SoK: How (not) to Design and Implement Post-Quantum Cryptography

James Howe[1], Thomas Prest[1], and Daniel Apon[2]

[1] PQShield, Oxford, UK.
{james.howe,thomas.prest}@pqshield.com
[2] National Institute of Standards and Technology, USA.
daniel.apon@nist.gov

Abstract Post-quantum cryptography has known a Cambrian explosion in the last decade. What started as a very theoretical and mathematical area has now evolved into a sprawling research field, complete with side-channel resistant embedded implementations, large scale deployment tests and standardization efforts. This study systematizes the current state of knowledge on post-quantum cryptography. Compared to existing studies, we adopt a transversal point of view and center our study around three areas: (i) paradigms, (ii) implementation, (iii) deployment. Our point of view allows to cast almost all classical and post-quantum schemes into just a few paradigms. We highlight trends, common methodologies, and pitfalls to look for and recurrent challenges.

1 Introduction

Since Shor's discovery of polynomial-time quantum algorithms for the factoring and discrete logarithm problems, researchers have looked at ways to manage the potential advent of large-scale quantum computers, a prospect which has become much more tangible of late. The proposed solutions are cryptographic schemes based on problems assumed to be resistant to quantum computers, such as those related to lattices or hash functions. *Post-quantum cryptography* (PQC) is an umbrella term that encompasses the design, implementation, and integration of these schemes. This document is a Systematization of Knowledge (SoK) on this diverse and progressive topic.

We have made two editorial choices. First, an exhaustive SoK on PQC could span several books, so we limited our study to signatures and key-establishment schemes, as these are the backbone of the immense majority of protocols. This study will not cover more advanced functionalities such as homomorphic encryption schemes, threshold cryptography, et cetera.

Second, most surveys to-date are either (i) organized around each *family* [23] – (a) lattices, (b) codes, (c) multivariate equations, (d) isogenies, (e) hash and one-way functions – or (ii) focused on a single family [146, 83]. Our study instead adopts a transversal approach, and is organized as follows: (a) paradigms, (b) implementation, and (c) deployment. We see several advantages to this approach:

- Compared to previous surveys, it provides a new point of view that abstracts away much of the mathematical complexity of each family, and instead emphasizes common paradigms, methodologies, and threat models.
- In practice, there are challenges that have been solved by one family of scheme and not another. This document's structure makes it easy to highlight *what* these problems are, and *how* they were solved. Consequently, it aims to provide specific direction for research; i.e., (i) problems to solve, and (ii) general methodologies to solve them.
- If a new family of hardness assumptions emerges – as isogeny-based cryptography recently has – we hope the guidelines in this document will provide a framework to safely design, implement, and deploy schemes based on it.

1.1 Our Findings

A first finding is that almost all post-quantum (PQ) schemes fit into one of four paradigms: Fiat-Shamir signatures, Hash-then-sign, Diffie-Hellman key-exchange, and encryption. Moreover, the same few properties (e.g., homomorphism) and folklore tricks are leveraged again and again.

Successful schemes do not hesitate to *bend* paradigms in order to preserve the security proof *and* the underlying assumption. In contrast, forcing an assumption into a paradigm may break the assumption, the security proof, or both.

Our second finding is that many PQ schemes fell short in secure, isochronous implementations which in turn lead to undeserved opinions on side-channel vulnerabilities. We also find some PQ schemes are significantly more amenable to implementations in hardware, software, their efficiencies with masking, which then translates into how performant they are in various use-cases.

Our last finding (see the full version [110]) is that all real-world efforts to deploy post-quantum cryptography will have to contend with new, unique problems. They may require a diverse combination of computational assumptions *woven together* into a single hybrid scheme. They may require special attention to *physical management* of sensitive state. And they have very unbalanced performance profiles, requiring different solutions for different application scenarios.

2 The Raw Material: Hard Problems

We first present the raw material from which cryptographic schemes are made of: hard problems. Although there exists a myriad of post-quantum hard problems, many of them share similarities that we will highlight.

2.1 Baseline: Problems that are not Post-Quantum

We first present problems that are classically hard but quantumly easy. The first family of problems relates to the discrete logarithm in finite groups; that is, the Discrete Logarithm (DLOG) problem, the Decisional Diffie-Hellman (DDH), and the Computational Diffie-Hellman (CDH) problems.

Definition 1 (DLOG/DDH/CDH). *Let $\mathbb{G}$ be a cyclic group of generator g. The discrete logarithm problem (DLOG) and the decisional/computational Diffie-Hellman problems (DDH/CDH) are defined as follows:*

- ***DLOG:*** *Given g^a for a random $a \in |\mathbb{G}|$, find a.*
- ***DDH:*** *Given g^a, g^b and g^c for random $a, b \in |\mathbb{G}|$, determine if $c = ab$.*
- ***CDH:*** *Given g^a, g^b for random $a, b \in |\mathbb{G}|$, compute g^{ab}.*

In cryptography, $\mathbb{G}$ is usually the ring $\mathbb{Z}_p$ for a large prime p, or the group of rational points of an elliptic curve. The following algebraic relations are extremely useful to build cryptosystems, for example Schnorr signatures [168] use (1) and (2) whereas the Diffie-Hellman key-exchange [72] uses (2):

$$g^a \cdot g^b = g^{a+b}, \tag{1}$$

$$(g^a)^b = \left(g^b\right)^a = g^{ab}. \tag{2}$$

The second family of problems relates to factoring.

Definition 2 (RSA and Factoring). *Let p, q be large prime integers, $N = p \cdot q$ and e be an integer.*

- ***Factoring:*** *Given N, find p and q.*
- ***RSA:*** *Efficiently invert the following function over a non-negligible fraction of its inputs:*

$$x \in \mathbb{Z}_N \mapsto x^e \bmod N. \tag{3}$$

For adequate parameters, the problems in Def. 1 and 2 are believed hard to solve by classical computers. However, Shor has shown that they are solvable in polynomial time by a quantum computer [172]. As these problems underlie virtually all current public-key cryptosystems, Shor's discovery motivated the following research for alternative, quantum-safe problems.

2.2 Problems on Lattices

The most well-known problems based on lattices are Learning With Errors (LWE) [158, 134], Short Integer Solution (SIS) [2, 130] and "NTRU" [107].

Definition 3 (SIS, LWE, and NTRU). *Let $\mathcal{R} = \mathbb{Z}_q[x]/(\phi(x))$ be a ring, and $\mathbf{A} \in \mathcal{R}^{n \times m}$ be uniformly random. The Short Integer Solution (SIS) and Learning with Errors (LWE) problems are defined as follows:*

- ***SIS:*** *Find a short nonzero $\mathbf{v} \in \mathcal{R}^m$ such that $\mathbf{A}\mathbf{v} = 0$.*
- ***LWE:*** *Let $\mathbf{b} = \mathbf{A}^t\mathbf{s} + \mathbf{e}$, where $\mathbf{s} \in \mathcal{R}^n$ and $\mathbf{e} \in \mathcal{R}^m$ are sampled from the 'secret' distribution and 'error' distribution, respectively.*
  - ***Decision:*** *Distinguish $(\mathbf{A}, \mathbf{b})$ from uniform.*
  - ***Search:*** *Find $\mathbf{s}$.*
- ***NTRU:*** *Let $h = f/g \in \mathcal{R}$, where $f, g \in \mathcal{R}$ are 'short.' Given h, find f, g.*

SIS, LWE, and NTRU exist in many variants [158, 134, 130, 150], obtained by changing $\mathcal{R}, n, m$, or the error distributions. To give a rough idea, a common choice is to take $\mathcal{R} = \mathbb{Z}_q[x]/(x^d + 1)$, with d a power-of-two, and n, m such that nd and md are in the order of magnitude of 1000. The versatility of SIS, LWE, and NTRU is a blessing and a curse for scheme designers, as it offers freedom but also makes it easy to select insecure parameters [148].

We are not aware of closed formulae for the hardness of SIS, LWE, and NTRU. However, the most common way to attack these problems is to interpret them as lattice problems, then run lattice reduction algorithms [7, 5]. For example, the BKZ algorithm [169] with a blocksize $B \leq nd$ is estimated to solve these in time $\tilde{O}(2^{0.292 \cdot B})$ classically [18], and $\tilde{O}(2^{0.265 \cdot B})$ quantumly [127] via Grover's algorithm.

2.3 Problems on Codes

Error-correcting codes provide some of the oldest post-quantum cryptosystems. These usually rely on two problems:

- The Syndrome Decoding (SD) problem, see Def. 4.
- Hardness of distinguishing a code in a family $\mathcal{F}$ from a pseudorandom one.

We first present SD. Note that it is similar to SIS (Def. 3).

Definition 4 (SD). *Given a matrix $\mathbf{H} \in \mathbb{F}_2^{k \times n}$ and a syndrome $\mathbf{s} \in \mathbb{F}_2^k$, the Syndrom Decoding (SD) problem is to find $\mathbf{e} \in \mathbb{F}_2^n$ of Hamming weight w such that $\mathbf{He} = \mathbf{s}$.*

Since 1962, several algorithms have been presented to solve the SD problem, their complexity gradually improving from $2^{0.1207n}$ [155] to $2^{0.0885n}$ [39]. These algorithms share similarities in their designs and [177] recently showed that when $w = o(n)$, they all have the same asymptotic complexity $\approx 2^{w \log_2(n/k)}$. For many of these algorithms, quantum variants have been proposed. They achieve quantum complexities that are essentially square roots of the classical ones, by using either Grover or quantum walks.

The second problem is not as clearly defined, as it is rather a class of problems. Informally, it states that for a given family $\mathcal{C} = (C_i)_i$ of codes, a matrix $\mathbf{G}$ generating a code $C_i \in \mathcal{C}$ is hard to distinguish from a random matrix. For example, two variants of BIKE [9] assume that it is hard to distinguish from random either of these *quasi-cyclic codes* (or QC codes):

$$h_0/h_1 \tag{4}$$

$$g, g \cdot h_0 + h_1 \tag{5}$$

where $g, h_0, h_1 \in \mathbb{F}_2[x]/(x^r - 1)$, g is random and h_0, h_1 have small Hamming weight. Note that (4) and (5) are reminiscent of NTRU and (ring-)LWE, respectively (see Def. 3). Hence all the lattice problems we have defined have code counterparts, and reciprocally. Besides the QC codes of (4)-(5), another popular family of codes are Goppa codes [135, 55, 24].

2.4 Problems on Multivariate Systems

The third family of problems is based on multivariate systems. In practice, only multivariate *quadratics* (i.e., of degree 2) are used. They are the Multivariate Quadratic (MQ) and Extended Isomorphism of Polynomials (EIP) problems.

Definition 5 (MQ and EIP). *Let $\mathbb{F}$ be a finite field. Let $\mathbf{F} : \mathbb{F}^n \to \mathbb{F}^m$ of the form $\mathbf{F}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x}))$, where each $f_i : \mathbb{F}^n \to \mathbb{F}$ is a multivariate polynomial of degree at most 2 in the coefficients of $\mathbf{x}$.*

- ***MQ:*** *Given $\mathbf{y} \in \mathbb{F}^m$ and the map $\mathbf{F}$:*
  - ***Decision:*** *Is there an $\mathbf{x}$ such that $\mathbf{F}(\mathbf{x}) = \mathbf{y}$?*
  - ***Search:*** *Find $\mathbf{x}$ such that $\mathbf{F}(\mathbf{x}) = \mathbf{y}$.*
- ***EIP:*** *Let $\mathbf{S} : \mathbb{F}^n \to \mathbb{F}^n$ and $\mathbf{T} : \mathbb{F}^m \to \mathbb{F}^m$ be uniformly random affine maps. Given $\mathbf{P} = \mathbf{S} \circ \mathbf{F} \circ \mathbf{T}$ and the promise that the map $\mathbf{F}$ is in a publicly known set $\mathcal{F}$, find $\mathbf{F}$.*

Note that MQ is solvable in polynomial time for $m^2 = O(n)$ or $n^2 = O(m)$; therefore this problem is more interesting when $n = \Theta(m)$, which we assume henceforth. Also note that EIP can be parameterized by the set $\mathcal{F}$ to which the secret map $\mathbf{F}$ belongs. For example, the Unbalanced Oil and Vinegar (UOV) and Hidden Field Equation (HFEv) problems, used by Rainbow [73] and GeMSS [43] respectively, are instantiations of the EIP "framework".

