Post-quantum cryptography

NIST standardization and beyond

Public Key Infrastructure and its Applications (PKIA 2024)

6th September, 2024

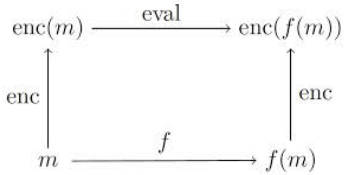
Dr. Ir. Angshuman Karmakar

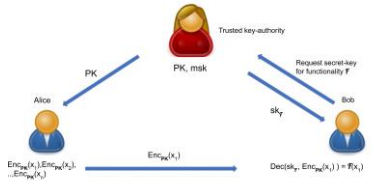
Assistant professor, Dept of CSE

IIT Kanpur, India

Asymmetric/public-key cryptography

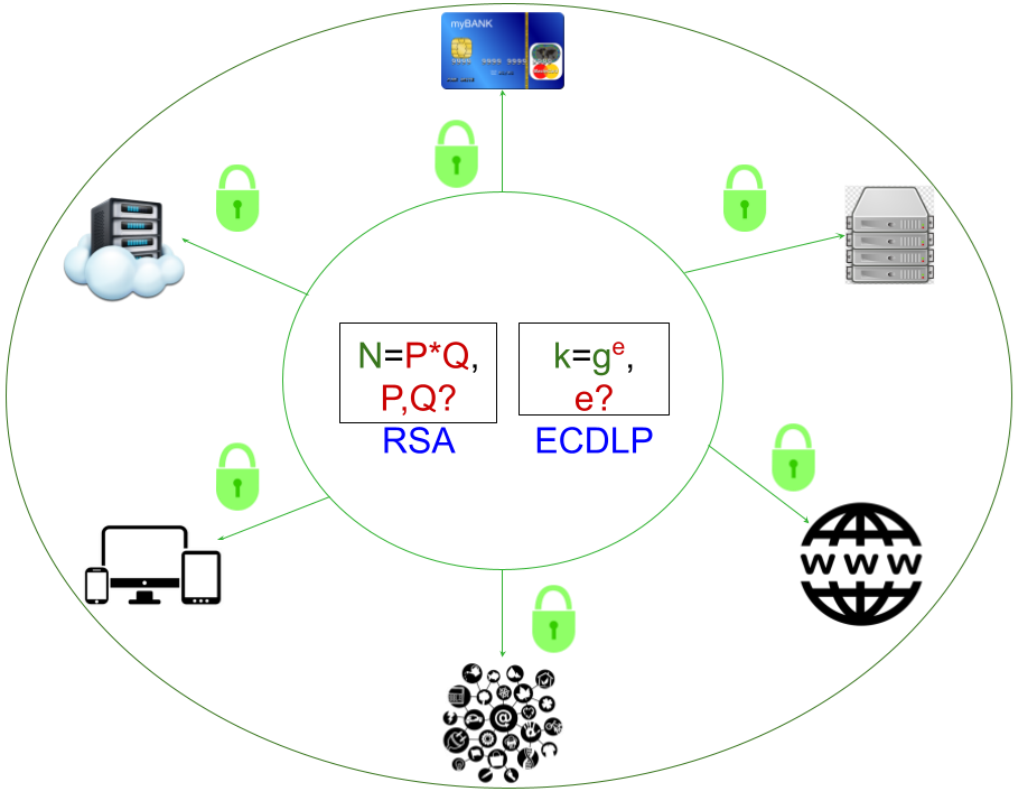
- An indispensable & crucial component for security in the digital sphere





- Security depends on *hardness* of computational problems
  - Two most widely used cryptosystems are RSA and ECC
  - Depends on hardness of Integer factorization & ECDLP

Public-key cryptography

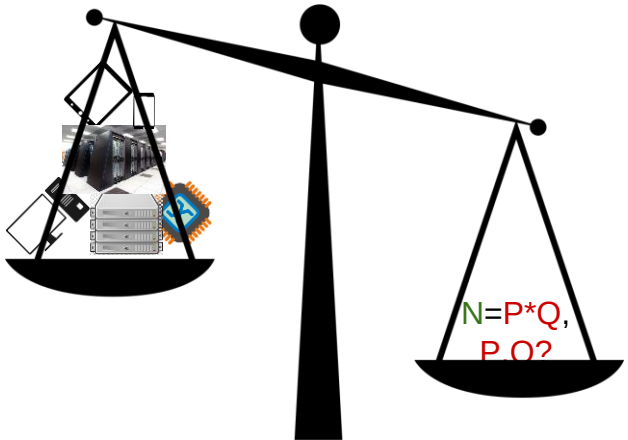


Current Public-key cryptography

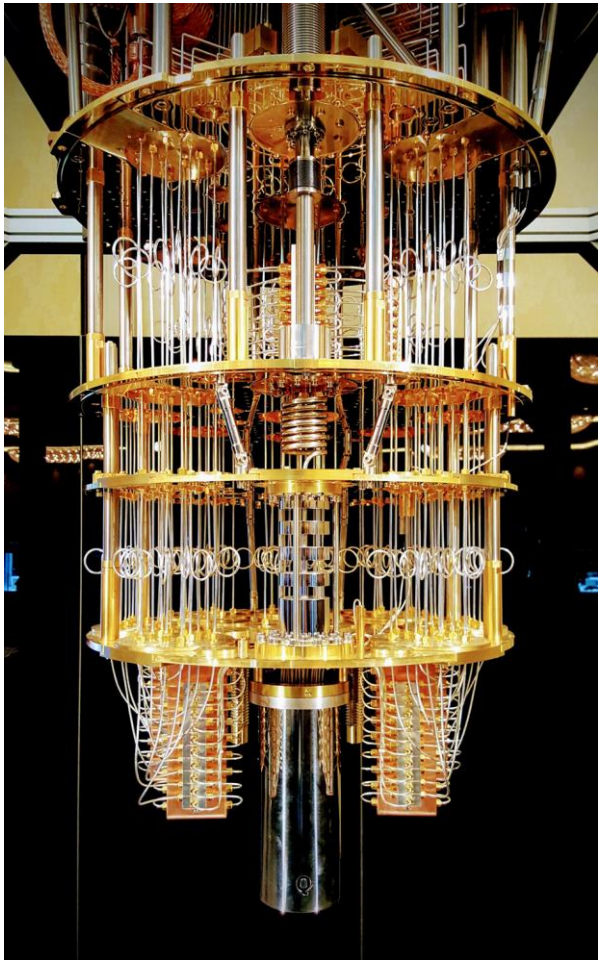
- Majority of our current public-key infrastructure is built using RSA and ECC

