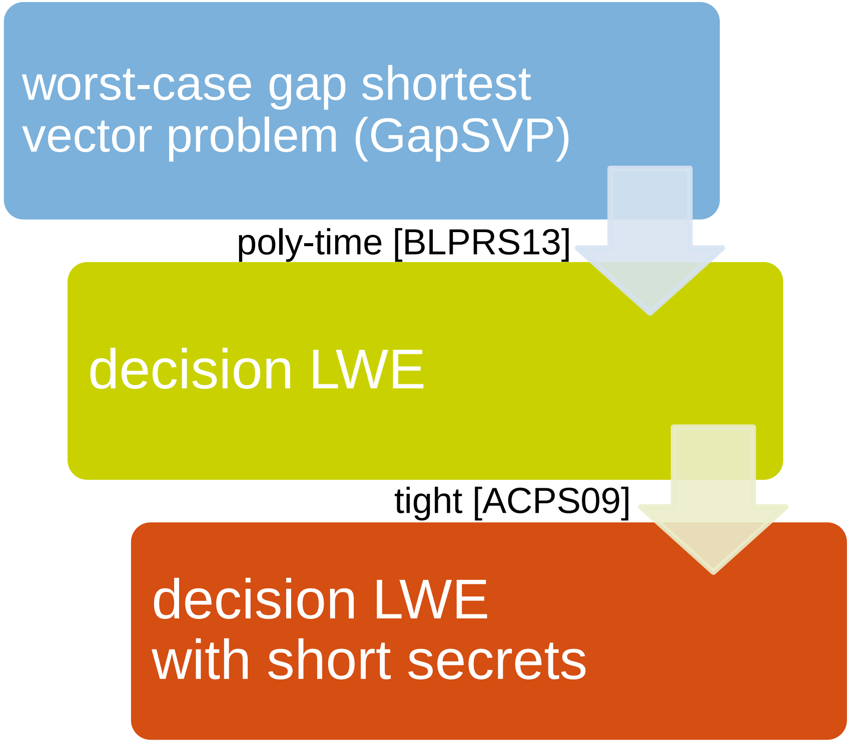
Let $\mathbf{s} \overset{\$}{\leftarrow} \chi^n$.

Define:

- $O_{\chi,\mathbf{s}}$: Sample $\mathbf{a} \overset{\$}{\leftarrow} \mathcal{U}(\mathbb{Z}_q^n)$, $e \overset{\$}{\leftarrow} \chi$; return $(\mathbf{a}, \mathbf{a} \cdot \mathbf{s} + e)$.
- U: Sample $(\mathbf{a}, b') \overset{\$}{\leftarrow} \mathcal{U}(\mathbb{Z}_q^n \times \mathbb{Z}_q)$; return $(\mathbf{a}, b')$.

The *decision LWE problem with short secrets* for n, q, χ is to distinguish $O_{\chi,\mathbf{s}}$ from U.

Hardness of decision LWE



worst-case gap shortest vector problem (GapSVP)

poly-time [BLPRS13]

decision LWE

tight [ACPS09]

decision LWE with short secrets

<u>Practice:</u>

- Assume the best way to solve DLWE is to solve LWE.
- Assume solving LWE involves a lattice reduction problem.
- Estimate parameters based on runtime of lattice reduction algorithms.
- (Ignore non-tightness.)