Algorithms for solving MQ or EIP include F4/F5 [81], XL [56, 71] or Crossbred [121]. The best algorithms [181, 30, 121] combine algebraic techniques – e.g., solving Gröbner bases – with exhaustive search, which can be sped up using Grover's algorithm in the quantum setting, see [28] as an example. The asymptotic complexities of these algorithms are clearly exponential in n, but we did not find simple formulae to express them (either classically or quantumly), except for special cases ($q = 2$ and $n = m$) which do not accurately reflect concrete instantiations such as the signature schemes Rainbow [73] and MQDSS [165].

2.5 Problems on One-Way and Hash Functions

The most peculiar family of PQ problems relates to properties of (generic) one-way and hash functions. These problems are algebraically unstructured, which is desirable security-wise, but tends to imply more inefficient schemes.

Definition 6 (Problems on hash functions). *Let $H : X \to Y$ be a function, where $Y = 2^n$.*

- ***Preimage:*** *Given $y \in Y$, find $x \in X$ such that $H(x) = y$.*
- ***Second preimage:*** *Given $x_1 \in X$, find $x_2 \neq x_1$ such that $H(x_1) = H(x_2)$.*
- ***Collision:*** *Find $x_1 \neq x_2$ such that $H(x_1) = H(x_2)$.*

The best classical algorithm against (second) preimage is exhaustive search, hence a complexity $O(2^n)$. Grover's famous quantum algorithm [97] performs this search with a quadratic speed-up, hence a complexity $O(2^{n/2})$. Regarding collision, the best classical algorithm is the birthday attack with a complexity $O(2^{n/2})$, and (disputed) results place the complexity of the best quantum attack between $O(2^{2n/5})$ [47] and $\Theta(2^{n/3})$ [184].

2.6 Problems on Isogenies

Isogeny problems provide a higher-level twist on Def. 1. Elliptic curve cryptography posits that when given g and g^a, with g being a point on an elliptic curve E, it is hard to recover a. Similarly, isogeny-based cryptography posits that given elliptic curves E and E' over $\mathbb{F}_{p^2}$, it is hard to find a surjective group morphism (or *isogeny*, in this context) $\phi : E \to E'$.

Isogeny-based cryptography is a fast-moving field. Elliptic curves can be ordinary ($E[p] \simeq \mathbb{Z}_p$) or supersingular ($E[p] \simeq \{0\}$). Recall that the torsion subgroup $E[n]$ is the kernel of the map $P \in E \mapsto [n]P$. Most isogeny schemes work with supersingular curves, which parameters scale better. Two problems (or variations thereof) have emerged. Def. 7 provides simplified descriptions of them.

Definition 7 (Problems on isogenies). *We define the Supersingular Isogeny Diffie-Hellman (SIDH) and Commutative SIDH (CSIDH) problems as follows:*

- ***SIDH:*** *Given two elliptic curves E, E_A and the value of an isogeny $\phi : E \to E_A$ on $E[\ell^e]$, find ϕ.*
- ***CSIDH:*** *Given two elliptic curves E, E_A, find an efficiently computable isogeny $\phi \in \mathcal{C}\ell(\mathcal{O})$ s.t. $E_A = \phi \cdot E$, where $\mathcal{C}\ell(\mathcal{O})$ is the class group of $\mathcal{O} = \mathbb{Z}[\sqrt{-p}]$.*

Note that the CSIDH problem adapts DDH to the isogeny setting, and one can similarly adapt CDH (see Def. 1). Note that both problems are quantumly equivalent [89], whereas CDH and DDH are not known to be classically equivalent, except in special cases.

For SIDH, the best classical attack is via a claw-finding algorithm due to van Oorschot-Wiener [178]. Surprisingly, a recent result [120] shows that the best known quantum attack performs *worse* than [178]. The hardness of CSIDH reduces to solving a hidden shift problem, for which Kuperberg proposed quantum sub-exponential algorithms [125, 126]. The actual quantum security of CSIDH is still being debated [37, 147].

2.7 Summary of Problems

Fig. 1 summarizes the classical and quantum hardness estimates of the problems we presented. Quantum estimates are particularly prone to change, notably due to (a) the lack of clear consensus on the cost of quantum memory, (b) the prospect of future algorithmic improvements.

Figure 1: Classical and quantum hardness of some problems.

Problem	Factoring /DLOG	SIS /LWE	SD	MQ	EIP	SIDH	CSIDH	(Second) Preimg.	Coll.
Classical	$e^{\tilde{O}((\log p)^{1/3})}$	$2^{0.292 \cdot B}$	$2^{0.0885 \cdot n}$	?	?	$O(p^{1/4})$	$O(p^{1/4})$	$O(2^n)$	$O(2^{n/2})$
Quantum	poly(N)	$2^{0.265 \cdot B}$	$2^{0.05804 \cdot n}$	?	?	$O(p^{1/4})$	$e^{\tilde{O}(\sqrt{\log p})}$	$O(2^{n/2})$	$\Theta(2^{n/3})$

3 Paradigms are Guidelines, not Panaceas

In the classical world, there are two paradigms for signing:

- Fiat-Shamir (FS) [85], proven in the random oracle model (ROM) by [153]. One example is Schnorr signatures and the (Elliptic Curve) Digital Signature Algorithm, (EC)DSA.
- Hash-then-sign. The most prominent formalization of this paradigm is the Full Domain Hash [21] (FDH), proven in the ROM by [22, 54]. Numerous instantiations exist, such as RSA-PSS (Probabilistic Signature Scheme) and Rabin signatures.

There are also two paradigms for key establishment:

- Public-key encryption (PKE), like El Gamal [78] or RSA [160].
- Diffie-Hellman (DH) key-exchange [72].

At a conceptual level, this section shows that most PQ signature or key establishment schemes can be cast under one of these four paradigms. This is summarized by Table 1, which also provides us with two open questions:

(Q1) Can we have isogeny-based Hash-then-sign schemes?
(Q2) Can we have multivariate key establishment schemes?

The prospect that we will have practical key establishment schemes based on symmetric primitives only seems unlikely, see [14]. For (Q1) and (Q2), we hope that the guidelines provided in this section will help to answer them. Our main

Table 1: Correspondence between post-quantum schemes and problems.

	Signature		Key Establishment	
	Hash-&-Sign	Fiat-Shamir	DH-style	PKE
Lattices	[156, 50]	[133, 36]	[149]	[170, 61, 185]
Codes	[68]	[174, 180]	[24, 9]	[1]
Isogenies	?	[65, 34]	[45]	[117]
Multivariate	[73, 43]	[165]	?	?
Symmetric	[115]	[183, 32]	-	-

takeaway is that scheme designers should treat paradigms as guidelines. In particular, a fruitful approach is to weaken some properties, as long as the final scheme achieves meaningful security notions. For example:

- Efficient PQ variants of the FDH framework discards trapdoor permutations for weakened definitions, which suffice for signatures, see Sec. 3.3.
- *Fiat-Shamir with Aborts* changes the protocol flow and may only prove knowledge of an approximate solution. This suffices for signatures, see Sec. 3.1

On the other hand, designers should not try to cram a problem into a predefined paradigm, as it often results in impractical (if not broken) parameters. Examples are rigid adaptations of:

– DH with lattices [102] and isogenies [66], see Sec. 3.4.
– FDH with codes [55] or lattices [105], see Sec. 3.3.

3.1 Schnorr Signatures over Lattices

Fig. 2 recalls the structure of an identification scheme, or ID scheme. Any ID scheme can be converted into a signature via the Fiat-Shamir transform [85]. A efficient ID scheme is Schnorr's 3-move protocol [168]. It instantiates Fig. 2 with the parameters in Table 2 (column 2). It also requires additive and multiplicative properties similar to (1)-(2).

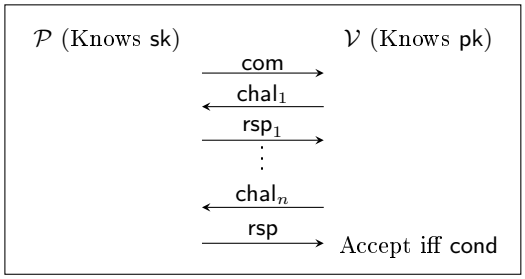

Figure 2: A $(2n+1)$-move ID scheme.

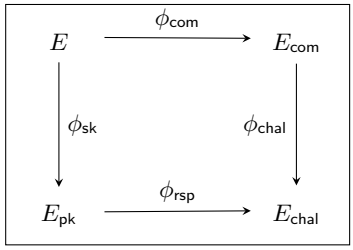

Figure 3: SQISign

Fortunately, lattice and code problems do have properties similar to (1)-(2). An early attempt to propose Schnorr lattice signatures is NSS (NTRU-based Signature Scheme) [106], which was broken by statistical attacks [92]. The high-level explanation is that the ID scheme in NSS did not satisfy the *honest verifier zero-knowledge* (HVZK) property. Each transcript leaked a bit of information about sk, which [92] exploited to recover sk. This was fixed by Lyubashevsky's scheme [132], by giving the prover the possibility to abort the protocol with a probability chosen to factor out the dependency to sk from the signature. This changes the flow of the ID scheme, but allows to prove HVZK. It is also invisible to the verifier as the signer will simply restart the signing procedure in case of an abort. An example instantiation is shown in Table 2 (column 3).

On the other hand, properties of lattices enable specific tricks tailored to this setting. For example, for LWE, least significant bits (LSBs) do not really matter. Let $\lfloor \mathbf{u} \rfloor_b$ be a lossy representation of $\mathbf{u}$ that discards the b LSBs for each coefficient of $\mathbf{u}$. Finding a search-LWE solution $(\mathbf{s}_1, \mathbf{s}_2)$ for $(\mathbf{A}, \lfloor \mathbf{t} \rfloor_b)$ implies a solution $(\mathbf{s}_1, \mathbf{s}_2')$ for $(\mathbf{A}, \mathbf{t})$, with $\|\mathbf{s}_2 - \mathbf{s}_2'\|_\infty \leq 2^b$. This indicates that, as long as b is not too large, LSBs are not too important for LWE.

This intuition was formalized by [13], who show that dropping $\mathbf{z}_2$ and checking only the high bits of com allowed to reduce the signature size by about 2, for essentially the same (provable) security guarantees. Similarly, [98] applied this idea to reduce the public key size. The idea was improved upon by

Table 2: Instantiations of Schnorr Signatures.

Element	Schnorr	Lyubashevsky (w/ LWE)
sk	Uniform x	Short $(\mathbf{s}_1, \mathbf{s}_2)$
pk	$g, h = g^x$	$\mathbf{A}, \mathbf{t} = \mathbf{A} \cdot \mathbf{s}_1 + \mathbf{s}_2$
com	g^r for uniform r	$\mathbf{A} \cdot \mathbf{r}_1 + \mathbf{r}_2$ for short $(\mathbf{r}_1, \mathbf{r}_2)$
chal	Uniform c	Short c
rsp	$r - cx$	$(\mathbf{z}_1, \mathbf{z}_2) = (\mathbf{r}_1 - c\mathbf{s}_1, \mathbf{r}_2 - c\mathbf{s}_2)$
cond	$\mathsf{com} = g^{\mathsf{rsp}} \cdot h^c$	$(\mathsf{com} = \mathbf{A}\mathbf{z}_1 + \mathbf{z}_2 - c\mathbf{t}) \wedge ((\mathbf{z}_i)_i$ short)
Abort?	No	Yes

Dilithium [133]. However, qTESLA [36] provides a textbook example of what can go wrong by trying to apply this idea without checking that the security proof is preserved (in this case, soundness), as it was shown to be completely insecure.