Hardness of public-key cryptography

- RSA/factorization : given N=P*Q, P and Q are large prime numbers
  - Find P,Q?
- ECC/discrete log: given $k=g^e$, g is generator of elliptic curve group
  - Find e?
- Very hard to solve
  - Even if we combine all the computers in the world
  - Time to solve one practical instance > the age of universe



Quantum computer



*IBM Q quantum computer

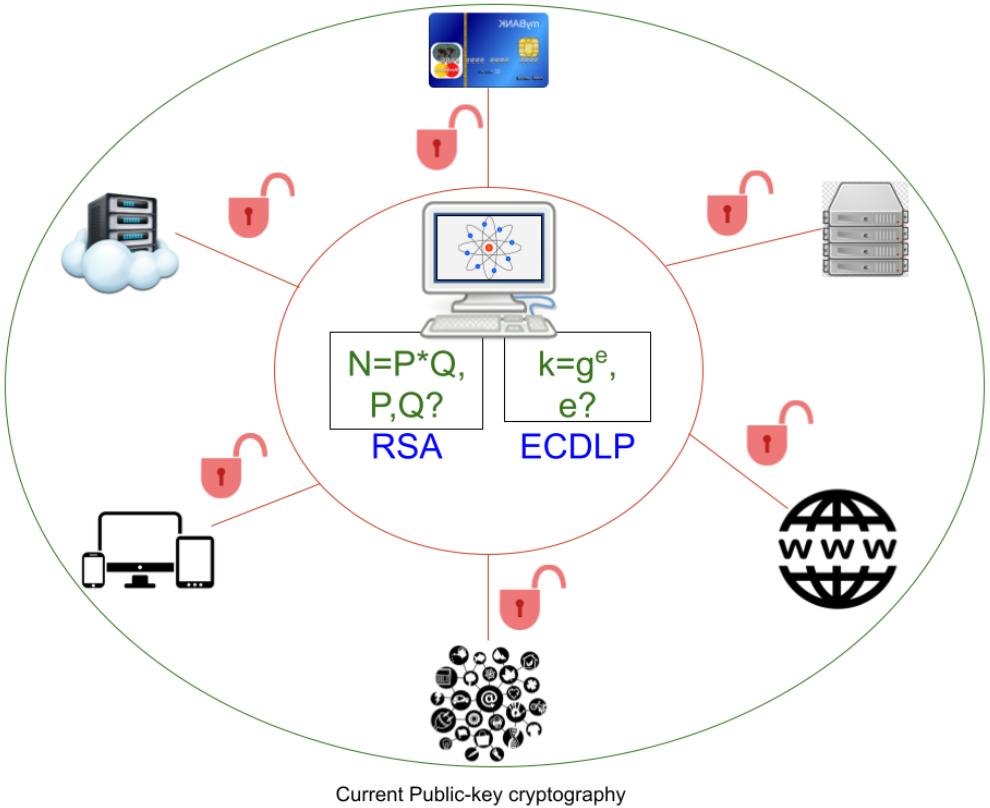
Shor's algorithm

- In 1994, Peter Shor discovered a quantum algorithm
  - Can factorize a number N into its prime factors
  - Runs in polynomial time
- Soon after a quantum algorithm to solve ECDLP was discovered
  - By John Proos & Christof Zalka

Peter Shor

Quantum computers & Public-key cryptography

Post-quantum cryptography



Current Public-key cryptography

- Shor's and Proos-Zalka's algorithm can solve integer factorization and ECDLP *easily*
- We need *quantum hard problems* to build our future public-key cryptography

NIST PQC standardization

A brief chronology

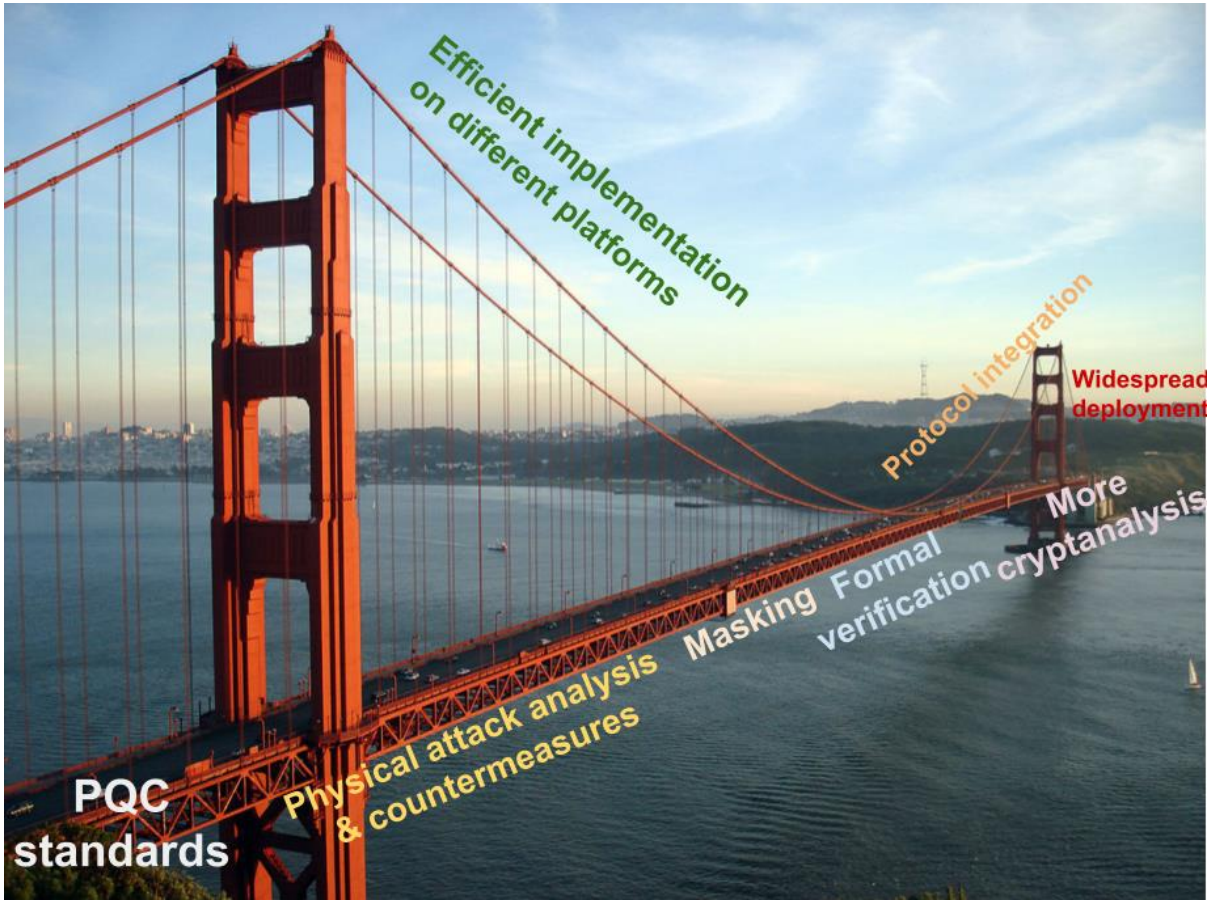
- NIST announced plans for standardization PQC schemes
  - Post-quantum cryptography conference, Fukuoka, Japan, 2016
- On July 22, 2022, NIST announced the selection of four candidate algorithms for standardization:
  - **CRYSTALS-KYBER:** A key encapsulation mechanism (KEM) based on lattice cryptography
  - **CRYSTALS-Dilithium:** A digital signature algorithm (DSA) also based on lattice cryptography
  - **FALCON:** A digital signature algorithm based on NTRU
  - **SPHINCS+:** A digital signature algorithm based on hash-based cryptography
- On August 24, NIST releases FIPS 203, 204, 205 as standard PQC algorithms
- Our designed KEM Saber was one of the 4 finalists

