Comparing proofs of security for lattice-based encryption

Daniel J. Bernstein[1,2]

[1] Department of Computer Science, University of Illinois at Chicago,
Chicago, IL 60607–7045, USA
[2] Horst Görtz Institute for IT Security, Ruhr University Bochum, Germany
djb@cr.yp.to

Abstract. This paper describes the limits of various "security proofs", using 36 lattice-based KEMs as case studies. This description allows the limits to be systematically compared across these KEMs; shows that some previous claims are incorrect; and provides an explicit framework for thorough security reviews of these KEMs.

1 Introduction

It is disastrous if a cryptosystem X is standardized, deployed, and then broken. Perhaps the break is announced publicly and users move to another cryptosystem (hopefully a secure one this time), but (1) upgrading cryptosystems incurs many costs and (2) attackers could have been exploiting X in the meantime, perhaps long before the public announcement.

"Security proofs" sound like they eliminate the risk of systems being broken. If X is "provably secure" then how can it possibly be insecure?

A closer look shows that, despite the name, something labeled as a "security proof" is more limited: it is a claimed proof that an attack of type T against the cryptosystem X implies an attack against some problem P. There are still ways to argue that such proofs reduce risks, but these arguments have to account for potentially devastating gaps between (1) what has been proven and (2) security. Section 2 classifies these gaps into four categories, illustrated by the following four examples of breaks of various cryptographic standards:

- 2000 [37]: Successful factorization of the RSA-512 challenge. At the time, 512-bit RSA moduli were used for "95% of today's E-commerce on the Internet". The same attack breaks a wide variety of protocols—including "provably secure" protocols—built upon 512-bit RSA moduli.

This work was supported by the U.S. National Science Foundation under grant 1314919, by the National Institute of Standards and Technology under grant 60NANB12D261, by DFG Cluster of Excellence 2092 "CASA: Cyber Security in the Age of Large-Scale Adversaries", and by the Cisco University Research Program. "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: 4c6385d1a904c0a83bc9fe9ab8651dcc456ef7db. Date: 2019.07.19.

system	parameter set	ROM	ROM2	fail	conj	dist	P
`frodo`	`640`	risk	risk	−138	safe	risk	risk
`frodo`	`976`	risk	risk	−199	safe	risk	risk
`frodo`	`1344`	risk	risk	−252	safe	risk	risk
`kyber`	`512`	risk	risk	−178	safe	risk	risk
`kyber`	`768`	risk	risk	−164	safe	risk	risk
`kyber`	`1024`	risk	risk	−174	safe	risk	risk
`lac`	`128`	risk	risk	−133	risk	risk	risk
`lac`	`192`	risk	risk	−142	risk	risk	risk
`lac`	`256`	risk	risk	−138	risk	risk	risk
`newhope`	`512`	risk	risk	−213	risk?	risk	risk
`newhope`	`1024`	risk	risk	−216	risk?	risk	risk
`ntru`	`hps2048509`	risk	safe	−∞	safe	safe	risk
`ntru`	`hps2048677`	risk	safe	−∞	safe	safe	risk
`ntru`	`hps4096821`	risk	safe	−∞	safe	safe	risk
`ntru`	`hrss701`	risk	safe	−∞	safe	safe	risk
`ntrulpr`	`653`	risk	risk	−∞	safe	risk	risk
`ntrulpr`	`761`	risk	risk	−∞	safe	risk	risk
`ntrulpr`	`857`	risk	risk	−∞	safe	risk	risk
`round5n1`	`1`	risk	risk	−146	risk	risk	risk
`round5n1`	`3`	risk	risk	−144	risk	risk	risk
`round5n1`	`5`	risk	risk	−144	risk	risk	risk
`round5nd`	`1.0d`	risk	risk	−155	risk	risk	risk
`round5nd`	`3.0d`	risk	risk	−147	risk	risk	risk
`round5nd`	`5.0d`	risk	risk	−143	risk	risk	risk
`round5nd`	`1.5d`	risk	risk	−142	risk	risk	risk
`round5nd`	`3.5d`	risk	risk	−256	risk	risk	risk
`round5nd`	`5.5d`	risk	risk	−227	risk	risk	risk
`saber`	`light`	risk	risk	−120	risk?	risk	risk
`saber`	`main`	risk	risk	−136	risk?	risk	risk
`saber`	`fire`	risk	risk	−165	risk?	risk	risk
`sntrup`	`653`	risk	safe	−∞	safe	safe	risk
`sntrup`	`761`	risk	safe	−∞	safe	safe	risk
`sntrup`	`857`	risk	safe	−∞	safe	safe	risk
`threebears`	`baby`	risk	risk	−156	risk	risk	risk
`threebears`	`mama`	risk	risk	−206	risk	risk	risk
`threebears`	`papa`	risk	risk	−256	risk	risk	risk

Table 1.1. Security risks that are not eliminated by “security proofs” for the target KEMs, even assuming that the proofs are correct. “ROM”, “ROM2”, “fail”, “conj”, “dist”, and “P” are risks; see text for definitions. Red entry (“risk”): Security risk is not eliminated for this KEM. Black entry (“safe”): Security risk is eliminated for this KEM.

- 2016 [31]: A successful break of "provably secure" Triple-DES-CBC. At the time, Triple-DES-CBC was used for "roughly 1–2% of HTTPS connections between mainstream browsers and web servers".
- 2018 [53]: A successful break of "provably secure" AES-OCB2. The AES-OCB2 "security proof" was published at Asiacrypt 2004, and AES-OCB2 was standardized by ISO, although deployment of AES-OCB2 was limited by patent issues.
- 2019 [69]: A successful break of various "provably secure" signature systems using SHA-1.

As these examples illustrate, "security proofs" do not eliminate all risks. This fact is important for the evaluation of "provably secure" systems that are not (yet?) known to be broken: the details of the risk analysis depend on, inter alia, the attack type T considered by the proofs, and the underlying problem P assumed to be secure. This dependence also means that some pairs (P, T) could be more effective than others in reducing risks.

1.2. A plan for security reviews of 36 target KEMs, and a metric for comparing proofs. The purpose of this paper is to make clear which pairs (P, T) are achieved by known proof strategies for a selected list of target KEMs X defined below. The detailed list of pairs (P, T) is the centerpiece of the following three-step plan for thoroughly evaluating risks that the target KEMs do not reach their claimed security levels:

- Security reviewer #1: Is each X in fact fully proven to reach its claimed security level against attacks of type T, assuming the hardness of P?
- Security reviewer #2: To what extent have attacks against P been studied? How confident are we that P is as hard as the proof assumes?
- Security reviewer #3: To what extent have attacks outside T been studied? How confident are we that these cannot do better than attacks of type T?

These three reviews can be carried out in parallel, although errors discovered in review #1 can force changes in the scope of the other two reviews.

The total cost of a thorough security review—including all three steps shown above—is the primary metric used in this paper to compare proofs of security. The underlying theory here is that security problems are generally more likely to remain undetected in systems where thorough security reviews are more complicated. More complicated reviews mean more opportunities for error, especially when constraints on community resources limit the time available for review. See Appendix B for further comments on this metric, and Appendix C for other ways to evaluate proofs.

One should not think that this paper constitutes a security review. Describing full details of the known attacks and of the remaining attack surface would require a much longer document. I have reviewed the relevant proofs in enough detail to confidently point out various errors in proofs and in previous claims regarding the list of pairs (P, T), but this does not mean that I vouch for the correctness of the remaining proofs; further review could reveal proof errors that force the list to be revised. I could have made mistakes as part of analyzing the submissions, and further mistakes as part of unifying notation for the list of pairs (P, T). I plan to issue online updates of this paper as appropriate.

1.3. The target KEMs, and a table of security risks. I selected target KEMs by the following procedure. Take the lattice-based KEM families in round 2 of NIST's Post-Quantum Cryptography Standardization Project: these are, in alphabetical order, Frodo [6], Kyber [14], LAC [71], NewHope [4], NTRU [39], NTRU Prime [27], Round5 [15], and Saber [44]. Also include ThreeBears [51], which is sometimes counted as a lattice proposal. Then list, within each KEM family, all parameter sets identified in the submissions as aiming for IND-CCA2 security.[3] These are, by definition, the target KEMs.

Some of the submissions also provide options that aim merely for IND-CPA security. However, I have not seen any publicly verifiable examples of applications where the extra cost of IND-CCA2 security is a significant part of the end user's total costs. I am focusing on IND-CCA2 security as the most important goal of the proofs and the obvious target for comparison.

Even though the KEMs aim for IND-CCA2 security, the proofs advertised for these KEMs are limited to QROM IND-CCA2 attacks, or further limited to ROM IND-CCA2 attacks. QROM IND-CCA2 proofs have been rapidly improving over the past two years, and it seems more productive to expend QROM efforts on further proof improvements than on analyzing the current status, so I have decided to focus on the narrower class of ROM IND-CCA2 attacks here.[4]

Table 1.1 lists security risks that are *not* eliminated by the known "security proofs" for these KEMs. The "ROM", "ROM2", "fail", "conj", "dist", and "P" risks are defined in Sections 4, 4, 5, 5, 6, and 8 respectively. Giving a complete list of the pairs (T, P) requires defining the underlying problems P; this is done in Section 8.

1.4. Previous work. This is a systematization-of-knowledge paper that draws heavily upon previous work, including hundreds of pages of KEM submission documents and related papers. I do not claim any novelty for the specific proof techniques mentioned in this paper.

It is generally difficult for readers of the previous literature to see all the *gaps* in what has been proven regarding the target KEMs. Notes regarding the gaps for any particular KEM are organized haphazardly, often not highlighted, often buried under other material, and sometimes obscured by errors in the content. KEMs also vary in notation and terminology for describing proofs, adding further difficulties in comparing the details across KEMs.

[3] Round5 defines KEMs that aim for IND-CCA2 security, but my understanding is that (for some reason) these are defined only as internal building blocks for PKEs, not as KEMs provided to users. As far as I can tell, the PKE wrapper is orthogonal to the question of how solidly proofs guarantee IND-CCA2 security for the KEMs.

[4] However, I do have a few preliminary QROM footnotes.

For example, the Frodo submission[5] lists a chain of proofs "supporting the security of FrodoKEM". This chain begins with [**6**, Theorem 5.1]. The theorem statement occupies 8 lines, plus 24 lines via [**6**, Definition 2.19], which has various details not shared with other submissions (even though at first glance the theorem statement sounds much more general than Frodo). Someone trying to check the proof finds nothing beyond a very short outline. Someone trying to figure out the underlying problem P—the problem assumed to reach a particular security level—is told that the problem is to break OW-Passive ("OW-CPA") security of a PKE. This sounds clear enough modulo details of the PKE, but, as far as I can tell, this is *not* what has actually been proven; there is a hidden risk of IND-CPA attacks being much faster than OW-Passive attacks. See Section 6.

"Security proofs" indisputably have an impact upon the critical processes of designing, evaluating, comparing, and selecting cryptosystems. The cryptographic literature frequently claims that these proofs reduce risks for particular cryptosystems. Design rationales and comparisons frequently refer to these claims. However, the community generally does not require these claims to meet basic scientific standards of clarity, falsifiability, justification, reproducibility, etc. Here are two examples from lattice-based cryptography:

- Peikert claimed in 2017 [**86**] that Ring-LWE is "at least as hard to break" as NTRU; see also [**85**, page 33] ("Ring-LWE is at least as hard as NTRU"). This claim communicates two levels of incorrect information to the reader:
  - "NTRU" normally refers to various cryptosystems. The typical reader thinks that Peikert is comparing these cryptosystems to an alternative often called "Ring-LWE cryptosystems". However, Table 1.1 shows that every "Ring-LWE" cryptosystem under consideration by NIST actually carries risks that are eliminated by some NTRU cryptosystems.
  - From context one can deduce that Peikert is actually focusing on an underlying problem sometimes called "the NTRU problem", and comparing this to the "Ring-LWE" problems underlying the "Ring-LWE" cryptosystems, while ignoring other cryptosystem risks. However, this focus still does not justify Peikert's claim: these "Ring-LWE" problems carry risks that are avoided by the "NTRU" problem. See Section 7.3.
- Stehlé claimed in 2019 that "NTRU LPRime does not enjoy a security proof that is analogous to that of the LPR scheme". Peikert [**87**] claimed that it was "clear" that Stehlé was referring to "the average-case Ring-LWE problem" (i.e., the proof is required to start from the assumption that certain "Ring-LWE" parameters are hard to break). However, Peikert's interpretation of Stehlé's claim categorically excludes *all* of the pure rounding KEMs (e.g., `firesaber`), contradicting another claim by Stehlé [**99**] that the exclusion is "due to" details specific to NTRU LPRime. Stehlé has refused to answer my clarification questions.

Of course, one can argue about the proper methodology for assigning weights to various types of proofs (and other cryptosystem features). These arguments

[5] I am selecting Frodo as an example here because Frodo seems to place heavier emphasis upon proofs than any other round-2 lattice-based KEM submission.

should, however, begin with a clear picture of the objective facts—in particular, a clear picture of what has been proven and what has not been proven. This paper systematically collects the facts.

Cryptography is notoriously fragile: the community has seen again and again how a single isolated error can destroy security. This does not imply that security will be destroyed specifically by errors in understanding how much has been proven—and how much has *not* been proven—regarding lattice-based cryptography; but anecdotal evidence provides many reasons to be concerned. I think all of the following statements are reasonable extrapolations from the evidence: (1) These errors are common. (2) These errors are often severe. (3) These errors are not distributed equally across KEMs. (4) These errors are almost always in the direction of thinking that the proofs guarantee *more* than they actually do, rather than less. (5) These errors tend to reduce the decision-making impact of security features other than proofs—features that again are not distributed equally across KEMs. (6) Errors in understanding what has been proven can easily lead to selection of an unnecessarily risky KEM.

2 Classification of risks in "provably secure" cryptosystems

> *Proofs of security give an iron-clad guarantee—relative to the definition and assumptions—that no attacker will succeed; this is much better than taking an unprincipled or heuristic approach to the problem. Without a proof that no adversary with the specified resources can break some scheme, we are left only with our intuition that this is the case. Experience has shown that intuition in cryptography and computer security is disastrous. There are countless examples of unproven schemes that were broken, sometimes immediately and sometimes years after being developed.* —Katz and Lindell [60]

The standard argument for the value of "security proofs" is the claim that they "guarantee" the security of various cryptosystems. In this paper, I will ignore the ill-defined marketing ("iron-clad" vs. "unprincipled" etc.), and focus on the logical structure of the argument that these cryptosystems are secure.

Write X for the cryptosystem in question, and write 2^λ (e.g., 2^{128}) for X's target security level. Here is the general argument that X has security level at least 2^λ:

- Any attack A against X must be of type T. (This is not proven—it is the "definition" of the type of attack under consideration.)
- By the theorem, this attack A implies an attack B against P. (This is the proven part—the "guarantee".)
- P reaches security level 2^λ; i.e., B costs at least 2^λ. (This is also not proven—it is the "assumption".)
- Ergo, A costs at least 2^λ.

There are four important ways that this type of argument can fail, corresponding to the four proof-quality measurements listed in [**26**, Section 6.1]:

- **Risk #1: Security failures in the underlying problem $\boldsymbol{P}$.** The third step in the argument fails if attacks against P actually cost below 2^λ. This is what happened in the successful break of various "provably secure" protocols using 512-bit RSA moduli N: the underlying problem of inverting exponentiation modulo N does not reach the security level that users want.
- **Risk #2: Looseness in the proof.** A closer look shows that most "security proofs" do not say that B is as fast as A: they create B that is (say) 2^ℓ times slower than A. The fourth step in the argument then fails: the conclusion is only security $2^{\lambda-\ell}$, which could be much smaller than 2^λ. This is what happened in the successful break of "provably secure" Triple-DES-CBC: all known Triple-DES attacks are quite expensive, and a known proof built a Triple-DES attack from any Triple-DES-CBC attack, but the proof was quantitatively too loose to rule out a fast Triple-DES-CBC attack.
- **Risk #3: Attacks outside type $\boldsymbol{T}$.** The first step in the argument fails if an attacker finds a faster attack against X that is not of type T. This is what happened in the successful break of various "provably secure" signature systems using SHA-1: the proofs applied only to ROM attacks, and the successful break was a non-ROM attack.
- **Risk #4: Errors in the proof.** The second step in the argument fails if the proof is incorrect. This is what happened in the successful break of "provably secure" AES-OCB2.

Accounting for all of these risks produces the following argument that X has security level at least 2^λ:

- Assume that there is an attack A against X that costs below 2^λ.
- Assume that A is of type T. (Risk #3 is that this is not true.)
- Assume that the proof is correct. (Risk #4 is that this is not true.)
- Then, by the proof, there is an attack B against P that costs below $2^{\ell+\lambda}$.
- Assume that P has security level at least $2^{\ell+\lambda}$. (Risk #1 is that P does not even reach security level 2^λ, and risk #2 is that P does not reach security level $2^{\ell+\lambda}$.)
- This is a contradiction, so at least one of the assumptions must be wrong.

For simplicity I have been treating the security level of a cryptosystem as a single number in the above description: X has target security level 2^λ, and P has target security level $2^{\ell+\lambda}$. Analogous comments apply to a more sophisticated notion of security level as a relation between attack cost and attack success probability. Below I will distinguish cost from probability.

As a concrete example (much simpler than the lattice examples later in this paper), there are "security proofs" for AES-CMAC, AES-EAX, AES-GCM, AES-OCB, AES-OCB2, AES-OCB3, and many other symmetric cryptosystems built from AES. These proofs leave the following risks:

- The proofs begin from an "SPRP" security assumption about AES (or, in some cases, a generically weaker "PRP" security assumption). Risk #1 is that AES does not actually reach the target SPRP security level.
- Each proof has some loss in success probability, typically on the scale of $q^2/2^{128}$ where q is the number of AES invocations. Risk #2 is that q reaches the scale of 2^{64}, making the conclusions content-free.
- Typically these proofs cover *all* MAC attacks, AEAD attacks, etc. under standard definitions, so risk #3 does not apply. (On the other hand, sometimes the standard definitions are argued to be too narrow.)
- The proofs usually have not been computer-verified, and sometimes turn out to be wrong. Risk #4, the risk of proof errors, is illustrated by the break of AES-OCB2.

One way to see that these proofs are useful, despite the continued risk of attacks, is as follows. Consider the review-cost metric from Section 1, the time required for a thorough security review. The SPRP assumption for AES has a simpler attack surface from a cryptanalyst's perspective than the AEAD security assumption for AES-GCM. Even better, the work of studying the SPRP assumption for AES is reused for AES-CMAC and many other systems, thanks to the proofs. Without the proofs, cryptanalysts would have to spend much more time searching for attacks against each system. Some of the cryptanalysis time saved by the proofs has to be spent verifying the proofs, but overall there still seems to be a savings of time.