3.2 Beyond Schnorr signatures

For the (vast majority of) problems that do not possess the algebraic properties needed to instantiate Schnorr signatures, there still exist several tricks that enable efficient FS signatures. Scheme designers need to consider two things:

- The soundness error ϵ of the ID protocol is often too large. For example, Stern's code-based protocol has a soundness error $\epsilon = 2/3$. A simple solution is to repeat the protocol k times so that $\epsilon^k \leq 2^{-\lambda}$ for security parameter λ, but finding ways to improve ϵ is also important.
- For some problems, a 3-move ID protocol may be less efficient than an n-move protocol with $n > 3$, or may even not be known.

We first elaborate on the first point. When the soundness ϵ of an ID protocol is too small, the protocol is repeated k times. Typically, all k iterations are performed in parallel (as opposed to sequentially). Parallel repetition is often *expected* by scheme designers to provide exponential soundness ϵ^k, however it is not the case in general; it is proven effective for 3-move *interactive* protocols, but counter-examples exist for protocols with 4 or more moves [20].

Next, we present 3-moves and 5-moves ID schemes. As long as the underlying problem admits some linearity properties, one can build an ID scheme on it [12]. It is the case of all the schemes presented below.

PKP: A 5-move protocol based on the Permuted Kernel Problem (PKP) was proposed in [171], with a soundness error of $\frac{p}{2p-2} \approx 1/2$, where p is the cardinal of the underlying ring. It was later instantiated by PKP-DSS [33].

MQ: The first ID schemes for MQ were proposed by [164]. A key idea of [164] was to use the polar form of $\mathbf{F}$: $\mathbf{G}(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{F}(\mathbf{x}_1 + \mathbf{x}_2) - \mathbf{F}(\mathbf{x}_1) - \mathbf{F}(\mathbf{x}_2)$.
$\mathbf{G}$ is bilinear, and this was exploited to propose a 3-move protocol with soundness error 2/3, and a 5-move one with soundness error $1/2 + 1/q \approx 1/2$. The latter protocol was instantiated by MQDSS [49, 165] using the Fiat-Shamir transform.

Codes: Many code-based schemes derive from Stern's elegant protocols [174, 175], which are based on the SD problem. Stern proposed a 3-move with soundness error 2/3, and a 5-move protocol with soundness error 1/2. The 3-move version was improved by Veron [180] using the generator matrix of a code instead of its parity check matrix, hence it is often seen as a dual of Stern's protocol. However, most derivatives of Stern's protocol are based on the 5-move variant.
Isogenies: The CSIDH problem has been used to propose an ID scheme that, interestingly, is very similar to the well-known proof of knowledge for graph isomorphism. A useful trick used by SeaSign [65] is to use n public keys; this improves the soundness error down to $\frac{1}{n+1}$. CSI-Fish [34] improved it to $\frac{1}{2n+1}$ by using symmetries specific to isogenies. Both schemes combine this with Merkle trees, which provides a trade-off between signing time and soundness error.
Cut-and-choose: This *generic* technique [124] provides a trade-off between signing time and soundness error. It had been used by [31] to provide MQ-based and PKP-based signatures that are more compact than MQDSS and PKP-DSS.

We end on a note of caution. A recent paper [122] shows that for 5-round ID schemes with k parallel repetitions, the soundness error may be larger than ϵ^k, provides a combinatorial attack against the MQ-based schemes of [49, 165], as well as the PKP-based scheme of [33], and warns that it might apply on 5-round variants of Stern's protocol. Designers of schemes that fit this pattern should be careful.

3.3 Full Domain Hash signatures

Hash-then-sign schemes are among the most intuitive schemes to understand at a high level. The standard way to construct them is via the *Full Domain Hash* (FDH) framework. Let $(\mathsf{sk}, \mathsf{pk})$ be an asymmetric keypair. Associate to it a pair $(f_{\mathsf{pk}}, g_{\mathsf{sk}})$ of efficiently computable functions $f_{\mathsf{pk}} : D \to R$ (surjective) and $g_{\mathsf{sk}} : R \to D$ (injective). We say $(f_{\mathsf{pk}}, g_{\mathsf{sk}})$ is:

- A trapdoor permutation (TP) if:
  - (T1) given only pk, f_{pk} is computationally hard to invert.
  - (T2) $f_{\mathsf{pk}} \circ g_{\mathsf{sk}}$ is the identity over R.
  - (T3) For any y, the distribution of $g_{\mathsf{sk}}(y)$ is (statistically) independent of sk.
  - (T4) f_{pk} and g_{sk} are permutations (hence $D = R$).
- A trapdoor preimage sampleable function (TPSF) if it satisfies (T1), (T2), (T3). Hence (T4) is no longer required.
- An average TPSF if it satisfies (T1), (T2), and this relaxation of (T3):
  - (T3*) On average over y, the distribution of $g_{\mathsf{sk}}(y)$ is (statistically) independent of sk.

Note that we have the following relation: TP $\Rightarrow$ TPSF $\Rightarrow$ Average TPSF. The FDH framework [21, 22] allows, in its original form, to build hash-then-sign schemes from a hash function and a TP family as in Fig. 4. Note that the function of (3) is a RSA-based TP for whoever knows the factorization $N = p \cdot q$.

Notable efforts at transposing the FDH framework in a post-quantum setting are the code-based schemes CFS [55] and RankSign [88]. The bit-security of

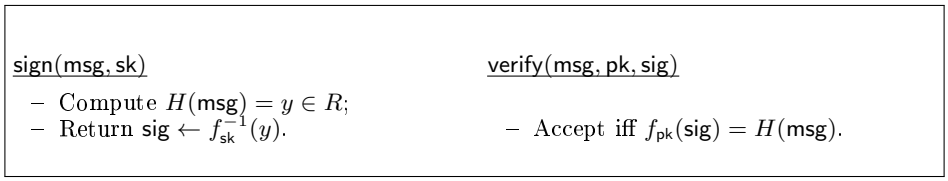

Figure 4: The Full-Domain Hash (FDH) framework.

CFS scales logarithmically in its parameters, making the scheme impractical, and [82] showed that its security proof requires infeasible parameters. Similarly, [69] showed that RankSign's proposed parameters made the underlying problem easy, and that it required impractical parameters for the scheme to be secure. Both CFS and RankSign indicate that a rigid transposition of FDH framework (using TP) in a post-quantum setting seems highly nontrivial.

Early lattice-based attempts such as GGHSign [95] and NTRUSign [105] instead chose to replace TPs with trapdoor one-way functions (with $|D| \gg |R|$), so that only (T1) and (T2) were verified. In particular, the independence property (T3) was no longer verified. However, (T3) plays an important role in the original security proof of the FDH,[1] which did no longer apply. More critically, each $y \in R$ now admitted many $x_i \in D$ such that $f_{\mathsf{pk}}(x_i) = y$, and the x_i picked by the signing algorithm depended of sk. This dependency was exploited by learning attacks [141, 77] to recover the signing key.

For lattices, the first real progress was done by [93]. Its main contribution was to introduce TPSFs, to prove that they can be used to instantiate the FDH, and to propose provably secure lattice-based TPSFs. Several follow-up schemes have been proposed [137, 76], including Falcon [156].

However, it is not known how to instantiate TPSFs from code-based assumptions. Hence the work of [68, 46] relaxed – again – this notion by proposing average TPSFs, showed that they suffice to instantiate the FDH framework, and proposed a signature scheme based on code-based average TPSFs, Wave [68]. Interestingly, this idea was proposed independently by [50], which show that lattice-based average TPSFs require milder parameters than TPSFs, hence improving upon the efficiency of some TPSF-based lattice signatures [29].

$$\text{Trapdoor Permutation} \Rightarrow \text{TPSF} \Rightarrow \text{Average TPSF}.$$

Multivariate cryptography encountered and solved this problem independently. It was first noticed in [163] that some multivariate hash-then-sign schemes relied on a trapdoor function that only verified (T1) and (T2). Hence [163] introduced of a salt during the signing procedure in order to satisfy (T3) and enable a FDH-style proof. This solution is now used by GeMSS [43] and Rainbow [73].

[1] In the case of TPs, the situation is simpler since (T2) + (T4) ⇒ (T3).

3.4 Diffie-Hellman and El Gamal

The Diffie-Hellman (DH) key-exchange protocol [72], as well as the encryption scheme by El Gamal that is derived from it [78], are staples of classical public key cryptography. El Gamal has been notably easier to adapt to PQ assumptions than DH. Classically, DH relies on (2), which provides a simple way for two parties to agree on a shared secret g^{ab}, by instantiating Fig. 5 with Table 3 (column 2). Unfortunately, such a simple relation is harder to obtain with PQ assumptions, as we will see.

Isogenies over elliptic curves are the most natural candidate to instantiate Fig. 5. Unfortunately, the most natural way to do that requires either ordinary curves [57, 162] – which parameters don't scale well [66] –, or supersingular curves with a restricted class of isogenies like CSIDH [45] – which quantum security is debated [37, 147]. A "standard" approach is to use supersingular curves with low-degree isogenies, however it requires to apply the private isogeny $\phi_A : E \to E_A$ to two special points P_B, Q_B of the elliptic curve E, and send the result in addition to E_A. Only with this extra information can the two parties agree on a common curve E_{AB}. A straightforward adaptation of DH to codes and lattices

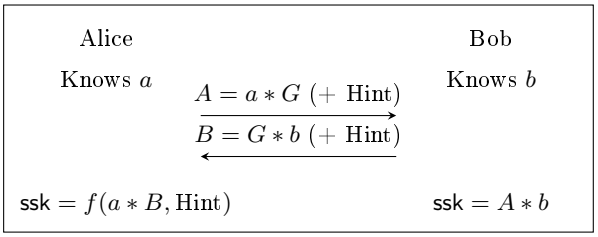

Figure 5: Diffie-Hellman with Reconciliation.

is challenging as well, this time due to *noise*. For example, a rigid transposition with LWE gives:

$$(\mathbf{s}_a^t \cdot \mathbf{A} + \mathbf{e}_a^t)\mathbf{s}_b \approx \mathbf{s}_a^t(\mathbf{A} \cdot \mathbf{s}_b + \mathbf{e}_b) \tag{6}$$

Both parties would end up with "noisy secrets" that differ on their lower bits, which is problematic. In a purely non-interactive setting, this approach does not seem to work, except if q is very large, say $q \geq 2^\lambda$, which is impractical [102]. This is resolved in [74, 149] by sending a hint indicating "how to round the noisy secret". Note that this approach comes at the cost of non-interactivity.

Table 3 summarizes the two approaches to achieve "post-quantum DH" (besides CSIDH). In addition to being interactive, these solutions cannot be used with static key shares, as it would enable key-recovery attacks [86, 90]. As such, they cannot be used as drop-in replacements to non-interactive (semi-)static DH.

Many desirable properties of classical DH are lost in translation when transposing it to a PQ setting. As such, most practical schemes take El Gamal as a starting point instead, replacing DLOG with LWE [140, 170], Learning With Rounding (LWR) [61], or SIDH [117]. Schemes that rely on "trapdoors" – like

Table 3: A few ways to instantiate Fig. 5.

	(EC)DH	SIDH [118, 84]	LWE [74, 149]
G	$g \in \mathbb{G}$	$(P_i, Q_i)_i$	$\mathbf{A} \in R_q^{k \times k}$
a	$a \in \lvert\mathbb{G}\rvert$	Isogeny $\phi_A : E \to E_A$	$(\mathbf{s}_a, \mathbf{e}_a)$ short
A	g^a	$E_A, \phi_A(P_B), \phi_A(Q_B)$	$\mathbf{s}_a^t \cdot \mathbf{A} + \mathbf{e}_a^t$
B	g^b	$E_B, \phi_B(P_A), \phi_B(Q_A)$	$\mathbf{A} \cdot \mathbf{s}_b + \mathbf{e}_b$
Hint	No	Two-way	One-way
Static	Yes	No	No

McEliece [135, 24] or BIKE-2 [9] – are more akin to RSA encryption, though this analogy is a weaker one.

