Quantifying risks in cryptographic selection processes

Daniel J. Bernstein[1,2]

[1] Department of Computer Science, University of Illinois at Chicago, USA
[2] Horst Görtz Institute for IT Security, Ruhr University Bochum, Germany
djb@cr.yp.to

Abstract. There appears to be a widespread belief that some processes of selecting cryptosystems are less risky than other processes. As a case study of quantifying the difference in risks, this paper compares the currently-known-failure rates of three large groups of cryptosystems: (1) the round-1 submissions to the NIST Post-Quantum Cryptography Standardization Project, (2) the round-1 submissions not broken by the end of round 1, and (3) the round-1 submissions selected by NIST for round 2 of the same project. These groups of cryptosystems turn out to have currently-known-failure rates that are strikingly high, and that include statistically significant differences across the groups, not matching the pattern of differences that one might expect. Readers are cautioned that the actual failure rates could be much higher than the currently-known-failure rates.

Keywords: cryptographic risk analysis, post-quantum cryptography, cryptosystem selection, standardization

1 Introduction

To study the question of how *safely* user data is encrypted, one needs to start by asking *how* the data is encrypted.

A typical paper quantifying the security level of encryption answers this preliminary question by picking a cryptosystem X and postulating that the encryption process uses cryptosystem X. The paper then studies the security level of that encryption process, meaning the tradeoff between the attacker's success probability and the attacker's resources. Often the tradeoff curve is

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) as part of the Excellence Strategy of the German Federal and State Governments—EXC 2092 CASA—390781972 "Cyber Security in the Age of Large-Scale Adversaries"; by the U.S. National Science Foundation under grant 1913167; and by the Taiwan's Executive Yuan Data Safety and Talent Cultivation Project (AS-KPQ-109-DSTCP). "Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation" (or other funding agencies). Permanent ID of this document: de2ef124dae60c9c0fc77d61266044ddb319f692. Date: 2023.11.24.

simplified down to a single number, a ratio between cost and probability. Often a target is established for this number, and the paper simply asks whether X succeeds or fails at reaching this target.

This paper instead considers a more realistic model of how data is encrypted:

- There are multiple cryptosystem specifications $X_1, X_2, \ldots$.
- Data is encrypted with cryptosystem X_i for some distribution of indices i.
- This distribution, in turn, is the output $S(X_1, X_2, \ldots)$ of a selection process S that is given the specifications $X_1, X_2, \ldots$ as input.

In other words, the encryption process is modeled as having three stages—first run S to obtain a distribution of indices, then pick an index i from that distribution, then encrypt using X_i—rather than having just one stage of encrypting using a prespecified X.

This overall process again has a security level, a tradeoff between the attacker's success probability and the attacker's resources. One can view this as a combination of the security levels of the component cryptosystems X_i. One can also define the failure rate of the overall process as the chance of picking i such that X_i has security level below a specified target.

Compared to the one-stage model, the three-stage model has the disadvantage of being more complicated, but has the advantage of allowing analysis of the impact of the selection process S upon security. One can, for example, ask about the impact upon failure rates if one selects the fastest cryptosystems (which might vary from one environment to another, so the distribution output by S can still include multiple values of i), or if one selects the cryptosystem specifications that have the maximum number of theorems. "Specification" here is meant in a broad sense: S can inspect theorems and proofs and attacks and other documentation available for the cryptosystem, not just algorithms for key generation and encryption and decryption.

As a case study of quantifying differences in risks of cryptographic selection processes, this paper quantifies and compares the currently-known-failure rates of three different selection processes applied to the same large pool of cryptosystem specifications (mostly encryption systems, but also some signature systems). Section 2 explains how this case study was chosen. Section 3 explains the criteria for labeling cryptosystems as broken. Section 4 collects data regarding which cryptosystems from the pool are broken. Section 5 tallies which cryptosystems are selected by each of the three selection processes. Section 6 presents the currently-known-failure rate of each process. Section 7 compares the results to various hypotheses, in particular drawing the following conclusions:

- The currently-known-failure rates for all three selection processes are higher than one might expect. Readers are cautioned that the actual failure rates could be much higher than the currently-known-failure rates.
- The currently-known-failure rates have statistically significant differences across the three selection processes.
- These differences do not match the pattern that that one might expect.

These results suggest that further analysis and optimization of selection processes will significantly improve the cryptographic community's management of security risks.

1.1. Related work. [16, Appendix B] says "Even when the standardization process is not under attack, one can ask how reliable it is at producing secure standards. ... It would be useful to refine the concept of a thorough security review, formulating more detailed definitions of review processes and then collecting evidence regarding the effectiveness of different processes at reducing security risks." This paper follows up on that idea, and collects such evidence for three different cryptosystem-selection processes.

The first selection process considered in this paper is a baseline for which the high failure rate was already pointed out in, e.g., [28, PDF page 62, "Do cryptographers have any idea what they're doing?"]. However, the literature does not seem to have compared that failure rate to the failure rates of other selection processes. This paper also takes initial steps in analyzing security levels beyond failure rates.

[18, Section 3.2] proposes a model in which each cryptosystem has "probability $p(M)$ of being publicly broken within M months"; uses data from 15 submissions to the AES competition to estimate $p(12) \approx 1/3$; proposes a refined model, split between $p_1(M)$ for faster cryptosystems and $p_2(M)$ for slower cryptosystems; and evaluates the long-term failure rates of two model cryptosystem-selection processes in terms of p, p_1, p_2. This paper's case study (see Section 2) can be viewed as collecting data regarding p for a much larger pool of cryptosystems, including another split of cryptosystems that appears more important than "faster" vs. "slower".

Another model appears in [37]. In this model, every cryptosystem is breakable (so $p(\infty) = 1$ in the notation of [18]), and the chance of finding a break converges exponentially to 100% with the amount of time spent searching for an attack algorithm; see [37, Section 3.1]. This model allows each cryptosystem to have its own base for the exponential, so any broken cryptosystem can be taken as confirming the model, and any unbroken cryptosystem can be taken as not enough time having been spent yet to break that particular cryptosystem; it is not obvious how this model could be extended to make testable predictions.

It is easy to find literature suggesting that cryptographic competitions reduce security risks—see, e.g., [46] claiming that "competitions can focus the attention of cryptographers around the world"—but [18, Section 3.3] lists various mechanisms by which competitions could decrease *or increase* security risks. There do not appear to have been any quantitative studies of these effects. This paper studies a competition, but does not compare the competition process to non-competition processes.

It is also easy to find literature suggesting that "provable security" reduces security risks. For example, [47]

- mentions a chosen-ciphertext attack by Bleichenbacher that broke "an older version of the PKCS#1 standard";

- says that "a security reduction" from RSA "would have excluded attacks such as Bleichenbachers";
- claims that "from experience, design errors usually materialize exactly as attacks that circumvent the hard problem we were trying to use"; and
- concludes that "the only reasonable approach is to construct cryptographic systems with the objective of being able to give security reductions for them".

However, no evidence is provided in [47] for the "usually" claim beyond isolated anecdotes. Evidently [47] recommends selecting cryptographic systems with "security reductions", but studying the security impact of this recommendation (positive, for reasons explained in [47]? negative, for reasons missed in [47]? statistically insignificant?) would require clarity regarding the selection process:

- Are the reductions required to be from RSA? This covers only a small corner of cryptography, and is content-free in post-quantum cryptography; presumably [47] meant something broader.
- Or are the reductions allowed to be from any "hard problem"? This begs the question of how the selection process will decide which problems are "hard".
- Or is *any* "security reduction" allowed? This would allow a content-free proof saying that X is as secure as X, and a content-free requirement cannot control security risks, so presumably [47] meant something narrower. See [16, Appendix C] for a survey of different proposals for addressing content-free proofs.

In the opposite direction, [66] surveys many failures of cryptosystems identified as "provably secure". However, no data is collected in [66] regarding the number of failures of other cryptosystems, nor is there baseline information regarding the number of cryptosystems of each type.

Analysis of the reliability of cryptographic selection processes should not be confused with analysis of how vulnerable those processes are to sabotage. For the latter type of analysis, see [100], [26], [29], [56], [17], and [18, Sections 3.6–3.7].

2 Choosing a case study

This section explains this paper's choice of (1) a pool of cryptosystems to study and (2) selection processes to compare for that pool.

2.1. Taking a high-profile case study. A well-known risk in choosing topics to study is "p-hacking", in which one (intentionally or unintentionally) biases results by studying many choices and publishing only the choices that produce interesting results.

This paper takes two publicly verifiable steps to reduce this risk. First, this paper chooses a high-profile pool of cryptosystem specifications: namely, all 69 specifications that NIST posted in 2017 as complete submissions to the NIST Post-Quantum Cryptography Standardization Project. For comparison, taking a more artificial pool would allow more "wiggle room".

Second, this paper considers only high-profile selection processes that, as a historical matter, were applied to those specifications. Specifically, the highlight of this paper is studying NIST's January 2019 selection of some of the 69 specifications as "round 2" submissions. As two baselines for comparison, this paper considers (1) NIST's initial process of taking all 69 specifications and (2) the obvious intermediate process applied by various commentators (e.g., [96]), namely eliminating the specifications of cryptosystems that were publicly broken by the end of "round 1" of the project.

This paper uses the following notation for these three selection processes:

- S_0 is the baseline of taking all 69 specifications.
- S_1 is taking the subset not publicly broken during round 1.
- S_2 is taking the subset selected by NIST for round 2.

Note that these selection processes are defined as actual events that took place. The submitters did something to create the 69 initial specifications (plus some further specifications that NIST reportedly did not post); cryptanalysts did something to break 20 of those specifications during round 1; and NIST did something to select specifications for round 2. For evaluating the currently-known-failure rates of these processes, it is enough to see which cryptosystems were selected by these processes and how those cryptosystems fared against subsequently published attacks.

Section 7 considers various hypotheses for *why* the processes failed at the rates that they did. Investigating these hypotheses would rely on more information about what happened within the processes. NIST has released a report [2] and talk slides [80] with some small bits of information about NIST's process. See [23] for further information that has been released as a result of ongoing court proceedings under the Freedom of Information Act (FOIA).

2.2. Why not also round 3 and round 4? It would also be possible to study NIST's round-3 selections (perhaps filtered into "finalists" and "alternates") and round-4 selections (perhaps filtered into "selected for standardization" and "selected for further consideration"), along with the lists of submissions that were unbroken at the same moments. Here are two factors counseling against such a study:

- Narrower lists of selections are more vulnerable to randomness, making them less statistically meaningful as information about the risks of selection processes. For example, it is not easy to extract information out of the fact that NIST selected SIKE for round 4 a few weeks before SIKE was broken. There were very few round-4 selections, so the number of broken selections leaves a very wide confidence interval (at, e.g., 95% confidence) for the failure rate of NIST's selection process.
- Any selection process that skips publicly broken algorithms will, at the exact moment of selection, *appear* to have a perfect success rate, even if some of the algorithms are actually breakable. This bias (unlike survival bias) can reasonably be expected to dissipate, but the dissipation takes time: one needs to see what further attacks are developed.

For understanding differences in risk between NIST's process and simply eliminating specifications that are already broken, it is useful to have more data points and more time to find attacks. Both of these considerations say that studying NIST's 2019 round-2 selection is more useful than studying NIST's narrower, more recent selections.

2.3. Avoiding overfitting. There is a risk of "overfitting" when the study of selection processes S is used to optimize S.

One standard way to reduce this risk is to split the available data into some data used for training and some data used for evaluation. This does not address the risk of overfitting in the choice of training mechanism.

Overfitting should not be a concern for this paper given the aforementioned constraints on the set of S considered, but followup work exploring unconstrained optimization of S would need to take steps to prevent overfitting.

2.4. The distribution of cryptosystems and parameter sets. NIST's selection of round-2 submissions did not specify any particular distribution across those submissions. Furthermore, each cryptosystem had a list of parameter sets; NIST did not specify any particular distribution across those parameter sets.

