Cryptology in the Crowd

Hans Heum

Thesis for the degree of Philosophiae Doctor (PhD)
University of Bergen, Norway
2023

UNIVERSITY OF BERGEN

Cryptology in the Crowd

Hans Heum

Thesis for the degree of Philosophiae Doctor (PhD)
at the University of Bergen

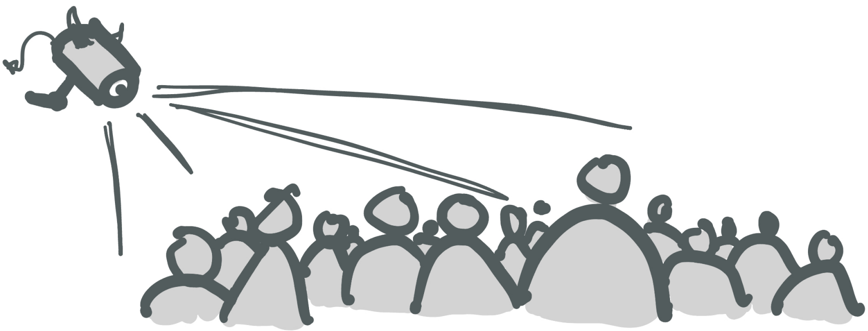
Date of defense: 08.12.2023

Year: 2023

Title: Cryptology in the Crowd

Name: Hans Heum

Print: Skipnes Kommunikasjon / University of Bergen



Abstract

Accidents happen: you may misplace the key to your home, or maybe the PIN to your home security system was written on an ill-placed post-it note. And so they end up in the hands of a bad actor, who is then granted the power to wreak all kinds of havoc in your life: the security of your home grants no guarantees when keys are stolen and PINs are leaked. Nonetheless your neighbour, whose key-and-pin routines leave comparatively little to be desired, should feel safe that just because you can't keep your house safe from intruders, their home remains secured.

It is likewise with cryptography, whose security also relies on the secrecy of key material: intuitively, the ability to recover the secret keys of other users should not help an adversary break into an uncompromised system. Yet formalizing this intuition has turned out tricky, with several competing notions of security of varying strength. This begs the question: when modelling a real-world scenario with many users, some of which may be compromised, which formalization is the right one? Or: how do we build cryptology in a crowd?

Paper I embarks on the quest to answer the above questions by studying how various notions of multi-user IND-CCA compare to each other, with and without the ability to adaptively compromise users. We partly answer the question by showing that, without compromise, some notions of security really are preferable over others. Still, the situation is left largely open when compromise is accounted for.

Paper II takes a detour to a related set of security notions in which, rather than attacking a single user, an adversary seeks to break the security of many. One imagines an unusually powerful adversary, for example a state-sponsored actor, for whom brute-forcing a single system is not a problem. Our goal then shifts from securing every user to making mass surveillance as difficult as possible, so that the vast majority of uncompromised users can remain secure.

Paper III picks up where Paper I left off by comparing and systemizing the same security notions with a wider array of security notions that aim to capture the same (or similar)

scenarios. These notions appear under the names of Selective Opening Attacks (SOA) and Non-Committing Encryption (NCE), and are typically significantly stronger than the notions of IND-CCA studied in Paper I. With a system in place, we identify and highlight a number of gaps, some of which we close, and many of which are posed as open problems.

Sammendrag

Uhell skjer: Kanskje mistet du nøkkelen til huset, eller hadde PIN-koden til innbruddsalarmen skrevet på en dårlig plassert post-it lapp. Og kanskje endte de slik opp i hendene på feil person, som nå kan påføre livet ditt all slags ugagn: Sikkerhetssystemer gir ingen garantier når nøkler blir stjålet og PIN-koder lekket. Likevel burde naboen din, hvis nøkkel-og-PIN-kode rutiner er heller vanntette, kunne føle seg trygg i vissheten om at selv om *du* ikke evner å sikre huset ditt mot innbrudd, så forblir deres hjem trygt.

Det er tilsvarende for kryptologi, som også lener seg på at nøkkelmateriale hemmeligholdes for å kunne garantere sikkerhet: Intuitivt forventer man at kjennskap til ett systems hemmelige nøkkel ikke burde være til hjelp for å bryte inn i andre, urelaterte systemer. Men det har vist seg overraskende vanskelig å sette denne intuisjonen på formell grunn, og flere konkurrerende sikkerhetsmodeller av varierende styrke har oppstått. Det blir dermed naturlig å spørre seg: Hvilken formalisme er den riktige når man skal modellere realistiske scenarioer med mange brukere og mulige lekkasjer? Eller: hvordan bygger man kryptografi i en folkemengde?

Artikkel I begir seg ut på reisen mot et svar ved å sammenligne forskjellige flerbrukervarianter av sikkerhetsmodellen IND-CCA, med og uten evnen til å motta hemmelige nøkler tilhørende andre brukere. Vi finner et delvis svar ved å vise at uten denne evnen, så *er* noen modeller faktisk å foretrekke over andre. Med denne evnen, derimot, forblir situasjonen uavklart.

Artikkel II tar et sidesteg til et sett relaterte sikkerhetsmodeller hvor, heller enn å angripe én enkelt bruker (ut fra en mengde av mulige ofre), angriperen ønsker å bryte kryptografien til så mange brukere som mulig på én gang. Man ser for seg en uvanlig mektig motstander, for eksempel en statssponset aktør, som ikke har problemer med å bryte kryptografien til en enkelt bruker: Målet skifter dermed fra å garantere trygghet for alle brukerne, til å gjøre masseovervåking så vanskelig som mulig, slik at det store flertall av brukere kan forbli sikret.

Artikkel III fortsetter der Artikkel I slapp ved å sammenligne og systematisere de samme

IND-CCA sikkerhetsmodellene med en større mengde med sikkerhetsmodeller, med det til felles at de alle modellerer det samme (eller lignende) scenarioet. Disse modellene, som går under navnene SOA (Selective Opening Attacks; utvalgte åpningsangrep) og NCE (Non-Committing Encryption; ikke-bindende kryptering), er ofte vesentlig sterkere enn modellene studert i Artikkel I. Med et system på plass er vi i stand til å identifisere en rekke hull i litteraturen; og dog vi tetter noen, etterlater vi mange som åpne problemer.

Acknowledgements

I would like to extend my thanks to Håvard and Kjell-Jørgen for taking a chance on a physics student that knew nothing about cryptology going in, and to the leadership and administration of Simula UiB for never failing to have my back. Thank you Mari for all of the headaches you've resolved, which reaches far beyond just maintaining my caffeine addiction. And a special thank you to Åsfrid for agreeing to join me in this crazy project. These four years would have been very different without you.

I would like to thank my supervisor, Martijn, for being as excellent a human being as he is researcher, who always strives to lift those around him to likewise excellence: to the degree that I can be called a cryptographer today, I owe it all to you. Thank you Martha, for enthusiastically taking up any proofreading I have tossed at you, and for being exactly the fellow language nerd that I need in my life. And thank you Carlo for being the perfect ball-tosser, and for the countless bugs, typos, and cognitive faults that you've uncovered. Without your many ideas and constant vigilance (and refusal to succumb to my arrogant insistence that I was right when I, in fact, was not), I'm sure I'd still be editing this thesis a year from now.

Thank you Mithilesh for igniting my interest in quantum computing, and for showing me how easily (given the right teacher) it can be grasped. As anyone who has joined me for a beer in the last four years can attest, this interest has since blossomed into a full-blown passion. Thank you Morten for entertaining it in countless long discussions, and Øyvind and Carlos for giving me the opportunity to convey it to a wider audience at Arendalsuka 2022. Should my hopes come true, this is only the start.

Thank you to IMF at NTNU for hosting me in the final legs of this journey, and to Tjerand and Kristian for making it possible.

Finally, thank you to my parents for your constant encouragement and support. I would not have made it to where I am today without you. A special thanks is in order for my dad, who was the one to suggest this whole venture to begin with. I am glad I (finally) listened. Takk!

Contents

1 Introduction: A Brief History of Concrete Security **11**

1.1 Kerckhoffs's Principles . . . 12

1.2 Diffie and Hellman's New Directions . . . 15

1.3 Goldwasser and Micali's Mental Poker . . . 21

1.4 Bellare and Rogaway get Practice Oriented . . . 26

1.5 Looking Ahead . . . 37

2 Preliminaries: Modelling the Real World **43**

2.1 Multi-Challenge . . . 44

2.2 Multi-User . . . 46

2.3 Corruptions Empower . . . 48

2.4 Multi-Target . . . 50

3 Our Contribution **53**

Article I Tightness Subtleties for Multi-user PKE Notions **63**

Article II Multi-Instance Secure Public-Key Encryption **87**

Article III SoK: Public Key Encryption with Openings **121**

Chapter 1

Introduction: A Brief History of Concrete Security

We stand today on the brink of a revolution in cryptography.

Thus opens "New Directions in Cryptography" [22], the seminal 1976 paper by Whitfield Diffie and Martin Hellman. Their work, which introduced the world to the magic of asymmetric cryptography, indeed brought on a paradigm shift in the science of secret writing known as cryptography.

In this introduction I shall describe three such paradigm shifts, each of which is largely due to a legendary cryptologist duo: the shift from symmetric to asymmetric encryption, as pioneered by Diffie and Hellman (Sect. 1.2); the shift from ad-hoc to provable security, as developed by Shafi Goldwasser and Silvio Micali (Sect. 1.3); and the shift from existential to practice-oriented security proofs, as initiated by Mihir Bellare and Phillip Rogaway (Sect. 1.4). Along the way we will see events leading up to the shifts, and their impact on cryptographic research, as well as on society at large. And, ultimately, we will see how the stage was set for us to consider a number of questions (Sect. 2), some of which are tackled in the three research articles that make up this thesis (Sect. 3). Finally, I will briefly muse on what lies ahead for the field of concrete security in Sect. 1.5.

But we are getting ahead of ourselves. Let us start by looking at the paradigm of cryptographic research that was in place *before* the year 1976.

1.1 Kerckhoffs's Principles

The history of cryptography is long and complex, and like most histories of human affairs, it is filled with interesting characters and fascinating intrigues ("The Code Book" by Simon Singh [75] provides an excellent overview). It holds a unique position in computer science in that it stretches back to ancient times, with the earliest hypothesized appearance found in the tomb of Egyptian nobleman Khnumhotep II [48, 55]. Hailing from the second millennium BCE, these inscriptions employ uncommon hieroglyphic symbols and grammatical inconsistencies to obscure the meaning of the text. While their precise purpose is unknown, the ease with which they are decoded suggests they may have been meant as an amusing puzzle for the literate visitor.

The earliest known use of cryptography for the purpose of secret messaging is the Caesar cipher [75]. This simple cipher worked by shifting the letters of the message by three positions, wrapping around at the end. An enemy intercepting the message weattackatdawn would thus see ZHDWWDFNDWGDZQ, hopefully discarding it as nonsense.

Crucially, the Caesar cipher's security relied on the method of encryption—or that encryption was being employed at all—being kept secret from the interceptor. Anyone with knowledge of the encryption method could easily decode any ciphertext, and Caesar would have ensured that any intended recipient had been informed in secret of the encryption method beforehand. If an interceptor even suspected that an alphabet shift had taken place, he would only have to try a small number of possible shifts before uncovering the message.

Keeping cryptography a hidden art eventually turned infeasible, marking a need for stronger ciphers. *Substitution ciphers* became commonplace, in which each letter is exchanged for a different one, without the requirement that the shifts are alphabetically related. However, memorizing a full substitution alphabet is costly. Instead, keywords or keyphrases were often employed. These allowed for both ease of memorization and ease of transmission, as for instance the first phrase of an otherwise innocuous-looking book might serve as a decryption key.

Keyphrase ciphers function as follows: assume the keyphrase is "Once upon a time". Then, the letters a through j are to be replaced with O, N, C, E, U, P, A, T, I, and M, respectively. The remaining letters are to be substituted alphabetically as usual, beginning where the keyphrase ended with k being replaced by Q (since N, O, and P have already been assigned), skipping any letter that has already been assigned a substitution and wrapping around at the end of the alphabet (see Table 1.1).

Table 1.1: A substitution cipher based on the keyphrase "Once upon a time".

a	b	c	d	e	f	g	h	i	j	k	l	m
O	N	C	E	U	P	A	T	I	M	Q	R	S
n	o	p	q	r	s	t	u	v	w	x	y	z
V	W	X	Y	Z	B	D	F	G	H	J	K	L

Clearly, the security of the cipher now depends on the number of different letters that appear in the key phrase. In the extreme case that the keyword is a single letter, it reduces to a shift cipher: let the keyword be "D", then a is to be replaced by D, b by E, c by F and so on, and we have recovered Caesar's three-shift cipher. At the other extreme are keyphrases that substitute all 26 letters. An example: "Sphinx of Black Quartz, Judge My Vow!"

There are $26! \approx 2^{88} \approx 10^{26}$ possible substitution ciphers. Observing this, one might come to the belief that a would-be attacker would have to cycle through 26! possible substitutions before coming to the correct one, an insurmountable task for any human. But even a randomly chosen, complete substitution is easily broken with a large enough sample of encrypted text through the use of frequency analysis.

Language is not uniform: the letter "e" for instance appears roughly 13% of the time in English text, making it by far the most common letter of the language. Substitution ciphers do not mask the distribution of letters, and once a small number of them have been deduced by looking at their relative frequencies, matching the remaining letters essentially reduces to a few educated guesses (such as "e" often being preceded by "th").

Counting arguments such as the above were a common fallacy among mathematicians attempting to prove the security of their encryption schemes in previous centuries. Diffie and Hellman write [22]:

> *During the sixteenth and seventeenth centuries, mathematical arguments were often invoked to argue the strength of cryptographic methods, usually relying on counting methods which showed the astronomical number [of] possible keys. Though the problem is far too difficult to be laid to rest by such simple methods, even the noted alraist [*sic*] Cardano fell into this trap [48, p. 145]. As systems whose strength had been so argued were repeatedly broken, the notion of giving mathematical proofs for the security of systems fell into disrepute and was replaced by certification via cryptanalytic assault.*

This requirement, that the trust in an encryption method is to come from failed attempts to break it rather than any attempt at proving it unbreakable, is summarized in the first of Auguste Kerckhoffs's six principles for military ciphers, published in 1883 (translated from French) [50]:

> *The system must be practically, if not mathematically, indecipherable.*

Another common fallacy that has plagued the employment of encryption systems is that keeping an encryption method hidden increases the security of the cipher. It is easy to see how anyone would fall victim to this trap, for surely, inverting any function must require knowledge of the function to be inverted. However, as evidenced time and again by countless broken ciphers, all that is needed is an ability to recognize a correct invertion, or in other words, a successful break.

To the contrary, any good cipher should be secure whether or not its encryption method is known, as long as the *key* used remains secret. This is codified in Kerckhoffs's second principle, otherwise known as simply "Kerckhoffs's principle" [50]:

> *[The system] should not require secrecy, and it should not be a problem if it falls into enemy hands.*

Modern cryptographers interpret Kerckhoffs's principle in the extreme, advocating that all cryptographic algorithms should be public and implementations open-sourced. After all, given that security of a cipher is to be judged by cryptanalytic attacks against it, having the encryption method be public knowledge and inviting any interested party to try and crack it can only help put confidence in its security—unless the cipher was in fact insecure to begin with.

Thus the paradigm is set: encryption methods should remain secure even if released to the public, their security should rely on the secrecy of the encryption key *only*, and trust in their security should stem from continuous failed efforts to break it.

Kerckhoffs's remaining principles are mostly irrelevant for cryptography today. However, his third principle is interesting in that it foreshadows a challenge that would come to a point with the development of increasingly efficient communication technology, leading to the first major paradigm shift almost a century later [50]:

> *It must be possible to communicate and remember the key without using written notes, and correspondents must be able to change or modify it at will.*

1.2 Diffie and Hellman's New Directions

While Kerckhoffs's third principle talks of the ease with which a system should be able to be reinstated with a fresh key, changing a key necessarily involves the correspondents agreeing on a new one. Bar an alternative, presumed-secure channel (for instance a human courier), this would only have been possible by having the correspondents meet in person.

Imagine the correspondents are located on different continents, and their key gets compromised; agreeing on a new one would then require a comparatively gigantic effort, involving careful (non-secret) communication of a time and place, and long-distance travel to get there, potentially hindering secure communications for months in the meantime. They would certainly not be able to replace the key "at will".

A secret discovery. In the late 1960s, within the top-secret walls of the UK Government Communications Headquarters (GCHQ), James Ellis had been put to the task of finding some way of having two parties communicate securely without the need for a pre-shared secret [25]. The conventional wisdom was that this was impossible; in fact, it seemed self-evident to most people, including Ellis, that no security could be achieved without a pre-shared secret.[1] Then he came across a classified WWII report on a technique developed by Bell Labs that achieved exactly this, securing analogue voice transmissions without the need for a shared secret [24].

A bit of background: during the war, Bell Labs had developed a method for secure speech that involved distributing pairs of records of random noise between sender and recipient. A sender would then add this noise to the line, so that anyone intercepting it would hear only static. At the receiver end, the noise would be subtracted, using their copy of the record. The records would then be destroyed, security requiring that they were each used once only. The technique can thus be viewed as an analogue equivalent of the one-time pad (see Sect. 1.3).

In the final report of the related "Project C43", the author muses that a different system might be possible, one in which the noise is added to the line by the *receiver*, rather than the sender. Then, there would be no need to distribute pairs of records—the receiver, having full knowledge of the noise that was added, could simply remove it himself. While intriguing, the technique seems to have never been implemented, as it was too impractical for long-distance radio transmission, the purpose of Project C43. The author writes:

[1] "I think he was sort of sidetracked", a colleague would later say about Ellis being put to what was believed to be an impossible task [53].

> *This system, however, cannot be used for radio at all because the level of the noise decreases with distance from the receiving station, while the level of the signal increases. The interceptor, therefore, will get good speech signals if he is close to the transmitter. With telephone lines this differential can be kept small.*

Reading this prompted Ellis to ask himself whether the same thing could be achieved digitally, and late one night as he was lying in bed, he realized that he could prove that it was possible in principle [25]. His proof, involving computers modelled as large look-up tables, did not give any hint towards whether such a thing could be achieved *practically*. But it did rule out any sort of impossibility theorem, which most of his colleagues had at the time simply assumed must exist.[2]

Ellis's main insight, which he had got from Project C43, was that to achieve "non-secret encryption" (as he called it), the recipient would have to take an active part in the encryption process. His scheme had the recipient send an "encrypted key", what we today would call a public key, to the sender over an open channel. The sender would then encrypt as usual using this encrypted key, and send it back. Crucially, decryption could only happen with access to the "unencrypted key", today known as the private key, by assumption known to the receiver only.

Being careful to note that he only showed that such a system was "theoretically plausible", particularly given that he had "established something, which, to most people, seems inherently impossible", he nevertheless writes [24]:

> *Attempts to find solutions [by looking for suitable hard-to-invert functions] have so far been unsuccessful but have seemed tantalisingly near to success at times, so that one feels that a solution probably does exist. There is certainly no apparent reason to believe it impossible.*

Ellis published his internal report in January of 1970, and appears to have stopped working on the problem thereafter, feeling that his engineering background was insufficient to solve what was clearly to him a number-theoretical problem. His report, "The Possibility of Secure Non-secret Digital Encryption", concludes [24]:

> *In any case it is hoped that this report will stimulate those proficient in these*

[2] Upon receiving Ellis' manuscript, GCHQ's chief mathematician set about finding the flaw that *obviously* had to exist in the proof. Six months later he conceded: "Unfortunately, I can't see anything wrong with this." [53]

> *matters to find a practical solution. The potential advantages are too obvious to need specification.*

Three years later, a young recruit was introduced to Ellis' idea by his mentor. This was Clifford Cocks, and at age 22 he had only arrived at the GCHQ's electronics-security group three months earlier. Finding the problem intriguing, he found himself musing over it later that same evening. Due to the strict rules of the GCHQ, he was not allowed to write anything down when working out-of-office. "Happily", he recalls, "the first idea seemed to work just fine." [53]

Having previously worked in number theory, it was natural for him to consider the problem of factoring large primes as the underlying irreversible function for the encryption scheme. He wrote down his idea the following day and submitted it for publishing internally.

This caught the attention of Malcolm Williamson, a childhood friend of Cocks, surprised to learn that Cocks had an internal paper coming out this soon after his recruitment. Williamson, who still believed that secret communication was impossible without pre-sharing some secret, set out to find a flaw in the concept. He found that he couldn't. Instead, he not only became convinced of the correctness of Cocks's scheme, but discovered a different one, this time relying on the hardness of computing discrete logarithms. Through a minimal amount of open communication, two parties could, using his scheme, agree on a shared secret key.

Nevertheless, in part due to the impracticality of the schemes at the time,[3] and in part due to the conservative nature of government secrecy, the schemes didn't see any field employment. Particularly the fact that security seemed to reduce to a single simple-to-state mathematical problem, rather than the jumbled mess of typical contemporary encryption schemes, made several members of the GCHQ nervous about putting too much trust in the schemes. "Some graduate-student mathematician could really cause a disaster," Williamson said, summarizing the prevailing view at the time [53].

And so their discoveries remained secret, only getting declassified in 1997. Three years after their discoveries, Ellis' concept of non-secret encryption would be rediscovered by a young hacker and a Stanford professor, simultaneously rediscovering Williamson's key establishment scheme. This scheme is today known as Diffie–Hellman key exchange, after its inventors. A year later, three other MIT researchers would, after much trial-and-error, discover a variant of the scheme that Cocks had cooked up in an evening.

[3] Cocks says, "It took minutes to generate a key. We looked at the circumstances under which you would find it useful to have a machine that took that long to produce keys and immediately thought the applications were too limited to make it worth floating." [53]

They were Ron Rivest, Adi Shamir, and Leonard Adleman, and that scheme is known today as RSA.

A public revolution. In 1974, as part of an undergraduate class project, a young Ralph Merkle devised a scheme with which a secret key could be established between two parties without the need for pre-shared secrets. He proved that an eavesdropper would need to spend more computational effort to uncover the secret than it took for either party to establish it (roughly n^2 computational steps, versus roughly n steps for each of the honest parties). In a paper published in 1978[4] explaining the protocol, Merkle writes [56]:

> *[This work] develops a new paradigm, which differs significantly from the traditional one. (. . .) Assume two forces, Us and Them, are fighting. They are winning, because they have broken our codes and ciphers. We only find out about this when we discover that they attack exactly where we are weakest, retreat just before our attacks, and generally seem to know too much too quickly. Our forces are in the field, fully deployed, with no chance of distributing new keys in accordance with the traditional paradigm. Under the traditional paradigm, we are lost. Using the new paradigm, we can easily change all our keys, and re-establish security. The difference is dramatic.*

His method relies on the concept of a “puzzle”, a weakened (symmetric) cipher that is assumed to take effort n to break. It works as follows.

Let us say Alice and Bob want to establish a key over an open channel, while keeping it secret from the eavesdropper Eve. Alice then spends n effort to produce n puzzles, each containing a candidate secret key and a randomly chosen key ID. She sends all n puzzles to Bob, who chooses one of them at random. He spends the required n effort to solve the puzzle (i.e., crack the cipher), uncovering the key and the corresponding ID, and then sends the ID back to Alice. Alice compares the ID with her own list of keys and key IDs, and identifies the key that Bob must have opened. They now have a shared key.

Eve, meanwhile, listening in on their conversation, stores her own copy of the n puzzles. She sees the ID sent by Bob, but this gives her little information about which of the

[4] It appears that the paper was first submitted for publication some time before Diffie and Hellman’s landmark paper appeared, but only actually published two years later. Yet Diffie and Hellmann were aware of Merkle’s work: Martin Hellman, in a 2004 interview, says, “If you’re going to put names on it, it should be called Diffie–Hellman–Merkle key exchange, since it’s actually based on a concept of Merkle’s. We give him credit for that in the paper, but it was in a paper by Diffie and Hellman, so it’s called Diffie–Hellman key exchange.” [79]

puzzles to attack to uncover the corresponding key. Intuitively, the best she can do is to open one at a time until the transmitted ID is uncovered. This is expected to happen after opening $n/2$ of them on average. As each puzzle is assumed to take n steps to open, this leads to the expectation that Eve will have to spend $n^2/2$ effort to recover the secret key.

This scheme is not at all practical, however: in order for the computational gap between Eve's effort and that of Alice and Bob to bring any security guarantees, n would have to be set astronomically large. This would not only lead to prohibitive computation times, but also large communication costs, as all n puzzles have to be transmitted from Alice to Bob. Merkle writes [56]:

> *The amount of work required of [Eve] to determine the key will increase as the square of the amount of work put in by [Alice] and [Bob] to select the key. Clearly, it would be desirable to find a solution in which the amount of work put in by [Eve] increases exponentially as a function of the amount of work put in by [Alice] and [Bob]. We see no reason why such exponential methods should not exist.*

Of course, Whitfield Diffie and Martin Hellman soon after discovered a much more practical approach [22]. But their paper, aptly titled "New Directions in Cryptography", contains so much more; while unaware that similar discoveries had been made in secret before them, it is fair to say that Diffie and Hellman's treatise on the possibility of "public-key cryptography" (PKC) was by far the most complete to date.[5] Like Ellis before them, they realized that by splitting the key up in a *public* and a *private* key, one could devise a public key encryption (PKE) system that would allow Alice and Bob to communicate privately, with no need to meet up beforehand to exchange secrets. Unlike Ellis, though, they realized that such a system was actually fit to solve more than just one problem.

Diffie had for some time been fascinated with the idea of a "digital office" [21]. One of the central tools for any office is the ability to authenticate documents by signing them. But with the world of commerce going digital, what would "digital signatures" look like? Well, once Diffie and Hellman had devised the framework of public and private keys, they realized that they could solve the digital signing problem by flipping the encryption

[5] Apart from the primitives discussed in the text, the list of concepts appearing in their paper, some for the first time, include one-way functions, password hashing, chosen plaintext attacks, trapdoor one-way functions, the possibility of built-in hidden backdoors, program obfuscation, lazily sampled random functions, the suggestion to build cryptosystems from NP-complete problems, and the possibility of proving schemes secure through reductions. We will meet many of these concepts again later.

protocol on its head: instead of using the public key to encrypt a message, Alice could now *sign* a message by “encrypting” it with her private key, and send this “ciphertext”, which now acts as a signature, to Bob along with the document to be signed. This would allow Bob, as well as anyone else with access to her public key, to “decrypt” the signature and verify that the resulting message matches the message to be signed. Since only Alice could have produced such a signature, the document is authenticated. What’s more, Alice cannot deny that it was she who signed the message, making the signature potentially valid also in a court of law.

Of course, a PKE scheme would just as easily yield a key distribution scheme, as defined by Merkle, simply by letting the plaintext be the desired shared key. However, the reverse does not hold, and so, while they were able to present their famous method of Diffie–Hellman key exchange, they left the construction of a full public key encryption system as an intriguing open problem.

Ron Rivest and Adi Shamir took up the challenge, designing scheme after scheme that they hoped would satisfy the properties of a PKE. They sent each one over to Leonard Adleman, who in turn broke every one—until one day they sent him one he could not break [75]. They had discovered a variant of Cocks’s scheme from four years earlier, and it seemed to be unbreakable unless an efficient method of factoring large numbers were to be found. This is the scheme that we have come to know as RSA encryption [65]. Published only a year after Diffie and Hellman suggested that such a scheme might be possible, it took the world by storm, in no small part thanks to an early exposition in Martin Gardner’s legendary “Mathematical Games” column of Scientific American [30].

For the next several decades, RSA helped transform digital communication, enabling among other things practical e-commerce through encrypted traffic and digital signatures, and bringing secure email to the masses through Phil Zimmerman’s “Pretty Good Privacy” (PGP) software [31]. The latter program used RSA encryption to encrypt a key, to be used together with a conventional, symmetric cipher. This method, today known as “hybrid encryption” [72], provides the best of both worlds of asymmetric and symmetric ciphers: given that the symmetric key, encrypted using RSA, will typically be relatively short compared to the message to be encrypted, this ensures that the less efficient asymmetric algorithm does not have to do too much of the heavy lifting, as the much more efficient symmetric cipher will be used to encrypt the bulk of the message.

1.3 Goldwasser and Micali's Mental Poker

Rivest versus Rabin. Rivest, Shamir, and Adleman relied on the conjectured difficulty of factoring to argue that their scheme was secure, writing [65]:

> *While factoring large numbers is not provably difficult, it is a well-known problem that has been worked on for the last three hundred years by many famous mathematicians. (...) However, no one has yet found an algorithm which can factor a 200-digit number in a reasonable amount of time. We conclude that our system has already been partially "certified" by these previous efforts to find efficient factoring algorithms.*

However, while they could not come up with a faster way of reversing the RSA function than factoring, *proving* the equivalence was beyond them [65]:

> *It may be possible to prove that any general method of breaking our scheme yields an efficient factoring algorithm. This would establish that any way of breaking our scheme must be as difficult as factoring. We have not been able to prove this conjecture, however.*

To prove it, they would have to provide a *reduction*, showing that anyone with an algorithm to reverse the RSA function would get an algorithm for factoring large composite numbers for free. Such a reduction has yet to be found.

This is however exactly what Michael Rabin did for his own scheme two years later, when he showed that by modifying the RSA function slightly, a simple reduction from breaking the security of the scheme to factoring appears [63]. In other words, he *proved* that anyone who can break his scheme could by necessity also factor large numbers. Given that factoring large numbers is assumed to be hard, his scheme must therefore be assumed secure.

And yet, it did not take long for Ron Rivest to present a simple but effective attack against Rabin's altered scheme [51]. Ironically, the attack follows directly from the security proof, as the attacker would use exactly the same technique as the reduction to recover the factorization of the public key. All Eve would need to do is trick Alice into decrypting a few ciphertexts with her private key and sharing the result.

Maybe Alice used to be a close collaborator of Eve, and sees no reason to distrust Eve's strange requests. She happily complies, safe in the knowledge that her cryptosystem has anyway been proven secure. One can only imagine her confusion as she wakes up the next morning to the havoc that Eve, now with full access to Alice's private key, has brought upon her.

Now would be a good time to take a step back and ask what is going on. Rabin *proved* that reversing the encryption of his scheme is as hard as factoring the public key. And yet Eve was able to mount a simple attack to recover Alice's key. The answer, of course, lies in the assumptions underlying the proof. In other words, what *model* of security was the scheme proven to achieve?

First off, note that any public key system allows anyone to create as many plaintext–ciphertext pairs as they want: simply encrypt any plaintext using the target's public key. This is known as a Chosen-Plaintext Attack (CPA). Rabin showed that reversing the one-way function underlying encryption was as hard as factoring, even given the adversary's ability to mount such chosen plaintext attacks. Using OW for "One-Wayness", and using the standard nomenclature of [security goal]-[attack power], we may conclude that Rabin proved the scheme OW-CPA secure under the factoring assumption.

However, Eve did something more in her attack on Alice. Namely, she tricked Alice into using her secret key to decrypt ciphertexts of Eve's choosing, and somehow this gave Eve enough information to bypass Rabin's proof and recover the key. In other words, Rabin's security model was not general enough to cover Eve's attacks.

Eve's attack is known as a Chosen-Ciphertext Attack, or CCA. So, the attack shows that Rabin's scheme is not OW-CCA secure. In fact, it shows something stronger: the scheme is not even KU-CCA secure. Here we have weakened the security goal from "it should be hard to recover the plaintext" (One-Wayness) to "it should be hard to recover the private key" (Key-Unrecoverability). Why is this a weakening? Right: because someone who can recover the private key can also use it to recover plaintexts, but not necessarily vice versa. Eve's attack breaking a weaker security assumption means a stronger result.

It is at this point worth noting that "plain" RSA encryption is *also* broken in the CCA setting: there exists a simple chosen ciphertext attack with which an adversary can recover the plaintext from any ciphertext it desires [51]. So RSA is not OW-CCA-secure. However, the attack does not recover private keys, and for all we know it might therefore be KU-CCA-secure. Can we therefore say that RSA is *more* secure against chosen ciphertext attacks than the Rabin scheme?

Well, from a modern perspective, we would simply say that both schemes are broken in the CCA setting, as neither one-wayness nor key unrecoverability are considered particularly lofty security goals. In fact, by strengthening our security goals, we would say that both are broken in the CPA setting too. We will see why next.

Probabilistic encryption. In the early eighties, Shafi Goldwasser and Silvio Micali, two PhD students who had met each other at their graduate program at the University of California, Berkeley, were pondering how one would go about playing "mental poker", i.e. a remote game of poker played over a telephone line. Given that neither player trusts the other (and in fact expects the other to cheat), the protocol would need to somehow guarantee fairness, and ideally allow the players to verify later on that the game was played correctly. Rivest, Shamir, and Adleman had introduced the problem, and given a partial solution [69]. While their scheme looked sound on the surface, their suggested implementation, employing RSA encryption, was only secure against the recovery of a full card, i.e. its value and suit[6]. It was clear to Goldwasser and Micali that if Bob were to learn any partial information about one of Alice's cards—say, its suit—he would gain an unfair advantage, unacceptable in any game of poker claiming to be fair.

Defining "partial information" to be any efficiently computable predicate on the plaintext (i.e. any function taking the plaintext as input and outputting a single bit, for instance a yes/no answer to the question "is the colour of the card red?"), their first realization was that no deterministic encryption scheme (i.e. involving no randomness aside from the choice of keys) can hide all partial information [35]. They gave a trivial example: let the predicate $P(m) = 1$ if and only if encrypting m under a given public key pk yields a ciphertext that is even. Clearly, anyone with access to pk can calculate this predicate, and so the ciphertext leaks partial information on the plaintext. If the encryption process were instead *probabilistic*, so that re-encrypting messages would yield random-looking results, then P would no longer be a well-defined predicate on the message.

Of course, while probabilistic encryption is a *necessary* requirement for the hiding of all partial information, it is far from sufficient. Goldwasser and Micali therefore went one step further, and constructed a scheme for which they could prove the following claim [35]:

> *An adversary, who knows the encryption algorithm and is given the cyphertext* [sic]*, cannot obtain any information about the cleartext.*

In more detail, they showed that if an adversary, given a ciphertext, could learn the value of *any* predicate on the plaintext with a significantly higher likelihood than by random guessing, then they could also reverse the underlying *trapdoor predicate* serving as their

[6] Their write-up is bereft with statements like "Alice has no way of knowing anything about Bob's hand since the encryption key B is known only to Bob", which in hindsight is clearly a fallacy [69].

hardness assumption. They then gave an implementation based on a trapdoor predicate known as the quadratic residuosity problem, a well-known problem in number theory for which no efficient solution is known.[7]

While their scheme is highly unpractical, producing ciphertexts hundreds of times larger than the plaintexts, it served as a useful proof-of-concept: it showed that it is possible to come up with very strong notions of security, and build systems that provably achieve them. And thus, they had finally showed how to play a truly fair game of mental poker.

But the impact of their work reached far beyond the realm of card games.

What is meant by "security". In the mid-forties, a young Claude Shannon had proven that the one-time pad, which had been in use for military encryption since the late 1800s, achieves *perfect secrecy* [70]. The one-time pad is a deceptively simple scheme: given a message encoded as a bit string, produce an equally long, randomly sampled bit string. This will be your key. Then produce the ciphertext by taking the xor of each bit: if the first bit of the key is 1, then you flip the first bit of the message; if the second bit of the key is 0, then you leave the second bit of the message alone; and so on. What Shannon showed is that the resulting ciphertext is *unbreakable*, no matter how much computational effort you throw at it: you could be calculating until the end of the universe and be no closer to uncovering the plaintext.

There are two important caveats. First, the key must only be used once: if an adversary is given two ciphertexts using the same key, it can already recover non-trivial information about the plaintexts. Second, the key must be at least as long as the plaintext. Taken together, this makes for cumbersome key management, and there are many stories of cold-war era agents walking around with briefcases handcuffed to their wrists filled with stacks of paper containing nothing but random bits [75].

Shannon defined perfect secrecy to mean that any information about the plaintext given the ciphertext, would be equally available to the attacker without the ciphertext. In other words, having access to the ciphertext should not help at all in guessing the message.

Goldwasser and Micali's original notion of security, based on the hardness of guessing predicates on the plaintext, was quite unwieldy to work with. When they later expanded on their work, they provided two alternative notions of security. One, which they called

[7] They furthermore showed that if the quadratic residuosity problem is hard to solve for *any* instance of the problem, then it is also hard *on average*. Such worst-case-to-average-case reductions are extremely valuable in cryptography, as there are many problems for which no efficient general algorithm is known, and yet most instances are easily solved, making them completely unsuitable as cryptographic primitives. A worst-case-to-average-case reduction rules out this possibility.

semantic security, was essentially a polynomially bounded version of Shannon's perfect secrecy [35]:

> *Informally, a system is semantically secure if whatever an eavesdropper can compute about the cleartext given the cyphertext, he can also compute without the cyphertext.*

If the attacker is restricted to chosen plaintext attacks, we call the notion SEM-CPA.

Their other notion of security says that, for any choice of messages m_0 and m_1, the adversary should be unable to guess which of the two has been encrypted in a given ciphertext with probability significantly better than $1/2$. This notion, which is today known as *indistinguishability* (IND), has become a standard notion of security for encryption schemes, due to being both a strong security goal, and being comparatively easy to work with. Somewhat surprisingly, this rather simple notion is enough to capture security at the same level as that of semantic security; Goldwasser and Micali proved this by showing that any scheme that is IND-CPA secure is also SEM-CPA secure. They then showed that their scheme is IND-CPA secure, and therefore also SEM-CPA secure.

The reverse implication was later shown by Silvio Micali together with Charles Rackoff and Bob Sloan, implying that IND-CPA and SEM-CPA are really equivalent. "This equivalence", they write, "provides evidence that the right formalization of the notion of security has been reached." [57] (Almost two decades later, a similar equivalence was finally established between IND-CCA and SEM-CCA [76].)

The rise of provable security. It is hard to overstate the impact that Goldwasser and Micali's work has had on the field of cryptography. Putting in place the paradigm of definition-based, reduction-based provable security opened the floodgates, as people went on to show how to achieve all sorts of things traditionally assumed to be impossible. For instance, Goldwasser, Micali, and Rackoff showed in 1985 how to construct proofs for any efficiently verifiable statement (i.e., for any language belonging to the complexity class NP) that reveal nothing but their validity [37], today known as zero-knowledge proofs. Increasingly, cryptography was having an impact on the study of algorithms, known as complexity theory, too, for instance with the introductions of the IP (Interactive Proofs) and MIP (Multi-prover Interactive Proofs) complexity classes, and the work leading up to the much-celebrated Probabilistically Checkable Proofs (PCP) Theorem [3], some of which involved Silvio Micali, and all of which involved Shafi Goldwasser.

Nevertheless, the emerging paradigm of provable security seemed to have little impact on the *practice* of cryptography. Phil Rogaway, who was a student of Micali at the time, recalls of his time at MIT [67]:

> *While a word or two might be uttered in a paper to play-up some far-fetched application, it would be done with a wink of the eye; minimally, practical considerations would have to wait for some distant, less ecstatic day. My wonderful advisor, Silvio Micali, would wax philosophically about how some definition or proof should go, and it is no exaggeration to claim that, as I saw it, philosophy and beauty had unquestioned primacy over utility in determining what was something good to do.*

In the world of cryptographic practice, meanwhile, RSA was still the primary method for public-key encryption, despite lacking any formal proof establishing its security; sometimes implemented with ad-hoc methods to add randomness to the encryption, or to provide some protection against the known chosen-ciphertext attacks, if even that [68]. This divide seemed to stem in part from practitioners not being trained mathematicians[8], in part from proofs being treated by theoreticians as a sort of existence proof (saying in effect, "we have shown that there exist parameters for which these schemes are secure; now it's up to you to go find them"), and, maybe most of all, from all provably secure schemes at the time being horribly inefficient, way too much so to be even considered for use in practice.