4 Return of Symmetric Cryptography

Another takeaway is that, despite PQC being mostly a public-key matter, symmetric cryptography plays a surprisingly important role and should not be neglected. In particular, two families of signatures based on one-way and hash functions have emerged, with two radically different philosophies:

- Hash-based signatures treat hash functions as *black boxes* and build signatures using only generic data structures and combinatorial tricks, see Sec. 4.1.
- Signatures based on zero-knowledge proofs treat one-way functions as *white boxes* and leverage knowledge of their internal structure to maximize their efficiency, see Sec. 4.2.

Interestingly, some techniques developed by these schemes have also benefited more "standard" schemes. Examples are Merkle trees, used by multivariate [35] and isogeny-based [65, 34] schemes, or the *cut-and-choose* technique [124].

4.1 Hash-based signatures

Hash-based signatures (HBS) are a peculiar family of schemes for two reasons; (a) they rely solely on the hardness properties of hash functions, (b) they follow a paradigm of their own. At a high level:

- The public key pk commits secret values using one or more hash functions.
- Each signature reveals (intermediate) secret values that allow to recompute pk and convince the verifier that the signer does indeed know sk.

Lamport's HBS [129] epitomizes this idea. In its simplest form, the public key is: $\mathsf{pk} = (\mathsf{pk}_{i,0}, \mathsf{pk}_{i,1})_{i \in [\lambda]} = (H(\mathsf{sk}_{i,0}), H(\mathsf{sk}_{i,1}))_{i \in [\lambda]}$, and the signature of a message $\mathsf{msg} = (b_i)_i \in \{0,1\}^\lambda$ is $\mathsf{sig} = (\mathsf{sk}_{i,b_i})_i$. The verifier can then hash sig componentwise and check it against pk. It is easily shown that Lamport's signature scheme is secure under the preimage resistance of H. However, there are two caveats:

- pk and sig require $O(\lambda^2)$ bits, which is rather large.

– It is a one-time signature (OTS), meaning it is only secure as long as it performs no more than one signature.

For four decades, several tricks have been proposed to mitigate these caveats. Because of the unstructured nature of hash functions, these tricks typically rely on combinatorics and/or generic data structures.

One line of research proposes efficient data structures that use OTS as building blocks. By hashing public keys into a tree, Merkle trees [136] allow to improve efficiency and sign more than one message. Goldreich trees [94] use trees' leaves to sign other trees' roots. Both ideas can be combined, as done by SPHINCS(+) [26, 27, 115]. Finally, efficient Merkle tree traversal algorithms were proposed [176].

Another line of research proposed more efficient OTS. The most efficient one so far is a variant of Winternitz's OTS (see [136, 42]), called WOTS+ [114], which uses bitmasks to rely on second-preimage resistance – instead of collision resistance for the original scheme. Stateless few-time signatures (FTS) were also proposed, such as BiBa [151], HORS (Hash to Obtain Random Subsets) [159], a HORS variant with trees, HORST [26], one with PRNGs, PORS [11], and another one with forests, FORS [27, 115]. These can be used to build *stateless* signatures, discussed below.

These tools allow to build hash-based signatures, which can be categorized in two families: *stateful* and *stateless* signatures.

Stateful schemes require the signer to maintain an internal state in order to keep track of the key material used. This encompasses XMSS, its multi-tree variant XMSS^{MT} and LMS, all recently standardized by NIST [52]. Stateful schemes can be efficient but their statefulness is often an undesirable property.

Stateless signatures set their parameters so that, even without maintaining a state, signing many messages will preserve security with overwhelming probability. As a result, they are less efficient than their stateful counterparts, but more flexible. For example, SPHINCS^+ [27, 115] combines Merkle and Goldreich trees with WOTS+ as an OTS, FORS as a FTS, plus a few other tricks.

4.2 Signatures based on ZKPs and OWFs

Signatures based on zero-knowledge proofs (ZKPs) and one-way functions (OWFs) leverage this principle:

– The public key is $\mathsf{pk} = F(\mathsf{sk})$, where F is a OWF.
– A signature is a ZKP that $\mathsf{pk} = F(\mathsf{sk})$; using the MPC-in-the-head [116].

Note that all Fiat-Shamir signatures can already be interpreted as ZKP that $\mathsf{pk} = F(\mathsf{sk})$, however they usually leverage algebraic structure to gain efficiency, and as a result rely on assumptions that are algebraic in nature.

The protocols discussed here are fully generic as they work with any OWF. This is done by leveraging the *MPC-in-the-head* technique [116]. This technique creates non-interactive proofs for an arbitrary circuit (Boolean or arithmetic), by simulating the execution of an MPC (*multiparty computation*) protocol, committing to the execution, and revealing the state of a subset of the parties in order

to let the verifier (partially) check correctness of the execution. Two parallel yet connected lines of research turned this abstract idea into a reality.

The first line of research provides protocols for generic statements. Such protocols have only recently become practical, see ZKB++[48] and KKW [124]. For bit-security λ and a circuit with $|C|$ AND gates, total proof sizes are $O(\lambda|C|)$, for ZKB++, and $O(\lambda|C|/\log n)$, for KKW, respectively, where the *cut-and-choose* approach of KKW allows a trade-off between signing and signature size, via the parameter n. For boolean (resp. arithmetic) circuits of cryptographic sizes, these two schemes (resp. the sacrificing method [17]) are the current state of the art.

The second line of research provides circuits with low multiplicative complexity. Because of their unusual constraints, their internal structure is typically very different from classical symmetric primitives and they require new approaches to be studied. Prominent examples are LowMC [8], which has been extensively studied [75, 119, 131], or the Legendre PRF [59, 96]. Note that these primitives have applications that go far beyond PQC; for example, the Legendre PRF is used by the Ethereum 2.0 protocol.

Combining these two lines of research, one obtain signature schemes. For example, Picnic [183] combines LowMC with either ZKB++ or KKW, BBQ [67] combines AES with KKW, and finally LegRoast [32] combines the Legendre PRF with the sacrificing method [17]. Due to the novely of this approach, it is likely that we will see many more schemes based on it in the future.

5 The Implementation Challenges in PQC

This section discusses the implementation challenges in PQC; specifically discussing attacks via implementation pitfalls and side-channels, countermeasures, and finally the jungle of embedded devices and use-cases for PQC schemes. We somewhat focus on NIST PQC candidates due to similarities in the operations each PQC family requires.

5.1 Decryption Failures and Reaction Attacks

Attacks based on decryption failures – also known as reaction attacks – were first discovered about 20 years ago, with an attack [103] on the McEliece [135] and Ajtai-Dwork [3] cryptosystems, and another [112] on NTRU [107]. They were forgotten for more than a decade before being recently rediscovered. It is clear by now that designers of noisy cryptosystems, such as lattice-based and code-based, need to take these into account. We explain how reaction attacks work and how to thwart them. At a high level, *all* lattice-based and code-based encryption schemes follow this high-level description: $\mathsf{ct} = \mathsf{pk}\cdot\mathbf{e}+\mathbf{e}'+\mathsf{Encode}(\mathsf{msg})$, where $\mathsf{Encode}(\mathsf{msg})$ is an encoding of msg and $(\mathbf{e},\mathbf{e}')$ is a noisy error vector. The decryption key sk is used to obtain $\mathsf{Encode}(\mathsf{msg})$ plus some noise, then recover msg. However, this may fail for a small portion of the admissible $(\mathbf{e},\mathbf{e}')$, and this portion depends on sk. The high-level strategy of reaction attacks uses:

- **Precomputation.** Precompute "toxic" errors $(\mathbf{e}, \mathbf{e}')$ that have a high probability of leading to decryption failures;
- **Query.** Use these toxic errors to send ciphertexts to the target; observe decryption failures.
- **Reconstruction.** Deduce sk from the decryption failures.

Note that reaction attacks are CCA attacks. In CCA schemes, $(\mathbf{e}, \mathbf{e}')$ is generated by passing msg and/or pk into a pseudo-random generator (PRG), so adversaries have to find toxic vectors through exhaustive search. Hence precomputation is often the most computationally intensive phase.

Reaction attacks have been proposed against code-based schemes in the Hamming metric [100], in the rank metric [166], and for lattice-based schemes [60, 64, 101]. Interestingly, attacks against schemes that use lattices or the Hamming metric are very geometric (learning the geometry of the private key), whereas those that target rank metric schemes learn algebraic relations.

For lattice-based schemes, *directional failure boosting* [62] allows, once a toxic error $(\mathbf{e}, \mathbf{e}')$ has been found, to find many more at little cost. Therefore, lattice schemes *must* keep their failure probability negligible, as they are otherwise directly vulnerable to reaction attacks. No such conclusion has been made for code-based schemes yet, but we recommend scheme designers to err on the safe side. Scheme designers need to consider two things with respect to reaction attacks. First, the probability of decryption failures should be negligible.

- This can be achieved by selecting the parameters accordingly, as done by Kyber [170], Saber [61], HQC [1] and FrodoKEM [140]. One may even eliminate them completely like NTRU [185] and NTRU Prime [25], but this may result in slightly larger parameters.
- Another solution is to use redundancy; KEMs need to encapsulate a symmetric key of λ bits, however schemes can often encrypt a much larger message msg. One can use the extra bits to embed an error-correcting code (ECC). However, this solution has two caveats. First, the ECC should be constant-time (e.g., XEf [185] and Melas codes [104]), as timing attacks have been observed when that was not the case [63]. Second, this requires to perform a tedious analysis of the noise distribution; incorrect analyses have led to theoretical attacks [64, 101].

Second, schemes with decryption failures – even negligible – should use CCA transforms that take these into account. In effect, most PQ KEMs in this situation use variants of the transforms described [108], which do handle them.

5.2 Implementation Attacks in PQC

Before NIST began their PQC standardization effort, many PQC schemes were susceptible to implementation attacks; meaning that due to bad coding practices, some attack vectors were found which led to successful attacks. Definition 5 in [111] provides a fairly formal definition for isochronous algorithms (i.e., an algorithm with no timing leakage) which allows us to differentiate between

these initial implementation attacks, of which many did not qualify. Good programming practices exist for ensuring timing analysis resilience and have been well discussed before[2]. These practices cover much more low-level instances of isochronous designs; as conditional jumps, data-dependent branching, and memory accesses of secret information can also lead to detrimental attacks. Some tools such as `ctgrind`, `ctverif`, and `flow-tracker` exist to check whether functions are isochronous, however with operations in PQC such as rejection sampling it is not clear how effective these tools will be. Thus, it would also be prudent to check post-compilation code of the sensitive operations within an implementation.

The first types of implementation attacks on PQC were mainly on the BLISS signature scheme and exploited the cache-timing leakages from the Gaussian samplers, as they mostly operate by accessing pre-computed values stored in memory [40, 152]. The attacks use the FLUSH+RELOAD [182] technique and exploit cache access patterns in the samplers to gain access to some coefficients of values that are added during the signature's calculation. However, optimisations to the Gaussian samplers, such as using guide-tables, and non-isochronous table access enabled these attacks. More leakage sources and implementation attacks against the StrongSwan implementation of BLISS were also found [79], which range from data dependent branches present in the Gaussian sampling algorithm to using branch tracing in the signature's rejection step. These attacks can be mitigated by bypassing conditional branches; that is, using a consistent access pattern (e.g., using linear searching of the table) and having isochronous runtime. In particular, making Gaussian samplers provably secure and statistically proficient have been researched [111] and thus should be followed for secure implementations of lattice-based schemes such as Falcon and FrodoKEM or more advanced primitives such as IBE and FHE.