For simplicity, this paper defines each of the three selection processes S to output a uniform distribution across the selected submissions, and specifically to take the smallest parameter set for each submission (not counting clearly labeled toy parameters). This paper considers only the original 2017 submissions without any subsequent "tweaks", and considers only the specified cryptosystems without any errors introduced by software.

These choices certainly influence the results. Other choices would also be interesting to study: for example, asking how failure rates would be affected if one instead takes the largest parameter set for each submission, or (as in [18, Section 3.1]) the largest parameter set for each submission subject to a specified performance constraint, skipping submissions that have no parameter sets meeting the constraint. Such a study would, however, have to take extra steps to reduce the risk of p-hacking as in Section 2.1, the risk of noise as in Section 2.2, and the risk of overfitting as in Section 2.3.

3 Measuring failure rates

Within Section 2's pool of cryptosystems, namely the 69 cryptosystems in round 1 of the NIST post-quantum competition, Section 4 classifies some of the cryptosystems as "broken". This section explains the classification rules. The failure rate of a selection process S is then defined as the probability that X_i is broken when i is chosen from the distribution output by S.

3.1. The security target. NIST's call for submissions specified the cost of AES-128 key search as the minimum security level allowed in the NIST competition:

> Each category will be defined by a comparatively easy-to-analyze reference primitive, whose security will serve as a floor for a wide variety of metrics that NIST deems potentially relevant to practical security. ... Any attack that breaks the relevant security definition must require computational resources comparable to or greater than those required for key search on a block cipher with a 128-bit key (e.g. AES128) ... At the present time, NIST would give the following estimates for the classical and quantum gate counts for the optimal key recovery and collision attacks on AES and SHA3, respectively ... [for "AES 128":] 2^{170}/MAXDEPTH quantum gates or 2^{143} classical gates

Each submission was required to specify a targeted security property: e.g., existential unforgeability for a signature system.

NIST did not say how to handle low-probability attacks. This paper follows the literature in considering the cost/probability ratio.

3.2. Focusing on currently known failures. A cryptosystem X fails to meet the NIST "floor" quoted in Section 3.1 if breaking X's targeted security property is easier than AES-128 key search. There are various difficulties in evaluating whether this is the case: see Sections 3.3–3.5.

This paper adopts the following policy: if it is not clear whether cryptography is safe or unsafe, evaluate it as being safe. In particular, this paper focuses on *currently known* failures. "Broken" means that X is *known* to be easier to break than AES-128 key search. If it is not clear that X is broken then X is declared to *not* be broken.

Obviously this policy risks misleading cryptographers and users into thinking that cryptography is safer than it actually is. This paper takes two steps to counteract this risk. First, the paper systematically describes its results as evaluating "currently-known-failure rates". Second, the paper prominently cautions readers that the actual failure rates could be much higher than the currently-known-failure rates.

Rather than focusing on currently known failures, one could *extrapolate* from known failures. See, e.g., [**20**]:

> Let's say you've seen some examples of the terrible track record of public-key encryption systems that portray non-commutativity as a security improvement: ... You then see another public-key encryption system advertising non-commutativity as a security improvement. What's the chance that there's a real security gain this time? What's the chance that there's a security loss, where the attacks were simply obscured by the complications of non-commutativity? How much did I bias the answers to these questions by using the phrase "terrible track record"?

For further examples of extrapolation, see the "provable security" citations in Section 1.1.

Compared to focusing on currently known failures, extrapolation can reduce security risks by correctly predicting failures and thus discouraging use of

insecure systems, but it can also increase security risks by incorrectly predicting failures and thus discouraging use of secure systems. Presumably the error rates here depend heavily on details of the extrapolation mechanism.

One could study these effects within this paper's framework by defining selection mechanisms that use various forms of extrapolation and then evaluating the rates of failures known after N years. If such a study shows an extrapolation mechanism having better predictive value than simply looking at currently known failures, it would be easy to justify a followup study of the currently extrapolated failure rates of various selection mechanisms. This is an interesting direction for future work, but is also more complicated than this paper's policy of focusing on currently known failures. This direction also exacerbates the concerns from Section 2 regarding p-hacking, overfitting, etc.

3.3. Focusing on currently known attack algorithms. The most obvious source of difficulty in evaluating the security level of X is that *known* attacks against X can be worse, perhaps much worse, than the *best* attacks against X. Cryptography has a long history of attack algorithms being superseded by better attack algorithms. Often this has crossed the line between security level above the target and security level below the target (see, e.g., Section 4), and presumably there will be many future examples.

Nevertheless, as per the policy from Section 3.2, this paper considers only known attacks. Also, recall from Section 2.4 that this paper considers only the smallest proposed parameter set for each cryptosystem. Consequently, in this paper, saying that X is "broken" means that the security property targeted by X was publicly shown to be easier to break than AES-128 for the smallest proposed parameters for X.

3.4. Handling algorithm-analysis issues. Another source of difficulty in evaluating the security level of a cryptosystem X is that known attack algorithms against X can be inherently complicated to analyze.

For example, regarding the latest version of Kyber-512, the latest Kyber documentation [**10**, page 27] says that its "preliminary analysis gives a cost of 2^{151} gates, which is a 2^8 factor margin over the targeted security of the 2^{143} gates required for attacks against AES. Our discussion of the 'known unknowns' conclude that this number could be affected by a factor of up to 2^{16} in either direction". More precisely, [**10**, pages 29–30] says that because of "further refinements of the analysis of various aspect of sieving, as well as some foreseeable algorithmic improvements, the estimates may move by a factor somewhere between 2^{-16} and 2^{14}".

This is not claiming that Kyber-512 requires more "gates" to break than AES-128: it is explicitly allowing for the opposite possibility. There have also been various papers since then on advances in lattice attacks, including algorithmic improvements not foreseen in [**10**]. The simplest is [**25**, Appendix D], which very easily cuts almost 10 bits out of the "gate" count for the attacks considered in [**10**]. The uncertainty range in the Kyber documentation has not been updated.

Pinning down the actual costs of attacks is outside the scope of this paper; this paper merely looks at what the literature *says* about the costs of attacks. When the literature does not make clear whether the cost of a known attack is above or below the cost of breaking AES-128, this paper assumes above, as per the policy from Section 3.2: the cryptosystem is not *known* to be broken. Readers are cautioned that the cryptosystem could, in fact, be breakable.

3.5. Handling cost-metric issues. Another source of difficulty is that the cost of an attack depends on the choice of cost metric.

NIST wrote that "any attack must require computational resources comparable to or greater than the stated threshold, with respect to *all* metrics that NIST deems to be potentially relevant to practical security". This begs the question of which metrics those are.

NIST did not define the allowed set of "gates" for its estimate that an "optimal" AES-128 key-recovery attack would use "2^{143} classical gates". The literature defines many different gate sets. Some of these variations have only minor effects on security levels, but some have much larger effects and can easily reverse comparisons: for example, allowing a memory-access "gate" can change the cost of high-memory attacks by many bits, while having relatively little effect on AES-128. Here is a numerical example of how important the choice of cost metrics can be:

- NIST's August 2016 draft call for submissions [**83**, page 15] estimated the hardness of "collision attacks against SHA-256/ SHA3-256" as "128 bits classical security / 80 bits quantum security".
- NIST's December 2016 final call for submissions [**84**, pages 17–18] estimated the hardness of "collision search on a 256-bit hash function (e.g. SHA256/ SHA3-256)" as "2^{146} classical gates" with no quantum speedup.

The numbers 80, 128, and 146 are on strikingly different scales. See [**17**, Section 5.4] for an explanation of how different assessments of collision cost come from different cost metrics.

Announcements by NIST in late 2022 specifically regarding Kyber-512 appear to have abandoned such "gate" counts in favor of somehow accounting for memory-access costs. NIST's accounting mechanism is even less clearly defined than NIST's "gate" set, but has the striking features of (1) multiplying the "real cost of memory access" by a tally of bit operations (as if every bit operation incurred a memory access) and (2) not asking for attacks to be re-optimized for this accounting mechanism (never mind the question of how one could do that without a definition). See generally [**21**] and [**22**].

For some cryptosystems X, the lack of a stable definition of NIST's cost metrics produces a lack of clarity as to whether X is broken by known attacks. This paper, following the policy from Section 3.2, then lists X as *not* currently known to be broken. Readers are again cautioned that cryptosystems not currently known to be broken could, in fact, be breakable.

4 Collecting security levels of the cryptosystems

This section reviews what the literature says about the security levels of the 69 round-1 submissions to the NIST competition. For this paper's case study, it suffices to track which submissions are broken, and which submissions were already broken by the end of round 1.

To simplify data collection, this paper does not track security levels of submissions that are not claimed in the literature to be broken. However, this paper *does* track the estimated costs of breaks. Evaluating whether attacks qualify as breaks requires inspecting what is reported about costs (except when breaks are demonstrated), and seeing the spread of numbers gives an idea of the extent to which this paper's results would be affected by changes in the numbers.

4.1. Lattice attacks. [5] reported pre-quantum "Core-SVP" (0.292β) levels of 76 for (the smallest parameter set for) EMBLEM, 84 for Round2, 93 for NTRU, 99 for Ding, 101 for Dilithium, 113 for Kyber, 113 for NewHope, 125 for SABER, 136 for LIMA, 136 for NTRU-HRSS, 139 for Lizard, 141 for Frodo, 141 for Falcon, 149 for LAC, 155 for NTRU Prime, 156 for LOTUS, 157 for ThreeBears, etc.

"Core-SVP" is a rough estimate of the number of iterations in a generic lattice attack. As noted in Section 3.4, the latest Kyber documentation [10] says that the number of "gates" to attack the latest version of Kyber-512 is somewhere between 2^{135} and 2^{165}. For comparison, the documentation says that Kyber-512 has "Core-SVP" 2^{118}.

NIST's new accounting for memory-access costs (see Section 3.5) then adds, in NIST's words, "something like 20 to 40 bits of security more". If "something like" is understood as allowing anything between 15 and 45 then the resulting attack cost is between 2^{150} and 2^{210}.

Under the questionable assumption that these exponents scale down linearly with Core-SVP, the same range is between 2^{96} and 2^{136} for EMBLEM, between 2^{106} and 2^{150} for Round2, between 2^{118} and 2^{166} for NTRU, etc.

For NTRU (and for all of the larger lattice systems), the high end of this range is well above 2^{143}. The literature after [10] does not make clear whether advances in generic lattice attacks have reduced the high end below 2^{143}. As per the policy of focusing on *currently known* failures, this paper classifies NTRU and above as not being broken by generic lattice attacks. This is not consistent with NIST in [3] discarding a larger NTRU parameter set, but this paper's evaluation of brokenness prioritizes consistency with NIST's handling of Kyber-512.

This paper classifies Round2 as broken because of a different, simpler, Round2-specific attack published after round 1. Since 2^{150} is above 2^{143} and there were no Round2-specific attacks during round 1, this paper classifies Round2 as not broken during round 1. [59] during round 1 asked whether Round2 was misinterpreting NIST's requested security levels, but this is not the same as claiming a clear break.

This paper *does* classify EMBLEM as broken during round 1, since 2^{136} is below 2^{143}. It is conceivable that this comparison would be reversed by a more

precise calculation of scaling factors, by [10] relying on post-round-1 speedups (although "dimensions for free" [54] was published during round 1), or by NIST's "something like" allowing an even wider range.

Readers are cautioned that known lattice attacks could have already broken many more submissions than EMBLEM, and that improved lattice attacks would be unsurprising given that there have been advances in lattice attacks in 2018, 2019, 2020, 2021, 2022, and 2023. However, because of very wide uncertainty intervals regarding the costs of known attacks, those attacks are not *known* to have broken many more submissions.