2.1. Chains of lattice proofs, and organization of this paper. The known proof strategies relevant to the target KEMs generally take short steps down the following list of problems:

- IND-CCA2 security of the KEM. See Section 3.
- ROM IND-CCA2 security of the KEM. See Section 4.
- IND-CPA security of the underlying PKE. See Section 5.
- OW-Passive ("OW-CPA") security of the PKE. See Section 6.
- For "Product NTRU" using public seeds to create pseudorandom multipliers: ROM security of the PKE, followed by security of a simplified PKE where multipliers are chosen at random. This is included in Section 4.
- Separate problems for keys and ciphertexts. See Section 7.
- Lattice problems. See Section 9.

Each of these steps adds its own risks of proof errors; often a step adds its own looseness issues; often a step adds its own restrictions upon the type of attacks under consideration. Furthermore, it is often not clear that one step will plug into the next step. Sometimes single steps are stated as formal theorems, but this is not the same as a grand unified end-to-end theorem that states the cumulative looseness, all of the restrictions upon the attack type, and the exact underlying problem for each target KEM. Many of the risks identified in this paper are at the interfaces between steps, and I would not be surprised if doing the work to state and prove an end-to-end theorem identifies even more risks.

3 A proof that applies to all possible attacks

Each of the target KEMs X has a tight "security proof" that applies to the entire class T of IND-CCA2 attacks (not just ROM IND-CCA2 attacks). The proof has been carefully verified. The underlying assumption P is something quite plausible, namely the IND-CCA2 security of X.

I did not put serious effort into obfuscating the previous paragraph. I expect the reader to see immediately how content-free this "security proof" is, and to wonder why I have selected such a useless starting point for my analysis of lattice "security proofs". This example is not new; see generally Appendix C.

My objective in this section is to identify a robust method for decision-makers to systematically reject this type of "security proof". Later sections will consider what this means for various proofs advertised by the target KEMs.

One possibility is to reject any proof where P is exactly the T security of X. However, this is not robust against minor changes to P. There are many easy ways to replace P with a marginally obfuscated problem P', at the expense of one line in the proof.

Another possibility is to reject any short proof: say, any proof below 10 lines. But this is again not robust: one can easily obfuscate P heavily enough that a useless proof occupies more than 10 lines.[6] Furthermore, anecdotal evidence suggests that proofs are more likely to be checked if theorems are factored into small, separately verifiable pieces; from this perspective, short proofs are good, and rejecting short proofs creates a bad incentive to avoid factoring theorems.

What I instead recommend is using the review-cost metric from Section 1. As a baseline, if there are no proofs, then cryptanalysts must study the security of X. For comparison, if there is a proof where P is exactly the security of X, then cryptanalysts are still required to study the security of X, so no time has been saved. As another example, if there is a proof where P is an obfuscated version of the security of X, then cryptanalysts need more time—first strip away the obfuscation, then study the security of X—and someone must also take the time to check the proof.

The situation is different in, e.g., Section 5. In that section, T allows the attacker to issue decapsulation queries, while P does not. It takes considerably less time for cryptanalysts to gain confidence in the security of P than it would take for cryptanalysts without the proof to gain the same level of confidence in the security of the original problem. This savings could still be outweighed by other costs—the proofs mentioned in Section 5 are limited to ROM IND-CCA2 attacks (so cryptanalysts need to take time to consider the possibility of non-ROM attacks) and take effort to review—but the core fact that P takes less time to review gives Section 5 a chance of being useful in the review-cost metric.

More generally, in planning a review of risks that X does not reach its claimed security level, one has a choice of which proofs to use and which proofs to ignore.

[6] See, e.g., the Frodo proof covered in Section 4.1. Obfuscation is the predictable result of evolutionary pressure upon "security proofs"; I do not mean to suggest that any malice is involved.

Each extra proof requires time to review, but could simplify cryptanalysis enough to save time overall.

For comparison, Damgård [42] asks for problems to be as "simple" and "natural" as possible, but does not describe a way to measure this. If some way to measure simplicity says that P is simpler than X, but cryptanalysts need more time to analyze P than to analyze X, then the review-cost metric says that the proof is not useful. Damgård also asks for problems to be as "well studied" as possible; this is the natural result of reducing the cost of review.

4 Random oracles

> *The Random Oracle Model has caused more harm than good, because many people confuse it for the "real thing" ... At the very minimum, one should issue a fierce warning that security in the Random Oracle Model does not provide any indication towards security in the standard model.* —Goldreich [49]

> *There is no evidence that the need for the random oracle assumption in a proof indicates the presence of a real-world security weakness in the corresponding protocol. We give several examples of attempts to avoid random oracles that have led to protocols that have security weaknesses that were not present in the original ones whose proofs required random oracles.* —Koblitz and Menezes [63]

Subsequent sections focus on proofs for ROM IND-CCA2 attacks. The proofs do not eliminate the risk of non-ROM IND-CCA2 attacks; this risk[7] is the "ROM" column in Table 1.1. This column is marked "risk" for each of the target KEMs. For each of these KEMs, the only known proofs that apply to *all* IND-CCA2 attacks are useless proofs (see Section 3).

The premier examples of non-ROM attacks are signature forgeries via chosen-prefix collisions in MD5 and SHA-1. One can also build artificial public-key systems that seem safe against all ROM attacks but allow non-ROM attacks with, e.g., SHA-512. It is not inconceivable that such attacks also exist against reasonable public-key systems. Presumably these attacks would combine attack techniques from symmetric cryptology and public-key cryptology; does the lack of known attacks mean that the attacks do not exist, or does it mean that the attacks exist and nobody has been looking for them? Note that there are many cryptanalysts specialized in symmetric cryptology, many others specialized in public-key cryptology, and very few who are experts in both.

4.1. Random multipliers. All of the target KEMs prove ROM IND-CCA2 security from IND-CPA security of an underlying PKE (Section 5) or, in some cases, OW-Passive security of an underlying PKE (Section 6). However, some of the KEMs then further restrict attention to ROM attacks against the underlying

[7] This can be divided into (1) the risk of QROM attacks much faster than ROM attacks, and (2) the risk of attacks much faster than QROM attacks.

PKE. The risk of a non-ROM attack against the PKE is marked as the "ROM2" column in Table 1.1. This risk is distinct from the risk of a non-ROM attack against the KEM: any particular hash function might trigger one type of non-ROM attack and not the other. Cryptanalysts searching for non-ROM attacks against the KEM are faced with a different problem from cryptanalysts searching for non-ROM attacks against the PKE.

The issue with these PKEs is that they generate public multipliers as cipher output or hash output starting from short public seeds.[8] The underlying problems highlighted by these PKEs instead choose the multipliers at random from a distribution with a reasonably simple mathematical definition, certainly simpler than the definition of a cipher or hash function. This mismatch means that, even if the underlying problem is secure, there is a risk of a much faster attack against the PKE—an attack that exploits the pseudorandom multipliers.

This risk is often hidden by expository failures or outright proof errors. For example, the Saber submission uses a PKE of this type, and claims [**44**, Theorem 6.1] to prove IND-CPA security—not just ROM IND-CPA security—from a "prf" assumption and two "mod-lwr" assumptions.[9] A proof is not given, and a skeptical reviewer finds somewhat different notation and theorem structure in [**43**], but the core point here seems to be [**43**, Theorem 3, proof, second paragraph]. The paragraph briefly claims that an attacker can win a PRG game, distinguishing a pseudorandom matrix from a uniform random matrix, if the attacker can distinguish these matrices in the context of an attack against a more complicated problem. I see no way that this can be proven for IND-CPA attacks against the PKE, or other types of attacks that show the seed to the attacker: the PKE attacker can simply try hashing the seed, while the PRF/PRG attacker cannot.

The Frodo submission, instead of simply stating that it is limiting attention to ROM attacks against the PKE, presents a page-long argument [**6**, Section 5.1.4] (also [**7**, Section 5.1.3]) claimed to be a "reduction" between uniform random multipliers and pseudorandom multipliers. This "reduction" is not encapsulated as a theorem, and considerable effort is required for the reader to see that this "reduction" is an obfuscated version of the following content-free statement:

- Assume that there is an attack against Frodo's pseudorandom multipliers.
- Assume that the attack is just as effective against uniform random multipliers as Frodo's pseudorandom multipliers.
- Then there is an attack against uniform random multipliers.

This does not eliminate the risk of non-ROM attacks faster than ROM attacks; it simply hypothesizes that non-ROM attacks do not exist. This "reduction" is a loss in the review-cost metric: it consumes time for reviewers, without providing any improvement of the attack surface presented to cryptanalysts.

[8] Otherwise the public keys would be bigger.

[9] The literal theorem statement is correct but useless: it does not define the reductions and does not otherwise put any constraints on the cost of the PRF attack, so one can plug in a very slow high-probability PRF attack. I am presuming that the theorem statement will be corrected to add reasonable cost limits.

5 Decryption failures

Each of the target KEMs is built by combining (1) a simpler underlying PKE and (2) a "CCA conversion" designed to protect against chosen-ciphertext attacks.

A paper by Hofheinz, Hövelmanns, and Kiltz [52] presents several CCA conversion options in a unified and generalized way, proving ROM IND-CCA2 security of each of the resulting KEMs assuming IND-CPA security of the underlying PKE. All of the target KEMs seem to be covered by the proof strategies in the paper. Sometimes there are modifications (e.g., minor tweaks to hashing) that take the KEMs outside the theorems from [52], but the proof strategies seem robust to these tweaks, modulo small changes in probability formulas.[10]

The risk of non-ROM IND-CCA2 attacks was covered in Section 4. This section covers a different risk that appears when one analyzes the tightness of these proofs. The IND-CCA2 success probabilities shown in [52, page 21, table, rightmost column] have the following terms:

- The IND-CPA success probability against the KEM, multiplied by 3.
- For the first three options: $3Q/\#\mathcal{M}$, where Q is the number of hash queries, and $\#\mathcal{M}$ is the size of the plaintext space allowed by the PKE. Most of the target PKEs have $\#\mathcal{M} \ge 2^{256}$, and then $3Q/\#\mathcal{M}$ can safely be ignored.
- For the second and fourth options ("explicit rejection"): $Q/2^\gamma$ when the PKE is "γ-spread". It seems clear that whatever is standardized will use implicit rejection (almost all of the target KEMs use implicit rejection; ThreeBears uses explicit rejection but appears to be adding support for implicit rejection), so there is little point in commenting further upon this issue.
- $Q\delta$ or $(2Q+q)\delta$ (depending on the option), where q is the number of decryption queries, and δ is the decryption failure probability. This can compromise tightness, depending on the size of δ.

The rest of this section focuses on the loss of tightness from decryption failures.

5.1. Risk classification. How large can $Q\delta$ and $(2Q+q)\delta$ be? The number of decryption queries q is limited by communication with the legitimate user; NIST allows[11] submissions to assume $q \le 2^{64}$. The number of hash calls Q could be much larger: it is limited only by the attacker's computational power. This still allows a tight ROM proof[12] if δ is *proven* to be sufficiently small.

One risk here is that δ is too large. The "fail" column in Table 1.1 shows the logarithm, base 2, of the upper bounds on δ claimed by the submissions. The column shows $-\infty$ if δ is claimed to be 0.

[10] I am not saying that all of the proofs have been written down and verified. My paper [30] with Persichetti presents counterexamples to two of the main theorems from [52], and it is possible that there are further problems. However, at the moment I don't see reasons to believe that the target KEMs are affected by any such problems.

[11] On the other hand: "NIST is open to considering attacks involving more queries, and would certainly prefer algorithms that did not fail catastrophically if the attacker exceeds 2 to the 64 queries." See [81].

[12] For the QROM context, current theorems such as [33, Lemma 6] generally multiply $Q\delta$ by the depth of the attacker's computation, reflecting the usual Grover speedup.

Often the KEM claims a pre-quantum security level of (e.g.) 2^{256} but does not claim that δ is as small as 2^{-256}. Tightness fails in such cases. Even under the assumption that the PKE reaches the claimed IND-CPA security level, the proof does not rule out the risk of a much faster IND-CCA2 attack against the KEM; the claim of a tight proof is thus incorrect in these cases.[13] I have marked the "fail" column in red in these cases.

Another risk is that the claimed upper bounds on δ turn out to be wrong. There are two different scenarios here:

- The claimed upper bounds are theorems. This risk is then the general risk of errors in proofs; Table 1.1 explicitly assumes that the proofs are correct.
- The claimed upper bounds are not theorems: they are merely conjectures (with varying levels of evidence). This risk is then a different type of risk that would not be eliminated by verification of all theorems.

The "conj" column in Table 1.1 is "risk" if the claimed upper bounds are conjectures, and "safe" if the claimed upper bounds are theorems.

Logically, "conj: risk" means that ROM IND-CCA2 security of the KEM has *not* been proven tightly from IND-CPA security of the underlying PKE: there is a gap in the proof, specifically in the claim that δ is sufficiently small. This gap is often not clear from submissions: the reader sees an explicit theorem statement where δ is a variable, and the reader tends to assume—incorrectly—that this leads to a theorem for the particular KEMs that have been proposed.

5.2. Subtleties in the definition of failure probability. The literature contains at least three different definitions of (upper bounds on) failure probability:

- Encrypt a random message to a random public key. The probability that decryption fails is (at most) δ.
- Take a message selected by an attacker. Encrypt this message to a random public key. The probability that decryption fails is (at most) δ.
- Show a random *public key and secret key* to an attacker. Take a message selected by the attacker, and encrypt it to this public key. The probability that decryption fails is (at most) δ.

The third definition is used for the proofs in [52]. The second definition was stated in a preliminary version of [52]. The first and second definitions do not seem to be compatible with any known ROM IND-CCA2 proof strategies.

I have marked "conj: risk?" in Table 1.1 if a KEM submission indicates that it has proven upper bounds on failure probabilities, but does not state that this proof uses the third definition. The question mark recognizes two ways that the proof status might turn out to be better than "conj: risk": (1) The proof authors might have been using the third definition. (2) The underlying proof strategy might be able to handle the third definition in any case. (This depends on the PKE details.) If the authors add clear theorems bounding δ under the third definition then I will be happy to update the table to say "conj: safe".

[13] Various KEM submissions observe, correctly, that *known* attacks do not exploit this looseness. However, in this paper I am focusing on what has been proven.

The current lack of clarity regarding what has supposedly been proven is a deterrent to review of the proofs, as the following scenario illustrates:

- A reviewer is evaluating the entire picture of what has been proven and deciding which proofs are most urgent to review.
- A claimed proof of ROM IND-CCA2 security from IND-CPA security depends on a claimed δ proof.
- The reviewer decides to slog through the details of the δ proof, and eventually discovers that a step in the proof misses the possibility of an attacker choosing a failing message based upon the secret key.
- The proof authors respond that they were using the first definition.

This scenario wastes a considerable amount of review time.[14] If there had been, from the outset, a proper theorem statement using the first definition, then the reviewer could have skipped the proof details and simply pointed to the gap in definitions as a gap in the claimed ROM IND-CCA2 proof.

5.3. Other sources of IND-CCA2 non-tightness. The theorems from [52] state that the resulting IND-CPA attack takes "about" the same time as the given IND-CCA2 attack. This violates the usual mathematical requirement for each statement in a theorem to have a clear definition.

My impression is that the main cost in the IND-CPA attack, beyond the cost of the IND-CCA2 attack, is the cost of maintaining a database of queries. This can be a problem in realistic cost models that ask how quickly an attacker can carry out a computation using a specified amount of hardware. Perhaps this problem can be eliminated by the techniques suggested in [20] and [13].

6 One-wayness

> *What concerns us about the DDH assumption is the fact that this assumption refers to a setting that is less simple than usual (e.g., DDH is less simple than DH), which makes this assumption harder to evaluate.* —Goldreich [49]

Some KEMs have a tight proof of ROM IND-CCA2 security assuming merely OW-Passive ("OW-CPA") security of the underlying PKE. The proof strategy was introduced by Persichetti [89], generalized by Saito–Xagawa–Yamakawa [95], and modularized for verification in my paper [30] with Persichetti. The resulting theorem has two requirements:

- The PKE is deterministic: i.e., the entire randomness used to produce a ciphertext (aside from the public key) is the message recovered by decryption. This seems essential for the proof strategy.

[14] Perhaps the reviewer can still provide useful information to cryptanalysts or to reviews of other proofs, but this possibility does not noticeably affect the reviewer's evaluation of the costs and benefits of reviewing a proof.

- The PKE has no decryption failures. It seems reasonable to guess that this requirement can be dropped at the cost of the tightness loss explained in Section 5, but this does not matter for any of the target KEMs: the target PKEs that are designed to be deterministic are also designed to have no decryption failures.

Other aspects of tightness of the proof are even better than in Section 5: for example, the OW-Passive success probability is not multiplied by 3. I do not mean to draw excessive attention to security levels changing by 1 or 2 bits; my main concern in this paper is with larger risks.

In Table 1.1, “dist: risk” means that there is a risk of ROM IND-CCA2 attacks being much faster than OW-Passive attacks, beyond the decryption-failure risk from Section 5. This extra risk occurs when the underlying PKE is randomized: the tight ROM IND-CCA2 proof needs to assume IND-CPA security and does not rule out IND-CPA attacks much faster than OW-Passive attacks. The central issue is Fujisaki–Okamoto derandomization,[15] which chooses the randomness in encryption as a hash of the message that will be recovered by decryption. Even within the limited class of ROM attacks, no known proof strategies eliminate the risk of this pseudorandomness being exploited by an attacker:

- These KEMs have a *loose* proof of ROM IND-CCA2 security from OW-Passive; see [**52**]. The looseness factor is $2Q$, where Q is the number of hash queries. This does not eliminate the risk of ROM IND-CCA2 attacks being much faster than OW-Passive attacks.
- There is a construction [**52**, Section 3.4] that achieves ROM IND-CPA for the constructed PKE tightly from OW-Passive for the original PKE. None of the target KEMs use this construction.[16]
- Known “search-to-decision reductions” are frequently summarized as showing that distinguishing attacks are as difficult as search attacks. However, checking the details shows that this summary is an overstatement. These theorems do not eliminate the risk of ROM IND-CCA2 attacks being much faster than OW-Passive attacks.

Most of the target KEMs use randomized PKEs and thus carry this risk. The only exceptions (“dist: safe”) are the four KEMs from the NTRU submission (`ntru`) and the three Streamlined NTRU Prime KEMs (`sntrup`).