1.4 Bellare and Rogaway get Practice Oriented

Having simply assumed that practitioners would take up the challenge and go find practical, efficient implementations of these highly theoretical constructions, Mihir Bellare and Phillip Rogaway, who had met as students at the Theory of Computation group at MIT (at the time including both Shafi Goldwasser and Silvio Micali), were surprised to learn that their new colleagues at IBM were doing no such thing. Rogaway, who felt he had come to IBM to do some "community-service work" by "bringing a bit of the Science of cryptography to the masses of poorly informed security practitioners", found the character of work at IBM to be completely different from what he and Bellare had anticipated [67]:

[8] Neal Koblitz and Alfred Menezes write in their infamous survey paper *Another Look at 'Provable Security'*: "Regrettably, many 'provable security' papers seem to have been written to meet the goal of semantic security against comprehension by anyone outside the field." [51]

> *It was what we came to call, in nearly daily phone conversations between the two of us, the* two-hat approach. *Here's how it works.*
>
> - *You could wear your* theory hat*, in which case you'd try to write a nice paper, preferably one for STOC, FOCS, or CRYPTO; or,*
> - *You could don your* practice hat*, which, fortunately, nobody felt compelled to do all that often. There you'd use your intuitions, intelligence, and predisposition to skepticism to design, attack, or refine some real-world scheme that, against all odds, somehow came your way.*
>
> *These two modes didn't much interact or interfere. They're just different ways to work.*

While there are many occupations in which one is required to switch between different "hats" in the above manner, Bellare and Rogaway were just not very comfortable with the great gap sitting between these particular hats. Their problem was that, while wearing the practice hat [67],

> *you're effectively asked to abandon not just your theory-rooted knowledge but, worse, it seemed like you needed to abandon your theory-rooted sensibilities too. Basic things, like the expectation that you'll formalize things before getting too far trying to analyze a scheme.*

This frustration led them to initiate a program they came to refer to as "practice-oriented provable security". It started off with three papers.

The first one recast the Kerberos protocol as a general protocol which they named Authenticated Key Exchange, and studied the level of security it achieved under some natural assumptions [12]. Among practitioners, Kerberos was the most famous cryptographic protocol to come out of MIT since RSA. Rogaway, however, had to admit to his coworkers that he had never heard of the thing, its design having had no influence from the Theory of Computation group (allegedly leading some to doubt whether he was truthful about having studied cryptography at MIT at all) [67].

The second paper aimed to bring symmetric cryptography, which had thus far been completely removed from the world of cryptographic theory, into the realm of provable security. To start off, they studied a popular construction of message authentication codes (MAC) from blockciphers, and quantified the level of security the MAC could achieve given the assumed security of the blockcipher. As with any theory paper, this involved giving formal definitions quantifying security, and reductions supporting each claim [9].

Both of these papers spawned sub-fields of cryptographic research that, while not particularly relevant for this thesis, are still very much alive today.

The reader should note the subtle shift in language: we are no longer talking about a scheme being proven “secure” or “insecure”. Instead, we are now asking *how* secure a scheme is, given the properties of its constituent parts. This shift is at the heart of practice-oriented provable security. In fact, it is one of the main reasons why symmetric ciphers had so far been considered incompatible with the paradigm of provable security: with symmetric schemes being finite mathematical objects, asymptotic concepts like “polynomial-time”, “security parameter”, and “negligible function” lose their meaning. What Bellare and Rogaway realized, however, was that as long as you did not try to make any final claim about whether or not a scheme was “secure”, you could perfectly well employ the same definition-and-reduction based paradigm to quantify *what* security a scheme would achieve. Even better, with theorems now stated in a concretely quantified manner, a practitioner could in principle choose a desired level of security (say, that an adversary should not be able to break the scheme in less than 2^{128} steps), and get appropriate parameters right out of the formula.

Random oracles are practical. The third paper of this era, “Random Oracles are Practical” [11], has ended up as the pair’s most famous—and most controversial [67]. Here, they introduce the random oracle model (ROM), suggesting it is both a powerful proof technique, and a paradigm that “provides a bridge between cryptographic theory and cryptographic practice”. They write [11]:

> *Cryptographic theory has provided a potentially invaluable notion for cryptographic practice: the idea of provable security. Unfortunately, theoretical work often seems to gain provable security only at the cost of efficiency.*

The inefficiency of all provably secure schemes to date had severely hindered their uptake in implementations, as the typical practitioner would value efficiency of their protocols much higher than arcane claims of proven security from some esoteric theoreticians, many of whom had not coded a line in their life.

Clearly, then, the burden was on theoreticians to come up with schemes *as efficient as those used in practice*, but for which strong security guarantees *could* be given.

One primitive possessing many useful properties that had been ignored by the theoretical community, despite being much used in practice, was that of a *hash function*. Simply put, a hash function H is an object that takes an input of any length, x, and produces a fixed-size, random-looking output $H(x)$; furthermore they are deterministic, so the same input always yields the same output. Cryptographic hash functions are typically required to be *preimage-resistant*, meaning that given $H(x)$ it should be difficult to recover x, and *collision-resistant*, meaning that it should be difficult to find distinct x_1 and x_2 such that $H(x_1) = H(x_2)$.

Despite their apparent simplicity, practitioners had long realized the utility of hash functions for building secure and efficient cryptography, and a growing body of anecdotal evidence showed that by cleverly utilizing them, a construction could often achieve much better security with very little cost to the efficiency of the scheme. Bellare and Rogaway, donning their theory hats, write [11]:

> *What really* is *this object? To date, there has been no satisfactory answer. That is, there is no formal definition which captures a large fraction of the properties this function seems to possess—and it is not clear that one can be found.*

Their proposed answer is to model hash functions as truly random functions, and give everyone involved in a protocol or security proof *oracle access* to it, meaning that:

1. Anyone can call it on some input and instantly get an answer.
2. The output will be fully consistent with what anyone else would see if they were to call it on the same input.
3. No one can observe any of its inner workings, only its input/output behaviour.

Truly random functions can of course not be implemented in practice, as it would essentially require a look-up table containing random values for every possible input. This quickly becomes infeasible to implement: even for a rather conservative input limit of, say, 512 bits, the table would have 2^{512} elements, making it too large to fit in the observable universe (a fact that would remain true even if you were to somehow encode each entry in a single electron and densely pack them all across the observable universe with no space to spare).

Now for the trick: random oracles may be impossible to implement, but they are easy to *simulate*. Assume the output is of some fixed length ℓ. Now, every time someone calls the random oracle, you just check on your internal notepad whether the oracle has been called before on that value. If yes, you return that value. If not, you flip ℓ coins and return the result, making sure to jot down the result and the associated input value for future reference.

To see why this is helpful, it is useful to bear in mind how a typical reduction works. Say you want to show that breaking your newly cooked-up cryptosystem is as hard as solving some underlying hard problem. You prove this by showing that *if* someone can break your cryptosystem, *then* you can solve the underlying problem.

The first step is to somehow embed your *challenge*, the thing you seek to solve, into a challenge ciphertext for the adversary to crack. In addition, you now *simulate* the random oracle in the above manner, flipping coins and returning the result each time the random oracle is called. Given the above equivalence, the adversary will be none the wiser.

Crucially, if the encryption scheme is well constructed, running the encryption and decryption algorithms will *also* involve calling the random oracle. You want two things:

1. That successfully breaking the system (for some definition of "breaking") must by necessity involve calling the random oracle on the value that you, the reduction, seek.
2. If the adversary does *not* call the random oracle on that value, then it has no winning strategy better than random guessing.

The final step is easy: if the adversary succeeded in breaking your system, then it must have called the random oracle on the value that you care about. Simply extract the correct value from your notepad, and you are done! (Though in order for this strategy to work, you better be able to recognize the correct value when you see it. If the underlying hard problem is reversing some one-way function f, as is often the case, this is of course as easy as computing $f(x)$ for a candidate value x and checking if it matches the challenge you were issued. In other cases, it might require additional assumptions on the underlying problem, see e.g. Article 2, Thm. 9.)

So, we have proven our scheme secure. Now all that remains is to pick a good hash function, and replace every oracle call in our construction with a hash function call. Right?

The random oracle controversy. If the above argument leaves a weird taste in your mouth, yours is certainly not the first. There is something intrinsically *weird* about giving the reduction total control over the random oracle, and having the central security argument rely on an ability to read off and answer oracle calls. To make matters weirder, many reductions (including one featured in this thesis, see Article 2, Thm. 9) utilize the fact that they control the random oracle to program certain values into some of the responses, rather than just flipping random coins, the argument being that it's fine as long as the adversary does not notice. When allowing for the latter behaviour, we say we are in the *programmable* random oracle model, to emphasize the additional power utilized by the reduction [62].

Say you instantiate the random oracle with the state-of-the-art hash function SHA3. This hash function features in countless protocols, and is getting evaluated millions of times every day by users all over the globe. Are we somehow saying that in our model, the reduction should be able to read off and respond to all those oracle calls? Or, in the case of the programmable random oracle model, adaptively change its output?

Nonetheless, the random oracle model, with its associated proof techniques, has been remarkably successful: if someone broke a system with a random oracle proof, it has usually been the case that it is because they found some weakness, or structure, in the hash function construction, making it behave *not* as a random oracle [58]. Presumably, switching the broken hash function out for a stronger one would solve the problem.

There are exceptions, though. Canetti, Goldreich, and Haveli were the first to give an example of a cryptographic scheme that is provably secure in the random oracle model, and yet is insecure when instantiated with *any* hash function [20]. Bellare and Rogaway had already noted [11] that it is easy to construct counter-examples by cooking up schemes that rely in some contrived way on the hash function that you intend to replace the random oracle with, concluding that for a ROM proof to be valid, all such dependencies must be disallowed. The significance of the result of Canetti et al., then, is that their construction has no such dependency: there *cannot exist* a hash function making their scheme secure when instantiated.

In the years since, many other such random oracle separations have been found [8, 27, 62]. To some, these results cast severe doubts on the validity of the random oracle model. Others feel strangely reassured by them, due to the highly contrived nature of the separating schemes. In reference to one such separation (due to Bellare et al. [8]) Koblitz and Menezes write [51]:

> *(...) if one of the world's leading specialists in provable security (and coauthor of the first systematic study of the random oracle model) puts forth his best effort to undermine the validity for practical cryptography of the random*

```
Experiment $\mathsf{Exp}^{\text{IND-CCA}}_{\mathsf{PKE}}(\mathbb{A})$
    $(\mathsf{pk}, \mathsf{sk}) \leftarrow\!\!\$ \ \mathsf{PKE.Kg}(\mathsf{pm})$
    $b \leftarrow\!\!\$ \ \{0, 1\}$
    $\hat{b} \leftarrow\!\!\$ \ \mathbb{A}^{\mathcal{E},\mathcal{D}}(\mathsf{pk})$
    return $b = \hat{b}$
```

```
Oracle $\mathcal{E}(m_0, m_1)$
    if $|m_0| \neq |m_1|$ : return $\lightning$
    $c^* \leftarrow\!\!\$ \ \mathsf{PKE.Enc}_{\mathsf{pk}}(m_b)$
    return $c^*$
```

```
Oracle $\mathcal{D}(c)$
    if $c = c^*$ : return $\lightning$
    $m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}}(c)$
    return $m$
```

Figure 1.1: IND-CCA security, codified as a game in the manner popularized by Bellare and Rogaway [14], and Shoup [74].

> *oracle assumption, and if the flawed construction in* [the cited work] *is the best he can do, then perhaps there is more reason than ever to have confidence in the random oracle model.*

Chosen-ciphertext attacks, revisited. As explained in Sect. 1.3, Eve can break any RSA ciphertext encrypted under Alice's public key if she can trick Alice into decrypting *other* ciphertexts on her behalf. A natural question then is, is it possible to patch RSA to resist such chosen-ciphertext attacks?

As good theoreticians, the first step to answering this question is to define what we mean by "resisting chosen-ciphertext attacks". As demonstrated by Bellare and Rogaway [14] and Victor Shoup [74], notions of security—as well as many security reductions—become much easier to reason about if we envision them as *games*, or experiments, played by our adversaries. We then measure the strength of a cryptosystem by how hard the game, when instantiated with the cryptosystem, is to win. Typically, these games are won by correctly guessing a hidden challenge bit: in one extreme, the best the adversary can do is to guess the bit, thus guessing correctly and winning the game with probability $1/2$; we then say that the adversary has *advantage* $= 0$. At the other extreme, the adversary guesses correctly every time; we then say that the advantage is 1, and may conclude—if the adversary in question is reasonably efficient—that the scheme is broken.

A couple of different formalizations of CCA-security have appeared over the years, the first one due to Naor and Yung [59], who formalized indistinguishability in the presence of chosen-ciphertext attacks in the public-key setting and provided a scheme they could show achieved their notion, today known as IND-CCA1. It goes as follows. First, the game initializes by drawing a public–private keypair $(\mathsf{pk}, \mathsf{sk})$, and a uniformly random challenge bit b. Next, the adversary $\mathbb{A}$ is given pk, and access to a decryption oracle $\mathcal{D}$, to which it can submit ciphertexts c and have them decrypted using sk. The oracle returns the result: a message if the ciphertext was well-formed, or the special symbol $\perp$ if not.

Once $\mathbb{A}$ has played around with $\mathcal{D}$ to its satisfaction, it indicates to the game that it is ready to be challenged by submitting two equal-length plaintexts m_0 and m_1; the game then encrypt m_b, and returns the resulting challenge ciphertext c^*. Crucially, $\mathbb{A}$'s access to $\mathcal{D}$ is at this point revoked. With the aid of whatever it learned from the first stage of the game, it tries to pry from c^* which of the two messages it is hiding. Finally, it outputs a guess $\hat{b}$, and wins the game if it guessed correctly.

This notion is somewhat unsatisfactory in that it seems to model an attacker that only has access to the decryption algorithm *before* it discovers the real target of interest. There is little reason to expect real attackers to adhere to this restriction, and CCA1 attacks are therefore sometimes referred to as "lunchtime attacks", playfully alluding to an attacker that only gains access to their target's computer while the target is out for lunch.

Rackoff and Simon [64] suggested a model of chosen-ciphertext attacks in which no such restriction is made, except that the attacker should of course not be allowed to win the game by asking for the decryption of the challenge ciphertext. This model of "adaptive" chosen-ciphertext attacks is sometimes called IND-CCA2 to distinguish it from its non-adaptive counterpart; we will simply refer to it as IND-CCA.

Fig. 1.1 provides the game code of IND-CCA, presented in the general style employed in our works. Here, $\mathsf{PKE.Kg}$, $\mathsf{PKE.Enc}$, and $\mathsf{PKE.Dec}$ are the key generation, encryption, and decryption algorithms, respectively, that are associated with the scheme "PKE". Note how we now also provide the challenge through an oracle interface; among other things, this makes for an easy transition to multiple challenges (see Sect. 2).

The game starts at the top left of the figure: it starts out by initializing itself, generating a keypair from the system parameters pm before calling the player $\mathbb{A}$, who is given the public key and oracle access to an arbitrary number of decryptions ($\mathcal{D}$) and a single challenge ($\mathcal{E}$). Notice how the decryption oracle will refuse any attempt to have the challenge ciphertext decrypted, returning the error symbol $\lightning$. As indicated by the dollar-tailed arrow $\leftarrow\!\!\$$, $\mathbb{A}$ is allowed to use randomness when producing its output, such as a coin flip; at some point it outputs some guess $\hat{b}$, and as usual wins if it guessed correctly.

Dolev, Dwork, and Naor [23] were the first to present a scheme that achieves this stronger notion of IND-CCA, though—as was typical of the time—the scheme was highly theoretical in nature, employing expensive zero-knowledge proofs that gave little hope for practical implementations any time soon. Would it be possible to give a scheme that *both* could be shown to achieve such a strong notion of security, *and* be practical enough to actually be used?

Bellare and Rogaway showed that the answer is yes—at least in the programmable random oracle model. As an example, they showed how a simple construction based on two hash functions is provably IND-CCA-secure when the hash functions are modelled as random oracles [11], while achieving near-optimal efficiency, with encryption and decryption both requiring only two hash function evaluations plus one execution of the underlying trapdoor permutation. (We meet this construction again in Article 3, Sect. 5.1.)

How to make RSA IND-CCA secure. Encouraged by their positive result, and recognizing that RSA was by far the most popular PKE scheme at the time, Bellare and Rogaway turned to the problem of making RSA IND-CCA secure. Of course, RSA was not (and is still not) known to reduce to factoring, so they would settle for showing that anyone who can win the IND-CCA game with significant advantage would also be able to reverse the RSA function with high advantage.

They named their solution OAEP, for "Optimal Asymmetric Encryption Padding" [13]. As the name implies the scheme specifies a *padding*, i.e. an extension of the messages to be encrypted that ensures that encryption involves randomness, and, through clever use of hash functions, makes it provably hard to form well-formed ciphertexts by altering known ones, assuming the RSA function is hard to invert. The RSA standard [68] had in fact already been recommending the use of encryption padding for several years, among other things for the randomization of encryption, but their recommended padding scheme was highly heuristic in nature, and not based on any formal arguments. What it was, was very efficient. With f referring to the underlying k-bit-to-k-bit trapdoor permutation (i.e., the RSA function), Bellare and Rogaway write [13]:

> *What practitioners want is the following: encryption should require just one computation of f; decryption should require just one computation of f^{-1}; the length of the enciphered text should be precisely k; and the length n of the text x that can be encrypted is close to k. Since heuristic schemes achieving these conditions exist [47, 68], if provable security is provided at the cost of violating any of these conditions (e.g., two applications of f to encrypt, message length $n + k$ rather than k) practitioners will prefer the heuristic constructions. Thus to successfully impact practice one must provide provably-secure schemes which meet the above constraints.*

As the reader may have already guessed, they achieved this through a clever construction using hash functions modelled as random oracles. They thus provided a padding scheme that was as efficient as the ones already in use, while achieving "an assurance benefit almost as good as that obtained by provable security" [13]. By 1998, the RSA standard had been altered to recommend OAEP as the standard encryption padding.

There was just one small problem: their proof was, in fact, invalid. Surprisingly, the fallacy took no less than seven years to be discovered[9], so when Victor Shoup published his observation in 2001 [73], in which he showed not only that the proof was wrong but that it *couldn't be patched*, OAEP had already been incorporated into the RSA standard for several years. He remedied the situation by giving an equally efficient variation of the scheme, which he called OAEP+, for which he could finally show IND-CCA security.

His objection against RSA-OAEP was aimed at the claim of proven security, and did not yield a practical attack. Actually, RSA-OAEP turned out to be something of a lucky bet. He writes [73]:

> *It should be stressed that these results do not imply that a particular instantiation of OAEP, such as RSA-OAEP, is insecure. (...) In fact, it turns out—essentially by accident, rather than by design—that RSA-OAEP is secure in the random oracle model; however, this fact relies on special algebraic properties of the RSA function, and not on the general OAEP scheme.*

Tightness. The OAEP and OAEP+ proofs alike have another problem: they are highly *non-tight.* This means that the reduction used in going from breaking the encryption system to breaking the underlying problem incurs a lot of computational overhead. Security reductions provide lower bounds on attacks: say you want a system that is guaranteed secure unless an adversary spends $t_1 = 2^{128}$ computing steps (more than any modern-day supercomputer could achieve before being swallowed by the sun along with the earth), and say that your reduction from breaking the system to solving the underlying hard problem requires time $t_2 = t_1^2$. That would mean that you should choose the parameters such that solving the underlying problem is believed to take $(2^{128})^2 = 2^{256}$ steps (making for a less efficient scheme). Thus, the reduction is *lossy*; had $t_2 \approx t_1$, the reduction would have been *tight.*

This is exactly what the OAEP reduction (original as well as corrected) states—and yet, no recommendation was made to increase the parameters to compensate for the loss. On the contrary, Bellare and Rogaway's claim was that OAEP could provide provable security guarantees while remaining *as efficient* as the schemes already employed. But the status quo was (and in many parts of the community, to a degree remains) that, if

[9] Koblitz and Menezes lament that "the strange history of OAEP—where a 'proof' was accepted for seven years before a fallacy was noticed—hardly inspires confidence [in provable security]" [51].

you want (for example) RSA with 128 bits of security, you choose the parameters of the underlying RSA function such that it is believed to be hard to break in time 2^{128}—*not* in time 2^{256}. Note how we are not saying that there are more efficient attacks when instantiated with such parameters; we are just saying that the possibility can no longer be ruled out within the given model.

Strictly adhering to this reasoning seems to take us back to square one: if our schemes are to have provable security guarantees, they can no longer compete with the ones used in practice, and will therefore likely be ignored by practitioners, or instantiated with parameters not known to be secure.

But any proof is better then no proof, right? So maybe details like these should just be swept under the proverbial rug, for the betterment of practice? Koblitz and Menezes write [51]:

> *Unfortunately, [concrete] analysis is generally missing from papers that argue for a new protocol on the basis of a "proof" of its security. Typically, authors of such papers (...) give a non-tight reductionist argument, and at the end give key-length recommendations that would make sense if their proof had been tight. They fail to inform the potential users of their protocol of the true security level that is guaranteed by the "proof" if, say, a 1024-bit prime is used. It seems to us that cryptographers should be consistent. If one really believes that reductionist security arguments are very important, then one should give recommendations for parameter sizes based on an honest analysis of the security argument, even if it means admitting that efficiency must be sacrificed.*

Shortly after, Dan Boneh gave more efficient variants of both OAEP and OAEP+, which he called SAEP(+) (for "Simple" OAEP(+), that reduced the number of hash function evaluations with no impact on security [15]. That is all well and good; what is more interesting is that he *also* showed, in what amounts to an unexpected comeback, that employing Rabin encryption in place of RSA—the very scheme so infamous for being badly broken by chosen-ciphertext attacks—makes the security proof *tight*. As an added bonus, of course, the security reduction now also goes all the way down to the hardness of factoring, just as it had when Rabin first introduced his scheme.

In other words, an ability to break Rabin-SAEP in t time, would now immediately yield an algorithm for factoring in essentially t time (with some comparatively minor additional terms in the running time, easily accounted for). Thus, one can pick parameters competing with the most efficient schemes out there, and still have a well-founded security guarantee.

At this point we are essentially done, are we not? We have schemes that are just as efficient as those used in practice, that tightly achieve IND-CCA-security under the assumption that well-studied number-theoretical problems like factoring are hard—at least as long as we believe in the programmable random oracle model. Have we finally solved public-key encryption?

1.5 Looking Ahead

Post-quantum cryptography. In his 1977 column on the RSA cryptosystem, Martin Gardner writes [30]:

> *Rivest and his associates have no proof that at some future time no one will discover a fast algorithm for factoring composite numbers as large as the [modulo size] they used or will break their cipher by some other scheme they have not thought of. They consider both possibilities extremely remote.*

As it turns out, all it took was to consider a remote technology. In 1994, Peter Shor showed that if one could construct a *quantum* computer, then there would be an algorithm to factor any number in polynomial time, completely breaking any cryptosystem relying on the hardness of factoring such as RSA and Rabin encryption [71]. A variation of the same algorithm also solves discrete logarithms efficiently, rendering also Diffie–Hellman key exchange insecure. While the potential utility of quantum computers had been pondered more than a decade earlier by legendary physicist Richard Feynman (in particular for simulating, well, quantum physics[10]) [26], it was ironically the *threat* of quantum computers, namely Shor's algorithm, that kickstarted a huge momentum of research trying to figure out whether these devices can be built.

We have come a long way since then: last year, IBM revealed their new flagship quantum chip "Osprey", comprising 433 quantum bits, or qubits [44]. Classically simulating a program run on such a device would in the worst case require around 2^{433} steps, far beyond the capability of any conceivable computer. And yet 433 qubits is not a lot, particularly given the thousand-fold increase in numbers of qubits required by the expensive error-correcting codes keeping the computation stable from environmental interference:

[10] Feynman's 1982 talk concludes, in a characteristically clairvoyant statement, "Nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical, and by golly it's a wonderful problem, because it doesn't look so easy." [26]

state-of-the-art estimates put the required qubit count to factor 2048-bit RSA moduli at roughly 20 million [34]. Yet, if we see a Moore's Law style evolution in the coming years, such a device could be only a decade or two away.

So while we may feel safe from the quantum threat for now, there are settings in which they are already posing a risk, namely for secrets that need to *stay* secret for a long time. If these were to be stored or transmitted using RSA, an interested party might snatch them up and store a local copy, waiting for a large enough quantum computer to be built for them to decrypt it.

Gardner goes on [30]:

> *Even in the unlikely event that the M.I.T. system is breakable there are probably all kinds of other trapdoor functions that can provide virtually unbreakable ciphers.*

And indeed: recently, the US National Institute for Standards and Technology (NIST) ended the third round in an ongoing selection process to find new, *post-quantum* secure key encapsulation mechanisms and signature schemes, based on hard problems *not* known to be solvable by quantum computers [2].

Key encapsulation mechanisms, or KEMs, are a limited kind of public-key primitive that draw and encapsulate symmetric encryption keys, and are among other things used to instantiate hybrid PKE schemes (as explained in Sect. 1.2) and authenticated key exchange (as mentioned in Sect. 1.4). KEMs will play a prominent role in Article II of this thesis.

For encryption, the initial winner of the selection process is CRYSTALS-Kyber, an IND-CCA-secure KEM based on the hardness of finding certain vectors in high-dimensional lattices [5], though more standards will follow.

Schemes built on lattice problems often come with an interesting characteristic, namely that *correctness* of the scheme may sometimes fail [5]. By "correctness", we mean that if a message is encrypted using a public key, decrypting it using the corresponding private key should lead to the same message. Utility requires that this failure rate is low (say, 2^{-128}). Yet, many proofs rely on perfect correctness in subtle ways, making imperfect correctness an important notion to consider if you want your proofs to hold for soon-to-be standardized post-quantum schemes. We will expand on imperfect correctness in Article 2, where we give a generalized definition, as well as a novel generalization of CCA that allows an attempted decryption of a challenge ciphertext (which, remember, must normally be disallowed) to go through in the case that the decryption was erroneous.

Schemes with high error rates thus become trivially insecure in our model, as long as even a small part of the original message remains after an incorrect decryption, capturing potential attacks that exploit this weakness.

Quantum random oracles. The random oracle assumption runs into difficulties when considering quantum attackers. The reason is that a random oracle is supposed to model a publicly available algorithm (e.g. a hash function), which the attacker is free to load up on its quantum hardware, performing quantum computations on it as it sees fit. In keeping with the random oracle model, these computations are black-box only, i.e. anything it learns from the algorithm has to be learned from its input/output behaviour alone.

However, one of the unique capabilities of a quantum computer is running in *superposition*. You can think of superpositions as a sort of generalized way of running a computation in parallel, except without having to use any more hardware than you would need to run one instance of the program. On the other hand, you only get one answer out the other end. However, you can manipulate the probability weights of each possible answer through clever manipulation of the superposition state.

Therefore, in order to model hash functions in the presence of quantum attackers, we would have to switch to the Quantum Random Oracle Model, or QROM [16], in which superposition queries to the random oracle are allowed.

A lot of things that are easy to do in the ROM are very challenging in the QROM. As a simple example, imagine you are a (non-quantum) reduction, and you want to simulate the random oracle by "lazily sampling" it in the manner described in Sect. 1.4, i.e. sampling a fresh output for each query, saving the results in a gradually growing table. If the quantum adversary now queries the oracle on a uniform superposition of every possible input, then you would in principle have to define the full table of the random oracle before responding. Being a mere probabilistic polynomial-time Turing Machine, this is clearly beyond your capabilities. If you additionally want to *program* the random oracle, you're in even more trouble. Nevertheless, recent years have seen the development of a number of useful techniques for dealing with these difficulties [49,54,80].

Provable security in a quantum world. In the distant future, when quantum computers are ubiquitous among users and adversaries alike, we will have to upgrade our notions of security once more to accommodate the fact that also challenge queries, decryption queries, etc., may be called in superposition. This is an interesting challenge, as the upgrade to so-called "fully-quantum" security notions does not appear to be straightforward. Several suggestions have appeared [17, 28, 29], and research is very much ongoing. I personally find it a fascinating—if somewhat esoteric—challenge to recast the questions tackled in the present thesis in a fully quantum world.

The vision of provable security. In "New Directions in Cryptography", Diffie and Hellman write [22]:

> *The failure of numerous attempts to demonstrate the soundness of cryptographic systems by mathematical proof led to the paradigm of certification by cryptanalytic attack set down by Kerchoffs [*sic*] in the last century. (...) The development of computers has led for the first time to a mathematical theory of algorithms which can begin to approach the difficult problem of estimating the computational difficulty of breaking a cryptographic system. The position of mathematical proof may thus come full circle and be reestablished as the best method of certification.*

We have over the past several sections seen how this vision has slowly been coming to fruition. Martin Gardner's column on RSA concluded on a similar, if somewhat more sentimental, note [30]:

> *All over the world, there are clever men and women, some of them geniuses, who have devoted their lives to the mastery of modern cryptanalysis. Since World War II even those government and military ciphers that are not one-time pads have become so difficult to break that the talents of these experts have gradually become less useful. Now these people are standing on trapdoors that are about to spring open and drop them completely from sight.*

Gardner lamented needlessly, however, as evidenced e.g. by the aforementioned NIST post-quantum selection process. Their call for proposals stated that "submitters are not required to provide a proof of security, although such proofs will be considered if they are available." [61] Consequently, very few schemes came with one, and discussions on what schemes to accept for standardization was heavily tailored towards best-known attacks, rather than formal justifications for their security claims [60].

Of course, the competition-like format of the selection process is specially tailored towards the testing of proposed schemes by cryptanalytic attack. Nevertheless, one is left with the feeling that we remain as a community with one foot stuck in the paradigm of Kerckhoffs, and that there may be a long road ahead before the vision of Gardner, and of Diffie and Hellman, Goldwasser and Micali, Bellare and Rogaway, Rabin and Shoup, and many others, come to full bloom.

Chapter 2

Preliminaries: Modelling the Real World

Imagine you are in charge of some big company “A” that provides web services to millions of customers across the globe. You tout state-of-the-art, top-of-the-line cryptography, enough to secure any sort of web business model your clients could dream up. Your greatest fear, of course, is the possibility that the security of one of your clients is one day broken in a manner that can be traced back to your service. The millions of dollars lost from lawsuits notwithstanding, it would mean a fatal blow to your reputation, and a mass migration of clients to your biggest competitor, “G”.

You are not really worried, though: being well-versed in the subtle challenges of cryptography, you have opted for a cryptosystem with a tight security proof to an underlying hard problem, and with no loss in performance to boot. Estimating that the top supercomputer in the world would be able to perform roughly 2^{60} computing steps within a reasonable time, and with the security proof being tight, you set the system parameters so that the best known algorithm would require at least 2^{80} steps to solve the underlying hard problem. With a significant security margin in place, you roll out your system, safe in the knowledge that even the world’s biggest supercomputer could not break your system in any foreseeable future.

Six months later, your worst nightmare comes true: it is revealed that a high-value customer was compromised through a direct attack. What’s worse, an independent investigation concluded that the attack was not due to some weakness in their system, but a direct break of the cryptography issued by your service. As the cherry on top, the investigation revealed the attacker to be not some huge, government-backed agent, but rather a tech-savvy teen with a school laptop and a free weekend, with no apparent

```
Experiment $\mathsf{Exp}^{\text{q-IND-CCA}}_{\mathsf{PKE}}(\mathbb{A})$
$(\mathsf{pk}, \mathsf{sk}) \leftarrow\!\!\$\ \mathsf{PKE.Kg}(\mathsf{pm})$
$b \leftarrow\!\!\$\ \{0,1\}$
$\hat{b} \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D}}(\mathsf{pk})$
$\mathbf{return}\ b = \hat{b}$
```

```
Oracle $\mathcal{E}(m_0, m_1)$
$\mathbf{if}\ |m_0| \neq |m_1| : \mathbf{return}\ \lightning$
$c^* \leftarrow\!\!\$\ \mathsf{PKE.Enc}_{\mathsf{pk}}(m_b)$
$\mathcal{C} \overset{\cup}{\longleftarrow} c^*$
$\mathbf{return}\ c^*$
```

```
Oracle $\mathcal{D}(c)$
$\mathbf{if}\ c \in \mathcal{C} : \mathbf{return}\ \lightning$
$m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}}(c)$
$\mathbf{return}\ m$
```

Figure 2.1: The multi-challenge q-IND-CCA security game, where q upper bounds the number of challenge queries. We use the notation $\mathtt{X} \overset{\cup}{\longleftarrow} x$ for the operation $\mathtt{X} \leftarrow \mathtt{X} \cup x$.

motivation other than to see if she could.

What went wrong? You've checked and double-checked, and you're sure all system components were implemented to specification. Has this teen really made a great leap ahead in algorithmic theory that she's choosing to keep hidden from everyone else? Or was there a flaw in the reasoning?

2.1 Multi-Challenge

As mentioned in Sect. 1.4, $\mathsf{Exp}^{\text{IND-CCA}}_{\mathsf{PKE}}(\mathbb{A})$ (Fig. 1.1) is easily upgraded to allow for several calls to the encryption oracle $\mathcal{E}$. All that needs to be done is to define a set $\mathcal{C}$, and add each challenge encryption to it as they're produced, see Fig. 2.1. Then, when the decryption oracle $\mathcal{D}$ is called on some c, it checks whether $c \in \mathcal{C}$, returning $\lightning$ as usual if it is. Now, the adversary can produce as many challenges as it likes, and as long as it breaks at least one of them, it will learn the value of the challenge bit b and win the game.

How does the new notion compare to the original one in terms of strength? Well, one direction is easy: security under a notion with multiple challenges implies security under a notion with only one challenge. Let $\mathbb{B}$ be a reduction playing the multi-challenge game. It can then simulate the single-challenge game to any adversary $\mathbb{A}$ by forwarding the first challenge, and ignoring its ability to ask for more challenges. Denote the probability that $\mathbb{A}$ wins the single-challenge game by $\Pr\big[\mathsf{Exp}^{\text{IND-CCA}}_{\mathsf{PKE}}(\mathbb{A})\big]$, and the probability that $\mathbb{B}$ wins the multi-challenge game by $\Pr\Big[\mathsf{Exp}^{\text{q-IND-CCA}}_{\mathsf{PKE}}(\mathbb{B})\Big]$, where q is maximum number of challenge queries made by $\mathbb{B}$. We define their *advantage* to be this win probability, offset so that randomly guessing yields advantage 0 and always winning yields advantage 1:

$$\mathsf{Adv}^{\text{IND-CCA}}_{\mathsf{PKE}}(\mathbb{A}) := 2 \cdot \Pr\big[\mathsf{Exp}^{\text{IND-CCA}}_{\mathsf{PKE}}(\mathbb{A})\big] - 1\,,$$

and similarly for $\mathbb{B}$. Then what the above argument shows is that there is a reduction

```
Experiment $\mathsf{Exp}^{(q,\kappa)\text{-IND-CCA}}_{\mathsf{PKE}}(\mathbb{A})$
$(\mathsf{pk}_1, \mathsf{sk}_1), \ldots, (\mathsf{pk}_\kappa, \mathsf{sk}_\kappa) \leftarrow\!\!\$\ \mathsf{PKE.Kg}(\mathsf{pm})$
$b \leftarrow\!\!\$\ \{0,1\}$
$\hat{b} \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
return $b = \hat{b}$
```

```
Oracle $\mathcal{E}(i, m_0, m_1)$
if $|m_0| \neq |m_1|$ : return $\lightning$
$c^* \leftarrow\!\!\$\ \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m_b)$
$\mathcal{C}_i \xleftarrow{\cup} c^*$
return $c^*$
```

```
Oracle $\mathcal{D}(i, c)$
if $c \in \mathcal{C}_i$ : return $\lightning$
$m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
return $m$
```

Figure 2.2: The multi-user generalization of q-IND-CCA, denoted (q, κ)-IND-CCA, where κ is the number of users.

$\mathbb{B}$ such that for all adversaries $\mathbb{A}$,

$$\mathsf{Adv}^{\text{IND-CCA}}_{\mathsf{PKE}}(\mathbb{A}) \leq \mathsf{Adv}^{\text{q-IND-CCA}}_{\mathsf{PKE}}(\mathbb{B}) \,.$$

This relation is tight, as one advantage is upper bounded strictly in terms of the other, with no extra factors or terms, and the reduction incurs no overhead in running time.

In the other direction, one can show an implication in the form of a so-called hybrid argument [36]. Here, the game is gradually changed, one challenge at a time, from $b = 0$ to $b = 1$, giving $\mathbb{B}$ the advantage of $\mathbb{A}$, but with a factor q tightness loss [7]. In other words, there is a reduction $\mathbb{B}$ such that for all adversaries $\mathbb{A}$,

$$\mathsf{Adv}^{\text{q-IND-CCA}}_{\mathsf{PKE}}(\mathbb{A}) \leq q \cdot \mathsf{Adv}^{\text{IND-CCA}}_{\mathsf{PKE}}(\mathbb{B}) \,.$$

What does this mean? Well, it means that any scheme that is proven tightly IND-CCA-secure, where IND-CCA was modelled as a single-challenge game, can only be said to be secure in the multi-challenge game with a tightness loss of q. Say that a user encrypts ten thousand messages a day, each one of high enough value that any one of them getting broken would be devastating to the user. This would in less than four months sum up to over a million potential target ciphertexts, or roughly 2^{20}. In other words, while the executive's reasoning guaranteed that it would take at least 2^{80} computing steps to break any given ciphertext, this was proven in the single-challenge setting, and so he only actually has a guarantee that it would take at least 2^{60} steps to break one out of all the possible target ciphertexts, within reach for modern supercomputers.

So attacks from supercomputer clusters can no longer be ruled out. Still, it hardly explains the actual attack, performed on a device built for browsing Wikipedia and writing the occasional exam paper.

2.2 Multi-User

The multi-challenge setting was introduced to the public-key setting by Bellare, Boldyreva, and Micali in 2000 [7], and they simultaneously introduced a further generalization by allowing the game to host multiple users. Once again, the update to Fig. 1.1 is straightforward: instead of generating a single keypair at the outset, κ keypairs are generated, where κ is the number of users in the system. Then, when calling $\mathcal{E}$ or $\mathcal{D}$, the adversary specifies which user it wants to challenge or ask for a decryption.

Each user comes with a separate set for keeping track of challenge ciphertexts, for the purpose of disallowing trivial wins by use of the decryption oracle. Note how this means that if c^* was produced by user 1's public key, then the adversary may ask to have it decrypted under user 2's private key, but not user 1's. This means that in the case of a key collision, i.e. that equal public and/or private keys are generated for different users, the adversary may easily win the game through use of the decryption oracle; thus, a scheme in which this happens with high likelihood is rendered insecure (as it should).

The security reduction from multi-user, multi-challenge IND-CCA to single-user (but multi-challenge) IND-CCA once again incurs a tightness loss, namely a factor κ, from a hybrid argument very similar to the multi-challenge to single-challenge case: there is a reduction $\mathbb{B}$, such that for all $\mathbb{A}$,

$$\mathsf{Adv}_{\mathsf{PKE}}^{(\mathrm{q},\kappa)\text{-IND-CCA}}(\mathbb{A}) \leq \kappa \cdot \mathsf{Adv}_{\mathsf{PKE}}^{\mathrm{q}\text{-IND-CCA}}(\mathbb{B})\,.$$

By stringing the reductions together, we see that the reduction to single-user single-challenge IND-CCA therefore loses a total factor $q\kappa$, where q bounds the number of challenge queries per user:

$$\mathsf{Adv}_{\mathsf{PKE}}^{(\mathrm{q},\kappa)\text{-IND-CCA}}(\mathbb{A}) \leq q \cdot \kappa \cdot \mathsf{Adv}_{\mathsf{PKE}}^{\text{IND-CCA}}(\mathbb{B})\,.$$

The company of our unfortunate executive was hosting over a million clients, which in a couple of months produced roughly a million ciphertexts each. Combining the losses, we therefore see that we've gone from a security guarantee of 2^{80}, to one of 2^{40}. Again, note the difference in what we're modelling: a randomly chosen ciphertext from a randomly targeted user would still in all likelihood require 2^{80} steps to break. But among a trillion potential targets, we have no guarantee that at least one of them won't be broken in 2^{40} steps.

Say it turns out that the cryptosystem put in place by "A" is slightly imperfect, and with some small probability outputs a weak ciphertext that is much easier to crack than the usual ciphertexts; let us also say that such ciphertexts are easy to identify. However, with the probability of this happening being so small, the security proof easily accounted for this eventuality without having to patch the construction. But that was for one target ciphertext, not a trillion.

