Post-quantum key exchange for the Internet based on lattices

Joppe Bos, **Craig Costello**, Léo Ducas, Ilya Mironov, Michael Naehrig, Valeria Nikolaenko, Ananth Raghunathan, Douglas Stebila

Part 1: Motivation

Part 2: Lattice basics

Part 3: PQ key exchange based on (R)LWE

Diffie-Hellman key exchange

$q =$

58096059953699580628595025333045743706869751763628952366614861522872037309971102257373360445331184072513261577549805174439905295945400471216628856721870324010321116397
06440498844049850989051627200244765807041812394729680540024104827976584369381522292361208779044769892743225751738076979568811309579125511333093243519553784816306381580
16186020024749256844815024251530444957718760413642873858099017255157393414625583036640591500086964373205321856683254529110790372283163413859958640669032595972518744716
90595408050123102096390117507487600170953607342349457574162729948560133086169585299583046776370191815940885283450612858638982717634572948835466388795543116154464463301
99254382340016292057090751175533888161918987295591531536698701292267685465517437915790823154844634780260102891718032495396075041899485513811126977307478969074857043710
716150121315922024556759241239013152919710956468406379442914941614357107914462567329693649

$g = 123456789$

$g^a \pmod q =$

197496648183227193286262018614250555971909799762533760654008147994875775445667054218578105133138217497206890599554928429450667899476
854668595594034093493637562451078938296960313488696178848142491351687253054602202966247046105770771577248321682117174246128321195678
537631520278649403464797353691996736993577092687178385602298873558954121056430522899619761453727082217823475746223803790014235051396
799049446508224661850168149957401474638456716624401906701394472447015052569417746372185093302535739383791980070572381421729029651639
304234361268764971707763484300668923972868709121665568669830978657804740157916611563508569886847487772676671207386096152947607114559
706340209059103703018182635521898738094546294558035569752596676346614699327742088471255741184755866117812209895514952436160199336532
6052422101474898256696660124195726100495725510022002932814218768060112310763455404567248761396399633344901857872119208518550803791724

411604662069593306683228525653441872410777999220572079993574397237156368762038378332742471939666544968793817819321495269833613169937
986164811320795616949957400518206385310292475529284550626247132930124027703140131220968771142788394846592816111078275196955258045178
705254016469773509936925361994895894163065551105161929613139219782198757542984826465893457768888915561514505048091856159412977576049
073563225572809880970058396501719665853110101308432647427786565525121328772587167842037624190143909787938665842005691911997396726455
110758448552553744288464337906540312125397571803103278271979007681841394534114315726120595749993896347981789310754194864577435905673
172970033596584445206671223874399576560291954856168126236657381519414592942037018351232440467191228145585909045861278091800166330876
4073238447199488070126873048860279221761629281961046255219584327714817248626243962413613075956770018017385724999495117779149416882188

$= g^b \pmod q$

$a =$

7147687166405; 9571879053605547396582
692405186145916522354912615715297097
100679170037904924330116019497881089
087696131592831386326210951294944584
4004974889298038584931918128447572321
023987160439062006177648318875457556
2337708539125052923646318332191217321
464134655845254917228378772756695589
845219962202945089226966507426526912
7802446416400\9025927104004338958261
1419862375878988193612187945591802864
062679\864839578139273043684955597764
13009721221824915810964579376354556\6
554629883777859568089157882151127357
4220422646379170599917677567\30420698
422392494816906777896174923072071297
603455802621072109220\54662739697748
553543758990879608882627763290293452
560094576029847\3913613887675543866
22479265299978059886472414530462194
52761811989\9746477252908878060493
17954195146382922889045577804592943
73052654\10485180264002079415193983
85114342508427311982036827478946058
7100\304977477069244278989689910572
12096357725203480402449913844583448

$g^{ab} =$

330166919524192149323761733598426244691224199958894654036331526394350099088627302979833339501183059198113987880066739
419999231378970715307039317876258453876701124543849520979430233302777503265010724513551209279573183234934359636696506
968325769489511028943698821518689496597758218540767517885836464160289471651364552490713961456608536013301649753975875
610659655755567474438180357958360226708742348175045563437075840969230826767034061119437657466993989389348289599600338
950372251336932673571743428823026014699232071116171392219599691096846714133643382745709376112500514300983651201961186
613464267685926563624589817259637248558104903657371981684417053993082671827345252841433337325420088380059232089174946
086536664984836041334031650438692639106287627157575758383128971053401037407031731509582807639509448704617983930135028
7596589383292751993079161318839043121329118930009948197899907586986108953591420279426874779423560221038468

$b =$

655456209464694; 93360682685816031704
969423104727624468251177438749706128
879957701\93698826859762790479113062
308975863428283798589097017957365590
672\83571386389571224667609499300898
554802446403039544300748002507962036
386619315229886063541005322448463915
89798641210273772558373965\486539312
854838650709031919742048649235894391
90352993032676961005\088404319792729
916038927477470940948581926791161465
02863521484987\086232861934222391717
121545686125300672760188085915004248
49476686\706784051068715397706852664
532638332403983747338379697022624261
377163163204493828299206039808703403
575100467337085017748387148822224875
309641791879395483731754620034884930
540399950519191679471224\05558557093
2193507471555777569598163700850920394
705281936392411084\43600686183528465
724969562186437214972625833222544865
996160464558\54629937016589470425264
445624157899586972652935647856967092
689604\42796501209877036845001246792
761563917639959736383038665362727158

ECDH key exchange

$$p = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$$

$p = 115792089210356248762697446949407573530086143415290314195533631308867097853951$

$$E/\mathbf{F}_p: y^2 = x^3 - 3x + b$$

$\#E = 115792089210356248762697446949407573529996955224135760342422259061068512044369$

$P = (48439561293906451759052585252797914202762949526041747995844080717082404635286,$
$36134250956749795798585127919587881956611106672985015071877198253568414405109)$

$a =$
89130644591246033577639
77064146285502314502849
28352556031837219223173
24614395

[a]$P = (84116208261315898167593067868200525612344221886333785331584793435449501658416,$
$102885655542185598026739250172885300109680266058548048621945393128043427650740)$

[b]$P = (101228882920057626679704131545407930245895491542090988999577542687271695288383,$
$77887418190304022994116595034556257760807185615679689372138134363978498341594)$

[ab]$P = (101228882920057626679704131545407930245895491542090988999577542687271695288383,$
$77887418190304022994116595034556257760807185615679689372138134363978498341594)$

$b =$
10095557463932786418806
93831619070803277191091
90584053916797810821934
05190826

Quantum computers ↔ Cryptopocalypse

- Quantum computers break elliptic curves, finite fields, factoring, everything currently used for PKC

- Aug 2015: NSA announces plans to transition to quantum-resistant algorithms

NIST
National Institute of Standards and Technology

- Feb 2016: NIST calls for quantum-secure submissions

Cryptopocalypse now?

x = how long information needs to be secure

y = how long it takes to deploy PQ crypto

z = how far away is a quantum computer

if $x + y > z$, we're screwed!

Post-quantum key exchange

Quantum-hard problem(s) for key exchange???

Codes?

Isogenies?

This talk: lattice problems

Multivariate eq's?

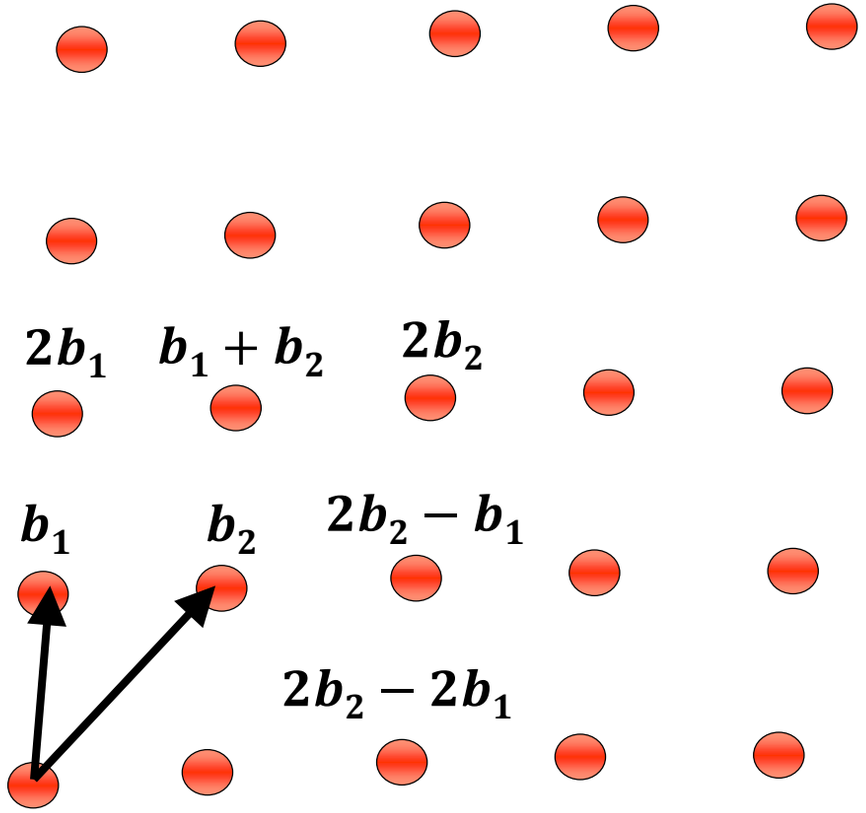
Part 1: Motivation

Part 2: Lattice basics

Part 3: PQ key exchange based on (R)LWE

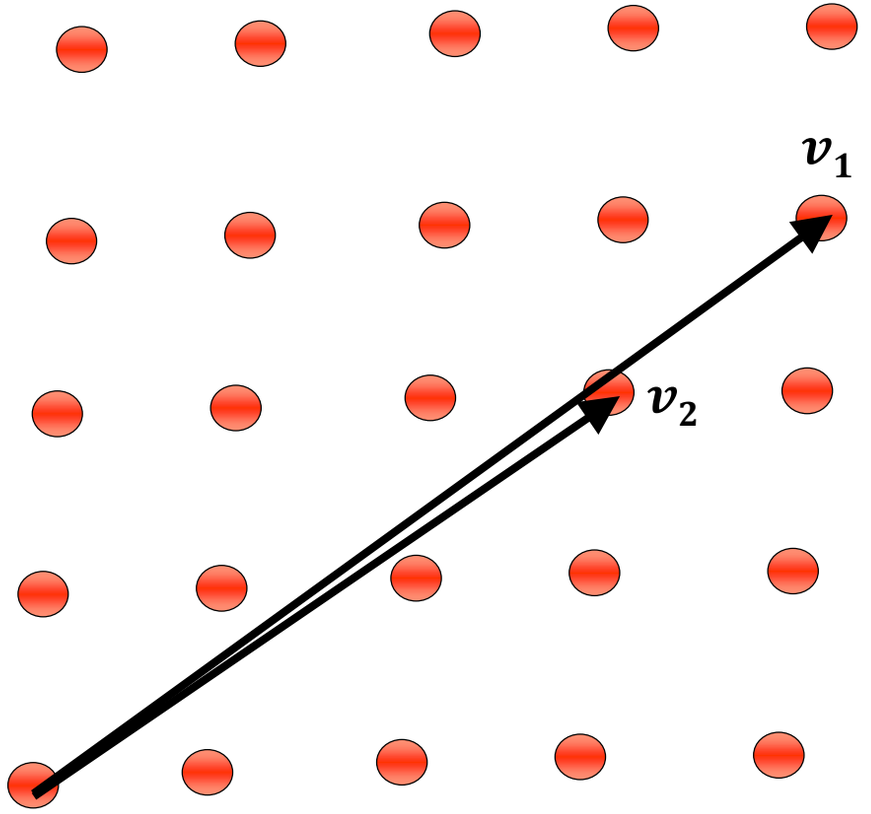
Lattices

- Basis $\boldsymbol{b_1, \ldots, b_n} \in \mathbb{R}^n$
- Lattice $\boldsymbol{L} = \{a_1\boldsymbol{b_1} + \cdots + a_n\boldsymbol{b_n} : a_i \in \mathbb{Z}\}$