Although these attacks are on a scheme's implementation, rather than something inherently insecure in its algorithm, they have acted as a cautionary note for how some schemes have operations, which do not use secret information, but could be described as *sensitive* as if they are implemented incorrectly, they can lead to a successful attack. A clear example of this is for Gaussian samplers, which is why they were not used in Dilithium. Once an attacker finds the error vector, $\mathbf{e}$, using these side-channels from a LWE equation of the form $\mathbf{b} = \mathbf{A} \times \mathbf{s} + \mathbf{e} \mod q$, then gaining the secret can be achieved using Gaussian elimination. Moreover, it is not always necessary to find the entire secret, as was the case in the past for RSA [53], and side-channels can be combined with lattice reduction algorithms efficiently to significantly improve attacks on post-quantum schemes. This has been built into a framework [58], which builds in side information into lattice reduction algorithms in order to predict the performance of lattice attacks and estimate the security loss for given side-channel information.

Another sensitive component is in the transient version of the HQC cryptosuite proposed during the NIST PQC standardization process. In the proposed (but now deprecated) reference implementation of decryption, the most costly component was a multiplication in $\mathbb{F}_2[X]/(X^n - 1)$. The crucial operation dur-

[2] See for example https://www.bearssl.org/constanttime.html.

ing decryption is a sparse-dense polynomial multiplication over $\mathbb{F}_2[X]$. At one point in time (specifically, for less than a month in the overall NIST PQC process), it was proposed to use an special algorithm for sparse-dense multiplication, where the complexity of the multiplication was better than the obvious schoolbook algorithm, by utilizing the sparseness of the secret-key polynomial. That is, the multiplication would only access the secret-key polynomial h times, for a secret-key containing only h 1's. In particular, a further, "shielded" version of this algorithm was proposed which applied a permutation (on the memory-access locations) in order to attempt to hide the fact that only h locations were ever accessed in the memory cells corresponding to the secret-key polynomial, while retaining the efficiency benefits of an algorithm specialized to the case of sparse-dense polynomial multiplication. Unfortunately, if an adversary can only observe the memory cells accessed during memory (even without seeing the contents of those memory cells), then – by analogy to an "Oblivious RAM" adversary, the secret key can be directly recovered after one decryption is performed.

A sensitive component that can potentially affect all PQC candidates is in the Fujisaki-Okamoto (FO) transformation. This component is required in lattice-based and code-based KEMs in order to covert the CPA-secure part into an IND-CCA secure scheme. However, it has been shown that this operation is also sensitive to timing attacks, even though the operations do not use any secret information. This attack [99] was shown on FrodoKEM, and was enabled due to its use of non-isochronous `memcmp` in the implementation of the ciphertext comparison step, which allows recovery of the secret key with about 2^{30} decapsulation calls. This attack is directly applied to FrodoKEM, but is likely that other PQC candidates such as BIKE, HQC, and SIKE are also susceptible.

An algorithm used within the FO transform is Keccak, or more specifically SHAKE, which was standardized by NIST in FIPS-202 for SHA-3 and is used extensively within NIST PQC candidates for so-called seed-expansion and computation of the shared secret. This symmetric operation is also sensitive to side-channels and could potentially lead to recovery of the shared-secret generated in the KEM. In particular, a single trace attack was demonstrated on the Keccak permutation in the ephemeral key setting [123], but seemingly realistic only on 8-bit devices.

Finally we consider the peculiar nature of BIKE's (sensitive) decryption module. The BIKE decryption algorithm is naturally designed to proceed in a repetitive sequence of steps. Some operations are performed, then the message is properly decrypted, or not. Such operations can then be repeated, and the likelihood of proper decryption will increase. Unlike most other PQ decryption procedures, the BIKE decryption algorithm is not inherently isochronous, nor is the decryption failure rate well-understood. Given the real-world requirement that all secret-sensitive procedures are isochronous, it has been proposed to therefore artificially truncate this iterative decryption procedure at some fixed number of steps. Experimentally, a round-count as small as 10 is sufficient to guarantee proper decryption. However, in contrast to the case of lattice-based KEMs, there is no mathematical guarantee that, e.g., 10 iterations is sufficient to reduce

the decryption failure rate of the scheme below 2^{λ}, where $\lambda \in \{128, 192, 256\}$ is the concrete security parameter.[3] Therefore, despite the BIKE scheme being designed as first a CPA scheme along with a CPA-to-CCA transform implemented as low cost, the BIKE team has only formally claimed CPA-security (that is, ephemeral key security) for their construction, as opposed to CCA-security (that is, long-term key security). It remains open to provide a "proper analysis" of the BIKE decryption algorithm guaranteeing sufficient precision of decryption failures to ensure long-term key security for the scheme.

5.3 Side-Channels and Countermeasures

In the Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process [4] it is stated that:

> *NIST hopes to see more and better data for performance in the third round. This performance data will hopefully include implementations that protect against side-channel attacks, such as timing attacks, power monitoring attacks, fault attacks, etc.*

In their initial submission requirements [142] NIST also noted that "schemes that can be made resistant to side-channel attacks at minimal cost are more desirable than those whose performance is severely hampered by any attempt to resist side-channel attacks". Thus, some of the remaining candidates also have offer masked implementations, or this has been contributed by the research community.

Migliore et al. [138] demonstrate DPA weaknesses in the unmasked Dilithium implementation, and in addition to this provide a masking scheme using the Ishai-Sahai-Wagner (ISW) probing model following the previous techniques for masking GLP and BLISS [15, 16]. Like the previous provably secure masking schemes, they alter some of the procedures in Dilithium by adding in efficient masking of its sensitive operations. Moreover, some parameters are changed to gain extra performance efficiencies in the masked design, such as making the prime modulus a power-of-two, which increases the performance by 7.3 to 9 times compared to using the original prime modulus during masking. A power-of-two modulus means the optimised multiplication technique, the NTT multiplier, is no longer possible so they proposed Karatsuba multiplication. The results for key generation and signing are between 8 to 12 times slower for order 2 masking and 13 to 28 times slower for order 3 masking, compared to the reference implementations. This is also backed-up by experimental leakage tests on the masked designs.

Similarly, Verhulst [179] provides DPA on Saber, as well as developing a masking scheme for its decryption protocol, which is later extended in [19]. The masking schemes only use additive first-order masking which thus makes it only 2 to 2.5 times slower than being unprotected. However it is probably still vulnerable to template attacks [143]. Saber lends itself to practical masking due to its use of LWR, as opposed to other KEMs using (M-)LWE. However, Saber uses

[3] Known, formal analyses guarantees are closer to 2^{-40} at 128-bit security.

a less efficient multiplication method (a combination of Toom-Cook, Karatsuba, and schoolbook multiplication) compared to schemes which use number theoretic transform (NTT); thus it is an interesting open question as to whether NTT is the most practical multiplication method (due to its conflict with efficient masking) and how these masked PQC schemes practically compare, particularly with the recent research improving the performance of Saber and others using NTTs [51].

NTRU and NTRU Prime both have the potential of using a combination of Toom-Cook and Karatsuba to speed-up their polynomial multiplication, thus whether they can reuse techniques from Saber's masked implementation is an important research question. NTRU Prime in particular requires masking since some power analysis attacks can read off the secret key with the naked eye [113]. Attacks on these multiplication methods, which are in the time-domain, are likely to be simpler than those in the NTT or FFT domains as there is only one multiplication per coefficient of the secret, which thus makes protection of this multipliers more urgent. A single-trace power analysis attack on FrodoKEM exploits the fact that the secret matrix is used multiple times during the matrix multiplication operation, enabling horizontal differential power analysis [38].

Correlation power analysis and algebraic key recovery attacks have also been shown on the schemes Rainbow and UOV [144] by targeting the secret maps within the MQ signature schemes, during the matrix-vector computations. This attack is relevant for many MQ schemes that use the affine-substitution quadratic-affine (ASA) structure. They also discuss countermeasures to simple and differential power analysis by using standard methods seen before such as shuffling of the indices or adding a pseudo-random matrix (i.e., additive masking).

QcBits, a variant of McEliece PKE, was shown to be susceptible to DPA [161]. The attack partially recovers the secret key during the syndrome computation of the decoding phase. They also propose a simple countermeasure for the syndrome calculation stage, which exploits the fact that since QC-MDPC (quasi-cyclic moderate-density parity-check) codes are linear, the XOR of two codewords is another codeword. Thus, a codeword can be masked by XORing it with another random codeword before the syndrome calculation.

This attack was then extended [173] to recover the *full* secret of QcBits, with more accuracy, using a multi-trace attack. Moreover, using the DPA countermeasures proposed in [161] and in the ephemeral key setting, they provide a single-trace attack on QcBits. Lastly and most interestingly, they describe how these attacks can be applied to BIKE, by targetting the private syndrome decoding computation stage where long-term keys are utilized. For ephemeral keys, the multi-target attacks are not applicable, however the single-trace attack can be applied to recover the private key and also the secret message.

Classic McEliece is also not immune from side-channel attacks targeting this operation. A reaction attack [128] using iterative chunking and information set decoding can enable recovery of the values of the error vector using a single decryption oracle request.

Masking schemes which use matrix multiplication have the potential to be efficiently masked using affine masking (i.e., a combination of additive and multiplicative masking) similarly used in the Advanced Encryption Standard (AES) [87]. First-order additive masking has already been proposed for FrodoKEM [109]. Warnings for side-channel protection were also seen in Picnic, where the attack was able to recover the shared secret and the secret key, by targetting the LowMC block cipher, a core component to the signature scheme [91].

PQC schemes have also been shown to be susceptible to cold-boot attacks [154, 6], which was previously shown on NTRU [145]. Cold-boot attacks exploit the fact that secret data can remain in a computer's memory (DRAM) after it is powered down and supposedly deleted. Albrecht et al. [6] describe how to achieve this by attacking the secret-keys stored for use in the NTT multiplier in Kyber and NewHope, and after some post-processing using lattice reductions, is able to retrieve the secret-key.

Fault attacks have also been investigated for PQC schemes. One of the most famous (microarchitectural) fault attacks is the Rowhammer exploit (CVE-2015-0565), which allows unprivileged attackers to corrupt or change data stored in certain, vulnerable memory chips, and has been extended to other exploits such as RAMBleed (CVE-2019-0174). QuantumHammer [139] utilises this exploit to recover secret key bits on LUOV, a second round NIST PQC candidate for multivariate-quadratic signatures. The attack does somewhat exploit the 'lifted' algebraic structure that is present in LUOV, so whether this attack could be applied to other PQC schemes is an open question.

Determinism in signatures is generally considered preferable from a security perspective, as attacks are possible on randomly generated nonces (e.g., [80]). This prompted EdDSA, which uses deterministically generated nonces. NIST [4] noted the potential for nonce reuse in PQC schemes such as Kyber. Indeed, fault attacks which exploit the scheme's determinism have been demonstrated on SPHINCS$^+$ [44] and Dilithium [41, 157], with EdDSA also showing susceptibility to DPA [167]. As such, some PQC candidates offer an optional non-deterministic variant, such as SPHINCS$^+$ using `OptRand`, or random *salt* used in Dilithium, Falcon, GeMSS, Picnic, and Rainbow.

An interesting alternative to mitigating these fault attacks (and randomness failures) is by using *hedging*, which creates a middle-ground between fully deterministic and fully probabilistic signatures, by deriving the per-signature randomness from a combination of the secret-key, message, and a nonce. This is formalized for Fiat-Shamir signatures and apply the results to hedged versions of XEdDSA, a variant of EdDSA used in the Signal messaging protocol, and to Picnic2, and show hedging mitigates many of the possible fault attacks [10].

Key reuse attacks, which have been shown to cause issues for real-world implementations of the EMV ("Europay, Mastercard, Visa") standard [70], are also applicable in PQC; such as lattice-based schemes [86], supersingular isogeny-based schemes [90], and potentially more.