And so our tech-savvy teen simply set up a bot monitoring the traffic to and from the servers of “A”, and set it to notify her if ever a weak ciphertext turned up. After running for a couple of months, it did. She went to work, brute-forcing the ciphertext with known techniques, and broke it in 2^{40} steps using her school laptop over the course of a weekend.

Although this story should of course be taken as nothing but a playful anecdote, it points to a real issue: if we want real-world parameters to be derived from proofs of security rather than the best-known attack against them, then we better make sure that our security notions model the real world.

There are several ways to do so, however, Fig. 2.2 being only one option. For instance, instead of the game containing a single challenge bit, each user could be issued their own challenge bit. Additionally, one could give the adversary the power to *corrupt* a subset of the users, gaining control of their private keys. This leads us finally to our first research question.

Research Question 1. *How do different choices in formalization affect models of multi-user security?*

Håstad’s broadcast attack. The above anecdote might give the impression that multi-user attacks mostly involve finding and breaking the rare weak key or ciphertext. However, this need not be the case: sometimes having multiple targets to choose from already suffices for an attack. For example, with access to three RSA ciphertexts that each encrypt the same message under three separate public keys, Håstad’s broadcast attack [41] allows for the recovery of the message. Hence a “broadcast attack”, as each of the users received a broadcast issued to all (presumably excluding the attacker).

Sharp reductions. Reductions like the above being lossy do not rule out the possibility that someone might someday come up with a better reduction that *does* achieve tightness. To rule out this possibility, Bellare et al. [7] provided a contrived scheme for which there exists an attack in the multi-user setting that is exactly $q\kappa$ times more efficient than in the single-user, single-challenge setting, matching their bound. Later, Bader et al. [6] ruled out tight (black-box) reductions for larger classes of schemes through the

```
Experiment $\mathsf{Exp}^{(q,\kappa)\text{-IND-CCA}\star}_{\mathsf{PKE}}(\mathbb{A})$
    $(\mathsf{pk}_1, \mathsf{sk}_1), \ldots, (\mathsf{pk}_\kappa, \mathsf{sk}_\kappa) \leftarrow\!\$\ \mathsf{PKE.Kg}(\mathsf{pm})$
    $b \leftarrow\!\$\ \{0,1\}$
    $\hat{b} \leftarrow\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{R}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
    if $\mathcal{K} \cap \mathcal{I} \neq \emptyset : \hat{b} \leftarrow 0$
    return $b = \hat{b}$
```

```
Oracle $\mathcal{E}(i, m_0, m_1)$
    if $|m_0| \neq |m_1|$ : return $\lightning$
    $\mathcal{K} \xleftarrow{\cup} i$
    $c^* \leftarrow\!\$\ \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m_b)$
    $\mathcal{C}_i \xleftarrow{\cup} c^*$
    return $c^*$
```

```
Oracle $\mathcal{D}(i, c)$
    if $c \in \mathcal{C}_i$ : return $\lightning$
    $m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
    return $m$
```

```
Oracle $\mathcal{R}(i)$
    $\mathcal{I} \xleftarrow{\cup} i$
    return $\mathsf{sk}_i$
```

Figure 2.3: Multi-user indistinguishability with corruptions, denoted κ-IND-CCA$\star$. If $\mathbb{A}$ both challenges and corrupts a user, then its advantage goes to 0, as enforced by overwriting its guess by 0.

use of a technique known as a *meta-reduction.* (Interestingly, this result requires the adversary to have the ability to corrupt users, see below.)

Thus, the tightness losses associated with the multi-user setting really are inevitable in general; in other words, the bounds are *sharp.* Thus, the search for constructions with tight security proofs becomes a worthwhile endeavour (and a number of such schemes have indeed appeared [32, 33, 39, 40, 43, 52, 77]).

2.3 Corruptions Empower

Leaks happen, of course: someone could have written their private key down on a post-it note and dropped it on the way home for that matter. Therefore, if the goal is to model realistic settings, the adversary should be given the ability to corrupt some of the users, learning their private key, the hope being that unaffected parties remain secure.

One way to model this is to let your notion be a multi-user notion as before (Fig. 2.2), except that now the adversary is additionally given access to a corruption oracle, to which it can request any of the private keys in play. Some care has to be taken to make sure this doesn't lead to trivial wins; in particular, the adversary can ask to challenge or corrupt a user, but the same user cannot be both challenged and corrupted, as this would reveal the underlying challenge bit, see Fig. 2.3.

But wait—is *that* a realistic requirement? After all, one could imagine that an adversary learns some high-value messages (i.e. challenge) by compromising the receiver (for instance by stealing their post-it notes)—you'd still want a guarantee that high-value ciphertexts of *other* clients remain secure, wouldn't you?

In a model in which each user is given their own challenge bit, the adversary could ask for challenges from as many users as it likes, and corrupt whomever it pleases, as long as at least one uncorrupted user stands left to challenge at the end. Then it would only learn the challenge bits for *other* users, unrelated to the one it is going to be guessing at the end.

However, working with multiple challenge bits is not always desirable, as they can make compositions that would otherwise be tightly secure turn lossy [45]. As an alternative, then, one might turn to the notions of *selective opening attacks* (SOA). Indistinguishability-based SOA with receiver opening is a notion modelling the exact same thing, in which the adversary is also free to corrupt and challenge the same users, while having the game formalized using a single challenge bit. The price one pays is that the notion is no longer implied by, but strictly stronger than (and therefore harder to achieve than) IND-CCA. What's more, the notion employs message samplers, and requires message distributions that are *efficiently conditionally resamplable*, a condition that may not be suitable for all settings. If so, one might have to move on to simulation-based SOA, for which no such restriction is made. But at that point one inevitably leaves the real world, as this notion has been shown to be unachievable unless we are in a programmable idealized model, like the random oracle model [78]. And then there is non-committing encryption, an even stronger security notion that also aims to model security against corruptions, and which is necessarily also unachievable in the standard model [62].

So what is going on here? How is it that we have all these different security notions of vastly different strengths when they all claim to model the same thing? Can security against key compromise be achieved in the real world, or not?

In the words of Micali, Rackoff, and Sloan from 25 years ago, when they were working to make sense of security notions like semantic security and indistinguishability [57]:

> *Not completely knowing the strength of these definitions is rather unpleasant. For instance, several protocols have been proved correct adapting the notion of [indistinguishability]. Are these protocols that are secure with respect to a particular definition or are they secure protocols in a more general sense? In other words, a natural question arises: Which of the definitions is the "correct" one? Even better: How should we decide the "correctness" of a definition?*

And so we are led to our second research question.

```
Experiment $\mathsf{Exp}_{\mathsf{PKE}}^{(\mathrm{q,n},\kappa)\text{-IND-CCA}\star}(\mathbb{A})$
    $(\mathsf{pk}_1, \mathsf{sk}_1), \ldots, (\mathsf{pk}_\kappa, \mathsf{sk}_\kappa) \leftarrow\!\!\$\ \mathsf{PKE.Kg}(\mathsf{pm})$
    $b_1, \ldots, b_\kappa \leftarrow\!\!\$\ \{0,1\}$
    $(\mathcal{J}, \hat{b}) \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{R}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
    if $|\mathcal{J}| \neq n \vee \mathcal{J} \cap \mathcal{I} \neq \emptyset : \hat{b} \leftarrow\!\!\$\ \{0,1\}$
    return $\bigoplus_{j \in \mathcal{J}} b_j = \hat{b}$
```

```
Oracle $\mathcal{E}(i, m_0, m_1)$
    if $|m_0| \neq |m_1|$ : return $\lightning$
    $c^* \leftarrow\!\!\$\ \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m_{b_i})$
    $\mathcal{C}_i \xleftarrow{\cup} c^*$
    return $c^*$
```

```
Oracle $\mathcal{D}(i, c)$
    if $c \in \mathcal{C}_i$ : return $\lightning$
    $m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
    return $m$
```

```
Oracle $\mathcal{R}(i)$
    $\mathcal{I} \xleftarrow{\cup} i$
    return $\mathsf{sk}_i$
```

Figure 2.4: Multi-instance indistinguishability with corruptions, denoted (q, n, κ)-IND-CCA⋆, in which the adversary, given q oracle calls $\mathcal{E}$, is tasked with correctly guessing n out of κ challenge bits.

Research Question 2. *How do the different approaches to modelling corruptions compare in terms of strength and utility, and is there a "correct" approach?*

2.4 Multi-Target

In June 2013, Edward Snowden, who had until that time been an employee at the notorious US government National Security Agency, leaked hundreds of thousands of highly classified documents. They detail, among other things, a global mass surveillance effort of immense scale, and showed that the NSA had amassed virtually unlimited power to wiretap any target they like. Or as he put it [38]:

> *Any analyst at any time can target anyone, any selector, anywhere. Where those communications will be picked up depends on the range of the sensor networks and the authorities that that analyst is empowered with. Not all analysts have the ability to target everything. But I, sitting at my desk, certainly had the authorities to wiretap anyone, from you or your accountant, to a federal judge, to even the president if I had a personal email.*

The revelation came as a shock to a populace who—while they might have grown accustomed to the thought of low-level government surveillance efforts for the supposed purpose of national security—would regularly write off those shouting warnings of the ever-watchful eyes of Big Brother (alluding to George Orwell's dystopian vision of a mass surveillance state in *1984*) as nothing but doomsayers and conspiracy theorists. It sent a shockwave through the cryptographic community too, which had long been skeptical of the NSA, but now saw a reality that was even worse than they had feared. Phil Rogaway writes, in his essay *The Moral Character of Cryptographic Work* [66]:

> *If cryptography's most basic aim is to enable secure communications, how could it* not *be a colossal failure of our field when ordinary people lack even a modicum of communication privacy when interacting electronically?*

He posed a challenge to his fellow cryptographers:

> *I call for a community-wide effort to develop more effective means to resist mass surveillance.*

In 2020, Auerbach, Giacon, and Kiltz took up the challenge to develop meaningful notions of security against mass surveillance [4]. Building on earlier work of Bellare et al. [10], they provided security notions similar to those of the multi-user setting, except that now every user *does* come with their own challenge bit. The idea that an adversary has the power to conduct mass surveillance is then captured by having it guess correctly a subset of the challenge bits of size at least n (out of κ potential targets) in order to win the game, as shown by correctly guessing the xor of the n challenge bits, see Fig. 2.4.

These notions have come to be known under the (unfortunately rather vague) name of "multi-instance" security. Auerbach et al. then quantified the resistance of various schemes against mass surveillance, by how much effort is required to break n instances relative to the effort required to break one: if breaking n users requires at least n times the effort of breaking a single user, then they say the scheme scales optimally.

To give some intuition as to why some schemes might *not* scale optimally, consider again a massive, state-backed agency that wants to break into n users. The users all have unique public/private keypairs, but they all share the same system parameters, describing the underlying algebraic properties of the scheme. The agency may then rather want to spend its immense computational power doing pre-computations on these properties, to prepare for a large-scale attack on the system. Security guarantees that in order for the pre-computations to be helpful, they must be at least as costly as breaking any single user, but once completed may at worst allow them to break any public key, giving a massive return on investment in terms of surveillance capacity. Auerbach et al. study several schemes, giving examples both of schemes with optimal and with suboptimal scaling, as well as some with scaling as poor as in the above example (and even one that paradoxically gets *easier* to break as the number of required targets increases).

However, while claiming to study public-key encryption, Auerbach et al.'s results are exclusively stated for key encapsulation mechanisms, or KEMs. As explained in Sect. 1.5, combined with a secure symmetric-key data encapsulation mechanism (DEM), these make up a secure PKE [72]; this proof of compositional security is also known to tightly hold in the multi-user setting with corruptions, as long as the security notion uses only a single challenge bit [33,52]. With multiple challenge bits, however, the reduction becomes untight, and in the multi-instance setting, this loss appears to turn exponential, making it hard to infer anything about PKEs from the study of KEMs alone.

And so we are led to our third and final research question.

Research Question 3. *How can we bridge the gap left by Auerbach et al. [4] so that their KEM results carry over to the PKE setting?*

Chapter 3

Our Contribution

Article 1 [42] tackles Research Question 1, investigating the relative strengths of formalizations of IND-CCA in the presence of multiple users, with and without corruptions. In particular, we investigate how the choice between one global challenge bit, and one challenge bit per user, affects tightness results. We find that sans corruptions, having a single challenge bits yields a stronger, and therefore preferred, notion. To show this, we provide a tight reduction from single-bit κ-IND-CCA to multi-bit κ-IND-CCA, which is to the best of our knowledge the first non-trivial tight reduction going directly between two notions of multi-user security.

When corruptions are accounted for, this relation breaks down, and none of the two seem to tightly imply the other. We therefore suggest the use of a generalized notion, dubbed “free-bit”, first formalized by Jager et al. [46], and which tightly implies both the single-bit and multi-bit notions.

We make complete maps of known relations between the aforementioned notions with and without corruptions, and for completeness also study how the relation between left-or-right indistinguishability and real-or-random indistinguishability evolves in the multi-bit setting.

Article 2 [18] provides an affirmative answer to Research Question 3 by giving a hybrid PKE construction with tight *inheritance* in the multi-instance setting, i.e. with the property that any characteristics of the underlying KEM is tightly inherited by the PKE. Thus, the results previously shown by Auerbach et al. [4] to hold for KEMs will also hold for our construction when instantiated with said KEMs. The construction is based on the TagKEM framework [1], and in order to achieve tightness we give a novel notion of TagKEM security, called real-or-permuted. We construct a TagXEM (TagKEM with extendable output) tightly achieving (n, κ)-ROP-CCA from a KEM, a Message Authentication Code (MAC), and an Extendable Output Function (XOF) modelled as a programmable random oracle. Finally, we update Auerbach et al.'s analysis of an ElGamal KEM instantiated from the multi-instance GapCDH problem in the generic group model to include corruptions.

Along the way, we give a generalized notion of imperfect correctness called (γ, δ)-correctness, give refined definitions of multi-instance key unrecoverability and indistinguishability, and study their relations in the presence of (γ, δ)-correctness, including real-or-random indistinguishability.

Article 3 [19] sheds light on Research Question 2 by providing a novel systematization of notions of security with corruptions, placing them in a strict hierarchy based on their relative strengths.

The four main notions of IND-CCA, indistinguishability-based SOA, simulatability-based SOA, and non-committing encryption (NCE), are categorized based on whether they are indistinguishability or simulation based, and of the a priori or a posteriori variant, making for four possible philosophies of confidentiality to choose from. Additionally, we consider four types of corruptions, or "opening": transmission opening, in which a ciphertext is opened to reveal the underlying message; sender opening, in which a ciphertext is opened to reveal both the underlying message and the randomness used for encryption; receiver opening, in which a user is "opened" to reveal their private key; and bi-opening, for which all the above are present.

We recall and generalize each of the notions, discuss the potential use cases and limitations of each, re-cast the implications of the hierarchy in a concrete security framework, summarize and contextualize known relations, identify and highlight many open problems, and close a few gaps.

Finally, we identify a simple technique hailing all the way back to Bellare and Rogaway's hallmark paper *Random Oracles are Practical* [11], with which even the strongest notion of the hierarchy, NCE-CCA with bi-opening, can be achieved (in the programmable random oracle model), and survey other constructions known to achieve the various notions, in programmable idealized models, and in the standard model.

Bibliography

[1] Abe, M., Gennaro, R., Kurosawa, K.: Tag-KEM/DEM: A new framework for hybrid encryption. Journal of Cryptology **21**(1), 97–130 (Jan 2008). https://doi.org/10.1007/s00145-007-9010-x

[2] Alagic, G., Apon, D., Cooper, D., Dang, Q., Dang, T., Kelsey, J., Lichtinger, J., Miller, C., Moody, D., Peralta, R., Perlner, R., Robinson, A.: Status report on the third round of the nist post-quantum cryptography standardization process. NIST (7/2022 2022). https://doi.org/https://doi.org/10.6028/NIST.IR.8413

[3] Arora, S., Lund, C., Motwani, R., Sudan, M., Szegedy, M.: Proof verification and hardness of approximation problems. In: 33rd FOCS. pp. 14–23. IEEE Computer Society Press (Oct 1992). https://doi.org/10.1109/SFCS.1992.267823

[4] Auerbach, B., Giacon, F., Kiltz, E.: Everybody's a target: Scalability in public-key encryption. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 475–506. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45727-3_16

[5] Avanzi, R.M., Bos, J.W., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Schwabe, P., Seiler, G., Stehlé, D.: Crystals-kyber algorithm specifications and supporting documentation (2017)

[6] Bader, C., Jager, T., Li, Y., Schäge, S.: On the impossibility of tight cryptographic reductions. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 273–304. Springer, Heidelberg (May 2016). https://doi.org/10.1007/978-3-662-49896-5_10

[7] Bellare, M., Boldyreva, A., Micali, S.: Public-key encryption in a multi-user setting: Security proofs and improvements. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 259–274. Springer, Heidelberg (May 2000). https://doi.org/10.1007/3-540-45539-6_18

[8] Bellare, M., Boldyreva, A., Palacio, A.: An uninstantiable random-oracle-model scheme for a hybrid-encryption problem. In: Cachin, C., Camenisch, J. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 171–188. Springer, Heidelberg (May 2004). https://doi.org/10.1007/978-3-540-24676-3_11

[9] Bellare, M., Kilian, J., Rogaway, P.: The security of cipher block chaining. In: Desmedt, Y. (ed.) CRYPTO'94. LNCS, vol. 839, pp. 341–358. Springer, Heidelberg (Aug 1994). https://doi.org/10.1007/3-540-48658-5_32

[10] Bellare, M., Ristenpart, T., Tessaro, S.: Multi-instance security and its application to password-based cryptography. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 312–329. Springer, Heidelberg (Aug 2012). https://doi.org/10.1007/978-3-642-32009-5_19

[11] Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Denning, D.E., Pyle, R., Ganesan, R., Sandhu, R.S., Ashby, V. (eds.) ACM CCS 93. pp. 62–73. ACM Press (Nov 1993). https://doi.org/10.1145/168588.168596

[12] Bellare, M., Rogaway, P.: Entity authentication and key distribution. In: Stinson, D.R. (ed.) CRYPTO'93. LNCS, vol. 773, pp. 232–249. Springer, Heidelberg (Aug 1994). https://doi.org/10.1007/3-540-48329-2_21

[13] Bellare, M., Rogaway, P.: Optimal asymmetric encryption. In: Santis, A.D. (ed.) EUROCRYPT'94. LNCS, vol. 950, pp. 92–111. Springer, Heidelberg (May 1995). https://doi.org/10.1007/BFb0053428

[14] Bellare, M., Rogaway, P.: Code-based game-playing proofs and the security of triple encryption. Cryptology ePrint Archive, Report 2004/331 (2004), https://eprint.iacr.org/2004/331

[15] Boneh, D.: Simplified OAEP for the RSA and Rabin functions. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 275–291. Springer, Heidelberg (Aug 2001). https://doi.org/10.1007/3-540-44647-8_17

[16] Boneh, D., Dagdelen, Ö., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 41–69. Springer, Heidelberg (Dec 2011). https://doi.org/10.1007/978-3-642-25385-0_3

[17] Boneh, D., Zhandry, M.: Secure signatures and chosen ciphertext security in a quantum computing world. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 361–379. Springer, Heidelberg (Aug 2013). https://doi.org/10.1007/978-3-642-40084-1_21

[18] Brunetta, C., Heum, H., Stam, M.: Multi-instance secure public-key encryption. In: Boldyreva, A., Kolesnikov, V. (eds.) PKC 2023, Part II. LNCS, vol. 13941, pp. 336–367. Springer, Heidelberg (May 2023). https://doi.org/10.1007/978-3-031-31371-4_12

[19] Brunetta, C., Heum, H., Stam, M.: Sok: Public key encryption with openings. Cryptology ePrint Archive, Report 2023/XXX (2023), https://eprint.iacr.org/2023/XXX

[20] Canetti, R., Goldreich, O., Halevi, S.: The random oracle methodology, revisited (preliminary version). In: 30th ACM STOC. pp. 209–218. ACM Press (May 1998). https://doi.org/10.1145/276698.276741

[21] Diffie, W.: Cryptology and security: the view from 2016, https://youtu.be/FXWsmHoFs6Y, accessed: 2023-01-18

[22] Diffie, W., Hellman, M.E.: New directions in cryptography. IEEE Transactions on Information Theory **22**(6), 644–654 (1976). https://doi.org/10.1109/TIT.1976.1055638

[23] Dolev, D., Dwork, C., Naor, M.: Non-malleable cryptography (extended abstract). In: 23rd ACM STOC. pp. 542–552. ACM Press (May 1991). https://doi.org/10.1145/103418.103474

[24] Ellis, J.H.: The possibility of secure non-secret digital encryption (1970)

[25] Ellis, J.H.: The history of non-secret encryption. Cryptologia **23**, 267–273 (1999)

[26] Feynman, R.P.: Quantum mechanical computers. Foundations of Physics **16**, 507–531 (1984)

[27] Fischlin, M., Lehmann, A., Ristenpart, T., Shrimpton, T., Stam, M., Tessaro, S.: Random oracles with(out) programmability. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 303–320. Springer, Heidelberg (Dec 2010). https://doi.org/10.1007/978-3-642-17373-8_18

[28] Gagliardoni, T., Hülsing, A., Schaffner, C.: Semantic security and indistinguishability in the quantum world. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016, Part III. LNCS, vol. 9816, pp. 60–89. Springer, Heidelberg (Aug 2016). https://doi.org/10.1007/978-3-662-53015-3_3

[29] Gagliardoni, T., Krämer, J., Struck, P.: Quantum indistinguishability for public key encryption. In: Cheon, J.H., Tillich, J.P. (eds.) Post-Quantum Cryptography - 12th International Workshop, PQCrypto 2021. pp. 463–482. Springer, Heidelberg (2021). https://doi.org/10.1007/978-3-030-81293-5_24

[30] Gardner, M.: A new kind of cipher that would take millions of years to break (1977)

[31] Garfinkel, S.L., Russell, D.: Pgp: Pretty good privacy (1994)

[32] Gay, R., Hofheinz, D., Kohl, L.: Kurosawa-desmedt meets tight security. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part III. LNCS, vol. 10403, pp. 133–160. Springer, Heidelberg (Aug 2017). https://doi.org/10.1007/978-3-319-63697-9_5

[33] Giacon, F., Kiltz, E., Poettering, B.: Hybrid encryption in a multi-user setting, revisited. In: Abdalla, M., Dahab, R. (eds.) PKC 2018, Part I. LNCS, vol. 10769, pp. 159–189. Springer, Heidelberg (Mar 2018). https://doi.org/10.1007/978-3-319-76578-5_6

[34] Gidney, C., Ekerå, M.: How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits. Quantum **5**, 433 (Apr 2021). https://doi.org/10.22331/q-2021-04-15-433, https://doi.org/10.22331/q-2021-04-15-433

[35] Goldwasser, S., Micali, S.: Probabilistic encryption and how to play mental poker keeping secret all partial information. In: 14th ACM STOC. pp. 365–377. ACM Press (May 1982). https://doi.org/10.1145/800070.802212

[36] Goldwasser, S., Micali, S.: Probabilistic encryption. Journal of Computer and System Sciences **28**(2), 270–299 (1984)

[37] Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof-systems (extended abstract). In: 17th ACM STOC. pp. 291–304. ACM Press (May 1985). https://doi.org/10.1145/22145.22178

[38] Greenwald, G., MacAskill, E., Poitras, L.: Edward snowden: the whistleblower behind the nsa surveillance revelations (video) (2013), https://www.theguardian.com/world/2013/jun/09/edward-snowden-nsa-whistleblower-surveillance, accessed: 2023-08-31

[39] Han, S., Liu, S., Gu, D.: Almost tight multi-user security under adaptive corruptions & leakages in the standard model. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part III. LNCS, vol. 14006, pp. 132–162. Springer, Heidelberg (Apr 2023). https://doi.org/10.1007/978-3-031-30620-4_5

[40] Han, S., Liu, S., Wang, Z., Gu, D.: Almost tight multi-user security under adaptive corruptions from LWE in the standard model. In: Handschuh, H., Lysyanskaya, A. (eds.) Advances in Cryptology – CRYPTO 2023. pp. 682–715. Springer Nature Switzerland, Cham (2023)

[41] Håstad, J.: Solving simultaneous modular equations of low degree. SIAM Journal on Computing **17**(2), 336–341 (1988). https://doi.org/10.1137/0217019, https://doi.org/10.1137/0217019

[42] Heum, H., Stam, M.: Tightness subtleties for multi-user pke notions. In: Paterson, M.B. (ed.) Cryptography and Coding. pp. 75–104. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-92641-0_5

[43] Hofheinz, D., Jager, T.: Tightly secure signatures and public-key encryption. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 590–607. Springer, Heidelberg (Aug 2012). https://doi.org/10.1007/978-3-642-32009-5_35

[44] IBM Corporation: IBM quantum summit 2022 (2022), https://ibm-com-qc-dev.quantum-computing.ibm.com/quantum/summit

[45] Jager, T., Kiltz, E., Riepel, D., Schäge, S.: Tightly-secure authenticated key exchange, revisited. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part I. LNCS, vol. 12696, pp. 117–146. Springer, Heidelberg (Oct 2021). https://doi.org/10.1007/978-3-030-77870-5_5

[46] Jager, T., Stam, M., Stanley-Oakes, R., Warinschi, B.: Multi-key authenticated encryption with corruptions: Reductions are lossy. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 409–441. Springer, Heidelberg (Nov 2017). https://doi.org/10.1007/978-3-319-70500-2_14

[47] Johnson, D.B., Le, A.V., Martin, W., Matyas, S.M., Wilkins, J.D.: Hybrid key distribution scheme giving key record recovery. IBM Technical Disclosure Bulletin **37**(2A), 5–16 (Feb 1994)

[48] Kahn, D.: The Codebreakers – The Story of Secret Writing (1967)

[49] Katsumata, S.: A new simple technique to bootstrap various lattice zero-knowledge proofs to QROM secure NIZKs. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021, Part II. LNCS, vol. 12826, pp. 580–610. Springer, Heidelberg, Virtual Event (Aug 2021). https://doi.org/10.1007/978-3-030-84245-1_20

[50] Kerckhoffs, A.: La cryptographie militaire, ou les chiffres usitÃ©s en temps de guerre, avec un nouveau procÃ©dÃ© de dÃ©chiffrement applicable aux systÃ¨mes Ã double clef. Librairie militaire de L. Baudoin (1883)

[51] Koblitz, N., Menezes, A.J.: Another look at "provable security". Journal of Cryptology **20**(1), 3–37 (Jan 2007). https://doi.org/10.1007/s00145-005-0432-z

[52] Lee, Y., Lee, D.H., Park, J.H.: Tightly cca-secure encryption scheme in a multi-user setting with corruptions. Des. Codes Cryptogr. **88**(11), 2433–2452 (2020)

[53] Levy, S.: The open secret. Wired (1999), https://www.wired.com/1999/04/crypto/, accessed: 2023-01-18

[54] Liu, Q., Zhandry, M.: Revisiting post-quantum Fiat-Shamir. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part II. LNCS, vol. 11693, pp. 326–355. Springer, Heidelberg (Aug 2019). https://doi.org/10.1007/978-3-030-26951-7_12

[55] Machinae.com: Cryptology: Khnumhotep ii (2005 – 2007), http://www.machinae.com/crypto/khnumhotep.html, accessed: 2023-01-18

[56] Merkle, R.: Secure communication over insecure channels. Commun. ACM **21**, 294–299 (04 1978). https://doi.org/10.1145/359460.359473

[57] Micali, S., Rackoff, C., Sloan, B.: The notion of security for probabilistic cryptosystems. In: Odlyzko, A.M. (ed.) CRYPTO'86. LNCS, vol. 263, pp. 381–392. Springer, Heidelberg (Aug 1987). https://doi.org/10.1007/3-540-47721-7_27

[58] Mouha, N.: Sha–3 buffer overflow (2022), https://mouha.be/sha-3-buffer-overflow/, accessed: 2023-01-18

[59] Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: 22nd ACM STOC. pp. 427–437. ACM Press (May 1990). https://doi.org/10.1145/100216.100273

[60] National Institute of Standards and Technology: pqc-forum, https://groups.google.com/a/list.nist.gov/g/pqc-forum?pli=1, accessed: 2023-01-18

[61] National Institute of Standards and Technology: Submission requirements and evaluation criteria for the post-quantum cryptography standardization process (2016), https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf, accessed: 2023-01-18

[62] Nielsen, J.B.: Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 111–126. Springer, Heidelberg (Aug 2002). https://doi.org/10.1007/3-540-45708-9_8

[63] Rabin, M.O.: Digital signatures and public key functions as intractable as factorization. Technical Report MIT/LCS/TR-212, Massachusetts Institute of Technology (Jan 1979)

[64] Rackoff, C., Simon, D.R.: Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In: Feigenbaum, J. (ed.) CRYPTO'91. LNCS, vol. 576, pp. 433–444. Springer, Heidelberg (Aug 1992). https://doi.org/10.1007/3-540-46766-1_35

[65] Rivest, R.L., Shamir, A., Adleman, L.M.: A method for obtaining digital signatures and public-key cryptosystems. Communications of the Association for Computing Machinery **21**(2), 120–126 (Feb 1978). https://doi.org/10.1145/359340.359342

[66] Rogaway, P.: The moral character of cryptographic work. Cryptology ePrint Archive, Report 2015/1162 (2015), https://eprint.iacr.org/2015/1162

[67] Rogaway, P.: Practice-oriented provable security and the social construction of cryptography. IEEE Security & Privacy **14**, 10–17 (2016)

[68] RSA Laboratories: PKCS#1: RSA encryption standard, version 1.5 (1993)

[69] Shamir, A., Rivest, R.L., Adleman, L.M.: Mental Poker, pp. 37–43. Springer US, Boston, MA (1981). https://doi.org/10.1007/978-1-4684-6686-7_5, https://doi.org/10.1007/978-1-4684-6686-7_5

[70] Shannon, C.E.: Communication theory of secrecy systems. Bell Systems Technical Journal **28**(4), 656–715 (1949)

[71] Shor, P.W.: Algorithms for quantum computation: Discrete logarithms and factoring. In: 35th FOCS. pp. 124–134. IEEE Computer Society Press (Nov 1994). https://doi.org/10.1109/SFCS.1994.365700

[72] Shoup, V.: Using hash functions as a hedge against chosen ciphertext attack. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 275–288. Springer, Heidelberg (May 2000). https://doi.org/10.1007/3-540-45539-6_19

[73] Shoup, V.: OAEP reconsidered. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 239–259. Springer, Heidelberg (Aug 2001). https://doi.org/10.1007/3-540-44647-8_15

[74] Shoup, V.: Sequences of games: a tool for taming complexity in security proofs. Cryptology ePrint Archive, Report 2004/332 (2004), https://eprint.iacr.org/2004/332

[75] Singh, S.: The Code Book: The Science of Secrecy from Ancient Egypt to Quantum Cryptography. Fourth Estate, Doubleday (1999)

[76] Watanabe, Y., Shikata, J., Imai, H.: Equivalence between semantic security and indistinguishability against chosen ciphertext attacks. In: Desmedt, Y. (ed.) PKC 2003. LNCS, vol. 2567, pp. 71–84. Springer, Heidelberg (Jan 2003). https://doi.org/10.1007/3-540-36288-6_6

[77] Wu, G., Guo, F., Susilo, W.: Generalized public-key cryptography with tight security. Information Sciences **504**, 561–577 (2019). https://doi.org/https://doi.org/10.1016/j.ins.2019.07.041

[78] Yang, R., Lai, J., Huang, Z., Au, M.H., Xu, Q., Susilo, W.: Possibility and impossibility results for receiver selective opening secure PKE in the multi-challenge setting. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020, Part I. LNCS, vol. 12491, pp. 191–220. Springer, Heidelberg (Dec 2020). https://doi.org/10.1007/978-3-030-64837-4_7

[79] Yost, J.R.: An interview with Martin Hellman. Charles Babbage Institute, Center for the History of Information Technology, University of Minnesota, Minneapolis (Nov 2004)

[80] Zhandry, M.: How to record quantum queries, and applications to quantum indifferentiability. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part II. LNCS, vol. 11693, pp. 239–268. Springer, Heidelberg (Aug 2019). https://doi.org/10.1007/978-3-030-26951-7_9

Article I

Tightness Subtleties for Multi-user PKE Notions

Hans Heum and Martijn Stam

In Maura Paterson, editor, *Cryptography and Coding, IMACC 2021*, pages 75–104. Springer, Cham, December 2021. Presented by the candidate at IMACC in December 2021.

Version reproduced here in Cryptology ePrint Archive, Report 2022/658, 2022.
`https://eprint.iacr.org/2022/855`

Research, writing, and editing was split equally between candidate and supervisor.

Tightness Subtleties for Multi-user PKE Notions[*]

Hans Heum and Martijn Stam

Simula UiB,
Merkantilen (3rd floor)
Thormøhlensgate 53D
N-5006 Bergen, Norway.
{hansh,martijn}@simula.no

Abstract. Public key encryption schemes are increasingly being studied concretely, with an emphasis on tight bounds even in a multi-user setting. Here, two types of formalization have emerged, one with a single challenge bit and one with multiple challenge bits. Another modelling choice is whether to allow key corruptions or not. How tightly the various notions relate to each other has hitherto not been studied in detail. We show that in the absence of corruptions, single-bit left-or-right indistinguishability is the preferred notion, as it tightly implies the other (corruption-less) notions. However, in the presence of corruptions, this implication no longer holds; we suggest the use of a more general notion that tightly implies both existing options. Furthermore, for completeness we study how the relationship between left-or-right versus real-or-random evolves in the multi-user PKE setting.

Keywords: Indistinguishability · Public Key Encryption · Multi-User Security · Adaptive Corruptions

[*] This article is the full author version of a paper appearing at IMA Coding and Cryptography Conference 2021, published in Springer's LNCS. The final authenticated version is available online at https://doi.org/10.1007/978-3-030-92641-0_5.

1 Introduction

Historically, a primitive like public key encryption (PKE) is often studied in a setting where a single key-pair is generated for an adversary to attack, often based on a single challenge ciphertext only [27]. Yet, in reality there will be many users, each generating their own key pairs, to be used repeatedly. To study the concrete security risk of having very many keys in play simultaneously, Bellare et al. [5] introduced the multi-user setting. They considered an adversary with access to n different public keys and the ability to challenge (in an indistinguishability fashion) each of them, and concluded that the security loss is at worst linear in the total number challenge queries. Loosely speaking, such a linear security loss implies that a scheme that is believed to offer, say, 128-bit security in the single user setting, may only guarantee 80-bit security if there are are 2^{20} users each receiving 2^{28} messages (based on the same hardness assumption).

Unfortunately, there have been ample examples of schemes where practical attacks can indeed exploit the increased attack surface, demonstrating that these theoretical security losses can be realized. Consequently, the generic tightness losses to move from a single-user, single-challenge setting to a more realistic multi-user, multi-challenge setting are problematic as, conservatively, one would have to increase key sizes to compensate. Alternatively, a growing number of works have looked at schemes with tighter security guarantees, either if the number of users goes up, the number of challenge encryptions per key goes up, or both [2, 5, 12, 16, 21, 22, 28].

Moreover, in a system with many users, it is not inconceivable that some private keys eventually become available to an adversary, which can be modelled using key corruptions. An adversary learning a private key can obviously decrypt all ciphertexts that were encrypted under the corresponding public key, thus some care has to be taken to avoid trivial wins when allowing key corruptions. The two simplest mechanisms are either using independent challenge bits for each key or disallowing an adversary to both challenge and corrupt a single key. As we detail in Appendix A, both these mechanisms have been used, also in related contexts such as key encapsulation mechanisms (KEMs), authenticated encryption (AE), and authenticated key exchange (AKE), raising the inevitable question which notion should be the preferred one.

In the context of lower bounding tightness losses for multi-user AE, Jager et al. [25] employed a novel multi-key, multi-challenge-bit notion that generalizes both mechanisms; however, the main motivation of this generalized mechanism was universality of their impossibility result, allowing them to side-step the question which mechanism to focus on. Recently, in the context of AKE, Jager et al. [24] argued in favour of the single-bit notion, primarily as it composes more easily. For KEMs a similar argument holds, yet for PKE composition is arguably less relevant. Instead, a more direct interpretation of what the various notions entail might well be preferable.

Our contribution. Both the single-bit and multi-bit approaches are implied by the single user notion at the cost of a tightness loss linear in the number of users. Consequently, the two multi-user notions are also within that linear factor in the number of user. As our goal is to avoid such tightness losses, we are interested in identifying the most suitable, general notion as possible, guaranteeing that there are no "hidden" linear losses in the choice of notion—an issue already pointed to by Jager et al. [24]

To this end, we adapt the multi-key, multi-bit notion of Jager et al. [25] to the PKE setting, slightly generalizing it in the process. We show how it tightly implies, and therefore unifies, the previous multi-user notions, and give novel interpretations of each (see Section 3).

We then shift our focus to how tightly the different notions relate to each other, with the goal of identifying the strongest, and therefore preferred, multi-user notions. We find that the answer depends on whether or not corruptions are present: in the absence of corruptions, we find that the single-challenge-bit notion is *as strong or stronger* than any of the other (see Section 4.3). Given that this notion is significantly simpler than the fully general game, this makes the single-bit notion the preferred one in the absence of corruptions. With corruptions, this relation breaks down, and the general "free-bit" game indeed seems the stronger, and therefore preferred, notion (see Section 4.4).

Finally, we fill some holes largely left as folklore until now regarding how the well-known factor-2 reduction from real-or-random to left-or-right indistinguishability, as shown by Bellare et al. [7] for the single-user, single-challenge setting, generalizes to the multi-user setting. We find that, as expected, the relation remains intact in the single-bit setting, regardless of whether corruptions are present (see Section 4.5). In contrast, with multiple challenge bits the best-known reductions turn lossy. Whether these losses are inevitable remains open; however, it reinforces the by now established notion that left-or-right indistinguishability is to be preferred over its real-or-random counterpart whenever possible.

The appendices provide some additional material: highlights include Appendix A giving context to the present work by presenting a more complete history of multi-user indistinguishability than that presented here, and Appendix C, illustrating the difficulty of achieving tight composition in multi-bit settings, as alluded to by

Jager et al. [24], by giving an overview of how additional losses can appear in PKE schemes built through the widely adopted KEM/DEM paradigm.

2 Preliminaries

2.1 Notation.

For an integer n, we will write $[n]$ for the set $\{1, \ldots, n\}$. We will also use the abbreviation $\mathtt{X} \overset{\cup}{\longleftarrow} x$ for the operation $\mathtt{X} \leftarrow \mathtt{X} \cup \{x\}$. The event of an adversary $\mathbb{A}$ outputting 0 is denoted $0 \leftarrow \mathbb{A}$. We use $\Pr[Code\ :\ Event\ |\ Condition]$ to denote the conditional probability of $Event$ occuring when $Code$ is executed, conditioned on $Condition$. We omit $Code$ when it is clear from the context and $Condition$ when it is not needed.

2.2 PKE Syntax

A public key encryption scheme PKE consists of three algorithms: the probabilistic *key generation* algorithm $\mathsf{Pk.Kg}$, which takes as input some system parameter pm and outputs a public/private key pair $(\mathsf{pk}, \mathsf{sk}) \in (\mathcal{PK}, \mathcal{SK})$; the probabilistic *encryption* algorithm $\mathsf{Pk.Enc}$, which on input a public key $\mathsf{pk} \in \mathcal{PK}$ and a message $m \in \mathcal{M}$, outputs a ciphertext c; and the deterministic *decryption* algorithm $\mathsf{Pk.Dec}$, which on input of a secret key $\mathsf{sk} \in \mathcal{SK}$ and a ciphertext c, outputs either the message m, or a special symbol $\perp$ denoting failure.

We allow the message space $\mathcal{M}$ to depend on the parameters pm, but insist it is independent of the public key pk. We furthermore assume that there exists an equivalence relation $\sim$ on the message space that partitions $\mathcal{M}$ into finite equivalence classes. For $m \in \mathcal{M}$, we let $[\![m]\!]$ denote its equivalence class, so $[\![m]\!] = \{\tilde{m} \in \mathcal{M} : m \sim \tilde{m}\}$. Often $\mathcal{M}$ consists of arbitrary length bitstrings, or at least all bitstrings up to some large length (e.g. 2^{64}), and two messages are equivalent iff they have the same length, so $[\![m]\!] = \{0,1\}^{|m|}$; for other cryptosystems, such as ElGamal, messages are group elements that are essentially all equivalent, so $[\![m]\!] = \mathcal{M}$. (Note that the case where $[\![m]\!] = \{m\}$ for all m is degenerate and ‘security’ is often trivially satisfied.)