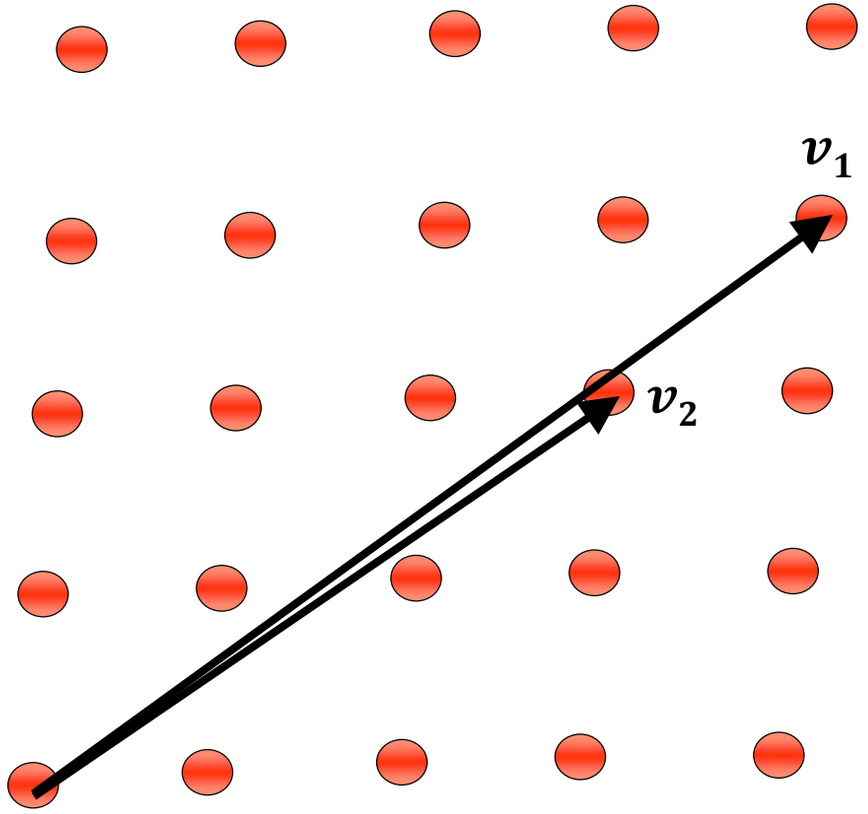
Lattices

- Basis $\boldsymbol{b_1, \ldots, b_n} \in \mathbb{R}^n$
- Lattice $\boldsymbol{L} = \{a_1\boldsymbol{b_1} + \cdots + a_n\boldsymbol{b_n} : a_i \in \mathbb{Z}\}$
- Bases not unique $\boldsymbol{L} = \sum a_i \boldsymbol{v_i}$
- e.g., $b_1 = (-2,\ 1),\ b_2 = (10,6)$
  $v_1 = (4, -3),\ v_2 = (2,4)$



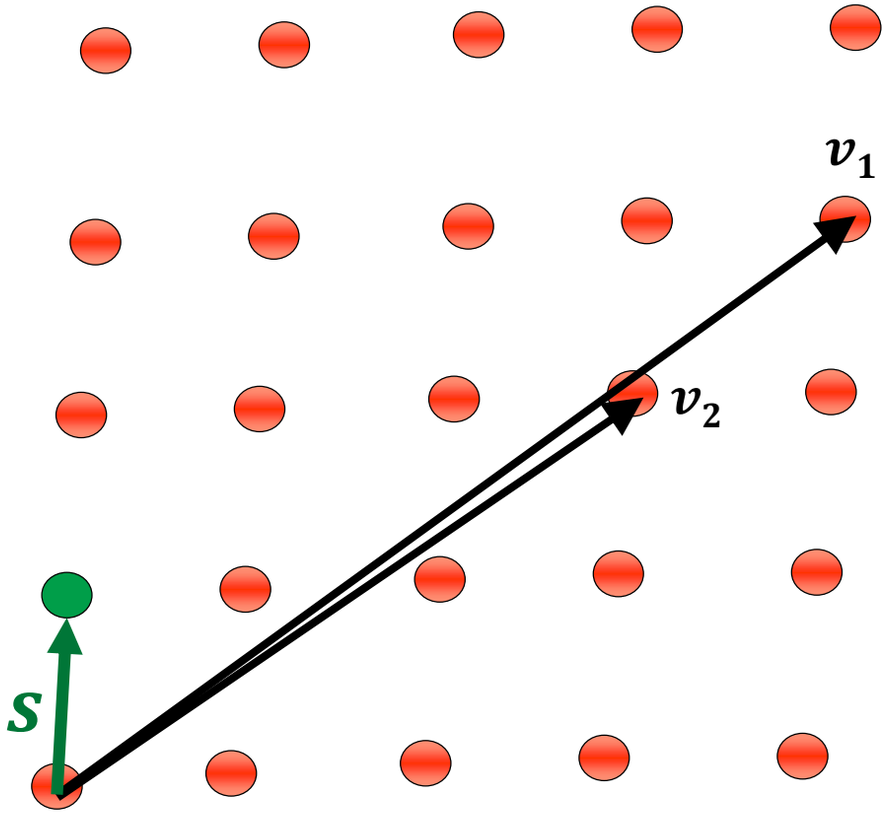
Lattices

- Basis $\boldsymbol{b_1, \ldots, b_n} \in \mathbb{R}^n$
- Lattice $\boldsymbol{L} = \{a_1\boldsymbol{b_1} + \cdots + a_n\boldsymbol{b_n} : a_i \in \mathbb{Z}\}$
- Bases not unique $\boldsymbol{L} = \sum a_i \boldsymbol{v_i}$
- e.g., $b_1 = (-2,\ 1),\ b_2 = (10,6)$
  $v_1 = (4, -3),\ v_2 = (2,4)$
- Invariant $\det(\boldsymbol{L}) = |\det(\boldsymbol{b_i})| = |\det(\boldsymbol{v_i})|$



$$\underbrace{\begin{bmatrix} -2 & 1 \\ 10 & 6 \end{bmatrix}}_{\boldsymbol{b_i}} = \underbrace{\begin{bmatrix} 4 & -3 \\ 2 & 4 \end{bmatrix}}_{\boldsymbol{v_i}} \cdot \underbrace{\begin{bmatrix} 1 & 1 \\ 2 & 1 \end{bmatrix}}_{\boldsymbol{\det = \pm 1}}$$

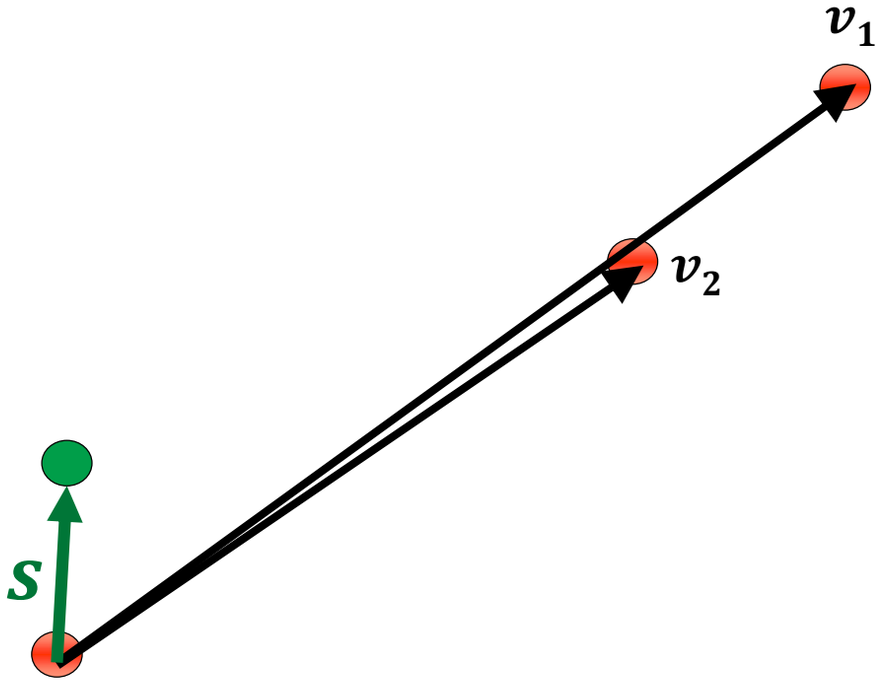
Hard Lattice Problem #1: Shortest Vector Problem (SVP_γ)



SVP: Given lattice $L = \{v_1, v_2\}$, find short vector $|s| \leq \gamma \cdot \lambda(L)$

($\gamma = 1$ means shortest vector)

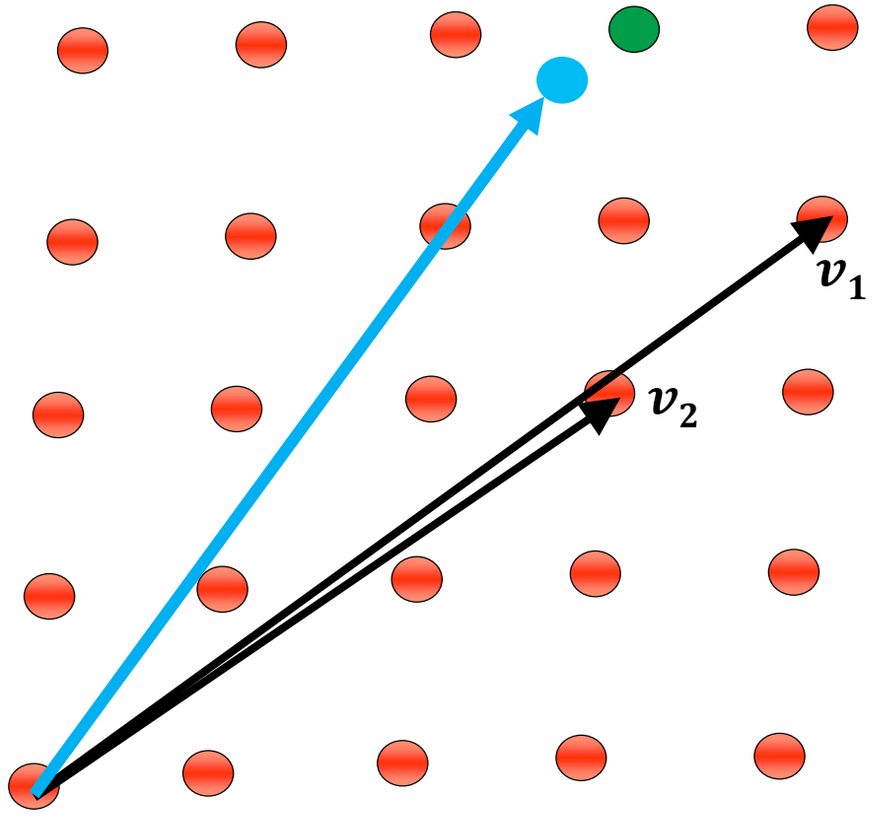
Hard Lattice Problem #1: Shortest Vector Problem (SVP_γ)