4.2. Submissions broken in round 1. This paper classifies the following cryptosystems as being broken during round 1 because of the following attacks:

- CFPKM: Albrecht, Postlethwaite, and Virdia [6] demonstrated attacks for the "128" parameter sets. Steinfeld [98] independently demonstrated attacks, saying they were "faster than the run-time of the legitimate ... reference implementation".
- Compact LWE: Li, Liu, Pan, and Xie [71] pointed out attacks. Bootle, Tibouchi, and Xagawa [36] demonstrated attacks.
- DAGS: Barelli and Couvreur [13] demonstrated a break of "DAGS_1, claimed 128 bits security key, in 20 minutes".
- DME: Beullens [30] estimated attacks as having "complexity" 2^{81} for parameters larger than the original DME parameters. Various newer versions of DME have also been broken; perhaps the attacks reduce the security levels of the original version of DME. For simplicity, this paper takes 81 as an estimate for the security level of the original DME parameters.
- DRS: Ducas and Yu [55] (conference version: Yu and Ducas) estimated attacks to be "below 80-bits (maybe even 70-bits)".
- Edon-K: Lequesne and Tillich [70] demonstrated an attack recovering "the secret in less than a minute".
- EMBLEM/R.EMBLEM: As explained in Section 4.1, this paper classifies EMBLEM as broken (by 7 bits).
- Giophantus: Beullens, Castryck, and Vercauteren [35] demonstrated that "for a typical public key for security level 1, we can distinguish ciphertexts with an advantage of about 90%", breaking the IND-CPA security property targeted by Giophantus.
- Guess Again: Panny [86] demonstrated an attack that "quickly recovers the message from a given 'Guess Again' ciphertext without knowledge of the private key".
- HILA5: Bernstein, Bruinderink, Lange, and Panny [24] demonstrated an attack using "less than 6000 queries" to break HILA5's IND-CCA security claim.
- HK17: Bernstein and Lange [27] demonstrated an attack taking, for the 128-bit parameter set, "about ... 2^{16} ... simple computations". The attack was subsequently sped up by Wang, Malluhi, Li, Liu, Pan, Xie, and Bernstein.

- Lepton: Gaborit [57] reported attacks having "complexities of 2^{35} to 2^{70} when the paper announces security from 128 to 262 bits".
- McNie: Gaborit [58] reported an attack that "divides the dimension of the searched message m (now m') by a factor 2 or 3 depending on the considered cases in the parameters" and "divides almost directly the complexity by the announced factor". This paper treats [58] as estimating security level $128/3 \approx 43$.
- Picnic: Dinur and Nadler [52] reported an attack that uses, e.g., 2^{86} operations to recover a key given 2^{35} signatures, or 2^{64} operations to recover a key given 2^{57} signatures. (NIST said it would consider attacks using as many as 2^{64} signatures.)
- pqsigRM: Perlner [88] reported, as joint work with Alperin-Sheriff and Moody, an "attack on your proposed 128 and 192 bit parameters. Our implementation of the attack on the 192 bit parameters can recover an equivalent private key in a matter of seconds and we expect similar performance for the 128 bit parameters".
- RaCoSS: Hülsing, Bernstein, Panny, and Lange demonstrated two attacks, where one [61] "was done overnight" and the other [62] was faster.
- RankSign: Debris-Alazard and Tillich [48] reported a fast attack. The core of the attack, finding "a codeword of rank 2 in any public code", was reported in [48, Table 1] to take "20.12 s" for the smallest RankSign parameters.
- RLCE-KEM: Couvreur, Lequesne, and Tillich [45] presented an attack recovering "short secret keys of RLCE in polynomial time"; "short" is not for all RLCE parameters but covers the smallest RLCE parameter set. [45, page 17] says the attack has an "overall complexity in $O(wn^2k^2)$ operations". The smallest parameters have $w = 96$, $n = 532$, and $k = 376$, so $wn^2k^2 \approx 2^{42}$; since no O constant is specified, this paper treats [45] as estimating attack cost 2^{42}.
- RVB: Panny [87] demonstrated an attack that "quickly computes the secret key from a given public key in the RVB submission".
- SRTPI: Yang, Bernstein, and Lange [102] demonstrated an attack "faster than Alice, after Eve's one-time processing of the public key (which takes a fraction of a second, less time than key generation)".
- WalnutDSA: Beullens and Blackburn [34] demonstrated "that it is possible to forge signatures or compute equivalent secret keys in under a second for 128-bit security parameters".

4.3. Submissions broken after round 1. This paper classifies the following cryptosystems as being broken after round 1 because of the following attacks:

- BIG QUAKE: Bellare, Davis, and Günther [14] said "Our attacks on BIG QUAKE ... recover the symmetric key K from the ciphertext C^* and public key. ... These attacks are very fast, taking at most about the same time as taken by the (secret-key equipped, prescribed) decryption algorithm to recover the key."
- LAKE: Bardet, Bros, Cabarcas, Gaborit, Perlner, Smith-Tone, Tillich, and Verbel [12, Table 1] reported a "complexity" of 2^{71} to break "ROLLO-I-128".

The smallest version of round-1 LAKE is a smaller version of ROLLO-I-128, with (n, m, r, d) reduced from $(47, 79, 5, 6)$ to $(47, 67, 5, 6)$. For simplicity, this paper takes [12] as also estimating security level 71 for the smallest version of LAKE.

- LEDAkem and LEDApkc: Apon, Perlner, Robinson, and Santini [8] presented a "major, practical break" of a merged "LEDAcrypt" submission. [8, Theorem 1.1] says that there is "an algorithm that costs the same as $2^{49.22}$ AES-256 operations and recovers 1 in $2^{47.72}$ of LEDAcrypt's Category 5 (i.e. claimed 256-bit-secure) ephemeral / IND-CPA keys", for a cost/probability ratio around 2^{97}. As for the smallest parameter set, the comments after the theorem say that "this class of very weak keys is present in every parameter set of LEDAcrypt". (A fast chosen-ciphertext attack had been presented in [101] during round 1, but this was not a break: LEDAkem and LEDApkc did not target IND-CCA.)
- LOCKER: Bardet, Bros, Cabarcas, Gaborit, Perlner, Smith-Tone, Tillich, and Verbel [12, Table 1] reported a "complexity" of 2^{93} to break "ROLLO-II-128". The smallest version of round-1 LOCKER is a smaller version of ROLLO-II-128, with (n, m, r, d) reduced from $(149, 83, 5, 8)$ to $(83, 71, 5, 7)$. For simplicity, this paper takes [12] as also estimating security level 93 for the smallest version of LOCKER.
- LUOV: Ding, Deaton, Vishakha, and Yang [50] demonstrated an attack that forges a signature "in under 210 minutes" for the LUOV-7-57-197 parameter set. The smallest round-1 LUOV parameter set was somewhat larger, LUOV-8-63-256, but nevertheless clearly broken.
- NTS-KEM: Chou [44] demonstrated a fast chosen-ciphertext attack breaking NTS-KEM.
- Ouroboros-R: Bardet, Bros, Cabarcas, Gaborit, Perlner, Smith-Tone, Tillich, and Verbel [12, Table 1] reported a "complexity" of 2^{70} to break "ROLLO-III-128". The smallest version of round-1 Ouroboros-R is a smaller version of ROLLO-III-128, with (n, m, w, w_r) reduced from $(47, 101, 5, 6)$ to $(53, 89, 5, 6)$. For simplicity, this paper takes [12] as also estimating security level 70 for the smallest version of Ouroboros-R.
- Rainbow: Beullens [33] demonstrated an attack that "given a Rainbow public key for the SL 1 parameters of the second-round submission" finds the "corresponding secret key after on average 53 hours (one weekend) of computation time on a standard laptop". The parameters attacked in [33] also appear to match the smallest parameters of the round-1 version of Rainbow.
- Round2: Bellare, Davis, and Günther [14] said "Our attacks on ... Round2 recover the symmetric key K from the ciphertext C^* and public key. ... These attacks are very fast, taking at most about the same time as taken by the (secret-key equipped, prescribed) decryption algorithm to recover the key."
- RQC: Bardet, Bros, Cabarcas, Gaborit, Perlner, Smith-Tone, Tillich, and Verbel [12, Table 1] reported a "complexity" of 2^{77} to break "RQC-I". The smallest version of round-1 RQC is similar to RQC-I in [12], with

(m, n, k, w, w_r) changed from $(97, 67, 4, 5, 6)$ to $(89, 67, 7, 5, 6)$. For simplicity, this paper takes [12] as also estimating security level 77 for the smallest version of RQC.

- SIKE: Castryck and Decru [40] and independently Maino and Martindale [74] reported fast attacks against SIKE. The attack from [40] was demonstrated to break "SIKEp434, which aims at security level 1, in about ten minutes on a single core". Followups to [40] and [74] led to, e.g., Decru and Kunzweiler [49] recovering "Alice's secret isogeny in 11 seconds for the SIKEp751 parameters". It is clear that the attacks would be faster for the smallest round-1 SIKE parameters.

4.4. Other submissions. The pool of cryptosystems in this paper is only the round-1 submissions; and, as per the policy from Section 3.2, this paper classifies these cryptosystems as broken only when they are currently *known* to be easier to break than AES-128. These rules imply that many attacks do not constitute breaks.

Some attacks do not apply to the smallest parameter sets for the round-1 submissions. For example, Lyubashevsky and Schwabe [72] demonstrated an attack against a tweaked version of qTESLA from 2019, but not against the round-1 version of qTESLA from 2017. As another example, Kales and Zaverucha [65] reported attacks breaking tweaked versions of MQDSS, but not breaking the round-1 version of MQDSS from 2017.

Some attacks do not reduce attack costs below the cost of breaking AES-128. For example, [19, Appendix C] disproves FrodoKEM's claim that "the FrodoKEM parameter sets comfortably match their target security levels with a large margin", but does not reduce the cost of breaking FrodoKEM below the cost of breaking AES-128. As another example, [93] reports an attack taking 2^{217} operations against the largest SHA-256 parameter sets for SPHINCS+.

For some cryptosystems, attacks *could* reduce attack costs below the cost of breaking AES-128 but are not *known* to. Lattice attacks provide various examples; see Section 4.1. As another example, Tao, Petzoldt, and Ding [99] reported an attack against the HFEv- signature scheme—used in the DualModeMS, GeMSS, and Gui submissions in round 1—costing "$O\left((n+v)^2\binom{2d+2}{d}+(n+v)\binom{2d+2}{d}^2\right)^\omega$" with "$2<\omega\le 3$". The wide range of ω, combined with uncertainty regarding memory-access costs and the O constant, means that having the quantity $(n+v)^2\binom{2d+2}{d}+(n+v)\binom{2d+2}{d}^2$ as low as 2^{30} was still not a clear break. For comparison, GeMSS had been tweaked by the time of [99] to include parameters with $n=177$, $v=15$, and $d=5$, so that quantity was about $2^{27.16}$; but the smallest parameters for the round-1 version of GeMSS had $n=174$, $v=12$, and $d=10$, so that quantity was about $2^{46.14}$. DualModeMS had $n=266$, $v=11$, and $d=8$, so that quantity was about $2^{39.96}$. Gui had $n=184$, $v=16$, and $d=6$, so that quantity was about $2^{30.84}$. Another attack against Gui using "2^{112} evaluations of the public map" and "roughly $3*112*2^{56}$ bits of memory" was presented in [31] during round 1, and similarly was not a clear break.

Similarly, it is not clear whether the HiMQ-3 attack from Beullens [32], reportedly taking "$2^{109.9}$ field operations for the HiMQ-3F(256,24,11,17,15) parameter set", is faster than breaking AES-128.

There are various further attacks that do not seem to warrant comment. If there are disputes about particular attacks, this paper will be updated to address those disputes. If an error in data collection meant that a break was omitted or that a non-break was incorrectly classified as a break, this paper will be updated to correct the error. Readers are once again cautioned that submissions correctly classified as unbroken could in fact be breakable.

5 Listing the selected cryptosystems

Within the pool of 69 round-1 submissions to the NIST competition, this section identifies the cryptosystems selected by selection processes S_0, S_1, S_2.

5.1. Selection process S_0. Identifying the output of S_0 is easy: by definition, S_0 selects all of the round-1 submissions.