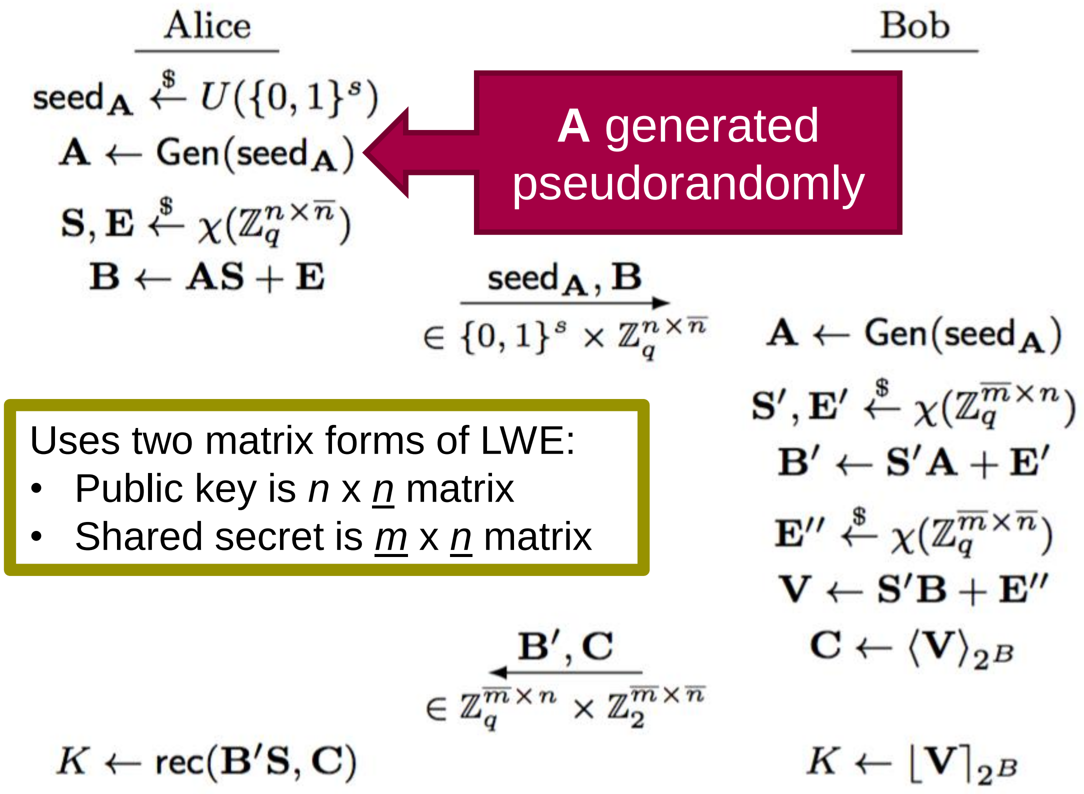
[BLPRS13] Brakerski, Langlois, Peikert, Regev, Stehlé. *STOC 2013.*
[ACPS15] Applebaum, Cash, Peikert, Sahai. *CRYPTO 2009*.

Generic vs. ideal lattices

- Ring-LWE matrices have additional structure
  - Relies on hardness of a problem in **ideal** lattices
- LWE matrices have no additional structure
  - Relies on hardness of a problem in **generic** lattices
- NTRU also relies on a problem in a type of ideal lattices

- Currently, best algorithms for ideal lattice problems are essentially the same as for generic lattices
  - Small constant factor improvement in some cases (e.g. sieving)
- If we want to eliminate this additional structure, can we still get an efficient algorithm?

“Frodo”: LWE-DH key agreement



Alice

$\mathsf{seed}_{\mathbf{A}} \xleftarrow{\$} U(\{0,1\}^s)$

$\mathbf{A} \leftarrow \mathsf{Gen}(\mathsf{seed}_{\mathbf{A}})$

$\mathbf{S}, \mathbf{E} \xleftarrow{\$} \chi(\mathbb{Z}_q^{n \times \bar{n}})$

$\mathbf{B} \leftarrow \mathbf{AS} + \mathbf{E}$

$\xrightarrow[\in \{0,1\}^s \times \mathbb{Z}_q^{n \times \bar{n}}]{\mathsf{seed}_{\mathbf{A}}, \mathbf{B}}$

Bob

$\mathbf{A} \leftarrow \mathsf{Gen}(\mathsf{seed}_{\mathbf{A}})$

$\mathbf{S}', \mathbf{E}' \xleftarrow{\$} \chi(\mathbb{Z}_q^{\bar{m} \times n})$

$\mathbf{B}' \leftarrow \mathbf{S}'\mathbf{A} + \mathbf{E}'$

$\mathbf{E}'' \xleftarrow{\$} \chi(\mathbb{Z}_q^{\bar{m} \times \bar{n}})$

$\mathbf{V} \leftarrow \mathbf{S}'\mathbf{B} + \mathbf{E}''$

$\mathbf{C} \leftarrow \langle \mathbf{V} \rangle_{2^B}$

$\xleftarrow[\in \mathbb{Z}_q^{\bar{m} \times n} \times \mathbb{Z}_2^{\bar{m} \times \bar{n}}]{\mathbf{B}', \mathbf{C}}$