SVP_γ is NP-hard for $\gamma = O(1)$

SVP_γ is P for $\gamma = 2^{\Omega(n)}$

Hard Lattice Problem #2: Closest Vector Problem (CVP_d)



$\mathbf{CVP_d}$: Given lattice $L = \{v_1, v_2\}$ and target vector $\boldsymbol{v} \notin L$ within distance $\boldsymbol{d}$, find the closest lattice point

SVP in dimension 10

$$L = \{b_1, \dots b_{10}\}$$

$$
\begin{array}{ll}
b_1 = & (\ 7170 \quad 4881 \quad -1954 \quad 3314 \quad 3373 \quad -7930 \quad -2481 \quad 9519 \quad -9689 \quad -3270\ ) \\
 & (-3191 \quad -1872 \quad 4453 \quad 6941 \quad -5097 \quad 5545 \quad -9969 \quad 3475 \quad 1718 \quad -3284\ ) \\
 & (-1352 \quad -8990 \quad 500 \quad 3286 \quad -8972 \quad -214 \quad 2752 \quad 8083 \quad 1672 \quad 1415\ ) \\
\vdots & (-3227 \quad 2727 \quad 7734 \quad 2358 \quad -4539 \quad 3937 \quad 954 \quad -9577 \quad 8350 \quad -3447\ ) \\
 & (\ 1666 \quad 7326 \quad 2373 \quad -6856 \quad 4071 \quad 1420 \quad -3460 \quad -8335 \quad 9275 \quad 4273\ ) \\
 & (\ 3058 \quad -3064 \quad -8459 \quad 1416 \quad -2107 \quad -8603 \quad -1053 \quad -4284 \quad 272 \quad 6617\ ) \\
\vdots & (\ 8067 \quad 8868 \quad -6895 \quad -7580 \quad -1360 \quad -2532 \quad 5588 \quad -7695 \quad 7236 \quad -7663\ ) \\
 & (\ 1557 \quad -4692 \quad -4264 \quad 9292 \quad -8033 \quad 1663 \quad -1516 \quad 6894 \quad -2016 \quad -8920\ ) \\
 & (\ 1510 \quad -9994 \quad -3330 \quad 555 \quad -8660 \quad 8108 \quad -9438 \quad 3032 \quad 9518 \quad -1103\ ) \\
b_{10} = & (-3052 \quad 4834 \quad 969 \quad -8352 \quad -5097 \quad -369 \quad -8607 \quad -4815 \quad -2567 \quad -2782\ )
\end{array}
\in \mathbb{Z}^{10}
$$

Shortest vector $\lambda(L) =$ (2528 2219 -59 1440 -756 4606 -2734 148 -75 4948)

$$\lambda(L) = b_1 - 14b_2 + 2b_4 + 13b_5 - 2b_6 - 9b_7 + 15b_8 + 3b_{10}$$

Why are they hard?

- Gaussian elimination? Least-squares?

- What about Gram-Schmidt to reduce basis?

$$b_i^* \leftarrow b_i - \sum_{1 \le j \le i-1} \mu_{ij} \cdot b_j^*$$

$$\mu_{ij} = \frac{\langle b_i^*, b_j^* \rangle}{\|b_j\|^2}$$

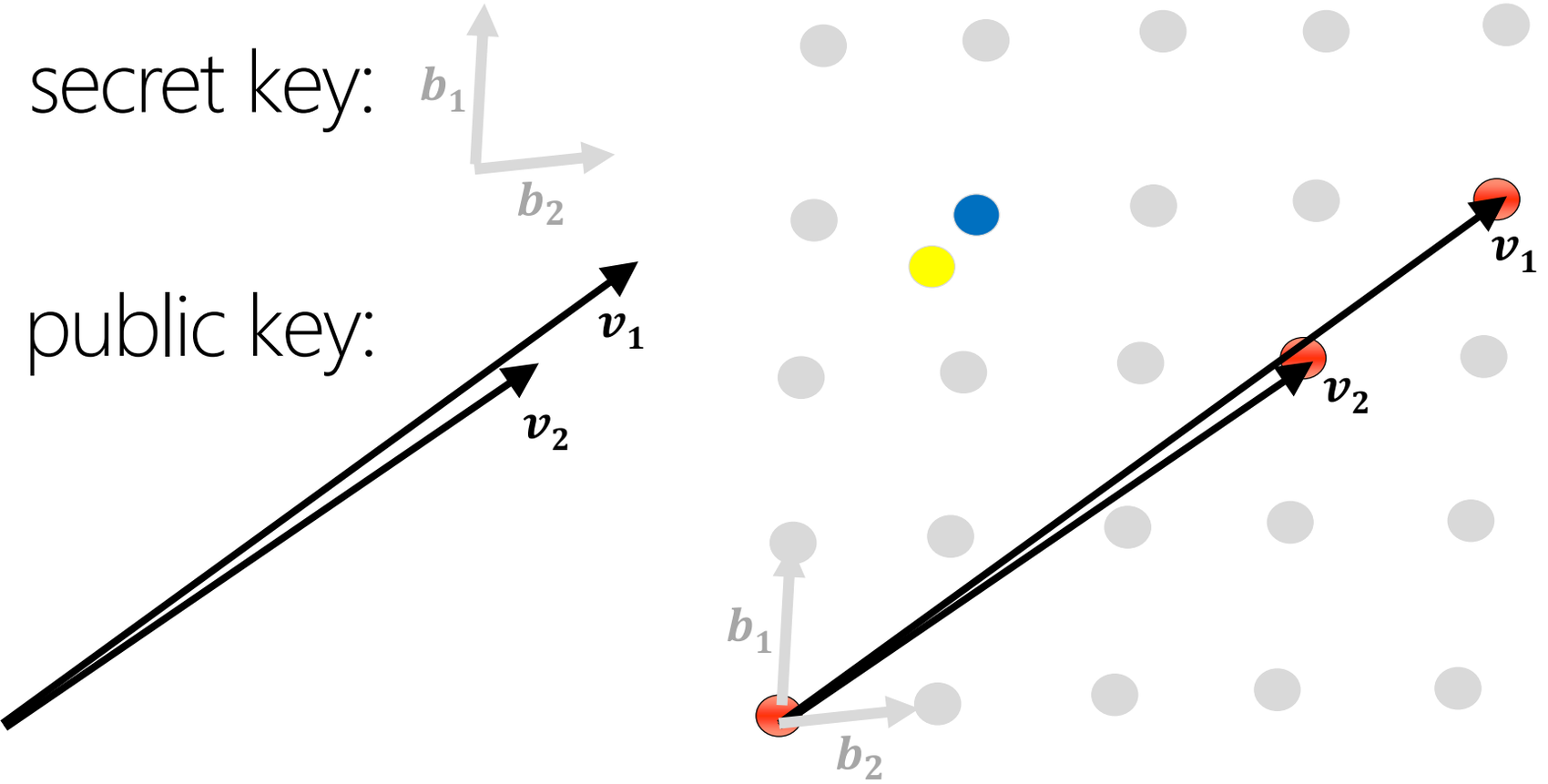
- SVP_γ NP-hard for $\gamma = O(1)$: "at least as hard as the hardest problems in NP" (if $P \neq NP$, then no polynomial time alg.)

e.g., GGH'97 signatures (≈ NTRUsign)

Idea: CVP is hard, but easy with good basis

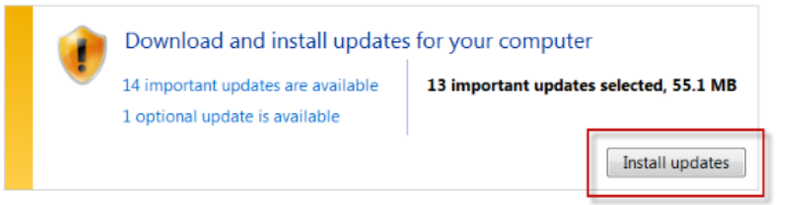
secret key: b_1 b_2

public key: v_1 v_2



message:

signature:



Security reductions

- GGH'97 (≈ right idea, but) did not come with a "security proof"
- If you can solve CVP, you can obviously forge messages, but this scheme was completely broken without solving CVP
- We want **Thm: e.g., "if you can forge signatures, you can solve CVP"**
- **Ajtai'96:** worst-to-average-case reduction unlocks lattice-based crypto "if you can break an average case, you can break the worst case"

Part 1: Motivation

Part 2: Lattice basics

Part 3: PQ key exchange based on (R)LWE

Regev'05

- Introduces the "Learning with Errors" (LWE) problem
- Uses it to construct LWE encryption
- Shows that breaking LWE implies (quantum) solving hard lattice problems (GapSVP$_{\beta}$ and SIVP)

see his 2012 talk
http://research.microsoft.com/apps/video/default.aspx?id=166559