We continue the practical discussions on PQC in the full version of this paper [110], focusing on embedded implementations and use cases, and then providing

an overview of how PQC is being standardized, what new protocols are being designed, and any large scale experiments that have been conducted thus far.

References

[1] C. Aguilar Melchor, N. Aragon, S. Bettaieb, L. Bidoux, O. Blazy, J.-C. Deneuville, P. Gaborit, E. Persichetti, and G. Zémor. *HQC*. Tech. rep. National Institute of Standards and Technology, 2019.

[2] M. Ajtai. "Generating Hard Instances of Lattice Problems (Extended Abstract)". In: *28th ACM STOC*. 1996.

[3] M. Ajtai and C. Dwork. "A Public-Key Cryptosystem with Worst-Case/Average-Case Equivalence". In: *29th ACM STOC*. 1997.

[4] G. Alagic, J. Alperin-Sheriff, D. Apon, D. Cooper, Q. Dang, J. Kelsey, Y.-K. Liu, C. Miller, D. Moody, R. Peralta, et al. "Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process". In: *NIST, Tech. Rep., July* (2020).

[5] M. R. Albrecht, B. R. Curtis, A. Deo, A. Davidson, R. Player, E. W. Postlethwaite, F. Virdia, and T. Wunderer. "Estimate All the LWE, NTRU Schemes!" In: *SCN 18*. 2018.

[6] M. R. Albrecht, A. Deo, and K. G. Paterson. "Cold Boot Attacks on Ring and Module LWE Keys Under the NTT". In: *IACR TCHES* 3 (2018).

[7] M. R. Albrecht, R. Player, and S. Scott. "On the concrete hardness of Learning with Errors". In: *J. Math. Cryptol.* 3 (2015).

[8] M. R. Albrecht, C. Rechberger, T. Schneider, T. Tiessen, and M. Zohner. "Ciphers for MPC and FHE". In: *EUROCRYPT*. 2015.

[9] N. Aragon, P. Barreto, S. Bettaieb, L. Bidoux, O. Blazy, J.-C. Deneuville, P. Gaborit, S. Gueron, T. Guneysu, C. Aguilar Melchor, R. Misoczki, E. Persichetti, N. Sendrier, J.-P. Tillich, G. Zémor, and V. Vasseur. *BIKE*. Tech. rep. National Institute of Standards and Technology, 2019.

[10] D. F. Aranha, C. Orlandi, A. Takahashi, and G. Zaverucha. "Security of Hedged Fiat-Shamir Signatures Under Fault Attacks". In: *EUROCRYPT*. 2020.

[11] J.-P. Aumasson and G. Endignoux. "Improving Stateless Hash-Based Signatures". In: *CT-RSA*. 2018.

[12] M. Backendal, M. Bellare, J. Sorrell, and J. Sun. "The Fiat-Shamir Zoo: Relating the Security of Different Signature Variants". In: *NordSec*. 2018.

[13] S. Bai and S. D. Galbraith. "An Improved Compression Technique for Signatures Based on Learning with Errors". In: *CT-RSA*. 2014.

[14] B. Barak and M. Mahmoody-Ghidary. "Merkle's Key Agreement Protocol is Optimal: An $O(n^2)$ Attack on Any Key Agreement from Random Oracles". In: *Journal of Cryptology* 3 (2017).

[15] G. Barthe, S. Belaïd, T. Espitau, P.-A. Fouque, B. Grégoire, M. Rossi, and M. Tibouchi. "Masking the GLP Lattice-Based Signature Scheme at Any Order". In: *EUROCRYPT*. 2018.

[16] G. Barthe, S. Belaïd, T. Espitau, P.-A. Fouque, M. Rossi, and M. Tibouchi. "GALACTICS: Gaussian Sampling for Lattice-Based Constant-Time Implementation of Cryptographic Signatures, Revisited". In: *ACM CCS*. 2019.

[17] C. Baum and A. Nof. "Concretely-Efficient Zero-Knowledge Arguments for Arithmetic Circuits and Their Application to Lattice-Based Cryptography". In: *PKC*. 2020.

[18] A. Becker, L. Ducas, N. Gama, and T. Laarhoven. "New directions in nearest neighbor searching with applications to lattice sieving". In: *SODA*. 2016.

[19] M. V. Beirendonck, J.-P. D'Anvers, A. Karmakar, J. Balasch, and I. Verbauwhede. *A Side-Channel Resistant Implementation of SABER*. Cryptology ePrint Archive, Report 2020/733. 2020.

[20] M. Bellare, R. Impagliazzo, and M. Naor. "Does Parallel Repetition Lower the Error in Computationally Sound Protocols?" In: *38th FOCS*. 1997.

[21] M. Bellare and P. Rogaway. "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols". In: *ACM CCS 93*. 1993.

[22] M. Bellare and P. Rogaway. "The Exact Security of Digital Signatures: How to Sign with RSA and Rabin". In: *EUROCRYPT'96*. 1996.

[23] *Post-Quantum Cryptography*. 2009.

[24] D. J. Bernstein, T. Chou, T. Lange, I. von Maurich, R. Misoczki, R. Niederhagen, E. Persichetti, C. Peters, P. Schwabe, N. Sendrier, J. Szefer, and W. Wang. *Classic McEliece*. Tech. rep. National Institute of Standards and Technology, 2019.

[25] D. J. Bernstein, C. Chuengsatiansup, T. Lange, and C. van Vredendaal. *NTRU Prime*. Tech. rep. National Institute of Standards and Technology, 2019.

[26] D. J. Bernstein, D. Hopwood, A. Hülsing, T. Lange, R. Niederhagen, L. Papachristodoulou, M. Schneider, P. Schwabe, and Z. Wilcox-O'Hearn. "SPHINCS: Practical Stateless Hash-Based Signatures". In: *EUROCRYPT*. 2015.

[27] D. J. Bernstein, A. Hülsing, S. Kölbl, R. Niederhagen, J. Rijneveld, and P. Schwabe. "The SPHINCS$^+$ Signature Framework". In: *ACM CCS*. 2019.

[28] D. J. Bernstein and B.-Y. Yang. *Asymptotically faster quantum algorithms to solve multivariate quadratic equations*. Cryptology ePrint Archive, Report 2017/1206. 2017.

[29] P. Bert, P.-A. Fouque, A. Roux-Langlois, and M. Sabt. "Practical Implementation of Ring-SIS/LWE Based Signature and IBE". In: *Post-Quantum Cryptography - 9th International Conference, PQCrypto*. 2018.

[30] L. Bettale, J. Faugère, and L. Perret. "Solving polynomial systems over finite fields: improved analysis of the hybrid approach". In: *ISSAC*. 2012.

[31] W. Beullens. "Sigma Protocols for MQ, PKP and SIS, and Fishy Signature Schemes". In: *EUROCRYPT*. 2020.

[32] W. Beullens and C. de Saint Guilhem. "LegRoast: Efficient Post-quantum Signatures from the Legendre PRF". In: *Post-Quantum Cryptography - 11th International Conference, PQCrypto*. 2020.

[33] W. Beullens, J.-C. Faugère, E. Koussa, G. Macario-Rat, J. Patarin, and L. Perret. "PKP-Based Signature Scheme". In: *INDOCRYPT*. 2019.

[34] W. Beullens, T. Kleinjung, and F. Vercauteren. "CSI-FiSh: Efficient Isogeny Based Signatures Through Class Group Computations". In: *ASIACRYPT*. 2019.

[35] W. Beullens, B. Preneel, and A. Szepieniec. "Public Key Compression for Constrained Linear Signature Schemes". In: *SAC*. 2019.

[36] N. Bindel, S. Akleylek, E. Alkim, P. S. L. M. Barreto, J. Buchmann, E. Eaton, G. Gutoski, J. Kramer, P. Longa, H. Polat, J. E. Ricardini, and G. Zanon. *qTESLA*. Tech. rep. National Institute of Standards and Technology, 2019.

[37] X. Bonnetain and A. Schrottenloher. "Quantum Security Analysis of CSIDH". In: *EUROCRYPT*. 2020.

[38] J. W. Bos, S. Friedberger, M. Martinoli, E. Oswald, and M. Stam. "Assessing the Feasibility of Single Trace Power Analysis of Frodo". In: *SAC*. 2019.

[39] L. Both and A. May. "Decoding Linear Codes with High Error Rate and Its Impact for LPN Security". In: *Post-Quantum Cryptography - 9th International Conference, PQCrypto*. 2018.

[40] L. G. Bruinderink, A. Hülsing, T. Lange, and Y. Yarom. "Flush, Gauss, and Reload - A Cache Attack on the BLISS Lattice-Based Signature Scheme". In: *CHES*. 2016.

[41] L. G. Bruinderink and P. Pessl. "Differential Fault Attacks on Deterministic Lattice Signatures". In: *IACR TCHES* 3 (2018).

[42] J. Buchmann, E. Dahmen, S. Ereth, A. Hülsing, and M. Rückert. "On the Security of the Winternitz One-Time Signature Scheme". In: *AFRICACRYPT 11*. 2011.

[43] A. Casanova, J.-C. Faugère, G. Macario-Rat, J. Patarin, L. Perret, and J. Ryckeghem. *GeMSS*. Tech. rep. National Institute of Standards and Technology, 2019.

[44] L. Castelnovi, A. Martinelli, and T. Prest. "Grafting Trees: A Fault Attack Against the SPHINCS Framework". In: *Post-Quantum Cryptography - 9th International Conference, PQCrypto*. 2018.

[45] W. Castryck, T. Lange, C. Martindale, L. Panny, and J. Renes. "CSIDH: An Efficient Post-Quantum Commutative Group Action". In: *ASIACRYPT*. 2018.

[46] A. Chailloux and T. Debris-Alazard. "Tight and Optimal Reductions for Signatures Based on Average Trapdoor Preimage Sampleable Functions and Applications to Code-Based Signatures". In: *PKC*. 2020.

[47] A. Chailloux, M. Naya-Plasencia, and A. Schrottenloher. "An Efficient Quantum Collision Search Algorithm and Implications on Symmetric Cryptography". In: *ASIACRYPT*. 2017.

[48] M. Chase, D. Derler, S. Goldfeder, C. Orlandi, S. Ramacher, C. Rechberger, D. Slamanig, and G. Zaverucha. "Post-Quantum Zero-Knowledge and Signatures from Symmetric-Key Primitives". In: *ACM CCS*. 2017.

[49] M.-S. Chen, A. Hülsing, J. Rijneveld, S. Samardjiska, and P. Schwabe. "From 5-Pass MQ-Based Identification to MQ-Based Signatures". In: *ASIACRYPT*. 2016.

[50] Y. Chen, N. Genise, and P. Mukherjee. "Approximate Trapdoors for Lattices and Smaller Hash-and-Sign Signatures". In: *ASIACRYPT*. 2019.

[51] C.-M. M. Chung, V. Hwang, M. J. Kannwischer, G. Seiler, C.-J. Shih, and B.-Y. Yang. *NTT Multiplication for NTT-unfriendly Rings*. Cryptology ePrint Archive, Report 2020/1397. 2020.

[52] D. Cooper, D. Apon, Q. Dang, M. Davidson, M. Dworkin, and C. Miller. *Recommendation for Stateful Hash-Based Signature Schemes*. 2019.

[53] D. Coppersmith. "Small Solutions to Polynomial Equations, and Low Exponent RSA Vulnerabilities". In: *Journal of Cryptology* 4 (1997).

[54] J.-S. Coron. "On the Exact Security of Full Domain Hash". In: *CRYPTO*. 2000.

[55] N. Courtois, M. Finiasz, and N. Sendrier. "How to Achieve a McEliece-Based Digital Signature Scheme". In: *ASIACRYPT*. 2001.

[56] N. Courtois, A. Klimov, J. Patarin, and A. Shamir. "Efficient Algorithms for Solving Overdefined Systems of Multivariate Polynomial Equations". In: *EUROCRYPT*. 2000.