The scheme must satisfy ϵ-correctness [20], namely that for any pm:

$$\mathbb{E}_{(\mathsf{pk},\mathsf{sk}) \leftarrow\!\!\$\, \mathsf{Pk.Kg}(\mathsf{pm})} \left[\max_{m \in \mathcal{M}} \Pr[c \leftarrow\!\!\$\, \mathsf{Pk.Enc}_{\mathsf{pk}}(m))\ :\ \mathsf{Pk.Dec}_{\mathsf{sk}}(c) \neq m] \right] \leq \epsilon\,.$$

If $\epsilon = 0$ we speak of perfect correctness; the case $\epsilon > 0$ is especially useful to model decryption errors typical to lattice-based schemes.

Remark 1. The system parameters pm are implicitly input to $\mathsf{Pk.Enc}$ and $\mathsf{Pk.Dec}$ as well; for concreteness, they can for instance be the description of an elliptic curve group with generator for an ECDLP-based system or the dimensions and noise sampling algorithm for an LWE-based system. When one is interested in re-phrasing our results in an asymptotic setting, the parameters pm will be generated by a probabilistic, polynomial-time algorithm that only takes the security parameter as input.

2.3 Concrete Security

Indistinguishability. The standard notion of security for encryption systems has become that of indistinguishability. Here the adversary is given access to a challenge encryption oracle implementing one of two “worlds”; the adversary needs to find out which. Several choices appear regarding the exact nature of these worlds, leading to different notions of indistinguishability such as real-or-random and left-or-right. Henceforth we refer to those two notions ROR and LOR, respectively, and we will refer to them collectively as IND. We will flesh out the details in Section 3.

Security definitions furthermore take into account the POWER given to the adversary, for example that of chosen plaintext attacks (CPA), or chosen ciphertext attacks (CCA). The distinguishing advantage of an adversary $\mathbb{A}$ against a scheme relative to some notion will then be $\mathrm{IND\text{-}POWER}_{\mathrm{PKE}}(\mathbb{A})$, see Definition 1. As randomly guessing a world is correct half of the time, the distinguishing advantage is of course suitably offset.

Definition 1. *The* distinguishing advantage *of an adversary* $\mathbb{A}$ *against an encryption scheme* PKE *is*

$$\mathrm{IND\text{-}POWER}_{\mathrm{PKE}}(\mathbb{A}) := 2 \cdot \Pr\left[\mathsf{Exp}^{\mathrm{IND\text{-}POWER}}_{\mathrm{PKE}}(\mathbb{A}) = 1\right] - 1\,.$$

Implications and separations. Our main focus will be comparing different notions of security, especially showing that if security is met under one notion, then it is also met under another one. We will prove these implication using fully black box reductions [4, 31] that are furthermore simple [29]. A fully black box reduction works for all schemes and adversaries, and only accesses them in a black box manner. Moreover, if the reduction only runs its adversary once and without rewinding, then the reduction is simple.

To allow for black-box access to the scheme, we will add an auxiliary oracle for the PKE to operate on the message space and the key space. A simple fully-black box (SFBB) reduction has access to this auxiliary oracle, as well as to the oracles corresponding to the PKE's algorithms, the oracles provided to the reduction by the game it is playing, and finally its single straight copy of the adversary. We will insist that the overhead of such a reduction, namely the number of oracle calls it makes more than the adversary it is running, is not undue: it can be upper bounded in terms of the parameters that define the security game(s) at hand, such as the number of keys in the system.

Definition 2 (Tightness). *Let* IND_1 *and* IND_2 *be two indistinguishability notions for PKE schemes, let* c *be a positive real number, then* $\mathrm{IND}_1 \overset{\leqslant c}{\Longrightarrow} \mathrm{IND}_2$ *iff there exists a simple fully-black box reduction* $\mathbb{B}_1$ *such that for all PKE schemes* PKE *and adversaries* $\mathbb{A}_2$*,*

$$\mathrm{IND}_2(\mathbb{A}_2) \leq c \cdot \mathrm{IND}_1(\mathbb{B}_1^{\mathbb{A}_2,\mathrm{PKE}})$$

and the overhead of $\mathbb{A}_2$ *is not undue.*

Refer also to Jager et al. [25] for a discussion on how to express tightness for more general reductions. They also formalize the folklore that simple reductions compose neatly; in our case if $\mathrm{IND}_1 \overset{\leqslant c}{\Longrightarrow} \mathrm{IND}_2$ and $\mathrm{IND}_2 \overset{\leqslant d}{\Longrightarrow} \mathrm{IND}_3$ then also $\mathrm{IND}_1 \overset{\leqslant c \cdot d}{\Longrightarrow} \mathrm{IND}_3$.

If $c = 1$, the reduction is called tight; if $c > 1$ we call the reduction lossy. Note that our notion of tightness is stricter than in some other works where a constant factor of say 2 will still be considered tight [18]; our convention has the benefit of not depending on any (security) parameter. A natural question for lossy reductions is whether the loss is inevitable or not—if it is, the bound is called sharp. Questions of sharpness are not the focus of our work, although we do remark upon it in more detail in Appendix B.

3 A General Definition of PKE Multi-User Security

3.1 A General Game

In order to compare various flavours of multi-user notions for PKE, we take Jager et al.'s framework for multi-user AE notions [25] and port it to the PKE setting, using some slightly different game-mechanics in the process. A multi-user security game is parametrized by the number of keys κ and the number of bits β. Usually one can imagine $\beta \leq \kappa$ and in fact Jager et al. only considered $\beta = \kappa$. However, keeping κ and β distinct helps when expressing and interpreting security losses.

Given a public key encryption scheme PKE, let $\mathsf{Exp}_{\mathrm{PKE}}^{\text{IND-CCA},\not\kappa,\beta}(\mathbb{A})$ be the experiment given in Fig. 1, where $\mathbb{A}$ is the adversary. The corresponding distinguishing advantage (see Definition 1) is denoted by $\text{IND-CCA}_{\mathrm{PKE}}^{\not\kappa,\beta}(\mathbb{A})$. The κ is slashed to denote the presence of a key corruption oracle; the corresponding notion without corruptions is $\text{IND-CCA}_{\mathrm{PKE}}^{\kappa,\beta}$. Without the decryption oracle the notion becomes a chosen-plaintext attack (CPA) instead. Often our results are oblivious of whether the power is CPA or CCA; we will then use CXA to refer to them collectively.

In the game, an adversary is given κ public keys, and a choice of β bits to try and attack through one of the two challenge oracles depending on the flavour of indistinguishability: for left-or-right indistinguishability, it gains access to $\mathcal{E}_{\mathrm{LOR}}$, whereas for real-or-random, it instead gains access to $\mathcal{E}_{\mathrm{ROR}}$. Both oracles have the usual interface, augmented by a key handle i and a bit handle j. For instance, for $\mathcal{E}_{\mathrm{LOR}}$ an adversary picks handles i and j as well as two equivalent messages m_0 and m_1 to receive the encryption of m_{b_j} under public key pk_i. For $\mathcal{E}_{\mathrm{ROR}}$ only a single message m is provided in addition to the two handles and, depending on the value of b_j, $\mathbb{A}$ receives the encryption of either the message or of a uniformly chosen equivalent message.

The adversary has possible access to two additional powers: a decryption oracle $\mathcal{D}$ and a corruption oracle $\mathcal{K}$. The former takes as input a ciphertext c together with a key handle i, and returns the decryption of c under private key sk_i. The latter takes as input a key handle i and directly returns said sk_i.

The adversary has in principle unlimited adaptive access to the available oracles, necessitating some admin in the game to deal with trivial wins. Firstly, if $m_0 \not\sim m_1$ for $\mathcal{E}_{\mathrm{LOR}}$, or if a challenge ciphertext is submitted

```
$\mathsf{Exp}^{\mathrm{IND\text{-}CCA},\not\kappa,\beta}_{\mathrm{PKE}}(\mathbb{A})$
    $(\mathsf{pk}_1,\mathsf{sk}_1),\ldots,(\mathsf{pk}_\kappa,\mathsf{sk}_\kappa) \leftarrow\$ \mathsf{Pk.Kg}$
    $b_1,\ldots,b_\beta,\delta \leftarrow\$ \{0,1\}$
    $\mathtt{C}_1,\ldots,\mathtt{C}_\kappa,\mathtt{I}^{\mathcal{E}}_1,\ldots,\mathtt{I}^{\mathcal{E}}_\beta,\mathtt{I}^{\mathcal{K}} \leftarrow \emptyset$
    $(j,\hat{b}_j) \leftarrow\$ \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{K}}(\mathsf{pk}_1,\ldots,\mathsf{pk}_\kappa)$
    if $\mathtt{I}^{\mathcal{E}}_j \cap \mathtt{I}^{\mathcal{K}} \neq \emptyset$ then return $\delta$
    else return $b_j = \hat{b}_j$
```

```
$\mathcal{E}_{\mathrm{LOR}}(i,j,m_0,m_1)$
    if $m_0 \not\sim m_1$ then return $\lightning$
    $\mathtt{I}^{\mathcal{E}}_j \xleftarrow{\cup} i$
    $c^* \leftarrow\$ \mathsf{Pk.Enc}_{\mathsf{pk}_i}(m_{b_j})$
    $\mathtt{C}_i \xleftarrow{\cup} c^*$
    return $c^*$

$\mathcal{E}_{\mathrm{ROR}}(i,j,m)$
    $m' \leftarrow\$ [\![m]\!]$
    return $\mathcal{E}_{\mathrm{LOR}}(i,j,m,m')$
```

```
$\mathcal{D}(i,c)$
    if $c \in \mathtt{C}_i$ then return $\lightning$
    $m \leftarrow \mathsf{Pk.Dec}_{\mathsf{sk}_i}(c)$
    return $m$

$\mathcal{K}(i)$
    $\mathtt{I}^{\mathcal{K}} \xleftarrow{\cup} i$
    return $\mathsf{sk}_i$
```

Fig. 1: The generalised multi-user distinguishing experiment $\mathsf{Exp}^{\mathrm{IND\text{-}CCA},\not\kappa,\beta}_{\mathrm{PKE}}(\mathbb{A})$; the adversary has access to either the left-or-right $\mathcal{E}_{\mathrm{LOR}}$ or the real-or-random $\mathcal{E}_{\mathrm{ROR}}$ challenge oracle.

to the decryption oracle under its handle of creation, then the adversary receives the special symbol $\lightning$ instead. Secondly, once the adversary outputs a bit handle j and a guess $\hat{b}_j$, the game checks through $\mathtt{I}^{\mathcal{E}}_j \cap \mathtt{I}^{\mathcal{K}} = \emptyset$ whether the challenge bit has become compromised by virtue of being challenged together with a corrupted key. If so, the game outputs the uniformly random bit δ, yielding the adversary no advantage; otherwise, the game outputs whether $\hat{b}_j = b_j$.

Unlike Jager et al., we do not consider valid or invalid adversaries, but rather deal with bad behaviour in-game. Specifically, we want the adversary to be able to challenge on a key both before and after it becomes corrupted, but trying to win by attacking any of the corrupted challenge bits must of course be disallowed, regardless of the order of the queries. Thus, for problematic combinations of challenge/corrupt/target we necessarily had to wait until the adversary announced its target j before, if need be, penalizing. For bad decryption queries, penalizing at the end is discouraged [8], moreover it is easy to check on-the-fly.

Finally, we use $q^{\mathcal{E}}_i$ to refer to the number of challenge queries on public key pk_i; $q^{\mathcal{E}}_{\Sigma}$ for the total number of challenge oracle calls; and $q^{\mathcal{E}}_{\max}$ for the maximum number of challenge queries per key. Similarly, $q^{\mathcal{D}}_i$ is the number of decryption calls on private key sk_i and $q^{\mathcal{K}}$ the number of corruption calls.

3.2 Notational Conventions

Jager et al. [25] introduced their unified game in order to show that, for authenticated encryption, tightness losses are inevitable in a multi-key with corruption setting, irrespective of certain definitional choices. Thus they can avoid having to choose one notion over the other. We are interested in finding out, for public key encryption, whether some notion is preferred over the other. To that end, we will introduce some notation to more easily identify known notions and express relationships between them.

One can visualize the IND-CXA$^{\not\kappa,\beta}$ experiment using a binary matrix of dimension $\kappa \times \beta$, where an entry be set wherever a key and a bit may be called together. For the general game, the matrix has every entry filled (see the leftmost matrix of Fig. 2). We will refer to this as the free-bit notion. By restricting the matrix, we can easily express existing notions.

Bellare et al.'s original single-challenge-bit notion [5] corresponds to a $\kappa \times \beta$-matrix (for arbitrary β) with only a single set row to force all challenge queries to the same bit handle (see the middle matrix of Fig. 2). If $\beta = 1$, the notion matches the free-bit notion, so we may write IND-CXA$^{\kappa,1}$, or IND-CXA$^{\not\kappa,1}$ if corruptions are present, for the single-bit notion.

On the other hand, for the one-challenge-bit-per-key notion we have that $\beta = \kappa$ and the restriction $i = j$ for all challenge queries. These restrictions correspond to a square matrix in which only the diagonal is set (see the rightmost matrix of Fig. 2), inspiring us to refer to this notion as *diagonal-bit*, or just diagonal, and denote it by IND-CXA$^{\kappa,\boxbslash}$, or IND-CXA$^{\not\kappa,\boxbslash}$ with corruptions.

The single-bit and diagonal-bit notions we will collectively refer to as the simple notions. Our notation and terminology differs from prior art, which is to some extent inevitable. The distinction between the various notions has only recently received explicit attention [24, 25] and no clear terminology has yet been set. For instance, we drop the prefix MU (for multi-user, to contrast with the older single user notions) as on the one hand we believe that these days multi-user security should be the default from which single user notions can be derived if needed, and on the other hand we wish to maintain a clean GOAL–POWER nomenclature: having multiple users to target primarily modifies an adversary's power, not its goal.

$$\begin{array}{c} \\ \mathsf{pk}_1 \\ \mathsf{pk}_2 \\ \mathsf{pk}_3 \\ \vdots \\ \mathsf{pk}_\kappa \end{array} \begin{array}{c} \begin{array}{ccccc} b_1 & b_2 & b_3 & \dots & b_\beta \end{array} \\ \begin{pmatrix} \circ & \circ & \circ & \dots & \circ \\ \circ & \circ & \circ & \dots & \circ \\ \circ & \circ & \circ & \dots & \circ \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \circ & \circ & \circ & \dots & \circ \end{pmatrix} \\ \mathrm{IND}^{\kappa,\beta} \end{array} \qquad \begin{array}{c} \begin{array}{ccccc} b_1 & b_2 & b_3 & \dots & b_\beta \end{array} \\ \begin{pmatrix} \circ & \times & \times & \dots & \times \\ \circ & \times & \times & \dots & \times \\ \circ & \times & \times & \dots & \times \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \circ & \times & \times & \dots & \times \end{pmatrix} \\ \mathrm{IND}^{\kappa,1} \end{array} \qquad \begin{array}{c} \begin{array}{ccccc} b_1 & b_2 & b_3 & \dots & b_\kappa \end{array} \\ \begin{pmatrix} \circ & \times & \times & \dots & \times \\ \times & \circ & \times & \dots & \times \\ \times & \times & \circ & \dots & \times \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \times & \times & \times & \dots & \circ \end{pmatrix} \\ \mathrm{IND}^{\kappa,\boxbslash} \end{array}$$

Fig. 2: Matrices of allowed key/bit combinations in challenge oracle calls for the free-bit, single-bit, and diagonal-bit multi-user notion, respectively; circles mark allowed queries, while crosses mark disallowed ones. The visualization highlights that the free-bit notion is a strict generalization of the two other, simple notions.

3.3 Interpretation

Both simple notions with corruptions have appeared in the literature, both in a PKE setting but also in related KEM, AKE, and to a lesser extent AE settings. One key question is which notion to opt for when. Establishing relationships between the notions, as in the next section, helps answer this question. Here, we want to address the meaning and usefulness of the notions as they are.

In the context of AKE, Jager et al. [24] discuss the difference between the single-bit notion ("single-bit guess") and the diagonal notion ("multi-bit guess"). Earlier works on tight security for AKE focused on the diagonal setting [2], yet as Cohn-Gorden et al. [13, Section 3] point out, that notion does not lend itself very well for tight composition: when the keys produced by an AKE are subsequently used, in a proof it is convenient to swap out all keys from real to random in one fell swoop. The single-bit notion allows such a massive substitution, but the diagonal notion does not. Moreover, Jager et al. wonder whether the diagonal notion is meaningful, which would "provide a good intuition of what [it] tries to model".

Whereas AKE and KEMs are primarily tools to set up symmetric keys for subsequent use, the situation for PKE is different as it is much closer to the end user. The difference is reflected in the kind of indistinguishability as well: for AKE and KEMs, a ROR-style notion is used where the adversary cannot even control the real world's "message", yet for PKE's LOR-notion, an adversary has full control over the left-versus-right challenge messages. Thus, for PKE the diagonal-LOR notion does seem meaningful, as we explain below.

Suppose we interpret each key to correspond to a *user* and each challenge bit to correspond to a *conversation*. Then the different notions model different scenarios. For instance, the diagonal notion models a scenario where the users take part in independent conversations, and an adversary can decide which honest conversation to target after corrupting a number of other ones. In contrast, the single-bit notion models a scenario where all users are engaged in the *same* conversation. The latter scenario allows an adversary to accumulate information on the conversation across users, although none of the active parties may be corrupted. Finally, the free-bit notion models a situation where there are a number of independent conversations, each with their own potentially overlapping set of users. The adversary can adaptively corrupt a number of users, and finally targets a conversation conducted by honest users only.

Of course, there are already existing notions that study PKE security in the presence of corruptions, under the term "selective opening attacks" (SOA, [9, 15]). There are various formalizations of SOA, the most relevant ones to our work are receiver SOA [19] where an adversary can corrupt private keys (as opposed to sender SOA, where an adversary learns how a ciphertext was created). Most of these SOA notions are considerably stronger than the notions we consider: our strongest notion is still implied by the customary single-user single-challenge LOR–CCA (just rather lossy), yet for SOA strong separations, and in some cases impossibility results, are known [23]. The link between multi-user security with corruptions on the one hand and SOA on the other has largely been ignored and appears worth expanding further.

We remark that the multi-bit notion also occurs naturally when studying multi-instance security [10], which has been studied in the context of PKE [1]. We leave the adaptation of our work, and specifically the general free-bit game to that setting as an enticing open problem.

Table 1: A modular framework for multi-user security notions.

Shorthand	Stand-in for	Relates to
IND	$\{\text{LOR}, \text{ROR}\}$	Type of challenge oracle
CXA	$\{\text{CPA}, \text{CCA}\}$	Presence of decryption oracle
u (“users”)	$\{\kappa, \not\kappa\}$	Number of keys; presence of corruption oracle
c (“conversations”)	$\{1, \boxbslash, \beta\}$	Number of challenge bits; relation with keys

4 Relations between Indistinguishability Notions

In this section we investigate how tightly the various multi-user notions relate to each other, and how each relate to single-user notions. Some implications are known or folklore and others follow quite naturally from the literature, but not all. As expected, most of the notions are equivalent within a factor linear in the number of users. Yet, some notions turn out to be more, or less, tightly related.

There is for instance the surprising and completely tight reduction from $\text{LOR-CXA}^{\kappa,1}_{\text{PKE}}$ to $\text{LOR-CXA}^{\kappa,\beta}_{\text{PKE}}$ (Theorem 3). However, the proof technique breaks down for real-or-random indistinguishability and in notions with corruptions. Furthermore, for the latter, there doesn’t seem to be a way of relating the notions more tightly than by a linear loss. We conjecture this linear loss to be sharp, yet proving so we leave open.

Shorthand for unified implications. Given the large number of notions resulting from the various orthogonal definitional choices, we use shorthand, as presented in Table 1, to state various theorems. The shorthand serves as an implicit quantifier, so a theorem statement in shorthand essentially holds for all notions included in the shorthand. To avoid clutter, we will sometimes abbreviate $\text{IND-CXA}^{u,c}$ to just $\text{IND}^{u,c}$, and let it be implied that the result holds for both CPA and CCA. We will refer to single-user, multi-challenge notions by dropping the superscripts, e.g. IND.

As a concrete example, consider the trivial statement

$$\text{IND}^{u,c} \implies \text{IND}\,.$$

This is then to be read as, “Both in the cpa and the cca setting, and regardless of the nature of the challenge oracle, the presence or absense of corruptions, or the number and structure of the challenge bits, security under a multi-user notion implies security under the corresponding single-user notion.” Written out in full, the statement becomes:

Lemma 1. *For all* $\text{IND} \in \{\text{LOR}, \text{ROR}\}$*,* $\text{CXA} \in \{\text{CPA}, \text{CCA}\}$*,* $u \in \{\kappa, \not\kappa\}$*, and* $c \in \{1, \boxbslash, \beta\}$*, there is an SFBB reduction* $\mathbb{B}$ *such that, for every adversary* $\mathbb{A}$*,*

$$\text{IND-CXA}_{\text{PKE}}(\mathbb{A}) \leq \text{IND-CXA}^{u,c}_{\text{PKE}}(\mathbb{B})\,.$$

4.1 Tight Implications from Strict Generalizations

Some experiments can easily be seen by inspection to be generalizations of others, in the sense that an adversary’s task has only been made easier. As a well known example, CCA-security tightly implies CPA-security, as the added decryption oracle can only help an adversary. For completeness, we summarize these trivial yet tight implications below.

First, left-or-right indistinguishability is strictly more general than real-or-random indistinguishability, as an adversary playing the former game can simulate the latter with only a small overhead required for the sampling of the random messages.

Lemma 2 **(**$\text{LOR} \implies \text{ROR}$**).** *There is an SFBB reduction* $\mathbb{B}$ *such that, for every adversary* $\mathbb{A}$*,*

$$\text{ROR-CXA}^{u,c}_{\text{PKE}}(\mathbb{A}) \leq \text{LOR-CXA}^{u,c}_{\text{PKE}}(\mathbb{B})\,,$$

where $\mathbb{B}$*’s overhead consists of sampling at most* $q^{\mathcal{E}}_{\Sigma}$ *messages.*

Proof (sketch). Upon receiving m from $\mathbb{A}$, $\mathbb{B}$ lets $m_0 \leftarrow m$, and then samples a random message m_1 from the equivalence class $[\![m]\!]$, before forwarding to its own challenge oracle; all other oracles are forwarded (see also Fig. 1).

$$\begin{tikzcd}
\mathrm{IND}^{\not\kappa,1} \arrow[d, Rightarrow] & & \mathrm{IND}^{\not\kappa,\boxslash} \arrow[d, Rightarrow] \\
\mathrm{IND}^{\kappa,1} \arrow[dr, Rightarrow, bend left] & & \mathrm{IND}^{\kappa,\boxslash} \arrow[dl, Rightarrow, bend right] \\
& \mathrm{IND} \arrow[uul, bend right, "\leq\kappa"'] \arrow[uur, bend left, "\leq\kappa"] \arrow[ul, bend left, "\leq\kappa"] \arrow[ur, bend right, "\leq\kappa"'] &
\end{tikzcd}$$

Fig. 3: Known relations between single-user (but multi-challenge) indistinguishability and the two different generalisations to multi-user indistinguishability, with and without corruptions; refer to Table 1 for an overview of the shorthand. Recall that IND without any superscripts means single-user (multi-challenge) notions. (Double arrows: trivially tight.)

Next, note that each game has a single challenge oracle and a number of helper oracles. As suggested by the name, adding helper oracles can only help the adversary, making the resulting notion strictly more general—simulating the former game is as simple as ignoring the added oracle. We consider in particular two such helper oracles: the *decryption* oracle of CCA security, and the *corruption* oracle.

Lemma 3 (Adding oracles). *There are SFBB reductions $\mathbb{B}_i$ such that, for every adversary $\mathbb{A}_i$,*

$$\mathrm{IND\text{-}CPA}^{u,c}_{\mathrm{PKE}}(\mathbb{A}_1) \leq \mathrm{IND\text{-}CCA}^{u,c}_{\mathrm{PKE}}(\mathbb{B}_1)\,,$$
$$\mathrm{IND\text{-}CXA}^{\kappa,c}_{\mathrm{PKE}}(\mathbb{A}_2) \leq \mathrm{IND\text{-}CXA}^{\not\kappa,c}_{\mathrm{PKE}}(\mathbb{B}_2)\,.$$

Finally, increasing the parameters of the games can also only help the adversary.

Lemma 4 (Increasing parameters). *Let $\kappa' \geq \kappa$, and $\beta' \geq \beta$. Then there are SFBB reductions $\mathbb{B}_i$ such that, for every adversary $\mathbb{A}_i$,*

$$\mathrm{IND\text{-}CXA}^{\kappa,\beta}_{\mathrm{PKE}}(\mathbb{A}_1) \leq \mathrm{IND\text{-}CXA}^{\kappa',\beta'}_{\mathrm{PKE}}(\mathbb{B}_1)\,,$$
$$\mathrm{IND\text{-}CXA}^{\not\kappa,\beta}_{\mathrm{PKE}}(\mathbb{A}_2) \leq \mathrm{IND\text{-}CXA}^{\not\kappa',\beta'}_{\mathrm{PKE}}(\mathbb{B}_2)\,.$$

Corresponding relations for the simple notions follow as corollaries to Lemma 4; also note that setting $\kappa = \beta = 1$ gives the trivial implication that multi-user security is strictly stronger than single-user security.

4.2 Simple Multi-User Notions versus Classical Single-Key Notions

Bellare et al. [5] used a hybrid argument to show that single-user single-challenge security implies $\mathrm{LOR}^{\kappa,1}$ with a security loss linear in the total number of challenge encryption queries. They phrased this total as the product of the number of users and the number of challenges per user. As all our notions are explicitly multi-challenge, we will ignore the number of challenge queries, meaning the loss simply becomes linear in the number of users (in line with the original claim).

Bellare et al. did not consider the diagonal notion or corruptions, however later, when Jager et al. [25] introduced the free-bit notion to the setting of AE, they also showed that the simple notions are implied by the single-user notion, again with a linear loss, even when corruptions are considered. For completeness, we reprove the relevant linear losses in our new PKE context next. The resulting relations are summarized in Fig. 3.

As explained in Section 3.1, Jager et al. used slightly different game mechanics by prohibiting certain adversarial behaviour. In contrast, we allow such bad behaviour and just ignore the adversary's output instead. We introduce a useful lemma (Lemma 5) that formalizes that, in the single-key setting, our mechanism is sound and corrupting that single-key yields no adversarial advantage. This single-key-with-corruptions game is often easier to use in reductions.

Existing sharpness results can be used to show that linear losses are inevitable, see Appendix B.2 for details.

A single-user notion with corruptions. First, let us establish the trivial yet useful Lemma 5. Let $\mathsf{Exp}^{\mathrm{LOR\text{-}CXA},\not1,1}_{\mathrm{PKE}}(\mathbb{A})$ be exactly as the single-key game, except that the player now has the option to corrupt the key. In other words, the game will be equivalent to that of Fig. 1, with $\kappa = \beta = 1$ (and with or without decryption oracle). Given that in this game, an adversary that both challenges and corrupts will trigger the game to output the uniformly random value δ, the presence of a corruption oracle should yield no extra power. We formalize this intuition next.

Lemma 5 **(IND $\implies$ IND$^{\not{1},1}$).** *There is an SFBB reduction $\mathbb{B}$ with no additional overhead such that, for every adversary $\mathbb{A}$,*

$$\text{IND-CXA}^{\not{1},1}_{\text{PKE}}(\mathbb{A}) \leq \text{IND-CXA}_{\text{PKE}}(\mathbb{B})\,.$$

Proof. The following argument works the same whether IND $=$ LOR or ROR, and whether CXA $=$ CCA or CPA. The reduction $\mathbb{B}$, playing the regular single-key game, simulates the game with corruptions to $\mathbb{A}$ by forwarding every oracle call and mimicking $\mathbb{A}$'s output, unless at some point $\mathbb{A}$ asks to corrupt: in that case $\mathbb{B}$ aborts $\mathbb{A}$ and simply returns 0. This works because if $\mathbb{A}$ corrupts, either $\mathbb{A}$ also challenges, in which case the advantage will be forced to 0, or it corrupts the key and outputs a guess without challenging, in which case the challenge bit will be information-theoretically hidden from it, so that its advantage is 0 by necessity. Thus, in the event that $\mathbb{A}$ corrupts at all, its win advantage will be exactly 0; the same that $\mathbb{B}$ gets from simply aborting $\mathbb{A}$ and outputting 0. We provide a formal derivation below.

$$\begin{aligned}
\Pr\left[\mathsf{Exp}^{\text{IND-CXA}}_{\text{PKE}}(\mathbb{B}) = 1\right] &= \Pr[\mathbb{A} \text{ did not corrupt} \wedge \mathbb{A} \text{ wins}] \\
&\qquad + \Pr[\mathbb{A} \text{ did corrupt} \wedge b = 0] \\
&= \Pr[\mathbb{A} \text{ did not corrupt} \wedge \mathbb{A} \text{ wins}] \\
&\qquad + \Pr[\mathbb{A} \text{ did corrupt}] \cdot \Pr[b = 0 \mid \mathbb{A} \text{ did corrupt}] \\
&= \Pr[\mathbb{A} \text{ did not corrupt} \wedge \mathbb{A} \text{ wins}] \\
&\qquad + \Pr[\mathbb{A} \text{ did corrupt}] \cdot {}^1\!/_2 \\
&= \Pr[\mathbb{A} \text{ did not corrupt} \wedge \mathbb{A} \text{ wins}] \\
&\qquad + \Pr[\mathbb{A} \text{ did corrupt}] \cdot \Pr[\mathbb{A} \text{ wins} \mid \mathbb{A} \text{ did corrupt}] \\
&= \Pr[\mathbb{A} \text{ did not corrupt} \wedge \mathbb{A} \text{ wins}] \\
&\qquad + \Pr[\mathbb{A} \text{ did corrupt} \wedge \mathbb{A} \text{ wins}] \\
&= \Pr[\mathbb{A} \text{ wins}]\,, \\
\Rightarrow \text{IND-CXA}_{\text{PKE}}(\mathbb{B}) \geq \text{IND-CXA}^{\not{1},1}_{\text{PKE}}(\mathbb{A})\,. &
\end{aligned}$$

□

Single-bit security with corruptions. We can then show a reduction from $\text{IND-CXA}^{\not{\kappa},1}_{\text{PKE}}$ to $\text{IND-CXA}^{\not{1},1}_{\text{PKE}}$, using the exact same hybrid argument that was used by Bellare et al. [5] in the absence of corruptions, and let Lemma 5 imply the result.

Theorem 1 **(IND $\overset{\leq\kappa}{\implies}$ IND$^{\not{\kappa},1}$).** *There is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\text{IND-CXA}^{\not{\kappa},1}_{\text{PKE}}(\mathbb{A}) \leq \kappa \cdot \text{IND-CXA}_{\text{PKE}}(\mathbb{B})\,,$$

where $\mathbb{B}$'s overhead consists of $\kappa - 1$ fresh keypair generations.

We will show the result for IND $=$ LOR and CXA $=$ CCA; the proof transfers directly to the ROR and CPA cases.

Proof. We now show the first statement of the theorem. The proof is a standard hybrid argument, except that instead of playing $\mathsf{Exp}^{\text{LOR-CCA}}_{\text{PKE}}$, we let our reduction $\mathbb{B}$ play $\mathsf{Exp}^{\text{LOR-CCA},\not{1},1}_{\text{PKE}}$, (see Fig. 1 with $\kappa = \beta = 1$). Then, Lemma 5 implies the theorem statement.

Let G_0 be the game $\mathsf{Exp}^{\text{LOR-CCA},\not{\kappa},1}_{\text{PKE}}$ with the challenge bit set to 0, and let G_κ be the game with the bit set to 1 (see Fig. 1). Then, the advantage of an adversary $\mathbb{A}$ against $\mathsf{Exp}^{\text{LOR-CCA},\not{\kappa},1}_{\text{PKE}}$ becomes

$$\text{LOR-CCA}^{\not{\kappa},1}_{\text{PKE}}(\mathbb{A}) = \Pr[G_0(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] - \Pr[G_\kappa(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}]\,. \tag{1}$$

Next, let G_1 be the same game as G_0, except that the bit has been flipped to 1 for challenge queries using key pk_1. Similarly, let G_2 be the same as G_1, except now the challenge bit is set to one for challenge queries involving

```
$\mathbb{B}(\mathsf{pk})$
---
$\mathsf{pk}_{i^*} \leftarrow \mathsf{pk}$
for $i \in [\kappa], i \neq i^*$, do
    $(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow\!\!\$\ \mathsf{Pk.Kg}$
$\mathsf{C}_1, \ldots, \mathsf{C}_\kappa \leftarrow \emptyset$
$\hat{b} \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{K}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
return $\hat{b}$
```

```
if $\mathbb{A}$ calls $\mathcal{E}(i, m_0, m_1)$
---
if $m_0 \not\sim m_1$ then return $\lightning$
if $i < i^*$ : $c^* \leftarrow \mathsf{Pk.Enc}_{\mathsf{pk}_i}(m_1)$
else if $i = i^*$ : $c^* \leftarrow \mathcal{E}(m_0, m_1)$
else : $c^* \leftarrow \mathsf{Pk.Enc}_{\mathsf{pk}_i}(m_0)$
$\mathsf{C}_i \leftarrow c^*$
return $c^*$
```

```
if $\mathbb{A}$ calls $\mathcal{D}(i, c)$
---
if $c \in \mathsf{C}_j$ then return $\lightning$
if $i = i^*$ : $m \leftarrow \mathcal{D}(c)$
else : $m \leftarrow \mathsf{Pk.Dec}_{\mathsf{sk}_i}(c)$
return $m$
```

```
if $\mathbb{A}$ calls $\mathcal{K}(i)$
---
if $i = i^*$ : $\mathsf{sk}_i \leftarrow \mathcal{K}$
return $\mathsf{sk}_i$
```

Fig. 4: The adversary $\mathbb{B}$, playing $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not{1},1}$ while simulating the game G_{i^*-1+b} for $\mathbb{A}$.

either pk_1 or pk_2. In general, let G_i, $i \in [\kappa]$, be the game where every bit up to and including bit i is set to 1, and the rest are set to 0. Then, by “adding zeroes” to equation 1, we get that

$$\begin{aligned}\mathrm{LOR\text{-}CCA}_{\mathrm{PKE}}^{\not{\kappa},1}(\mathbb{A}) = \Pr[G_0(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] &- \Pr[G_1(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] \\ + \Pr[G_1(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] &- \Pr[G_2(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] \\ + \ldots + \Pr[G_{i-1}(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] &- \Pr[G_i(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] \\ + \ldots + \Pr[G_{\kappa-1}(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] &- \Pr[G_\kappa(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}]\,. \end{aligned} \tag{2}$$

The main observation is then that, given equation 1, there must be at least one $i^* \in [\kappa]$ such that

$$\Pr[G_{i^*-1}(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] - \Pr[G_{i^*}(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] \geq \frac{1}{\kappa} \cdot \mathrm{LOR\text{-}CCA}_{\mathrm{PKE}}^{\not{\kappa},1}(\mathbb{A})\,. \tag{3}$$

This distinguishing advantage can then be leveraged by an adversary $\mathbb{B}$ playing the single-key game $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not{1},1}$, in the manner outlined in Fig. 4. Denote the challenge bit that $\mathbb{B}$ is trying to guess by b; then $\mathbb{B}$ achieves the following advantage, $\Pr\left[\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not{1},1}(\mathbb{B}) = 1\right]$

$$\begin{aligned} &= \Pr[b = 0 \wedge 0 \leftarrow \mathbb{A}] + \Pr[b = 1 \wedge 1 \leftarrow \mathbb{A}] \\ &= \frac{1}{2}\left(\Pr[0 \leftarrow \mathbb{A} \mid b = 0] + \Pr[1 \leftarrow \mathbb{A} \mid b = 1]\right) \\ &= \frac{1}{2}\left(\Pr[G_{i-1}(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] + \Pr[G_i(\mathbb{A}) \,:\, 1 \leftarrow \mathbb{A}]\right) \\ &= \frac{1}{2}\left(\Pr[G_{i-1}(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] + (1 - \Pr[G_i(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}])\right) \\ &= \frac{1}{2}\left(\Pr[G_{i-1}(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}] - \Pr[G_i(\mathbb{A}) \,:\, 0 \leftarrow \mathbb{A}]\right) + \frac{1}{2} \\ &\geq \frac{1}{2} \cdot \frac{1}{\kappa} \cdot \mathrm{LOR\text{-}CCA}_{\mathrm{PKE}}^{\not{\kappa},1}(\mathbb{A}) + \frac{1}{2}\,, \end{aligned}$$

which implies that $\Rightarrow$ $\mathrm{LOR\text{-}CCA}_{\mathrm{PKE}}^{\not{1},1}(\mathbb{B})$

$$\begin{aligned} &= 2 \cdot \Pr\left[\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not{1},1}(\mathbb{B}) = 1\right] - 1 \\ &\geq 2 \cdot \left(\frac{1}{2} \cdot \frac{1}{\kappa} \cdot \mathrm{LOR\text{-}CCA}_{\mathrm{PKE}}^{\not{\kappa},1}(\mathbb{A}) + \frac{1}{2}\right) - 1 \\ &= \frac{1}{\kappa} \cdot \mathrm{LOR\text{-}CCA}_{\mathrm{PKE}}^{\not{\kappa},1}(\mathbb{A})\,. \end{aligned}$$

Taken together with Lemma 5, this implies the result. □

Diagonal-bit security with corruptions. For the diagonal notion, showing the relation to the single-user notion is done using a different—and arguably simpler—proof technique: the reduction $\mathbb{B}$ simply guesses which user $\mathbb{A}$ is going to attack, forwarding the oracles called to that user to its own oracles and simulating the rest; it will guess correctly with probability $1/\kappa$, leading to the κ security loss.

```
$\mathbb{B}(\mathsf{pk})$
$i^* \leftarrow\!\$\ [\kappa], \mathsf{pk}_{i^*} \leftarrow \mathsf{pk}$
for $i \in [\kappa], i \neq i^*,$ do
    $(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow\!\$\ \mathsf{Pk.Kg}$
    $b_i \leftarrow\!\$\ \{0,1\}$
    $\mathtt{C}_i \leftarrow \emptyset$
$(i, \hat{b}_i) \leftarrow\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{K}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
if $i \neq i^*$ then return $0$
return $\hat{b}_i$
```

```
if $\mathbb{A}$ calls $\mathcal{E}(i, m_0, m_1)$
if $m_0 \not\sim m_1$ then return $\lightning$
if $i = i^*$ then $c^* \leftarrow \mathcal{E}(m_0, m_1)$
else $c^* \leftarrow\!\$\ \mathsf{Pk.Enc}_{pk_i}(m_{b_i})$
$\mathtt{C}_i \leftarrow c^*$
return $c^*$
```

```
if $\mathbb{A}$ calls $\mathcal{D}(i, c)$
if $i = i^*$ then $m \leftarrow \mathcal{D}(c)$
else if $c \in \mathtt{C}_i$ then return $\lightning$
else $m \leftarrow \mathsf{Pk.Dec}_{\mathsf{sk}_i}(c)$
return $m$
```

```
if $\mathbb{A}$ calls $\mathcal{K}(i)$
if $i = i^*$ then $\mathsf{sk}_i \leftarrow \mathcal{K}$
return $\mathsf{sk}_i$
```

Fig. 5: The adversary $\mathbb{B}$, playing $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not 1,1}$ while simulating $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not\kappa,\boxslash}$ for $\mathbb{A}$.

Theorem 2 **(**IND $\overset{\leqslant\kappa}{\Longrightarrow}$ IND$^{\not\kappa,\boxslash}$**).** *There is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\text{IND-CXA}_{\text{PKE}}^{\not\kappa,\boxslash}(\mathbb{A}) \leq \kappa \cdot \text{IND-CXA}_{\text{PKE}}(\mathbb{B})\,,$$

where $\mathbb{B}$'s overhead consists of generating $\kappa - 1$ fresh keypairs and drawing $\kappa - 1$ challenge bits uniformly at random.