5.2. Selection process S_1. Identifying the output of S_1 is easy assuming accuracy of the data collected in Section 4: simply take the 69 submissions minus the 21 submissions listed in Section 4.2 as broken by the end of round 1. This leaves the other 48 submissions, namely BIG QUAKE, BIKE, Classic McEliece, CRYSTALS-DILITHIUM, CRYSTALS-KYBER, Ding Key Exchange, DualModeMS, FALCON, FrodoKEM, GeMSS, Gravity-SPHINCS, Gui, HiMQ-3, HQC, KCL (pka OKCN/AKCN/CNKE), KINDI, LAC, LAKE, LEDAkem, LEDApkc, LIMA, Lizard, LOCKER, LOTUS, LUOV, Mersenne-756839, MQDSS, NewHope, NTRUEncrypt, NTRU-HRSS-KEM, NTRU Prime, NTS-KEM, Odd Manhattan, Ouroboros-R, Post-quantum RSA-Encryption, Post-quantum RSA-Signature, pqNTRUSign, QC-MDPC KEM, qTESLA, Rainbow, Ramstake, Round2, RQC, SABER, SIKE, SPHINCS+, Three Bears, and Titanium.

5.3. Selection process S_2. Identifying the output of S_2 might *seem* easy—simply inspect NIST's list of round-2 selections—but a closer look shows definitional problems here.

As a preliminary matter, notice that the set of 26 "Second Round Candidates" from NIST's report on the selection [2, page 6]—namely BIKE, Classic McEliece, CRYSTALS-DILITHIUM, CRYSTALS-KYBER, FALCON, FrodoKEM, GeMSS, HQC, LAC, LEDAcrypt, LUOV, MQDSS, NewHope, NTRU, NTRU Prime, NTS-KEM, Picnic, qTESLA, Rainbow, ROLLO, Round5, RQC, SABER, SIKE, SPHINCS+, and Three Bears—is not a subset of the set of unbroken round-1 candidates.

There are two reasons for this. First, four of the round-2 candidates were obtained by merging round-1 candidates:

- LEDAcrypt was a "merger of LEDAkem/LEDApkc".
- NTRU was a "merger of NTRUEncrypt/NTRU-HRSS-KEM".

- ROLLO was a "merger of LAKE/LOCKER/Ouroboros-R".
- Round5 was a "merger of Hila5/Round2".

In the case of Round5, recall that HILA5 was broken during round 1. Does this mean that NIST was selecting an already-broken candidate for round 2? No: Round5 had already been defined as Round2 plus *some* aspects of HILA5, not including the aspects that produced the round-1 break of HILA5.

Second, recall that Picnic was broken during round 1. This again does not mean that NIST was selecting an already-broken candidate for round 2. A new version of Picnic had already been defined that avoided the break, and obviously NIST was selecting that version, not the broken version of Picnic from round 1.

Given that NIST knew that Round5 and new-Picnic had broken ancestors and was avoiding those ancestors, this paper treats NIST's selection of Round5 and new-Picnic as NIST selecting new cryptosystems to consider in round 2: i.e., NIST did *not* select HILA5, Round2, or original-Picnic. The subsequent break of Round2 noted in Section 4.3, like the earlier breaks of HILA5 and original-Picnic, was not a failure of S_2. The security of Round5 and new-Picnic is outside the scope of this paper.

For all other mergers (and other cryptosystem tweaks), this paper treats NIST as having selected the original round-1 cryptosystems—systems that were not broken at the time of NIST's selection. In particular, this paper treats NIST's selections of LEDAcrypt, NTRU, and ROLLO as selections of LEDAkem, LEDApkc, NTRUEncrypt, NTRU-HRSS-KEM, LAKE, LOCKER, and Ouroboros-R.

To summarize, NIST selected 2 new cryptosystems, namely Round5 and new-Picnic; and selected 28 out of the original 69 submissions.

6 Results

This section presents the results of this paper's case study, and (in Section 6.3) various double-checks. See Section 7 for analysis of the results.

6.1. Tables of cryptosystems. Tables 6.1.1 and 6.1.2 tabulate the data collected in Sections 4 and 5.

6.2. Graphing the currently-known-failure rates. Figure 6.2.1 reports the currently-known-failure rate for each of the three selection processes defined in Section 2. The figure is split into three graphs. In each graph, the vertical axis plots the estimated attack costs for broken systems, or 0 for demonstrated attacks; as noted in Section 4, this gives an idea of the extent to which this paper's results would be affected by changes in the numbers.

6.3. Reducing risks of error in data collection. Given the number of cryptosystems involved in this case study and the amount of manual effort involved in collecting data, there is an obvious risk of error. The process of preparing this paper took the following steps to reduce this risk.

	broken round 1	selected round 2	broken later
BIG QUAKE	no	no	**yes (0)**
BIKE	no	**yes**	no
CFPKM	**yes (demo)**	no	
Classic McEliece	no	**yes**	no
Compact LWE	**yes (demo)**	no	
CRYSTALS-DILITHIUM	no	**yes**	no
CRYSTALS-KYBER	no	**yes**	no
DAGS	**yes (demo)**	no	
Ding Key Exchange	no	no	no
DME	**yes (81)**	no	
DRS	**yes (80)**	no	
DualModeMS	no	no	no
Edon-K	**yes (demo)**	no	
EMBLEM and R.EMBLEM	**yes (121)**	no	
FALCON	no	**yes**	no
FrodoKEM	no	**yes**	no
GeMSS	no	**yes**	no
Giophantus	**yes (demo)**	no	
Gravity-SPHINCS	no	no	no
Guess Again	**yes (demo)**	no	
Gui	no	no	no
HILA5	**yes (demo)**	no	
HiMQ-3	no	no	no
HK17	**yes (demo)**	no	
HQC	no	**yes**	no
KCL (pka OKCN/AKCN/CNKE)	no	no	no
KINDI	no	no	no
LAC	no	**yes**	no
LAKE	no	**yes**	**yes (71)**
LEDAkem	no	**yes**	**yes (97)**
LEDApkc	no	**yes**	**yes (97)**
Lepton	**yes (35)**	no	
LIMA	no	no	no
Lizard	no	no	no
LOCKER	no	**yes**	**yes (93)**

Table 6.1.1. Cryptosystems in round 1 of the NIST competition, alphabetical order, part 1. "Broken round 1": smallest parameter set for cryptosystem was *known* by the end of round 1 to be easier to break than AES-128; "demo" means an attack was demonstrated, and a number means an estimated attack cost. "Selected round 2": cryptosystem was selected by NIST for round 2 of the competition. NIST's selections of Round5 and a new version of Picnic are treated as selecting new cryptosystems, not as selecting HILA5, Round2, and the original Picnic; see Section 5.3. "Broken later", only for cryptosystems not listed under "broken round 1": smallest parameter set for cryptosystem was *known* by 1 November 2023 to be easier to break than AES-128. Readers are cautioned that unbroken cryptosystems could be breakable.

	broken round 1	selected round 2	broken later
LOTUS	no	no	no
LUOV	no	**yes**	**yes (demo)**
McNie	**yes (43)**	no	
Mersenne-756839	no	no	no
MQDSS	no	**yes**	no
NewHope	no	**yes**	no
NTRUEncrypt	no	**yes**	no
NTRU-HRSS-KEM	no	**yes**	no
NTRU Prime	no	**yes**	no
NTS-KEM	no	**yes**	**yes (demo)**
Odd Manhattan	no	no	no
Ouroboros-R	no	**yes**	**yes (70)**
Picnic	**yes (64)**	no	
Post-quantum RSA-Encryption	no	no	no
Post-quantum RSA-Signature	no	no	no
pqNTRUSign	no	no	no
pqsigRM	**yes (demo)**	no	
QC-MDPC KEM	no	no	no
qTESLA	no	**yes**	no
RaCoSS	**yes (demo)**	no	
Rainbow	no	**yes**	**yes (demo)**
Ramstake	no	no	no
RankSign	**yes (demo)**	no	
RLCE-KEM	**yes (42)**	no	
Round2	no	no	**yes (demo)**
RQC	no	**yes**	**yes (77)**
RVB	**yes (demo)**	no	
SABER	no	**yes**	no
SIKE	no	**yes**	**yes (demo)**
SPHINCS+	no	**yes**	no
SRTPI	**yes (demo)**	no	
Three Bears	no	**yes**	no
Titanium	no	no	no
WalnutDSA	**yes (demo)**	no	

Table 6.1.2. Cryptosystems in round 1 of the NIST competition, alphabetical order, part 2. "Broken round 1": smallest parameter set for cryptosystem was *known* by the end of round 1 to be easier to break than AES-128; "demo" means an attack was demonstrated, and a number means an estimated attack cost. "Selected round 2": cryptosystem was selected by NIST for round 2 of the competition. NIST's selections of Round5 and a new version of Picnic are treated as selecting new cryptosystems, not as selecting HILA5, Round2, and the original Picnic; see Section 5.3. "Broken later", only for cryptosystems not listed under "broken round 1": smallest parameter set for cryptosystem was *known* by 1 November 2023 to be easier to break than AES-128. Readers are cautioned that unbroken cryptosystems could be breakable.

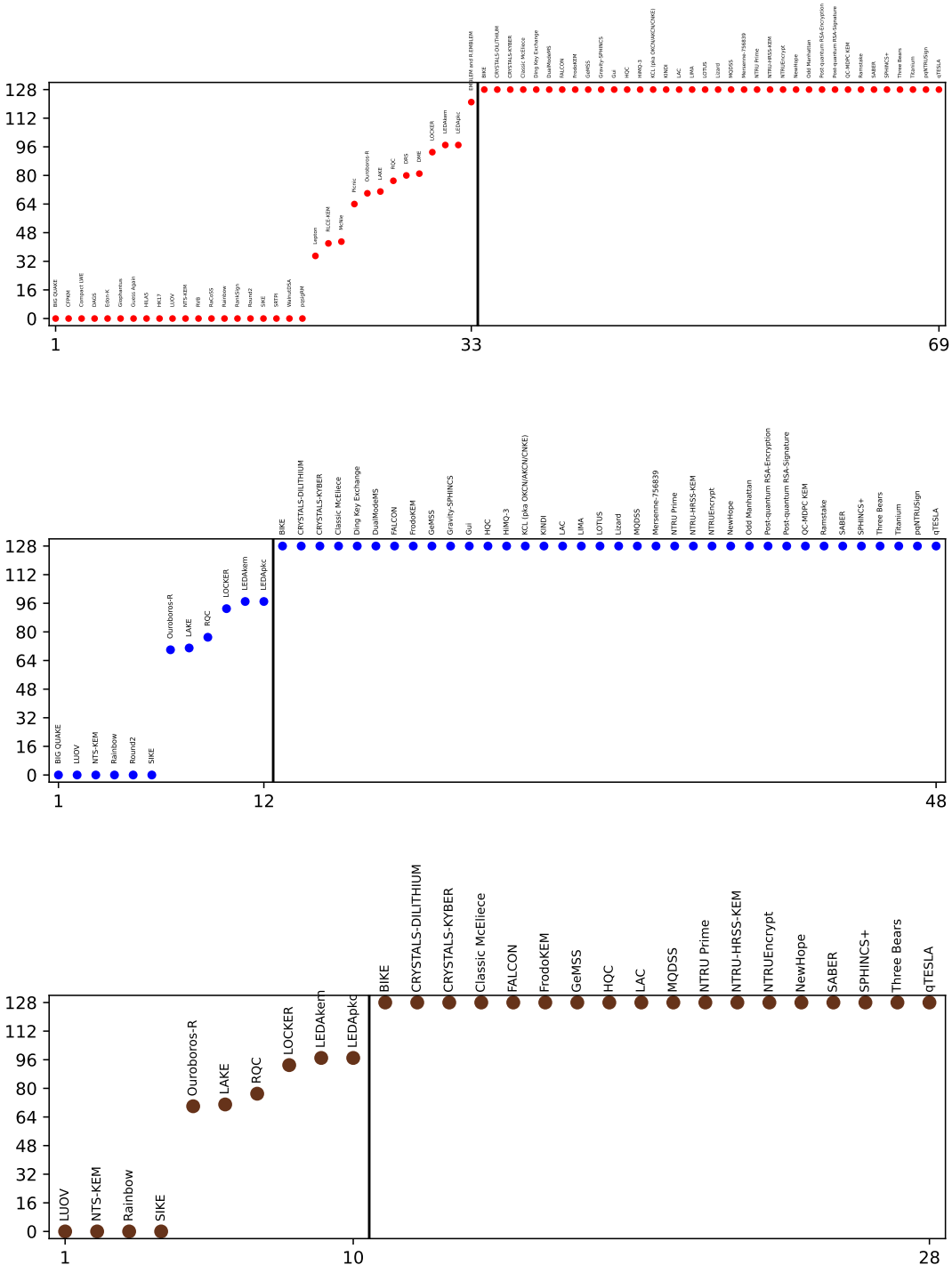

Fig. 6.2.1. Currently-known-failure rates for three selection processes. Selection process S_0 (top, red) selects all round-1 submissions to the NIST competition. Selection process S_1 (middle, blue) selects the round-1 submissions not broken by the end of round 1. Selection process S_2 (bottom, brown) selects the round-1 submissions that were selected by NIST for round 2. Vertical axis is 128 for unbroken systems, 0 when attacks have been demonstrated, intermediate numbers for log base 2 of estimated attack costs. Readers are cautioned that unbroken cryptosystems could be breakable.