$K \leftarrow \mathsf{rec}(\mathbf{B}'\mathbf{S}, \mathbf{C})$

$K \leftarrow \lfloor \mathbf{V} \rceil_{2^B}$

A generated pseudorandomly

Uses two matrix forms of LWE:

- Public key is *n* x $\underline{n}$ matrix
- Shared secret is $\underline{m}$ x $\underline{n}$ matrix

Secure if decision learning with errors problem is hard (and Gen is a secure PRF).

Parameters

All known variants of the sieving algorithm require a list of vectors to be created of this size

“Recommended”

- 156-bit classical security,
  142-bit quantum security,
  112-bit plausible lower bound

- n = 752, m = 8, q = 2^{15}
- χ = approximation to rounded Gaussian with 11 elements (< 16 bytes LUT)
- Failure: $2^{-36.5}$
- Total communication: 22.6 KiB

“Paranoid”

- 191-bit classical security,
  174-bit quantum security,
  138-bit plausible lower bound

- n = 864, m = 8, q = 2^{15}
- χ = approximation to rounded Gaussian with 13 elements (< 16 bytes LUT)
- Failure: $2^{-35.8}$
- Total communication: 25.9 KiB

Error distribution close to discrete Gaussian in terms of Rényi divergence

Improved Security Proofs in Lattice-Based Cryptography: Using the Rényi Divergence Rather Than the Statistical Distance. By S. Bai , A. Langlois, T. Lepoint, D. Stehlé, R. Steinfeld. In ASIACRYPT 2016

STANDALONE PERFORMANCE

Implementations

Our implementations

- BCNS15
- Frodo

Pure C implementations

Constant time

Compare with others

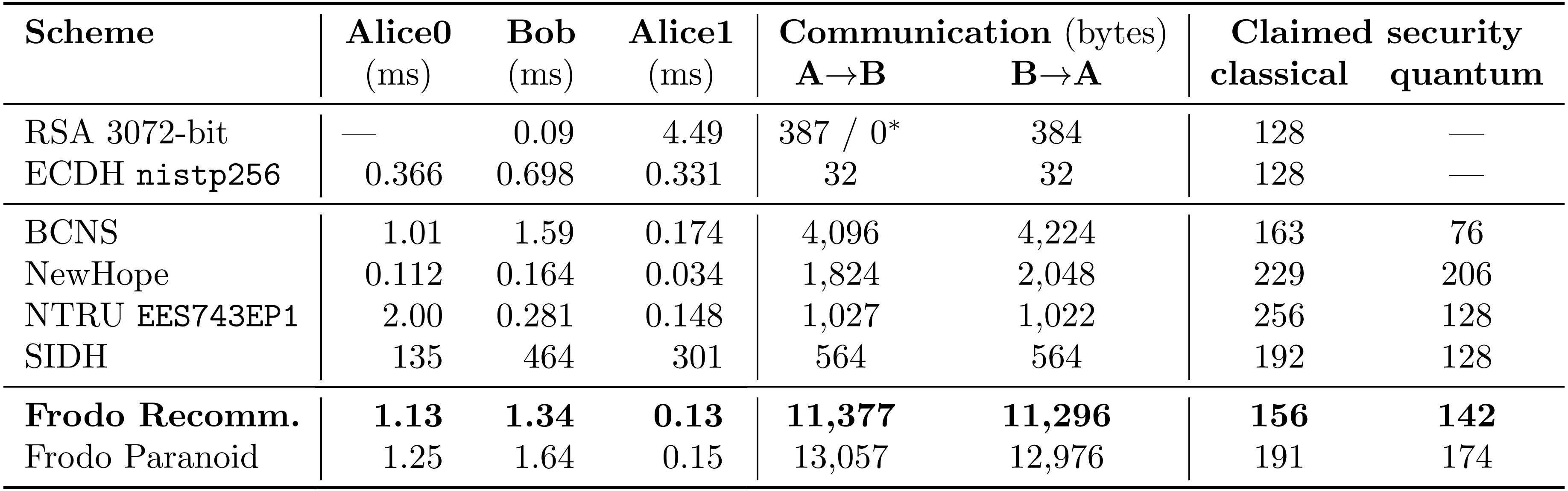
- RSA 3072-bit (OpenSSL 1.0.1f)
- ECDH `nistp256` (OpenSSL)