[15] Derandomization seems to be an even larger problem for QROM proofs. For the deterministic target KEMs, the latest QROM proofs [**33**] obtain IND-CCA2 tightly from OW-Passive, modulo the tightness issues from Section 5. For the randomized target KEMs, I am not aware of any tight QROM IND-CCA2 proofs, even if one assumes IND-CPA security for the underlying PKEs. But let me emphasize that the QROM proof situation is rapidly improving, and further QROM proofs could close this gap.

[16] The construction produces much larger ciphertexts, and thus cannot be used by lattice designers aiming for keys and ciphertexts around a kilobyte. The only target KEMs that allow much larger ciphertexts are the `frodo` and `round5n1` KEMs, and these KEMs still do not use this construction.

Internally, the target KEMs use two different types of public keys, corresponding to two different mechanisms of encryption and decryption. In the NTRU Prime terminology, the `ntru` and `sntrup` KEMs are "Quotient NTRU" KEMs where the public key is obtained as a quotient of small polynomials; the encryption details make it reasonably easy for the receiver to recover the entire randomness that was used to produce the ciphertext. The other 29 KEMs—the `ntrulpr` KEMs, and the KEMs not named after NTRU—are "Product NTRU" KEMs following the general approach of [**72**]; here it seems difficult to recover the entire randomness. The Product NTRU KEMs all handle this by Fujisaki–Okamoto derandomization.

6.1. Eliminating risks of distinguishers. The distinction between IND-CPA and OW-Passive is standard in provable security.[17] IND-CPA tightly implies OW-Passive for any PKE with a large plaintext space.[18] In the opposite direction, OW-Passive is not believed to imply IND-CPA; for example, a deterministic OW-Passive scheme is not IND-CPA.

Consequently, assuming that the underlying PKE provides IND-CPA involves risks that are not present in merely assuming that the PKE provides OW-Passive. This is (modulo an exchange of "first" and "second") an example of the following situation described by Katz and Lindell [**60**]:

> If the assumption on which the first scheme is based is *weaker* than the assumption on which the second scheme is based (i.e., the second assumption implies the first), then the first scheme is preferable since it may turn out that the second assumption is false while the first assumption is true.

This does not imply that assuming OW-Passive for one PKE is safer than assuming IND-CPA for a *different* PKE: the differences between the PKEs can outweigh the gap between OW-Passive and IND-CPA. As Katz and Lindell write:

> If the assumptions used by the two schemes are not comparable, then the general rule is to prefer the scheme that is based on the better-studied assumption in which there is greater confidence.

Furthermore, the mere fact that one assumption is weaker than another does not imply that the gap is important. However, there are several ways to see that

[17] The standard *name* for OW-Passive is "OW-CPA". However, the only plaintext in the "OW-CPA" definition is chosen randomly, *not* by the attacker. My paper [**30**] with Persichetti renames "OW-CPA" as "OW-Passive".

[18] Given an OW-Passive attack that succeeds with probability p, attack IND-CPA by checking whether the OW-Passive attack finds a random m_0. This IND-CPA attack correctly outputs 0 with probability $p/2$. This IND-CPA attack correctly outputs 1 with probability $(1 - 1/\#\mathcal{M})/2$ where $\#\mathcal{M}$ is the number of possible plaintexts: whichever plaintext the OW-Passive attack finds has conditional probability $1/\#\mathcal{M}$ of matching m_0 given that the ciphertext was obtained by encrypting m_1. Overall the IND-CPA attack has advantage $|(p - 1/\#\mathcal{M})/2|$, so if this advantage is small then p must be close to $1/\#\mathcal{M}$, and thus close to 0 if $\#\mathcal{M}$ is large.

the gap between OW-Passive and IND-CPA *is* important, as explained in the rest of this section.

6.2. Are distinguishers well studied? The claim that lattice problems have been "well studied" consists primarily of pointers to the long literature for algorithms to attack various *search* problems such as SVP and CVP. For example, Ajtai [**3**] credits Dirichlet with formulating "the question of finding a short vector in a lattice"; points to the LLL and Schnorr algorithms as finding a short (nonzero) vector; and highlights three related problems, each of which begins with the word "Find". Regev [**93**] highlights "two of the main computational problems on lattices" and cites the "best known polynomial time algorithms": again the LLL and Schnorr algorithms, and a newer algorithm using sieving. Peikert [**85**] claims that certain lattice problems—including a decision problem—have been "intensively studied", but his list of references for this claim actually consists of one search algorithm after another.

This pattern matches a broader pattern of mathematical algorithm designers emphasizing search problems. See, e.g., [**62**]:

> . . . a large proportion of all of the mathematical research in public-key cryptography is concerned with algorithms for inverting the most important one-way functions. Hundreds of papers in mathematics as well as cryptography journals have been devoted to index calculus methods for factoring integers and for finding the discrete logarithm in the multiplicative group of a finite field, to improved Pollard-ρ algorithms and Weil descent methods for finding discrete logarithms on elliptic curves, and to searches for "weak parameters," i.e., RSA moduli n that are a little easier to factor than most, finite fields over which the elliptic curve discrete logarithm problem is slightly easier to solve, and so on.

The output of a successful search is generally much more informative and much more applicable than a mere distinguisher. For example, the application that motivated the LLL paper [**67**]—"Factoring polynomials with rational coefficients", the title of the paper—uses short vectors found by the LLL algorithm.

Of course, applications sometimes lead algorithm designers to consider distinguishing problems, as the following examples illustrate:

- The problem of distinguishing prime numbers from composite numbers is a subroutine in many other number-theoretic computations. At the time that Gauss highlighted this problem, there were already solutions that were fairly fast and fairly reliable (such as Fermat's test: does n divide $2^n - 2$?). This prompted real-world usage of those solutions. The real-world usage prompted searches for improvements in speed and in reliability.
- Pairings were introduced into cryptology as an attack tool against a search problem. They were then observed to break a distinguishing problem, namely DDH, much more quickly than breaking a related search problem. Some subsequent applications, such as pairing-based signatures, can be viewed as applications of fast distinguishers. The applications of pairings prompted searches for faster pairings.

For comparison, “security proofs” in lattice-based cryptography have assumed the hardness of particular decisional problems, but the occasional study of algorithms for these problems does not seem to have led to broader applications.

6.3. The cost of review. The risk of a devastating attack against a problem is most obvious when there has been little study of the problem, but the risk exists in any case. The usual strategies to control this risk are (1) to review existing cryptanalysis and (2) to carry out and review further cryptanalysis. The complexity of these processes—and, correspondingly, the likelihood that a devastating attack will be publicly discovered (say) 5 years later—varies from one problem to another. As mentioned earlier, I recommend measuring the cost of a thorough security review.

From the perspective of a cryptanalyst, IND-CPA offers every attack avenue that OW-Passive offers, plus extra information and extra flexibility: the plaintext is guaranteed to be one of two messages known to the attacker—even chosen by the attacker on the basis of the public key. Minor deviations of ciphertexts from random can easily break IND-CPA without breaking OW-Passive. A reviewer needs to check not just whether there has been adequate study of algorithms to invert the (randomized) map from plaintext to ciphertext, but also whether there has been adequate study of ways to exploit knowing just two possibilities for the plaintext. As a result, IND-CPA assumptions tend to be worse in this metric than OW-Passive assumptions.

6.4. Interactions with divergence proofs. Imagine that there has been such thorough study of an OW-Passive problem that we are confident in security—but the distribution of messages and keys in the actual cryptosystem is different from what has been studied:

- The OW-Passive problem for the cryptosystem is finding a message that was chosen randomly from distribution M, given a public key chosen randomly from distribution K and an encryption of the message under that key.
- The studied OW-Passive problem is finding a message that was chosen randomly from distribution M', given a public key chosen randomly from distribution K' and an encryption of the message under that key.

There could be an arbitrarily large gap between the security levels of these problems.

Sometimes this gap is addressed by a proof that each message-key pair is at most twice as likely to be produced by (M, K) as by (M', K'). Whichever message-key pairs are broken by an attack are then at most twice as likely to occur for (M, K) as they are for (M', K'). In total, the attack has at most twice the success probability for (M, K) as it does for (M', K'). Our confidence in the security of (M', K') then implies the same confidence in the security of (M, K), except for changing the security level by 1 bit.

The previous paragraph is a simple[19] example of a tight "divergence proof". Note that if OW-Passive were replaced with IND-CPA then this proof would fail: for example, an IND-CPA attack with *zero* advantage against (M', K') produces the correct answer with probability $1/2$ for (M', K'), and multiplying this by 2 says that the attack produces the correct answer with probability at most 1 for (M, K), which is content-free. Bai, Langlois, Lepoint, Sakzad, Stehlé, and Steinfeld [**16**, Section 4] state more advanced divergence theorems for some distinguishing problems, but these theorems are not tight.

If (M', K') is what has been studied, why does the cryptosystem use (M, K) instead? The usual answer is efficiency. For example, Frodo [**6**, Section 5.1.3] claims a divergence proof saying that switching from a discrete Gaussian distribution to another distribution loses at most 1.7 bits of security for `frodo640`; and then focuses on the security of a discrete Gaussian distribution. The stated rationale for the switch of distributions is that (1) a discrete Gaussian distribution is "key" to part of the security analysis, but (2) sampling from this distribution is "difficult" on a "finite computer" and "impossible" in "constant time".

There are two main issues for a reviewer checking a claimed divergence proof. First, the underlying probability calculations are part of the proof and need to be checked carefully. For example, the claimed divergence proof for round-1 Frodo has an apparently unfixable error in the probability calculations, as pointed out by Phong [**91**]. This prompted a change in parameters from round-1 Frodo to round-2 Frodo, according to [**6**, page 49, fourth bullet item]. Despite this history, the probability calculations for round-2 Frodo are still not stated as theorems with detailed proofs.

Second, the applicability of divergence arguments to OW-Passive is straightforward and easy to check, but many of the target KEMs do not have tight proofs of ROM IND-CCA2 security from OW-Passive. As mentioned earlier, Frodo's first security theorem [**6**, Theorem 5.1] claims such a proof, but the claimed proof is not given, and I see no reason to believe that the claimed proof exists.

Even if [**6**, Theorem 5.1] is withdrawn, maybe there is another way to apply divergence arguments to Frodo, for example via intermediate security notions such as OW-PCVA. But this needs a proper theorem statement and proof, so that the tightness and cost of verification can be assessed.[20]

6.5. Updates regarding Frodo. I published my objections to [**6**, Theorem 5.1] on 24 May 2019. I posted the first version of this paper on 8 June 2019. My only subsequent modifications to the text in this section were (1) adding the Goldreich quote, (2) updating reference numbers, and (3) adding this subsection.

Daniel Apon—not speaking for NIST, as far as I know—wrote in [**9**] that the Frodo specification "would be more clear" if "OW-CPA" were changed to "IND-CPA". I disputed this, writing in [**21**] that the change "makes a different (wimpier) statement with exactly the same level of clarity", and pointing out

[19] To be more precise, the portion of the proof that I have displayed here is simple. The underlying analysis of the probability of achieving any particular message-key pair could be much more complicated.

[20] Of course there is not just a ROM question here but also a QROM question.

in [22] that Frodo had changed its theorem from assuming IND-CPA in round 1 (see [5, Theorem 5.1]) to assuming OW-CPA in round 2 (see [6, Theorem 5.1]).

Apon also wrote that "Interpretations that lead to the claimed outcome are probably the best interpretations to use." Of course, if the "claimed outcome" is merely that there is *some* sort of security proof, then a correct proof from IND-CPA would achieve this outcome. However, anyone planning a thorough security review needs to know what the cryptanalytic goals are. A serious evaluation of security risks cannot simply ignore a change in the goalposts.

I wrote the following in [23] a month later: "I still don't see how to prove Frodo's claimed Theorem 5.1. However, the claimed theorem still hasn't been withdrawn." Apon wrote in [10] that "The outstanding question was whether the proof could be made tight (or tighter), not whether a proof exists." I asked in [24] "Are you claiming that you see how to prove Theorem 5.1 as stated in the round-2 Frodo submission?"

On 2 July 2019, in [78], the Frodo team withdrew [6, Theorem 5.1], claiming that "the change in hypothesis from IND-CPA to OW-CPA was a typo that was inadvertently introduced in the revisions between the round-1 and round-2 submissions". In fact, comparing Frodo's round-1 submission document [5] to Frodo's round-2 submission document [6] shows (inter alia) the following changes:

- The round-1 theorem [5, Theorem 5.1, fifth line] considered a "classical algorithm $\mathcal{B}$ against the IND-CPA security of PKE". The round-2 submission [6, Theorem 5.1, fifth line] changed this to consider a "classical algorithm $\mathcal{B}$ against the OW-CPA security of PKE".
- The round-1 theorem [5, Theorem 5.1, display] presented a probability formula involving the "ind-cpa" advantage of $\mathcal{B}$. The round-2 submission [6, Theorem 5.1, display] changed this to the "ow-cpa" advantage.
- The round-1 theorem title [5, Theorem 5.1, first line] said "Theorem 5.1 (IND-CPA PKE $\Rightarrow \ldots$)". The round-2 submission [6, Theorem 5.1, first line] changed this to say "Theorem 5.1 (OW-CPA PKE $\Rightarrow \ldots$)".
- The "summary of the reductions supporting the security of FrodoKEM" earlier on this page of the round-1 submission summarized this theorem as assuming "that FrodoPKE is an IND-CPA-secure public-key encryption scheme". The round-2 submission changed "IND-CPA" to "OW-CPA", and added a footnote saying "OW-CPA is for example defined in [63] and is implied by IND-CPA".

I will leave it to the reader to decide whether it is plausible to describe this consistent collection of changes as a "typo".

Regarding divergence, Frodo's updated specification [7] sketches a way to apply divergence arguments to Frodo via an intermediate security notion, OW-PCA. The specification continues to omit many details of the claimed divergence proofs, so it remains difficult to assess the cost of reviewing the proofs.

7 Splitting key problems from ciphertext problems

Write K for the distribution of public keys in the PKE under consideration, and write K' for another distribution—typically a simplified model of K, perhaps the uniform distribution over a specified set of public keys. Can an OW-Passive attack achieve higher success probability against K than against K'?

If so, then the attacker can distinguish K from K'. The proof is trivial: given a public key, choose a random message and see whether the OW-Passive attack works. This can be phrased as a tight "security proof":

- Assume that the OW-Passive problem for K' is hard.
- Assume that K is indistinguishable from K'.
- Then the OW-Passive problem for K (i.e., for the PKE) is hard.

Some of the KEMs need to assume IND-CPA for the PKEs, and thus use the following equally trivial "security proof":

- Assume that the IND-CPA problem for K' is hard.
- Assume that K is indistinguishable from K'.
- Then the IND-CPA problem for K is hard.

I have chosen notation here to be able to easily highlight how these trivial proofs differ from the more sophisticated divergence proofs considered in Section 6.4. In the situation of Section 6.4, K might be distinguishable from K', but a provable divergence limit guarantees a relationship between the OW-Passive problems. In this section, the actual content of the proof disappears: K is simply assumed to be indistinguishable from K'.

7.1. Example 1: Quotient NTRU. As a concrete illustration of this trivial split, consider Streamlined NTRU Prime 4591^{761} (`sntrup761`), an example of a Quotient NTRU KEM. The underlying problem is to break OW-Passive security of a PKE called "Streamlined NTRU Prime Core 4591^{761}".

In this PKE, a public key G is (technically, an encoding of) e/a for a particular distribution of pairs (a, e). Specifically, e is a uniform random "invertible small" element of a particular field, and a, independent of e, is a uniform random "short" element of the field. A ciphertext is obtained by "rounding" $Gb/3$, where the plaintext b is a short element of the field; the ciphertext thus has the form $Gb/3 + d$ where d is a small element of the field.

The trivial proof says that the OW-Passive problem for this PKE is hard under the following two assumptions:

- Key indistinguishability vs. K': A key e/a is indistinguishable from a key chosen from distribution K'.
- OW-Passive for K': It is hard to recover a uniform random plaintext b given a key G chosen from distribution K' and the ciphertext $Gb/3 + d$.

This is parameterized by the choice of K'. The first choice of K' that comes to mind is the uniform distribution over the field. Another choice is the uniform distribution over nonzero elements of the field. These two choices are equivalent in the sense that, since the field is large, they are provably indistinguishable.

7.2. Example 2: Product NTRU. As another concrete illustration, consider NTRU LPRime 4591^{761} (`ntrulpr761`), an example of a Product NTRU KEM. Here the underlying problem is to break IND-CPA security of a PKE called "NTRU LPRime Core 4591^{761}".

In this PKE, a public key has two components (G, A). The first component G is a uniform random element of the field. The second component A is obtained by rounding aG to $aG+e$, where a, independent of G, is a uniform random short element of the field.

A ciphertext has two components (B, T). The first component is obtained by rounding Gb to $Gb + d$, where b is a uniform random short element of the field. The second component is various "top bits" of $Ab + M$, where M is a field element that encodes a 256-bit plaintext.

The trivial proof says that the IND-CPA problem for this PKE is hard under the following two assumptions:

- Key indistinguishability vs. K': A key $(G, aG + e)$ is indistinguishable from a key chosen from distribution K'.
- IND-CPA for K': It is hard to distinguish ciphertext $(Gb + d, \mathsf{Top}(Ab + M))$ from ciphertext $(Gb' + d', \mathsf{Top}(Ab' + M'))$ for chosen plaintexts M, M', given a key (G, A) chosen from distribution K'.

Given that $aG + e$ is obtained by rounding, the obvious choice of K' is pairs (G, A) where G is a uniform random field element and A is obtained by rounding a uniform random field element independent of G.

7.3. Obstructions to proving relationships between problems. Beyond the difference between Quotient NTRU and Product NTRU, the target KEMs vary in rounding (Rounded NTRU) vs. noise (Noisy NTRU), the choice of ring, etc. See Section 8.

Sometimes there is a proof that a change in problem does not lose security: consider, e.g., the OW-Passive vs. IND-CPA example from Section 6. However, the changes covered by such proofs do not reach all the way from one of the target KEMs to another, or even from one of the target PKEs to another.

There are some "LWR"-vs.-"LWE" theorems saying, in spirit, that rounding cannot be much easier to break than noise. The basic idea is simple: given As plus noise, anyone can round to obtain a rounded As, so anyone can convert an attack against rounded As to an attack against As plus noise. But these theorems quantitatively degrade as the modulus decreases. Rounding As plus noise becomes more and more likely to produce a result different from rounding As, and guessing the differences becomes increasingly expensive; see generally [8]. As far as I know, none of the theorems apply to moduli as small as the moduli used in the target KEMs.