[57] J.-M. Couveignes. *Hard Homogeneous Spaces*. Cryptology ePrint Archive, Report 2006/291. 2006.

[58] D. Dachman-Soled, L. Ducas, H. Gong, and M. Rossi. "LWE with Side Information: Attacks and Concrete Security Estimation". In: *CRYPTO*. 2020.

[59] I. Damgård. "On the Randomness of Legendre and Jacobi Sequences". In: *CRYPTO'88*. 1990.

[60] J.-P. D'Anvers, Q. Guo, T. Johansson, A. Nilsson, F. Vercauteren, and I. Verbauwhede. "Decryption Failure Attacks on IND-CCA Secure Lattice-Based Schemes". In: *PKC*. 2019.

[61] J.-P. D'Anvers, A. Karmakar, S. S. Roy, and F. Vercauteren. *SABER*. Tech. rep. National Institute of Standards and Technology, 2019.

[62] J.-P. D'Anvers, M. Rossi, and F. Virdia. "(One) Failure Is Not an Option: Bootstrapping the Search for Failures in Lattice-Based Encryption Schemes". In: *EUROCRYPT*. 2020.

[63] J. D'Anvers, M. Tiepelt, F. Vercauteren, and I. Verbauwhede. "Timing Attacks on Error Correcting Codes in Post-Quantum Schemes". In: *TIS@CCS*. 2019.

[64] J.-P. D'Anvers, F. Vercauteren, and I. Verbauwhede. "The Impact of Error Dependencies on Ring/Mod-LWE/LWR Based Schemes". In: *Post-Quantum Cryptography - 10th International Conference, PQCrypto*. 2019.

[65] L. De Feo and S. D. Galbraith. "SeaSign: Compact Isogeny Signatures from Class Group Actions". In: *EUROCRYPT*. 2019.

[66] L. De Feo, J. Kieffer, and B. Smith. "Towards Practical Key Exchange from Ordinary Isogeny Graphs". In: *ASIACRYPT*. 2018.

[67] C. de Saint Guilhem, L. De Meyer, E. Orsini, and N. P. Smart. "BBQ: Using AES in Picnic Signatures". In: *SAC*. 2019.

[68] T. Debris-Alazard, N. Sendrier, and J.-P. Tillich. "Wave: A New Family of Trapdoor One-Way Preimage Sampleable Functions Based on Codes". In: *ASIACRYPT*. 2019.

[69] T. Debris-Alazard and J.-P. Tillich. "Two Attacks on Rank Metric Code-Based Schemes: RankSign and an IBE Scheme". In: *ASIACRYPT*. 2018.

[70] J. P. Degabriele, A. Lehmann, K. G. Paterson, N. P. Smart, and M. Strefler. "On the Joint Security of Encryption and Signature in EMV". In: *CT-RSA*. 2012.

[71] C. Diem. "The XL-Algorithm and a Conjecture from Commutative Algebra". In: *ASIACRYPT*. 2004.

[72] W. Diffie and M. E. Hellman. "New Directions in Cryptography". In: *IEEE Transactions on Information Theory* 6 (1976).

[73] J. Ding, M.-S. Chen, A. Petzoldt, D. Schmidt, and B.-Y. Yang. *Rainbow*. Tech. rep. National Institute of Standards and Technology, 2019.

[74] J. Ding, X. Xie, and X. Lin. *A Simple Provably Secure Key Exchange Scheme Based on the Learning with Errors Problem*. Cryptology ePrint Archive, Report 2012/688. 2012.

[75] I. Dinur, D. Kales, A. Promitzer, S. Ramacher, and C. Rechberger. "Linear Equivalence of Block Ciphers with Partial Non-Linear Layers: Application to LowMC". In: *EUROCRYPT*. 2019.

[76] L. Ducas, V. Lyubashevsky, and T. Prest. "Efficient Identity-Based Encryption over NTRU Lattices". In: *ASIACRYPT*. 2014.

[77] L. Ducas and P. Q. Nguyen. "Learning a Zonotope and More: Cryptanalysis of NTRUSign Countermeasures". In: *ASIACRYPT*. 2012.

[78] T. ElGamal. "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms". In: *IEEE Transactions on Information Theory* (1985).

[79] T. Espitau, P.-A. Fouque, B. Gérard, and M. Tibouchi. "Side-Channel Attacks on BLISS Lattice-Based Signatures: Exploiting Branch Tracing against

strongSwan and Electromagnetic Emanations in Microcontrollers". In: *ACM CCS*. 2017.

[80] fail0verflow. "Console Hacking 2010: PS3 Epic Fail". In: *27th Chaos Communications Congress*. 2010.

[81] J. C. Faugère. "A New Efficient Algorithm for Computing GröBner Bases without Reduction to Zero (F5)". In: *ISSAC*. Lille, France, 2002. ISBN: 1581134843.

[82] J. Faugère, V. Gauthier-Umaña, A. Otmani, L. Perret, and J. Tillich. "A Distinguisher for High-Rate McEliece Cryptosystems". In: *IEEE Trans. Inf. Theory* 10 (2013).

[83] L. D. Feo. *Mathematics of Isogeny Based Cryptography*. 2017. arXiv: 1711.04062 [cs.CR].

[84] L. D. Feo, D. Jao, and J. Plût. "Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies". In: *Journal of Mathematical Cryptology* 3 (2014).

[85] A. Fiat and A. Shamir. "How to Prove Yourself: Practical Solutions to Identification and Signature Problems". In: *CRYPTO'86*. 1987.

[86] S. Fluhrer. *Cryptanalysis of ring-LWE based key exchange with key share reuse*. Cryptology ePrint Archive, Report 2016/085. 2016.

[87] G. Fumaroli, A. Martinelli, E. Prouff, and M. Rivain. "Affine Masking against Higher-Order Side Channel Analysis". In: *SAC*. 2011.

[88] P. Gaborit, O. Ruatta, J. Schrek, and G. Zémor. "RankSign: An Efficient Signature Algorithm Based on the Rank Metric". In: *Post-Quantum Cryptography - 6th International Workshop, PQCrypto*. 2014.

[89] S. Galbraith, L. Panny, B. Smith, and F. Vercauteren. *Quantum Equivalence of the DLP and CDHP for Group Actions*. Cryptology ePrint Archive, Report 2018/1199. 2018.

[90] S. D. Galbraith, C. Petit, B. Shani, and Y. B. Ti. "On the Security of Supersingular Isogeny Cryptosystems". In: *ASIACRYPT*. 2016.

[91] T. Gellersen, O. Seker, and T. Eisenbarth. *Differential Power Analysis of the Picnic Signature Scheme*. Cryptology ePrint Archive, Report 2020/267. 2020.

[92] C. Gentry, J. Jonsson, J. Stern, and M. Szydlo. "Cryptanalysis of the NTRU Signature Scheme (NSS) from Eurocrypt 2001". In: *ASIACRYPT*. 2001.

[93] C. Gentry, C. Peikert, and V. Vaikuntanathan. "Trapdoors for hard lattices and new cryptographic constructions". In: *40th ACM STOC*. 2008.

[94] O. Goldreich. "Two Remarks Concerning the Goldwasser-Micali-Rivest Signature Scheme". In: *CRYPTO'86*. 1987.

[95] O. Goldreich, S. Goldwasser, and S. Halevi. "Public-Key Cryptosystems from Lattice Reduction Problems". In: *CRYPTO'97*. 1997.

[96] L. Grassi, C. Rechberger, D. Rotaru, P. Scholl, and N. P. Smart. "MPC-Friendly Symmetric Key Primitives". In: *ACM CCS*. 2016.

[97] L. K. Grover. "A Fast Quantum Mechanical Algorithm for Database Search". In: *28th ACM STOC*. 1996.

[98] T. Güneysu, V. Lyubashevsky, and T. Pöppelmann. "Practical Lattice-Based Cryptography: A Signature Scheme for Embedded Systems". In: *CHES*. 2012.

[99] Q. Guo, T. Johansson, and A. Nilsson. "A Key-Recovery Timing Attack on Post-quantum Primitives Using the Fujisaki-Okamoto Transformation and Its Application on FrodoKEM". In: *CRYPTO*. 2020.

[100] Q. Guo, T. Johansson, and P. Stankovski. "A Key Recovery Attack on MDPC with CCA Security Using Decoding Errors". In: *ASIACRYPT*. 2016.

[101] Q. Guo, T. Johansson, and J. Yang. "A Novel CCA Attack Using Decryption Errors Against LAC". In: *ASIACRYPT*. 2019.

[102] S. Guo, P. Kamath, A. Rosen, and K. Sotiraki. "Limits on the Efficiency of (Ring) LWE Based Non-interactive Key Exchange". In: *PKC*. 2020.

[103] C. Hall, I. Goldberg, and B. Schneier. "Reaction Attacks against several Public-Key Cryptosystems". In: *ICICS 99*. 1999.

[104] M. Hamburg. *Three Bears*. Tech. rep. National Institute of Standards and Technology, 2019.

[105] J. Hoffstein, N. Howgrave-Graham, J. Pipher, J. H. Silverman, and W. Whyte. "NTRUSIGN: Digital Signatures Using the NTRU Lattice". In: *CT-RSA*. 2003.

[106] J. Hoffstein, J. Pipher, and J. H. Silverman. "NSS: An NTRU Lattice-Based Signature Scheme". In: *EUROCRYPT*. 2001.

[107] J. Hoffstein, J. Pipher, and J. H. Silverman. "NTRU: A Ring-Based Public Key Cryptosystem". In: *ANTS*. 1998. ISBN: 3-540-64657-4.

[108] D. Hofheinz, K. Hövelmanns, and E. Kiltz. "A Modular Analysis of the Fujisaki-Okamoto Transformation". In: *TCC*. 2017.

[109] J. Howe, M. Martinoli, E. Oswald, and F. Regazzoni. "Optimised Lattice-Based Key Encapsulation in Hardware". In: *NIST's Second PQC Standardization Conference* (2019).

[110] J. Howe, T. Prest, and D. Apon. *SoK: How (not) to Design and Implement Post-Quantum Cryptography*. Cryptology ePrint Archive, Report 2021. 2021.

[111] J. Howe, T. Prest, T. Ricosset, and M. Rossi. "Isochronous Gaussian Sampling: From Inception to Implementation". In: *Post-Quantum Cryptography - 11th International Conference, PQCrypto*. 2020.

[112] N. Howgrave-Graham, P. Q. Nguyen, D. Pointcheval, J. Proos, J. H. Silverman, A. Singer, and W. Whyte. "The Impact of Decryption Failures on the Security of NTRU Encryption". In: *CRYPTO*. 2003.

[113] W.-L. Huang, J.-P. Chen, and B.-Y. Yang. "Power Analysis on NTRU Prime". In: *IACR TCHES* 1 (2020).

[114] A. Hülsing. "W-OTS+ - Shorter Signatures for Hash-Based Signature Schemes". In: *AFRICACRYPT 13*. 2013.

[115] A. Hulsing, D. J. Bernstein, C. Dobraunig, M. Eichlseder, S. Fluhrer, S.-L. Gazdag, P. Kampanakis, S. Kolbl, T. Lange, M. M. Lauridsen, F. Mendel, R. Niederhagen, C. Rechberger, J. Rijneveld, P. Schwabe, and J.-P. Aumasson. *SPHINCS+*. Tech. rep. National Institute of Standards and Technology, 2019.

[116] Y. Ishai, E. Kushilevitz, R. Ostrovsky, and A. Sahai. "Zero-knowledge from secure multiparty computation". In: *39th ACM STOC*. 2007.

[117] D. Jao, R. Azarderakhsh, M. Campagna, C. Costello, L. De Feo, B. Hess, A. Jalali, B. Koziel, B. LaMacchia, P. Longa, M. Naehrig, J. Renes, V. Soukharev, D. Urbanik, and G. Pereira. *SIKE*. Tech. rep. National Institute of Standards and Technology, 2019.