We will show the result for IND $=$ LOR and CXA $=$ CCA; the proof transfers directly to the ROR and CPA cases.

Proof. We will prove the statement by constructing an adversary $\mathbb{B}$ that achieves the claimed advantage by leveraging any advantage an adversary $\mathbb{A}$ has against the diagonal multi-key game, and making a guess on the key that $\mathbb{A}$ is going to attack. $\mathbb{B}$ will guess correctly with probability $1/\kappa$, leading to the κ security loss.

As before, we will simplify the analysis by letting $\mathbb{B}$ play the game $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not 1,1}(\mathbb{A})$, and then let Lemma 5 imply the result.

$\mathbb{B}$ is given in Fig. 5. In the following, let pk and b be the public key and challenge bit of $\mathbb{B}$'s game $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not 1,1}$ (see Fig. 1 with $\kappa = \beta = 1$), let the set of key handles that $\mathbb{A}$ asks to corrupt be denoted by $\mathtt{I}^{\mathcal{K}}$, and assume that $\mathbb{A}$ returns the guess $(i, \hat{b}_i)$. Finally, note that the value of i^* is information-theoretically hidden from $\mathbb{A}$. Then, $\mathbb{B}$ achieves the following advantage.

$$\begin{aligned}
\Pr\Big[\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not 1,1}(\mathbb{B}) = 1\Big] &= \Pr\Big[i = i^* \wedge i^* \notin \mathtt{I}^{\mathcal{K}} \wedge b_{i^*} = \hat{b}_{i^*}\Big] + \Pr\big[i = i^* \wedge i^* \in \mathtt{I}^{\mathcal{K}} \wedge \delta = 1\big] \\
&\quad + \Pr\big[i \neq i^* \wedge i^* \notin \mathtt{I}^{\mathcal{K}} \wedge b = 0\big] + \Pr\big[i \neq i^* \wedge i^* \in \mathtt{I}^{\mathcal{K}} \wedge \delta = 1\big] \\
&= \Pr[i = i^*] \left(\Pr\Big[i^* \notin \mathtt{I}^{\mathcal{K}} \wedge b_{i^*} = \hat{b}_{i^*} \,\Big|\, i = i^*\Big] + \Pr\big[i^* \in \mathtt{I}^{\mathcal{K}} \wedge \delta = 1 \,\big|\, i = i^*\big]\right) \\
&\quad + \Pr[i \neq i^*] \left(\Pr[b = 0] \cdot \Pr\big[i^* \notin \mathtt{I}^{\mathcal{K}} \,\big|\, i \neq i^*\big] + \Pr[\delta = 1] \cdot \Pr\big[i^* \in \mathtt{I}^{\mathcal{K}} \,\big|\, i \neq i^*\big]\right) \\
&= \frac{1}{\kappa} \left(\Pr\Big[i \notin \mathtt{I}^{\mathcal{K}} \wedge b_i = \hat{b}_i\Big] + \Pr\big[i \in \mathtt{I}^{\mathcal{K}} \wedge \delta = 1\big]\right) \\
&\quad + \frac{1}{2}\left(1 - \frac{1}{\kappa}\right) \left(\Pr\big[i^* \notin \mathtt{I}^{\mathcal{K}} \,\big|\, i \neq i^*\big] + \Pr\big[i^* \in \mathtt{I}^{\mathcal{K}} \,\big|\, i \neq i^*\big]\right) \\
&= \frac{1}{\kappa} \Pr\Big[\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not\kappa,\boxslash}(\mathbb{A}) = 1\Big] + \frac{1}{2}\left(1 - \frac{1}{\kappa}\right) \\
&= \frac{1}{2\kappa} \left(2 \cdot \Pr\Big[\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not\kappa,\boxslash}(\mathbb{A}) = 1\Big] - 1\right) + \frac{1}{2} \\
\implies \text{LOR-CCA}_{\text{PKE}}^{\not 1,1}(\mathbb{B}) &= 2 \cdot \Pr\Big[\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not 1,1}(\mathbb{B}) = 1\Big] - 1 \\
&= 2 \cdot \left(\frac{1}{2\kappa} \text{LOR-CCA}_{\text{PKE}}^{\not\kappa,\boxslash}(\mathbb{A}) + \frac{1}{2}\right) - 1 \\
&= \frac{1}{\kappa} \cdot \text{LOR-CCA}_{\text{PKE}}^{\not\kappa,\boxslash}(\mathbb{A})\,.
\end{aligned}$$

Taken together with Lemma 5, this implies the result. □

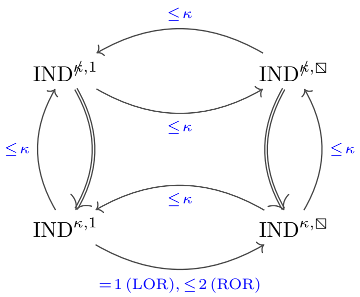

$\leq\kappa$
$\mathrm{IND}^{\not\kappa,1}$
$\mathrm{IND}^{\not\kappa,\boxbslash}$
$\leq\kappa$
$\leq\kappa$
$\leq\kappa$
$\leq\kappa$
$\mathrm{IND}^{\kappa,1}$
$\mathrm{IND}^{\kappa,\boxbslash}$
$=1$ (LOR), ≤ 2 (ROR)

Fig. 6: Relations between the simple multi-user notions, including the non-trivially tight relation between $\mathrm{IND}^{\kappa,\beta}$ and $\mathrm{IND}^{\kappa,\boxbslash}$ as captured by Corollaries 1 and 2 for LOR and ROR, respectively. (Double arrows: trivially tight.)

4.3 Relationship between Simple Multi-User Notions

Now that we have affirmed that the single-user notion implies any of the four simple multi-user notions with a loss linear in the number of users, a natural question is how the simple multi-user notions relate to each other. As the multi-user notions all tightly imply the single-user notion, one can always just go via the single-user notion. As already noted by Jager et al. [25], this strategy will again lead to a loss linear in the number of users. Lemma 6 formalizes this trivial loss and Fig. 6 provides an overview of the relations. One notable exception from the linear losses is the implication from the single-bit notion to the diagonal notion if there are no corruptions, which is tight for the case of left-or-right indistinguishability and almost tight for real-or-random indistinguishability. We will explain why this is in the next paragraph.

Lemma 6 **($\mathrm{IND}^{u,c} \overset{\leq\kappa}{\Longrightarrow} \mathrm{IND}^{u,c'}$).** *Let $c' \in \{1, \boxbslash\}$. Then, there is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\mathrm{IND\text{-}CXA}^{u',c'}_{\mathrm{PKE}}(\mathbb{A}) \leq \kappa \cdot \mathrm{IND\text{-}CXA}^{u,c}_{\mathrm{PKE}}(\mathbb{B})\,.$$

Proof (sketch). Lemma 4 implies that $\mathrm{IND}^{\kappa,c} \implies \mathrm{IND}$. Meanwhile, Theorems 1 and 2 together say that $\mathrm{IND} \overset{\leq\kappa}{\Longrightarrow} \mathrm{IND}^{\not\kappa,c}$. Combining (in the manner discussed in Section 2) gives $\mathrm{IND}^{\kappa,c} \implies \mathrm{IND} \overset{\leq\kappa}{\Longrightarrow} \mathrm{IND}^{\not\kappa,c'}$. The remaining relations follow. □

A tight relation: from single-bit to multi-bit without corruptions. Surprisingly, left-or-right indistinguishability allows for a 'bit-hiding' argument that lets an adversary playing a single-bit multi-user game simulate the full free-bit game (and therefore also the diagonal-bit game), by simply exchanging the order in which it forwards its two messages. We formalize this argument in Theorem 3 and its proof. Consequently, $\mathrm{LOR}^{\kappa,1}$ tightly implies $\mathrm{LOR}^{\kappa,\boxbslash}$ (Corollary 1), whereas the implication in the other direction appears lossy. This clearly renders $\mathrm{LOR}^{\kappa,1}$ the preferred notion.

Theorem 3 **($\mathrm{LOR}^{\kappa,1} \implies \mathrm{LOR}^{\kappa,\beta}$).** *There is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\mathrm{LOR\text{-}CXA}^{\kappa,\beta}_{\mathrm{PKE}}(\mathbb{A}) \leq \mathrm{LOR\text{-}CXA}^{\kappa,1}_{\mathrm{PKE}}(\mathbb{B})\,,$$

where $\mathbb{B}$'s overhead is limited to drawing β uniformly random bits.

Proof. The reduction $\mathbb{B}$, playing $\mathsf{Exp}^{\mathrm{LOR\text{-}CXA},\kappa,1}_{\mathrm{PKE}}$, simulates $\mathsf{Exp}^{\mathrm{LOR\text{-}CXA},\kappa,\beta}_{\mathrm{PKE}}$ for $\mathbb{A}$ by drawing β fresh challenge bits b_j, and simply exchanging the order of m_0 and m_1 whenever $b_j = 1$ (see Fig. 7). Denoting the challenge bit of $\mathsf{Exp}^{\mathrm{LOR\text{-}CXA},\kappa,1}_{\mathrm{PKE}}(\mathbb{B})$ by b, the ciphertext that $\mathbb{A}$ receives upon the query $\mathcal{E}(i, j, m_0, m_1)$ will be an encryption of the message $m_{b\oplus b_j}$ under pk_i; $\mathbb{B}$ then simply makes sure to undo the xor before returning its final guess. □

Corollary 1 **($\mathrm{LOR}^{\kappa,1} \implies \mathrm{LOR}^{\kappa,\boxbslash}$).** *There is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\mathrm{LOR\text{-}CXA}^{\kappa,\boxbslash}_{\mathrm{PKE}}(\mathbb{A}) \leq \mathrm{LOR\text{-}CXA}^{\kappa,1}_{\mathrm{PKE}}(\mathbb{B})\,,$$

where $\mathbb{B}$'s overhead is limited to drawing β uniformly random bits.

```
$\mathbb{B}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
    $b_1, \ldots, b_\beta \leftarrow\!\!\$\ \{0,1\}$
    $(j, \hat{b}_j) \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
    return $\hat{b}_j \oplus b_j$

if $\mathbb{A}$ calls $\mathcal{E}(i, j, m_0, m_1)$
    $c^* \leftarrow \mathcal{E}(i, m_{b_j}, m_{\bar{b}_j})$
    return $c^*$

if $\mathbb{A}$ calls $\mathcal{D}(i, c)$
    $m \leftarrow \mathcal{D}(i, c)$
    return $m$
```

Fig. 7: The adversary $\mathbb{B}$, playing $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CXA},\kappa,1}$ while simulating $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CXA},\kappa,\beta}$ for $\mathbb{A}$.

In the presence of a corruption oracle, the reduction breaks down as it is no longer able to simulate properly: an adversary playing the game cannot both challenge and corrupt the same key, as this would lead to a trivial win. In contrast, challenging and corrupting a key is a perfectly viable strategy in the diagonal- and free-bit games, as long as the corresponding bit is not chosen at the end. We will return to the free-bit game in the presence of corruptions below, but first we turn our attention to that other indistinguishability notion, real-or-random.

Extending the argument to real-or-random. The proof of Theorem 3 makes use of the fact that the LOR challenge oracle allows both a left and a right message to be input, enabling us to hide the bit in the ordering of the two messages. For ROR, the challenge oracle only accepts a single message, so hiding the bit as above is no longer possible.

However, when Bellare et al. [6] introduced the distinction between LOR versus ROR indistinguishability in the context of single-user probabilistic symmetric encryption, they also showed a factor-2 loss from ROR to LOR. As we will show in Theorem 5 (to be presented shortly), their proof technique is readily adapted to a relation between single-bit multi-user PKE notions. Theorems 3 and 5 can then be combined into the corollary below (which itself implies the equivalent of Corollary 1 for ROR, again with a factor 2 loss).

Corollary 2 **($\mathrm{ROR}^{\kappa,1} \overset{\leq 2}{\Longrightarrow} \mathrm{ROR}^{\kappa,\beta}$).** *There is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\mathrm{ROR\text{-}CXA}_{\mathrm{PKE}}^{\kappa,\beta}(\mathbb{A}) \leq 2 \cdot \mathrm{ROR\text{-}CXA}_{\mathrm{PKE}}^{\kappa,1}(\mathbb{B}),$$

where $\mathbb{B}$'s overhead is limited to drawing β uniformly random bits.

Proof (Sketch). Theorem 5 states that $\mathrm{ROR}^{\kappa,1} \overset{\leq 2}{\Longrightarrow} \mathrm{LOR}^{\kappa,1}$, while we know from Lemma 2 that $\mathrm{LOR}^{\kappa,\beta} \Longrightarrow \mathrm{ROR}^{\kappa,\beta}$. Then using Theorem 3, we get $\mathrm{ROR}^{\kappa,1} \overset{\leq 2}{\Longrightarrow} \mathrm{LOR}^{\kappa,1} \Longrightarrow \mathrm{LOR}^{\kappa,\beta} \Longrightarrow \mathrm{ROR}^{\kappa,\beta}$. □

4.4 The Free-Bit Game with Corruptions

In the free-bit game, the adversary can both challenge and corrupt keys, provided the final targeted bit remains uncompromised. In the single-bit game, however, challenging and corrupting a key are mutually exclusive, causing the bit-hiding argument that tightly related $\mathrm{LOR}^{\kappa,1}$ to $\mathrm{LOR}^{\kappa,\beta}$ to break down. It seems the best we can do is a standard bit-guessing argument, suffering a β loss, as formalized in Theorem 4 below.

Theorem 4 **($\mathrm{IND}^{\not\kappa,1} \overset{\leq \beta}{\Longrightarrow} \mathrm{IND}^{\not\kappa,\beta}$).** *There is an SFBB reduction $\mathbb{B}$ such that, for any adversary $\mathbb{A}$,*

$$\mathrm{IND\text{-}CXA}_{\mathrm{PKE}}^{\not\kappa,\beta}(\mathbb{A}) \leq \beta \cdot \mathrm{IND\text{-}CXA}_{\mathrm{PKE}}^{\not\kappa,1}(\mathbb{B}),$$

where $\mathbb{B}$'s overhead consists of drawing $\beta - 1$ uniformly random bits.

We will show the result for IND $=$ LOR and CXA $=$ CCA; the proof transfers directly to the ROR and CPA cases.

Proof. We will prove the statement by constructing an adversary $\mathbb{B}$ that achieves the claimed advantage by leveraging any advantage an adversary $\mathbb{A}$ has against the free-bit game, and making a guess on the bit that $\mathbb{A}$ is going to attack. $\mathbb{B}$ will guess correctly with probability $1/\beta$, leading to the β security loss. The proof is very similar to that of Theorem 2, the main complication being that we now need to keep track of compromised challenge bits, instead of just which keys are corrupted.

$\mathbb{B}$ is given in Fig. 8. In the following, let b be the challenge bit of $\mathbb{B}$'s game $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not\kappa,1}$ (see Fig. 1, with $\beta = 1$), let the set of compromised bits (i.e., bits used by $\mathbb{A}$ to challenge a corrupted key) be denoted by $\mathsf{J}^{\mathcal{K}}$, and

```
$\mathbb{B}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
$j^* \leftarrow\!\!\$ \; [\beta]$
for $j \in [\beta], j \neq j^*$ do :
    $b_j \leftarrow\!\!\$ \; \{0, 1\}$
$\mathsf{C}_1, \ldots, \mathsf{C}_\kappa \leftarrow \emptyset$
$(j, \hat{b}_j) \leftarrow\!\!\$ \; \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{K}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
if $j \neq j^*$ then return $0$
return $\hat{b}_j$

if $\mathbb{A}$ calls $\mathcal{E}(i, j, m_0, m_1)$
if $m_0 \not\sim m_1$ then return $\lightning$
if $j = j^*$
    $c^* \leftarrow \mathcal{E}(i, m_0, m_1)$
else $c^* \leftarrow\!\!\$ \; \mathsf{Pk.Enc}_{\mathsf{pk}_i}(m_{b_j})$
$\mathsf{C}_i \leftarrow c^*$
return $c^*$

if $\mathbb{A}$ calls $\mathcal{D}(i, c)$
if $c \in \mathsf{C}_i$ then return $\lightning$
$m \leftarrow \mathcal{D}(i, c)$
return $m$

if $\mathbb{A}$ calls $\mathcal{K}(i)$
$\mathsf{sk}_i \leftarrow \mathcal{K}(i)$
return $\mathsf{sk}_i$
```

Fig. 8: The adversary $\mathbb{B}$, playing $\mathsf{Exp}^{\text{LOR-CCA},\not\kappa,1}_{\text{PKE}}$ while simulating $\mathsf{Exp}^{\text{LOR-CCA},\not\kappa,\beta}_{\text{PKE}}$ for $\mathbb{A}$.

assume that $\mathbb{A}$ returns the guess $(j, \hat{b}_j)$. Finally, note that the value of j^* is information-theoretically hidden from $\mathbb{A}$. Then, $\mathbb{B}$ achieves the following advantage, $\Pr\left[\mathsf{Exp}^{\text{LOR-CCA},\not\kappa,1}_{\text{PKE}}(\mathbb{B}) = 1\right]$

$$\begin{aligned}
&= \Pr\left[j = j^* \wedge j^* \notin \mathtt{J}^{\mathcal{K}} \wedge b_{j^*} = \hat{b}_{j^*}\right] + \Pr\left[j = j^* \wedge j^* \in \mathtt{J}^{\mathcal{K}} \wedge \delta = 1\right] \\
&\quad + \Pr\left[j \neq j^* \wedge j^* \notin \mathtt{J}^{\mathcal{K}} \wedge b = 0\right] + \Pr\left[j \neq j^* \wedge j^* \in \mathtt{J}^{\mathcal{K}} \wedge \delta = 1\right] \\
&= \Pr[j = j^*] \left(\Pr\left[j^* \notin \mathtt{J}^{\mathcal{K}} \wedge b_{j^*} = \hat{b}_{j^*} \;\middle|\; j = j^*\right] + \Pr\left[j^* \in \mathtt{J}^{\mathcal{K}} \wedge \delta = 1 \;\middle|\; j = j^*\right]\right) \\
&\quad + \Pr[j \neq j^*] \left(\Pr[b = 0] \cdot \Pr\left[j^* \notin \mathtt{J}^{\mathcal{K}} \;\middle|\; j \neq j^*\right] + \Pr[\delta = 1] \cdot \Pr\left[j^* \in \mathtt{J}^{\mathcal{K}} \;\middle|\; j \neq j^*\right]\right) \\
&= \frac{1}{\beta} \left(\Pr\left[j \notin \mathtt{J}^{\mathcal{K}} \wedge b_j = \hat{b}_j\right] + \Pr\left[j \in \mathtt{I}^{\mathcal{K}} \wedge \delta = 1\right]\right) \\
&\quad + \frac{1}{2}\left(1 - \frac{1}{\beta}\right) \left(\Pr\left[j^* \notin \mathtt{J}^{\mathcal{K}} \;\middle|\; j \neq j^*\right] + \Pr\left[j^* \in \mathtt{J}^{\mathcal{K}} \;\middle|\; j \neq j^*\right]\right) \\
&= \frac{1}{\beta} \Pr\left[\mathsf{Exp}^{\text{LOR-CCA},\not\kappa,\beta}_{\text{PKE}}(\mathbb{A}) = 1\right] + \frac{1}{2}\left(1 - \frac{1}{\beta}\right) \\
&= \frac{1}{2\beta} \left(2 \cdot \Pr\left[\mathsf{Exp}^{\text{LOR-CCA},\not\kappa,\beta}_{\text{PKE}}(\mathbb{A}) = 1\right] - 1\right) + \frac{1}{2}
\end{aligned}$$

which implies that $\text{LOR-CCA}^{\not\kappa,1}_{\text{PKE}}(\mathbb{B})$

$$\begin{aligned}
&= 2 \cdot \Pr\left[\mathsf{Exp}^{\text{LOR-CCA},\not\kappa,1}_{\text{PKE}}(\mathbb{B}) = 1\right] - 1 \\
&= 2 \cdot \left(\frac{1}{2\beta} \text{LOR-CCA}^{\not\kappa,\beta}_{\text{PKE}}(\mathbb{A}) + \frac{1}{2}\right) - 1 \\
&= \frac{1}{\beta} \cdot \text{LOR-CCA}^{\not\kappa,\beta}_{\text{PKE}}(\mathbb{A}) \,,
\end{aligned}$$

which is what we set out to show. □

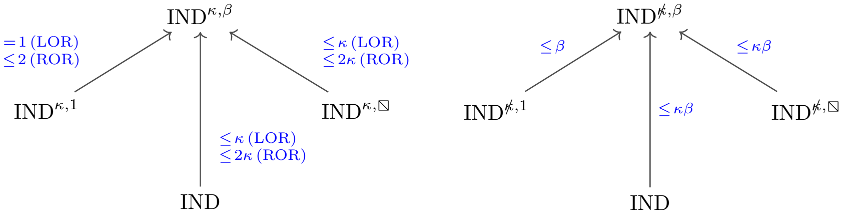
Combining with IND $\xRightarrow{\leq\kappa}$ IND$^{\not\kappa,1}$ (Theorem 1) yields an upper bound on the free-bit advantage as it relates to single-user advantage, see Corollary 3. Notably, when Jager et al. [25] introduced the free-bit notion (for AE), they observed that proving a linear loss was beyond them, yet they did not provide an alternative, looser bound instead. We therefore plug this gap in the literature. Fig. 9 provides an overview of how the single-user and simple multi-user notions relate to the free-bit notions.

Corollary 3 **(IND $\xRightarrow{\leq\kappa\beta}$ IND$^{\not\kappa,\beta}$).** *There is an SFBB reduction $\mathbb{B}$ such that, for any adversary $\mathbb{A}$,*

$$\text{IND-CXA}^{\not\kappa,\beta}_{\text{PKE}}(\mathbb{A}) \leq \kappa \cdot \beta \cdot \text{IND-CXA}_{\text{PKE}}(\mathbb{B}) \,.$$

where $\mathbb{B}$'s overhead consists of drawing $\kappa - 1$ fresh keypairs and $\beta - 1$ bits uniformly at random.

Interestingly, Corollary 3 tightly implies Theorem 1, but not Theorem 2: setting $\kappa = \beta$ in Corollary 3 yields a κ^2 loss. This gives some hope that a tighter relation than that of Corollary 3 might still be possible, one that would imply both Theorems 1 and 2. We leave this an open problem, although present some initial thoughts in Appendix B.3.



$\mathrm{IND}^{\kappa,\beta}$
$=1$ (LOR)
≤ 2 (ROR)
$\leq \kappa$ (LOR)
$\leq 2\kappa$ (ROR)
$\mathrm{IND}^{\kappa,1}$
$\mathrm{IND}^{\kappa,\boxbslash}$
$\leq \kappa$ (LOR)
$\leq 2\kappa$ (ROR)
IND
$\mathrm{IND}^{\not\kappa,\beta}$
$\leq \beta$
$\leq \kappa\beta$
$\mathrm{IND}^{\not\kappa,1}$
$\leq \kappa\beta$
$\mathrm{IND}^{\not\kappa,\boxbslash}$
IND

Fig. 9: Relations between different multi-user notions, without corruptions (left), and with corruptions (right).

4.5 LOR versus ROR, or When the Challenge Oracle Matters

Until now, we have for the most part treated the two flavours of indistinguishability as one. However, as we saw for Theorem 3, the choice of challenge oracle can sometimes make a difference. Of course, as stated in Lemma 2, left-or-right indistinguishability always implies real-or-random indistinguishability. Furthermore, for single-user notions, it has long been known that ROR implies LOR with only a factor 2 tightness loss [5]. However, for multi-instance security, the loss is known to blow up exponentially [10]. Thus, it is a priori unclear what losses one should expect for the multi-user setting, both between corresponding LOR and ROR notions, but also between the ROR notions themselves.

Jager et al. [26, Theorem 21] showed a general result that a loss L in the single user setting can be turned into a loss $L\kappa$ for the simple notions (for AE); the free-bit case is not addressed. We complement their results for the PKE setting, as summarized in Fig. 10.

Some relations are worth highlighting. First, note that the same factor 2 reduction still lends itself to the single-bit multi-key setting (with or without corruptions). The argument is very similar to that of the single-user case: either the bit is "real", in which case the simulated game is equivalent to the left-or-right one, or the bit is "random", in which case the simulated challenge bit is information-theoretically hidden from the adversary; the main complication in going to a multi-key setting with corruptions being dealing with disallowed guesses. See Theorem 5. This contrasts to the diagonal-bit setting, in which the tightest known reduction loses a factor 2κ, as achieved via the single-user relation: $\mathrm{ROR}^{u,\boxbslash} \implies \mathrm{ROR} \overset{\leq 2}{\implies} \mathrm{LOR} \overset{\leq \kappa}{\implies} \mathrm{LOR}^{u,\boxbslash}$.

Second, note that the fact that $\mathrm{LOR}^{\kappa,1} \implies \mathrm{LOR}^{\kappa,\beta}$ (Theorem 3) allows us to conclude that the factor 2 reduction still holds for the free-bit notion absent corruptions: $\mathrm{ROR}^{\kappa,\beta} \implies \mathrm{ROR}^{\kappa,1} \overset{\leq 2}{\implies} \mathrm{LOR}^{\kappa,1} \implies \mathrm{LOR}^{\kappa,\beta}$. Compare with the situation in the presence of corruptions, where the corresponding implications yield $\mathrm{ROR}^{\not\kappa,\beta} \overset{\leq 2\beta}{\implies} \mathrm{LOR}^{\not\kappa,\beta}$.

As before, we leave the question of whether there exist tighter reductions, or these losses really are inevitable, as open questions. Nevertheless, these additional losses serve to reinforce the folklore that left-or-right notions should be preferred over real-or-random whenever possible.

To start, we show that real-or-random indistinguishability implies left-or-random indistinguishability in the single-bit setting, with or without corruptions.

Theorem 5 **($\mathrm{ROR}^{u,1} \overset{\leq 2}{\implies} \mathrm{LOR}^{u,1}$).** *There is an SFBB reduction $\mathbb{B}$ such that, for any adversary $\mathbb{A}$,*

$$\text{LOR-CXA}^{u,1}_{\text{PKE}}(\mathbb{A}) \leq 2 \cdot \text{ROR-CXA}^{u,1}_{\text{PKE}}(\mathbb{B})\,,$$

where $\mathbb{B}$'s overhead consists of drawing one uniformly random bit.

Proof (Sketch). Essentially, there are only two, equally likely cases: either the bit is "real", in which case $\mathbb{B}$ is able to simulate the left-or-right game perfectly; or the bit is "random", in which case the advantage of $\mathbb{A}$ against the simulated game will be exactly 0—and the addition of corruptions does nothing to change this fact.

Proof. We will show the theorem for the case $u = \not\kappa$ and $\mathrm{CXA} = \mathrm{CCA}$; by inspection, the proof also holds for the cases $u = \kappa$ (by setting $\Pr\left[1 \in \mathsf{J}^{\mathcal{K}}\right] = 0$), and $\mathrm{CXA} = \mathrm{CPA}$.

In the following, let b be the challenge bit of $\mathbb{B}$'s game $\mathsf{Exp}^{\text{ROR-CCA},\not\kappa,1}_{\text{PKE}}$ (see Fig. 1, with $\beta = 1$). Let $\mathsf{J}^{\mathcal{K}}$ denote the set of compromised bits; note however that there is now only one challenge bit per game, meaning its bit

$$\begin{tikzcd}
\mathrm{LOR}^{\kappa,1} \arrow[r, bend left, "=1"] \arrow[d, Rightarrow, bend left] & \mathrm{LOR}^{\kappa,\beta} \arrow[l, Rightarrow, bend left] \arrow[d, Rightarrow, bend left] \\
\mathrm{ROR}^{\kappa,1} \arrow[u, bend left, "\leq 2"] & \mathrm{ROR}^{\kappa,\beta} \arrow[l, Rightarrow] \arrow[d, Rightarrow] \\
 & \mathrm{ROR}^{\kappa,\boxbslash}
\end{tikzcd}
\qquad
\begin{tikzcd}
 & \mathrm{LOR}^{\not\kappa,\beta} \arrow[dl, Rightarrow, bend right] \arrow[dd, Rightarrow] \arrow[dr, Rightarrow] & \\
\mathrm{LOR}^{\not\kappa,1} \arrow[ur, bend left, "\leq\beta"] \arrow[dd, Rightarrow, bend left] & & \mathrm{LOR}^{\not\kappa,\boxbslash} \arrow[dd, Rightarrow, bend right] \\
 & \mathrm{ROR}^{\not\kappa,\beta} \arrow[dl, Rightarrow, bend right] \arrow[dr, Rightarrow] & \\
\mathrm{ROR}^{\not\kappa,1} \arrow[uu, bend left, "\leq 2"] \arrow[ur, bend right, "\leq\beta"'] & & \mathrm{ROR}^{\not\kappa,\boxbslash} \arrow[uu, bend right, "\leq 2\kappa"']
\end{tikzcd}$$

Fig. 10: Relations between lor and ror for the different multi-user notions, without corruptions (left) and with corruptions (right). The placement of notions roughly translate to their relative strength, with stronger notions placed higher, (see Fig. 9 for the implications missing from the figure.) As before, double arrows indicate trivially tight.

handle is 1, and the event that it was compromised is denoted by $1 \in \mathsf{J}^{\mathcal{K}}$. Using the strategy of Fig. 11, we then get $\Pr\left[\mathsf{Exp}^{\text{ROR-CCA},\not\kappa,1}_{\text{PKE}}(\mathbb{B}) = 1\right]$

$$\begin{aligned}
&= \Pr\left[1 \notin \mathsf{J}^{\mathcal{K}} \wedge d = \hat{d} \wedge b = 0\right] + \Pr\left[1 \notin \mathsf{J}^{\mathcal{K}} \wedge d \neq \hat{d} \wedge b = 1\right] \\
&\qquad + \Pr\left[1 \in \mathsf{J}^{\mathcal{K}} \wedge \delta = 1\right] \\
&= \Pr[b = 0] \left(\Pr\left[1 \notin \mathsf{J}^{\mathcal{K}} \wedge d = \hat{d} \,\middle|\, b = 0\right] + \Pr\left[1 \in \mathsf{J}^{\mathcal{K}} \wedge \delta = 1 \,\middle|\, b = 0\right]\right) \\
&\qquad + \Pr[b = 1] \left(\Pr\left[1 \notin \mathsf{J}^{\mathcal{K}} \wedge d \neq \hat{d} \,\middle|\, b = 1\right] + \Pr\left[1 \in \mathsf{J}^{\mathcal{K}} \wedge \delta = 1 \,\middle|\, b = 1\right]\right)
\end{aligned}$$

Note that if $b = 1$, then the value of d is information-theoretically hidden from $\mathbb{A}$, so we have that $\Pr\left[d \neq \hat{d} \,\middle|\, b = 1\right] = \Pr[\delta = 1 \mid b = 1] = 1/2$, allowing us to write

$$\begin{aligned}
&= \Pr[b = 0] \left(\Pr\left[1 \notin \mathsf{J}^{\mathcal{K}} \wedge d = \hat{d} \,\middle|\, b = 0\right] + \Pr\left[1 \in \mathsf{J}^{\mathcal{K}} \wedge \delta = 1 \,\middle|\, b = 0\right]\right) \\
&\qquad + \Pr[b = 1] \left(\Pr\left[1 \notin \mathsf{J}^{\mathcal{K}} \wedge \delta = 1 \,\middle|\, b = 1\right] + \Pr\left[1 \in \mathsf{J}^{\mathcal{K}} \wedge \delta = 1 \,\middle|\, b = 1\right]\right) \\
&= \Pr[b = 0] \left(\Pr\left[1 \notin \mathsf{J}^{\mathcal{K}} \wedge d = \hat{d} \,\middle|\, b = 0\right] + \Pr\left[1 \in \mathsf{J}^{\mathcal{K}} \wedge \delta = 1 \,\middle|\, b = 0\right]\right) \\
&\qquad + \Pr[b = 1] \cdot \Pr[\delta = 1 \mid b = 1] \\
&= \frac{1}{2}\left(\Pr\left[1 \notin \mathsf{J}^{\mathcal{K}} \wedge d = \hat{d} \,\middle|\, b = 0\right] + \Pr\left[1 \in \mathsf{J}^{\mathcal{K}} \wedge \delta = 1 \,\middle|\, b = 0\right] + \frac{1}{2}\right) \\
&= \frac{1}{2}\left(\Pr\left[\mathsf{Exp}^{\text{LOR-CCA},\not\kappa,1}_{\text{PKE}}(\mathbb{A}) = 1\right] + \frac{1}{2}\right) .
\end{aligned}$$

Which implies that $\text{ROR-CCA}^{\not\kappa,1}_{\text{PKE}}(\mathbb{B})$

$$\begin{aligned}
&= 2 \cdot \Pr\left[\mathsf{Exp}^{\text{ROR-CCA},\not\kappa,1}_{\text{PKE}}(\mathbb{B}) = 1\right] - 1 \\
&= 2 \cdot \frac{1}{2}\left(\Pr\left[\mathsf{Exp}^{\text{LOR-CCA},\not\kappa,1}_{\text{PKE}}(\mathbb{A}) = 1\right] + \frac{1}{2}\right) - 1 \\
&= \frac{1}{2}\left(2 \cdot \Pr\left[\mathsf{Exp}^{\text{LOR-CCA},\not\kappa,1}_{\text{PKE}}(\mathbb{A}) = 1\right] - 1\right) \\
&= \frac{1}{2} \cdot \text{LOR-CCA}^{\not\kappa,1}_{\text{PKE}}(\mathbb{A}) ,
\end{aligned}$$

which is what we aimed to show. □

```
$\mathbb{B}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
    $d \leftarrow\!\!\$\ \{0,1\}$
    $\hat{d} \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{K}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
    $\hat{b} \leftarrow d \neq \hat{d}$
    return $\hat{b}$

if $\mathbb{A}$ calls $\mathcal{E}(i, m_0, m_1)$
    $c^* \leftarrow \mathcal{E}(i, m_d)$
    return $c^*$

if $\mathbb{A}$ calls $\mathcal{D}(i, c)$
    $m \leftarrow \mathcal{D}(i, c)$
    return $m$

if $\mathbb{A}$ calls $\mathcal{K}(i)$
    $\mathsf{sk}_i \leftarrow \mathcal{K}(i)$
    return $\mathsf{sk}_i$
```

Fig. 11: The adversary $\mathbb{B}$, playing $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{ROR\text{-}CCA},\not\kappa,1}$ while simulating $\mathsf{Exp}_{\mathrm{PKE}}^{\mathrm{LOR\text{-}CCA},\not\kappa,1}$ for $\mathbb{A}$.

Taken together with Theorem 3, this implies that the left-or-right free-bit notion without corruptions is separated from the single-bit real-or-random notion by at most a factor 2.

Corollary 4 **($\mathrm{ROR}^{\kappa,1} \overset{\leqslant 2}{\Longrightarrow} \mathrm{LOR}^{\kappa,\beta}$).** *There is as SFBB reduction $\mathbb{B}$ such that, for any adversary $\mathbb{A}$,*

$$\mathrm{LOR\text{-}CXA}_{\mathrm{PKE}}^{\kappa,\beta}(\mathbb{A}) \leq 2 \cdot \mathrm{ROR\text{-}CXA}_{\mathrm{PKE}}^{\kappa,1}(\mathbb{B})\,,$$

where $\mathbb{B}$'s overhead consists of drawing $\beta - 1$ uniformly random bits.

Proof (Sketch). Theorem 5 states that $\mathrm{ROR}^{\kappa,1} \overset{\leqslant 2}{\Longrightarrow} \mathrm{LOR}^{\kappa,1}$, while Theorem 3 states that $\mathrm{LOR}^{\kappa,1} \Longrightarrow \mathrm{LOR}^{\kappa,\beta}$, allowing us to conclude that $\mathrm{ROR}^{\kappa,1} \overset{\leqslant 2}{\Longrightarrow} \mathrm{LOR}^{\kappa,\beta}$. □

Given that the free-bit notion generalises the single-bit notion, this in turn implies that LOR and ROR are separated by at most a factor 2 between the corruptionless free-bit notions, even if the number of challenge bits varies between them.

With corruptions, however, any direct simulation would become trivially recognizable—meaning that in order to do a faithful simulation, the reduction would have to guess which bit the adversary is going to attack, leading to a loss linear in β. Instead of reformulating this argument, we let it follow as a corollary to previous results, yielding a slightly tighter statement by letting $\mathbb{B}$ play a single-bit game.

Corollary 5 **($\mathrm{ROR}^{\not\kappa,1} \overset{\leqslant 2\beta}{\Longrightarrow} \mathrm{LOR}^{\not\kappa,\beta}$).** *There is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\mathrm{LOR\text{-}CXA}_{\mathrm{PKE}}^{\not\kappa,\beta}(\mathbb{A}) \leq 2 \cdot \beta \cdot \mathrm{ROR\text{-}CXA}_{\mathrm{PKE}}^{\not\kappa,1}(\mathbb{B})\,.$$

where $\mathbb{B}$'s overhead consists of drawing $\beta - 1$ uniformly random bits.

Proof (Sketch). Theorem 5 states that $\mathrm{ROR}^{\not\kappa,1} \overset{\leqslant 2}{\Longrightarrow} \mathrm{LOR}^{\not\kappa,1}$, while Theorem 4 states that $\mathrm{LOR}^{\not\kappa,1} \overset{\leqslant \beta}{\Longrightarrow} \mathrm{LOR}^{\not\kappa,\beta}$, allowing us to conclude that $\mathrm{ROR}^{\not\kappa,1} \overset{\leqslant 2\beta}{\Longrightarrow} \mathrm{LOR}^{\not\kappa,\beta}$. □

Interestingly, the tightest known relation from the diagonal-bit $\mathrm{ROR}^{\kappa,\boxbslash}$ to $\mathrm{LOR}^{\kappa,\boxbslash}$ loses a factor 2κ, even in the absense of corruptions. This is once again achieved going through the single-user notion.

Corollary 6 **($\mathrm{ROR} \overset{\leqslant 2\kappa}{\Longrightarrow} \mathrm{LOR}^{u,\boxbslash}$).** *There is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\mathrm{LOR\text{-}CXA}_{\mathrm{PKE}}^{u,\boxbslash}(\mathbb{A}) \leq 2 \cdot \kappa \cdot \mathrm{ROR\text{-}CXA}_{\mathrm{PKE}}(\mathbb{B})\,,$$

where $\mathbb{B}$'s overhead consists of generating $\kappa - 1$ fresh keypairs and drawing $\kappa - 1$ uniformly random bits.

Proof (Sketch). It is well established [6] that $\mathrm{ROR} \overset{\leqslant 2}{\Longrightarrow} \mathrm{LOR}$, and we know from Theorem 2 that $\mathrm{LOR} \overset{\leqslant \kappa}{\Longrightarrow} \mathrm{LOR}^{\not\kappa,\boxbslash}$, allowing us to conclude that $\mathrm{ROR} \overset{\leqslant 2\kappa}{\Longrightarrow} \mathrm{LOR}^{\not\kappa,\boxbslash}$. □

5 Conclusion

In this article, we surveyed several possible notions of multi-user security, showing how they relate to each other, and identifying a unified and general free-bit notion. We also conclusively answered the question of which canonical multi-user notion is the preferred one in the absence of corruptions, namely the single-bit left-or-right notion, as it is as strong or stronger than any of the others. In the presence of corruptions, the situation is less clear, particularly as it is not currently known whether or not the ability to both challenge and corrupt a key yields the adversary any additional power. On the other hand, it seems that the ability to challenge the same bit on several keys really *does* give the adversary extra power. Until these questions have been definitively settled, we therefore suggest aiming for security under a free-bit notion whenever multi-user security with adaptive corruptions is to be considered.