If a Quotient NTRU PKE is split into a key problem and a ciphertext problem as in Section 7.1, and a Product NTRU PKE is split into a key problem and a ciphertext problem as in Section 7.2, then two of these four problems are related. Specifically, the Quotient NTRU ciphertext problem is to find b given G and $Gb + d$, and the Product NTRU key problem is to distinguish $aG + e$ from random given G, i.e., to distinguish $Gb + d$ from random given G. If the underlying distributions match then the search problem cannot be easier than the distinguishing problem.

On the other hand, the Quotient NTRU *key* problem is to distinguish e/a from random. The corresponding search problem is to find a given G and given $aG - e = 0$. There is no proof that this search problem is as hard as the Quotient NTRU ciphertext problem: homogeneity changes the input distribution and could make the problem easier. As far as I know, there is also no proof the other way: homogeneity could also make the problem harder. For the same reasons, there is no proof relating the Quotient NTRU key problem to the Product NTRU key problem.

There is, furthermore, no proof that the Product NTRU ciphertext problem is as hard as the Product NTRU key problem. A proof would have to somehow handle the fact that the ciphertext problem releases approximations to *two* multiples Gb, Ab of a secret b with public multipliers G, A, while the key problem releases an approximation to just one multiple.

From the perspective of *known* attacks, homogeneity produces an attack speedup for Quotient NTRU, but the extra complications of Product NTRU require lower noise, producing an attack speedup for Product NTRU. These two effects are roughly balanced: compare "`sntrup`" to "`ntrulpr`" in [27, Table 2]. But my focus in this paper is on what has been *proven*; from this perspective, either type of problem could be much weaker.

7.4. The cost of review, part 1: cryptanalysis time. How can a trivial proof that splits a key problem from a ciphertext problem possibly save time for cryptanalysts? The cryptanalyst has no trouble computing a simplified model of public keys. The cryptanalyst already knows that taking advantage of the actual key structure requires finding a way to detect deviations from the model. The cryptanalyst spends time searching for deviations from the model, and spends time searching for attacks in the model.

Of course, if a proof assumes that K is indistinguishable from K', and a cryptanalyst finds a fast distinguisher, then one might think that the cryptanalyst can end the analysis, since one of the proof assumptions has been shown to be false. But the proof authors then respond as follows:

- Here's a new (equally trivial) proof using a refined model K'' that avoids the distinguisher.
- The distinguisher for K' doesn't matter, since it doesn't break the cryptosystem.

The cryptanalyst then searches for deviations from the new model, and searches for attacks in the new model. This is also what would have happened without the proof.

7.5. The cost of review, part 2: proof-review time. One might think that the negligible benefit of this section's proof strategy for cryptanalysts is balanced by the negligible cost of writing down and verifying such a trivial proof. However, the proof is typically obfuscated enough that the necessary review time is beyond what the community has spent.

Consider, for example, NewHope's [4, Theorem 4.4], which deters reviewers in several ways. The theorem statement occupies 6 lines, plus 6 lines via the definition of "DRLWE", 30 lines via the definition of "NewHope-CPA-PKE", and many more lines via subsidiary definitions. No proof is given, beyond a claim that "the proof" is "essentially the same" as two cited proofs. The reader is not told that stripping away the notation produces a trivial proof that has nothing to do with the details of NewHope.[21]

Someone who does the work to read the "DRLWE" definition finds three explicit parameters m, q, χ and an implicit parameter n. The problem considers "m samples" from "the uniform distribution" on $R_q \times R_q$, where R_q is "the ring $Z[X]/(X^n + 1)$" reduced modulo q; the problem also specifies another distribution of m samples, and asks whether these two distributions can be distinguished.

The theorem statement involves two "DRLWE" advantages, each with m being specified as n. In short, the proof of NewHope-CPA-PKE security assumes the hardness of detecting patterns in n samples, where each sample contains n integers modulo q. Assumptions in "security proofs" are supposed to draw the attention of cryptanalysts; cryptanalysts paying attention to this particular theorem would note that n^2 integers are enough to enable the Arora–Ge attack strategy [12] for some error distributions, and would ask whether the strategy can be adapted to the error distribution used in NewHope.

However, the submission then continues by analyzing a different "DRLWE" assumption, in which the attacker is given only $2n$ integers rather than n^2 integers. There is no comment on the discrepancy between

- the "DRLWE" problem that is analyzed and
- the potentially much more easily broken "DRLWE" problem that the theorem assumes to be difficult.

I am not saying that the theorem is incorrect as stated: I am saying that the NewHope authors did not realize, and did not analyze, the strength of the hardness assumption made by the theorem that they stated.

My best guess is that the authors intended to state a different theorem. At various other points the NewHope submission uses "samples" to refer to the number of integers provided to the attacker; this is not consistent with how "samples" is used in the NewHope "DRLWE" definition. This inconsistency would disappear if the "DRLWE" definition were modified to say that m is required to be a multiple of n, and that the m samples consist of m/n pairs from $R_q \times R_q$ rather than m pairs. But this modified definition would break the correctness of the theorem. Product NTRU systems such as NewHope have

- a key that releases an approximation to a multiple of a—this means n integers—and

[21] This also tends to make readers think that there is a limit on the number of KEMs that enjoy "analogous" proofs.

- a ciphertext that releases approximations to two multiples of b—this means $2n$ integers, although compression could produce, e.g., $n + 256$ integers.

The numbers n and $2n$ in the key and ciphertext problems are bounded by n^2 and n^2 respectively (for the n in question), but they are not even close to being bounded by n and n, so the theorem would also have to be modified.

Perhaps these changes to the definition and theorem would produce a theorem that is simultaneously (1) correct and (2) making the same hardness assumption as the assumption that has (supposedly) been analyzed. But I don't vouch for this: this would require checking many details that I haven't checked. Furthermore, NewHope accounts for just 2 of the 36 target KEMs covered in this paper. My point here is that one cannot dismiss the proof-review time.

A month after I posted these observations, the NewHope team acknowledged the "inconsistency", and modified its theorem statement to replace n and n with 1 and 2. This is now consistent with the "DRLWE" definition of "samples", but is inconsistent with how the word "samples" is used in the attack analysis.

8 The core problems

Beyond the general risks of errors in the proofs, attacks not covered by the proofs, and tightness failures, there is an unavoidable risk of attacks against the underlying problem P. This risk is indicated by the "P" column in Table 1.1.

The purpose of this section is to describe these problems P for all of the target KEMs, with a unified notation that allows cryptanalysts to easily see and compare the details. This description is simpler than a description of the KEMs, for several reasons:

- CCA conversions are eliminated, leaving simpler PKEs that aim for IND-CPA or OW-Passive security. (See Section 5.)
- The PKE decryption algorithms are skipped. Decryption does not appear in the IND-CPA and OW-Passive security definitions.
- Various optimizations are stripped away, as justified by divergence arguments (see Section 6.4) or simple algebra.
- Product NTRU PKEs that expand short seeds into pseudorandom multipliers are replaced by simpler PKEs with random multipliers. (See Section 4.1.)
- PKEs that release rounded noisy ciphertexts are replaced by simpler PKEs that release noisy ciphertexts without rounding. (This could allow attacks that are not allowed by the original system; I have included this simplification only when it is advertised by the KEM authors as part of the underlying problem.)

Most of these simplifications are already visible inside KEM submissions, but variations in notation make it unnecessarily difficult to compare the results across all the target KEMs.

The Product NTRU PKEs all have "dist: risk" in Table 1.1; in each of these cases, P is the IND-CPA problem for the PKE described below. The Quotient

NTRU PKEs all have “dist: safe” in Table 1.1; in each of these cases, P is the OW-Passive problem for the PKE described below.

8.1. Key generation. A public key reveals a **multiplier** G and an approximation A to aG. Here a is something **short** chosen randomly during the key-generation process. The PKEs vary in the set of multipliers, the distribution of short elements, and how the approximation is obtained. At a high level, the key-generation procedures fall into three categories:

- The Quotient NTRU PKEs generate a random short a, generate a random **numerator** e, compute $G = e/a$, and output G as the public key. The homogeneous equation $0 = aG - e$ says that $A = 0$ is close to aG.
- The Noisy Product NTRU PKEs generate a random short a, generate a random multiplier G, generate random **noise** e, compute $A = aG + e$, and output (G, A) as the public key. The inhomogeneous equation $A = aG + e$ says that A is close to aG.
- The Rounded Product NTRU PKEs generate a random short a, generate a random multiplier G, compute A by deterministically **rounding** aG, and output (G, A) as the public key. There is again an inhomogeneous equation $A = aG + e$ saying that A is close to aG.

See Section 8.5 for per-target key-generation details.

8.2. Encryption. A ciphertext reveals an approximation B to Gb, where G is the multiplier included in the public key, and b is a short input to the encryption process (either plaintext or randomness). For the Product NTRU PKEs, a ciphertext also reveals an approximation C to $Ab + M$, where A is the approximation included in the public key, and M is an **encoded message**. At a high level, the encryption procedures fall into four categories:

- The Noisy Quotient NTRU PKEs take a short b and noise d as plaintext, compute $B = 3Gb + d$, and output B as the ciphertext.
- The Rounded Quotient NTRU PKEs take a short b as plaintext, compute B by rounding $Gb/3$, and output B as the ciphertext.
- The Noisy Product NTRU PKEs take an encoded message M as plaintext, generate a random short b, generate random noise d, generate random noise c, compute $B = Gb + d$, compute $C = Ab + M + c$, and output (B, C) as the ciphertext.
- The Rounded Product NTRU PKEs take an encoded message M as plaintext, generate a random short b, compute B by rounding Gb, compute C by rounding $Ab + M$, and output (B, C) as the ciphertext.

See Section 8.5 for per-target encryption details.

8.3. The cost of review. There are at least four reasons that a thorough security review of these PKEs needs a huge amount of time.

First, the state-of-the-art attacks against all of these PKEs use a complicated, hard-to-analyze attack strategy that involves many stages, many different choices of subroutines, many tunable knobs, and many recent papers. See [27, Section 6] for a survey. Reviewers are forced to evaluate risks of advances at many specific points within this attack strategy.

Second, even though the differences in data flow described above have little effect on this attack strategy, security reviewers are forced to consider whether there could be a larger effect. For example, requiring IND-CPA rather than merely OW-Passive raises extra questions for security reviewers; see Section 6. There could be weaknesses specifically for Noisy Quotient NTRU, or for Rounded Quotient NTRU, or for Noisy Product NTRU, or for Rounded Product NTRU. There are various obstructions to proving relationships between the attack problems; see generally Section 7.3.

Third, the PKEs differ in many further details (see Section 8.5), which can interact in many ways with the state-of-the-art attacks. For example, after [**71**], the security estimate for `lac192` was downgraded from 2^{286} to 2^{278} (see [**103**]), which might not sound like a dramatic change but reflects a pervasive failure to analyze and optimize "hybrid attacks". Each PKE needs to be analyzed; there are no theorems stating that one of the PKEs is at least as hard to break as another.

Fourth, other complicated attack strategies have appeared in state-of-the-art attacks against other lattice-based cryptosystems. For example, some lattice problems have been broken for ideal lattices arising from number fields with small Galois groups, as illustrated by the polynomial-time quantum break [**32**] of Gentry's original STOC 2009 FHE system [**46**] for cyclotomic fields, the quasipolynomial-time non-quantum break [**18**] of analogous FHE systems for a wide range of multiquadratic fields, and recently a similar break [**68**] for multicubic fields. Various claims of lines separating these attacks from the target PKEs have been disproven by more recent advances in attacks; reviewers are forced to consider whether further advances could damage the target PKEs. NTRU Prime tries to simplify this review by avoiding small Galois groups, and Frodo tries to simplify this review by avoiding number fields entirely, but most of the target PKEs use cyclotomic fields.

8.4. Misevaluations of the cost of review. There is a common belief that easy-to-state hardness assumptions are easy-to-review hardness assumptions. For example, Katz and Lindell [**60**, page 22] claim that "assumptions that are simpler to state ... are easier to study and to (potentially) refute".[22]

As pre-quantum examples, the hardness of factoring and the hardness of multiplicative-group discrete logarithms are simpler to state than the hardness of, e.g., Curve25519 discrete logarithms.[23] The belief then says that the first two hardness assumptions are easier to study than the third. But extensive cryptanalysis indicates the opposite. The factoring problem and the multiplicative-group discrete-logarithm problem provide many more tools to the attacker than the

[22] After I posted the first version of this paper, Lindell tweeted [**70**] that the "claims outlined in Katz–Lindell on this turn out to be overly simplified and naive".

[23] I am assuming here that all details are spelled out. Typically an ECDL definition is split into (1) a definition of an elliptic-curve group and (2) a statement of a DL problem for this group; stating the group definition is part of stating the problem.

Curve25519 discrete-logarithm problem does.[24] These tools have been combined into very complicated attacks[25] including several advances this decade (e.g., [17]). Further advances would not be surprising.

The literature on lattice-based cryptography has placed tremendous emphasis on particular hardness assumptions that are claimed to be "simple". This seems to mean that the assumptions are simple to state, but I have not seen a side-by-side comparison of the simplicity of the statements. More to the point, this emphasis seems to be a distraction from the issues that matter for cryptanalysts and other security reviewers.

Specifically, starting from the original PKE attack problems, one arrives at the emphasized problems as follows:

- Split each problem into a ciphertext problem and a key problem. This is trivial for all of the PKEs; see Section 7. I don't see how these proofs help cryptanalysts; see Section 7.4.
- Require Product NTRU rather than Quotient NTRU. I don't see how anyone comparing the problems can conclude that the Product NTRU problems are simpler to state than the Quotient NTRU problems, even if one disregards the difference between IND-CPA and OW-Passive. More importantly, this takes away some cryptanalytic concerns but adds others. Product NTRU could be weaker than Quotient NTRU, or vice versa; see Section 7.3.
- The ciphertext question for the Product NTRU PKEs is whether one can distinguish an approximation to $(Gb, Ab + M)$ from an approximation to $(Gb', Ab' + M')$. Compute an M-independent model of these pairs—e.g., the distribution of approximations to (R, S) for independent uniform random R and S—and ask whether one can distinguish an approximation to $(Gb, Ab + M)$ from this model.
- Require the PKEs to have the algebraic feature that approximation commutes with addition of M: the approximation to $Ab + M$ is M plus the approximation to Ab, so if the model is invariant under addition of M then the model question is equivalent to the special case $M = 0$. For example, Noisy NTRU can add noise independent of the input, and Rounded NTRU can round to an ideal containing M.

The special case $M = 0$ eliminates the attacker's ability to choose M, but from this perspective the Product NTRU requirement makes no sense: Quotient NTRU allows OW-Passive problems, eliminating chosen plaintexts *and* known plaintexts. Meanwhile the way that the algebraic feature is achieved raises questions for cryptanalysts: e.g., whether Noisy NTRU allows attacks that would not apply to Rounded NTRU, and whether the presence of nontrivial ideals could allow an extension of the attack ideas of [98].

[24] The most important tools are efficient ring morphisms. See, e.g., the explanation in [36, pages 53–54] of the ring morphisms that enable NFS. Pairings extend NFS to groups $E(\mathbf{F}_q)$ of orders dividing $q - 1$, $q + 1$, etc. See [76], [57], and [97] for the mathematical obstacles to handling curves with other group orders.

[25] Including, e.g., factoring subroutines that rely on elliptic-curve computations, so the cryptanalyst has to learn elliptic curves in any case.

8.5. Per-target problem details. Table 8.6 shows the set of multipliers G for each PKE. Table 8.7 shows how short elements a are generated. Table 8.8 shows how the difference $A - aG$ is generated; this concludes key generation. Encryption generates b and $B \approx Gb$ as in Tables 8.7 and 8.8 respectively, except that (1) `sntrup` rounds $Gb/3$ the same way that `ntrulpr` rounds Gb, and (2) `ntru` adds noise to $3Gb$. For Product NTRU, Table 8.9 shows how $Ab + M$ is converted to C, and Table 8.10 shows the set of encoded messages M.

9 Lattice problems

> *We reiterate the crucial point: if the reduction proving security is "loose," like the one above, the efficiency of the scheme is impacted, because we must move to a larger security parameter.*
>
> —Bellare and Rogaway [19]

> *When using schemes in practice one needs to know the exact complexity of the reduction (for that will determine the actual security of the concrete scheme).* —Goldreich [48, page 27]

Lyubashevsky, Peikert, and Regev [72] described their work as "proving" that Ring-LWE "enjoys very strong hardness guarantees". What they actually proved was a theorem saying the following: for some Ring-LWE parameters, an attack implies, *up to a polynomial loss of tightness*, an attack against a lattice problem, specifically an approximate Ideal-SVP problem.

Advances in attacks against approximate-Ideal-SVP in the six years since then have led researchers to question the hardness of approximate-Ideal-SVP. For example, Pellet-Mary, Hanrot, and Stehlé [88] write that their result "strongly suggests that approx-SVP for ideals ... may be weaker than Ring-LWE, for a vast family of number fields". But some of the target KEMs continue to advertise approximate-Ideal-SVP proofs.[26] Also, some of the target KEMs advertise lattice proofs in the non-ideal context, where there have been fewer advances in lattice attacks.

The rest of this section analyzes the consequences of a more serious problem with all of these lattice proofs.

9.1. Looseness to the point of disconnection. The original literature on these proofs did not quantify the polynomial loss of tightness. In other words, cryptanalysts were not told the target security level for the underlying lattice problem. This makes it more difficult to publish attacks—of course a dramatic speedup from exponential time to polynomial time is publishable in any case, but most cryptanalytic effort is spent on smaller speedups. Deterring cryptanalytic effort is contrary to the goal of having the hardness assumption be "well studied".

Even worse, the loss of tightness turns out to be gigantic, so these proofs are only for irrelevant cryptosystem parameters. As far as I can tell, the primary

[26] Peikert [87] claims that "worst-case hardness theorems for Ring-LWE appear to be of no consequence to the remaining NIST submissions". The intended meaning of this claim is not clear to me.

credit for this observation belongs to a 2016 paper [38] by Chatterjee, Koblitz, Menezes, and Sarkar.[27]

Concretely, [38] analyzed Regev's worst-case-to-average-case reduction for a cryptosystem that Regev had proposed, took lattice dimension 1024 with security target 2^{128} as a case study, and found an astonishing 2^{504} tightness gap in the proof. There appears to be consensus that known attacks break the underlying lattice problems with far fewer than $2^{504+128}$ operations. Recently Sarkar and Singha showed [96] that another step in Regev's proof fails for all lattice dimensions below 187150.

The obvious research challenge here is to present a complete proof that says something non-vacuous about dimension D for the minimum possible D. Reaching $D = 10000$ would be an impressive advance over the current state of the art. Note that the largest dimension in the target KEMs is just 1344.