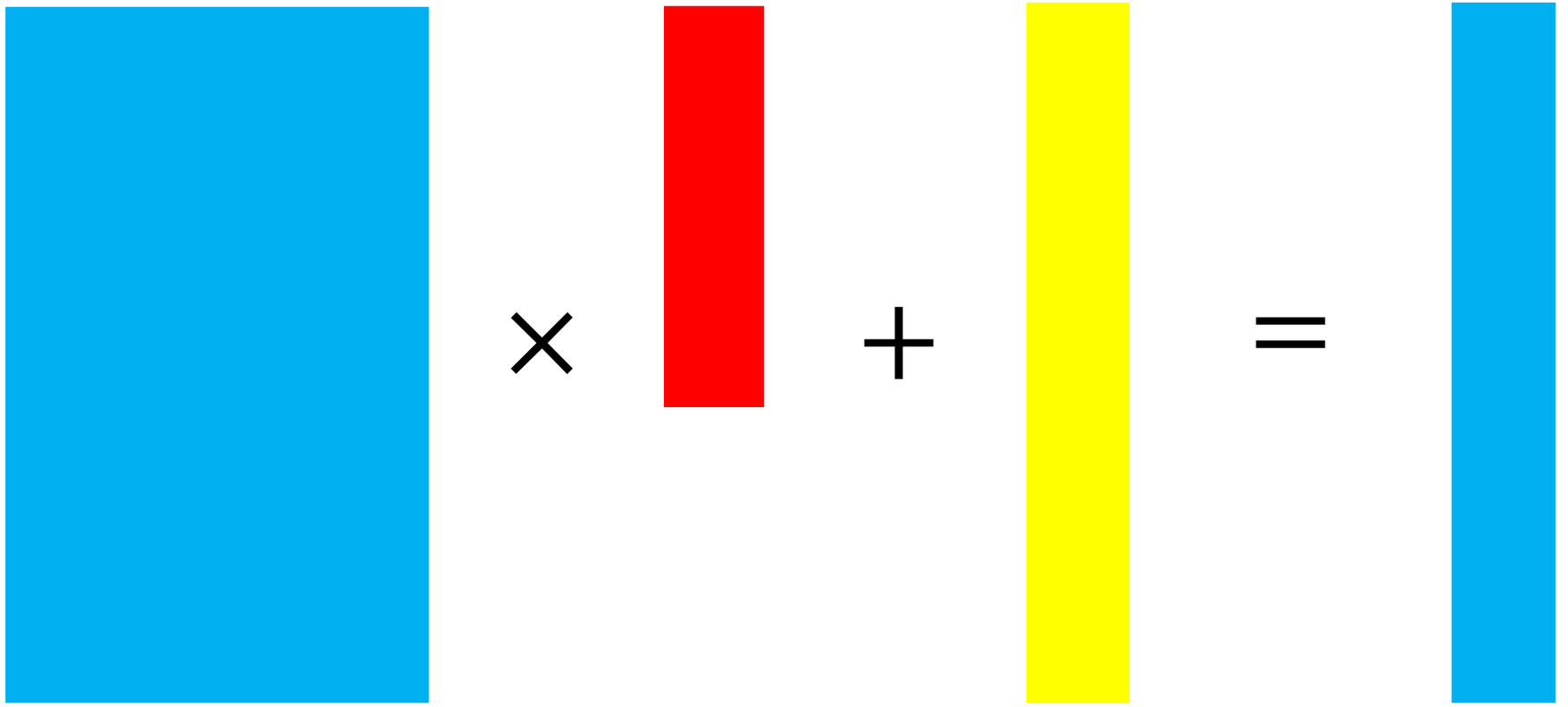
The learning with errors (LWE) problem

random $\mathbb{Z}_q^{m \times n}$

secret $\mathbb{Z}_q^{n \times 1}$

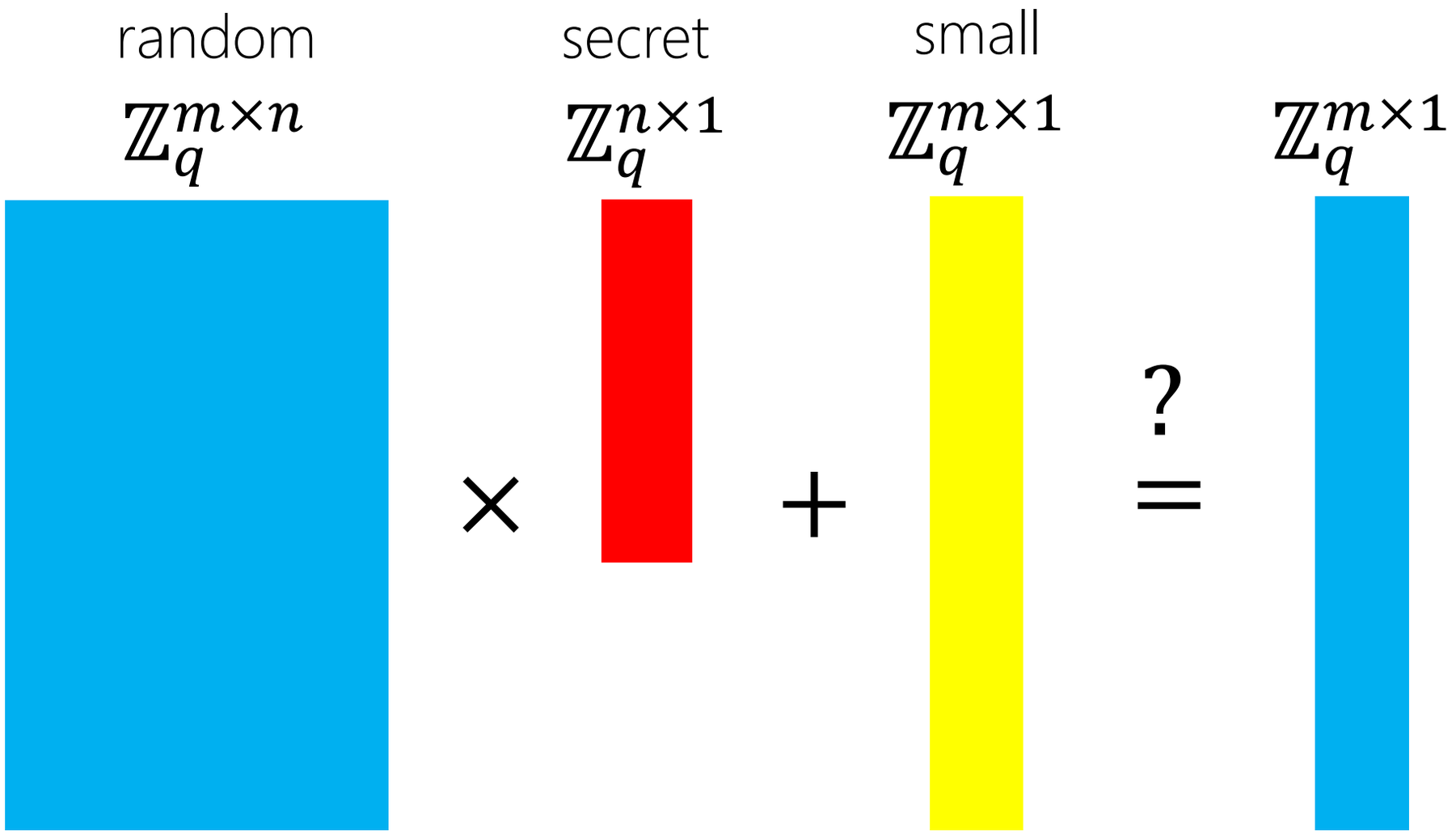
small $\mathbb{Z}_q^{m \times 1}$

$\mathbb{Z}_q^{m \times 1}$



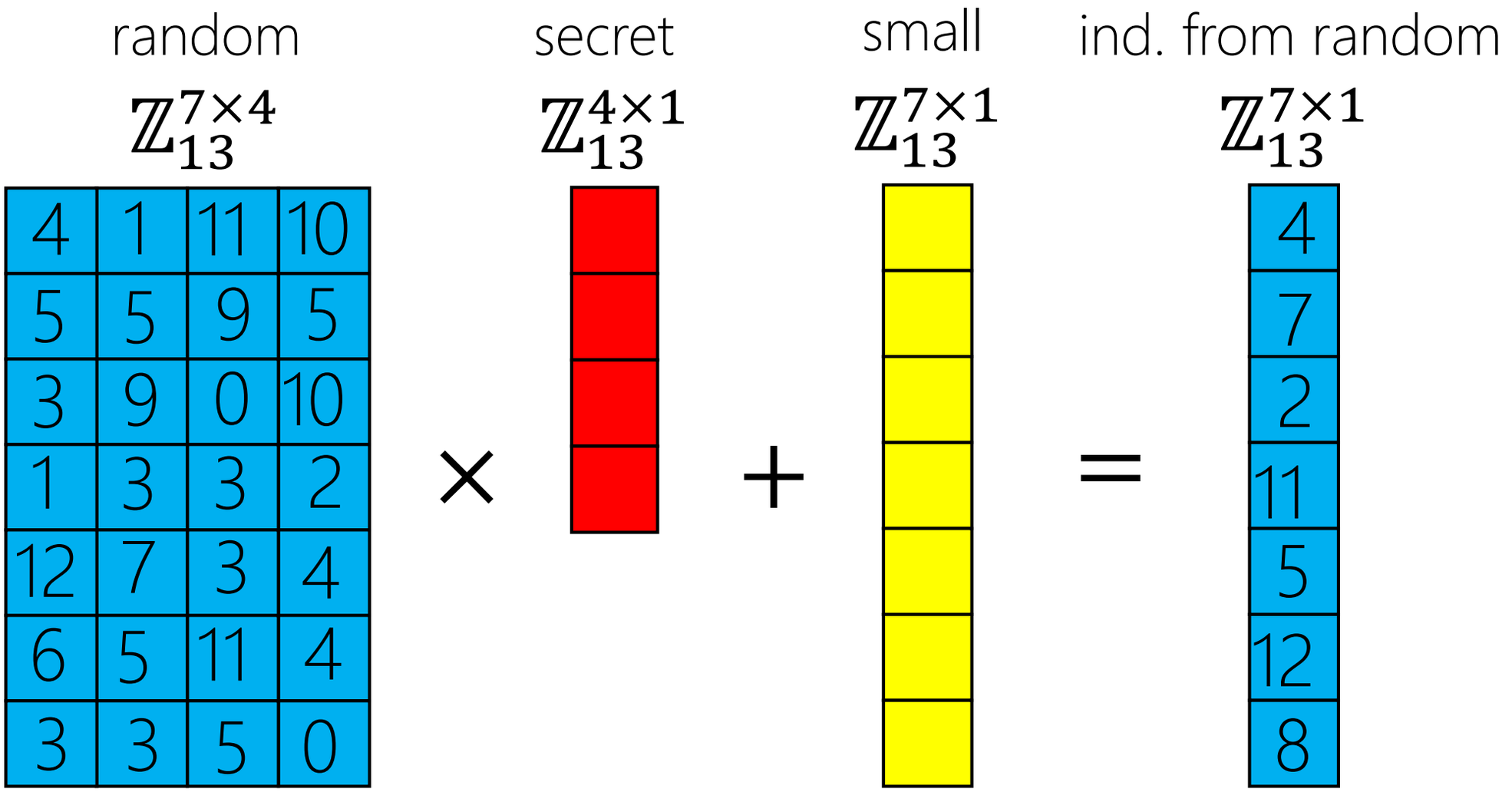
Search LWE problem: given blue, find red