NIST standardization and beyond

Country	PQC Algorithms Under Consideration	Published Guidance
Australia	NIST	CTPCO (2023)
European Commission	NIST	ENISA (2022)
France	NIST (but not restricted to)	ANSSI (2022, 2023)
Japan	Monitoring NIST	CRPTREC
Netherlands	Monitoring NIST, AES, SPHINCS-256, and XMSS	NCSC (2023)
South Korea	KpqC	Ongoing. First round completed

- And many more ……

NIST standardization and beyond



Challenges
Lightweight PQC

Learning with errors

$$(A, b = A \cdot s + e) \in \mathbb{Z}_q{}^{l \times l} \times \mathbb{Z}_q{}^{l}$$

$$\begin{pmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,n} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,n} \end{pmatrix} * \begin{pmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_n \end{pmatrix} \approx \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

- System of approximate equations
- Red→ public value, green→ secret/private value
- Regev's original proposal
  - Quantum hard
- Matrix-vector multiplication is expensive
- Example: Frodo KEM

Module-learning with errors

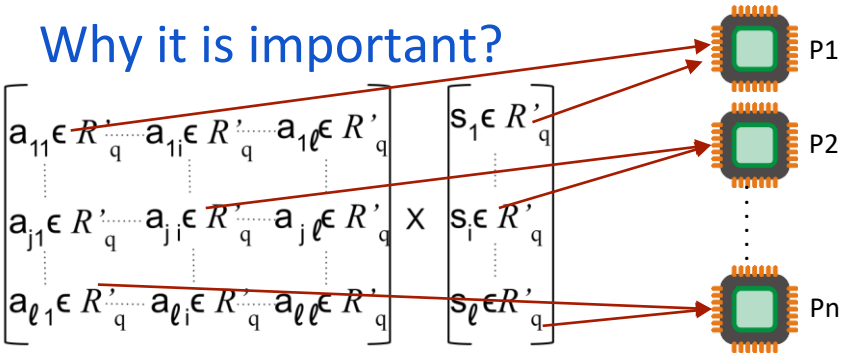
$$(A, b = A \cdot s + e) \in \mathbb{R}_q^{l \times l} \times \mathbb{R}_q^{l} \qquad \mathbb{R}_q = \mathbb{Z}_q[X]/(X^n + 1)$$

$$\begin{pmatrix} \begin{pmatrix} a_{1,1}^{1,1} & \cdots & a_{1,n'}^{1,1} \\ \vdots & \ddots & \vdots \\ a_{1,n'}^{1,1} & \cdots & a_{1,n'-1}^{1,1} \end{pmatrix} & \cdots & \begin{pmatrix} a_{1,1}^{1,l} & \cdots & a_{1,n'}^{1,l} \\ \vdots & \ddots & \vdots \\ a_{1,n'}^{1,l} & \cdots & a_{1,n'-1}^{1,l} \end{pmatrix} \\ \vdots & \ddots & \vdots \\ \begin{pmatrix} a_{1,1}^{l,1} & \cdots & a_{1,n'}^{l,1} \\ \vdots & \ddots & \vdots \\ a_{1,n'}^{l,1} & \cdots & a_{1,n'-1}^{l,1} \end{pmatrix} & \cdots & \begin{pmatrix} a_{1,1}^{l,l} & \cdots & a_{1,n'}^{l,l} \\ \vdots & \ddots & \vdots \\ a_{1,n'}^{l,l} & \cdots & a_{1,n'-1}^{l,l} \end{pmatrix} \end{pmatrix} * \begin{pmatrix} \begin{pmatrix} s_{1,1}^{1} & \cdots & s_{1,n'}^{1} \\ \vdots & \ddots & \vdots \\ s_{1,n'}^{1} & \cdots & s_{1,n'-1}^{1} \end{pmatrix} \\ \vdots \\ \begin{pmatrix} s_{1,1}^{l} & \cdots & s_{1,n'}^{l} \\ \vdots & \ddots & \vdots \\ s_{1,n'}^{l} & \cdots & s_{1,n'-1}^{l} \end{pmatrix} \end{pmatrix} + \begin{pmatrix} \begin{pmatrix} e_{1,1}^{1} & \cdots & e_{1,n'}^{1} \\ \vdots & \ddots & \vdots \\ e_{1,n'}^{1} & \cdots & e_{1,n'-1}^{1} \end{pmatrix} \\ \vdots \\ \begin{pmatrix} e_{1,1}^{l} & \cdots & e_{1,n'}^{l} \\ \vdots & \ddots & \vdots \\ e_{1,n'}^{l} & \cdots & e_{1,n'-1}^{l} \end{pmatrix} \end{pmatrix} \approx \begin{pmatrix} \begin{pmatrix} b_{1,1}^{1} & \cdots & b_{1,n'}^{1} \\ \vdots & \ddots & \vdots \\ b_{1,n'}^{1} & \cdots & b_{1,n'-1}^{1} \end{pmatrix} \\ \vdots \\ \begin{pmatrix} b_{1,1}^{l} & \cdots & b_{1,n'}^{l} \\ \vdots & \ddots & \vdots \\ b_{1,n'}^{l} & \cdots & b_{1,n'-1}^{l} \end{pmatrix} \end{pmatrix}$$