Use assembly code

- NewHope
- NTRU `EES743EP1`
- SIDH (Isogenies) (MSR)

Pure C implementations

Standalone performance



Scheme	Alice0	Bob	Alice1	Communication (bytes)		Claimed security	
	(ms)	(ms)	(ms)	A→B	B→A	classical	quantum
RSA 3072-bit	—	0.09	4.49	387 / 0*	384	128	—
ECDH `nistp256`	0.366	0.698	0.331	32	32	128	—
BCNS	1.01	1.59	0.174	4,096	4,224	163	76
NewHope	0.112	0.164	0.034	1,824	2,048	229	206
NTRU `EES743EP1`	2.00	0.281	0.148	1,027	1,022	256	128
SIDH	135	464	301	564	564	192	128
Frodo Recomm.	**1.13**	**1.34**	**0.13**	**11,377**	**11,296**	**156**	**142**
Frodo Paranoid	1.25	1.64	0.15	13,057	12,976	191	174

x86_64, 2.6 GHz Intel Xeon E5 (Sandy Bridge) – Google `n1-standard-4`

Note somewhat incomparable security levels

Standalone performance

RSA 3072-bit	Fast (4 ms)	Small (0.3 KiB)
ECDH `nistp256`	Very fast (0.7 ms)	Very small (0.03 KiB)
BCNS	Fast (1.5 ms)	Medium (4 KiB)
NewHope	Very fast (0.2 ms)	Medium (2 KiB)
NTRU `EES743EP1`	Fast (0.3–1.2 ms)	Medium (1 KiB)
SIDH	Very slow (400 ms)	Small (0.5 KiB)
Frodo Recommended	Fast (1.4 ms)	Large (11 KiB)
McBits*	Very fast (0.5 ms)	Very large (360 KiB)

* McBits results from source paper [BCS13] Bernstein, Chou, Schwabe. *CHES 2013.*

Note somewhat incomparable security levels

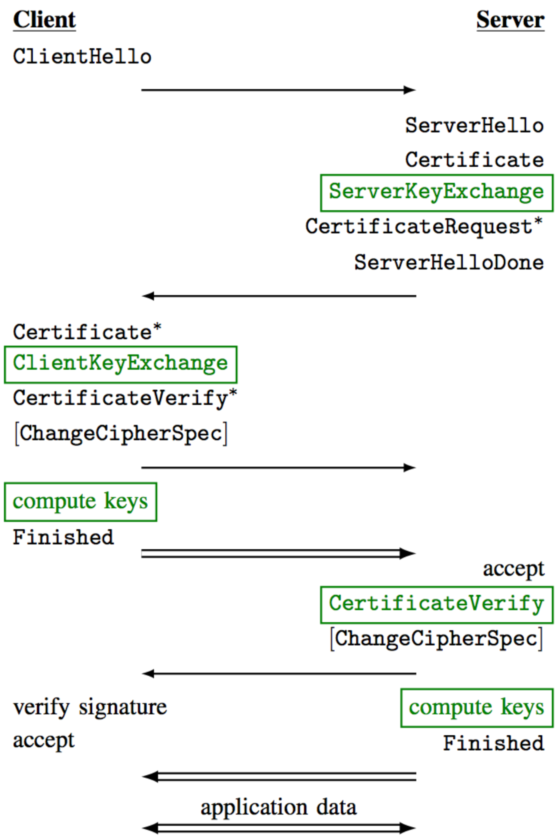
TLS INTEGRATION AND PERFORMANCE

Integration into TLS 1.2

New ciphersuite:

`TLS-KEX-SIG-AES256-GCM-SHA384`

- `SIG` = RSA or ECDSA signatures for authentication
- `KEX` = Post-quantum key exchange
- AES-256 in GCM for authenticated encryption
- SHA-384 for HMAC-KDF



Client **Server**

ClientHello

ServerHello
Certificate
ServerKeyExchange
CertificateRequest*
ServerHelloDone

Certificate*
ClientKeyExchange
CertificateVerify*
[ChangeCipherSpec]

compute keys
Finished

accept
CertificateVerify
[ChangeCipherSpec]

verify signature
accept

compute keys
Finished

application data

TLS performance

Handshake latency

- Time from when client sends first TCP packet till client receives first application data
- No load on server

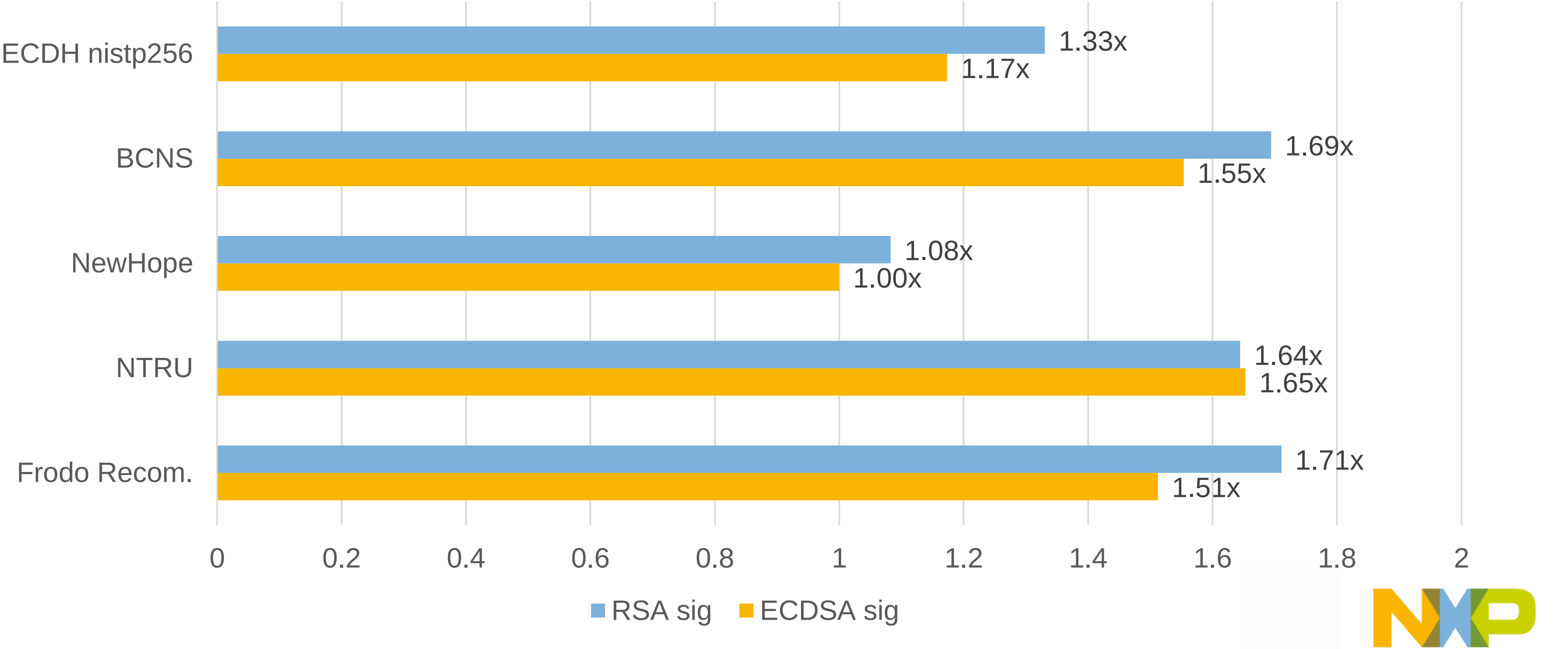
Connection throughput

- Number of connections per second at server before server latency spikes

TLS handshake latency

compared to NewHope-ECDSA

smaller (left) is better



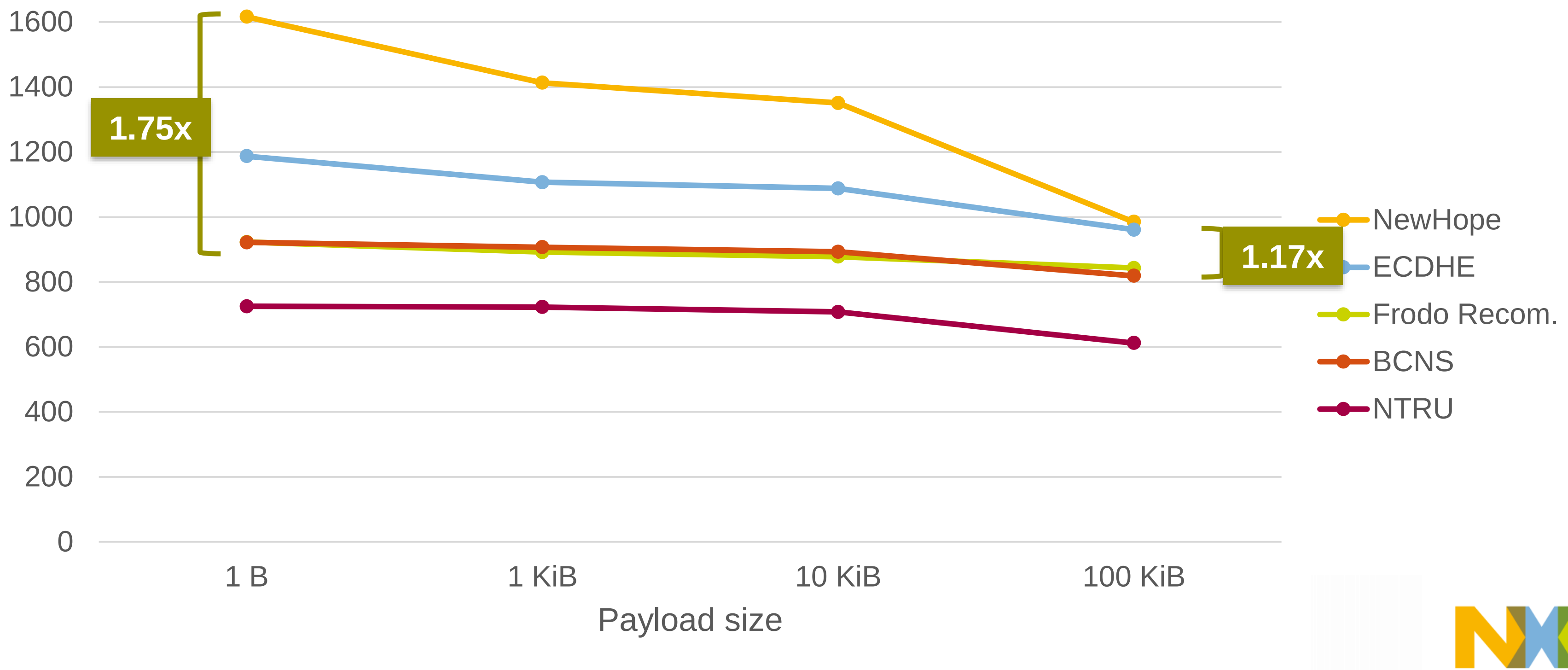
x86_64, 2.6 GHz Intel Xeon E5 (Sandy Bridge) – server Google `n1-standard-4`, client `-32`

Note somewhat incomparable security levels

TLS connection throughput

ECDSA signatures

bigger (top) is better



x86_64, 2.6 GHz Intel Xeon E5 (Sandy Bridge) – server Google `n1-standard-4`, client `-32`