The learning with errors (LWE) problem



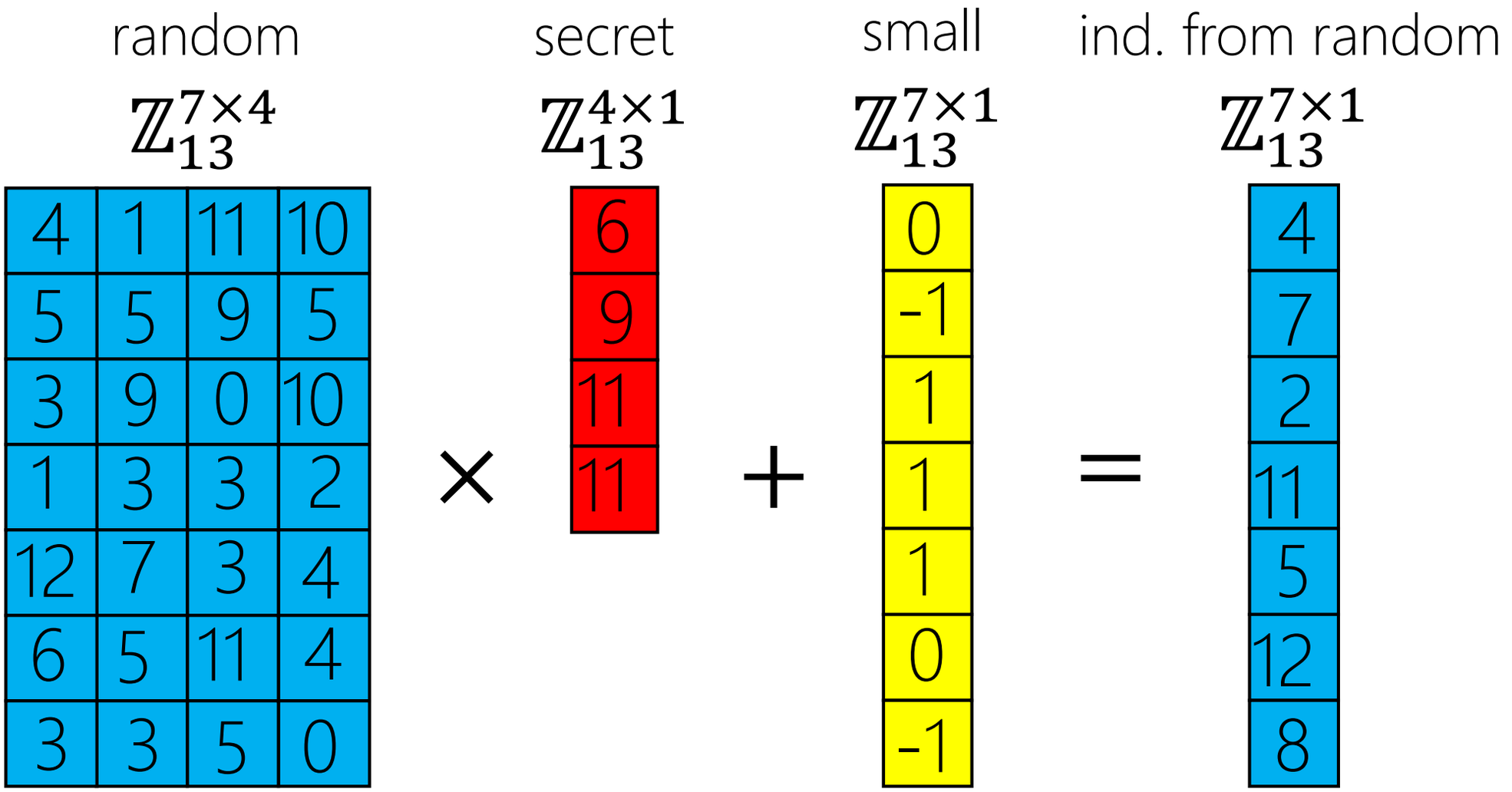
Decision LWE problem: given blue, does red exist?