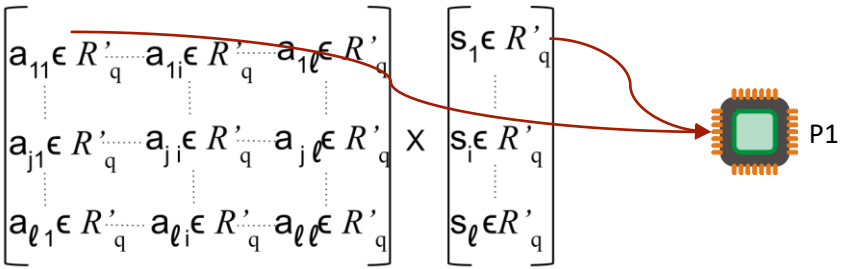
- Trade-off between standard and ring lattices
- Strong security reduction
- Polynomial multiplication → Faster than matrix-vector multiplication
- if $n_{LWE} = n_{R\text{-}LWE} = \ell * n'_{M\text{-}LWE}$ the concrete security is considered equal of all variants
- Example : Kyber, Dilithium, Saber

Module space exploration

- Why it is important?

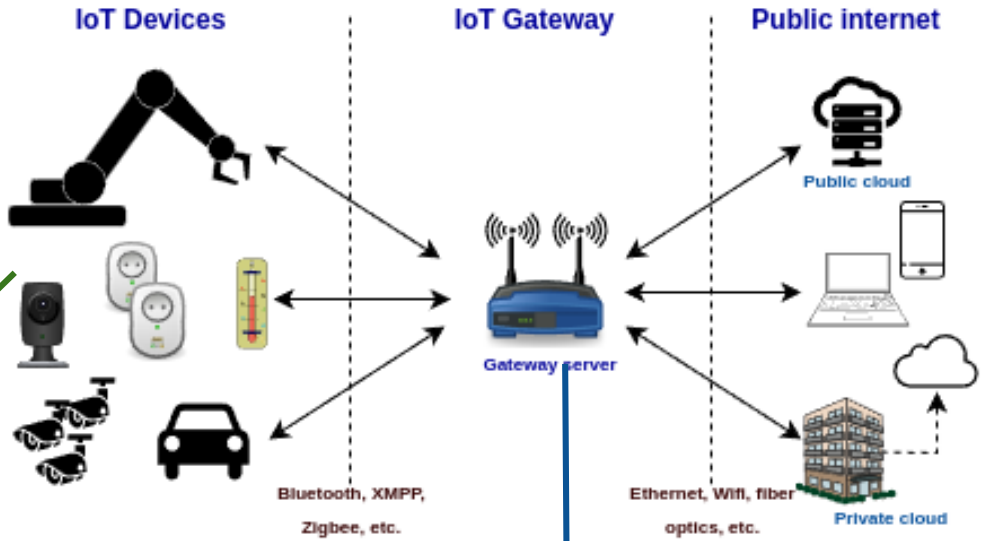


- Multiplications are independent
- Instantiate many parallel multipliers
- Low latency / high power /large area



- Use only one multiplier
- Perform multiplications serially
- High latency / low power /low area

Module space exploration



- Resource constrained devices
- Connects to the gateway server sporadically
- Low area/power is more important than latency
- 1 instance of multiplier and repeat I^2 times

- Powerful devices
- Serves thousands/millions of devices at a time
- Low latency/high throughput is more important than low power/energy
- I^2 parallel instance of multiplier and repeat 1 times

Recent results in ASIC

	Cortex-M4[7]	CICC'18[5]	ISSCC'19[4]	TCAS-I[a][6]	This work
Technology	-	40nm	40nm	65nm	65nm
Supply Voltage	3-5	0.9	0.68-1.1	1.1	0.7-1.1
Frequency (MHz)	100	300	12-72	400	40-160
Total Processor Area (mm^2)	-	2.05	0.28	0.38	0.158
Supported Lattice Crypto Primitives	All	Ring-LWE	Ring-LWE & Module-LWE	Module-LWR	Module-LWR
Supported Lattice Parameters	All	N: 64-2048 q: 32-bit conf.	N: 64-2048 q: 24-bit conf.	N: 256 q: 13-bit	N: 256 q: 13-bit
Average Power	-	140mW	519uW	35-41mW	333.9uW
Individual Multiplier Performance					
Multiplier Cycle	65459	160	1288	81+pipeline	1298*
Energy (nJ)	40.1×10^3	31	63.4	-	40.21