References

1. Auerbach, B., Giacon, F., Kiltz, E.: Everybody's a target: Scalability in public-key encryption. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 475–506. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45727-3_16
2. Bader, C., Hofheinz, D., Jager, T., Kiltz, E., Li, Y.: Tightly-secure authenticated key exchange. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part I. LNCS, vol. 9014, pp. 629–658. Springer, Heidelberg (Mar 2015). https://doi.org/10.1007/978-3-662-46494-6_26
3. Bader, C., Jager, T., Li, Y., Schäge, S.: On the impossibility of tight cryptographic reductions. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 273–304. Springer, Heidelberg (May 2016). https://doi.org/10.1007/978-3-662-49896-5_10
4. Baecher, P., Brzuska, C., Fischlin, M.: Notions of black-box reductions, revisited. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 296–315. Springer, Heidelberg (Dec 2013). https://doi.org/10.1007/978-3-642-42033-7_16
5. Bellare, M., Boldyreva, A., Micali, S.: Public-key encryption in a multi-user setting: Security proofs and improvements. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 259–274. Springer, Heidelberg (May 2000). https://doi.org/10.1007/3-540-45539-6_18
6. Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A concrete security treatment of symmetric encryption. In: 38th FOCS. pp. 394–403. IEEE Computer Society Press (Oct 1997). https://doi.org/10.1109/SFCS.1997.646128
7. Bellare, M., Desai, A., Pointcheval, D., Rogaway, P.: Relations among notions of security for public-key encryption schemes. In: Krawczyk, H. (ed.) CRYPTO'98. LNCS, vol. 1462, pp. 26–45. Springer, Heidelberg (Aug 1998). https://doi.org/10.1007/BFb0055718
8. Bellare, M., Hofheinz, D., Kiltz, E.: Subtleties in the definition of IND-CCA: When and how should challenge decryption be disallowed? Journal of Cryptology **28**(1), 29–48 (Jan 2015). https://doi.org/10.1007/s00145-013-9167-4
9. Bellare, M., Hofheinz, D., Yilek, S.: Possibility and impossibility results for encryption and commitment secure under selective opening. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 1–35. Springer, Heidelberg (Apr 2009). https://doi.org/10.1007/978-3-642-01001-9_1
10. Bellare, M., Ristenpart, T., Tessaro, S.: Multi-instance security and its application to password-based cryptography. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 312–329. Springer, Heidelberg (Aug 2012). https://doi.org/10.1007/978-3-642-32009-5_19
11. Bleichenbacher, D.: Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1. In: Krawczyk, H. (ed.) CRYPTO'98. LNCS, vol. 1462, pp. 1–12. Springer, Heidelberg (Aug 1998). https://doi.org/10.1007/BFb0055716
12. Canetti, R., Halevi, S., Katz, J.: Adaptively-secure, non-interactive public-key encryption. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 150–168. Springer, Heidelberg (Feb 2005). https://doi.org/10.1007/978-3-540-30576-7_9
13. Cohn-Gordon, K., Cremers, C., Gjøsteen, K., Jacobsen, H., Jager, T.: Highly efficient key exchange protocols with optimal tightness. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part III. LNCS, vol. 11694, pp. 767–797. Springer, Heidelberg (Aug 2019). https://doi.org/10.1007/978-3-030-26954-8_25
14. Cramer, R., Shoup, V.: A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: Krawczyk, H. (ed.) CRYPTO'98. LNCS, vol. 1462, pp. 13–25. Springer, Heidelberg (Aug 1998). https://doi.org/10.1007/BFb0055717
15. Dwork, C., Naor, M., Reingold, O., Stockmeyer, L.J.: Magic functions. In: 40th FOCS. pp. 523–534. IEEE Computer Society Press (Oct 1999). https://doi.org/10.1109/SFFCS.1999.814626
16. Gay, R., Hofheinz, D., Kiltz, E., Wee, H.: Tightly CCA-secure encryption without pairings. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part I. LNCS, vol. 9665, pp. 1–27. Springer, Heidelberg (May 2016). https://doi.org/10.1007/978-3-662-49890-3_1
17. Giacon, F., Kiltz, E., Poettering, B.: Hybrid encryption in a multi-user setting, revisited. In: Abdalla, M., Dahab, R. (eds.) PKC 2018, Part I. LNCS, vol. 10769, pp. 159–189. Springer, Heidelberg (Mar 2018). https://doi.org/10.1007/978-3-319-76578-5_6
18. Han, S., Liu, S., Gu, D.: Key encapsulation mechanism with tight enhanced security in the multi-user setting: Impossibility result and optimal tightness. In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021, Part II. LNCS, vol. 13091, pp. 483–513. Springer, Heidelberg (Dec 2021). https://doi.org/10.1007/978-3-030-92075-3_17
19. Hazay, C., Patra, A., Warinschi, B.: Selective opening security for receivers. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 443–469. Springer, Heidelberg (Nov / Dec 2015). https://doi.org/10.1007/978-3-662-48797-6_19
20. Hofheinz, D., Hövelmanns, K., Kiltz, E.: A modular analysis of the Fujisaki-Okamoto transformation. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 341–371. Springer, Heidelberg (Nov 2017). https://doi.org/10.1007/978-3-319-70500-2_12
21. Hofheinz, D., Jager, T.: Tightly secure signatures and public-key encryption. Des. Codes Cryptogr. **80**(1), 29–61 (2016)
22. Hofheinz, D., Nguyen, N.K.: On tightly secure primitives in the multi-instance setting. In: Lin, D., Sako, K. (eds.) PKC 2019, Part I. LNCS, vol. 11442, pp. 581–611. Springer, Heidelberg (Apr 2019). https://doi.org/10.1007/978-3-030-17253-4_20
23. Hofheinz, D., Rao, V., Wichs, D.: Standard security does not imply indistinguishability under selective opening. In: Hirt, M., Smith, A.D. (eds.) TCC 2016-B, Part II. LNCS, vol. 9986, pp. 121–145. Springer, Heidelberg (Oct / Nov 2016). https://doi.org/10.1007/978-3-662-53644-5_5

24. Jager, T., Kiltz, E., Riepel, D., Schäge, S.: Tightly-secure authenticated key exchange, revisited. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part I. LNCS, vol. 12696, pp. 117–146. Springer, Heidelberg (Oct 2021). https://doi.org/10.1007/978-3-030-77870-5_5
25. Jager, T., Stam, M., Stanley-Oakes, R., Warinschi, B.: Multi-key authenticated encryption with corruptions: Reductions are lossy. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 409–441. Springer, Heidelberg (Nov 2017). https://doi.org/10.1007/978-3-319-70500-2_14
26. Jager, T., Stam, M., Stanley-Oakes, R., Warinschi, B.: Multi-key authenticated encryption with corruptions: Reductions are lossy. Cryptology ePrint Archive, Report 2017/495 (2017), https://eprint.iacr.org/2017/495
27. Katz, J., Lindell, Y.: Introduction to Modern Cryptography, 2nd Edition. Chapman & Hall/CRC (2015)
28. Lee, Y., Lee, D.H., Park, J.H.: Tightly cca-secure encryption scheme in a multi-user setting with corruptions. Des. Codes Cryptogr. **88**(11), 2433–2452 (2020)
29. Lewko, A.B., Waters, B.: Why proving HIBE systems secure is difficult. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 58–76. Springer, Heidelberg (May 2014). https://doi.org/10.1007/978-3-642-55220-5_4
30. Luykx, A., Mennink, B., Paterson, K.G.: Analyzing multi-key security degradation. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017, Part II. LNCS, vol. 10625, pp. 575–605. Springer, Heidelberg (Dec 2017). https://doi.org/10.1007/978-3-319-70697-9_20
31. Reingold, O., Trevisan, L., Vadhan, S.P.: Notions of reducibility between cryptographic primitives. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 1–20. Springer, Heidelberg (Feb 2004). https://doi.org/10.1007/978-3-540-24638-1_1

A A Brief History of Indistinguishability

The traditional 'IND-CPA' security notion for public key encryption (PKE) is an indistinguishability notion (IND) under adaptively chosen plaintext attacks (CPA). Here an adversary receives a challenge ciphertext either for a plaintext of its choosing or an alternative challenge ciphertext, and needs to decide which one was received. The alternate challenge ciphertext can be generated in different ways, leading to subtly different notions [6]. The two common choices are: left-or-right (LOR), in which the adversary supplies two messages and receives the encryption of one of them; and real-or-random (ROR), in which the adversary supplies a message and either receives its encryption or the encryption of a random bit string. When Bellare et al. [7] considered various PKE security notions they showed that LOR-security tightly implies ROR-security, whereas the other direction incurs a modest security loss of a factor 2.

Stronger, more realistic notions are indistinguishability under adaptively chosen ciphertext attacks (CCA), or IND-CCA (historically also called IND-CCA2 to distinguish it from its non-adaptive counterpart IND-CCA1 [7]). Here, in addition to choosing the plaintexts to be challenged on, the adversary is given access to a decryption oracle, which it can query on any valid ciphertext receiving the corresponding plaintext, or the ciphertext-reject symbol $\perp$. To avoid trivial wins, some care needs to be taken when the challenge ciphertext is submitted to the encryption oracle; there are several mechanisms to deal with this subtlety [8]. Ignoring the decryption oracle gives back IND-CPA, making IND-CCA the stronger notion. Moreover, several real-world attacks not covered by IND-CPA (such as Bleichenbacher's attack [11]) are captured by IND-CCA, making the latter the preferred notion to aim for.

We are concerned with the multi-user setting, leading to further definitional choices. Although it might appear that these choices are largely irrelevant in an asymptotic context, as they are all polynomially equivalent, a concrete security treatment can surface non-trivial differences. These differences are often amplified with the introduction of multiple users, particularly when considering adaptive corruptions (see below).

First of all, while the notions above initially only allowed for a single challenge query, when Bellare et al. [5] investigated multi-user security, they simultaneously generalized the single-user notions by giving each user multiple challenges. Moreover, they showed that security under single-challenge implies security under multi-challenge with an (inevitable) loss linear in the number of challenges (cf. [6]).

In the present work, we consider all notions, including the single-user notions, to be multi-challenge. To adapt our results to a single-challenge setting, simply note that our single-user notions imply the corresponding single-challenge notion with a tightness loss $q^{\mathcal{E}}$, and insert the factor as needed. For instance, writing SC-IND for single-challenge indistinguishability, the analogue to Corollary 3 becomes $\text{SC-IND} \xRightarrow{\leqslant q^{\mathcal{E}} \kappa\beta} \text{IND}^{\not\kappa,\beta}$.

Another choice is how to 'multiplex' the challenge oracles: should each user be independent of the others, or should they depend on each other? When multi-user security was introduced [5], the game only had a single challenge bit shared across all users for an adversary to guess. This choice intuitively leads to a stronger notion than if each user was given its own challenge bit as, with a single shared bit, an adversary can 'gather evidence' for the true value of this challenge bit across the users (we provide evidence to this intuition in Corollary 1). Yet, the notion feels awkward when introducing corruptions, given that both corrupting and challenging on the same key would immediately yield a trivial win. One option is to disallow corrupting ‘challenge’ keys, and vice versa, leading to the single-bit notion $\mathrm{IND}^{\not k,1}$ [3, 24, 28]. Another option is to introduce user-specific bits. This was the approach employed by Bader et al. [2] in their study of authenticated key exchange: they considered a multi-user KEM notion with corruptions where each user was associated with its own challenge bit and the adversary had to declare at the end which uncorrupted bit it was guessing. Thus, even if a user was both challenged on and corrupted, a non-trivial win would still be possible. In the present work, we refer to this notion as *diagonal-bit* ($\mathrm{IND}^{\not k,\boxbslash}$), as explained in Section 3.

Recently, Jager et al. [24] pointed out that this notion is problematic in the AKE setting, as unlike in the single-bit setting, a KEM secure under the diagonal-bit notion is not known to tightly compose to an AKE. They went on to construct a KEM tightly secure under the single-bit notion instead, which was therefore guaranteed to compose tightly.

Apart from the multi-user setting, the diagonal notion has seen use in the multi-*instance* setting [1, 10], in which the adversary is asked to make a guess on *every* bit; in such settings, single-bit notions make little sense.

When Jager et al. [25] investigated the inevitability of multi-user tightness losses in the setting of authenticated encryption, they wanted their result to capture both the single-bit and the diagonal-bit notions, without having to provide separate proofs for the distinct cases. They therefore introduced a generalized notion, in which an adversary was free to choose the exact relations between the keys and challenge bits. This notion, which avoids the awkwardness of not being able to both challenge and corrupt the same key without sacrificing the ability to “gather evidence” on a bit over several keys, sits at the centre of much of the present work, and we refer to it as the *free-bit* notion ($\mathrm{IND}^{\not k,\beta}$).

B Sharpness or When Tightness Losses are Inevitable

B.1 Sharpness and Inevitably Lossy Reductions

A natural question for lossy reductions is whether the loss is inevitable or not. To determine inevitability, we only need to ‘invert’ Definition 2, as below in Definition 3.

Definition 3 (Lossy). *Let* IND_1 *and* IND_2 *be two indistinguishability notions for PKE schemes, and let* c *be a positive real number, then* $\mathrm{IND}_1 \overset{\geq c}{\Longrightarrow} \mathrm{IND}_2$ *iff for all simple fully-black box reductions* $\mathbb{B}_1$ *there exist PKE schemes* PKE *and adversary* $\mathbb{A}_2$*,*

$$\mathrm{IND}_2(\mathbb{A}_2) \geq c \cdot \mathrm{IND}_1(\mathbb{B}_1^{\mathbb{A}_2,\mathrm{PKE}}) .$$

If both $\mathrm{IND}_1 \overset{\leq c}{\Longrightarrow} \mathrm{IND}_2$ and $\mathrm{IND}_1 \overset{\geq c}{\Longrightarrow} \mathrm{IND}_2$, then the reduction (for the first term) is sharp and we may write $\mathrm{IND}_1 \overset{=c}{\Longrightarrow} \mathrm{IND}_2$.

B.2 Sharpness of Single-to-Simple Reductions

Below we discuss some relevant methods and results regarding the inevitability of lossy reductions in the context of multi-user PKE, showing that linear losses (in the number of users) is often sharp. Such results are often called impossibility results, yet to contrast with impossibility results that show that no constructions can achieve a notion (irrespective of the lossiness of the reduction), we prefer the term sharpness result when the impossibility is restricted to tightness only. The two main techniques are counterexamples and meta-reductions.

Counterexamples. As already pointed out by Bellare et al. [5], a simple counterexample shows that the bounds are generally sharp. They modified a PKE scheme that was identical to a ‘secure’ one except that with a small probability its encryption would be trivial and essentially just output the plaintext as the ciphertext (with some additional modifications to ensure correctness and that this event is easily recognizable publicly). Thus, when the challenge encryption oracle hits the trivial encryption, an adversary can trivially win its game; moreover the probability of this event happening at some point during the game is roughly linear with the number of challenge encryption queries.

However, given that we consider all our notions to be multi-challenge, we prefer a counterexample whose security degrades linearly in the number of available *keys*, not challenges. One might therefore instead consider a scheme for which a small-but-nonempty subset of the public keyspace returns messages in the clear. This “weak key” counterexample already works without corruptions for both of the simple multi-user notions, which implies sharpness for the more general notions.

Note that a similar critique of Bellare et al.’s original counterexample and (more refined) link with weak keys was made by Luykx et al. [30].

Meta-reductions. Another line of work has aimed to show sharpness through meta-reduction, thus ruling out tight reductions for a larger class of PKE schemes. The gain in generality is however traded for restrictions on the type of reductions that are ruled out, typically referred to as "simple" reductions (e.g., blackbox, no rewinding, etc.).

Bader et al. [3] showed that, for a large class of PKE systems, any simple reduction from a multi-user notion with corruptions to an underlying non-interactive hardness assumption must be lossy, with the loss linear in the number of keys. Meanwhile, Jager et al. [25] showed a similar result in the setting of authenticated encryption when reducing to single-user notions. In both cases, though, the proof technique crucially relied on the ability to corrupt keys, meaning that sharpness for the corruptionless notions aren't covered by their results.

Meta-reductions also don't rule out tight reductions for schemes outside the class considered; in fact, part of the usefulness of these results is the ability to look for tightly secure constructions outside these classes. This is exactly what Bader et al. [2] did when they constructed a tightly secure authenticated key exchange by deliberately breaking the requirement of public–private key uniqueness.

B.3 Tightening the Single-to-Free Implication?

Corollary 3, IND $\xRightarrow{\leqslant\kappa\beta}$ IND$^{\not{k},\beta}$, tightly implies Theorem 1, but not Theorem 2: setting $\kappa = \beta$ in Corollary 3 yields a κ^2 loss. This gives some hope that we might be able to show a tighter relation than that of Corollary 3, as in order to imply both Theorem 1 and 2, the statement would have to look something like the following.

*Conjecture 1 (*IND $\xRightarrow{\leqslant \mathtt{I}^{\mathcal{E}}_{\max}\beta}$ IND$^{\not{k},\beta}$*). Let* $\mathtt{I}^{\mathcal{E}}_{\max}$ be the maximum number of keys called together with any one challenge bit, (i.e., for any run of the game, we now require that $\forall j, |\mathtt{I}_j| \leq \mathtt{I}^{\mathcal{E}}_{\max}$; see Fig. 1). Then, there is a reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,

$$\text{IND-CXA}^{\not{k},\beta}_{\text{PKE}}(\mathbb{A}) \leq \mathtt{I}^{\mathcal{E}}_{\max} \cdot \beta \cdot \text{IND-CXA}_{\text{PKE}}(\mathbb{B})\,,$$

where the overhead of $\mathbb{B}$ is small.

Then, for IND$^{\not{k},1}$, we would set $\mathtt{I}^{\mathcal{E}}_{\max} = \kappa$ and $\beta = 1$, while for IND$^{\not{k},\boxbslash}$, $\mathtt{I}^{\mathcal{E}}_{\max} = 1$ and $\beta = \kappa$. Thus, both theorems are recovered.

To prove the statement, a natural strategy would be to combine the proof techniques of each of the theorems it is generalising, i.e. by first guessing a challenge bit, and then doing a hybrid argument over the keys relating to that bit. However, given that the free-bit game allows the adversary to choose the relations between keys and bits adaptively, this hybrid argument does not work without incurring losses larger than that of Corollary 3. We nevertheless present Conjecture 1 as an interesting open problem.

C Multi-Bit Composability of Hybrid Encryption

As shown by Cramer and Shoup [14], one can combine the practicality of asymmetric encryption with the efficiency of symmetric encryption into a highly efficient public key encryption system. The idea is to encrypt the message under an ephemeral symmetric key, which is itself encapsulated under a public key. This paradigm, which already saw widespread use at the time, has become known as the KEM/DEM paradigm, after its constituent Key Encapsulation Mechanism and Data Encapsulation Mechanism; it is also known as *hybrid* encryption.

Recently, Lee et al. [28] built on earlier work by Giacon et al. [17] and showed that a KEM and a DEM tightly compose to a PKE in a single-bit multi-user setting with corruptions. We paraphrase their result in Theorem 6.

Theorem 6 (Lee, Lee, Park, DCC'20). *There are SFBB reductions* $\mathbb{B}$ *and* $\mathbb{C}$ *such that, for every adversary* $\mathbb{A}$,

$$\text{LOR-CXA}^{\not{k},1}_{\text{PKE}}(\mathbb{A}) \leq 2 \cdot \text{ROR-CCA}^{\not{k},1}_{\text{KEM}}(\mathbb{B}) + \text{1LOR-CCA}_{\text{DEM}}(\mathbb{C})\,.$$

Here, 1LOR means "one-time left-or-right"; see their paper for definitions and proof. Combining their result with Theorem 4 yields the following, more general, corollary.

Corollary 7 (Free-bit composability). *There are SFBB reductions* $\mathbb{B}$ *and* $\mathbb{C}$ *such that, for every adversary* $\mathbb{A}$,

$$\text{LOR-CXA}^{\not{k},\beta}_{\text{PKE}}(\mathbb{A}) \leq 2 \cdot \beta \cdot \text{ROR-CCA}^{\not{k},1}_{\text{KEM}}(\mathbb{B}) + \beta \cdot \text{1LOR-CCA}_{\text{DEM}}(\mathbb{C})\,.$$

Proof. Immediately follows from Theorems 4 and 6. □

While lossy in the number of challenge bits, it matches Lee et al.'s Theorem for $\beta = 1$. However, the implication to the diagonal-bit notion, with $\beta = \kappa$, results in a rather lossy composition, as made explicit below.

Corollary 8 (Diagonal-bit composability). *There are SFBB reductions* $\mathbb{B}$ *and* $\mathbb{C}$ *such that, for every adversary* $\mathbb{A}$*,*

$$\text{LOR-CXA}_{\text{PKE}}^{\not\kappa,\boxslash}(\mathbb{A}) \leq 2 \cdot \kappa \cdot \text{ROR-CCA}_{\text{KEM}}(\mathbb{B}) + \kappa \cdot \text{1LOR-CCA}_{\text{DEM}}(\mathbb{C})\,.$$

Proof. Follows from Theorems 2 and 6. □

No tighter composition is known for multi-bit security notions, for much the same reason that no tight composition is known for AKE: as pointed out by Jager et al. [24], the multi-bit KEM notion does not easily allow for a game hop in which real keys are exchanged for fake ones, making the simulated game be something in between the 'real' and 'random' worlds. Any attempt to circumvent this issue (without specializing to specific constructions) seems to lead to hybrid or guessing arguments, yielding similar linear losses.

Article II

Multi-Instance Secure Public-Key Encryption

Carlo Brunetta, Hans Heum and Martijn Stam

In Boldyreva, A., Kolesnikov, V., editors, *Public-Key Cryptography PKC 2023*, Lecture Notes in Computer Science, vol 13941. Springer, Cham, May 2023. Presented by the candidate at PKC in May 2023.

Version reproduced here in Cryptology ePrint Archive, Report 2022/909, 2022.
`https://eprint.iacr.org/2022/909`

The candidate served as lead author of the work.

Multi-Instance Secure Public-Key Encryption [*]

Carlo Brunetta[1], Hans Heum[2], and Martijn Stam[1]

[1] Simula UiB,
Merkantilen (3rd floor)
Thormøhlensgate 53D
N-5006 Bergen, Norway.
carlob,martijn@simula.no

[2] Department of Mathematical Sciences, NTNU - Norwegian University of Science and Technology, Trondheim, Norway.
hans.heum@ntnu.no[**]

Abstract. Mass surveillance targets many users at the same time with the goal of learning as much as possible. Intuitively, breaking many users' cryptography simultaneously should be at least as hard as that of only breaking a single one, but ideally security degradation is gradual: an adversary ought to work harder to break more. Bellare, Ristenpart and Tessaro (Crypto'12) introduced the notion of multi-instance security to capture the related concept for password hashing with salts. Auerbach, Giacon and Kiltz (Eurocrypt'20) motivated the study of public key encryption (PKE) in the multi-instance setting, yet their technical results are exclusively stated in terms of key encapsulation mechanisms (KEMs), leaving a considerable gap.

We investigate the multi-instance security of public key encryption. Our contributions are twofold. Firstly, we define and compare possible security notions for multi-instance PKE, where we include PKE schemes whose correctness is not perfect. Secondly, we observe that, in general, a hybrid encryption scheme of a multi-instance secure KEM and an arbitrary data encapsulation mechanism (DEM) is unlikely to inherit the KEM's multi-instance security. Yet, we show how with a suitable information-theoretic DEM, and a computationally secure key derivation function if need be, inheritance is possible. As far as we are aware, ours is the first inheritance result in the challenging multi-bit scenario.

Keywords: Multi-Instance Security · Hybrid Encryption · Property Inheritance · Mass Surveillance

[*] This article is the full version of an earlier article: PKC 2023, ©IACR 2023.

[**] Work by Hans Heum performed as part of his PhD studies at Simula UiB.

1 Introduction

Security of cryptographic schemes is increasingly studied concretely. The question changes from whether a scheme is secure or not, to how secure it is. The change in emphasis also results in increased importance in more realistic security notions that model a world where an adversary might have many potential targets. If an adversary simply tries to learn something about one of its κ targets, then intuitively the more targets there are, the easier the adversary's job becomes. Indeed, using simple hybrid arguments results in a security degradation that is linear in κ. But what happens if the adversary is greedy and wants to learn more, maybe even targets everyone? On the one hand, one could argue that if breaking one instance is hard, then so is breaking many. Yet, on the other hand, one would hope that breaking multiple instances, say n, is strictly harder than breaking just a single one.

This second perspective made Bellare, Ristenpart and Tessaro [12], henceforth BRT, realize that new security notions are needed to reason about such greedy adversaries. They were motivated by how salts in password hashing protect against attackers re-using precomputation to retrieve multiple passwords. For their study into probabilistic symmetric schemes, they identified left-or-right indistinguishability under xor as the strongest notion. Roughly speaking, there are κ keys in the system each associated with its own left-or-right challenge bit b_i and the goal of the adversary is to guess the xor of all those bits.

Recently, Auerbach, Giacon and Kiltz [4], henceforth AGK, argued the importance of BRT's concept to protect against mass surveillance. They introduced the (n, κ) scaling factor as the effort to break n out of κ instances relative to the effort needed to break a single instance. After recalling several well-known greedy attacks against public key schemes with dubious scaling factors, they set out to provide an encryption scheme with good, non-trivial scaling factor.

They discussed various versions of Hashed ElGamal that differed in whether users shared group parameters and/or generators, plus whether the underlying group was elliptic curve or finite field based. In the programmable random oracle model, they showed that the multi-instance security of Hashed ElGamal tightly relates to a novel multi-instance Gap Computational Diffie–Hellman (MI-GapCDH) assumption, whose validity was further supported by an analysis in the generic group model.

There was, or rather is, just one small problem: Hashed ElGamal is a key encapsulation mechanism (KEM), not a public key encryption (PKE) scheme. Indeed, although AGK use PKE as their motivation, their formalization is entirely centred around KEMs. Of course, Cramer and Shoup [19] already showed how a secure KEM can be combined with a secure data encapsulation mechanism (DEM) to create a secure PKE (for various notions of security). This so-called hybrid encryption paradigm is widely deployed in the real world, yet, can its composition theorem be easily lifted to the multi-instance setting?

For key unrecoverability, all seems fine, but for indistinguishability one quickly uncovers various challenges. Consider an adversary $\mathbb{A}$ that wants to recover n out of κ challenge bits b_i: it can attempt to recover roughly half of its b_i by somehow breaking the DEM, and recovering the remaining half by breaking the KEM. Intuitively, such a divide-and-conquer strategy essentially rules out inheriting full multi-instance security of both KEM and DEM simultaneously. Instead, perhaps we should aim to bound an adversary's multi-instance advantage against the hybrid encryption in terms of either breaking the full multi-instance security of the KEM or breaking only one of many instances of the DEM.

Special care would have to be taken to ensure that the corresponding multi-user DEM advantage is not overwhelming the multi-instance KEM advantage. After all, already when showing multi-user security of hybrid encryption, ensuring the DEM advantage does not overshadow the multi-user KEM advantage is challenging [23]. Furthermore, the study of multi-user KEMs highlights a second, more technical problem.

For multi-user security, there are essentially two different formalizations possible: one where each user comes with its own challenge bit and one where the users share a global challenge bit. Jager et al. [29] recently observed that only the latter lends itself to an easy adaptation of composition theorems using KEMs, as it allows a simple game-hop where all KEM-derived ephemeral keys are replaced by randomly selected keys (decoupled from the KEM encapsulations). That proof technique fails when there are multiple challenge bits. Unfortunately, for multi-instance security, the only option available is a notion with multiple challenge bits. In such a setting, inheritance of security properties of the KEM to any construction based on the KEM is an open problem.

Our Contribution. As mentioned above, multi-instance security was introduced by BRT in the context of probabilistic symmetric primitives and later adapted to key encapsulation mechanisms by AGK, who provide an excellent motivation for the study of multi-instance security in a public key setting. We adapt those notions to multi-instance security for PKE schemes, but make a number of non-trivial changes in the process. Firstly, we observe that the mechanisms used by BRT and AGK to model multi-instance games differ, which seems to have gone unnoticed hitherto. BRT's mechanism is stronger as it allows for corruptions (denoted by $\star$), yet

$$\begin{tikzcd}
\text{IND-CCA}^{\star} \arrow[r, "\text{Thm. 2}", "\cdot c^{-1}"'] \arrow[d, bend left, "\text{Sect. 3}"] & \text{MKU-CCA}^{\star} \arrow[d, bend right, "\text{Thm. 1}"', "+\kappa\gamma"] \\
\text{ROR-CCA}^{\star} \arrow[u, bend left, "\text{Cor. 2}", "\cdot 2^{n}\binom{\kappa}{n}"'] & \text{UKU-CCA}^{\star} \arrow[u, bend right, "/" marking]
\end{tikzcd}$$

Fig. 1. An overview of multi-instance security notions for public-key encryption, where γ relates to imperfect correctness (Def. 1), and the loss factor c is explained in Thm. 2.

AGK's mechanism is more expressive by making explicit how many instances an adversary should break. We use elements of both in our notions, incorporating both BRT's corruptions and AGK's explicit expression of the number of targeted instances. Secondly, we allow for correctness to be imperfect, which has ramifications for how to deal with decryption oracles (for chosen-ciphertext attacks) and corruptions. We delve into the differences between the various mechanisms in Sect. 3.3, furthermore we use our revised mechanism to study a number of related notions, as summarized in Fig. 1.

In more detail, we start out by porting BRT's notion of key unrecoverability to the public-key setting. In fact, we consider two distinct versions of key unrecoverability: "Universal Key Unrecoverability" (UKU), where the adversary is tasked to recover the exact challenge private key(s) and "Matching Key Unrecoverability" (MKU), where it suffices to recover suitably equivalent private keys, where we leverage our imperfect correctness notion to define "suitably equivalent". As one would expect, this relaxed key unrecoverability notion implies the stronger, exact notion up to a small loss related to how we model imperfect correctness (Thm. 1).

For our main notion of multi-instance security, we follow BRT's identification of left-or-right xor-indistinguishability as the strongest notion and adapt it to the public key setting. As for the symmetric encryption setting studied by BRT, this indistinguishability notion implies the above key unrecoverability notions (Thm. 2); however, the differences between perfect symmetric encryption and imperfect PKE affect the corresponding implications and their proofs.

Finally, we explore an alternative notion, namely real-or-random xor-indistinguishability (ROR). Trivially, left-or-right tightly implies real-or-random and in the multi-instance setting BRT showed that the usual factor-2 loss from the single instance implication between real-or-random to left-or right, becomes an exponential factor-2^{κ} loss. A similar loss is possible in our setting, however, we can also achieve a typically preferable bound of $\binom{\kappa}{n}2^{n}$ (Cor. 2).

With suitable notions for multi-instance PKE available, we focus on how to turn a suitably multi-instance secure KEM into a multi-instance secure PKE scheme using hybrid encryption. For key unrecoverability, inheritance is immediate, yet we would like to guarantee good multi-instance indistinguishability (the left-hand branch of Fig. 1). We summarize our findings in Fig. 2.

Our first observation is that we can expand the length of the ephemeral key to any desired length using a pseudorandom extendable output function (XOF). The resulting extendable KEM, or XEM, inherits the multi-instance security of the underlying KEM, provided the XOF is secure against multi-challenge adversaries (Thm. 5). To ensure that the XOF does not become the weakest link, its seed will need to be long enough, which in turn implies that the underlying KEM already needs to output a sufficiently long ephemeral key.

The XOF above of course plays the role of key derivation function, but it is more common that it is modelled as part of any key expansion done by the DEM. Moving it into the KEM allows us to use an information-theoretic DEM, read one-time pad (OTP), irrespective of the message length. The OTP's properties enable a simplified proof for the security of hybrid encryption (Thm. 6), where the PKE does indeed inherit the multi-instance security of the XEM, with two important caveats. Firstly, the OTP is only passively secure, so the PKE only achieves CPA not CCA security, and secondly, standard KEM indistinguishability only tightly provides real-or-random indistinguishability for the PKE (see the top line of Fig. 2).

Switching to the TagKEM framework [2], or in our case TagXEM, takes care of the first shortcoming and tightly achieves multi-instance ROR-CCA secure PKE, or IND-CCA non-tightly (Thm. 7). For the PKE to inherit multi-instance IND-CCA security tightly, we introduce a novel KEM indistinguishability notion that more closely matches PKE's left-or-right idea, namely real-or-permuted (ROP). Finally, we can show tight multi-instance inheritance for the most desirable PKE notion, based on a ROP-secure TagXEM (Thm. 8).

One small hiccough remains, as our KEM-to-XEM result unfortunately only works for classical KEM indistinguishability, not for ROP indistinguishability, nor does it look feasible to convert a KEM or XEM to a TagKEM or TagXEM, respectively, inheriting multi-instance security using standard reductions. Here, the random oracle,

$$\begin{tikzcd}
\mathrm{IND\text{-}CCA}^{\star}_{\mathrm{KEM}} \arrow[r, "\text{Thm. 5}", "\text{+ XOF}"'] & \mathrm{IND\text{-}CCA}^{\star}_{\mathrm{XEM}} \arrow[r, "\text{Thm. 6}", "\text{+ OTP}"'] & \mathrm{ROR\text{-}CPA}^{\star}_{\mathrm{PKE}} \\
\mathrm{MI\text{-}GapCDH}^{\star} \arrow[u, dotted, "\text{AGK [4]}", "\text{+ RO}"'] \arrow[d] & \mathrm{IND\text{-}CCA}^{\star}_{\mathrm{TXEM}} \arrow[r, "\text{Thm. 7}", "\text{+ OTP}"'] & \mathrm{ROR\text{-}CCA}^{\star}_{\mathrm{PKE}} \\
\mathrm{OW\text{-}PCA}^{\star}_{\mathrm{KEM}} \arrow[r, "\text{Thm. 9}", "\text{+ MAC + RO}"'] & \mathrm{ROP\text{-}CCA}^{\star}_{\mathrm{TXEM}} \arrow[u, "\text{Lemma 4}"] \arrow[r, "\text{Thm. 8}", "\text{+ OTP}"'] & \mathrm{IND\text{-}CCA}^{\star}_{\mathrm{PKE}}
\end{tikzcd}$$

Fig. 2. An overview of our constructions achieving various flavours of multi-instance security. The left upwards arrow is dotted, as AGK did not consider corruptions.

as used by AGK to prove their construction secure, comes to the rescue, although rather than looking at Hashed ElGamal under the MI-GapCDH assumption, we consider more general KEMs that are multi-instance one-way under plaintext checking attacks (unfortunately, also at this point we need to restrict to perfect correctness), which we combine with Abe et al.'s TagKEM construction from a KEM and a MAC (message authentication code).

Recalling that the original random oracle [14] was in fact a XOF, we can bake the extendability into the random oracle, including the key needed for an information-theoretic secure MAC. Moreover, the power of the ROM allows proving the stronger ROP indistinguishability just as easily as classical KEM indistinguishability. All in all, with Thm. 9 we achieve a suitably multi-instance secure TagXEM based on a KEM that can be instantiated by Hashed ElGamal. In that case, the security relies on the $\mathrm{MI\text{-}GapCDH}^{\star}$ assumption, i.e. with corruptions. As an added benefit of using the random oracle, the resulting multi-instance bounds no longer rely on sufficiently long XOF inputs, thus for determining a suitable group size (when instantiating by Hashed ElGamal) the $\mathrm{MI\text{-}GapCDH}^{\star}$ advantage is leading.

For low granularity, which corresponds to a setting where every user generates its own group as part of its public key, AGK's technique can easily be extended to include corruptions and in the generic group model we arrive at the same bound for the hardness of $\mathrm{MI\text{-}GapCDH}^{\star}$, so with corruptions, as AGK did without corruptions. Unfortunately, for the more realistic high granularity setting, where users share the same (standardized) group, AGK's proof strategy does not easily allow incorporating corruptions. We provide details in App. C.

Thus, we can conclude that XOF-based Hashed ElGamal combined with a suitable information-theoretically secure MAC and the one-time-pad, provides good multi-instance security in the programmable random oracle model and generic group model, provided that users each select their own independent group. We briefly touch upon a concrete interpretation in App. D, where we also informally address AGK's scaling factor.

Related Work. Farshim and Tessaro [20] recently followed up BRT's line of work on the multi-instance security of password hashing by combining it with the related preprocessing setting. AGK [4] motivated their investigation into multi-instance security by the threat of mass surveillance. The latter had previously motivated Bellare et al. [11] to consider subversion, namely the ease with which a "big brother" might subvert an encryption algorithm by replacing it surreptitiously with a trapdoored one with otherwise identical behaviour.

The multi-instance setting is closely related to the multi-user setting, in which the adversary is tasked with breaking only one rather than n out of κ possible instances. Multi-user security was introduced by Bellare et al. [7] in the public-key setting, with the goal of deriving concrete security parameters in a more realistic setting. There have been many recent follow-up works, including how the hybrid paradigm generalizes to the setting without corruptions [23], and later with corruptions [33], as well as the construction of tightly-secure authenticated key exchange (AKE) from multi-user KEMs [29]. Various versions of the multi-user GapCDH problem with corruptions were recently proposed and analysed in that context [30].

One definitional subtlety of multi-user security is the number of challenge bits: either a single one, as originally conceived, or many, as typical for the multi-instance setting. The various definitions do not appear to imply each other tightly [26], which slightly hinders regarding the multi-user setting as a special case of the multi-instance setting (due to potential tightness losses).

2 Preliminaries

2.1 Notation

For a positive integer n, we write $[n]$ for the set $\{1, \ldots, n\}$. We use code-based experiments, where $\leftarrow$ denotes deterministic assignment and $\leftarrow\!\!\$$ denotes probabilistic assignment. By convention, all sets and lists are initialized empty. For a set $\mathtt{X}$, we use the shorthand $\mathtt{X} \xleftarrow{\cup} x$ for the operation $\mathtt{X} \leftarrow \mathtt{X} \cup \{x\}$. If $\mathtt{X}$ is a list, then $\mathtt{X} \xleftarrow{\frown} x$ denotes appending the element x to $\mathtt{X}$; to retrieve the ith element of the list, we write $\mathtt{X}[i]$ where by convention $\mathtt{X}[i] = \emptyset$ for out-of-bounds i.

We use $\Pr[Code : Event \,|\, Condition]$ to denote the conditional probability of $Event$ occurring when $Code$ is executed, conditioned on $Condition$. We omit $Code$ when it is clear from the context and $Condition$ when it is not needed. For Boolean values, we use $\{\mathsf{true}, \mathsf{false}\}$ and $\{0, 1\}$ interchangeably, where by convention 1 corresponds to true.

When proving relations between notions and security of constructions, we will often refer to simple fully black box (SFBB) reductions. A reduction is fully black box iff it works for all schemes and adversaries, and only accesses them in a black box manner [6, 38] (we leave the black box dependence implicit in our notation). Moreover, if the reduction only runs its adversary once and without rewinding, then the reduction is simple [34].

Finally, the respective games that the adversary and the reduction are playing often have matching (though not identical) oracles; for instance, both may have access to a decryption oracle or a key corruption oracle. We call a reduction type-preserving with respect to, say, a decryption oracle iff the reduction will make decryption queries iff its black-box adversary makes decryption queries. Type-preservation, without explicit mention of any oracles, is implicitly meant to imply for all meaningfully matching oracles (unless otherwise specified).

Type-preservation of reductions appears folklore and can easily be established by inspection. Intuitively, a type-preserving reduction can be used to show simultaneously that CCA security of some kind implies CCA security of another kind and that CPA security of the same kind implies CPA security of the other kind. In Sect. 3.3 we will encounter several reductions that are only partially type-preserving.

2.2 PKE Syntax

A public-key encryption scheme PKE consists of four algorithms: the probabilistic key generation algorithm $\mathsf{PKE.Kg}$, which takes as input some system parameter pm (see also Remark 1) and outputs a public/private key pair $(\mathsf{pk}, \mathsf{sk})$; the deterministic key validation algorithm $\mathsf{PKE.Check}$, which takes as input the system parameters pm as well as a purported public/private key pair $(\mathsf{pk}, \mathsf{sk})$ and returns true or false (see Remark 2 below), the probabilistic encryption algorithm $\mathsf{PKE.Enc}$, which on input a public key pk and a message $m \in \mathcal{M}$ (see Remark 3), outputs a ciphertext c; and the deterministic decryption algorithm $\mathsf{PKE.Dec}$, which on input of a secret key sk and a ciphertext c, outputs either a message m, or a special symbol $\perp$ denoting failure.

Remark 1. The system parameters pm are implicitly input to $\mathsf{PKE.Enc}$ and $\mathsf{PKE.Dec}$ as well; for concreteness, they can for instance be the description of an elliptic curve group with generator for an ECDLP-based system or the dimensions and noise sampling algorithm for an LWE-based system. When one is interested in re-phrasing our results in an asymptotic setting, the parameters pm will be generated by a probabilistic, polynomial-time algorithm that only takes the security parameter as input.