The learning with errors (LWE) problem



LWE problem: given blue, find red

The learning with errors (LWE) problem



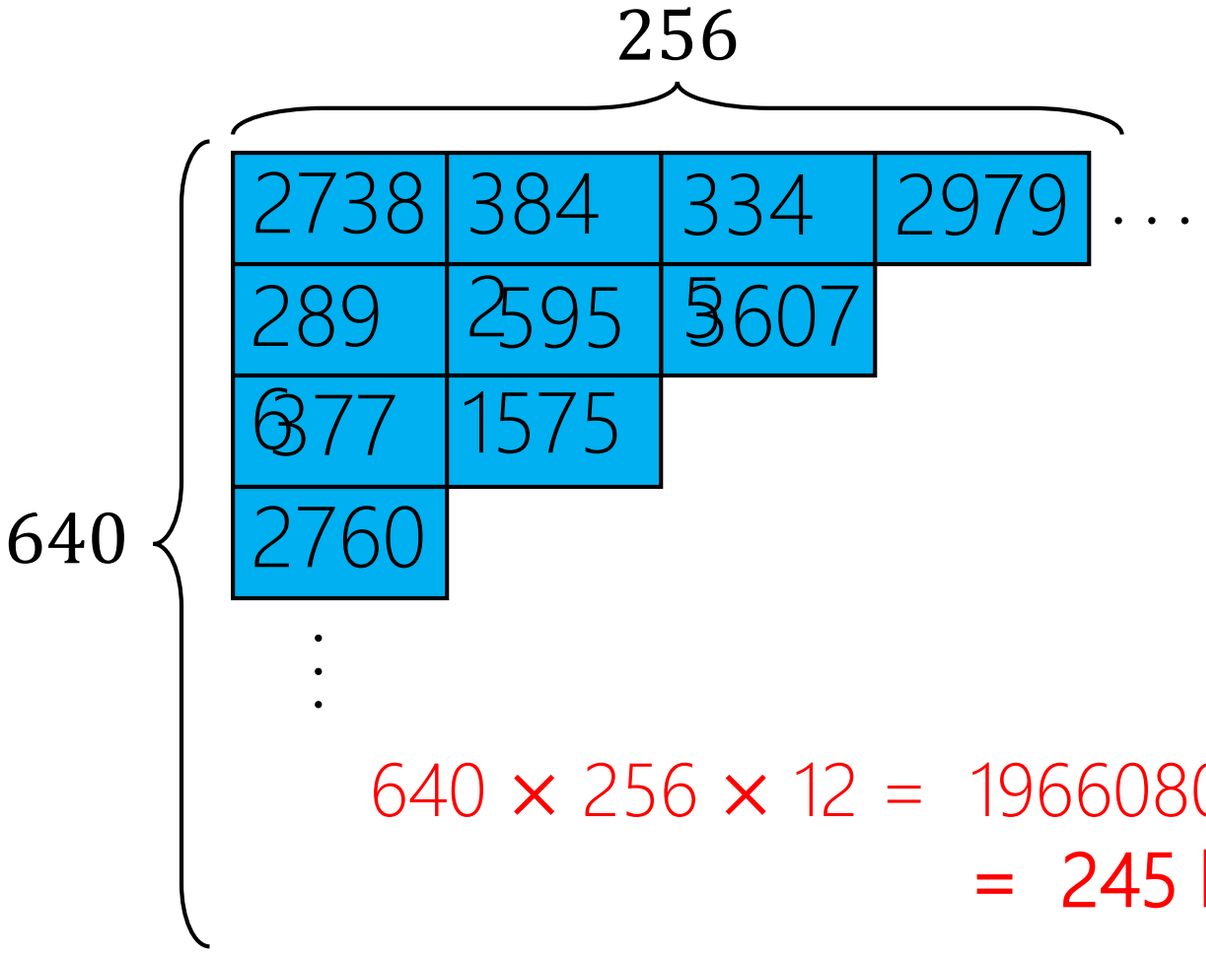
LWE problem: given blue, find red

Toy example versus real-world example

$\mathbb{Z}_{13}^{7\times4}$

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

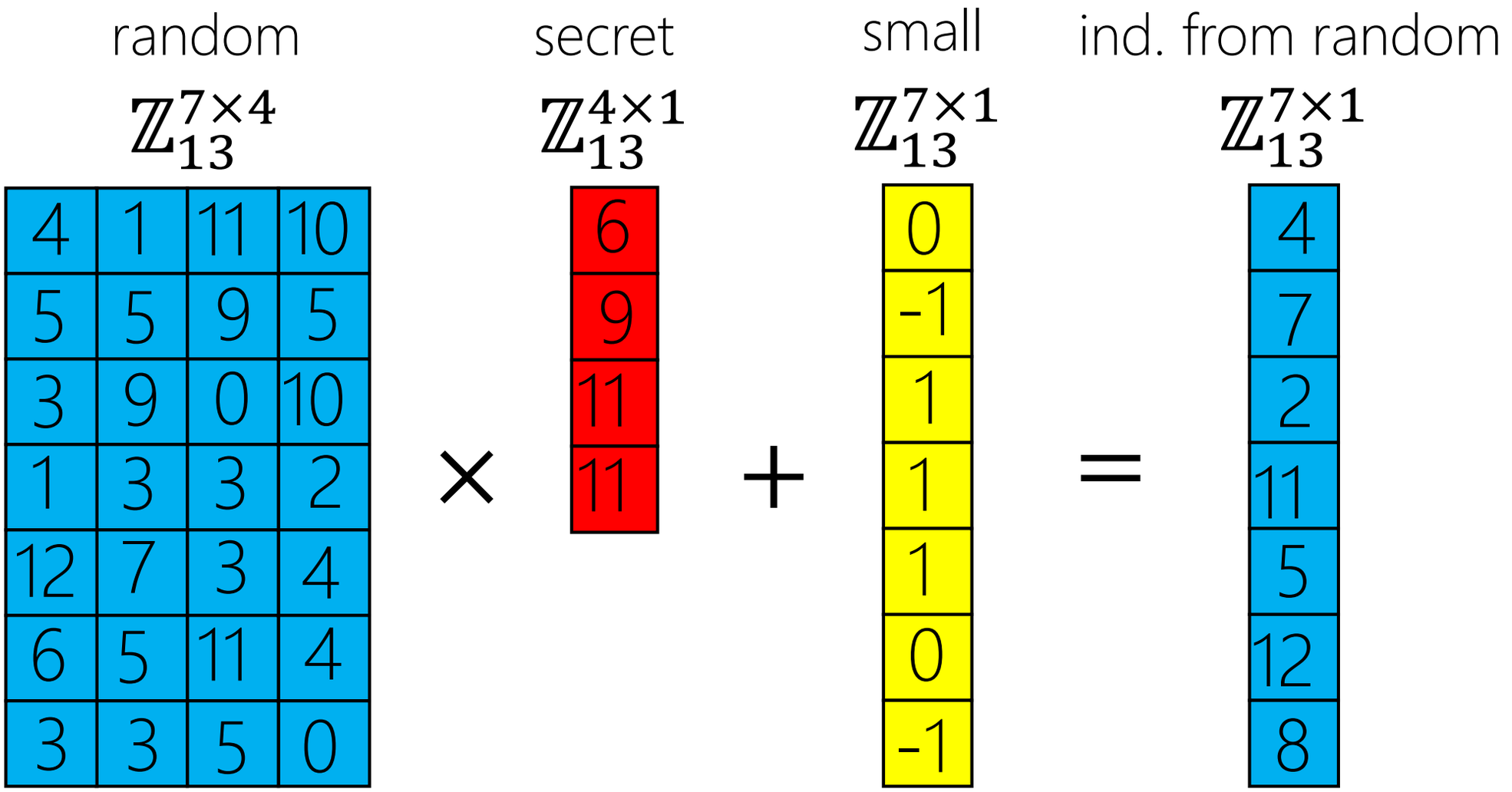
$\mathbb{Z}_{4093}^{640\times256}$



640 × 256 × 12 = 1966080 bits

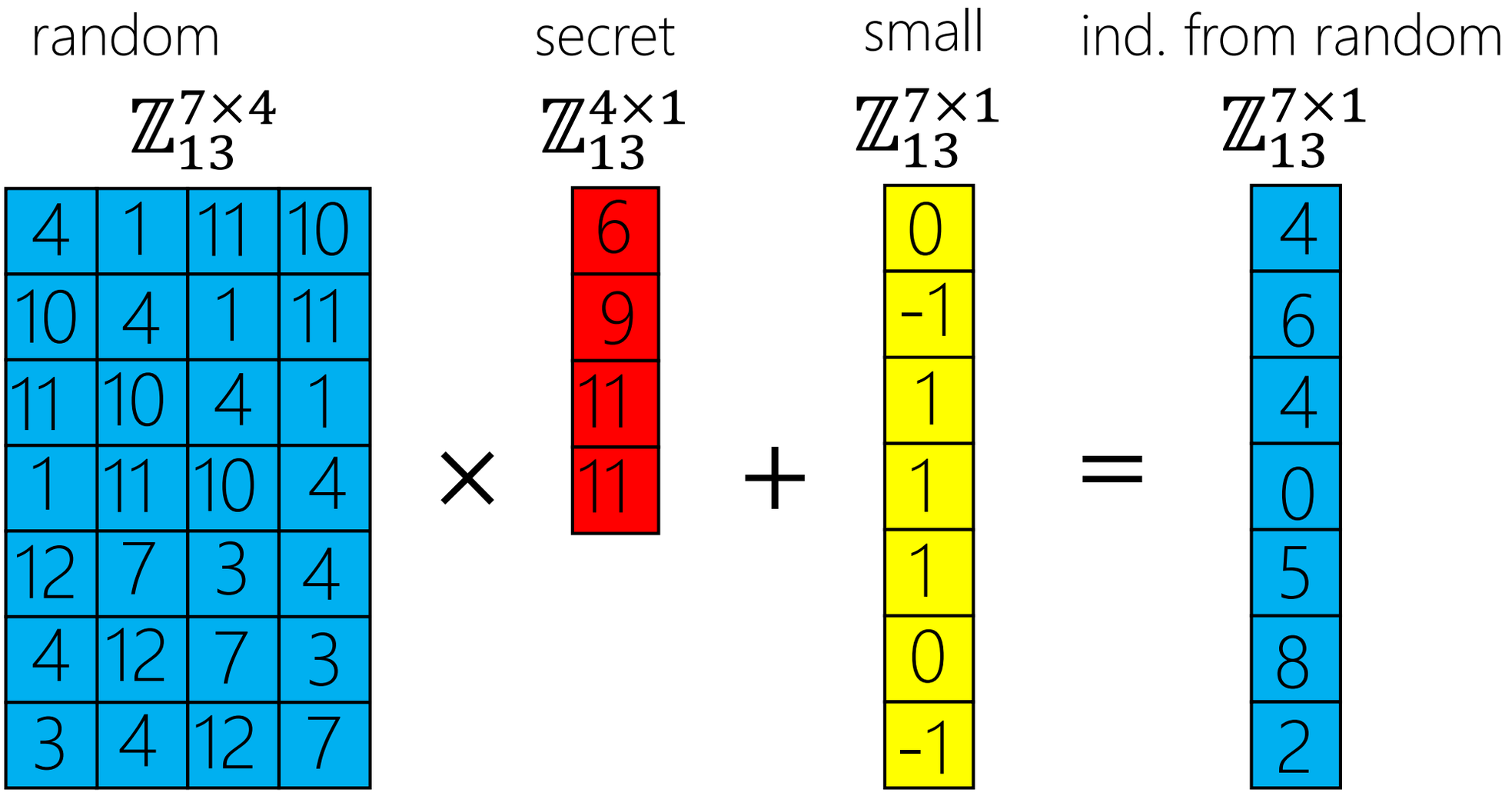
= 245 kB !!