Note somewhat incomparable security levels

Hybrid ciphersuites

- Use both post-quantum key exchange and traditional key exchange

- Example:
  - ECDHE + NewHope
    - Used in Google Chrome experiment
  - ECDHE + Frodo

- Session key secure if either problem is hard

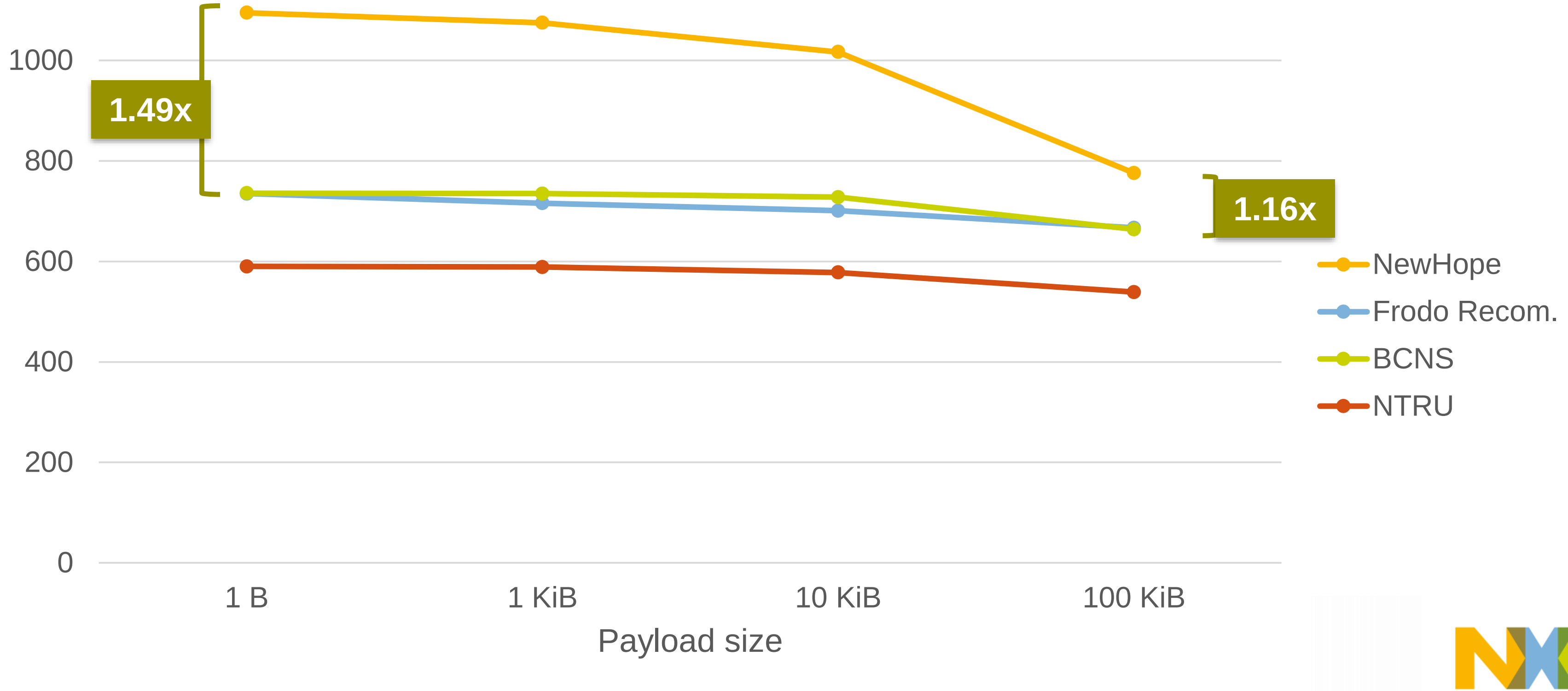
- Why use post-quantum?
  - (Potential) security against future quantum computer

- Why use ECDHE?
  - Security not lost against existing adversaries if post-quantum cryptanalysis advances