For comparison, "fail" in Table 1.1 shows cases where the claimed failure probability δ is too large for the proofs to say anything *at the claimed security level for the KEM*. The proofs still—modulo the other risks that I have listed—rule out attacks that cost, say, 2^{100} against these KEMs. Many users of the target KEMs will find this adequate, even though it is below the claimed security level.

In this section, the situation is different. For each target KEM X, there is no justification for the claim that these proofs guarantee *any* security for X, even under the assumption that there are no further advances in lattice attacks.

9.2. Attempting to use "families" as a substitute for tightness. There is still an argument that a proof of this type reduces risks. This argument has the astonishing feature of being blind to the quantitative security level of the underlying problems: cryptanalytic advances against the underlying problems are irrelevant to the argument *unless* the advances produce a polynomial-time attack. The general structure of this argument is as follows:

- Consider an attack against a KEM: e.g., an attack against `frodo1344`.
- This KEM is one member of a large "family" of KEMs: e.g., `frodo1344` is one member of a "family" of Frodo KEMs.
- Assume that the attack applies to the entire "family": e.g., assume that the `frodo1344` attack applies to the entire Frodo "family". Presumably it is possible to formally define a Frodo "family" attack.
- This implies an attack against another KEM: e.g., the attack applies to some huge member of the Frodo "family", sufficiently large for the next step to get beyond the looseness of the lattice proof.
- Under further assumptions—e.g., hardness assumptions for the underlying lattice problems—the proof rules out this attack.

[27] Peikert [87] claims, incorrectly, that this observation was already published in [74] in 2009. What [74] says is merely that setting parameters based on the worst-case proofs is "overly conservative". Peikert characterizes this as saying that the parameters are not "practical"; obviously it is possible for parameters to be practical and yet overly conservative, so Peikert's characterization is not a correct summary of the statement from [74]. Chatterjee, Koblitz, Menezes, and Sarkar did the work to quantify a typical proof; I have not found any evidence that this was done previously.

Obviously this argument leaves risks of non-"family" attacks, the same way that a ROM proof leaves risks of non-ROM attacks; but isn't this better than not having a proof, the same way that having a ROM proof is better than not having a ROM proof? Shouldn't I have

- a column in Table 1.1 indicating the risks of non-"family" attacks—this would be marked everywhere, just like the "ROM" column indicating the risks of non-ROM attacks—and
- another column in Table 1.1 indicating the additional risk of the submissions that don't have proofs regarding "family" attacks, the same way that I would have an extra column if there were submissions without ROM IND-CCA2 proofs?

As a concrete example, some people seem to believe that `kyber1024` "has" a proof of the type explained above, while `firesaber` does not "have" this type of proof. Shouldn't I add a table column indicating the extra risk of `firesaber`?

Let's look more closely at the claim that `firesaber` doesn't "have" this type of proof. To disprove this claim, it isn't necessary to prove anything about `firesaber` per se; it suffices to exhibit a member of the Saber "family" that has a proof. Why can't we do this by

- pointing to a proof that breaking Ring-LWE is "as hard as" a lattice problem (as Kyber does), and
- pointing to a proof that breaking Module-LWE is "as hard as" breaking Ring-LWE (as Kyber does), and
- pointing to a proof that breaking Module-LWR is "as hard as" breaking Module-LWE (this would be the extra step for Saber)?

Maybe the total looseness of such proofs prevents the proofs from applying to *any* member of the Saber "family": in particular, getting from LWR to LWE is loose when moduli are small, and perhaps the Saber "family" doesn't include any large enough moduli. But I don't see anything in the Saber specification that prevents large enough moduli—and, even if I've missed something in the specification, I don't see any obstacle to extending the Saber "family" to include large enough moduli. If `frodo1344` can claim a proof by bundling `frodo1344` together with a huge KEM that's declared to be in the same "family", and `kyber1024` can claim a proof by bundling `kyber1024` together with a huge KEM that's declared to be in the same "family", then why can't `firesaber` claim a proof by bundling `firesaber` together with a huge KEM that's declared to be in the same "family"?

Perhaps the state-of-the-art proofs (see [8]) say that LWR is "as hard as" LWE but don't quite say that Module-LWR is "as hard as" Module-LWE. But this issue is easily eliminated. The starting point is that we're allowed to claim proofs for a KEM by bundling it into a "family" together with a huge KEM for which the proofs apply. So let's define a "family" with three parameters: a modulus, a module dimension, and an overall lattice dimension. The huge KEM

- takes the modulus large enough for LWR to be "as hard as" LWE;

- takes the lattice dimension large enough for LWE to be "as hard as" a lattice problem; and
- takes the module dimension as large as the lattice dimension, so that Module-LWR is the same as LWR.

Why can't we conclude that `firesaber` "has" a lattice proof?

Similarly, since [**100**] proves that some large Quotient NTRU examples are "as secure as worst-case problems over ideal lattices", why can't we claim that smaller Quotient NTRU examples also "have" lattice proofs? Even more straightforwardly, given the lack of definition of what it means to "have" a proof, why can't we simply glue any target KEM X together with GodzillaKEM into a "family", and conclude that X "has" a lattice proof?

Previous sections of this paper followed the rule that "proofs of security" for a cryptosystem X "guarantee—relative to the definition and assumptions—that no attacker will succeed" against X. Obviously the hardness assumptions in this rule vary from one X to another, but the security definitions are independent of X. For example, one can give a complete definition of whether ROM IND-CCA2 attacks against a KEM can cost less than 2^{128} in a clear cost metric, without looking at any details of the KEM. Formally, a ROM KEM is different from a specific-hash KEM, but there is a clearly defined relationship between a ROM KEM and a KEM obtained by plugging in a particular hash; I am not aware of any claims that the variations in ROM KEMs allowed by this relationship would allow additional proofs for any of the target KEMs.

It is completely unclear to me how the proofs considered in this section are believed to follow this rule. Readers are asked

- to imagine that an attack against X also applies to some huge cryptosystem H, and
- to then consider proofs regarding H.

Meanwhile readers are, for completely unclear reasons, *not* allowed to imagine that an attack against another KEM X' also applies to H. Why is the security definition, the type of attack under consideration, varying between X and X'?

If this notion of "having" a proof is supposed to be a property of the cryptosystem X, then there needs to be a clear definition of the allowable relationship between X and H, just like the clear definition of the relationship between a ROM KEM and a KEM obtained by plugging in a specific hash. Here is another quote from Katz and Lindell (italics in original):

> One of the key contributions of modern cryptography has been the recognition that formal definitions of security are *essential* for the proper design, study, evaluation, and usage of cryptographic primitives. Put bluntly:
>
> *If you don't understand what you want to achieve, how can you possibly know when (or if) you have achieved it?*
>
> Formal definitions provide such understanding by giving a clear description of what threats are in scope and what security guarantees are desired.

A definition is the starting point for analyzing obvious questions such as

- whether proofs under this definition are available for some of the target KEMs and not for others;
- whether the availability of proofs under this definition saves any time for cryptanalysts; and
- whether there are reasons to believe that the availability of proofs under the definition is better than random guessing as a predictor of security.

As far as I know, none of this analysis—not even the starting definition—has been published by any of the people claiming that lattice proofs are an advantage of some of the target KEMs over others. Normally people issuing "provable security" claims are required to start with clear security definitions.

In the absence of a competing definition, I conclude that "families" can be defined in any way one wants. Each target KEM "has" a lattice proof under these nonexistent criteria, so these lattice proofs cannot say anything about the security of those KEMs. Consequently, the lattice proofs should be ignored by cryptanalysts, other security reviewers, standardization agencies, and users.

References

[1] Luca Aceto, Monika Henzinger, Jirí Sgall (editors), *Automata, languages and programming—38th international colloquium, ICALP 2011, Zurich, Switzerland, July 4–8, 2011, proceedings, part I*, Lecture Notes in Computer Science, 6755, Springer, 2011. ISBN 978-3-642-22005-0. See [12].

[2] Miklós Ajtai, *Generating hard instances of lattice problems (extended abstract)*, in STOC 1996 [**75**] (1996), 99–108; see also newer version [**3**].

[3] Miklós Ajtai, *Generating hard instances of lattice problems* (1996); see also older version [**2**]. URL: https://eccc.weizmann.ac.il/report/1996/007/. Citations in this document: §6.2.

[4] Erdem Alkim, Roberto Avanzi, Joppe Bos, Leo Ducas, Antonio de la Piedra, Thomas Poppelmann, Peter Schwabe, Douglas Stebila, Martin R. Albrecht, Emmanuela Orsini, Valery Osheter, Kenneth G. Paterson, Guy Peer, Nigel P. Smart, *NewHope: algorithm specifications and supporting documentation* (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §1.3, §7.5.

[5] Erdem Alkim, Joppe Bos, Leo Ducas, Patrick Longa, Ilya Mironov, Michael Naehrig, Valeria Nikolaenko, Christopher Peikert, Ananth Raghunathan, Douglas Stebila, *FrodoKEM: Learning With Errors key encapsulation* (2017). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions. Citations in this document: §6.5, §6.5, §6.5, §6.5, §6.5.

[6] Erdem Alkim, Joppe Bos, Leo Ducas, Patrick Longa, Ilya Mironov, Michael Naehrig, Valeria Nikolaenko, Christopher Peikert, Ananth Raghunathan, Douglas Stebila, *FrodoKEM: Learning With Errors key encapsulation*, "March 30" version (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §1.3, §1.4, §1.4, §4.1, §6.4, §6.4, §6.4, §6.4, §6.5, §6.5, §6.5, §6.5, §6.5, §6.5, §6.5.

[7] Erdem Alkim, Joppe Bos, Leo Ducas, Patrick Longa, Ilya Mironov, Michael Naehrig, Valeria Nikolaenko, Christopher Peikert, Ananth Raghunathan, Douglas Stebila, *FrodoKEM: Learning With Errors key encapsulation*, "July 2" version (2019). URL: https://frodokem.org/files/FrodoKEM-specification-20190702.pdf. Citations in this document: §4.1, §6.5.

[8] Jacob Alperin-Sheriff, Daniel Apon, *Dimension-preserving reductions from LWE to LWR* (2016). URL: https://eprint.iacr.org/2016/589. Citations in this document: §7.3, §9.2.

[9] Daniel Apon, *[pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: Frodo* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/pqc-forum/_kBMTq3RM28/drrdrh3nAgAJ. Citations in this document: §6.5.

[10] Daniel Apon, *Re: [pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: Frodo* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/pqc-forum/_kBMTq3RM28/9gQ1Ytz7BwAJ. Citations in this document: §6.5.

[11] 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 [42].

[12] Sanjeev Arora, Rong Ge, *New algorithms for learning in presence of errors*, in ICALP 2011 [1] (2011), 403–415. URL: https://users.cs.duke.edu/~rongge/LPSN.pdf. Citations in this document: §7.5.

[13] Benedikt Auerbach, David Cash, Manuel Fersch, Eike Kiltz, *Memory-tight reductions*, in Crypto 2017 [61] (2017), 101–132. URL: https://eprint.iacr.org/2017/675. Citations in this document: §5.3.

[14] Roberto Avanzi, Joppe Bos, Leo Ducas, Eike Kiltz, Tancrede Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, Gregor Seiler, Damien Stehlé, *CRYSTALS-Kyber: algorithm specifications and supporting documentation* (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §1.3.

[15] Hayo Baan, Sauvik Bhattacharya, Scott Fluhrer, Oscar Garcia-Morchon, Thijs Laarhoven, Rachel Player, Ronald Rietman, Markku-Juhani O. Saarinen, Ludo Tolhuizen, Jose-Luis Torre-Arce, Zhenfei Zhang, *Round5: KEM and PKE based on (Ring) Learning With Rounding* (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §1.3.

[16] Shi Bai, Adeline Langlois, Tancrède Lepoint, Damien Stehlé, Ron Steinfeld, *Improved security proofs in lattice-based cryptography: Using the Rényi divergence rather than the statistical distance*, in Asiacrypt 2015 [56] (2015), 3–24. URL: https://eprint.iacr.org/2015/483. Citations in this document: §6.4.

[17] Razvan Barbulescu, Pierrick Gaudry, Antoine Joux, Emmanuel Thomé, *A heuristic quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic*, in Eurocrypt 2014 [82] (2014), 1–16. URL: https://eprint.iacr.org/2013/400. Citations in this document: §8.4.

[18] Jens Bauch, Daniel J. Bernstein, Henry de Valence, Tanja Lange, Christine van Vredendaal, *Short generators without quantum computers: the case of multiquadratics*, in Eurocrypt 2017 [40] (2017), 27–59. URL: https://multiquad.cr.yp.to. Citations in this document: §8.3.

[19] Mihir Bellare, Phillip Rogaway, *The exact security of digital signatures—how to sign with RSA and Rabin*, in Eurocrypt 1996 [73] (1996), 399–416. URL:

https://web.cs.ucdavis.edu/~rogaway/papers/exact.html. Citations in this document: §9.

[20] Daniel J. Bernstein, *Extending the Salsa20 nonce*, Workshop Record of Symmetric Key Encryption Workshop 2011 (2011). URL: https://cr.yp.to/papers.html#xsalsa. Citations in this document: §5.3.

[21] Daniel J. Bernstein, *Re: [pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: Frodo* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/pqc-forum/_kBMTq3RM28/4VeFZOxpBQAJ. Citations in this document: §6.5.

[22] Daniel J. Bernstein, *Re: [pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: Frodo* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/pqc-forum/_kBMTq3RM28/bxbQub2LBQAJ. Citations in this document: §6.5.

[23] Daniel J. Bernstein, *Re: [pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: Frodo* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/pqc-forum/_kBMTq3RM28/HCXGJzIgBAAJ. Citations in this document: §6.5.

[24] Daniel J. Bernstein, *Re: [pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: Frodo* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/pqc-forum/_kBMTq3RM28/afIJBlpoBAAJ. Citations in this document: §6.5.

[25] Daniel J. Bernstein, Johannes Buchmann, Erik Dahmen (editors), *Post-quantum cryptography*, Springer, 2009. ISBN 978-3-540-88701-0. See [74].

[26] Daniel J. Bernstein, Tung Chou, Tanja Lange, Ingo von Maurich, Rafael Misoczki, Ruben Niederhagen, Edoardo Persichetti, Christiane Peters, Peter Schwabe, Nicolas Sendrier, Jakub Szefer, Wen Wang, *Classic McEliece: conservative code-based cryptography*, "Supporting Documentation" (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §2.

[27] Daniel J. Bernstein, Chitchanok Chuengsatiansup, Tanja Lange, Christine van Vredendaal, *NTRU Prime: round 2* (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §1.3, §7.3, §8.3.

[28] Daniel J. Bernstein, Tanja Lange, Ruben Niederhagen, *Vulnerability of the RNG ecosystem* (2014). URL: https://projectbullrun.org/dual-ec/vulnerability.html. Citations in this document: §B.

[29] Daniel J. Bernstein, Tanja Lange, Ruben Niederhagen, *Dual EC: a standardized back door*, in [**94**] (2015), 256–281. URL: https://eprint.iacr.org/2015/767. Citations in this document: §B.

[30] Daniel J. Bernstein, Edoardo Persichetti, *Towards KEM unification* (2018). URL: https://cr.yp.to/papers.html#tightkem. Citations in this document: §10, §6, §17.

[31] Karthikeyan Bhargavan, Gaëtan Leurent, *On the practical (in-)security of 64-bit block ciphers: collision attacks on HTTP over TLS and OpenVPN*, in CCS 2016 [**101**] (2016), 456–467. URL: https://sweet32.info. Citations in this document: §1.

[32] Jean-François Biasse, Fang Song, *Efficient quantum algorithms for computing class groups and solving the principal ideal problem in arbitrary degree number fields*, in SODA 2016 [**64**] (2016), 893–902. URL: https://fangsong.info/files/pubs/BS_SODA16.pdf. Citations in this document: §8.3.

[33] Nina Bindel, Mike Hamburg, Andreas Hülsing, Edoardo Persichetti, *Tighter proofs of CCA security in the quantum random oracle model* (2019). URL: https://eprint.iacr.org/2019/590. Citations in this document: §12, §15.

[34] Carlo Blundo, Stelvio Cimato (editors), *Security in communication networks, 4th international conference, SCN 2004, Amalfi, Italy, September 8–10, 2004, revised selected papers*, Lecture Notes in Computer Science, 3352, Springer, 2005. ISBN 3-540-24301-1. See [98].

[35] Dan Boneh (editor), *Advances in cryptology: CRYPTO 2003, 23rd annual international cryptology conference, Santa Barbara, California, USA, August 17–21, 2003, proceedings*, Lecture Notes in Computer Science, 2729, Springer. ISBN 3-540-40674-3. MR 2005d:94151. See [79].

[36] Joe P. Buhler, Hendrik W. Lenstra, Jr., Carl Pomerance, *Factoring integers with the number field sieve*, in [**66**] (1993), 50–94. URL: https://www.math.leidenuniv.nl/~hwl/PUBLICATIONS/1993e/art.pdf. Citations in this document: §24.

[37] Stefania Cavallar, Bruce Dodson, Arjen K. Lenstra, Walter M. Lioen, Peter L. Montgomery, Brian Murphy, Herman te Riele, Karen Aardal, Jeff Gilchrist, Gérard Guillerm, Paul C. Leyland, Joël Marchand, Francois Morain, Alec Muffett, Chris Putnam, Craig Putnam, Paul Zimmermann, *Factorization of a 512-bit RSA modulus*, in Eurocrypt 2000 [**92**] (2000), 1–18. URL: https://www.iacr.org/archive/eurocrypt2000/1807/18070001-new.pdf. Citations in this document: §1.

[38] Sanjit Chatterjee, Neal Koblitz, Alfred Menezes, Palash Sarkar, *Another look at tightness II: practical issues in cryptography*, in Mycrypt 2016 [**90**] (2017), 21–55. URL: https://eprint.iacr.org/2016/360. Citations in this document: §9.1, §9.1.

[39] Cong Chen, Oussama Danba, Jeffrey Hoffstein, Andreas Hulsing, Joost Rijneveld, John M. Schanck, Peter Schwabe, William Whyte, Zhenfei Zhang, *NTRU: algorithm specifications and supporting documentation* (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §1.3.

[40] Jean-Sébastien Coron, Jesper Buus Nielsen (editors), *Advances in cryptology—EUROCRYPT 2017—36th annual international conference on the theory and applications of cryptographic techniques, Paris, France, April 30–May 4, 2017, proceedings, part I*, Lecture Notes in Computer Science, 10210, 2017. ISBN 978-3-319-56619-1. See [18].

[41] Ronald Cramer, Victor Shoup, *A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack*, in Crypto 1998 [**65**] (1998), 13–25. URL: https://shoup.net/papers/cs.pdf. Citations in this document: §C.