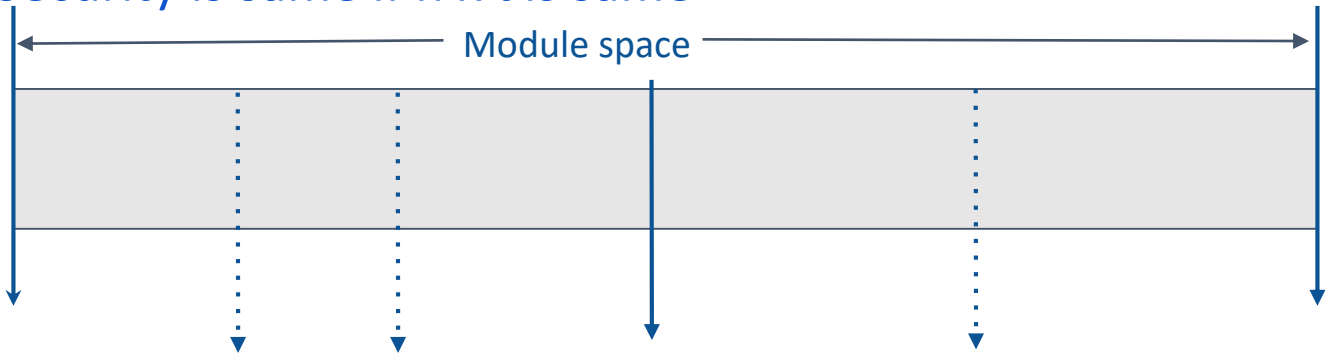
[a] Not silicon verified. Results reported in simulation. *Interpolation needs 70 clock cycle, which happens once in 3 multiplication, Evaluation clock cycle added.

- For module lattices with n'=256 (e.g. Kyber, Dilithium) , one single multiplier is expensive for small devices
- We have to explore other module lattices with different combinations of n' and ℓ

Module space exploration

A direction for lightweight PQC

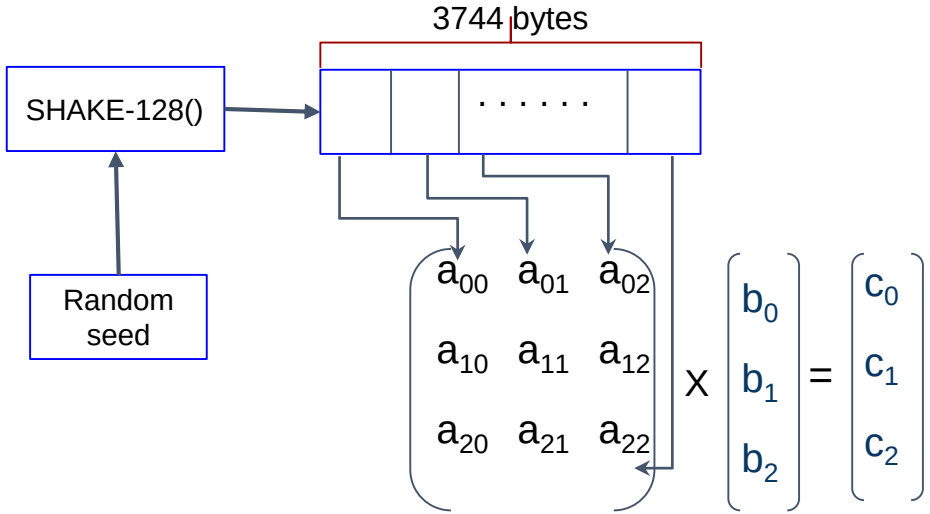
- In module-lattice
  - n'--> length of the smaller polynomials
  - ℓ → number of polynomials in each row/column
- Security is same if n'x ℓ is same



Lightweight PQC

Woe of random numbers

- Random number generation are often considered *free* while designing
- SHAKE-128 (SHA-3) is normally used for random numbers in lattice based PQC



- Researchers have worked to improve the *main* operation *i.e.* polynomial multiplication
- The random number generation now takes upto 70% of time and more than 50% of area
- ASCON is a lightweight cryptography standard that can be used

Lightweight PQC

Some new results

- Our new lightweight PQ design Rudraksh[1]
- Kyber-*esque* design
- n'=64 –> Smaller 64x64 multiplier
- ASCON as random number generator
- *Ultra-lightweight*
- 3x smaller area than state-of-the-art design of Kyber
- ASIC fabrication is on the way

Our lightweight PQ-KEM Rudraksh

- Another design Espada[2] with n'=64 based on learning with rounding
- Appeared in 2021, first introduced the concept module space exploration
- Primary inspiration of SMAUG, a 2nd round candidate in KpqC

[1]Suparna Kundu, Archisman Ghosh, **Angshuman Karmakar**, Shreyas Sen and Ingrid Verbauwhede, "Rudraksh: A compact and lightweight post-quantum key-encapsulation mechanism". https://eprint.iacr.org/2024/1170.pdf

[2]Jose Maria Bermudo Mera, **Angshuman Karmakar**, Suparna Kundu, Ingrid Verbauwhede "Scabbard: a suite of efficient learning with rounding key-encapsulation mechanisms". TCHES 2021

Challenges

Efficient deployment

Deployment of PQC

Large signature issues

- PQC is large

ECDSA signature size	Dilithium signature size
secp256k1: 64 bytes (512 bits) **secp256r1:** 64 bytes (512 bits) **secp384r1:** 96 bytes (768 bits) **secp521r1:** 132 bytes (1056 bits)	**Dilithium-2 (low) :** 2.5KB **Dilithium-2 (Medium) :** 3.3 KB **Dilithium-2 (low) :** 4.6 KB

- Huge overhead in time and space for
  - TLS
  - OIDC
  - Certificate authority
  - FIDO, etc.
- We need smaller PQ signatures with fast verification time
  - Primary motivation for NIST's new call for PQ signatures

Deployment of PQC

Migration issues

- How to Migrate to PQC?
  - Legacy systems?
  - Different organizations adapt to new protocols in different speed
  - Difficult to integrate in IoT devices
- Designing hybrid schemes
  - Classical + PQC
- Hybrid signature