The learning with errors (LWE) problem



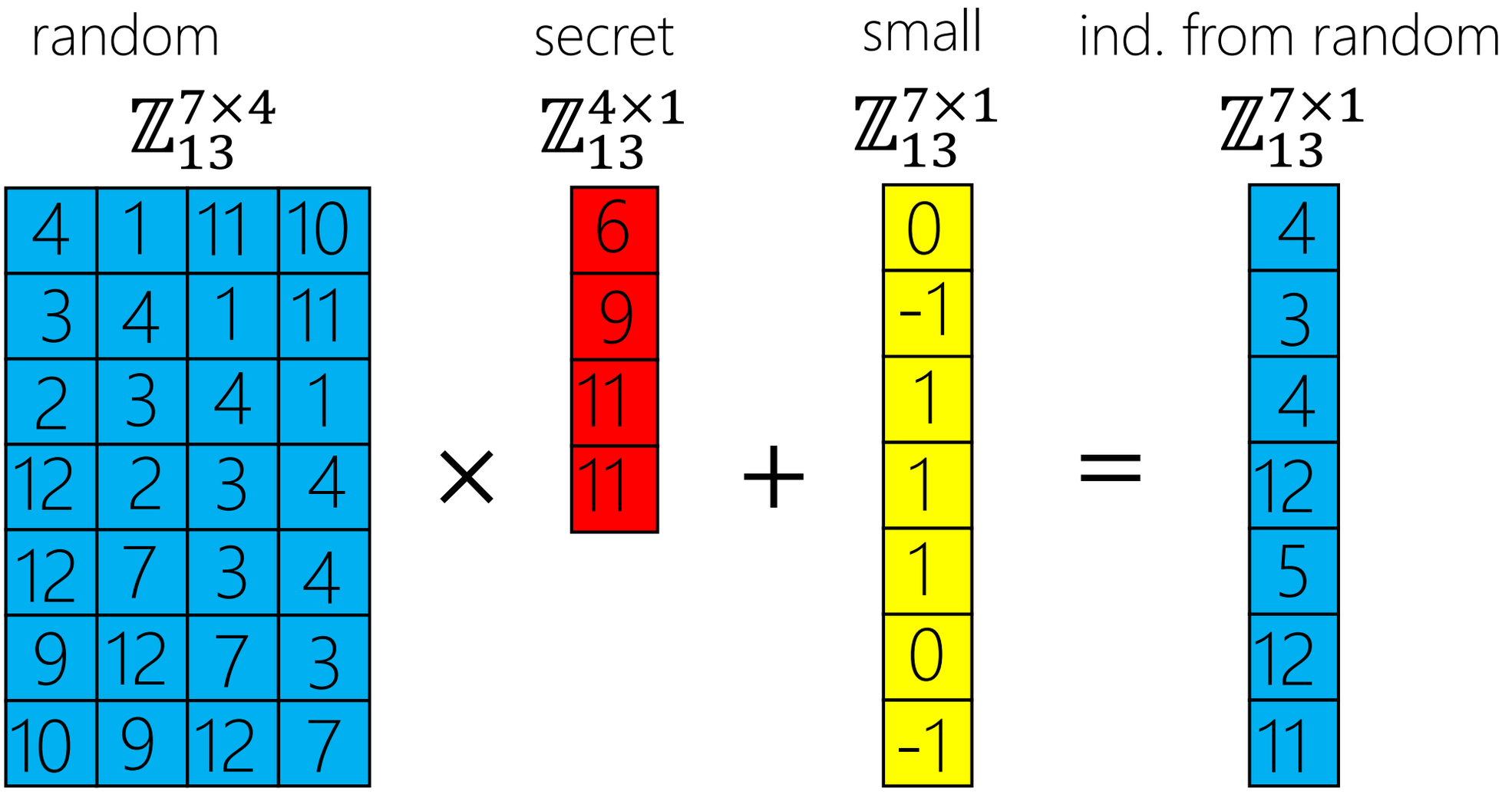
LWE problem: given blue, find red

The ring learning with errors (R-LWE) problem

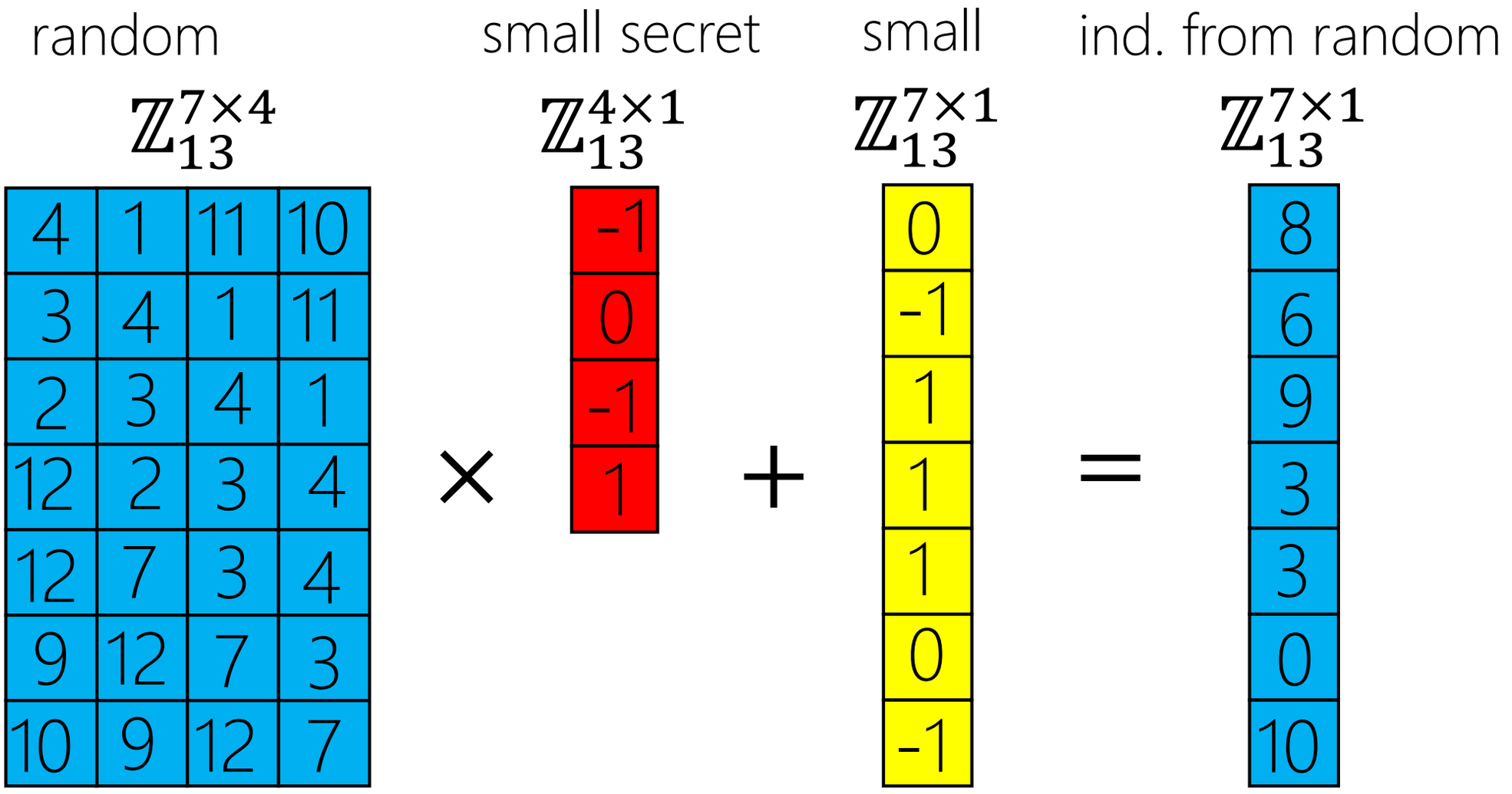


Lyubashevsky-Peikert-Regev ‘10: add ring structure

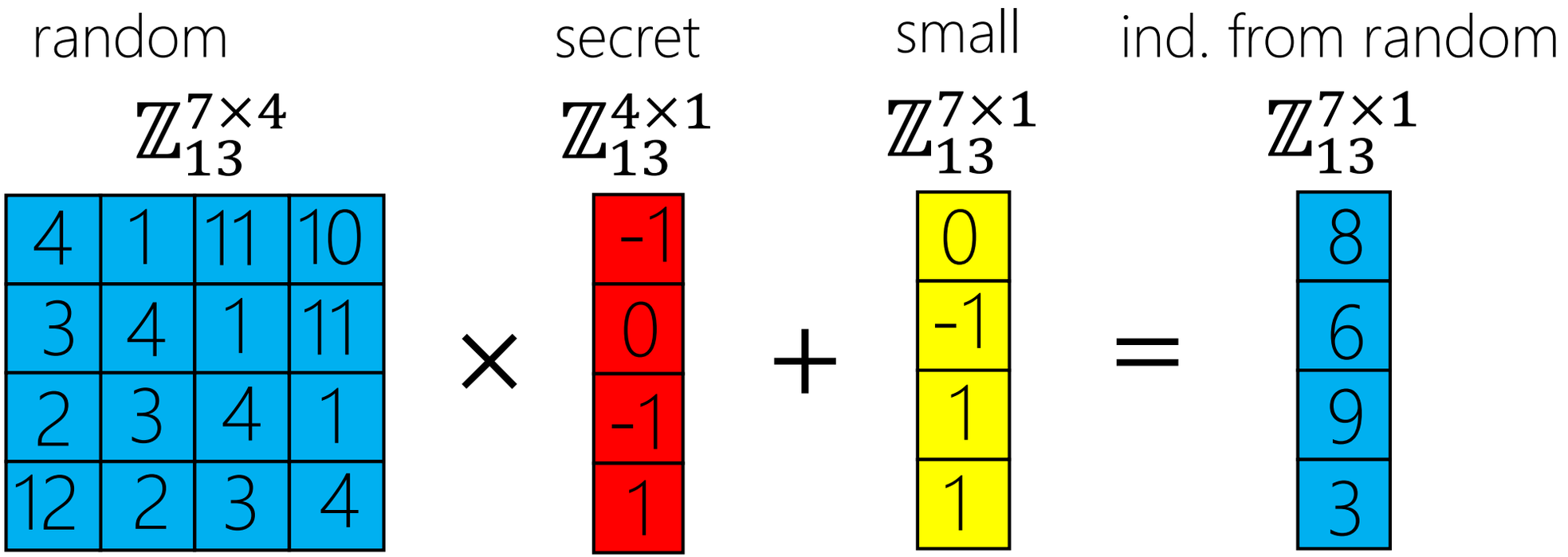
The ring learning with errors (R-LWE) problem



The ring learning with errors (R-LWE) problem



The ring learning with errors (R-LWE) problem



LWE problem: given blue, find red

The ring learning with errors (R-LWE) problem

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

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

$$\longrightarrow 4 + 1x + 11x^2 + 10x^3$$

$$\longrightarrow = x \cdot (4 + 1x + 11x^2 + 10x^3)$$

$$\longrightarrow = x^2 \cdot (4 + 1x + 11x^2 + 10x^3)$$

$$\longrightarrow = x^3 \cdot (4 + 1x + 11x^2 + 10x^3)$$

Ideal lattice: lattice modulo ideal

The ring learning with errors (R-LWE) problem

$$\begin{array}{rl} & \mathbf{4 + 1x + 11x^2 + 10x^3} \\ \times & \mathbf{-1 + 0x - 1x^2 + 1x^3} \\ + & \mathbf{0 - 1x + 1x^2 + 1x^3} \\ \hline & \mathbf{10 + 5x + 10x^2 + 7x^3} \\ \hline \end{array}$$

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

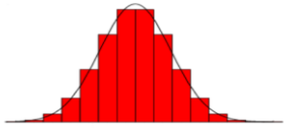
R-LWE problem: given blue, find red

The ring learning with errors (R-LWE) problem (the 128-bit secure version)

$$\frac{\mathbb{Z}_{2^{32}-1}[x]}{\langle x^{1024}+1\rangle}$$

$$2792930407 + \cdots + 2938465015x^{1023}$$

$$\times \quad 5 - 3x \ldots + 9x^{1022} - 1x^{1023}$$



$$+ \quad 2 + 4x \ldots - 0x^{1022} + 6x^{1023}$$

$$3159804584 + \cdots + 1153769078x^{1023}$$

R-LWE problem: given blue, find (small!) red

R-LWE-DH: key agreement in $R_q = \mathbb{Z}_q[x]/\langle x^n + 1 \rangle$

public: "big" a $\in R_q$

secret: "small" $e, s \in R_q$

$a \cdot s + e$

$a \cdot s' + e'$

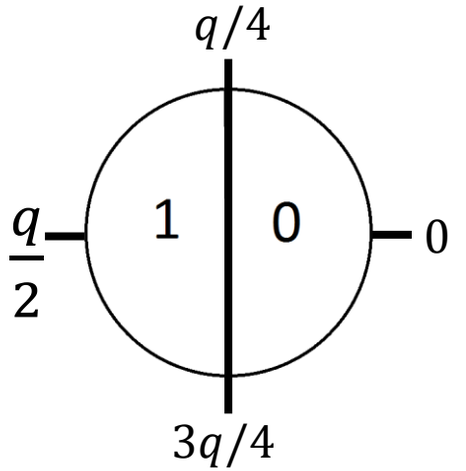
secret: "small" $e', s' \in R_q$

$\left(s \cdot (a \cdot s' + e')\right) \approx s \cdot a \cdot s'$

$\left(s' \cdot (a \cdot s + e)\right) \approx s \cdot a \cdot s'$

Approximate agreement mod q

the usual **ROUND** function



$$4079331841 + 1894732145 \cdot x + \cdots + 472608255 \cdot x^{1022} + 516748383 \cdot x^{1023}$$

$\approx \quad \approx \quad \approx \quad \approx$

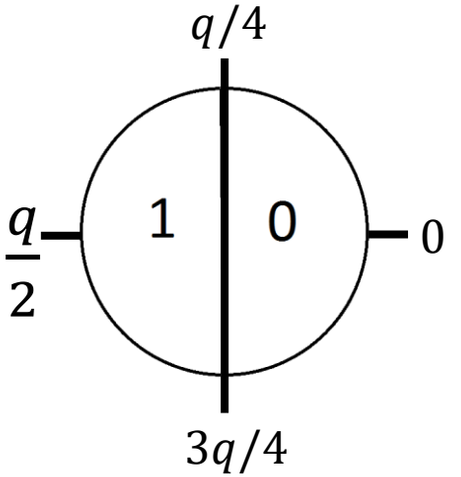
$$4079332556 + 1894733033 \cdot x + \cdots + 472607765 \cdot x^{1022} + 516748363 \cdot x^{1023}$$

$\| \quad \| \quad \| \quad \|$

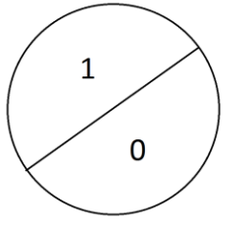
ROUND $\quad 0 \quad 1 \quad 0 \quad 0$

This will work most of the time (fails $\approx 1/2^{10}$), but we need **exact agreement**
i.e., what happens if one of the coefficients is in the "danger zone(s)"

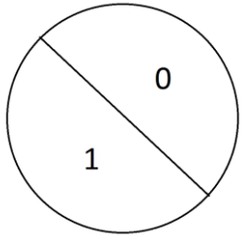
Making approximate agreement exact in $\mathbb{Z}_q$

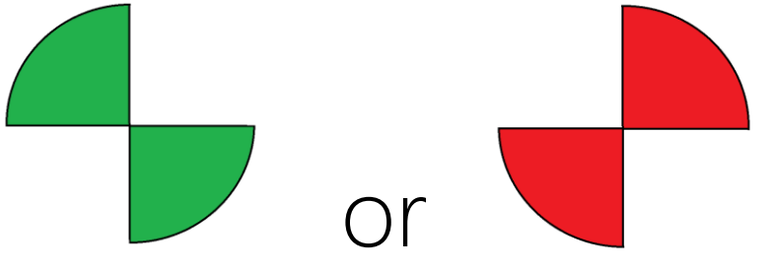


if

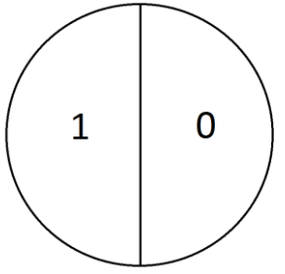


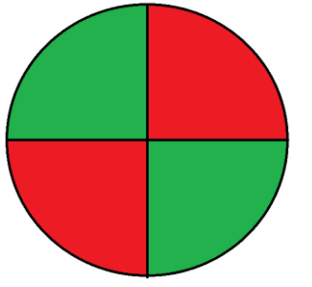
else





or





R-LWE-DH: exact key agreement

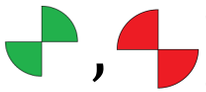
public: "big" a $\in R_q$

secret: "small" $e, s \in R_q$

$a \cdot s + e$

$a \cdot s' + e'$ and $\{$  $\}^n \in \{0,1\}^n$

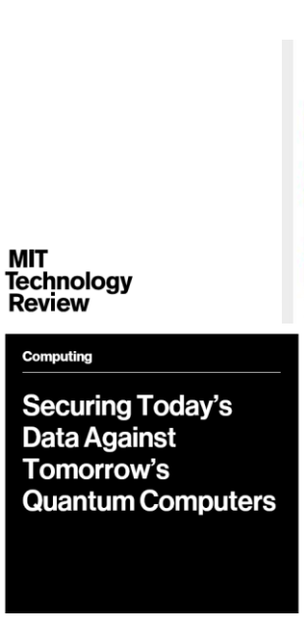
secret: "small" $e', s' \in R_q$

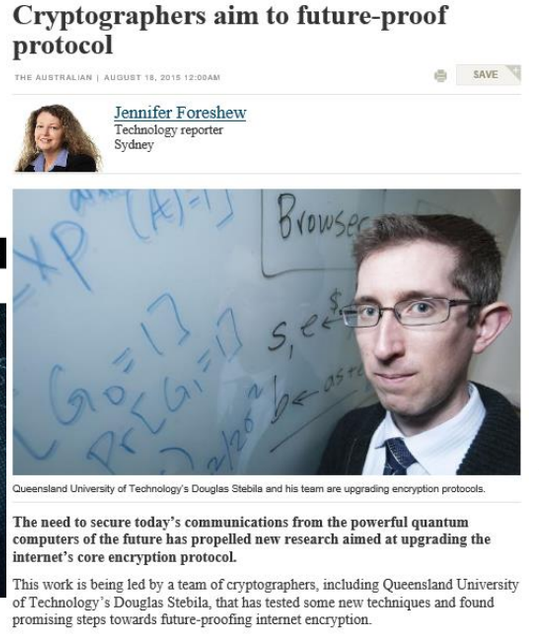
RECONCILE$(s \cdot (a \cdot s' + e'), \{$  $\}^n)$ $=$ ROUND$\big(s' \cdot (a \cdot s + e)\big)$

both parties now share $k \in \{0,1\}^n$

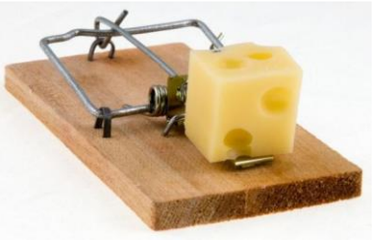
[BCNS'15]: our implementation

- Implemented ring-LWE key exchange based on Peikert'14
- Proof of security: if decision R-LWE is hard, then exact-DDH in our scheme is hard
- "Constant-time" software integrated into TLS (OpenSSL)
- Communication size: 8KiB roundtrip
- Runtime: 1.4-2.1 ms per party (TLS handshake 1.08-1.27x slower than ECDH/ECDSA)





Early bird may get the worm…
… but the second mouse gets the cheese!