Remark 2. For various modern cryptosystems, especially schemes targeting post-quantum security or tight multi-user security, the relationship between public and private keys is not one-to-one. For instance, a single public key can have various private keys [23] or a single private key can lead to various public keys [16]. Naively, one could check whether a public key and private key belong together by simply verifying whether encrypting and then decrypting a number of random messages always returns the original messages. With imperfect correctness, such a canonical checking algorithm can produce both false positives and false negatives. Yet, it is usually still possible to ckeck whether a private–public key pair matches more directly, which we model by the key validation algorithm $\mathsf{PKE.Check}$. We will define both correctness and key unrecoverability in terms of this key validation algorithm.

Remark 3. The message space $\mathcal{M}$ may depend on the parameters pm, but for simplicity we assume it independent of the public key pk. Often $\mathcal{M}$ consists of arbitrary length bitstrings, or at least all bitstrings up to some large length (e.g. 2^{64}) and messages of the same length are deemed equivalent as they are expected to yield ciphertexts of identical lengths. We will model these equivalences more abstractly by assuming that pm implicitly defines a number $\mathfrak{m}$ of equivalence classes, together with an efficient method $[\![\cdot]\!] : \mathcal{M} \to [\mathfrak{m}]$ to determine the class (e.g. length) of a message and an efficient algorithm to sample uniformly from a given equivalence class. We write $\sim$ for the equivalence, so for $m \in \mathcal{M}$, $m \sim m'$ iff $[\![m]\!] = [\![m']\!]$.

Correctness. Perfect correctness states that for all parameters pm, all key pairs $(\mathsf{pk}, \mathsf{sk})$ that can be output by $\mathsf{PKE.Kg}(\mathsf{pm})$, and all messages $m \in \mathcal{M}$, we always have that $\mathsf{PKE.Dec}_{\mathsf{sk}}(\mathsf{PKE.Enc}_{\mathsf{pk}}(m)) = m$. Yet modern schemes, especially lattice-based ones, often allow a small decryption error, where occasionally decryption will fail or it will return a wrong message.

Various relaxations of correctness have appeared in the literature in order to argue about such schemes as it turns out that some classical results implicitly or subtly relied on perfect correctness. In order for our work to be meaningful for a large range of both classical and modern schemes, we introduce a stronger version of imperfect correctness based on the key validation algorithm.

Definition 1 **((γ, δ)-Correctness).** *Let $\gamma, \delta \in [0, 1]$. Then a public-key encryption scheme PKE is called (γ, δ)-correct iff for all* pm*,*

1. $\Pr[(\mathsf{pk}, \mathsf{sk}) \leftarrow\!\$ \; \mathsf{PKE.Kg}(\mathsf{pm}) \;:\; \mathsf{PKE.Check}(\mathsf{pm}, \mathsf{pk}, \mathsf{sk}) = \mathsf{false}] \leq \gamma$*;*
2. *for all* $(\mathsf{pk}, \mathsf{sk})$ *and all* $m \in \mathcal{M}$*, if* $\mathsf{PKE.Check}(\mathsf{pm}, \mathsf{pk}, \mathsf{sk}) = \mathsf{true}$ *then*

$$\Pr[\mathsf{PKE.Dec}_{\mathsf{sk}}(\mathsf{PKE.Enc}_{\mathsf{pk}}(m)) \neq m] \leq \delta \; .$$

Perfect correctness corresponds to $(0, 0)$-correctness and any scheme is trivially both $(1, 0)$-correct and $(0, 1)$-correct. For good schemes γ and δ can simultaneously be chosen small, where typically increasing γ allows for decreasing δ. As we will see, both γ and δ will appear in various bounds, thus allowing larger γ to enable smaller δ (or vice versa) might give preferable bounds.

3 Multi-Instance Security of Public-Key Encryption

3.1 Two Flavours of Key Recovery

The minimal requirement for public-key encryption schemes is that, given a public key, it should be difficult to recover the private key. Although key unrecoverability is a very weak notion theoretically, its study has two main motivations: firstly, many multi-instance attacks target key recovery, and secondly, conceptually the notion is relatively simple, allowing both an instructive introduction of formalizing multi-instance security and an initial comparison between BRT's perfect symmetric encryption and our imperfect public key encryption.

At first sight, the generalization to the multi-instance setting appears immediate: an adversary tries to recover the respective private keys for a number of public keys. BRT introduced universal key unrecoverability (UKU) as a suitable notion for multi-instance security of symmetric encryption. We provide an analogue for public-key encryption, but there are some crucial changes in the game's mechanics (see also Sect. 3.3).

Let $0 < n \leq \kappa$ be integer parameters, then the universal key unrecoverability experiment $\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A})$ for public-key encryption scheme PKE and adversary $\mathbb{A}$ is described in Fig. 3. It generates κ key pairs and provides the public keys to $\mathbb{A}$, who is then tasked with recovering exactly n of the corresponding private keys.

The adversary has access to both a decryption oracle $\mathcal{D}$ and a key corruption oracle $\mathcal{K}$, giving rise to chosen ciphertexts attacks with corruptions (CCA$\star$; the $\star$ denotes corruptions). The decryption oracle $\mathcal{D}(i, c)$ takes as input an index i and a ciphertext c, and returns the output of the decryption algorithm $\mathsf{PKE.Dec}$ on input sk_i and c. The corruption oracle $\mathcal{K}(i)$ simply takes as input a key index i, and returns the corresponding private key sk_i. The game notes that the key pair with index i has been corrupted by adding it to the global set $\mathtt{K}$.

Eventually, $\mathbb{A}$ outputs a set of key indices $\mathtt{I}$ and a list $(\hat{\mathsf{sk}}_i)_{i \in \mathtt{I}}$ of guesses of the private keys corresponding to those indices. In order for $\mathtt{I}$ to be eligible, it needs to have cardinality n without containing any corrupted key pairs, that is, the sets of guessed keys $\mathtt{I}$ and corrupted keys $\mathtt{K}$ should be disjoint. If $\mathtt{I}$ is eligible and every guessed private key matches the corresponding sampled one, the adversary wins the game. In that case, the game halts with output 1; otherwise, it halts with output 0. The advantage is the probability that the game outputs 1.

Definition 2. *Let PKE be a public-key encryption scheme. Then the universal key unrecoverability advantage of an adversary $\mathbb{A}$ is*

$$\mathsf{Adv}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = 1\right] ,$$

where the experiment is defined in Fig. 3.

Weaker notions emerge by dropping either or both of the two oracles. Without key corruption, standard CCA security results. Without decryption oracle, chosen plaintext security (CPA$\star$ resp. CPA) emerges. As usual, an encryption oracle is superfluous in the PKE setting.

For cryptosystems where a single public key may have many matching private keys (such as Cramer–Shoup [18]), universal key unrecoverability is rather weak. Hence, we consider a second, slightly stronger notion

```
Experiment $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-(u/m)ku-cca}\star}(\mathbb{A})$
$(\mathsf{pk}_1,\mathsf{sk}_1),\ldots,(\mathsf{pk}_\kappa,\mathsf{sk}_\kappa) \leftarrow\!\!\$\ \mathsf{PKE.Kg}$
$(\mathtt{I},(\hat{\mathsf{sk}}_i)_{i\in\mathtt{I}}) \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{D},\mathcal{K}}(\mathsf{pk}_1,\ldots,\mathsf{pk}_\kappa)$
if $|\mathtt{I}| \neq n \lor \mathtt{I} \cap \mathtt{K} \neq \emptyset$ then return $0$
UKU : return $\bigwedge_{i\in\mathtt{I}} \mathsf{sk}_i = \hat{\mathsf{sk}}_i$
MKU : return $\bigwedge_{i\in\mathtt{I}} \mathsf{PKE.Check}(\mathsf{pk}_i,\hat{\mathsf{sk}}_i,)$
```

```
Oracle $\mathcal{D}(i,c)$
$m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
return $m$
```

```
Oracle $\mathcal{K}(i)$
$\mathtt{K} \xleftarrow{\cup} i$
return $\mathsf{sk}_i$
```

Fig. 3. The key recovery experiments $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-uku-cca}\star}(\mathbb{A})$ and $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-mku-cca}\star}(\mathbb{A})$; they only differ in their win condition.

```
Experiment $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{A})$
$(\mathsf{pk}_1,\mathsf{sk}_1),\ldots,(\mathsf{pk}_\kappa,\mathsf{sk}_\kappa) \leftarrow\!\!\$\ \mathsf{PKE.Kg}$
$b_1,\ldots,b_\kappa \leftarrow\!\!\$\ \{0,1\}$
$(\mathtt{I},\hat{b}) \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{K},\mathcal{B}}(\mathsf{pk}_1,\ldots,\mathsf{pk}_\kappa)$
if $|\mathtt{I}| \neq n \lor \mathtt{I} \cap (\mathtt{K} \cup \mathtt{B}) \neq \emptyset$ then $\hat{b} \leftarrow\!\!\$\ \{0,1\}$
return $\oplus_{i\in\mathtt{I}}\, b_i = \hat{b}$
```

```
Oracle $\mathcal{K}(i)$
$\mathtt{K} \xleftarrow{\cup} i$
return $\mathsf{sk}_i$
```

```
Oracle $\mathcal{B}(i)$
$\mathtt{B} \xleftarrow{\cup} i$
return $b_i$
```

```
Oracle $\mathcal{E}(i,m_0,m_1)$
if $m_0 \not\sim m_1$ then return $\lightning$
$c \leftarrow\!\!\$\ \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m_{b_i})$
$\mathtt{M}_i(c) \leftarrow m_{b_i}$
$\mathtt{C}_i \xleftarrow{\cup} c$
return $c$
```

```
Oracle $\mathcal{D}(i,c)$
$m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
if $c \in \mathtt{C}_i \land m = \mathtt{M}_i(c)$ then return $\lightning$
return $m$
```

Fig. 4. Our main notion of multi-instance indistinguishability. In blue the slightly non-standard strengthening of the decryption oracle in case of imperfect correctness.

of key recovery, in which the recovered private keys are no longer required to be identical to those sampled in the game. Instead, it suffices that each passes the keypair checking algorithm $\mathsf{PKE.Check}$; here we leverage our correctness definition (Def. 1). We call the resulting notion *matching key unrecoverability* (MKU), whose game is included in Fig. 3. That MKU security indeed implies UKU security is captured by Thm. 1 below, where the error term $\kappa\gamma$ results from the unique correct keys as output by the key generation not always passing the $\mathsf{PKE.Check}$ algorithm (see App. A.1 for the proof).

Theorem 1 **(MKU $\longrightarrow$ UKU).** *Let $0 < n \leq \kappa$ be integer parameters and let PKE be a (γ,δ)-correct encryption scheme. Then, there is a type-preserving SFBB reduction $\mathbb{B}_{\mathrm{mku}}$, such that for every adversary $\mathbb{A}_{\mathrm{uku}}$,*

$$\mathsf{Adv}_{\mathrm{PKE}}^{(n,\kappa)\text{-uku-cca}\star}(\mathbb{A}_{\mathrm{uku}}) \leq \mathsf{Adv}_{\mathrm{PKE}}^{(n,\kappa)\text{-mku-cca}\star}(\mathbb{B}_{\mathrm{mku}}) + \kappa\gamma\,.$$

3.2 Left-or-Right XOR Indistinguishability

To capture a stronger notion of security than simply hardness of key recovery, BRT considered various generalizations of indistinguishability to the multi-instance setting. For perfect probabilistic symmetric encryption, they concluded that left-or-right xor-indistinguishability is the strongest notion. Here each key comes with its own challenge bit that determines the left-or-right nature of the corresponding challenge encryption oracle; the adversary is tasked to retrieve the xor of all the challenge bits. In Def. 3, we use our modified game mechanics to adapt left-or-right xor-indistinguishability for potentially non-perfect public-key encryption.

Definition 3. *Let PKE be a public-key encryption scheme. Then the xor-indistinguishability advantage of an adversary $\mathbb{A}$ is*

$$\mathsf{Adv}_{\mathrm{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{A}) = 2 \cdot \Pr\Big[\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{A}) = 1\Big] - 1\,,$$

where the experiment is defined in Fig. 4.

In the experiment $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{A})$, the adversary gets access to κ independently drawn public keys and helper oracles $\mathcal{D}$ and $\mathcal{K}$ (as described in Sect. 3.1). Furthermore, $\mathbb{A}$ gets access to a challenge encryption oracle $\mathcal{E}$ and a separate bit corruption oracle $\mathcal{B}$.

On input two equivalent messages m_0 and m_1 and a public key index i, the challenge encryption oracle returns $\mathsf{PKE.Enc}_{\mathsf{pk}_i}(m_{b_i})$ where b_i is the challenge bit associated with the public key indexed by i. As usual for IND-CCA notions, challenge ciphertexts cannot be queried to the decryption oracle, which we catch on-the-fly [9]. Owing to the imperfect decryption, we allow a slight relaxation: if a challenge ciphertext decrypts incorrectly, we do not suppress the output and essentially allow the query. This relaxation strengthens the notion, but as challenge ciphertexts are honestly generated, the advantage gained by an adversary can be bound by the correctness parameters of the PKE using an identical-until-bad argument; however such a generic approach might not give bounds appropriate for the multi-instance setting.

Eventually, the adversary returns a set $\mathtt{I}$ of targets and a guess $\hat{b}$ of the xor of the corresponding challenge bits b_i. If $\mathtt{I}$ is a set of n uncorrupted indices, then intuitively an adversary's uncertainty about any of the n challenge bits will be affected in the final guess $\hat{b}$, so in that sense $\hat{b}$ neatly captures an adversary's need to break n instances in order to win. If $\mathtt{I}$ is not a set of n uncorrupted indices, it is considered ineligible, at which point the game overwrites $\mathbb{A}$'s guess $\hat{b}$ with a uniform guess. This mechanism is equivalent to the experiment returning true or false with equal probability, thus ensuring an adversary gains zero advantage from such a bad $\mathtt{I}$.

The Relationship with Key Recovery. BRT showed that in their perfect symmetric setting, multi-instance indistinguishability implies multi-instance universal key unrecoverability. While that may sound like a triviality, their proof [13, App. C] was not entirely straightforward and, to ensure that the advantages carried over neatly, the distinguishing reduction receiving recovered keys needed to amplify its success probability by repeated random challenge encryptions. Their bound ends up with an additive term that corresponds to the likelihood that decrypting using an incorrect key results in the opposite message from the decrypted one.

Our imperfect public key setting is slightly different. On the one hand, the reduction can check the recovered keys with the $\mathsf{PKE.Check}$ algorithm, yet on the other hand correct keys can still cause incorrect decryptions. As a result, our amplification based on multiple challenge encryptions differs from BRT's, as we move from unanimity to a plurality vote. Furthermore, our reduction can use fixed messages (to match how correctness is defined), which reduces a dependency (in the bound) on the size of the message space. We suspect that our amplification can be tightened further by a combination of exploiting randomness and more fine-tuned voting, coupled with more fine-grained bounding of probabilities.

As is, the complexity of the bound makes its behaviour somewhat opaque and for some parameter choices vacuous (when $c < 0$). The main idea is that $\mathbb{B}_{\text{ind}}$ can increase q, the number of challenge encryptions per user, to counteract the losses inferred by large n and/or large δ, with a small penalty to its running time. For $\delta = 2^{-64}$, $q = 1$ already suffices for $c > 1/2$ for $n < 2^{25}$. In case of perfect correctness for keys that check out, corresponding to $\delta = 0$, the bound is completely tight.

Theorem 2 **(IND $\longrightarrow$ MKU).** *Let PKE be a (γ, δ)-correct encryption scheme with $\delta < 1/2$. Then there is a type-preserving SFBB reduction $\mathbb{B}_{\text{ind}}$ such that, for every $\mathbb{A}_{\text{mku}}$,*

$$\mathsf{Adv}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{PKE}}(\mathbb{B}_{\text{ind}}) \geq c \cdot \mathsf{Adv}^{(n,\kappa)\text{-mku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}_{\text{mku}}) \,,$$

with $c = 2\left(1 - 2^q(\delta(1-\delta))^{\frac{q}{2}}\right)^n - 1$ where $q \in \mathbb{Z}_{>0}$ is an amplification parameter of the reduction; $\mathbb{B}_{\text{ind}}$'s overhead consists of $q \cdot n$ calls to $\mathcal{E}$, n offline key checks, and $q \cdot n$ offline decryptions.

Proof. Let $\mathbb{B}_{\text{ind}}$ run adversary $\mathbb{A}_{\text{mku}}$ on the same κ public keys as it received itself. Whenever $\mathbb{A}_{\text{mku}}$ makes a decryption or corruption query, $\mathbb{B}_{\text{ind}}$ simply forwards the queries to its own oracle, relaying the response back to $\mathbb{A}_{\text{mku}}$. Eventually, $\mathbb{A}_{\text{mku}}$ terminates with output $(\mathtt{I}, (\hat{\mathsf{sk}}_i)_{i \in \mathtt{I}})$ and $\mathbb{B}_{\text{ind}}$ first confirms whether $\mathbb{A}_{\text{mku}}$ won, by checking, for all the returned private keys, whether $\mathsf{PKE.Check}(\mathsf{pk}_i, \hat{\mathsf{sk}}_i)$ holds. If any check fails, $\mathbb{B}_{\text{ind}}$ halts with output $(\emptyset, 0)$, prompting the eligibility check of $\mathsf{Exp}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{PKE}}(\mathbb{A})$ to fail.

Let m_0 and m_1 be two distinct yet equivalent messages. Then for all $i \in \mathtt{I}$, $\mathbb{B}_{\text{ind}}$ creates a guess $\hat{b}_i$ by querying its challenge encryption oracle q times on those two messages, so q queries $\mathcal{E}(i, m_0, m_1)$ resulting in c_{ij}, for $j \in [q]$. It then decrypts those ciphertexts using the private key $\hat{\mathsf{sk}}_i$ it obtained from $\mathbb{A}_{\text{mku}}$, resulting in purported messages $m_{ij} \leftarrow \mathsf{PKE.Dec}_{\hat{\mathsf{sk}}_i}(c_{ij})$. If, for a fixed i, there are strictly more than $q/2$ appearances of m_0 amongst the m_{ij}, it sets $\hat{b}_i$ to 0; if there are strictly more than $q/2$ appearances of m_1, then it sets $\hat{b}_i$ to 1. If neither message appears more than $q/2$ times, $\mathbb{B}_{\text{ind}}$ halts with output $(\emptyset, 0)$. Once $\mathbb{B}_{\text{ind}}$ has created a guess $\hat{b}_i$ for all $i \in \mathtt{I}$, it terminates on output $(\mathtt{I}, \bigoplus_{i \in \mathtt{I}} \hat{b}_i)$.

For $i \in \mathtt{I}$, let Check_i be the event that $\mathbb{A}_{\text{mku}}$ outputs a key $\hat{\mathsf{sk}}_i$ that passes the test and let Good_i be the event that $\mathbb{B}_{\text{ind}}$'s guess $\hat{b}_i$ actually equals b_i. Let $\mathsf{Check}_{\mathtt{I}}$ be the event that all Check_i hold (for $i \in \mathtt{I}$) and define $\mathsf{Good}_{\mathtt{I}}$ analogously.

As $\mathbb{B}_{\text{ind}}$'s simulation of $\mathsf{Exp}_{\text{PKE}}^{(n,\kappa)\text{-mku-cca}\star}$ is perfect, we know that

$$\mathsf{Adv}^{(n,\kappa)\text{-mku-cca}\star}(\mathbb{A}_{\text{mku}}) = \Pr[\mathsf{Check}_{\mathtt{I}}]\,,$$

moreover,

$$\begin{aligned}\Pr\Big[\mathsf{Exp}_{\text{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{B}_{\text{ind}}) = 1\Big] &\geq \Pr[\mathsf{Check}_{\mathtt{I}} \wedge \mathsf{Good}_{\mathtt{I}}] + \Pr[\neg\mathsf{Check}_{\mathtt{I}} \wedge b = 0] \\ &= \Pr[\mathsf{Good}_{\mathtt{I}} \mid \mathsf{Check}_{\mathtt{I}}]\Pr[\mathsf{Check}_{\mathtt{I}}] + \frac{1}{2}\left(1 - \Pr[\mathsf{Check}_{\mathtt{I}}]\right)\end{aligned}$$

which implies that

$$\mathsf{Adv}_{\text{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{B}_{\text{ind}}) \geq \left(2\Pr[\mathsf{Good}_{\mathtt{I}} \mid \mathsf{Check}_{\mathtt{I}}] - 1\right)\mathsf{Adv}^{(n,\kappa)\text{-mku-cca}\star}(\mathbb{A}_{\text{mku}})\,.$$

To bound $\Pr[\mathsf{Good}_{\mathtt{I}} \mid \mathsf{Check}_{\mathtt{I}}]$ we exploit the correctness definition, specifically that its quantification (Def. 1) ensures that whenever Check_i holds, we have that $\Pr\Big[\mathsf{PKE.Dec}_{\hat{\mathsf{sk}}_i}(\mathsf{PKE.Enc}_{\mathsf{pk}_i}(m)) = m\Big] \geq 1-\delta$, irrespective of m and where the probability is only over the randomness of $\mathsf{PKE.Enc}$.

If, for a given i, decryption is correct strictly more than $q/2$ times, then we are guaranteed that Good_i occurs. If we let $B\big(q,p\big)$ be the binomial distribution over q trials and with probability p, then

$$\Pr[\mathsf{Good}_i \mid \mathsf{Check}_i] \geq \Pr\Big[B\big(q,(1-\delta)\big) > \frac{q}{2}\Big]$$

and, as this bound only relies on the randomness of the challenge encryption oracle, guaranteed independent for differing i, we may conclude that

$$\Pr[\mathsf{Good}_{\mathtt{I}} \mid \mathsf{Check}_{\mathtt{I}}] \geq \left(\Pr\Big[B\big(q,(1-\delta)\big) > \frac{q}{2}\Big]\right)^n\,.$$

Finally, we note that

$$\Pr\Big[B\big(q,(1-\delta)\big) > \frac{q}{2}\Big] \geq 1 - 2^q\left(\delta(1-\delta)\right)^{\frac{q}{2}}$$

by a standard application of known bounds on binomial tails, requiring $\delta \leq 1/2$ (see details below). Plugging in all the various bounds recovers the theorem statement.

For the binomial tail bound, we use the Chernoff–Hoeffding bound [27], which states that, for a binomial distribution $B\big(q,p\big)$ over q trials and with probability p, and any k satisfying $p < \frac{k}{q} < 1$ the tail bound

$$\Pr\big[B\big(q,p\big) \geq k\big] \leq \exp\left[-qD\left(\frac{k}{q}\middle\|\, p\right)\right]$$

holds, where $D(a\|b)$ is the Kullback–Leibler divergence defined as $D(a\|b) = a\ln\left(\frac{a}{b}\right) + (1-a)\ln\left(\frac{1-a}{1-b}\right)$.

We further use the trick that $\Pr\big[B\big(q,(1-\delta)\big) > \frac{q}{2}\big] = 1 - \Pr\big[B\big(q,\delta\big) \leq \frac{q}{2}\big]$, so the relevant Kullback–Leibler divergence becomes

$$\begin{aligned}D\left(\frac{1}{2}\middle\|\,\delta\right) &= \frac{1}{2}\ln\left(\frac{\frac{1}{2}}{\delta}\right) + \left(1-\frac{1}{2}\right)\ln\left(\frac{\left(1-\frac{1}{2}\right)}{1-\delta}\right) \\ &= \frac{1}{2}\ln\left(\frac{1}{2\delta}\right) + \frac{1}{2}\ln\left(\frac{1}{2(1-\delta)}\right) \\ &= \ln\left[\left(\frac{1}{4\delta(1-\delta)}\right)^{\frac{1}{2}}\right]\,,\end{aligned}$$

which allows us to compute the bound

$$\begin{aligned}\Pr\Big[B\big(q,(1-\delta)\big) > \frac{q}{2}\Big] &\geq 1 - \exp\left[-qD\left(\frac{1}{2}\middle\|\,\delta\right)\right] \\ &= 1 - \exp\left[-q\ln\left[\left(\frac{1}{4\delta(1-\delta)}\right)^{\frac{1}{2}}\right]\right] \\ &= 1 - 2^q\left(\delta(1-\delta)\right)^{\frac{q}{2}}\,.\end{aligned}$$

□

```
Experiment $\mathsf{Exp}_{\mathrm{PKE}}^{(\leq\kappa,\kappa)\text{-ind-cca}\star}(\mathbb{A})$
4 : if $|\mathtt{I}| \neq \kappa$ then $\hat{b} \leftarrow 0$

Experiment $\mathsf{Exp}_{\mathrm{PKE}}^{(\geq n,\kappa)\text{-ind-cca}\star}(\mathbb{A})$
4 : if $|\mathtt{I}| < n \vee \mathtt{I} \cap (\mathtt{K} \cup \mathtt{B}) \neq \emptyset$ then $\hat{b} \leftarrow 0$
```

Fig. 5. The main differences between our mechanism for multi-instance indistinguishability (Fig. 4) and prior art revolve around line 4: BRT's experiment $\mathsf{Exp}_{\mathrm{PKE}}^{(\leq\kappa,\kappa)\text{-ind-cca}\star}(\mathbb{A})$ (left) and AGK's experiment $\mathsf{Exp}_{\mathrm{PKE}}^{(\geq n,\kappa)\text{-ind-cca}\star}(\mathbb{A})$ (right). The differences are highlighted in blue.

Corollary 1 (IND $\longrightarrow$ UKU). *Let PKE be a (γ,δ)-correct encryption scheme with $\delta < 1/2$. Then there is a type-preserving SFBB reduction $\mathbb{B}_{\mathrm{ind}}$ such that, for every $\mathbb{A}_{\mathrm{uku}}$,*

$$\mathsf{Adv}_{\mathrm{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{B}_{\mathrm{ind}}) \geq c \cdot \mathsf{Adv}_{\mathrm{PKE}}^{(n,\kappa)\text{-uku-cca}\star}(\mathbb{A}_{\mathrm{uku}}) - \kappa\gamma\,,$$

with c, q, and $\mathbb{B}_{\mathrm{ind}}$'s overhead as above (Thm. 2).

3.3 Alternative Mechanisms

As we mentioned before, our mechanism to capture multi-instance security differs slightly from those used by BRT and AGK, respectively, even when accounting for changes in primitive and correctness. At first sight, the differences might appear mostly cosmetic, though there are some subtleties involved.

The BRT Notion: Requiring $n = \kappa$, Possibly Corrupted, Targets. BRT require an adversary to return the xor of all bits, but allow those bits or corresponding users to be corrupted. Fig. 5 reflects the small change needed in the code of our security experiment to match BRT's mechanism (ignoring a minor, inconsequential difference, as BRT have a single, merged corruption oracle that returns both key and bit). As motivation for including corruptions, BRT discuss the scenario that, say, half of the keys generated are hopelessly insecure: an adversary breaks the insecure half and corrupts the rest, thus being successful. Moreover, they mention that their choice implies security under a corruptionless notion with dynamically chosen $\mathtt{I}$.

Although the implication is of course true, and something can be said to target the strongest possible notion, corruptions have a habit of creating complications for reductions and provable security in general. Yet, we believe the inclusion of corruptions, or not, should reflect the threat model of the adversary and that choice should be orthogonal to the number of users being targeted. BRT, instead of having an explicit hardness parameter n, restrict an adversary to make at most q_c corruption queries to avoid trivial wins when $q_c = \kappa$. Yet, whether the resulting, intuitive hardness will or should then match $n = \kappa - q_c$, is unclear.

We address the equivalence between BRT's mechanism and our general mechanism (with corruptions) in Lemmas 1 and 2. Both lemmas have in common that the respective reductions may make up to $\kappa - n$ additional bit corruptions. In other words, the reductions are not type-preserving, making the equivalence somewhat sloppy. As an aside, using techniques similar to those to prove Thm. 2, the key corruption oracle could be used (at a loss) to simulate the bit corruption oracle instead.

Lemma 1 (main notion $\Longrightarrow$ BRT). *Let $n \leq \kappa$ and $q_c \leq \kappa - n$. Then there is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$ making at most q_c corruption oracle calls,*

$$\mathsf{Adv}_{\mathrm{PKE}}^{(\leq\kappa,\kappa)\text{-ind-cca}\star}(\mathbb{A}) \leq \mathsf{Adv}_{\mathrm{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{B})\,,$$

where $\mathbb{B}$ makes at most $\kappa - n$ additional bit corruption oracle calls.

Proof. As both games provide the same interface to $\mathbb{A}$, $\mathbb{B}$ can simply simulate the game for $\mathbb{A}$ by forwarding every oracle call, where $\mathbb{B}$ stores the input and output for later use. Upon halting, $\mathbb{A}$ returns a guess $\hat{b}_{\mathbb{A}}$ for the value of $\oplus_{i\in[\kappa]} b_i$.

As $\mathbb{A}$ makes at most $\kappa - n$ corruption queries, there are at least n uncorrupted keys left for $\mathbb{B}$, who may select an arbitrary uncorrupted subset $\mathtt{I}$ of cardinality n. Subsequently $\mathbb{B}$ bit-corrupts all remaining instances, possibly forgoing those that $\mathbb{A}$ already bit-corrupted. Eventually, $\mathbb{B}$ sets $\hat{b}_{\mathbb{B}} \leftarrow \hat{b}_{\mathbb{A}} \oplus \bigoplus_{i\in\mathtt{B}} b_i$, where $\mathtt{B}$ is the set of corrupted bits $\mathtt{B} = [\kappa]\backslash\mathtt{I}$, and halts with output $(\mathtt{I}, \hat{b}_{\mathbb{B}})$.

As the oracles behave exactly the same, the simulation is perfect and, by inspection, $\mathbb{B}$ wins iff $\mathbb{A}$ wins. □

Lemma 2 (BRT $\Longrightarrow$ main notion). *Let $n \leq \kappa$. Then there is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\mathsf{Adv}_{\mathrm{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{A}) \leq \mathsf{Adv}_{\mathrm{PKE}}^{(\leq\kappa,\kappa)\text{-ind-cca}\star}(\mathbb{B})\,,$$

where $\mathbb{B}$ makes at most $\kappa - n$ additional bit corruption oracle calls.

Proof. As both games provide the same interface to the adversary, $\mathbb{B}$ can simply simulate the game for $\mathbb{A}$ by forwarding every oracle call, where $\mathbb{B}$ stores the input and output for later use. Upon halting, $\mathbb{A}$ returns a list $\mathtt{I}$ and a guess $\hat{b}_{\mathbb{A}}$ for the value of $\oplus_{i \in \mathtt{I}} b_i$. Let $\mathtt{B} = [\kappa] \backslash \mathtt{I}$, then $\mathbb{B}$ bit-corrupts all of $\mathtt{B}$, so it can set $\hat{b}_{\mathbb{B}} \leftarrow \hat{b}_{\mathbb{A}} \oplus \bigoplus_{i \in \mathtt{B}} b_i$ and halt with output $\hat{b}_{\mathbb{B}}$. As the oracles behave exactly the same, the simulation is perfect and, by inspection, $\mathbb{B}$ wins iff $\mathbb{A}$ wins. □

The AGK Notion: Allowing More than n Targets without Corruptions. When AGK studied KEMs in the multi-instance setting, they used a xor notion with the n as the *minimum* number of targets to attack (out of κ possible) as an explicit parameter; moreover, an adversary would not have access to any corruption oracles. Fig. 5 reflects the small change needed in the code of our security experiment to match AGK's mechanism with corruptions added (where we fixed a minor bug in their code: rather than resampling $\hat{b}$, their experiment would immediately return 0 instead, inadvertently granting an adversary that deliberately returns a compromised handle the significant advantage of -1).

Absent corruptions, AGK indicated that for some pathological schemes, breaking more targets might paradoxically be easier than breaking fewer [3, App. C]. In those cases, the freedom to return a set $\mathtt{I}$ of cardinality greater than n would make life easier for an adversary, leading to a stronger notion.

In the presence of corruptions, requiring the adversary to target exactly n users as we do is without loss of generality. As an example, if an adversary can figure out the xor of $n+1$ honest bits, it can bit-corrupt any single one of these $n+1$, and xor the resulting bit out of the initial guess to obtain a final one on n bits instead. We formalize this intuition below.

Lemma 3 (main notion $\implies$ AGK*). *There is an SFBB adversary $\mathbb{B}$ such that, for every $\mathbb{A}$,*

$$\mathsf{Adv}_{\mathrm{PKE}}^{(\geq n,\kappa)\text{-ind-cca}\star}(\mathbb{A}) \leq \mathsf{Adv}_{\mathrm{PKE}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{B}) .$$

If $\mathbb{A}$ returns a list of n' targets, $\mathbb{B}$ makes $n'-n$ additional calls to its bit corruption oracle.

3.4 Real-or-Random XOR Indistinguishability

An alternative notion of indistinguishability, known as real-or-random indistinguishability (ROR), sees the adversary tasked with figuring out whether a challenge ciphertext contains the adversarially chosen message m or an unknown, randomly chosen message. The game $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-ror-cca}\star}$ is exactly as in Fig. 4, apart from the challenge encryption oracle $\mathcal{E}_{\mathrm{ROR}}(i,m)$, which sets $m_0 \leftarrow m$ and $m_1 \leftarrow\!\!\$ \; [m]$ to then call (left-or-right) $\mathcal{E}(i,m_0,m_1)$.

By construction, left-or-right indistinguishability easily implies real-or-random indistinguishability. That statement is as true in the multi-instance setting as it is in the classical single-user setting. Conversely, in the single-user setting, it has long been established that the reduction from ROR to IND loses a factor 2 [8]. However, BRT showed that in the multi-instance setting, the factor 2 blows up exponentially to, in their case, 2^κ. Yet, BRT argue that this exponential loss is not as bad as it might seem, given that the multi-instance advantages are supposed to be exponentially smaller than their single-user counterparts. Thus, reductions incurring losses exponential in κ or n can still be valuable.

To adapt BRT's reduction to our setting, we require $n=\kappa$, implying that $\mathbb{A}$ cannot access its corruption oracles. Otherwise, corruptions would make the reduction noticeable once at least one b_i is set to 1, potentially influencing an adversary's behaviour in unpredictable ways.

Theorem 3. *There is an SFBB reduction $\mathbb{B}$ such that, for every adversary $\mathbb{A}$,*

$$\mathsf{Adv}_{\mathrm{PKE}}^{(\kappa,\kappa)\text{-ind-cca}}(\mathbb{A}) \leq 2^\kappa \cdot \mathsf{Adv}_{\mathrm{PKE}}^{(\kappa,\kappa)\text{-ror-cca}}(\mathbb{B}) ,$$

where $\mathbb{B}$ additionally draws κ bits uniformly at random.

Proof. First, $\mathbb{B}$ uniformly draws κ independent bits d_i. Let $d = \oplus_{i\in[\kappa]} d_i$ denote their xor. Then, whenever $\mathbb{A}$ calls $\mathcal{E}(i,m_0,m_1)$, $\mathbb{B}$ calls $\mathcal{E}_{\mathrm{ROR}}(i,m_{d_i})$. Calls to $\mathcal{D}$ are simply forwarded. Once $\mathbb{A}$ halts with output $\hat{d}$, $\mathbb{B}$ will guess "real" iff $\mathbb{A}$ guessed d correctly, that is, $\mathbb{B}$ sets $\hat{b} = \hat{d} \oplus d$ and halts with the output $\hat{b}$.

If $b_i = 0$ for all $i \in [\kappa]$, then the simulation is perfect, and $\mathbb{B}$ wins whenever $\mathbb{A}$ wins. If, on the other hand, $b_i = 1$ for some i, then the corresponding d_i is information-theoretically hidden from $\mathbb{A}$ and consequently, d itself will be perfectly hidden. Hence, $\hat{d} = d$ will occur with probability 1/2, irrespective of $\mathbb{A}$'s behaviour. Using these observations, we obtain:

$$\Pr\Big[\mathsf{Exp}_{\mathrm{PKE}}^{(\kappa,\kappa)\text{-ror-cca}}(\mathbb{B}) = 1\Big] = \Pr\big[\forall_{i\in[\kappa]} b_i = 0\big] \cdot \Pr\Big[\hat{d} = d \;\Big|\; \forall_{i\in[\kappa]} b_i = 0\Big]$$

$$
\begin{aligned}
&+ \Pr\Big[\hat{d} = d \;\Big|\; \exists_{i\in[\kappa]} b_i = 1\Big] \cdot \big(1 - \Pr\big[\forall_{i\in[\kappa]} b_i = 0\big]\big) \\
&= \frac{1}{2^\kappa} \Pr\Big[\mathsf{Exp}^{(\kappa,\kappa)\text{-ind-cca}}_{\mathrm{PKE}}(\mathbb{A}) = 1\Big] + \frac{1}{2}\left(1 - \frac{1}{2^\kappa}\right) \\
&= \frac{1}{2^\kappa}\left(\Pr\Big[\mathsf{Exp}^{(\kappa,\kappa)\text{-ind-cca}}_{\mathrm{PKE}}(\mathbb{A}) = 1\Big] - \frac{1}{2}\right) + \frac{1}{2} \\
\implies \mathsf{Adv}^{(\kappa,\kappa)\text{-ror-cca}}_{\mathrm{PKE}}(\mathbb{B}) &= \frac{1}{2^\kappa}\left(2 \cdot \Pr\Big[\mathsf{Exp}^{(\kappa,\kappa)\text{-ind-cca}}_{\mathrm{PKE}}(\mathbb{A}) = 1\Big] - 1\right) \\
&= \frac{1}{2^\kappa} \mathsf{Adv}^{(\kappa,\kappa)\text{-ind-cca}}_{\mathrm{PKE}}(\mathbb{A})\,.
\end{aligned}
$$

□

Furthermore, a reduction playing an (n, n) game can exploit an adversary playing a (n, κ) game by guessing in advance the set $\mathtt{I}$ of targets that the adversary will return. A correct, eligible guess allows the reduction to simulate the remaining keys without being noticed.

Theorem 4. *There is an SFBB reduction* $\mathbb{B}$ *such that, for every adversary* $\mathbb{A}$,

$$\mathsf{Adv}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) \le \binom{\kappa}{n} \cdot \mathsf{Adv}^{(n,n)\text{-ind-cca}}_{\mathrm{PKE}}(\mathbb{B})\,.$$

$\mathbb{B}$*'s overhead consists of generating* $\kappa - n$ *fresh keypairs, sampling* $\kappa - n$ *bits, and choosing a subset of* $[\kappa]$ *of cardinality* n *uniformly at random.*

Composing Thm. 3 and 4, we obtain the following bound.

Corollary 2 ($\mathrm{ROR} \implies \mathrm{IND}$**).** *There is an SFBB reduction* $\mathbb{B}$ *such that, for any adversary* $\mathbb{B}$,

$$\mathsf{Adv}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) \le \binom{\kappa}{n} \cdot 2^n \cdot \mathsf{Adv}^{(n,n)\text{-ror-cca}}_{\mathrm{PKE}}(\mathbb{B})\,.$$

$\mathbb{B}$*'s overhead consists of generating* $\kappa - n$ *fresh keypairs, sampling* κ *bits, and choosing a subset of* $[\kappa]$ *of cardinality* n *uniformly at random.*

An alternative bound losing a factor 2^κ is possible by combining Thm. 3 with Lemma 2, however a simple analysis shows that whenever $n < \kappa/5$ the corollary above is preferable.

At first glance, an exponential-looking loss of 2^κ might seem severe, potentially rendering the resulting bound vacuous. Yet, as BRT already highlighted, the multi-instance advantages themselves might vanish exponentially in n, making the bounds relevant for the notions being compared. Nonetheless, tigher bounds still matter; unfortunately achieving even tighter bounds in the general case seems challenging [5, 12].

4 Inheriting Multi-Instance Security

4.1 TagKEM: Definition and Notion of Security

Our goal is to turn the AGK multi-instance secure KEM into a PKE. Yet, for the construction of hybrid encryption, the more general TagKEMs, where encapsulation is split into two algorithms ($\mathsf{TKEM.Key}$ and $\mathsf{TKEM.Enc}$) have proven more powerful [2]: intuitively speaking, splitting the algorithm allows the tag and consequently the key encapsulation to depend on the data encapsulation, making CCA security of the hybrid construction easier to achieve (cf. the Kurosawa–Desmedt scheme [31]). In Def. 4 we introduce a further generalization, called TagXEM, by allowing extendable output lengths for the ephemeral keys produced by the TagXEM.