[42] Ivan Damgård, *A "proof-reading" of some issues in cryptography*, in ICALP 2007 [**11**] (2007), 2–11. URL: http://www.daimi.au.dk/ivan/positionpaper.pdf. Citations in this document: §3, §A, §C, §C.

[43] Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, Frederik Vercauteren, *Saber: Module-LWR based key exchange, CPA-secure encryption and CCA-secure KEM*, in Africacrypt 2018 [**58**] (2018), 282–305. URL: https://eprint.iacr.org/2018/230. Citations in this document: §4.1, §4.1.

[44] Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, Frederik Vercauteren, *SABER: Mod-LWR based KEM (round 2 submission)* (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §1.3, §4.1.

[45] Lance Fortnow, Salil P. Vadhan (editors), *Proceedings of the 43rd ACM symposium on theory of computing, STOC 2011, San Jose, CA, USA, 6–8 June 2011*, ACM, 2011. ISBN 978-1-4503-0691-1. See [47].

[46] Craig Gentry, *Fully homomorphic encryption using ideal lattices*, in STOC 2009 [**77**] (2009), 169–178. URL: https://www.cs.cmu.edu/~odonnell/hits09/gentry-homomorphic-encryption.pdf. Citations in this document: §8.3.

[47] Craig Gentry, Daniel Wichs, *Separating succinct non-interactive arguments from all falsifiable assumptions*, in STOC 2011~[**45**] (2011), 99–108. Citations in this document: §C.4.

[48] Oded Goldreich, *Foundations of cryptography, volume 1: basic tools*, Cambridge University Press, 2001. ISBN 978-0521035361. Citations in this document: §9.

[49] Oded Goldreich, *On post-modern cryptography* (2006). URL: https://eprint.iacr.org/2006/461. Citations in this document: §4, §6.

[50] Shafi Goldwasser, Yael Tauman Kalai, *Cryptographic assumptions: a position paper* (2015). URL: https://eprint.iacr.org/2015/907.pdf. Citations in this document: §C, §C.5, §C.5, §C.5, §C.7, §C.7.

[51] Mike Hamburg, *Post-quantum cryptography proposal: ThreeBears* (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §1.3.

[52] Dennis Hofheinz, Kathrin Hövelmanns, Eike Kiltz, *A modular analysis of the Fujisaki-Okamoto transformation*, in TCC 2017-1 [**59**] (2017), 341–371. URL: https://eprint.iacr.org/2017/604. Citations in this document: §5, §5, §10, §5, §5.2, §5.2, §5.3, §6, §6.

[53] Akiko Inoue, Tetsu Iwata, Kazuhiko Minematsu, Bertram Poettering, *Cryptanalysis of OCB2: attacks on authenticity and confidentiality*, in Crypto 2019, to appear (2019). URL: https://eprint.iacr.org/2019/311. Citations in this document: §1.

[54] 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 II*, Lecture Notes in Computer Science, 11477, Springer, 2019. ISBN 978-3-030-17655-6. See [88].

[55] 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*, Lecture Notes in Computer Science, 11478, Springer, 2019. ISBN 978-3-030-17658-7. See [69].

[56] Tetsu Iwata, Jung Hee Cheon (editors), *Advances in cryptology—ASIACRYPT 2015—21st international conference on the theory and application of cryptology and information security, Auckland, New Zealand, November 29–December 3, 2015, proceedings, part I*, Lecture Notes in Computer Science, 9452, Springer, 2015. ISBN 978-3-662-48796-9. See [16].

[57] Michael J. Jacobson Jr., Neal Koblitz, Joseph H. Silverman, Andreas Stein, Edlyn Teske, *Analysis of the xedni calculus attack*, Designs, Codes and Cryptography **20** (2000), 1–64. URL: https://pages.cpsc.ucalgary.ca/~jacobs/PDF/xedni.pdf. Citations in this document: §24.

[58] 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 [43].

[59] Yael Kalai, Leonid Reyzin (editors), *Theory of cryptography—15th international conference, TCC 2017, Baltimore, MD, USA, November 12–15, 2017, proceedings, part I*, Lecture Notes in Computer Science, 10677, Springer, 2017. ISBN 978-3-319-70499-9. See [52].

[60] Jonathan Katz, Yehuda Lindell, *Introduction to modern cryptography*, 2nd edition, Cryptography and Network Security Series, CRC Press, 2018. ISBN 978-1466570269. Citations in this document: §2, §6.1, §8.4, §C.

[61] Jonathan Katz, Hovav Shacham (editors), *Advances in cryptology—CRYPTO 2017—37th annual international cryptology conference, Santa Barbara, CA, USA, August 20–24, 2017, proceedings, part I*, Lecture Notes in Computer Science, 10401, Springer, 2017. ISBN 978-3-319-63687-0. See [13].

[62] Neal Koblitz, Alfred Menezes, *Another look at "provable security"*, Journal of Cryptology **20** (2007), 3–37. URL: https://eprint.iacr.org/2004/152. Citations in this document: §6.2.

[63] Neal Koblitz, Alfred Menezes, *The random oracle model: a twenty-year retrospective*, Designs, Codes and Cryptography **77** (2015), 587–610. URL: https://eprint.iacr.org/2015/140. Citations in this document: §4.

[64] Robert Krauthgamer (editor), *Proceedings of the twenty-seventh annual ACM-SIAM symposium on discrete algorithms, SODA 2016, Arlington, VA, USA, January 10–12, 2016*, SIAM, 2016. ISBN 978-1-61197-433-1. See [32].

[65] Hugo Krawczyk (editor), *Advances in cryptology—CRYPTO '98, 18th annual international cryptology conference, Santa Barbara, California, USA, August 23–27, 1998, proceedings*, Lecture Notes in Computer Science, 1462, Springer, 1998. ISBN 3-540-64892-5. See [41].

[66] Arjen K. Lenstra, Hendrik W. Lenstra, Jr. (editors), *The development of the number field sieve*, Lecture Notes in Mathematics, 1554, Springer, 1993. ISBN 3-540-57013-6. MR 96m:11116. See [36].

[67] Arjen K. Lenstra, Hendrik W. Lenstra, Jr., László Lovász, *Factoring polynomials with rational coefficients*, Mathematische Annalen **261** (1982), 515–534. URL: https://www.math.leidenuniv.nl/~hwl/PUBLICATIONS/1982f/art.pdf. Citations in this document: §6.2.

[68] Andrea Lesavourey, Thomas Plantard, Willy Susilo, *On ideal lattices in multicubic fields*, NutMiC 2019 (2019). URL: http://nutmic2019.imj-prg.fr/confpapers/MultiCubic.pdf. Citations in this document: §8.3.

[69] Gaëtan Leurent, Thomas Peyrin, *From collisions to chosen-prefix collisions: application to full SHA-1*, in Eurocrypt 2019 [**55**] (2019), 527–555. URL: https://eprint.iacr.org/2019/459. Citations in this document: §1.

[70] Yehuda Lindell, *Important food for thought by @hashbreaker about establishing which assumptions are "better" in cryptography: https://eprint.iacr.org/2019/691.pdf. The claims outlined in Katz-Lindell on this turn out to be overly simplified and naive (to be fixed in the next version)*, tweet (2019). URL: http://archive.is/NeuvD. Citations in this document: §22.

[71] Xianhui Lu, Yamin Liu, Dingding Jia, Haiyang Xue, Jingnan He, Zhenfei Zhang, Zhe Liu, Hao Yang, Bao Li, Kunpeng Wang, *LAC: Lattice-based Cryptosystems* (2019). URL: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions. Citations in this document: §1.3, §8.3.

[72] Vadim Lyubashevsky, Chris Peikert, Oded Regev, *On ideal lattices and learning with errors over rings*, Journal of the ACM **60** (2013), Article 43, 35 pages. URL: https://eprint.iacr.org/2012/230. Citations in this document: §6, §9.

[73] Ueli M. Maurer (editor), *Advances in cryptology—EUROCRYPT '96: proceedings of the fifteenth international conference on the theory and application of cryptographic techniques held in Saragossa, May 12–16, 1996*, Lecture Notes in

Computer Science, 1070, Springer, 1996. ISBN 3-540-61186-X. MR 97g:94002. See [19].

[74] Daniele Micciancio, Oded Regev, *Lattice-based cryptography*, in Post-quantum cryptography [**25**] (2009), 147–191. URL: https://cims.nyu.edu/~regev/papers/pqc.pdf. Citations in this document: §27, §27, §27.

[75] Gary L. Miller (editor), *Proceedings of the twenty-eighth annual ACM symposium on the theory of computing, Philadelphia, PA, May 22–24, 1996*, Association for Computing Machinery, 1996. ISBN 0-89791-785-5. MR 97g:68005. See [2].

[76] Victor S. Miller, *Use of elliptic curves in cryptography*, in Crypto 1985 [**102**] (1986), 417–426. MR 88b:68040. URL: https://link.springer.com/content/pdf/10.1007%2F3-540-39799-X_31.pdf. Citations in this document: §24.

[77] Michael Mitzenmacher (editor), *Proceedings of the 41st annual ACM symposium on theory of computing, STOC 2009, Bethesda, MD, USA, May 31–June 2, 2009*, ACM, 2009. ISBN 978-1-60558-506-2. See [46].

[78] Michael Naehrig, *RE: [pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: Frodo* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/pqc-forum/_kBMTq3RM28/Zj2CpnEzBgAJ. Citations in this document: §6.5.

[79] Moni Naor, *On cryptographic assumptions and challenges*, in Crypto 2003 [**35**] (2003), 96–109. URL: https://link.springer.com/content/pdf/10.1007/978-3-540-45146-4_6.pdf. Citations in this document: §C, §C.2, §C.5, §C.6.

[80] Moni Naor, Omer Reingold, *Number-theoretic constructions of efficient pseudo-random functions*, Journal of the ACM **51** (2004), 231–262. Citations in this document: §C.5.

[81] National Institute of Standards and Technology, *Post-quantum cryptography: FAQs* (2017). URL: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/faqs. Citations in this document: §11.

[82] Phong Q. Nguyen, Elisabeth Oswald (editors), *Advances in cryptology—EUROCRYPT 2014—33rd annual international conference on the theory and applications of cryptographic techniques, Copenhagen, Denmark, May 11–15, 2014, proceedings*, Lecture Notes in Computer Science, 8441, Springer, 2014. ISBN 978-3-642-55219-9. See [17].

[83] 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 [95].

[84] Kenneth G. Paterson (editor), *Advances in cryptology—EUROCRYPT 2011—30th annual international conference on the theory and applications of cryptographic techniques, Tallinn, Estonia, May 15–19, 2011, proceedings*, Lecture Notes in Computer Science, 6632, Springer, 2011. ISBN 978-3-642-20464-7. See [100].

[85] Chris Peikert, *A decade of lattice cryptography*, Foundations and Trends in Theoretical Computer Science **10** (2016), 283–424. URL: https://web.eecs.umich.edu/~cpeikert/pubs/lattice-survey.pdf. Citations in this document: §1.4, §6.2.

[86] Chris Peikert, *A useful fact about Ring-LWE that should be known better: it is *at least as hard* to break as NTRU, and likely strictly harder. 1/*, tweet (2017). URL: https://archive.is/B9KEW. Citations in this document: §1.4.

[87] Chris Peikert, *Re: [pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: NTRU Prime* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/

pqc-forum/V1RNjpio5Ng/uzEDmsogAgAJ. Citations in this document: §1.4, §26, §27.

[88] Alice Pellet-Mary, Guillaume Hanrot, Damien Stehlé, *Approx-SVP in ideal lattices with pre-processing* (2019), 685–716 [**54**]. URL: https://eprint.iacr.org/2019/215. Citations in this document: §9.

[89] Edoardo Persichetti, *Improving the efficiency of code-based cryptography*, Ph.D. thesis, 2012. URL: http://persichetti.webs.com/Thesis%20Final.pdf. Citations in this document: §6.

[90] Raphael C.-W. Phan, Moti Yung (editors), *Paradigms in cryptology—Mycrypt 2016. Malicious and exploratory cryptology—second international conference, Mycrypt 2016, Kuala Lumpur, Malaysia, December 1–2, 2016, revised selected papers*, Lecture Notes in Computer Science, 10311, Springer, 2017. ISBN 978-3-319-61272-0. See [38].

[91] Le Trieu Phong, *Re: [pqc-forum] OFFICIAL COMMENT: Frodo* (2018). URL: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/official-comments/Frodo-official-comment.pdf. Citations in this document: §6.4.

[92] Bart Preneel (editor), *Advances in cryptology—EUROCRYPT 2000, international conference on the theory and application of cryptographic techniques, Bruges, Belgium, May 14–18, 2000*, Lecture Notes in Computer Science, 1807, Springer, 2000. ISBN 3-540-67517-5. See [37].

[93] Oded Regev, *On lattices, learning with errors, random linear codes, and cryptography*, Journal of the ACM **56** (2009), article 34. URL: https://cims.nyu.edu/~regev/. Citations in this document: §6.2.

[94] 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].

[95] Tsunekazu Saito, Keita Xagawa, Takashi Yamakawa, *Tightly-secure key-encapsulation mechanism in the quantum random oracle model*, in Eurocrypt 2018 [**83**] (2018), 520–551. URL: https://eprint.iacr.org/2017/1005. Citations in this document: §6.

[96] Palash Sarkar, Subhadip Singha, *Verifying solutions to LWE with implications for concrete security* (2019). URL: https://eprint.iacr.org/2019/728. Citations in this document: §9.1.

[97] Joseph H. Silverman, *The four faces of lifting for the elliptic curve discrete logarithm problem* (2007). URL: https://www.math.brown.edu/~jhs/Presentations/ECC4FacesOfLifts.pdf. Citations in this document: §24.

[98] Joseph H. Silverman, Nigel P. Smart, Frederik Vercauteren, *An algebraic approach to NTRU ($q = 2n$) via Witt vectors and overdetermined systems of nonlinear equations*, in SCN 2004 [**34**] (2005), 278–293. URL: https://core.ac.uk/download/pdf/34291216.pdf. Citations in this document: §8.4.

[99] Damien Stehlé, *Re: [pqc-forum] Re: ROUND 2 OFFICIAL COMMENT: NTRU Prime* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/pqc-forum/V1RNjpio5Ng/zgATcj7_BAAJ. Citations in this document: §1.4.

[100] Damien Stehlé, Ron Steinfeld, *Making NTRU as secure as worst-case problems over ideal lattices*, in Eurocrypt 2011 [**84**] (2011), 27–47. URL: https://www.iacr.org/archive/eurocrypt2011/66320027/66320027.pdf. Citations in this document: §9.2.

[101] Edgar R. Weippl, Stefan Katzenbeisser, Christopher Kruegel, Andrew C. Myers, Shai Halevi (editors), *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, Vienna, Austria, October 24–28, 2016*, ACM, 2016. ISBN 978-1-4503-4139-4. See [31].

[102] Hugh C. Williams (editor), *Advances in cryptology: CRYPTO '85*, Lecture Notes in Computer Science, 218, Springer, 1986. ISBN 3-540-16463-4. MR 87d:94002. See [76].

[103] Zhenfei Zhang, *[pqc-forum] ROUND 2 OFFICIAL COMMENT: LAC* (2019). URL: https://groups.google.com/a/list.nist.gov/d/msg/pqc-forum/ohJZMetXes0/aC1TJzdlCgAJ. Citations in this document: §8.3.

A Choice of terminology

Damgård [**42**] refers to the "provable security" terminology as "somewhat misleading, for (at least) four reasons". He does not explicitly address the terminology "security proof", but his reasons also apply to this terminology.

Roughly, Damgård's list of reasons is the following:

- there is (normally) no proof that P is secure,
- sometimes the reduction is loose,
- P might not be "simple" and "natural" and "well studied", and
- there could be attacks outside T.

This overlaps the classification of risks that I am using. Damgård's second reason matches risk #2 from Section 2. Damgård's fourth reason matches risk #3. Damgård's first and third reasons sound like contributing factors to risk #1: the lack of a proof that P is secure is a prerequisite for the risk that P is not secure, and it is intuitively clear that this first risk is lower for "well studied" problems. However, Damgård's third reason has unclear boundaries—for example, there are different and incompatible concepts of whether P is "natural"—whereas I have been careful to clearly define each category of risks.

Details aside, it is clear that there is a mismatch between (1) the way that cryptographers use the terminology "provable security" and "security proof" and (2) the literal meaning of this terminology. I have not found examples of cryptographers disputing the idea that this mismatch misleads the broad community of cryptographic users.

On the other hand, this paper is aimed at cryptographers, and I do not think that switching to alternatives such as "reductionist security" and "reduction proof" would add any clarity for this audience.

For comparison, my impression is that the false analogy between "OW-CPA" and IND-CPA actively misleads cryptographers into an overly narrow view of the extra obstacles that "OW-CPA" poses for the attacker. Switching from "OW-CPA" to "OW-Passive" eliminates the incorrect suggestion that the attacker can choose plaintexts, and eliminates the incorrect suggestion that the attacker knows plaintexts. I think this increase in clarity outweighs the disadvantages of changing terminology.

B Engineering cryptographic standards

The literature on software engineering contains many methods to design and develop software. These methods have various goals, such as

- correctness of the resulting software,
- fitting the software execution within constraints on CPU time, and
- fitting the software development within constraints on human time.

There are hypotheses and experimental studies and analyses regarding, e.g., the effectiveness of different software-engineering techniques in eliminating various types of software failures.

One can similarly treat methods of developing cryptographic standards as an object of study. Goals include fitting cryptographic software execution within constraints on CPU time; fitting standards development within constraints on human time; and, of course, security. Studying the security of the standardization process is not a new idea: for example, Bernstein, Lange, and Niederhagen in [**28**] and [**29**] describe Dual EC as part of an attack against the process of "designing, evaluating, standardizing, selecting, implementing, and deploying PRNGs", and comment that this process "is a broader attack target than any particular RNG".

The standardization process, viewed broadly, includes the process of designing cryptosystems, the process of evaluating the designs, and the process of selecting a subset of the designs for standardization. Even when the standardization process is not under attack, one can ask how reliable it is at producing secure standards. My main interest in this paper is in one component of the evaluation process, namely the way that the designs are reviewed for security.

In software engineering, there is overwhelming evidence of variation in how long bugs take to be discovered. Software-reliability models use the available data to predict how many bugs remain. In cryptography, there has been far less study of the time for security problems to be discovered. One might hope that security problems in cryptographic designs are relatively easy to find, since the total volume of cryptographic designs is far below the total volume of software; but there are far fewer cryptanalysts than software reviewers, and a successful attack on a simple-sounding cryptosystem sometimes requires years of algorithm development.