TLS connection throughput – hybrid w/ECDHE

ECDSA signatures

bigger (top) is better



x86_64, 2.6 GHz Intel Xeon E5 (Sandy Bridge) – server Google `n1-standard-4`, client `-32`

Note somewhat incomparable security levels

SUMMARY

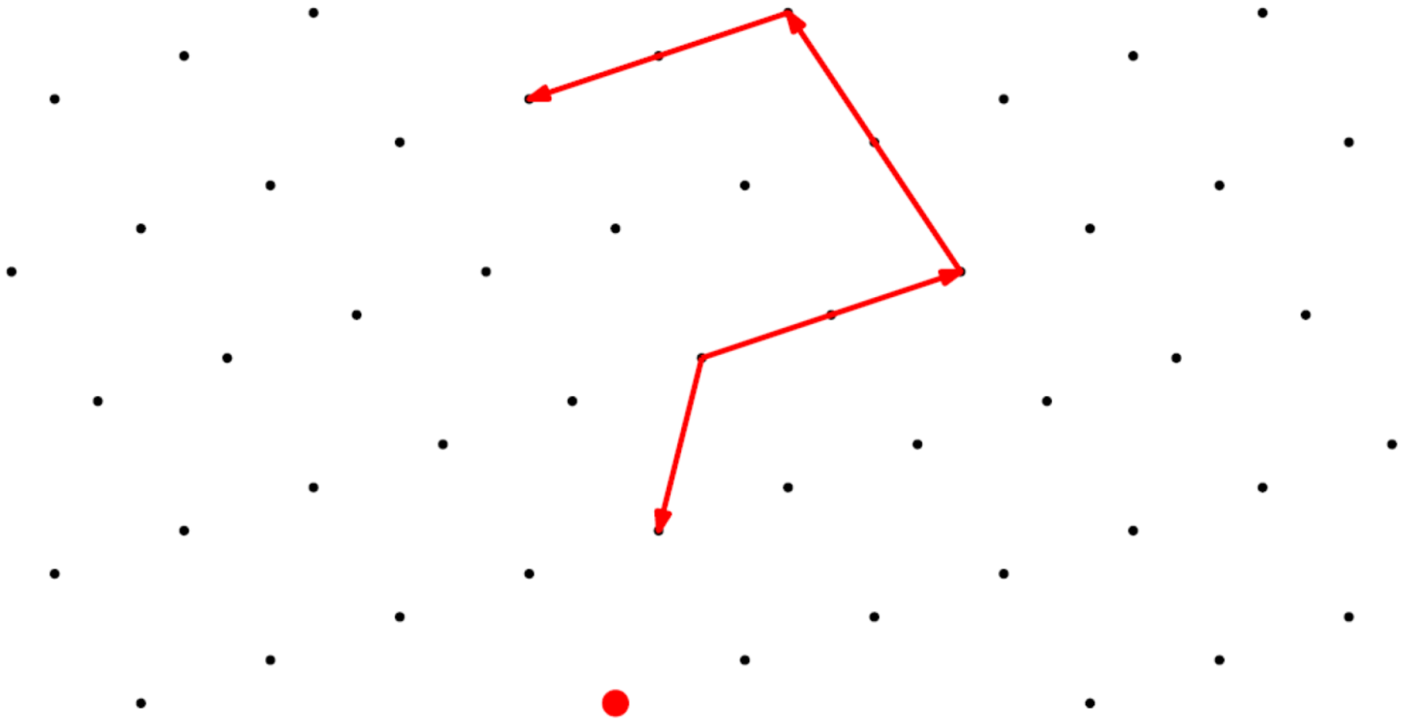
Summary

- Exciting research area – lots of opportunities!

- Ring-LWE is fast and fairly small
- LWE can achieve reasonable key sizes
- Hybrid ciphersuites will probably play a role in the transition
- Performance differences are muted in application-level protocols
- Parameter sizes and efficiency likely to evolve

- Post-quantum key exchange soon to be in demand

Questions?



NXP

SECURE CONNECTIONS
FOR A SMARTER WORLD