First, all data was collected into a unified file `data.txt` with one line for each cryptosystem. For example, the line

```
2:sig:later,demo:LUOV
```

means that LUOV was selected for round 2 and was later broken by a demonstrated attack. The tables and graphs in this paper were produced by simple Python scripts from that file. The file and scripts are included in this paper as PDF attachments.

Second, various manual spot-checks of the tables and graphs did not detect any errors. Note that the extra pieces of information contained in the graphs beyond failure rates—the vertical positions, and labels for each cryptosystem—help support spot-checks.

Third, the "broken round 1" column of Tables 6.1.1 and 6.1.2 was compared in detail to the attack classification in [96], and every discrepancy was investigated. The discrepancies are as follows:

- DME is not listed as attacked in the "status" column of [96], but is listed as "attacked-patch proposed" in notes accompanying [96].
- EMBLEM is not listed as attacked in [96]. Presumably this is because there was no comments filed specifically for EMBLEM in the NIST competition, except regarding an implementation discrepancy. See Section 4.1 for why this paper marks EMBLEM as broken in round 1.
- Gravity-SPHINCS has a note in [96] saying "Fault injection attack". In this paper, "broken" is defined (see Section 3.2) in terms of the targeted security properties of the specified cryptosystems; fault attacks are not relevant to those properties, such as EUF-CMA in the case of Gravity-SPHINCS.
- HILA5 is not listed as attacked in [96]. This is explained by [96] listing HILA5 as targeting only IND-CPA security. The original HILA5 submission targeted IND-CCA security; see [24, Section 5] for quotes and further references.
- LAC has notes in [96] saying "failure rate potentially worse than expected" and "timing attack on the underlying ECC to break IND-CCA security". Attacks that "potentially" work are disregarded in this paper (see Section 3.2), and the "break IND-CCA security" claim is incorrect: timing attacks are not relevant to the IND-CCA definition.
- LAKE has notes in [96] saying "minor implementation problem: fixed" and "key recovery attack stronger than originally anticipated". Implementation issues are not relevant to the targeted security property of the specified cryptosystem, and the "stronger" claim appears to be a misreading of [91], which stated that "the submission document for LAKE and LOCKER *underestimates* the complexity of a key recovery attack".
- LIMA has a note in [96] saying "concerns surrounding rejection sampling analysis - patch proposed". This does not indicate an attack.
- LOCKER: Same comment as LAKE above.
- LOTUS has a note in [96] saying "CCA attack-*patched*". This appears to be referring to [68], which says that the "reference implementation of KEM

LOTUS128 fails to achieve CCA security" and makes no claims about the specified cryptosystem.
- Odd Manhattan: Analogous to LOTUS, this time for [69].
- Picnic is not listed as attacked in [96]. However, [52] is a clear break of the round-1 version of Picnic, and was posted in 2018.
- pqNTRUsign is listed in [96] as "Vulnerable to CMA attack - *patched*". This appears to be based on misinformation from NIST. Specifically, NIST falsely claimed in [89] to have a "chosen message attack" against pqNTRUsign. NIST then admitted in [90] that it did not have a "concrete attack", but NIST falsely labeled [90] as a *clarification* rather than an *erratum*.
- Round2 has notes in [96] saying "Concerns surrounding proof of the IND-CPA security" and "Potential CCA attack". Neither of these is claiming that an attack is known.
- SIKE has notes in [96] saying "Quantum attacks overestimated" and "Potential lower-running-cost attack". This appears to be a misreading of [92], which reviewed reasons to believe that SIKE was *underestimating* the cost of attacks.

None of these discrepancies indicate any errors in this paper's data-collection process. Some of the discrepancies come from different choices of scope: [96] considering patched cryptosystems (DME, HILA5), [96] considering implementation attacks (Gravity-SPHINCS, LOTUS, Odd Manhattan), and [96] noting potential attacks (LAC, LIMA). The remaining discrepancies come from errors in [96] or its sources: [96] apparently missing attacks (EMBLEM, Picnic), [96] apparently misreading its sources (LAKE, LOCKER, SIKE), and NIST falsely claiming an attack (pqNTRUsign).

Errors are unsurprising given the amount of manual work involved in tracking the status of this number of submissions. Tracking will become easier and more reliable to the extent that attacks are integrated into the new CryptAttackTester (CAT) framework from [25]. CAT includes complete definitions of various attack algorithms and cost/probability predictions for each algorithm in a fully defined cost metric (compare Section 3.5), and tests the predictions against the observed algorithm behavior for small attack problems. The initial CAT release includes attacks against AES-128 and Classic McEliece.

7 Conclusions, and hpotheses for further investigation

The results from Section 6, in short, are that half of the round-1 submissions in the NIST Post-Quantum Cryptography Standardization Project have been broken, a quarter of the submissions that weren't broken by the end of round 1 have been broken, and a third of the submissions that were selected by NIST for round 2 have been broken. The exact tallies are, respectively, 33 out of 69 (ratio about 48%) for S_0, 12 out of 48 (ratio 25%) for S_1, and 10 out of 28 (ratio about 36%) for S_2.

This section considers various hypotheses for what is going on here; draws the conclusions listed in Section 1; and identifies various specific hypotheses that appear to warrant further investigation.

7.1. Initial hypotheses. Malkiel [76] famously stated fifty years ago that a blindfolded chimpanzee throwing darts could select a stock portfolio as well as the experts. This statement was based on the idea that stock prices already reflect all available information and change only because of unpredictable news. Subsequent evidence reported for the statement includes (1) the performance of index funds and (2) various smaller-scale simulations of blindfolded chimpanzees. See generally [77].

An analogous null hypothesis regarding cryptography states that the random processes of cryptosystem design and cryptosystem selection are uncorrelated with the actual failure rate of those cryptosystems, doing no better—and no worse!—than chance at predicting cryptanalytic news.

It is easy to formulate alternative hypotheses, centered on the idea that cryptographers do a good job of identifying security problems. Here are examples of such alternative hypotheses, with concrete numbers for falsifiability:

- Hypothesis 1: Submissions to a new cryptographic competition will have at least 90% chance of being secure. Rationale: The designers will already have looked for, recognized, and eliminated any security dangers, except that there are unfortunate and rare cases in which cryptographers make mistakes, and there are occasional submissions to competitions by pseudo-cryptographers.
- Hypothesis 2: Submissions that aren't broken after a round of public review will have at least 90% chance of being secure. Rationale: "Come on, how long does attack development take?"
- Hypothesis 3: Whatever the failure rate is for submissions and for a round of public review, submissions selected by an organization prioritizing security (as in [84, Section 4.A, "The security provided by a cryptographic scheme is the most important factor in the evaluation"]) will have at least 90% chance of being secure. Rationale: The organization will be putting an even higher priority than the designers on looking for, recognizing, and eliminating any security dangers.

These hypotheses are compared below to this paper's results.

7.2. High failure rates. This paper's first conclusion is that, contrary to the three alternative hypotheses formulated above, the currently-known-failure rates for selection processes S_0, S_1, S_2 are above 10%. Readers are cautioned that the actual failure rates could be even higher.

For users who want cryptography to fail with probability far *below* 10%, it is concerning to see how many failures were allowed by these processes.

To evaluate the level of statistical significance of the "above 10%" statement, consider 28 independent flips of a coin that is heads with probability 10%. The chance of at least 10 heads is $\sum_{10 \le j \le 28} \binom{28}{j} (9/10)^{28-j} (1/10)^j \approx 2^{-12.0}$. Similarly, the chance of at least 12 heads in 48 independent flips is about $2^{-8.8}$,

and the chance of at least 33 heads in 69 independent flips is about $2^{-49.4}$. All of these are well below the 5% level (about $2^{-4.3}$) that is typically taken as justifying a claim of statistical significance.

The attached `exact10.sage` script computes the exponents in the previous paragraph. As a double-check, the attached `sim10.c` program simulates 1000000 28-flip experiments, 1000000 48-flip experiments, and 1000000 69-flip experiments; a typical run prints out unsurprising exponents −11.931569, −8.819129, and $-\infty$.

7.3. Different failure rates. This paper's second and third conclusions are that the currently-known-failure rates of S_0, S_1, S_2 have statistically significant differences, but not matching the pattern that one might expect.

Two parts of this are easy. First, given that currently known failures include all of the submissions broken by the end of round 1, note that S_1 cannot have a higher failure rate than S_0, and S_1 has a lower failure rate than S_0 if there are *any* submissions broken by the end of round 1.[3] Statistical significance for S_1 having a lower failure rate than S_0 is thus a weak statement: it merely requires observing that at least one submission was broken by the end of round 1. More interesting is the number of submissions broken by the end of round 1, namely 21 out of the 69 submissions.

Second, even though one might expect S_2 to have significantly *lower* currently-known-failure rate than S_1, obviously that is not what the results show: S_2 in fact has *higher* currently-known-failure rate than S_1.

With more work, one can check as follows that the *increase* from the currently-known-failure rate of S_1 to the currently-known-failure rate of S_2 is statistically significant.

Consider the null hypothesis that S_2 is chosen by the blindfolded chimpanzee as a uniform random size-28 subset of S_1, while each of the 48 elements of S_1 fails independently with probability p. There is chance $\binom{28}{j}(1-p)^{28-j}p^j$ of exactly j failures within S_2, each of which also produces j failures with S_1; independently of that, there is chance $\binom{20}{k}(1-p)^{20-k}p^k$ of exactly k failures within the other 20 elements of S_2.

The chance of at least 10 failures within S_2, accompanied by at most 12 failures within S_1, is then the sum of $\binom{28}{j}(1-p)^{28-j}p^j\binom{20}{k}(1-p)^{20-k}p^k$ over all six pairs (j,k) with $10 \le j$ and $j+k \le 12$. A straightforward calculation shows that this polynomial in p increases from 0 to 0.0074653... as p increases from 0 to 0.24404..., and then decreases back down to 0 as p continues increasing to 1.

The attached `exacts2.sage` script computes the numbers 0.0074653... and 0.24404... in the previous paragraph, and, as a double-check, prints out the polynomial values for various values of p. As a triple-check, the attached `sims2.c` program simulates 1000000 experiments for various values of p.

[3] Of course, if a submission is known to have lower security than AES-128 and is excluded on that basis, but a subsequent decrease in certainty means that the submission is no longer known to have lower security than AES-128, then the exclusion could end up increasing the subsequently known failure rate. Further analysis of regressions of knowledge is outside the scope of this paper.

This calculation rejects the null hypothesis with above 99% confidence. No matter what the probability p of a bad apple is, the chimpanzee blindly picking 28 apples out of 48 would have had to be very unlucky—below 1 chance in 100—to end up with at least 10 bad apples out of the 28 and at the same time at most 12 bad apples out of the 48.