$$\sigma_1 \leftarrow Sign_{ECDSA}(m, SK_{ECDSA})$$

$$\sigma_2 \leftarrow Sign_{Dilithium}((m, \sigma_1), SK_{Dilithium})$$

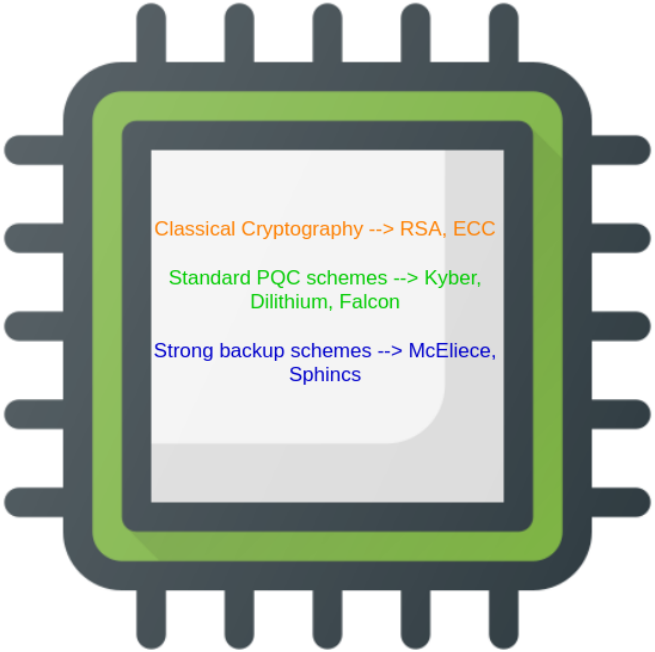
$$Final\ signature\ \sigma = (\sigma_1, \sigma_2)$$

- Can support both classical and PQ signatures
- Thorough security analysis is required

Deployment of PQC

Agility issues

- A cryptoprocessor should be agile → should support multiple schemes
- Why?
  - Breakthroughs happen in cryptanalysis
  - Rainbow and SIDH were NIST finalists
  - Were broken completely
- Some organizations may require stronger security over efficiency
- NIST mentioned SPHINCS and McEliece for as backup schemes if breakthrough in lattices problems happen
- For hybrid schemes the processor has to execute classical algorithms along with PQC

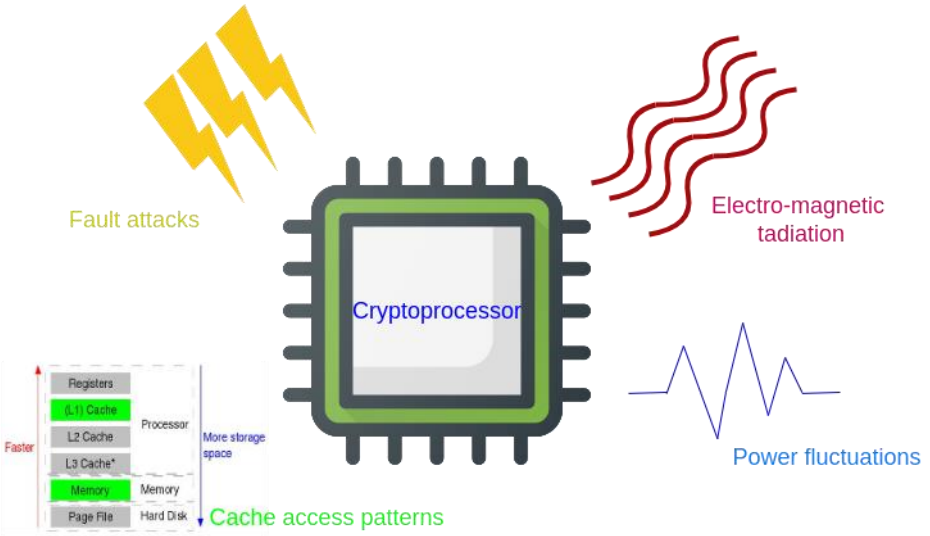


An agile Post-quantum cryptoprocessor

Deployment of PQC

Physical attacks and countermeasures

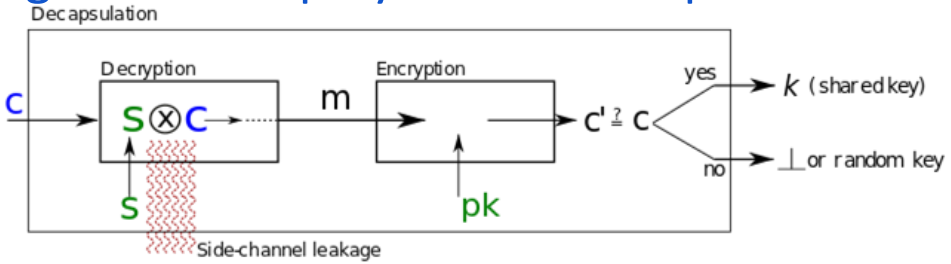
- Most potent threat for deployment of cryptographic schemes
- Platform which executes the scheme leaks information
- Passive attacks
  - Power side-channel, timing, electromagnetic radiation, etc
- Active attacks
  - Voltage glitch, rowhammer attacks, cache-attacks, electromagnetic fault



Deployment of PQC

Physical attacks and countermeasures

- All lattice based schemes use polynomial multiplications
- We targeted[1] the polynomial multiplication routine using CPA



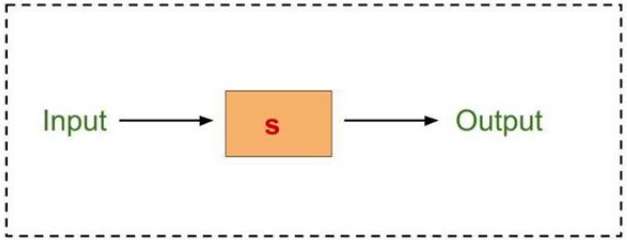
- Exploited the narrow distribution of secret values
- Attack worked on 3 out of 4 finalist schemes
  - NTRU-KEM, Kyber, Saber
- On all different multiplication schemes *i.e.* Toom-Cook, School book, NTT
- Full recovery of all secret values

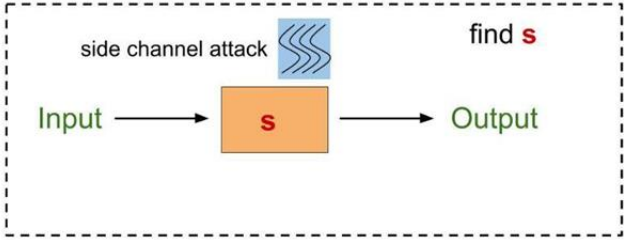
[1] Catinca Mujdei, Lennert Wouters, **Angshuman Karmakar**, Arthur Beckers, Jose Maria Bermudo Mera, Ingrid Verbauwhede: Side-channel Analysis of Lattice-based Post-quantum Cryptography: Exploiting Polynomial Multiplication. ACM TECS 2024.