- [ADPS'16]: much better implementation, error distribution, security analysis, pseudorandom parameters, etc etc
- Much faster than ours, even faster than classical (ECDH)
- PQ just means bigger keys (no slowdown)

2016 Internet Defense Prize Winner

Thomas Pöppelmann & Peter Schwabe, two co-authors of the 2016 Internet Defense Prize winning paper accept their award from Facebook at the 25th USENIX Security Symposium. Co-authors not pictured: Erdem Alkim and Léo Ducas.

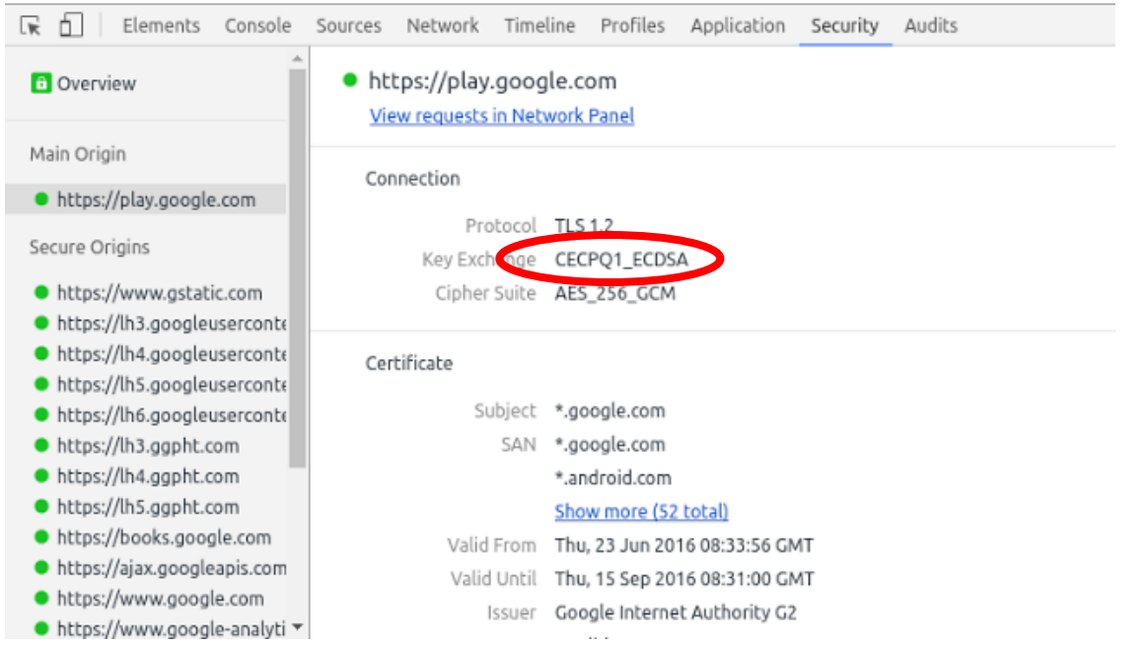
After careful consideration by our Award Committee, we decided to award the 2016 Internet Defense Prize and $100,000 to the authors of "Post-Quantum Key Exchange - A New Hope." The winning authors include: Erdem Alkim (Department of Mathemathics, Ege University, Turkey), Léo Ducas (Centrum Wiskunde & Informatica, Amsterdam, The Netherlands), Thomas Pöppelmann (Infineon Technologies AG, Munich, Germany), and Peter Schwabe (Digital Security Group, Radboud University, The Netherlands).

The authors proposed new

Google Security Blog

Experimenting with Post-Quantum Cryptography

July 7, 2016



Frodo: take off the ring!

$$\mathbb{Z}_q[x]/\langle x^n + 1\rangle$$

$$\mathbb{Z}_q^n$$

some highlights from Steven's 2016 PQcrypto keynote final slide:

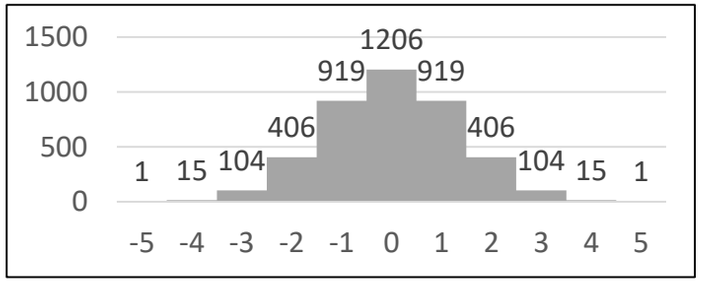
"We need to understand Ring-LWE"

"Final comment: Pqcrypto should be about greater security, not greater efficiency"

Frodo: recommended parameters

130-bit quantum
144-bit classical
103-bit plausible

$q = 2^{15} = 32768$



12 bits of randomness, Renyi div. 1.000301 to discrete Gaussian of variance 1.75

seed $\in \mathbb{Z}_2^{256}$

PRF

752 8 8

752 A x + e

c $\in \mathbb{Z}_q^{752\times 8}$ (& seed)

c′ $\in \mathbb{Z}_q^{8\times 752}$ rec $\in \mathbb{Z}_2^{64}$

seed

PRF

752 752 752

8 x′ A + e′ 8

c′ x = 8 K 8

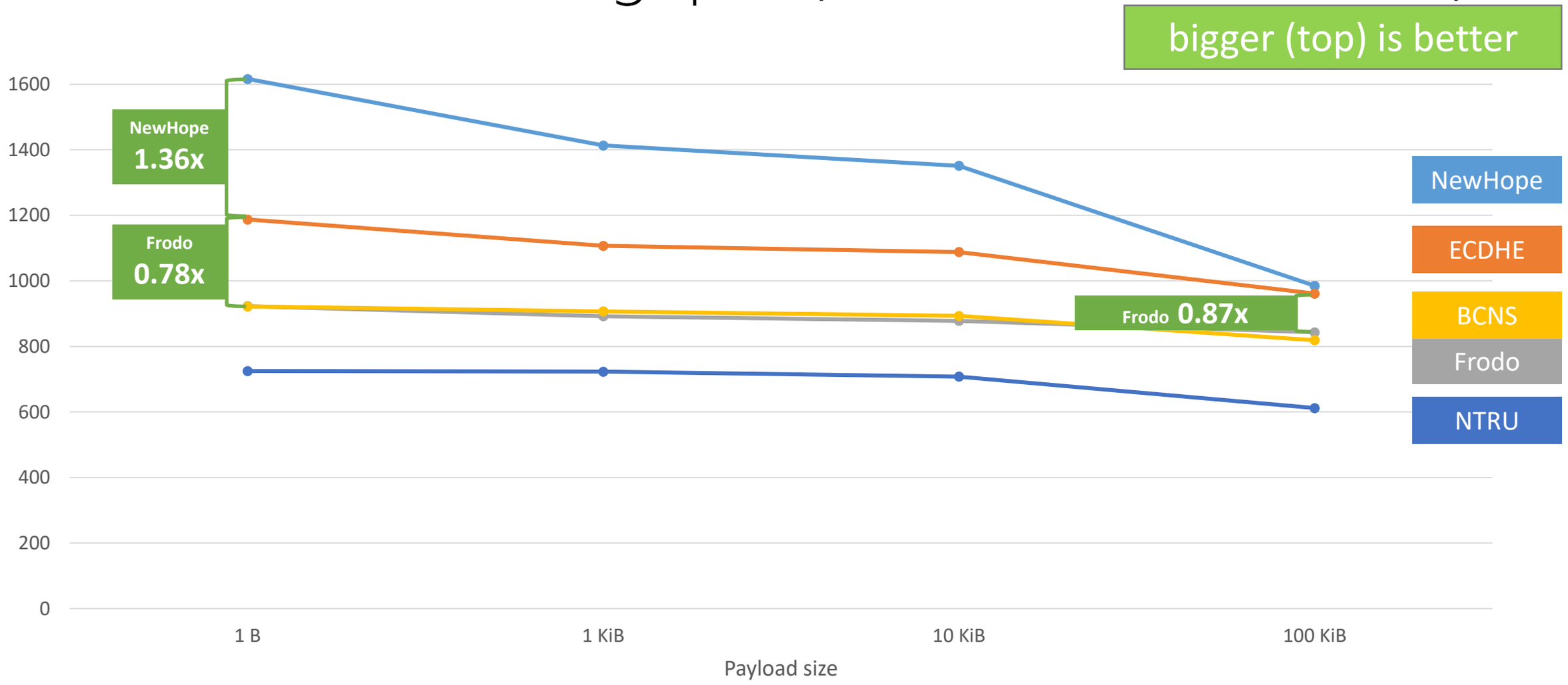
8 K′ = x′ c

K's indistinguishable from random if decision-LWE is hard!

Standalone performance of PQ primitives

	Speed		Communication		Quantum Security
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	80-bit
NewHope	Very fast	0.2 ms	Medium	2 KiB	206-bit
NTRU `EES743EP1`	Fast	0.3–1.2 ms	Medium	1 KiB	128-bit
SIDH	Very slow	35–400 ms	Small	0.5 KiB	128-bit
Frodo Recommended	Fast	1.4 ms	Large	11 KiB	130-bit
McBits*	Very fast	0.5 ms	Very large	360 KiB	161-bit

TLS connection throughput (#connections/second)



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

note somewhat incomparable security levels

References

[GGH97] O. Goldreich, S. Goldwasser, S. Halevi. Public-key cryptosystems from lattice reduction problems. CRYPTO 1997: 112-131.

[Regev05] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. STOC 2005: 84-93.

[LPR10] V. Lyubashevsky, C. Peikert, O. Regev. On ideal lattices and learning with errors over rings. EUROCRYPT 2010: 1-23.

[Peikert14] C. Peikert. Lattice Cryptography for the Internet. PQCrypto 2014: 197-219.

[BCNS15] J. Bos, C. Costello, M. Naehrig, D. Stebila. Post-quantum key exchange for the TLS protocol from the ring learning with errors problem. IEEE S&P 2015: 553-570.

[ADPS16] E. Alkim, L. Ducas, T. Poppelmann, P. Schwabe. Post-quantum key exchange – a new hope. USENIX 2016: 327-343.

[BCD+16] J. Bos, C. Costello, L. Ducas, I. Mironov, M. Naehrig, V. Nikolaenko, A. Raghunathan. Frodo: Take off the ring! Practical, quantum-secure key exchange from LWE. ACM CCS 2016: 1006-1018.

Questions?