But this is exactly what NIST managed to do with its round-2 selections. In other words, NIST managed to keep 10 out of the 12 submissions from S_1 that are now known to be failures, while keeping just 18 out of the other 36 submissions from S_1.

7.4. How and why did this happen? It is not safe to conclude from the above observations that NIST is applying a statistically significant ability to predict future attacks. One reason that this is unsafe is that the statistical calculations rely on modeling the brokenness of each submission as independent, but in fact there are obvious correlations: e.g., all 4 rank-based submissions in S_2 were broken simultaneously.

It is nevertheless plausible that NIST *is* applying such an ability. Consider, for example, the following two hypotheses:

- Hypothesis 4: NIST is actually putting heavier emphasis on cryptosystem speed than on security.
- Hypothesis 5: Speed is negatively correlated with security.

The combination of these two hypotheses would provide a reasonably simple explanation for NIST doing worse than the blindfolded chimpanzee from a security perspective.

It is easy to find literature that (1) suggests further predictors of security while (2) not scientifically studying the validity of the predictors; see, e.g., the "provable security" citations in Section 1.1. Perhaps NIST is using a specific predictor that it believes increases security but that actually decreases security.

A general difficulty in studying NIST's selection process is that the process is poorly documented. NIST has on many occasions claimed that its process is "transparent"—e.g., [97] says "We operate transparently. We've shown all our work"—but this claim is easily shown to be false. For example:

- NIST's round-1 report [2] states various features of the submissions NIST selected for round 2, but does not say what weight was put on each feature, and does not cover the deselected submissions. Furthermore, the definitions of the features are often unclear; see below for some examples.
- NIST's corresponding talk [80, page 11] lists broken submissions (the list in Tables 6.1.1 and 6.1.2 matches this except for EMBLEM, HILA5, and Picnic), and says that pqRSA and DualModeMS were "too inefficient", but does not explain how NIST arrived at its 17 further deselections.
- Ongoing FOIA proceedings mentioned in Section 2 have revealed various competition documents that were not previously publicly available, including documents marked by NIST as "not for public distribution". See generally [23]. The documents released so far cover only small fragments of the NIST

competition period, although they do reveal, e.g., NIST's rationale for not including BIG QUAKE.

Despite this difficulty, one can ask, for each cryptosystem feature that NIST is known to have listed, whether that feature has a nonzero (positive or negative) correlation with security. One has to begin by asking what each feature means. For example, [2] states that "in a few cases, a submitted design was selected in part for its uniqueness and elegance"; studying the effect of this selection factor will require a clear statement of how "uniqueness" and "elegance" were evaluated. As another example, [2] states that "NIST studied the security arguments presented in the submission package" and that NIST "considered the overall quantity" of security analysis; it would be easy to count, e.g., the number of pages labeled as security analysis or the number of occurrences of the word "conservative".

There are other possible explanations. Recall NIST claiming in [46] that "competitions can focus the attention of cryptographers around the world". This claim suggests the following hypothesis:

- Hypothesis 6: Cryptosystems selected by NIST are, because of the selection, attracting more cryptanalytic attention than they would otherwise have attracted.

It would be interesting to study the extent to which this occurs and how much effect it has on the discovery of attacks.

Consider, e.g., the HFEv- attack from [99]. That paper chose GeMSS, and not DualModeMS or Gui, to report quantitative examples of the performance of its attack. Perhaps this indicates that NIST's selection of GeMSS attracted cryptanalytic attention to GeMSS.

On the other hand, NIST's deselection of DualModeMS and Gui did not shield those submissions from the same attack. Similarly, the deselected submission Gravity-SPHINCS appears to be very similar to the selected submission SPHINCS+; the deselected submission QC-MDPC appears to be very similar to the selected submission BIKE; and many of the advances in lattice attacks appear to have reduced security levels by similar amounts for lattice submissions whether or not NIST selected them (although there have also been some attacks against specific lattice submissions, such as [19, Appendix C]). These overlaps do not appear to have been an accident: [80, page 20] says "NIST wanted to keep diversity, but reduce numbers".

There are other examples of deselected unbroken submissions that appear to be farther from any of NIST's selections, and vice versa. Certainly the security track records are different.

Consider, for example, the selection process $S_1 - S_2$: "look at what NIST did and do the opposite". This has a currently-known-failure rate of "only" 2 out of 20, which sounds very high for a safety mechanism but is still much lower than S_2. (The failures here are BIG QUAKE and Round2, both of which were broken by simple attacks from [14].) The speed hypothesis, Hypothesis 4, explains this gap as S_2 being relatively speed-focused while $S_1 - S_2$ is relatively security-focused.

The attention hypothesis, Hypothesis 6, explains this as $S_1 - S_2$ needing more study *because* it is not what NIST selected. Both hypotheses could be correct simultaneously.

A more extreme form of Hypothesis 6 is the following hypothesis:

- Hypothesis 7: NIST's selections actually *improved* security, while this has been hidden by gaps between the currently-known-failure rate and the actual failure rate, caused by a lack of attention to the deselected cryptosystems.

Imagine, for example, that attacks published in the next few years break 6 of the 20 submissions in $S_1 - S_2$, without breaking any of the submissions in S_2. The failure rate known for S_1 at that point will then be 18/48, i.e., 37.5%, above the 35.7% failure rate known for S_2.

Of course, further attacks could also have the opposite effect—perhaps most easily via continued lattice attacks as in Section 4.1, since 12 of the 28 submissions in S_2 are lattice-based cryptosystems.

There are ways to study attention hypotheses without waiting for attacks to be developed. One can study how cryptanalysts have chosen to spend their time, and study how these choices have influenced the schedule of attack discovery. It would be particularly interesting to quantify the workload that competitions are placing on cryptanalysts, compared to the cryptanalytic time devoted to those competitions; and to quantify the relationship between the cryptanalytic time spent on a cryptosystem and the chance of an attack being missed.

7.5. Further topics to investigate. There are many other cryptographic selection processes to consider, and there are many reasons that other case studies could produce different results.

For example, readers might formulate the following hypothesis:

- Hypothesis 8: NIST's post-quantum competition has more failures than other competitions, and failures that take longer to discover. Rationale: Quantum computers complicate cryptanalysis. As NIST put it in [**42**, page 10], "science for assessing classical security is better developed than that for assessing quantum security".

The same readers might be surprised to learn that none of the breaks in NIST's competition have come from quantum attacks. Here are three different hypotheses that could explain this:

- Hypothesis 9: Quantum attacks matter for only a small corner of cryptography, and all of the submissions managed to avoid that corner.
- Hypothesis 10: Quantum attacks matter for some submissions, but quantum cryptanalysis is so difficult that none of those attacks have been found yet.
- Hypothesis 11: Quantum attacks matter for some submissions, but NIST's rules for the competition discouraged attention to quantum attacks.

Hypothesis 10 and Hypothesis 11 appear to be compatible with each other, but neither is compatible with Hypothesis 9.

Hypothesis 11, like Hypotheses 6 and 7 from Section 7.4, is contrary to a null hypothesis stating that NIST's standardization process has no effect on research. This null hypothesis is claimed to be a fact in, e.g., [7, "The forward march of research inevitably continues regardless of this or that standardization process"]. The following paragraphs note evidence that appears to support Hypothesis 11.

Recall that NIST chose AES-128 key search as the minimum allowed security level for the post-quantum competition. One can search all AES-128 keys using about 2^{64} quantum operations, or using 2^{128} non-quantum operations, modulo caveats from Section 3.5. One way to break a submission is to present an attack beating 2^{64} quantum operations; another way is to present an attack beating 2^{128} non-quantum operations. For comparison, [15] had recommended eliminating the pre-quantum security requirement and setting the minimum allowed security level as 2^{64} post-quantum security.

For almost all of the submissions to the competition, the best attacks known have always been quantitatively closer to beating 2^{128} non-quantum operations than to beating 2^{64} quantum operations. NIST's choice of minimum allowed security level thus creates an incentive for cryptanalysts to focus on improving non-quantum attacks rather than on improving quantum attacks. (In many cases the resulting non-quantum breaks were so fast that they could have been trivially rephrased as quantum breaks, but none of the breaks cited in this paper were phrased in that way.)

Cryptanalysts have noticed this incentive. Consider [4], which introduced faster non-quantum algorithms for lattice enumeration. [4, page 5] estimated that quantum versions of the same algorithms would beat previous quantum lattice attacks up to rank "$k = 547$", well into the range of cryptographic interest. One would imagine that the fastest known quantum lattice attack would be of interest for lattice submissions in a post-quantum competition. However, [4, page 5] then said "we stress that this work does *not* invalidate the claimed NIST Security Level of such submissions. This is because a given security level is defined by both a classical and a quantum cost: roughly 2^{λ} classically and $2^{\lambda/2}$ quantumly". It took years until the appearance of the first concrete analysis [11] of the cost of quantum lattice enumeration.

It would be interesting to study the extent to which NIST's rules have slowed down the development of quantum attacks; to study whether NIST's post-quantum competition is in fact more failure-prone than other competitions; and, if so, to investigate why.

References

[1] — (no editor), *2018 IEEE international symposium on information theory, ISIT 2018, Vail, CO, USA, June 17–22, 2018*, IEEE, 2018. ISBN 978-1-5386-4781-3. See [70].

[2] Gorjan Alagic, Jacob Alperin-Sheriff, Daniel Apon, David Cooper, Quynh Dang, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, Ray

Perlner, Angela Robinson, Daniel Smith-Tone, *Status report on the first round of the NIST Post-Quantum Cryptography Standardization Process* (2019). NISTIR 8240. URL: https://web.archive.org/web/20190930153452/https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8240.pdf. Citations in this document: §2.1, §5.3, §7.4, §7.4, §7.4.

[3] Gorjan Alagic, Daniel Apon, David Cooper, Quynh Dang, Thinh Dang, John Kelsey, Jacob Lichtinger, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, Ray Perlner, Angela Robinson, Daniel Smith-Tone, *Status report on the third round of the NIST Post-Quantum Cryptography Standardization Process* (2022). NISTIR 8413. URL: https://web.archive.org/web/20230824124130/https://nvlpubs.nist.gov/nistpubs/ir/2022/NIST.IR.8413-upd1.pdf. Citations in this document: §4.1.

[4] Martin R. Albrecht, Shi Bai, Jianwei Li, Joe Rowell, *Lattice reduction with approximate enumeration oracles—practical algorithms and concrete performance*, in Crypto 2021 [**78**] (2021), 732–759. URL: https://eprint.iacr.org/2020/1260. Citations in this document: §7.5, §7.5, §7.5.

[5] Martin R. Albrecht, Benjamin R. Curtis, Amit Deo, Alex Davidson, Rachel Player, Eamonn W. Postlethwaite, Fernando Virdia, Thomas Wunderer, *Estimate all the {LWE, NTRU} Schemes!*, in SCN 2018 [**41**] (2018), 351–367. URL: https://eprint.iacr.org/2018/331. Citations in this document: §4.1.

[6] Martin R. Albrecht, Eamonn Postlethwaite, Fernando Virdia, *OFFICIAL COMMENT: CFPKM* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/t0IQjM-UhpU/m/iZ_kir6kAAAJ. Citations in this document: §4.2.

[7] Daniel Apon, *Re: Falsifiability* (2023). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/W2VOzy0wz_E/m/hPttnwfgBwAJ. Citations in this document: §7.5.

[8] Daniel Apon, Ray A. Perlner, Angela Robinson, Paolo Santini, *Cryptanalysis of LEDAcrypt*, in Crypto 2020 [**79**] (2020), 389–418. URL: https://eprint.iacr.org/2020/455. Citations in this document: §4.3, §4.3.

[9] Lars Arge, Christian Cachin, Tomasz Jurdzinski, Andrzej Tarlecki (editors), *Automata, languages and programming, 34th international colloquium, ICALP 2007, Wroclaw, Poland, July 9–13, 2007, proceedings*, Lecture Notes in Computer Science, 4596, Springer, 2007. ISBN 978-3-540-73419-2. See [47].