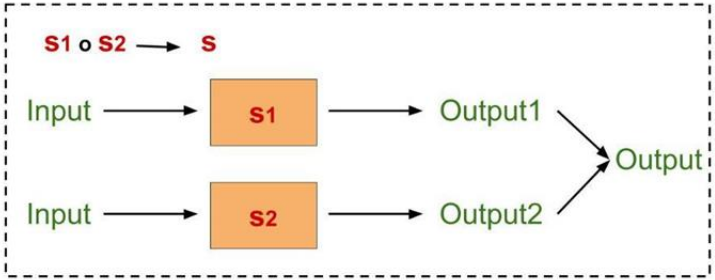
Deployment of PQC

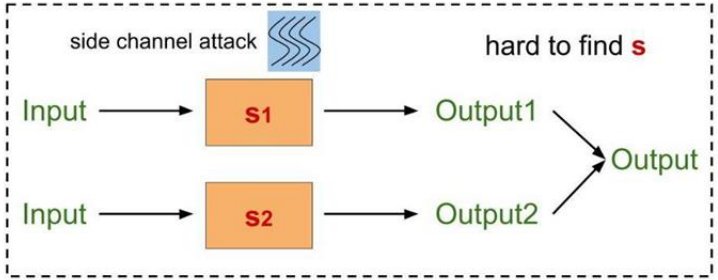
Physical attacks and countermeasures

- Masking of PQC schemes









Deployment of PQC

Physical attacks and countermeasures

- Masking provides provable security against side-channel attacks

Scheme	Masking order			
	unmask	1	2	3
Kyber[1]	804 kcycles	10,018 kcycles	16,747 kcycles	24,709 kcycles
Saber[2]	773 kcycles	3,022 kcycles	5,567 kcycles	8,649 kcycles

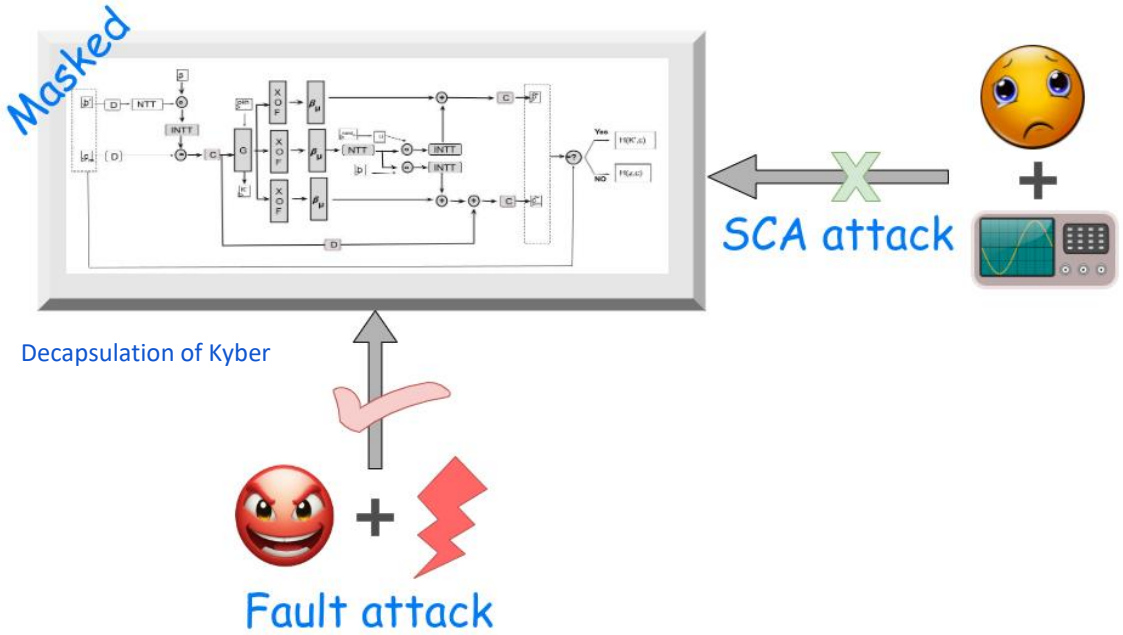
- Introduces huge overhead
- Unsuitable for small IoT devices
- Alternate methods such as shuffling or chip-level countermeasure is essential[3]

[1] Suparna Kundu, Jan-Pieter D'Anvers, Michiel Van Beirendonck, **Angshuman Karmakar,** Ingrid Verbauwhede: Higher-Order Masked Saber. SCN 2022.
[2] Olivier Bronchain, and Gaëtan Cassiers. Bitslicing arithmetic/boolean masking conversions for fun and profit with application to lattice-based kems. TCHES 2022.
[3] Debayan Das, Mayukh Nath, Baibhab Chatterjee, Santosh Ghosh, Shreyas Sen, "STELLAR: A Generic EM Side-Channel Attack Protection through Ground-Up Root-cause Analysis", IEEE HOST 2019

Deployment of PQC

Physical attacks

- Cross-attack: fault attack on masked MLWR based KEM (e.g. Kyber)[1]