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. Of course, the risks also depend on the design processes. As extreme examples, if each design is secure than whichever designs are standardized will be secure, and if each design has security problems than whichever designs are standardized will have security problems. Otherwise some designs are secure and others are not, and security review is critical so that we can tell the difference.

C Previous work on criteria for evaluating proofs

> *The significance of our contribution is that it provides a scheme that is provably secure and practical at the same time. There appears to be no other encryption scheme in the literature that enjoys both of these properties simultaneously.* —Cramer and Shoup [41]

> *As long as no one has proved that such an attack does not exist, we simply dont know whether the extra ingredient is superfluous or essential to security.*
> *We believe that the only reasonable approach is to construct cryptographic systems with the objective of being able to give security reductions for them.* —Damgård [42]

> *The two principles described above allow us to achieve our goal of providing a rigorous proof that a construction satisfies a given definition under certain specified assumptions. Such proofs are especially important in the context of cryptography where there is an attacker who is actively trying to "break" some scheme.* —Katz and Lindell [60]

The cryptographic literature frequently claims that some cryptographic systems are "provably secure" while others are not. This claim is the starting point for various assertions that having a "security proof" is an important security feature of a cryptographic system.

However, this claim is obviously wrong. Every system X has a "security proof": a useless proof concluding that X is secure under the assumption that X is secure, as in Section 3. Here are examples of previous literature pointing this out:

- In a hypothetical example of a proof for a signature scheme, Naor [79] asks "whether the assumption Alice uses is really weaker than the assumption 'this signature scheme is secure'."
- Damgård [42] notes the risk of "useless statements that are not much better that saying 'the system is secure under the assumption that it is secure'."
- Goldwasser and Kalai [50] note the possibility of "an absurdum, where the underlying assumption is that the scheme itself is secure, which will eventually endanger the mere existence of our field".

One would thus expect the claim to be withdrawn. One would also expect cryptographers to formulate and scientifically evaluate other hypotheses regarding the relationship between "security proofs" and security.

Two of the papers cited above have proposed criteria that could be used to reject proofs:

- Naor says that his "main question" is how to "differentiate between the strengths of assumptions and avoid circularity in our arguments". He then proposes to classify hardness assumptions according to their level of "falsifiability".

- Goldwasser and Kalai write that "we are in greet need of *measures* which will capture which assumptions are 'safe'," claim that Naor's classification "has proved to be too inclusive", and propose a "stricter classification".

At first glance, these two papers seem to be aiming for the same objective as Section 3, formulating criteria that allow decision-makers to systematically reject useless proofs. This appendix briefly surveys the criteria, analyzes the impact of the criteria, and gives a case study of the criteria producing different conclusions from the review-cost metric.

C.1. "Somewhat falsifiability": verifying that an attack works. A cryptanalyst publishes software and claims that the software takes time 2^{32} to achieve about 10% advantage against the IND-CCA2 security of a specified fast KEM. Anyone can verify this claim by running the software many times and observing that its advantage in these experiments is about 10%.

Naor defines "somewhat falsifiable" as an asymptotic formalization of this scenario. Asymptotic IND-CCA2 security is "somewhat falsifiable" under this definition. Most[28] asymptotic security concepts in the literature are "somewhat falsifiable".

C.2. "Falsifiability": verifying solutions to non-interactive problems. The 2048-bit exponent-3 RSA problem is to find an integer x modulo a 2048-bit RSA modulus N, given N and $x^3 \bmod N$. A cryptanalyst claims to have an algorithm taking at most a day—on whatever amount of hardware is available to the cryptanalyst—to break this problem with probability 1%. A verifier publishes a new 2048-bit exponent-3 RSA problem each day, and sees how many of the problems are solved by the cryptanalyst within a day. If the claim is correct then the cryptanalyst will succeed a few times each year.

Seeing these solutions convinces the verifier of a quantitatively weaker version of the claim. The cryptanalyst could be secretly using much more hardware than claimed. The cryptanalyst could have an attack that works with considerably lower success probability, and could be amplifying the success probability by applying a multi-target attack to many verifiers. But clearly the cryptanalyst does have a noticeable probability of breaking the RSA problem.

Naor defines "falsifiable" as an asymptotic formalization of this scenario. An asymptotic RSA problem is "falsifiable". Asymptotic IND-CCA2 security of a KEM does not seem to be "falsifiable": the verifier must not merely issue challenges, but must also answer decapsulation queries from the cryptanalyst. Asymptotic PRF security does not seem to be "falsifiable".

Asymptotic OW-CPA security is "falsifiable". Asymptotic IND-CPA security does not seem to be "falsifiable": Naor's definitions require solutions to be publicly verifiable without further input from the verifier. For the same reason, asymptotic PRG security does not seem to be "falsifiable". Naor claims the op-

[28] There are occasional exceptions, such as "knowledge of exponent" assumptions. Running a claimed "knowledge of exponent" attack many times does not provide enough information to be sure that the attack is successful.

posite in [79, page 103] but then several lines later notes that this claim is "in violation" of the definition of "falsifiable".

C.3. "Efficient falsifiability": verifying solutions to publicly generated challenges. Naor also defines a stronger notion, "efficient falsifiability", in which the cost of verifying a probability-p attack is asymptotically limited to $(\log(1/p))^{O(1)}$ as $p \to 0$. Cost $(1/p)^{O(1)}$ is still permitted for the cryptanalyst.

An asymptotic RSA problem does not seem to be "efficiently falsifiable". An asymptotic factoring problem is "efficiently falsifiable": e.g., ask the cryptanalyst to factor $H(r, s)$ into two same-length primes, where H produces output of the appropriate length, r is a long string chosen by the verifier, and s is a short string chosen by the cryptanalyst. "Short" is defined with somewhat more than $\log_2(1/p)$ bits, so that there are likely to be some choices of $H(r, s)$ that are products of two same-length primes and that are factored by the cryptanalyst's probability-p attack, whereas there are unlikely to be any such choices for an attack that actually has much lower success probability. The cryptanalyst must try factoring $H(r, s)$ for many choices of s, but the verifier's work is merely checking one successful factorization into two same-length primes.

C.4. Further definitions. Gentry and Wichs [47] give another definition that seems to have similar properties to Naor's definition of "somewhat falsifiable". Gentry and Wichs observe that most[29] asymptotic security concepts meet their definition. Beware that there is a clash of terminology: Gentry and Wichs use the name "falsifiable" for their definition, while many examples of security concepts meeting this definition do not seem to meet Naor's definition of "falsifiable".

Goldwasser and Kalai give a further definition that seems to have similar properties to Naor's definition of "falsifiable", formalizing the notion of non-interactive problems. Specifically, a "search complexity assumption"

- specifies an efficient algorithm to generate a random challenge x;
- specifies an efficient algorithm to verify a relationship between x and y; and
- assumes that every efficient attack, given x, has asymptotically negligible chance of finding y related to x.

Goldwasser and Kalai also give several variants of this definition, such as a "privately-verifiable search complexity assumption" and a "decisional complexity assumption", so as to be able to include (e.g.) IND-CPA as a "complexity assumption".

C.5. Reductions that eliminate interactivity, and a case study. Naor writes that falsifiability of an assumption "should be a major consideration in how acceptable the assumption is". For example, Naor criticizes PRF assumptions in general as being merely "somewhat falsifiable" and seemingly not being "falsifiable", while Naor praises some specific constructions of PRFs from "falsifiable" number-theoretic assumptions—even better, "efficiently falsifiable" number-theoretic assumptions.

[29] They show, however, that black-box reductions for succinct non-interactive arguments of knowledge cannot begin from assumptions meeting their definition.

Here is a case study of Naor's classification. The lattice KEMs covered in this paper are expected to be incorporated into KEM-DEM constructions that use a PRG to encrypt a user message. Consider the following two options for this PRG:

- AES-256-CTR.
- The Naor–Reingold cipher. Map (say) a 32-bit block counter $c_1, c_2, \ldots, c_{32}$ to $g^{a_0 a_1^{c_1} \cdots a_{32}^{c_{32}}} \bmod p$, and then extract 128 bits from the output as in [80, Construction 4.2]. Here g has 256-bit prime order q modulo a 2048-bit prime p, and $a_0, \ldots, a_{32}$ are secret integers between 1 and $q - 1$.

Regarding the first option, there is a proof of PRG security of AES-256-CTR assuming the PRP security of AES-256. Naor's "falsifiability" definition creates two obstacles to this PRP assumption:

- The definition is asymptotic, involving unspecified "polynomial" costs, so it is content-free for specific sizes such as 256 bits. This does not seem to be an intentional obstacle: Naor claims, incorrectly, that his definitions are "concrete" and not "asymptotic".
- The PRP assumption for AES-256 seems inherently interactive, while the "falsifiability" definition does not allow interactivity. This is obviously an intentional obstacle. This obstacle is shared by the "complexity assumption" definitions in [50].

Regarding the second option, there is a proof of asymptotic PRF (and thus PRG) security for the Naor–Reingold cipher, assuming asymptotic DDH hardness for the multiplicative group $(\mathbf{Z}/p)^*$. It is not clear to me that asymptotic DDH hardness is "falsifiable"—again, Naor's definitions require public verifiability—but asymptotic DDH hardness does qualify as a "complexity assumption" in [50].

To summarize, it is clear that Naor prefers the DDH assumption over the AES-256 PRP assumption, because the AES-256 PRP assumption is interactive while the DDH assumption is not (except for a private verification step at the end). An asymptotic version of this preference does not seem to be formalized by the definitions in [79] (because of the private verification step) but is formalized by the definitions in [50].

A surprising feature of this case study is that, by the same reasoning, Naor should also prefer the AES-256-CTR PRG assumption over the AES-256 PRP assumption, since the PRG assumption is not interactive (except for a private verification step). This is directly contrary to the usual idea that it is useful to prove AES-256-CTR PRG security, AES-256-OFB PRG security, etc. from AES-256 PRP security.

C.6. Comparison to the review-cost metric for the case study. The Naor–Reingold cipher described above is much worse than AES-256-CTR in the review-cost metric. As noted in Section 8, the multiplicative-group discrete-logarithm problem provides many tools to the attacker, which have been combined into very complicated attacks. Cryptanalysts also have to ask whether DDH attacks can be even faster than DL attacks, and whether the looseness of the Naor–Reingold proof can be exploited. The attack avenues against the PRP assumption for AES-256 are less numerous and less complicated, and have been explored much more thoroughly. There is a looseness issue in the AES-256-CTR proof, but this issue is relatively small.[30]

I am not saying that non-interactivity has zero value. The AES-256-CTR reviewer has to ask, among other things, whether attacks against AES-256 have been correctly analyzed; non-interactive verification of attacks might help weed out errors in the analysis. What I am saying is that this narrow focus on non-interactivity loses sight of many important security risks, and in particular is blind to the long history of security losses in the DL problem. Naor correctly notes in [79, page 107] that his definitions ignore the "history of computational attempts", but does not point out that this can reverse the preference between two assumptions.

A security reviewer could simply throw away the Naor–Reingold proof[31] and directly review attacks against the Naor–Reingold PRG. The extraction of 128 bits from integers modulo p might make this PRG hard to attack even for an attacker who can compute discrete logarithms. But how can one claim a thorough security review of this possibility? Beyond DL and DDH security, the security of this PRG has attracted a negligible level of attention from cryptanalysts. This situation seems unlikely to change: there are many other cryptanalytic targets whose importance is much more obvious.

C.7. Further comparison to the review-cost metric. It is quite unclear how to convert "falsifiability" into a constraint upon proofs. Naor writes that, for some "major results" in cryptography, "both the assumption and the outcome are in the class of falsifiable tasks"; surely Naor does not advocate ignoring "major results". It is not even clear how Naor suggests handling his "main question" of how to "avoid circularity in our arguments". Similarly, it is not clear how the definitions in [50] can prevent the "absurdum" described in [50].

For comparison, the review-cost metric provides a straightforward rule for deciding whether to allow a "security proof": if the proof does not save time in a thorough security review then the proof is skipped. This rule implies that circular proofs are skipped; see Section 3. This rule also applies directly to concrete cryptosystems such as the 36 target KEMs.

[30] The proof allows q-block attacks to succeed with probability approximately $q^2/2^{129}$ plus the AES-256 attack probability. Is $q^2/2^{129}$ an acceptable attack probability? Switching from AES-256-CTR to Salsa20 or ChaCha20 resolves this issue. For comparison, the Naor–Reingold proof allows the DDH attack probability to be *multiplied* by $\log_2 q$.

[31] This is mandatory in post-quantum cryptography, since the assumption in the Naor–Reingold proof is broken in quantum polynomial time: Shor's algorithm efficiently computes discrete logarithms in $(\mathbf{Z}/p)^*$.

system	parameter set	type	set of multipliers
frodo	640	Product	$(\mathbf{Z}/32768)^{640\times 640}$
frodo	976	Product	$(\mathbf{Z}/65536)^{976\times 976}$
frodo	1344	Product	$(\mathbf{Z}/65536)^{1344\times 1344}$
kyber	512	Product	$((\mathbf{Z}/3329)[x]/(x^{256}+1))^{2\times 2}$
kyber	768	Product	$((\mathbf{Z}/3329)[x]/(x^{256}+1))^{3\times 3}$
kyber	1024	Product	$((\mathbf{Z}/3329)[x]/(x^{256}+1))^{4\times 4}$
lac	128	Product	$(\mathbf{Z}/251)[x]/(x^{512}+1)$
lac	192	Product	$(\mathbf{Z}/251)[x]/(x^{1024}+1)$
lac	256	Product	$(\mathbf{Z}/251)[x]/(x^{1024}+1)$
newhope	512	Product	$(\mathbf{Z}/12289)[x]/(x^{512}+1)$
newhope	1024	Product	$(\mathbf{Z}/12289)[x]/(x^{1024}+1)$
ntru	hps2048509	Quotient	$(\mathbf{Z}/2048)[x]/(x^{509}-1)$
ntru	hps2048677	Quotient	$(\mathbf{Z}/2048)[x]/(x^{677}-1)$
ntru	hps4096821	Quotient	$(\mathbf{Z}/4096)[x]/(x^{821}-1)$
ntru	hrss701	Quotient	$(\mathbf{Z}/8192)[x]/(x^{701}-1)$
ntrulpr	653	Product	$(\mathbf{Z}/4621)[x]/(x^{653}-x-1)$
ntrulpr	761	Product	$(\mathbf{Z}/4591)[x]/(x^{761}-x-1)$
ntrulpr	857	Product	$(\mathbf{Z}/5167)[x]/(x^{857}-x-1)$
round5n1	1	Product	$(\mathbf{Z}/4096)^{636\times 636}$
round5n1	3	Product	$(\mathbf{Z}/32768)^{876\times 876}$
round5n1	5	Product	$(\mathbf{Z}/32768)^{1217\times 1217}$
round5nd	1.0d	Product	$(\mathbf{Z}/8192)[x]/(x^{586}+\cdots+1)$
round5nd	3.0d	Product	$(\mathbf{Z}/4096)[x]/(x^{852}+\cdots+1)$
round5nd	5.0d	Product	$(\mathbf{Z}/8192)[x]/(x^{1170}+\cdots+1)$
round5nd	1.5d	Product	$(\mathbf{Z}/1024)[x]/(x^{509}-1)$
round5nd	3.5d	Product	$(\mathbf{Z}/4096)[x]/(x^{757}-1)$
round5nd	5.5d	Product	$(\mathbf{Z}/2048)[x]/(x^{947}-1)$
saber	light	Product	$((\mathbf{Z}/8192)[x]/(x^{256}+1))^{2\times 2}$
saber	main	Product	$((\mathbf{Z}/8192)[x]/(x^{256}+1))^{3\times 3}$
saber	fire	Product	$((\mathbf{Z}/8192)[x]/(x^{256}+1))^{4\times 4}$
sntrup	653	Quotient	$(\mathbf{Z}/4621)[x]/(x^{653}-x-1)$
sntrup	761	Quotient	$(\mathbf{Z}/4591)[x]/(x^{761}-x-1)$
sntrup	857	Quotient	$(\mathbf{Z}/5167)[x]/(x^{857}-x-1)$
threebears	baby	Product	$(\mathbf{Z}/(2^{3120}-2^{1560}-1))^{2\times 2}$
threebears	mama	Product	$(\mathbf{Z}/(2^{3120}-2^{1560}-1))^{3\times 3}$
threebears	papa	Product	$(\mathbf{Z}/(2^{3120}-2^{1560}-1))^{4\times 4}$

Table 8.6. Set of multipliers for each of the target PKEs. Public key reveals multiplier G and reveals approximation A to aG, where a is short. "Quotient": Quotient NTRU; $A = 0$, so G is generated as a quotient. "Product": Product NTRU; G is generated randomly, and then A is generated as an approximation to aG. See Table 8.7 for distribution of short elements. See Table 8.8 for offsets from aG to A.