[10] Roberto Avanzi, Joppe Bos, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, Gregor Seiler, Damien Stehlé, *CRYSTALS-Kyber: Algorithm specifications and supporting documentation (version 3.02)* (2021). URL: https://web.archive.org/web/20211215150153/https://pq-crystals.org/kyber/data/kyber-specification-round3-20210804.pdf. Citations in this document: §3.4, §3.4, §3.4, §3.4, §4.1, §4.1, §4.1.

[11] Shi Bai, Iggy van Hoof, Floyd Johnson, Tanja Lange, Tran Ngo, *Concrete analysis of quantum lattice enumeration* (2023). Asiacrypt 2023, to appear. URL: https://eprint.iacr.org/2023/1623. Citations in this document: §7.5.

[12] Magali Bardet, Maxime Bros, Daniel Cabarcas, Philippe Gaborit, Ray Perlner, Daniel Smith-Tone, Jean-Pierre Tillich, Javier Verbel, *Improvements of algebraic attacks for solving the Rank Decoding and MinRank problems*, in Asiacrypt 2020 [**81**] (2020), 507–536. URL: https://arxiv.org/abs/2002.08322. Citations in this document: §4.3, §4.3, §4.3, §4.3, §4.3, §4.3, §4.3, §4.3, §4.3.

[13] Elise Barelli, Alain Couvreur, *An efficient structural attack on NIST submission DAGS* (2018). URL: https://arxiv.org/abs/1805.05429. Citations in this document: §4.2.

[14] Mihir Bellare, Hannah Davis, Felix Günther, *Separate your domains: NIST PQC KEMs, oracle cloning and read-only indifferentiability*, in Eurocrypt 2020 [**38**] (2020), 3–32. URL: https://eprint.iacr.org/2020/241. Citations in this document: §4.3, §4.3, §7.4.

[15] Daniel J. Bernstein, *Some challenges in post-quantum standardization* (2016). URL: https://blog.cr.yp.to/20161030-pqnist.html. Citations in this document: §7.5.

[16] Daniel J. Bernstein, *Comparing proofs of security for lattice-based encryption* (2019). Second PQC Standardization Conference. URL: https://cr.yp.to/papers.html#latticeproofs. Citations in this document: §1.1, §1.1.

[17] Daniel J. Bernstein, *A discretization attack* (2020). URL: https://cr.yp.to/papers.html#categories. Citations in this document: §1.1, §3.5.

[18] Daniel J. Bernstein, *Cryptographic competitions* (2023). URL: https://cr.yp.to/papers.html#competitions. Citations in this document: §1.1, §1.1, §1.1, §1.1, §2.4.

[19] Daniel J. Bernstein, *Multi-ciphertext security degradation for lattices* (2023). URL: https://cr.yp.to/papers.html#lprrr. Citations in this document: §4.4, §7.4.

[20] Daniel J. Bernstein, *Turbo Boost* (2023). URL: https://blog.cr.yp.to/20230609-turboboost.html. Citations in this document: §3.2.

[21] Daniel J. Bernstein, *The inability to count correctly* (2023). URL: https://blog.cr.yp.to/20231003-countcorrectly.html. Citations in this document: §3.5.

[22] Daniel J. Bernstein, *Reducing "gate" counts for Kyber-512* (2023). URL: https://blog.cr.yp.to/20231023-clumping.html. Citations in this document: §3.5.

[23] Daniel J. Bernstein, *NSA-NIST-PQC FOIA responses* (2023). URL: https://nist.pqcrypto.org/foia/index.html. Citations in this document: §2.1, §7.4.

[24] Daniel J. Bernstein, Leon Groot Bruinderink, Tanja Lange, Lorenz Panny, *HILA5 Pindakaas: On the CCA security of lattice-based encryption with error correction*, in Africacrypt 2018 [**64**] (2017), 203–216. URL: https://eprint.iacr.org/2017/1214. Citations in this document: §4.2, §6.3.

[25] Daniel J. Bernstein, Tung Chou, *CryptAttackTester: high-assurance attack analysis* (2023). URL: https://cr.yp.to/papers.html#cat. Citations in this document: §3.4, §6.3.

[26] Daniel J. Bernstein, Tung Chou, Chitchanok Chuengsatiansup, Andreas Hülsing, Eran Lambooij, Tanja Lange, Ruben Niederhagen, Christine van Vredendaal, *How to manipulate curve standards: a white paper for the black hat*, in SSR 2015 (2015). URL: https://bada55.cr.yp.to/. Citations in this document: §1.1.

[27] Daniel J. Bernstein, Tanja Lange, *OFFICIAL COMMENT: HK17* (2017). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/KaEUbyVSG1o/m/QuaVPTzgBgAJ. Citations in this document: §4.2.

[28] Daniel J. Bernstein, Tanja Lange, *Post-quantum cryptography* (2019). URL: https://cr.yp.to/talks.html#2019.06.10. Citations in this document: §1.1.

[29] Daniel J. Bernstein, Tanja Lange, Ruben Niederhagen, *Dual EC: a standardized back door*, in [**95**] (2015), 256–281. URL: https://eprint.iacr.org/2015/767. Citations in this document: §1.1.

[30] Ward Beullens, *OFFICIAL COMMENT: DME* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/YiRH4tDXe_k/m/9dz5X4n1AgAJ. Citations in this document: §4.2.

[31] Ward Beullens, *OFFICIAL COMMENT: Gui* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/8VE6gtFPSH8/m/tMjPqQZyBAAJ. Citations in this document: §4.4.

[32] Ward Beullens, *OFFICIAL COMMENT: HIMQ-3 :Key Recovery Attack* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/fIPh6v9-9Ac/m/S3mQadZQBgAJ. Citations in this document: §4.4.

[33] Ward Beullens, *Breaking Rainbow takes a weekend on a laptop*, in Crypto 2022 [**53**] (2022), 464–479. URL: https://eprint.iacr.org/2022/214. Citations in this document: §4.3, §4.3.

[34] Ward Beullens, Simon R. Blackburn, *Practical attacks against the Walnut digital signature scheme*, in Asiacrypt 2018 [**94**] (2018), 35–61. URL: https://eprint.iacr.org/2018/318. Citations in this document: §4.2.

[35] Ward Beullens, Wouter Castryck, Frederik Vercauteren, *OFFICIAL COMMENT: Giophantus* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/4umpcStUJrQ/m/Ocq6xfeWAwAJ. Citations in this document: §4.2.

[36] Jonathan Bootle, Mehdi Tibouchi, Keita Xagawa, *OFFICIAL COMMENT: Compact LWE* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/_fOnZNYHdE4/m/-WNWqwUFAwAJ. Citations in this document: §4.2.

[37] Daniel R. L. Brown, *Layering diverse cryptography to lower risks of future and secret attacks: post-quantum estimates* (2021). URL: https://eprint.iacr.org/2021/608. Citations in this document: §1.1, §1.1.

[38] Anne Canteaut, Yuval Ishai (editors), *Advances in cryptology—EUROCRYPT 2020—39th annual international conference on the theory and applications of cryptographic techniques, Zagreb, Croatia, May 10–14, 2020, proceedings, part II*, Lecture Notes in Computer Science, 12106, Springer, 2020. ISBN 978-3-030-45723-5. See [14].

[39] Anne Canteaut, François-Xavier Standaert (editors), *Advances in cryptology—EUROCRYPT 2021—40th annual international conference on the theory and applications of cryptographic techniques, Zagreb, Croatia, October 17–21, 2021, proceedings, part I*, 12696, Springer, 2021. ISBN 978-3-030-77869-9. See [50].

[40] Wouter Castryck, Thomas Decru, *An efficient key recovery attack on SIDH*, in Eurocrypt 2023 [**60**] (2022), 423–447. URL: https://eprint.iacr.org/2022/975. Citations in this document: §4.3, §4.3, §4.3.

[41] Dario Catalano, Roberto De Prisco (editors), *Security and cryptography for networks—11th international conference, SCN 2018, Amalfi, Italy, September 5–7, 2018, proceedings*, Lecture Notes in Computer Science, 11035, Springer, 2018. See [5].

[42] Lily Chen, *NIST PQC standardization—process, issues and strategies* (2017). URL: https://nist.pqcrypto.org/foia/20230105/Asia-PQC-3rd-03222017-p.pdf. Citations in this document: §7.5.

[43] Jung Hee Cheon, Thomas Johansson (editors), *Post-quantum cryptography—13th international workshop, PQCrypto 2022, virtual event, September 28–30, 2022, proceedings*, 13512, Springer, 2022. ISBN 978-3-031-17233-5. See [93].

[44] Tung Chou, *An IND-CCA2 attack against the 1st- and 2nd-round versions of NTS-KEM*, in SecITC 2020 [**73**] (2020), 165–184. URL: https://eprint.iacr.org/2020/1578. Citations in this document: §4.3.

[45] Alain Couvreur, Matthieu Lequesne, Jean-Pierre Tillich, *Recovering short secret keys of RLCE in polynomial time*, in PQCrypto 2019 [**51**] (2019), 133–152. URL: https://eprint.iacr.org/2018/528. Citations in this document: §4.2, §4.2, §4.2.

[46] Cryptographic Technology Group, *NIST cryptographic standards and guidelines development process* (2016). NISTIR 7977. URL: https://csrc.nist.gov/publications/detail/nistir/7977/final. Citations in this document: §1.1, §7.4.

[47] Ivan Damgård, *A "proof-reading" of some issues in cryptography*, in ICALP 2007 [**9**] (2007), 2–11. URL: https://users-cs.au.dk/%7Eivan/positionpaper.pdf. Citations in this document: §1.1, §1.1, §1.1, §1.1, §1.1, §1.1, §1.1.

[48] Thomas Debris-Alazard, Jean-Pierre Tillich, *Two attacks on rank metric code-based schemes: RankSign and an IBE scheme*, in Asiacrypt 2018 [**94**] (2018), 62–92. URL: https://eprint.iacr.org/2018/339. Citations in this document: §4.2, §4.2.

[49] Thomas Decru, Sabrina Kunzweiler, *Efficient computation of* $(3^n, 3^n)$*-isogenies*, in Africacrypt 2023 [**82**] (2023), 53–78. URL: https://eprint.iacr.org/2023/376. Citations in this document: §4.3.

[50] Jintai Ding, Joshua Deaton, Vishakha, Bo-Yin Yang, *The nested subset differential attack—A practical direct attack against LUOV which forges a signature within 210 minutes*, in Eurocrypt 2021 [**39**] (2021), 329–347. URL: https://eprint.iacr.org/2020/967. Citations in this document: §4.3.

[51] Jintai Ding, Rainer Steinwandt (editors), *Post-quantum cryptography—10th international conference, PQCrypto 2019, Chongqing, China, May 8–10, 2019 revised selected papers*, 11505, Springer, 2019. ISBN 978-3-030-25509-1. See [45].

[52] Itai Dinur, Niv Nadler, *Multi-target attacks on the Picnic signature scheme and related protocols*, in Eurocrypt 2019 [**63**] (2019), 699–727. URL: https://eprint.iacr.org/2018/1212. Citations in this document: §4.2, §6.3.

[53] Yevgeniy Dodis, Thomas Shrimpton (editors), *Advances in cryptology—CRYPTO 2022—42nd annual international cryptology conference, CRYPTO 2022, Santa Barbara, CA, USA, August 15–18, 2022, proceedings, part II*, 13508, Springer, 2022. ISBN 978-3-031-15978-7. See [33].

[54] Léo Ducas, *Shortest vector from lattice sieving: A few dimensions for free*, in Eurocrypt 2018 [**85**] (2018), 125–145. URL: https://eprint.iacr.org/2017/999. Citations in this document: §4.1.

[55] Léo Ducas, Yang Yu, *Learning strikes again: the case of the DRS signature scheme*, in Asiacrypt 2018 [**94**] (2018), 525–543. URL: https://eprint.iacr.org/2018/294. Citations in this document: §4.2.