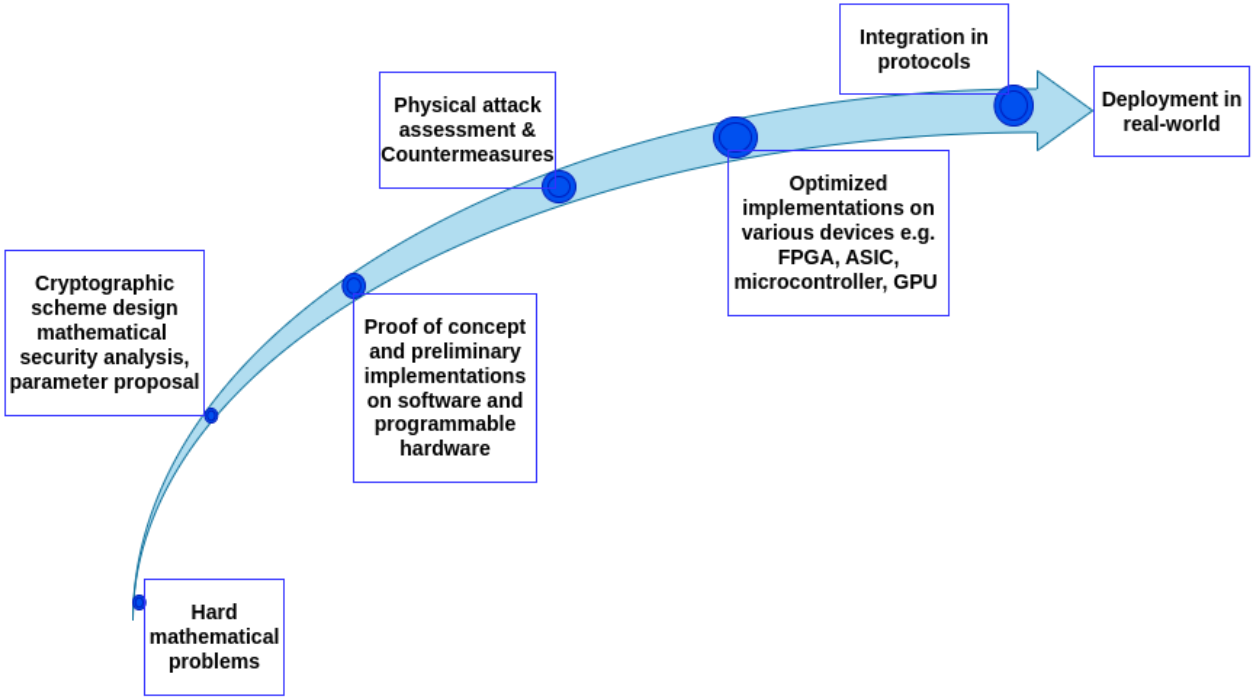


[1] Suparna Kundu, Sayandeep Saha, **Angshuman Karmakar**, Debdeep Mukhopadhyay, Siddhartha Chowdhury, Ingrid Verbauwhede: Carry Your Fault: A Fault Propagation Attack on Side-Channel Protected LWE-based KEM. Transactions in Cryptographic Hardware and Embedded systems 2024 Vol: 2.

Challenges

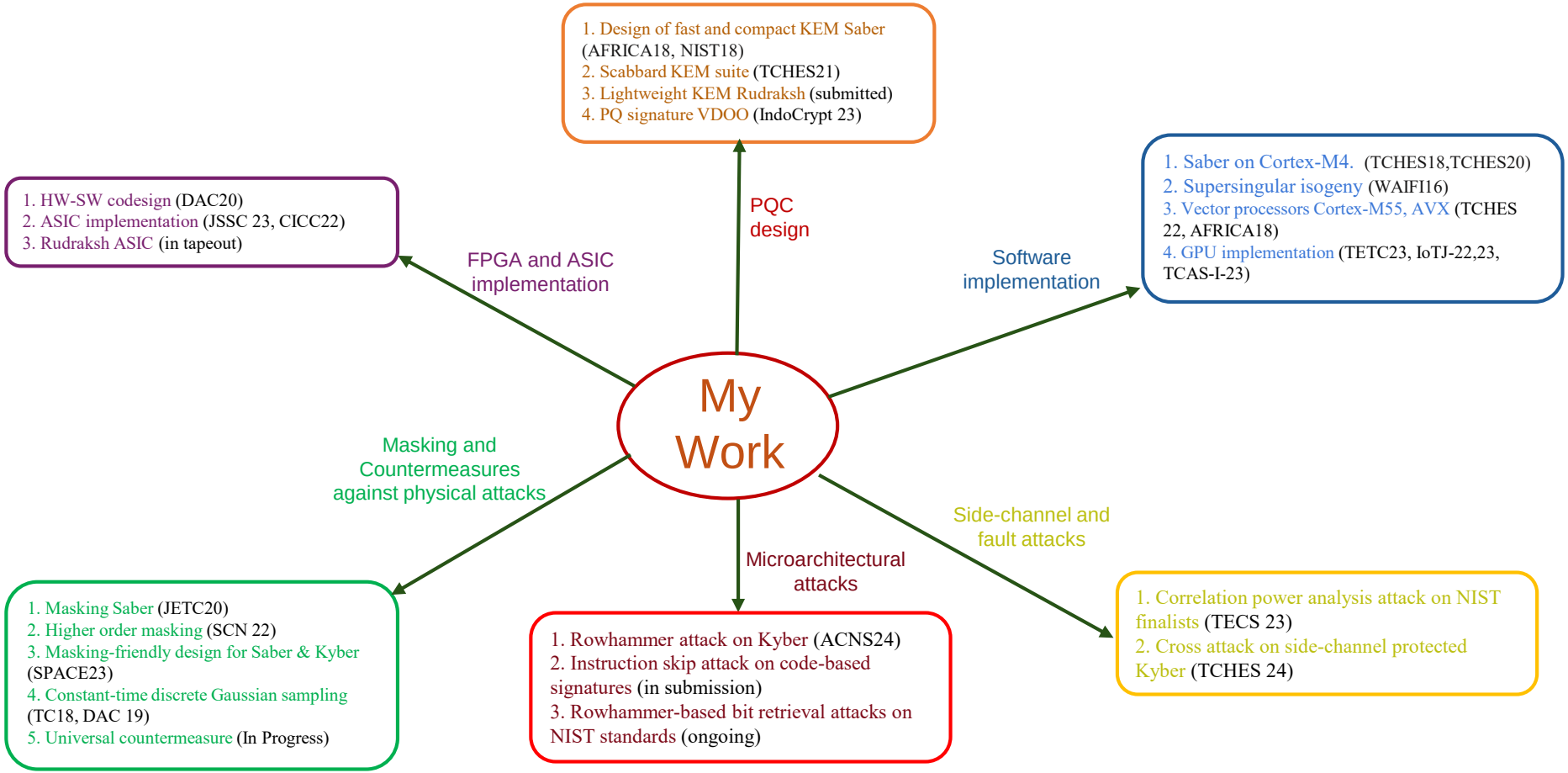
Indigenization

Indigenization



- Typical design-to-deployment life cycle of cryptographic scheme
- Our expertise covers almost the entire spectrum

Our expertise and prior experience



Thank you for your attention!

Questions?

Thank you !