[118] D. Jao and L. De Feo. "Towards Quantum-Resistant Cryptosystems from Supersingular Elliptic Curve Isogenies". In: *Post-Quantum Cryptography - 4th International Workshop, PQCrypto*. 2011.

[119] S. Jaques, M. Naehrig, M. Roetteler, and F. Virdia. "Implementing Grover Oracles for Quantum Key Search on AES and LowMC". In: *EUROCRYPT*. 2020.

[120] S. Jaques and J. M. Schanck. "Quantum Cryptanalysis in the RAM Model: Claw-Finding Attacks on SIKE". In: *CRYPTO*. 2019.

[121] A. Joux and V. Vitse. *A crossbred algorithm for solving Boolean polynomial systems*. Cryptology ePrint Archive, Report 2017/372. 2017.
[122] D. Kales and G. Zaverucha. *An Attack on Some Signature Schemes Constructed From Five-Pass Identification Schemes*. Cryptology ePrint Archive, Report 2020/837. 2020.
[123] M. J. Kannwischer, P. Pessl, and R. Primas. "Single-Trace Attacks on Keccak". In: *IACR TCHES* 3 (2020).
[124] J. Katz, V. Kolesnikov, and X. Wang. "Improved Non-Interactive Zero Knowledge with Applications to Post-Quantum Signatures". In: *ACM CCS*. 2018.
[125] G. Kuperberg. "A Subexponential-Time Quantum Algorithm for the Dihedral Hidden Subgroup Problem". In: *SIAM J. Comput.* 1 (2005).
[126] G. Kuperberg. "Another Subexponential-time Quantum Algorithm for the Dihedral Hidden Subgroup Problem". In: *TQC*. 2013.
[127] T. Laarhoven, M. Mosca, and J. van de Pol. "Finding shortest lattice vectors faster using quantum search". In: *Des. Codes Cryptogr.* 2-3 (2015).
[128] N. Lahr, R. Niederhagen, R. Petri, and S. Samardjiska. "Side Channel Information Set Decoding Using Iterative Chunking - Plaintext Recovery from the "Classic McEliece" Hardware Reference Implementation". In: *ASIACRYPT*. 2020.
[129] L. Lamport. *Constructing Digital Signatures from a One-way Function*. Technical Report SRI-CSL-98. SRI International Computer Science Laboratory, 1979.
[130] A. Langlois and D. Stehlé. "Worst-case to average-case reductions for module lattices". In: *Des. Codes Cryptogr.* 3 (2015).
[131] F. Liu, T. Isobe, and W. Meier. *Cryptanalysis of Full LowMC and LowMC-M with Algebraic Techniques*. Cryptology ePrint Archive, Report 2020/1034. 2020.
[132] V. Lyubashevsky. "Fiat-Shamir with Aborts: Applications to Lattice and Factoring-Based Signatures". In: *ASIACRYPT*. 2009.
[133] V. Lyubashevsky, L. Ducas, E. Kiltz, T. Lepoint, P. Schwabe, G. Seiler, and D. Stehlé. *CRYSTALS-DILITHIUM*. Tech. rep. National Institute of Standards and Technology, 2019.
[134] V. Lyubashevsky, C. Peikert, and O. Regev. "On Ideal Lattices and Learning with Errors over Rings". In: *EUROCRYPT*. 2010.
[135] R. J. McEliece. "A Public-Key Cryptosystem Based on Algebraic Coding Theory". In: *JPL DSN Progress Report* (1978).
[136] R. C. Merkle. "A Certified Digital Signature". In: *CRYPTO'89*. 1990.
[137] D. Micciancio and C. Peikert. "Trapdoors for Lattices: Simpler, Tighter, Faster, Smaller". In: *EUROCRYPT*. 2012.
[138] V. Migliore, B. Gérard, M. Tibouchi, and P.-A. Fouque. "Masking Dilithium - Efficient Implementation and Side-Channel Evaluation". In: *ACNS 19*. 2019.
[139] K. Mus, S. Islam, and B. Sunar. "QuantumHammer: A Practical Hybrid Attack on the LUOV Signature Scheme". In: *ACM CCS 20*. 2020.
[140] M. Naehrig, E. Alkim, J. Bos, L. Ducas, K. Easterbrook, B. LaMacchia, P. Longa, I. Mironov, V. Nikolaenko, C. Peikert, A. Raghunathan, and D. Stebila. *FrodoKEM*. Tech. rep. National Institute of Standards and Technology, 2019.
[141] P. Q. Nguyen and O. Regev. "Learning a Parallelepiped: Cryptanalysis of GGH and NTRU Signatures". In: *EUROCRYPT*. 2006.
[142] NIST. *Submission Requirements and Evaluation Criteria for the Post-Quantum Cryptography Standardization Process*. 2016.
[143] E. Oswald and S. Mangard. "Template Attacks on Masking - Resistance Is Futile". In: *CT-RSA*. 2007.

[144] A. Park, K.-A. Shim, N. Koo, and D.-G. Han. "Side-Channel Attacks on Post-Quantum Signature Schemes based on Multivariate Quadratic Equations". In: *IACR TCHES* 3 (2018).
[145] K. G. Paterson and R. Villanueva-Polanco. "Cold Boot Attacks on NTRU". In: *INDOCRYPT*. 2017.
[146] C. Peikert. *A Decade of Lattice Cryptography*. Cryptology ePrint Archive, Report 2015/939. 2015.
[147] C. Peikert. "He Gives C-Sieves on the CSIDH". In: *EUROCRYPT*. 2020.
[148] C. Peikert. "How (Not) to Instantiate Ring-LWE". In: *SCN 16*. 2016.
[149] C. Peikert. "Lattice Cryptography for the Internet". In: *Post-Quantum Cryptography - 6th International Workshop, PQCrypto*. 2014.
[150] C. Peikert and Z. Pepin. "Algebraically Structured LWE, Revisited". In: *TCC*. 2019.
[151] A. Perrig. "The BiBa One-Time Signature and Broadcast Authentication Protocol". In: *ACM CCS*. 2001.
[152] P. Pessl, L. G. Bruinderink, and Y. Yarom. "To BLISS-B or not to be: Attacking strongSwan's Implementation of Post-Quantum Signatures". In: *ACM CCS*. 2017.
[153] D. Pointcheval and J. Stern. "Security Proofs for Signature Schemes". In: *EUROCRYPT'96*. 1996.
[154] R. L. V. Polanco. "Cold Boot Attacks on Post-Quantum Schemes". PhD thesis. Royal Holloway, University of London, 2018.
[155] E. Prange. "The use of information sets in decoding cyclic codes". In: *IRE Trans. Inf. Theory* 5 (1962).
[156] T. Prest, P.-A. Fouque, J. Hoffstein, P. Kirchner, V. Lyubashevsky, T. Pornin, T. Ricosset, G. Seiler, W. Whyte, and Z. Zhang. *FALCON*. Tech. rep. National Institute of Standards and Technology, 2019.
[157] P. Ravi, M. P. Jhanwar, J. Howe, A. Chattopadhyay, and S. Bhasin. "Exploiting determinism in lattice-based signatures: practical fault attacks on pqm4 implementations of NIST candidates". In: *AsiaCCS*. 2019.
[158] O. Regev. "On lattices, learning with errors, random linear codes, and cryptography". In: *37th ACM STOC*. 2005.
[159] L. Reyzin and N. Reyzin. "Better than BiBa: Short one-time signatures with fast signing and verifying". In: *Information Security and Privacy 2002*. 2002.
[160] R. L. Rivest, A. Shamir, and L. M. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In: *Communications of the Association for Computing Machinery* 2 (1978).
[161] M. Rossi, M. Hamburg, M. Hutter, and M. E. Marson. "A Side-Channel Assisted Cryptanalytic Attack Against QcBits". In: *CHES*. 2017.
[162] A. Rostovtsev and A. Stolbunov. *Public-Key Cryptosystem Based On Isogenies*. Cryptology ePrint Archive, Report 2006/145. 2006.
[163] K. Sakumoto, T. Shirai, and H. Hiwatari. "On Provable Security of UOV and HFE Signature Schemes against Chosen-Message Attack". In: *Post-Quantum Cryptography - 4th International Workshop, PQCrypto*. 2011.
[164] K. Sakumoto, T. Shirai, and H. Hiwatari. "Public-Key Identification Schemes Based on Multivariate Quadratic Polynomials". In: *CRYPTO*. 2011.
[165] S. Samardjiska, M.-S. Chen, A. Hulsing, J. Rijneveld, and P. Schwabe. *MQDSS*. Tech. rep. National Institute of Standards and Technology, 2019.
[166] S. Samardjiska, P. Santini, E. Persichetti, and G. Banegas. "A Reaction Attack Against Cryptosystems Based on LRPC Codes". In: *LATINCRYPT*. 2019.

[167] N. Samwel, L. Batina, G. Bertoni, J. Daemen, and R. Susella. "Breaking Ed25519 in WolfSSL". In: *CT-RSA*. 2018.

[168] C.-P. Schnorr. "Efficient Identification and Signatures for Smart Cards". In: *CRYPTO'89*. 1990.

[169] C. Schnorr and M. Euchner. "Lattice basis reduction: Improved practical algorithms and solving subset sum problems". In: *Math. Program.* (1994).

[170] P. Schwabe, R. Avanzi, J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, G. Seiler, and D. Stehlé. *CRYSTALS-KYBER*. Tech. rep. National Institute of Standards and Technology, 2019.

[171] A. Shamir. "An Efficient Identification Scheme Based on Permuted Kernels (Extended Abstract) (Rump Session)". In: *CRYPTO'89*. 1990.

[172] P. W. Shor. "Algorithms for Quantum Computation: Discrete Logarithms and Factoring". In: *35th FOCS*. 1994.

[173] B.-Y. Sim, J. Kwon, K. Y. Choi, J. Cho, A. Park, and D.-G. Han. "Novel Side-Channel Attacks on Quasi-Cyclic Code-Based Cryptography". In: *IACR TCHES* 4 (2019).

[174] J. Stern. "A New Identification Scheme Based on Syndrome Decoding". In: *CRYPTO'93*. 1994.

[175] J. Stern. "A new paradigm for public key identification". In: *IEEE Trans. Inf. Theory* 6 (1996).

[176] M. Szydlo. "Merkle Tree Traversal in Log Space and Time". In: *EUROCRYPT*. 2004.

[177] R. C. Torres and N. Sendrier. "Analysis of Information Set Decoding for a Sub-linear Error Weight". In: *Post-Quantum Cryptography - 7th International Workshop, PQCrypto*. 2016.

[178] P. C. van Oorschot and M. J. Wiener. "Parallel Collision Search with Cryptanalytic Applications". In: *Journal of Cryptology* 1 (1999).

[179] K. Verhulst. "Power Analysis and Masking of Saber". MA thesis. Belgium: KU Leuven, 2019.

[180] P. Véron. "Improved identification schemes based on error-correcting codes". In: *Appl. Algebra Eng. Commun. Comput.* 1 (1996).

[181] B. Yang and J. Chen. "All in the XL Family: Theory and Practice". In: *ICISC*. 2004.

[182] Y. Yarom and K. Falkner. "FLUSH+RELOAD: A High Resolution, Low Noise, L3 Cache Side-Channel Attack". In: *USENIX Security*. 2014.

[183] G. Zaverucha, M. Chase, D. Derler, S. Goldfeder, C. Orlandi, S. Ramacher, C. Rechberger, D. Slamanig, J. Katz, X. Wang, and V. Kolesnikov. *Picnic*. Tech. rep. National Institute of Standards and Technology, 2019.

[184] M. Zhandry. "A note on the quantum collision and set equality problems". In: *Quantum Inf. Comput.* 7&8 (2015).

[185] Z. Zhang, C. Chen, J. Hoffstein, W. Whyte, J. M. Schanck, A. Hulsing, J. Rijneveld, P. Schwabe, and O. Danba. *NTRUEncrypt*. Tech. rep. National Institute of Standards and Technology, 2019.