[56] Orr Dunkelman, Léo Perrin, *Adapting rigidity to symmetric cryptography: towards "unswerving" designs*, in SSR 2019 (2019). URL: https://eprint.iacr.org/2019/1187. Citations in this document: §1.1.

[57] Philippe Gaborit, *A probable security issue for LEPTON* (2017). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/FRiMuhoI3Ng/m/_-ElLDwjCAAJ. Citations in this document: §4.2.

[58] Philippe Gaborit, *OFFICIAL COMMENT: McNie* (2017). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/b_6q0ZSWJWo/m/0a5dgecuBwAJ. Citations in this document: §4.2, §4.2.

[59] Mike Hamburg, *OFFICIAL COMMENT: Round2* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/IrptAazyrko/m/JgpGBxsRAgAJ. Citations in this document: §4.1.

[60] Carmit Hazay, Martijn Stam (editors), *Advances in cryptology—EUROCRYPT 2023—42nd annual international conference on the theory and applications of cryptographic techniques, Lyon, France, April 23–27, 2023, proceedings, part V*, 14008, Springer, 2023. ISBN 978-3-031-30588-7. See [40], [75].

[61] Andreas Huelsing, Daniel J. Bernstein, Lorenz Panny, Tanja Lange, *OFFICIAL COMMENT: RaCoSS* (2017). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/vt-9e_0-BFo/m/FXhIMEA5BgAJ. Citations in this document: §4.2.

[62] Andreas Huelsing, Daniel J. Bernstein, Lorenz Panny, Tanja Lange, *OFFICIAL COMMENT: RaCoSS* (2017). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/vt-9e_0-BFo/m/a5SaCZw5BgAJ. Citations in this document: §4.2.

[63] Yuval Ishai, Vincent Rijmen (editors), *Advances in cryptology—EUROCRYPT 2019—38th annual international conference on the theory and applications of cryptographic techniques, Darmstadt, Germany, May 19–23, 2019, proceedings, part III*, 11478, Springer, 2019. ISBN 978-3-030-17658-7. See [52].

[64] Antoine Joux, Abderrahmane Nitaj, Tajjeeddine Rachidi (editors), *Progress in cryptology—AFRICACRYPT 2018—10th international conference on cryptology in Africa, Marrakesh, Morocco, May 7–9, 2018, proceedings*, Lecture Notes in Computer Science, 10831, Springer, 2018. ISBN 978-3-319-89338-9. See [24].

[65] Daniel Kales, Greg Zaverucha, *An attack on some signature schemes constructed from five-pass identification schemes*, in CANS 2020 [**67**] (2020), 3–22. URL: https://eprint.iacr.org/2020/837. Citations in this document: §4.4.

[66] Neal Koblitz, Alfred Menezes, *Critical perspectives on provable security: fifteen years of "another look" papers*, Advances in Mathematics of Communications **13** (2019), 517–558. URL: https://eprint.iacr.org/2019/1336. Citations in this document: §1.1, §1.1.

[67] Stephan Krenn, Haya Schulmann, Serge Vaudenay (editors), *Cryptology and network security—19th international conference, CANS 2020, Vienna, Austria, December 14–16, 2020, proceedings*, 12579, Springer, 2020. ISBN 978-3-030-65410-8. See [65].

[68] Tancrède Lepoint, *OFFICIAL COMMENT: LOTUS* (2017). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/KZTDQNKYeB0/m/tnCTH_MxCAAJ. Citations in this document: §6.3.

[69] Tancrède Lepoint, *OFFICIAL COMMENT: Odd Manhattan* (2017). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/FRiMuhoI3Ng/m/N655wicRCAAJ. Citations in this document: §6.3.

[70] Matthieu Lequesne, Jean-Pierre Tillich, *Attack on the Edon-K key encapsulation mechanism*, in ISIT 2018 [**1**] (2018), 981–985. URL: https://arxiv.org/abs/1802.06157. Citations in this document: §4.2.

[71] Haoyu Li, Renzhang Liu, Yanbin Pan, Tianyuan Xie, *OFFICIAL COMMENT: Compact LWE* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/_fOnZNYHdE4/m/TQXna0vsAgAJ. Citations in this document: §4.2.

[72] Vadim Lyubashevsky, *ROUND 2 OFFICIAL COMMENT: qTESLA* (2019). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/HHnavSx4f5Q/m/fRsujb9ACgAJ. Citations in this document: §4.4.

[73] Diana Maimut, Andrei-George Oprina, Damien Sauveron (editors), *Innovative security solutions for information technology and communications—13th international conference, SecITC 2020, Bucharest, Romania, November 19–20, 2020, revised selected papers*, 12596, Springer, 2021. ISBN 978-3-030-69254-4. See [44].

[74] Luciano Maino, Chloe Martindale, *An attack on SIDH with arbitrary starting curve* (2022); see also newer version [**75**]. URL: https://eprint.iacr.org/2022/1026. Citations in this document: §4.3, §4.3.

[75] Luciano Maino, Chloe Martindale, Lorenz Panny, Giacomo Pope, Benjamin Wesolowski, *A direct key recovery attack on SIDH*, in Eurocrypt 2023 [**60**] (2023), 448–471; see also older version [**74**]. URL: https://eprint.iacr.org/2023/640.

[76] Burton G. Malkiel, *A random walk down Wall Street*, W. W. Norton, 1973. ISBN 978-1324051138. Citations in this document: §7.1.

[77] Burton G. Malkiel, *Reflections on the efficient market hypothesis: 30 years later*, Financial Review **40** (2005), 1–9. Citations in this document: §7.1.

[78] Tal Malkin, Chris Peikert (editors), *Advances in cryptology—CRYPTO 2021—41st annual international cryptology conference, CRYPTO 2021, virtual event, August 16–20, 2021, proceedings, part I*, 12825, Springer, 2021. ISBN 978-3-030-84241-3. See [4], [99].

[79] Daniele Micciancio, Thomas Ristenpart (editors), *Advances in cryptology—CRYPTO 2020—40th annual international cryptology conference, CRYPTO 2020, Santa Barbara, CA, USA, August 17–21, 2020, proceedings, part III*, 12172, Springer, 2020. ISBN 978-3-030-56876-4. See [8].

[80] Dustin Moody, *Round 2 of the NIST PQC "competition": what was NIST thinking?* (2019). URL: https://web.archive.org/web/20200301223113/https://csrc.nist.gov/CSRC/media/Presentations/Round-2-of-the-NIST-PQC-Competition-What-was-NIST/images-media/pqcrypto-may2019-moody.pdf. Citations in this document: §2.1, §7.4, §7.4.

[81] Shiho Moriai, Huaxiong Wang (editors), *Advances in cryptology—ASIACRYPT 2020—26th international conference on the theory and application of cryptology and information security, Daejeon, South Korea, December 7–11, 2020, proceedings, part I*, 12491, Springer, 2020. ISBN 978-3-030-64836-7. See [12].

[82] Nadia El Mrabet, Luca De Feo, Sylvain Duquesne (editors), *Progress in cryptology—AFRICACRYPT 2023—14th international conference on cryptology in Africa, Sousse, Tunisia, July 19–21, 2023, proceedings*, 14064, Springer, 2023. ISBN 978-3-031-37678-8. See [49].

[83] National Institute of Standards and Technology, *Proposed submission requirements and evaluation criteria for the post-quantum cryptography standardization process* (2016). URL: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-draft-aug-2016.pdf. Citations in this document: §3.5.

[84] National Institute of Standards and Technology, *Submission requirements and evaluation criteria for the post-quantum cryptography standardization process* (2016). URL: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf. Citations in this document: §3.5, §7.1.

[85] Jesper Buus Nielsen, Vincent Rijmen (editors), *Advances in cryptology—EUROCRYPT 2018—37th annual international conference on the theory and applications of cryptographic techniques, Tel Aviv, Israel, April 29–May 3, 2018, proceedings, part III*, Lecture Notes in Computer Science, 10822, Springer, 2018. ISBN 978-3-319-78371-0. See [54].

[86] Lorenz Panny, *OFFICIAL COMMENT: Guess Again* (2017). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/6ShLu5WFz30/m/tUNMb3O1BQAJ. Citations in this document: §4.2.

[87] Lorenz Panny, *OFFICIAL COMMENT: RVB* (2017). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/hueBUX68DOA/m/9P0XHKLIBgAJ. Citations in this document: §4.2.

[88] Ray Perlner, *OFFICIAL COMMENT: pqsigRM* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/3l4UmEJFi6k/m/MeccVRjPAAAJ. Citations in this document: §4.2.

[89] Ray Perlner, *OFFICIAL COMMENT: pqNTRUSign* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/sWCQvB6ggBQ/m/PAHqJXS7AQAJ. Citations in this document: §6.3.

[90] Ray Perlner, *RE: OFFICIAL COMMENT: pqNTRUSign* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/sWCQvB6ggBQ/m/SeYGvXzHAQAJ. Citations in this document: §6.3, §6.3.

[91] Ray Perlner, *OFFICIAL COMMENT: LAKE* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/GfXKX5W8T4E/m/sR6GKIXZAQAJ. Citations in this document: §6.3.

[92] Ray Perlner, *OFFICIAL COMMENT: SIKE* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/waHre1yjIDw/m/NWVo0kkkAgAJ. Citations in this document: §6.3.

[93] Ray A. Perlner, John Kelsey, David A. Cooper, *Breaking category five SPHINCS$^+$ with SHA-256*, in PQCrypto 2022 [**43**] (2022), 501–522. URL: https://eprint.iacr.org/2022/1061. Citations in this document: §4.4.

[94] Thomas Peyrin, Steven D. Galbraith (editors), *Advances in cryptology—ASIACRYPT 2018—24th international conference on the theory and application of cryptology and information security, Brisbane, QLD, Australia, December 2–6, 2018, proceedings, part I*, 11272, Springer, 2018. ISBN 978-3-030-03325-5. See [34], [48], [55].

[95] Peter Y. A. Ryan, David Naccache, Jean-Jacques Quisquater (editors), *The new codebreakers: essays dedicated to David Kahn on the occasion of his 85th birthday*, Lecture Notes in Computer Science, 9100, Springer, 2015. ISBN 978-3-662-49300-7. See [29].

[96] SAFEcrypto, *NIST round 1 candidates* (2018). URL: https://www.safecrypto.eu/pqclounge/round-1-candidates/. Citations in this document: §2.1, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3, §6.3.

[97] Matt Scholl, *Post-quantum encryption: a Q&A with NIST's Matt Scholl* (2021). URL: https://web.archive.org/web/20211115191840/https://www.nist.gov/blogs/taking-measure/post-quantum-encryption-qa-nists-matt-scholl. Citations in this document: §7.4.

[98] Ron Steinfeld, *OFFICIAL COMMENT: CFPKM* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/t0IQjM-UhpU/m/IyDGDQGjAAAJ. Citations in this document: §4.2.

[99] Chengdong Tao, Albrecht Petzoldt, Jintai Ding, *Efficient key recovery for all HFE signature variants*, in Crypto 2021 [**78**] (2021), 70–93. URL: https://eprint.iacr.org/2020/1424. Citations in this document: §4.4, §4.4, §7.4.

[100] Visiting Committee on Advanced Technology of the National Institute of Standards and Technology, *NIST cryptographic standards and guidelines development process* (2014). URL: https://www.nist.gov/system/files/

documents/2017/05/09/VCAT-Report-on-NIST-Cryptographic-Standards-and-Guidelines-Process.pdf. Citations in this document: §1.1.

[101] Keita Xagawa, *OFFICIAL COMMENT: LEDAkem* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/JKWz6E25VwU/m/q0YahEsGAQAJ. Citations in this document: §4.3.

[102] Bo-Yin Yang, Daniel J. Bernstein, Tanja Lange, *OFFICIAL COMMENT: SRTPI* (2018). URL: https://groups.google.com/a/list.nist.gov/g/pqc-forum/c/3Av-P47rbq8/m/FllAs3FxAAAJ. Citations in this document: §4.2.