system	parameter set	short element
frodo	640	$\mathbf{Z}^{640\times 8}$; $\{-12,\dots,12\}$; Pr $1,4,17,\dots$ (spec page 23)
frodo	976	$\mathbf{Z}^{976\times 8}$; $\{-10,\dots,10\}$; Pr $1,6,29,\dots$ (spec page 23)
frodo	1344	$\mathbf{Z}^{1344\times 8}$; $\{-6,\dots,6\}$; Pr $2,40,364,\dots$ (spec page 23)
kyber	512	$(\mathbf{Z}[x]/(x^{256}+1))^2$; $\sum_{0\le i<4}\{-0.5,0.5\}$
kyber	768	$(\mathbf{Z}[x]/(x^{256}+1))^3$; $\sum_{0\le i<4}\{-0.5,0.5\}$
kyber	1024	$(\mathbf{Z}[x]/(x^{256}+1))^4$; $\sum_{0\le i<4}\{-0.5,0.5\}$
lac	128	$\mathbf{Z}[x]/(x^{512}+1)$; $\{-1,0,1\}$; Pr $1,2,1$; weight $128,128$
lac	192	$\mathbf{Z}[x]/(x^{1024}+1)$; $\{-1,0,1\}$; Pr $1,6,1$; weight $128,128$
lac	256	$\mathbf{Z}[x]/(x^{1024}+1)$; $\{-1,0,1\}$; Pr $1,2,1$; weight $256,256$
newhope	512	$\mathbf{Z}[x]/(x^{512}+1)$; $\sum_{0\le i<16}\{-0.5,0.5\}$
newhope	1024	$\mathbf{Z}[x]/(x^{1024}+1)$; $\sum_{0\le i<16}\{-0.5,0.5\}$
ntru	hps2048509	$\mathbf{Z}[x]/(x^{509}-1)$; $\{-1,0,1\}$
ntru	hps2048677	$\mathbf{Z}[x]/(x^{677}-1)$; $\{-1,0,1\}$
ntru	hps4096821	$\mathbf{Z}[x]/(x^{821}-1)$; $\{-1,0,1\}$
ntru	hrss701	$\mathbf{Z}[x]/(x^{701}-1)$; $\{-1,0,1\}$; key correlation ≥ 0
ntrulpr	653	$\mathbf{Z}[x]/(x^{653}-x-1)$; $\{-1,0,1\}$; weight 252
ntrulpr	761	$\mathbf{Z}[x]/(x^{761}-x-1)$; $\{-1,0,1\}$; weight 250
ntrulpr	857	$\mathbf{Z}[x]/(x^{857}-x-1)$; $\{-1,0,1\}$; weight 281
round5n1	1	$\mathbf{Z}^{636\times 8}$; $\{-1,0,1\}$; weight $57,57$
round5n1	3	$\mathbf{Z}^{876\times 8}$; $\{-1,0,1\}$; weight $223,223$
round5n1	5	$\mathbf{Z}^{1217\times 8}$; $\{-1,0,1\}$; weight $231,231$
round5nd	1.0d	$\mathbf{Z}[x]/(x^{586}+\cdots+1)$; $\{-1,0,1\}$; weight $91,91$
round5nd	3.0d	$\mathbf{Z}[x]/(x^{852}+\cdots+1)$; $\{-1,0,1\}$; weight $106,106$
round5nd	5.0d	$\mathbf{Z}[x]/(x^{1170}+\cdots+1)$; $\{-1,0,1\}$; weight $111,111$
round5nd	1.5d	$\mathbf{Z}[x]/(x^{509}-1)$; $\{-1,0,1\}$; weight $68,68$; ending 0
round5nd	3.5d	$\mathbf{Z}[x]/(x^{757}-1)$; $\{-1,0,1\}$; weight $121,121$; ending 0
round5nd	5.5d	$\mathbf{Z}[x]/(x^{947}-1)$; $\{-1,0,1\}$; weight $194,194$; ending 0
saber	light	$(\mathbf{Z}[x]/(x^{256}+1))^2$; $\sum_{0\le i<10}\{-0.5,0.5\}$
saber	main	$(\mathbf{Z}[x]/(x^{256}+1))^3$; $\sum_{0\le i<8}\{-0.5,0.5\}$
saber	fire	$(\mathbf{Z}[x]/(x^{256}+1))^4$; $\sum_{0\le i<6}\{-0.5,0.5\}$
sntrup	653	$\mathbf{Z}[x]/(x^{653}-x-1)$; $\{-1,0,1\}$; weight 288
sntrup	761	$\mathbf{Z}[x]/(x^{761}-x-1)$; $\{-1,0,1\}$; weight 286
sntrup	857	$\mathbf{Z}[x]/(x^{857}-x-1)$; $\{-1,0,1\}$; weight 322
threebears	baby	$\mathbf{Z}^2$; $\sum_{0\le i<312}2^{10i}\{-2,-1,0,1,2\}$; Pr $1,32,62,32,1$; *
threebears	mama	$\mathbf{Z}^3$; $\sum_{0\le i<312}2^{10i}\{-1,0,1\}$; Pr $13,38,13$; *
threebears	papa	$\mathbf{Z}^4$; $\sum_{0\le i<312}2^{10i}\{-1,0,1\}$; Pr $5,22,5$; *

Table 8.7. Distribution of short elements for each of the target PKEs. General format: set of polynomials or vectors or matrices; distribution of each integer coefficient. By default, element of each "$\{\dots\}$" is chosen uniformly at random, but "Pr" indicates a different distribution; "weight w" indicates that coefficients have Hamming weight w; "weight w_-,w_+" indicates that w_- coefficients are -1 and w_+ coefficients are $+1$; "ending 0" indicates that last coefficient is 0; "key correlation $\ge$ 0" indicates that $\sum_i a_i a_{i+1} \ge 0$ for key generation; "*" for threebears indicates that short elements generated as shown in table are then multiplied by $2^{1560}-1$. For key generation, short vectors and short matrices generated as shown in table are then transposed.

system	parameter set	key offset (numerator or noise or rounding method)
`frodo`	640	$\mathbf{Z}^{640\times 8}$; $\{-12,\dots,12\}$; Pr $1,4,17,\dots$ (spec page 23)
`frodo`	976	$\mathbf{Z}^{976\times 8}$; $\{-10,\dots,10\}$; Pr $1,6,29,\dots$ (spec page 23)
`frodo`	1344	$\mathbf{Z}^{1344\times 8}$; $\{-6,\dots,6\}$; Pr $2,40,364,\dots$ (spec page 23)
`kyber`	512	$(\mathbf{Z}[x]/(x^{256}+1))^2$; $\sum_{0\le i<4}\{-0.5,0.5\}$
`kyber`	768	$(\mathbf{Z}[x]/(x^{256}+1))^3$; $\sum_{0\le i<4}\{-0.5,0.5\}$
`kyber`	1024	$(\mathbf{Z}[x]/(x^{256}+1))^4$; $\sum_{0\le i<4}\{-0.5,0.5\}$
`lac`	128	$\mathbf{Z}[x]/(x^{512}+1)$; $\{-1,0,1\}$; Pr $1,2,1$; weight $128,128$
`lac`	192	$\mathbf{Z}[x]/(x^{1024}+1)$; $\{-1,0,1\}$; Pr $1,6,1$; weight $128,128$
`lac`	256	$\mathbf{Z}[x]/(x^{1024}+1)$; $\{-1,0,1\}$; Pr $1,2,1$; weight $256,256$
`newhope`	512	$\mathbf{Z}[x]/(x^{512}+1)$; $\sum_{0\le i<16}\{-0.5,0.5\}$
`newhope`	1024	$\mathbf{Z}[x]/(x^{1024}+1)$; $\sum_{0\le i<16}\{-0.5,0.5\}$
`ntru`	`hps2048509`	$\mathbf{Z}[x]/(x^{509}-1)$; $\{-1,0,1\}$; weight $127,127$
`ntru`	`hps2048677`	$\mathbf{Z}[x]/(x^{677}-1)$; $\{-1,0,1\}$; weight $127,127$
`ntru`	`hps4096821`	$\mathbf{Z}[x]/(x^{821}-1)$; $\{-1,0,1\}$; weight $255,255$
`ntru`	`hrss701`	$\mathbf{Z}[x]/(x^{701}-1)$; $\{-1,0,1\}$; key correlation ≥ 0; $\cdot(x-1)$
`ntrulpr`	653	round $\{-2310,\dots,2310\}$ to $3\mathbf{Z}$
`ntrulpr`	761	round $\{-2295,\dots,2295\}$ to $3\mathbf{Z}$
`ntrulpr`	857	round $\{-2583,\dots,2583\}$ to $3\mathbf{Z}$
`round5n1`	1	round $\mathbf{Z}/4096$ to $8\mathbf{Z}$
`round5n1`	3	round $\mathbf{Z}/32768$ to $16\mathbf{Z}$
`round5n1`	5	round $\mathbf{Z}/32768$ to $8\mathbf{Z}$
`round5nd`	`1.0d`	round $\mathbf{Z}/8192$ to $16\mathbf{Z}$
`round5nd`	`3.0d`	round $\mathbf{Z}/4096$ to $8\mathbf{Z}$
`round5nd`	`5.0d`	round $\mathbf{Z}/8192$ to $16\mathbf{Z}$
`round5nd`	`1.5d`	reduce mod $x^{508}+\cdots+1$; round $\mathbf{Z}/1024$ to $8\mathbf{Z}$
`round5nd`	`3.5d`	reduce mod $x^{756}+\cdots+1$; round $\mathbf{Z}/4096$ to $16\mathbf{Z}$
`round5nd`	`5.5d`	reduce mod $x^{946}+\cdots+1$; round $\mathbf{Z}/2048$ to $8\mathbf{Z}$
`saber`	`light`	round $\mathbf{Z}/8192$ to $8\mathbf{Z}$
`saber`	`main`	round $\mathbf{Z}/8192$ to $8\mathbf{Z}$
`saber`	`fire`	round $\mathbf{Z}/8192$ to $8\mathbf{Z}$
`sntrup`	653	$\mathbf{Z}[x]/(x^{653}-x-1)$; $\{-1,0,1\}$; invertible mod 3
`sntrup`	761	$\mathbf{Z}[x]/(x^{761}-x-1)$; $\{-1,0,1\}$; invertible mod 3
`sntrup`	857	$\mathbf{Z}[x]/(x^{857}-x-1)$; $\{-1,0,1\}$; invertible mod 3
`threebears`	`baby`	$\mathbf{Z}^2$; $\sum_{0\le i<312} 2^{10i}\{-2,-1,0,1,2\}$; Pr $1,32,62,32,1$; *
`threebears`	`mama`	$\mathbf{Z}^3$; $\sum_{0\le i<312} 2^{10i}\{-1,0,1\}$; Pr $13,38,13$; *
`threebears`	`papa`	$\mathbf{Z}^4$; $\sum_{0\le i<312} 2^{10i}\{-1,0,1\}$; Pr $5,22,5$; *

Table 8.8. How an approximation $A \approx aG$ is obtained for each of the target PKEs. For Quotient NTRU (`ntru` and `sntrup`), e is generated randomly as shown in the table; $G = e/a$; and $A = 0$. For Noisy Product NTRU (`frodo`, `kyber`, `lac`, `newhope`, `threebears`), e is generated randomly as shown in the table; G is generated randomly; and $A = aG + e$. For Rounded Product NTRU (`ntrulpr`, `round5n1`, `round5nd`, `saber`), A is obtained by rounding aG as shown in the table. Random generation is specified as in Table 8.7, with the following additional modifiers: "$\cdot(x-1)$" for `ntruhrss701` means that, for key generation, e is then multiplied by $x-1$; "invertible mod 3" for `sntrup` means that e is required to be invertible in $(\mathbf{Z}/3)[x]/(x^n - x - 1)$.

system	parameter set	ciphertext offset (noise or rounding method)
`frodo`	`640`	$\mathbf{Z}^{8\times 8}$; $\{-12,\dots,12\}$; Pr $1,4,17,\dots$ (spec page 23)
`frodo`	`976`	$\mathbf{Z}^{8\times 8}$; $\{-10,\dots,10\}$; Pr $1,6,29,\dots$ (spec page 23)
`frodo`	`1344`	$\mathbf{Z}^{8\times 8}$; $\{-6,\dots,6\}$; Pr $2,40,364,\dots$ (spec page 23)
`kyber`	`512`	$\mathbf{Z}[x]/(x^{256}+1)$; $\sum_{0\le i<4}\{-0.5,0.5\}$
`kyber`	`768`	$\mathbf{Z}[x]/(x^{256}+1)$; $\sum_{0\le i<4}\{-0.5,0.5\}$
`kyber`	`1024`	$\mathbf{Z}[x]/(x^{256}+1)$; $\sum_{0\le i<4}\{-0.5,0.5\}$
`lac`	`128`	$\mathbf{Z}[x]/(x^{512}+1)$; $\{-1,0,1\}$; Pr $1,2,1$
`lac`	`192`	$\mathbf{Z}[x]/(x^{1024}+1)$; $\{-1,0,1\}$; Pr $1,6,1$
`lac`	`256`	$\mathbf{Z}[x]/(x^{1024}+1)$; $\{-1,0,1\}$; Pr $1,2,1$
`newhope`	`512`	$\mathbf{Z}[x]/(x^{512}+1)$; $\sum_{0\le i<16}\{-0.5,0.5\}$
`newhope`	`1024`	$\mathbf{Z}[x]/(x^{1024}+1)$; $\sum_{0\le i<16}\{-0.5,0.5\}$
`ntru`	`hps2048509`	not applicable
`ntru`	`hps2048677`	not applicable
`ntru`	`hps4096821`	not applicable
`ntru`	`hrss701`	not applicable
`ntrulpr`	`653`	bottom 256 coeffs; $z\mapsto\lfloor(114(z+2156)+16384)/32768\rfloor$
`ntrulpr`	`761`	bottom 256 coeffs; $z\mapsto\lfloor(113(z+2175)+16384)/32768\rfloor$
`ntrulpr`	`857`	bottom 256 coeffs; $z\mapsto\lfloor(101(z+2433)+16384)/32768\rfloor$
`round5n1`	`1`	round $\mathbf{Z}/4096$ to $64\mathbf{Z}$
`round5n1`	`3`	round $\mathbf{Z}/32768$ to $512\mathbf{Z}$
`round5n1`	`5`	round $\mathbf{Z}/32768$ to $64\mathbf{Z}$
`round5nd`	`1.0d`	bottom 128 coeffs; round $\mathbf{Z}/8192$ to $512\mathbf{Z}$
`round5nd`	`3.0d`	bottom 192 coeffs; round $\mathbf{Z}/4096$ to $128\mathbf{Z}$
`round5nd`	`5.0d`	bottom 256 coeffs; round $\mathbf{Z}/8192$ to $256\mathbf{Z}$
`round5nd`	`1.5d`	bottom 318 coeffs; round $\mathbf{Z}/1024$ to $64\mathbf{Z}$
`round5nd`	`3.5d`	bottom 410 coeffs; round $\mathbf{Z}/4096$ to $512\mathbf{Z}$
`round5nd`	`5.5d`	bottom 490 coeffs; round $\mathbf{Z}/2048$ to $64\mathbf{Z}$
`saber`	`light`	round $\mathbf{Z}/8192$ to $1024\mathbf{Z}$
`saber`	`main`	round $\mathbf{Z}/8192$ to $512\mathbf{Z}$
`saber`	`fire`	round $\mathbf{Z}/8192$ to $128\mathbf{Z}$
`sntrup`	`653`	not applicable
`sntrup`	`761`	not applicable
`sntrup`	`857`	not applicable
`threebears`	`baby`	$\mathbf{Z}$; $\sum_{0\le i<312}2^{10i}\{-2,-1,0,1,2\}$; Pr $1,32,62,32,1$; *
`threebears`	`mama`	$\mathbf{Z}$; $\sum_{0\le i<312}2^{10i}\{-1,0,1\}$; Pr $13,38,13$; *
`threebears`	`papa`	$\mathbf{Z}$; $\sum_{0\le i<312}2^{10i}\{-1,0,1\}$; Pr $5,22,5$; *

Table 8.9. How an approximation $C\approx Ab+M$ is obtained for each of the target Product NTRU PKEs. For Noisy Product NTRU (`frodo`, `kyber`, `lac`, `newhope`, `threebears`), c is generated randomly as shown in the table, and $C=Ab+M+c$, except that `threebears` adds M in a more complicated way to $Ab+c$ (see the specification). For Rounded Product NTRU (`ntrulpr`, `round5n1`, `round5nd`, `saber`), C is obtained by rounding $Ab+M$ as shown in the table.

system	parameter set	set of encoded messages
`frodo`	`640`	8×8 matrix over $\{0, 8192, 16384, 24576\}$
`frodo`	`976`	8×8 matrix over $\{0, 8192, \dots, 57344\}$
`frodo`	`1344`	8×8 matrix over $\{0, 4096, \dots, 61440\}$
`kyber`	`512`	$\sum_{0 \le i < 256} \{0, 1665\} x^i$
`kyber`	`768`	$\sum_{0 \le i < 256} \{0, 1665\} x^i$
`kyber`	`1024`	$\sum_{0 \le i < 256} \{0, 1665\} x^i$
`lac`	`128`	256-dim subcode (see spec) of $\sum_{0 \le i < 512} \{0, 126\} x^i$
`lac`	`192`	256-dim subcode (see spec) of $\sum_{0 \le i < 1024} \{0, 126\} x^i$
`lac`	`256`	256-dim subcode (see spec) of $\sum_{0 \le i < 1024} \{0, 126\} x^i$
`newhope`	`512`	$\sum_{0 \le i < 256} \{0, 6145\} x^i (1 + x^{256})$
`newhope`	`1024`	$\sum_{0 \le i < 256} \{0, 6145\} x^i (1 + x^{256} + x^{512} + x^{768})$
`ntru`	`hps2048509`	not applicable
`ntru`	`hps2048677`	not applicable
`ntru`	`hps4096821`	not applicable
`ntru`	`hrss701`	not applicable
`ntrulpr`	`653`	$\sum_{0 \le i < 256} \{0, 2310\} x^i$
`ntrulpr`	`761`	$\sum_{0 \le i < 256} \{0, 2295\} x^i$
`ntrulpr`	`857`	$\sum_{0 \le i < 256} \{0, 2583\} x^i$
`round5n1`	`1`	8×8 matrix over $\{0, 1024, 2048, 3072\}$
`round5n1`	`3`	8×8 matrix over $\{0, 4096, \dots, 28672\}$
`round5n1`	`5`	8×8 matrix over $\{0, 2048, \dots, 30720\}$
`round5nd`	`1.0d`	$\sum_{0 \le i < 128} \{0, 4096\} x^i$
`round5nd`	`3.0d`	$\sum_{0 \le i < 192} \{0, 2048\} x^i$
`round5nd`	`5.0d`	$\sum_{0 \le i < 256} \{0, 4096\} x^i$
`round5nd`	`1.5d`	128-dim subcode (see spec) of $\sum_{0 \le i < 318} \{0, 512\} x^i$
`round5nd`	`3.5d`	192-dim subcode (see spec) of $\sum_{0 \le i < 410} \{0, 2048\} x^i$
`round5nd`	`5.5d`	256-dim subcode (see spec) of $\sum_{0 \le i < 490} \{0, 1024\} x^i$
`saber`	`light`	$\sum_{0 \le i < 256} \{0, 4096\} x^i$
`saber`	`main`	$\sum_{0 \le i < 256} \{0, 4096\} x^i$
`saber`	`fire`	$\sum_{0 \le i < 256} \{0, 4096\} x^i$
`sntrup`	`653`	not applicable
`sntrup`	`761`	not applicable
`sntrup`	`857`	not applicable
`threebears`	`baby`	256-dim subcode (see spec) of $\sum_{0 \le i < 274} \{0, 512\} 2^{10i}$
`threebears`	`mama`	256-dim subcode (see spec) of $\sum_{0 \le i < 274} \{0, 512\} 2^{10i}$
`threebears`	`papa`	256-dim subcode (see spec) of $\sum_{0 \le i < 274} \{0, 512\} 2^{10i}$

Table 8.10. Set of encoded messages for each of the target Product NTRU PKEs. Error-correcting codes are as follows: `lac`: BCH codes (see specification for details); `newhope`: repetition codes (shown above); `round5nd.5d`: “XE” codes (see specification for details); `threebears`: Melas codes (see specification for details).