Definition 4 (TagXEM). *A TagXEM is a tuple of algorithms* ($\mathsf{TXEM.Kg}$, $\mathsf{TXEM.Key}$, $\mathsf{TXEM.Enc}$, $\mathsf{TXEM.Dec}$, $\mathsf{TXEM.Check}$)*, where long-term key generation* $\mathsf{TXEM.Kg}$ *on input* pm *outputs a random keypair* $(\mathsf{pk}, \mathsf{sk})$*; ephemeral key generation* $\mathsf{TXEM.Key}$ *on input* pk *and* $\ell \in \mathbb{Z}_{>0}$*, outputs a random ephemeral key* $K \in \{0,1\}^\ell$ *and an internal state* σ*, subsequently encapsulation* $\mathsf{TXEM.Enc}$ *on input a state* σ *and a tag* $\tau \in \mathcal{T}$*, deterministically outputs an encapsulation* c*, or a special symbol* $\perp$ *denoting failure. The deterministic decapsulation algorithm* $\mathsf{TXEM.Dec}$ *takes input a private key* sk*, an encapsulation* c*, a tag* τ*, and a length* ℓ*, and outputs either a key* $K \in \{0,1\}^\ell$ *or* $\perp$ *to denote failure. Finally, the deterministic* $\mathsf{TXEM.Check}$ *takes as input the system parameters* pm *as well as a purported public/private key pair* $(\mathsf{pk}, \mathsf{sk})$ *and returns* true *or* false.

```
Experiment $\mathsf{Exp}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{TXEM}}(\mathbb{A})$
  $(\mathsf{pk}_1, \mathsf{sk}_1), \ldots, (\mathsf{pk}_\kappa, \mathsf{sk}_\kappa) \leftarrow\$ \mathsf{TXEM.Kg}$
  $b_1, \ldots, b_\kappa \leftarrow\$ \{0,1\}$
  $(\mathtt{I}, \hat{b}) \leftarrow\$ \mathbb{A}^{\mathcal{C},\mathcal{E},\mathcal{D},\mathcal{K},\mathcal{B}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
  if $|\mathtt{I}| \neq n \vee \mathtt{I} \cap (\mathtt{K} \cup \mathtt{B}) \neq \emptyset$ then $\hat{b} \leftarrow\$ \{0,1\}$
  return $\bigoplus_{i \in \mathtt{I}} b_i = \hat{b}$

Oracle $\mathcal{D}(i, \langle c, \tau \rangle, \ell)$
  $K \leftarrow \mathsf{TXEM.Dec}_{\mathsf{sk}_i}(c, \tau, \ell)$
  if $\mathtt{P}_i(c, \tau) \neq \emptyset$
    $K' \leftarrow \mathtt{P}_i(c, \tau), \ell' \leftarrow \min\{\ell, |K'|\}$
  else
    $K' \leftarrow \varepsilon, \ell' \leftarrow 0$
  if $\langle c, \tau \rangle \in \mathtt{C}_i \wedge K[\![\ell']\!] = K'[\![\ell']\!]$
    return $\lightning$
  return $K$
```

```
Oracle $\mathcal{C}(i, \ell)$
  $(K_0, \sigma) \leftarrow\$ \mathsf{TXEM.Key}_{\mathsf{pk}_i}(\ell)$
  $\mathtt{E}_i \xleftarrow{\frown} \langle \sigma, K_0 \rangle$
  $K_1 \leftarrow\$ \{0,1\}^\ell$
  return $K_{b_i}$

Oracle $\mathcal{E}(i, j, \tau)$
  if $\mathtt{E}_i[j] = \emptyset$ then return $\lightning$
  $\langle \sigma, K \rangle \leftarrow \mathtt{E}_i[j], \mathtt{E}_i[j] \leftarrow \emptyset$
  $c \leftarrow\$ \mathsf{TXEM.Enc}(\sigma, \tau)$
  $\mathtt{P}_i(c, \tau) \leftarrow K$
  $\mathtt{C}_i \xleftarrow{\cup} \langle c, \tau \rangle$
  return $c$

Oracle $\mathcal{K}(i)$
  $\mathtt{K} \xleftarrow{\cup} i$
  return $\mathsf{sk}_i$

Oracle $\mathcal{B}(i)$
  $\mathtt{B} \xleftarrow{\cup} i$
  return $b_i$
```

Fig. 6. Multi-instance indistinguishability notion for TXEM. In blue the same strengthening as in Fig. 4 in the case of imperfect correctness, with a slightly more complex admin to accomodate tags and length extension. We take $K[\![\ell]\!]$ to mean the first ℓ bits of K and ε as the empty string.

If we restrict to a single value ℓ, the usual notion of TagKEMs appears; moreover if we restrict to a single value of τ, the $\mathsf{TXEM.Key}$ and $\mathsf{TXEM.Enc}$ algorithms can be merged into a single key encapsulation mechanism, leading to normal KEMs (or XEMs if the variable output length is still incorporated). Consequently, the correctness and security definitions for the more general TagXEMs, as discussed throughout this section, imply corresponding definitions for KEM, XEM, and TagKEM.

For correctness, we allow the effective tag space $\mathcal{T}_\ell$ to depend on the length ℓ of the ephemeral key. Similarly to Def. 1, we define (γ, δ)-correctness for TagXEM. To ensure correctness for all τ, including those that depend on K, τ's quantifier sits inside the probability statement.

Definition 5 ((γ, δ)-Correctness TagXEM). *Let $\gamma, \delta \in [0,1]$. Then a tag extendable-output key encapsulation mechanism* TXEM *is called (γ, δ)-correct iff*

1. $\Pr[(\mathsf{pk}, \mathsf{sk}) \leftarrow\$ \mathsf{TXEM.Kg}(\mathsf{pm}) \; : \; \mathsf{TXEM.Check}(\mathsf{pm}, \mathsf{pk}, \mathsf{sk}) = \mathsf{false}] \leq \gamma$*;*
2. *if* $\mathsf{TXEM.Check}(\mathsf{pm}, \mathsf{pk}, \mathsf{sk}) = \mathsf{true}$ *then for all $\ell \in \mathbb{Z}_{>0}$ it holds that*

$$\Pr\left[(K, \sigma) \leftarrow\$ \mathsf{TXEM.Key}_{\mathsf{pk}}(\ell) \; : \; \exists \tau \in \mathcal{T}_\ell \text{ } \mathbf{s.th.} \begin{array}{l} c \leftarrow \mathsf{TXEM.Enc}(\sigma, \tau) \\ \mathsf{TXEM.Dec}_{\mathsf{sk}}(c, \tau, \ell) \neq K \end{array}\right] \leq \delta \, .$$

For security, Abe et al.'s notion of TagKEM indistinguishability [2] transfers easily to the multi-instance setting. The relevant game is given in Fig. 6, where we also made the necessary changes to deal with the variable output length of TagXEMs, plus the strengthening of $\mathcal{D}$ in the case of imperfect correctness (cf. Sect. 3.2).

Definition 6. *Let* TXEM *be a TagXEM. Then the xor-indistinguishability advantage of an adversary $\mathbb{A}$ is*

$$\mathsf{Adv}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{TXEM}}(\mathbb{A}) = 2 \cdot \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{TXEM}}(\mathbb{A}) = 1\right] - 1 \, ,$$

where the experiment is defined in Fig. 6.

If we fix ℓ and set $\mathcal{T}_\ell$ to a single element, the notion captures multi-instance security for standard KEMs, which is near equivalent (see Sect. 3.3) the notion that AGK used. In other words, provided MI-gapCDH is hard, their construction achieves (n, κ)-IND-CCA security in the random oracle model, but only for fixed ℓ and trivial $\mathcal{T}_\ell$ [4, Thm. 2].

```
$\mathsf{TXEM.Key}_{\mathsf{pk}}(\ell)$
$(K^{\mathrm{kem}}, \sigma) \leftarrow\!\!\$\ \mathsf{TKEM.Key}_{\mathsf{pk}}$
$K^{\mathrm{xem}} \leftarrow F(K^{\mathrm{kem}}, \ell)$
$\sigma' \leftarrow \langle \sigma, \ell \rangle$
return $(K^{\mathrm{xem}}, \sigma')$
```

```
$\mathsf{TXEM.Enc}(\sigma', \tau)$
$\langle \sigma, \ell \rangle \leftarrow \sigma'$
if $\tau \notin \mathcal{T}_\ell$ :
    return $\perp$
$c \leftarrow \mathsf{TKEM.Enc}(\sigma, \tau)$
return $c$
```

```
$\mathsf{TXEM.Dec}(c, \tau, \ell)$
if $\tau \notin \mathcal{T}_\ell$ :
    return $\perp_{\mathrm{TAG}}$
$K^{\mathrm{kem}} \leftarrow \mathsf{TKEM.Dec}(c, \tau)$
if $K^{\mathrm{kem}} = \perp$:
    return $\perp_{\mathrm{KEM}}$
$K^{\mathrm{xem}} \leftarrow F(K^{\mathrm{kem}}, \ell)$
return $K^{\mathrm{xem}}$
```

Fig. 7. A TagXEM TXEM from a TagKEM TKEM with keyspace $\{0,1\}^k$ and a XOF with seed space $\mathcal{X} = \{0,1\}^k$. The key generation algorithm $\mathsf{TXEM.Kg}$ is unchanged from $\mathsf{TKEM.Kg}$.

```
Oracle $\mathcal{C}(i, \ell)$ (game $G_0$)
$(K_0^{\mathrm{kem}}, \sigma) \leftarrow\!\!\$\ \mathsf{TKEM.Key}_{\mathsf{pk}_i}$
$\mathsf{E}_i \overset{\frown}{\longleftarrow} \sigma$
$K_0^{\mathrm{xem}} \leftarrow F(K_0^{\mathrm{kem}}, \ell)$
$K_1^{\mathrm{xem}} \leftarrow\!\!\$\ \{0,1\}^\ell$
return $K_{b_i}^{\mathrm{xem}}$
```

```
Oracle $\mathcal{C}(i, \ell)$ (game $G_1$)
$(K_0^{\mathrm{kem}}, \sigma) \leftarrow\!\!\$\ \mathsf{TKEM.Key}_{\mathsf{pk}_i}$
$\mathsf{E}_i \overset{\frown}{\longleftarrow} \sigma$
$K_0^{\mathrm{xem}} \leftarrow F(K_0^{\mathrm{kem}}, \ell)$
$K_1^{\mathrm{kem}} \leftarrow\!\!\$\ \{0,1\}^k$
$K_1^{\mathrm{xem}} \leftarrow F(K_1^{\mathrm{kem}}, \ell)$
return $K_{b_i}^{\mathrm{xem}}$
```

Fig. 8. Encryption oracle for the proof of Thm. 5. In blue the code added to G_1 compared to G_0.

4.2 Extending the Output of a TagKEM

First, we show how combining a TagKEM with a fixed output length and a suitable pseudorandom extendable output function (XOF), yields a TagXEM that inherits the MI security of the underlying KEM. Recall that a XOF, for instance SHAKE128 and SHAKE256 as standardized by NIST [35], is a function $F : \mathcal{X} \times \mathbb{Z}_{>0} \to \{0,1\}^*$ for some finite domain $\mathcal{X}$ that on input a seed $s \in \mathcal{X}$ and a desired output length ℓ, outputs a value $y \in \{0,1\}^\ell$. Moreover, if $\ell < \ell'$, then $F(s, \ell)$ is a prefix of $F(s, \ell')$ for all s. This prefix preservation is not a requirement of our constructions; rather we model the property to ensure SHAKE128 and SHAKE256 are suitable real-world instantiations.

As security notion for a XOF F we use its multi-challenge pseudorandomness, which is a standard distinguishing advantage $\mathsf{Adv}_F^{\mathrm{psrnd}}(\mathbb{A})$: an adversary needs to distinguish between either a real oracle that, on input a desired length ℓ, samples a seed $s \leftarrow\!\!\$\ \mathcal{X}$ uniformly at random and returns $F(s, \ell)$, or an ideal oracle that, on input said ℓ, simply returns a uniformly sampled string of length ℓ.

The construction of the TagXEM is given in Fig. 7 and the security claim follows in Thm. 5 . If the PsRND advantage of F is sufficiently small, then TXEM inherits the multi-instance security of TKEM; moreover, as the result holds for arbitrary $\mathcal{T}$ and $\mathcal{T}_\ell$, it holds for the trivial spaces, yielding a slightly simpler XEM from KEM result.

Theorem 5. *Let* TKEM *be a* (γ, δ)*-correct TagKEM sampling keys from* $\{0,1\}^k$ *and with tagspace* $\mathcal{T}$*, let* $F : \{0,1\}^k \times \mathbb{Z}_{>0} \to \{0,1\}^*$ *be a XOF, and let* TXEM *be a TagXEM as given in Fig. 7 for arbitrary* $\mathcal{T}_\ell \subseteq \mathcal{T}$*. Then* TXEM *is* (γ, δ)*-correct, and there are SFBB reductions* $\mathbb{B}$ *and* $\mathbb{C}$ *such that, for every adversary* $\mathbb{A}$*,*

$$\mathsf{Adv}_{\mathrm{TXEM}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{A}) \leq \mathsf{Adv}_{\mathrm{TKEM}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{B}) + 2 \cdot \mathsf{Adv}_F^{\mathrm{psrnd}}(\mathbb{C}) \, .$$

If $\mathbb{A}$ *calls* $\mathcal{C}$ q_c *times and* $\mathcal{D}$ q_d *times, then* $\mathbb{B}$*'s overhead consists of at most* $q_c + q_d$ *evaluations of* F*, while* $\mathbb{C}$*'s overhead consists of doing* κ *executions of* $\mathsf{TKEM.Kg}$*, at most* q_c *executions of* $\mathsf{TKEM.Key}$ *and* $\mathsf{TKEM.Enc}$*, and at most* q_d *executions of* $\mathsf{TKEM.Dec}$*.*

Proof. Let $\mathbb{A}$ be an adversary for $\mathsf{Exp}_{\mathrm{TXEM}}^{(n,\kappa)\text{-ind-cca}\star}$. We will use $\mathrm{G}_0(\mathbb{A})$ as shorthand for that experiment, where in Fig. 8 we have expanded the challenge oracle $\mathcal{C}$ based on the actual construction of $\mathsf{TXEM.Key}$. When called on a desired key length ℓ and a tag τ, $\mathcal{C}$ returns $K_{b_i}^{\mathrm{xem}}$ such that either $K_{b_i}^{\mathrm{xem}}$ is computed as $F(K_0^{\mathrm{kem}}, \ell)$, or it is uniformly drawn from the keyspace $\{0,1\}^\ell$.

Next, let G_1 be as G_0, except that K_1^{xem} is computed as $F(K_1^{\text{kem}}, \ell)$, for some key K_1^{kem} uniformly sampled from the underlying TagKEMs keyspace $\{0,1\}^k$. Thus the challenge now takes the form $F(K_{b_i}^{\text{kem}}, \ell)$.

We claim that there is a reduction $\mathbb{B}$ such that

$$\mathsf{Adv}_{\mathrm{TKEM}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{B}) = 2\Pr[\mathrm{G}_1(\mathbb{A}) = 1] - 1\,.$$

To see this, let $\mathbb{B}$ answer $\mathbb{A}$'s challenge queries by quering its own oracle and, upon receiving K^{kem}, compute and forward $K^{\text{xem}} = F(K^{\text{kem}}, \ell)$. Similarly, whenever $\mathbb{A}$ queries the decryption oracle on (c, ℓ), $\mathbb{B}$ queries its own decryption oracle on c, receives K^{kem} (resp. $\bot/\lightning$) and forwards the reply $F(K^{\text{kem}}, \ell)$ (resp. $\bot/\lightning$) to $\mathbb{A}$. The public keys and calls to the other oracles are simply forwarded, and, when $\mathbb{A}$ outputs $(\mathtt{I}, \hat{b})$, $\mathbb{B}$ halts with the same output.

The simulation of G_1 is perfect, and $\mathbb{B}$ wins iff $\mathbb{A}$ wins. Therefore,

$$\Pr\left[\mathsf{Exp}_{\mathrm{TKEM}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{B}) = 1\right] = \Pr[\mathrm{G}_1(\mathbb{A}) = 1]\,,$$

implying the statement.

We next claim that there is a reduction $\mathbb{C}$ such that

$$\mathsf{Adv}_F^{\text{psrnd}}(\mathbb{C}) = \Pr[\mathrm{G}_0(\mathbb{A}) = 1] - \Pr[\mathrm{G}_1(\mathbb{A}) = 1]\,.$$

$\mathbb{C}$ simulates the game for $\mathbb{A}$ by generating key pairs and challenge bits as needed, forwarding the public keys to $\mathbb{A}$, and whenever $\mathbb{A}$ queries for the encapsulation, decapsulation and corruption oracle, using the TXEM algorithms to give faithful simulations. When $\mathbb{A}$ queries $\mathcal{C}(i, \ell)$, however, $\mathbb{C}$ first runs $\mathsf{TKEM.Enc}_{\mathsf{pk}_i}$ to get a $K^{\text{kem}} \in \{0,1\}^k$ and a state σ. It then sets K_0^{xem} to $F(K_0^{\text{kem}}, \ell)$, before it queries its own challenge oracle on length ℓ, and gives the result to K_1^{xem}. After adding (σ, ℓ) into $\mathtt{E}_i$, as required for the simulation, it returns $K_{b_i}^{\text{xem}}$ to $\mathbb{A}$. Note that if in the PsRND game b is set to 0, then $K_1^{\text{xem}} = F(K, \ell)$, meaning this is a faithful simulation of G_1. If $b = 1$, K_1^{xem} is drawn uniformly from $\{0,1\}^\ell$, making it a faithful simulation of G_0.

Once $\mathbb{A}$ halts with output $(\mathtt{I}, \hat{b})$, $\mathbb{C}$ checks whether $\mathtt{I}$ is eligible, i.e. of the correct cardinality and not containing any corrupted indices. If yes, $\mathbb{C}$ outputs 1 if $\oplus_{i\in\mathtt{I}} b_i = \hat{b}$ and 0 otherwise. If $\mathtt{I}$ is ineligible, it outputs the guess 0. Assuming $\mathbb{A}$'s output is eligible, $\mathbb{C}$ gets the following advantage.

$$\begin{aligned}
\Pr\left[\mathsf{Exp}_F^{\text{psrnd}}(\mathbb{C}) = 1\right] &= \Pr[\mathbb{A} \text{ wins} \land b = 1] + \Pr[\mathbb{A} \text{ loses} \land b = 0] \\
&= \Pr[b = 1]\Pr[\mathrm{G}_0(\mathbb{A}) = 1] + \Pr[b = 0]\Pr[\mathrm{G}_1(\mathbb{A}) = 0] \\
&= \frac{1}{2}\Pr[\mathrm{G}_0(\mathbb{A}) = 1] + \frac{1}{2}\left(1 - \Pr[\mathrm{G}_1(\mathbb{A}) = 1]\right) \\
&= \frac{1}{2}\left(\Pr[\mathrm{G}_0(\mathbb{A}) = 1] - \Pr[\mathrm{G}_1(\mathbb{A}) = 1] + 1\right)\,,
\end{aligned}$$

from which we obtain

$$\begin{aligned}
\mathsf{Adv}_F^{\text{psrnd}}(\mathbb{C}) &= 2 \cdot \Pr\left[\mathsf{Exp}_F^{\text{psrnd}}(\mathbb{C}) = 1\right] - 1 \\
&= 2 \cdot \frac{1}{2}\left(\Pr[\mathrm{G}_0(\mathbb{A}) = 1] - \Pr[\mathrm{G}_1(\mathbb{A}) = 1] + 1\right) - 1 \\
&= \Pr[\mathrm{G}_0(\mathbb{A}) = 1] - \Pr[\mathrm{G}_1(\mathbb{A}) = 1]\,.
\end{aligned}$$

If $\mathbb{A}$'s output was ineligible, $\mathbb{C}$ gets advantage

$$2\Pr[b = 0] - 1 = 0 = \Pr[\mathrm{G}_0(\mathbb{A}) = 1 \mid \mathtt{I} \text{ ineligible}] - \Pr[\mathrm{G}_1(\mathbb{A}) = 1 \mid \mathtt{I} \text{ ineligible}]\,,$$

so the relation holds also in this case. The advantage of $\mathbb{A}$ can be bounded as:

$$\begin{aligned}
\mathsf{Adv}_{\mathrm{TXEM}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{A}) &= 2\Pr[\mathrm{G}_0(\mathbb{A}) = 1] - 1 \\
&= 2\left(\Pr[\mathrm{G}_0(\mathbb{A}) = 1] + \Pr[\mathrm{G}_1(\mathbb{A}) = 1] - \Pr[\mathrm{G}_1(\mathbb{A}) = 1]\right) - 1 \\
&= 2\left(\Pr[\mathrm{G}_0(\mathbb{A}) = 1] - \Pr[\mathrm{G}_1(\mathbb{A}) = 1]\right) + 2\Pr[\mathrm{G}_1(\mathbb{A}) = 1] - 1 \\
&\leq 2 \cdot \mathsf{Adv}_F^{\text{psrnd}}(\mathbb{C}) + \mathsf{Adv}_{\mathrm{TKEM}}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{B})\,,
\end{aligned}$$

yielding the statement.

As for correctness, $\mathsf{TXEM.Kg} = \mathsf{TKEM.Kg}$, so $\gamma_{\mathrm{TXEM}} = \gamma_{\mathrm{TKEM}}$. If a decryption error happens in TKEM, it translates to F being called with the wrong seed, leading to a decryption error in TXEM except in the case of an output collision. Thus, $\delta_{\mathrm{TXEM}} \leq \delta_{\mathrm{TKEM}}$, allowing us to conclude that TXEM is $(\gamma_{\mathrm{TKEM}}, \delta_{\mathrm{TKEM}})$-correct. □

```
$\mathsf{PKE.Enc_{pk}}(m)$
    $(K, c_1) \leftarrow\!\!\$\ \mathsf{XEM.Enc_{pk}}(|m|)$
    $c_2 \leftarrow K \oplus m$
    return $\langle c_1, c_2 \rangle$
```

```
$\mathsf{PKE.Dec_{sk}}(\langle c_1, c_2 \rangle)$
    $K \leftarrow \mathsf{XEM.Dec_{sk}}(c_1, |c_2|)$
    if $K = \perp$ then return $\perp$
    $m \leftarrow K \oplus c_2$
    return $m$
```

```
$\mathsf{PKE'.Enc_{pk}}(m)$
    $(K, \sigma) \leftarrow\!\!\$\ \mathsf{TXEM.Key_{pk}}(|m|)$
    $c_2 \leftarrow K \oplus m$
    $c_1 \leftarrow \mathsf{TXEM.Enc}(\sigma, c_2)$
    return $\langle c_1, c_2 \rangle$
```

```
$\mathsf{PKE'.Dec_{sk}}(\langle c_1, c_2 \rangle)$
    $K \leftarrow \mathsf{TXEM.Dec_{sk}}(c_1, c_2, |c_2|)$
    if $K = \perp$ then return $\perp$
    $m \leftarrow K \oplus c_2$
    return $m$
```

Fig. 9. Two hybrid encryption schemes: PKE (top row) is a conventional hybrid scheme combining a XEM with the OTP to yield a CPA-secure PKE, while PKE' (bottom row) combines a TagXEM with the OTP to yield a CCA-secure PKE. The key generation and checking algorithms are equivalent to their XEM resp. TXEM counterparts.

One concern is whether the PsRND advantage of F will be sufficiently small. Suppose k is the output length of the underlying TagKEM. A generic attacker would always be able to fix $\ell > k$ and evaluate F for, say, N seeds offline in the hope of colliding with any of the challenge evaluations. The PsRND distinguishing advantage of such an adversary is of order $(q_c + q_d)N/2^k$, indicating that the underlying TagKEM already needs to provide keys long enough for Thm. 5 to yield meaningful multi-instance security.

4.3 A PKE Inheriting (Tag)XEM Security

As a multi-instance secure XEM provides us with ephemeral keys of any desired length, we can combine it with an information-theoretic DEM in order to achieve PKE. Here we opt for the one-time-pad (OTP), as it is the simplest and best-known primitive providing perfect secrecy. The beauty of the OTP is that whether you switch out the ephemeral key for a uniform random one, or the message for a uniform random one, the resulting ciphertext distribution is the same. It allows the PKE to tightly inherit the MI-security of the XEM, albeit yielding only real-or-random security under chosen-plaintext attacks. The construction is provided in full in Fig. 9 (top row); the security claim is captured in Thm. 6.

Theorem 6 (ROR-CPA PKE). *Let* XEM *be a* (γ, δ)*-correct XEM, and let* PKE *be a hybrid encryption scheme as given in Fig. 9. Then* PKE *is* (γ, δ)*-correct, and there is a type-preserving SFBB reduction* $\mathbb{B}$ *such that for every adversary* $\mathbb{A}$*,*

$$\mathsf{Adv}_{\mathrm{PKE}}^{(n,\kappa)\text{-ror-cpa}\star}(\mathbb{A}) \leq \mathsf{Adv}_{\mathrm{XEM}}^{(n,\kappa)\text{-ind-cpa}\star}(\mathbb{B}) \,.$$

Proof. $\mathbb{B}$, playing $\mathsf{Exp}_{\mathrm{XEM}}^{(n,\kappa)\text{-ind-cpa}\star}$ is able to perfectly simulate $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-ror-cpa}\star}$ for $\mathbb{A}$ as follows. If $\mathbb{A}$ calls its challenge encryption oracle, $\mathbb{B}$ first acquires an ephemeral OTP key of the correct length from its challenge encryption oracle, and an encapsulation c_1. It next uses K to encrypt the given message, producing the message encapsulation c_2, and returns $\langle c_1, c_2 \rangle$. Corruption oracle calls are simply forwarded. Eventually, when $\mathbb{A}$ terminates on an output $(\mathtt{I}, \hat{b})$, $\mathbb{B}$ terminates on that very same output.

We claim that $\mathbb{B}$ provides a perfect simulation of the $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-ror-cpa}\star}$ game, where the implicit challenge bits in $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-ror-cpa}\star}$ exactly match the unknown ones in $\mathsf{Exp}_{\mathrm{XEM}}^{(n,\kappa)\text{-ind-cpa}\star}$. After all,

$$\left\{K \oplus m \mid m \leftarrow\!\!\$\ \{0,1\}^{\ell}\right\} = \left\{K \oplus m \mid K \leftarrow\!\!\$\ \{0,1\}^{\ell}\right\} \,,$$

so a random ephemeral key in $\mathbb{B}$'s game neatly corresponds to a random message in $\mathbb{A}$'s game. As $\mathbb{A}$'s view after corrupting remains consistent with $\mathsf{Exp}_{\mathrm{PKE}}^{(n,\kappa)\text{-ror-cpa}\star}$, the simulation is perfect and $\mathbb{B}$ wins iff $\mathbb{A}$ wins.

Given the perfect correctness of the OTP, it is easy to see that the correctness parameters transfer directly from XEM to PKE: $\mathsf{PKE.Kg} = \mathsf{XEM.Kg}$, so $\gamma_{\mathrm{PKE}} = \gamma_{\mathrm{XEM}}$; meanwhile, an incorrect decryption in the XEM translates to an incorrect K being xored with c_2 (assuming $K \neq \perp$), leading to incorrect m; otherwise, decryption always succeeds. Therefore, $\delta_{\mathrm{PKE}} = \delta_{\mathrm{XEM}}$. □

One might hope that adding information-theoretic MACs to the DEM would result in the inheritance of CCA security, but that is easier said than shown. For instance, the usual proof technique of a game hop where all

If $\mathbb{A}$ calls oracle $\mathcal{E}(i, m)$	If $\mathbb{A}$ calls oracle $\mathcal{D}(i, \langle c_1, c_2 \rangle)$
$q_i \leftarrow q_i + 1$	$K \leftarrow \mathcal{D}_{\mathbb{B}}(i, \langle c_1, c_2 \rangle, \lvert c_2 \rvert)$
$K \leftarrow \mathcal{C}_{\mathbb{B}}(i, \lvert m \rvert)$	**if** $K = \lightning$ **then return** $\lightning$
$c_2 \leftarrow K \oplus m$	**if** $K = \perp$ **then return** $\perp$
$c_1 \leftarrow \mathcal{E}_{\mathbb{B}}(i, q_i, c_2)$	$m \leftarrow K \oplus c_2$
return $\langle c_1, c_2 \rangle$	**return** m

Fig. 10. The reduction $\mathbb{B}$, playing $\mathsf{Exp}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{TXEM}}$, while giving a perfect simulation of $\mathsf{Exp}^{(n,\kappa)\text{-ror-cca}\star}_{\mathrm{PKE}}$ for $\mathbb{A}$. Not shown are the counters q_i being initialized to 0, and the corruption oracles being forwarded directly. At the end of the game, $\mathbb{B}$ halts with the same output as $\mathbb{A}$.

decryption queries are disallowed does not work: after breaking only a single KEM private key, the reduction will be found out as not being faithful. Sadly, a single-instance break (of the reduction) suffices to show that that reduction cannot demonstrate multi-instance security.

Luckily, TagKEMs allow for a modified hybrid scheme for which the DEM no longer needs to satisfy CCA security for the resulting PKE to be guaranteed CCA-secure: in the single-instance setting, if the TagKEM is CCA-secure, then so is the PKE [2]. We upgrade the construction to use TagXEMs and the OTP in Fig. 9 (bottom row) and show its multi-instance inheritance in Thm. 7.

Theorem 7 (ROR-CCA PKE). *Let* TXEM *be a* (γ, δ)*-correct TagXEM, and let* PKE' *be a hybrid encryption scheme as given in Fig. 9. Then* PKE' *is* (γ, δ)*-correct, and there is a type-preserving SFBB reduction* $\mathbb{B}$ *such that for every adversary* $\mathbb{A}$*,*

$$\mathsf{Adv}^{(n,\kappa)\text{-ror-cca}\star}_{\mathrm{PKE}'}(\mathbb{A}) \leq \mathsf{Adv}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{TXEM}}(\mathbb{B})\,.$$

Proof. This time the reduction $\mathbb{B}$, playing $\mathsf{Exp}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{TXEM}}$, is able to perfectly simulate $\mathsf{Exp}^{(n,\kappa)\text{-ror-cca}\star}_{\mathrm{PKE}}$ for $\mathbb{A}$, including decryption queries. The way $\mathbb{B}$ handles $\mathbb{A}$'s challenge encryption and decryption queries is specified in Fig. 10. Encryption queries are handled essentially as before, just taking into account the slightly different syntax of TagXEMs. Calls to the decryption oracle are forwarded, yielding an ephemeral key that is used to produce the message from c_2. Note that, since the message encapsulation is used as a tag when producing the challenge encryption, $\mathbb{B}$'s decryption oracle will return $\lightning$ iff $\langle c_1, c_2 \rangle$ was previously issued as a challenge. (Essentially, $\mathbb{B}$ can rely on its own experiment to keep track of admin relevant for the experiment it simulates for $\mathbb{A}$.)

Corruption oracle calls are simply forwarded. Eventually, when $\mathbb{A}$ terminates on an output $(\mathtt{I}, \hat{b})$, $\mathbb{B}$ terminates on that very same output.

From here on out, the proof tracks that of Thm. 6, where this time we claim that $\mathbb{B}$ provides a perfect simulation of the $\mathsf{Exp}^{(n,\kappa)\text{-ror-cca}\star}_{\mathrm{PKE}}$ game, where the implicit challenge bits in $\mathsf{Exp}^{(n,\kappa)\text{-ror-cca}\star}_{\mathrm{PKE}}$ exactly match the unknown ones in $\mathsf{Exp}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{TXEM}}$. Furthermore, by inspection, both the corruption oracle and crucially the decryption oracle are impeccable. As the simulation is perfect, $\mathbb{B}$ wins iff $\mathbb{A}$ wins.

As with PKE, the correctness parameters of PKE' transfer directly from TXEM: $\mathsf{PKE.Kg} = \mathsf{XEM.Kg}$, so $\gamma_{\mathrm{PKE}} = \gamma_{\mathrm{XEM}}$. A decryption error in TXEM again translates to the wrong OTP key being xored with c_2, or an unexpected $\perp$. Def. 5 guarantees that with probability at least δ_{TXEM} there are no tags τ such that decrypting fails. Given that the decryption algorithm of PKE' does not allow tampering of the tags without also affecting c_2, PKE' could in fact well end up allowing for much smaller values of δ than that of TXEM. Nevertheless we conclude that $\delta_{\mathrm{PKE}} \leq \delta_{\mathrm{TXEM}}$, and that PKE' is $(\gamma_{\mathrm{TXEM}}, \delta_{\mathrm{TXEM}})$-correct. □

While encouraging, the claim that the constructed PKE inherits the multi-instance security of the TagXEM is dampened by the exponential separation between the ROR security notion and IND, as argued in Sect. 3.4. Indeed, extrapolating to the latter notion by combining Thm. 7 with Cor. 2, we have only achieved the following bound.

Corollary 3. *Let* TXEM *be a* (γ, δ)*-correct TagXEM, and let* PKE' *be a hybrid encryption scheme as given in Fig. 9. Then* PKE' *is* (γ, δ)*-correct, and there is a type-preserving SFBB reduction* $\mathbb{B}$ *such that for every adversary* $\mathbb{A}$*,*

$$\mathsf{Adv}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{PKE}'}(\mathbb{A}) \leq \binom{\kappa}{n} \cdot 2^n \cdot \mathsf{Adv}^{(n,n)\text{-ind-cca}}_{\mathrm{TXEM}}(\mathbb{B})\,,$$

where $\mathbb{B}$*'s overhead is dominated by generating* $\kappa - n$ *fresh keypairs, sampling* κ *bits, and choosing a subset of* $[\kappa]$ *of cardinality* n *uniformly at random.*

```
Oracle $\mathcal{C}_{\mathrm{ROP}}(i, \ell, \Pi)$
$(K_0, \sigma) \leftarrow\!\!\$\ \mathsf{TXEM.Key}_{\mathsf{pk}_i}(\ell)$
$\mathsf{E}_i \overset{\frown}{\longleftarrow} \sigma$
$K_1 \leftarrow \Pi(K_0)$
$\mathbf{return}\ K_{b_i}$
```

Fig. 11. Fig. 6 is upgraded to $\mathsf{Exp}^{(n,\kappa)\text{-rop-cca}\star}_{\mathrm{TXEM}}$ by letting $\mathcal{C}_{\mathrm{ROP}}$ replace $\mathcal{C}$.

4.4 Real-Or-Permuted: A Strengthened Notion for KEM Security

If we want to achieve an IND-CCA PKE more tightly, we seem to need a different notion of security for our TagXEMs. What could such a notion look like?

Our solution is a novel, stronger KEM notion, which we will refer to as "real-or-permuted", or ROP for short. Fig. 11 provides the crucial new challenge oracle. The adversary has to guess whether a tentative K is the one encapsulated under c, or whether an adaptively chosen permutation has been applied to it. As permutations preserve the distribution of the sampling space, there are no choices of Π that make the game generically and trivially winnable.

Technically, we need to specify how the adversary provides Π such that it is guaranteed, or can be checked, to be a permutation. Hence, formally we define ROP with respect to a class of permutations $\mathcal{P}$, reminiscent of for instance key-dependent message [24] or related-key attack [10] definitions. We require that membership $\Pi \in \mathcal{P}$ is easy to check (e.g. ROP can simply index an element in $\mathcal{P}$) and that, by definition, $\mathcal{P}$ can be verified to indeed only contain permutations. For our main results, it suffices if $\mathcal{P}$ is the class of one-time pads, in the sense that Π specifies the key (or pad) of the one-time pad enciphering. Henceforth, we will assume that ROP is defined with respect to that class, unless explicitly stated otherwise.

The new notion ROP and IND relate to each other much the same way as IND and ROR for PKE. It is not hard to see that ROP tightly implies IND, whereas the other direction seems to incur the same loss as the ROR-to-IND implication for PKE (see App. B). For completeness, ROP lends itself equally well to XEMs and KEMs, or notions without corruptions or a decryption oracle. Finally, if any of the above primitives are constructed using an IND-secure PKE (e.g. using a Fujisaki–Okamoto style transform [21, 22, 28]), then achieving ROP is as easy as achieving IND: simply let K be the "left" message, and $\Pi(K)$ be the "right"!

4.5 PKE′ Tightly Inherits IND-CCA Security

Using ROP in place of IND, we are able to show directly that the PKE constructions of Fig. 9 are IND-CPA resp. IND-CCA secure, by (as before) giving a (Tag)XEM reduction that provides a perfect simulation for the PKE adversary.

The crucial observation is that for any pair of messages $m_0, m_1 \in \{0,1\}^\ell$, there exist a permutation $\Pi_{m_0 \to m_1}$ on $\{0,1\}^\ell$ such that the message encapsulations are related as $K \oplus m_1 = \Pi_{m_0 \to m_1}(K) \oplus m_0$. Namely, the permutation that on input K, outputs $m_0 \oplus m_1 \oplus K$.

Theorem 8 (IND-CCA PKE). *Let* TXEM *be a* (γ, δ)*-correct TagXEM, and let* PKE' *be a hybrid encryption scheme as given in Fig. 9. Then* PKE' *is* (γ, δ)*-correct, and there is a type-preserving SFBB reduction* $\mathbb{B}$ *such that for every adversary* $\mathbb{A}$*,*

$$\mathsf{Adv}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{PKE}'}(\mathbb{A}) \leq \mathsf{Adv}^{(n,\kappa)\text{-rop-cca}\star}_{\mathrm{TXEM}}(\mathbb{B})\,.$$

Proof. $\mathbb{B}$ is given in Fig. 12. The simulation is as in the proof of Theorem 7, except for the simulation of the challenge encryption oracle: here, $\mathbb{B}$ now forwards the permutation $\Pi_{m_0 \to m_1}$ that on input K outputs $m_0 \oplus m_1 \oplus K$, receiving as a challenge either the honest key K or the permuted key $\Pi_{m_0 \to m_1}(K)$. To see that the simulation is perfect, consider the view of $\mathbb{A}$ in the case that $\mathbb{A}$ challenges instance i on messages m_0, m_1, then corrupts, receiving sk_i and decrypting the challenge. If the decryption is correct and $b_i = 0$, $\mathbb{A}$ recovers the honest key K_0, and therefore the message m_0. While, whenever $b_i = 1$, $\mathbb{A}$ still recovers K_0, but now the challenge was produced by encrypting m_0 using the permuted key. Decrypting will then yield

$$K_0 \oplus c_2 = K_0 \oplus K_1 \oplus m_0 = K_0 \oplus m_0 \oplus m_1 \oplus K_0 \oplus m_0 = m_1\,,$$

just as expected. Note how this implies that the bits of each game correspond exactly to each other; the simulation is perfect, and $\mathbb{B}$ wins iff $\mathbb{A}$ wins.

The correctness parameters of PKE' again transfer directly from TXEM (see Thm. 7). □

```
If $\mathbb{A}$ calls oracle $\mathcal{E}(i, m_0, m_1)$
    if $m_0 \not\sim m_1$ then return $\lightning$
    $q_i \leftarrow q_i + 1$
    $K \leftarrow \mathcal{C}_{\mathrm{ROP}}(i, |m_0|, \Pi_{m_0 \to m_1})$
    $c_2 \leftarrow K \oplus m_0$
    $c_1 \leftarrow \mathcal{E}(i, q_i, c_2)$
    return $\langle c_1, c_2 \rangle$
```

```
If $\mathbb{A}$ calls oracle $\mathcal{D}(i, \langle c_1, c_2 \rangle)$
    $K \leftarrow \mathcal{D}_{\mathbb{B}}(i, \langle c_1, c_2 \rangle, |c_2|)$
    if $K = \lightning$ then return $\lightning$
    if $K = \perp$ then return $\perp$
    $m \leftarrow K \oplus m$
    return $\langle c_1, c_2 \rangle$
```

Fig. 12. The reduction $\mathbb{B}$, playing $\mathsf{Exp}_{\mathrm{TXEM}}^{(n,\kappa)\text{-rop-cca}\star}$, while giving a perfect simulation of $\mathsf{Exp}_{\mathrm{PKE}'}^{(n,\kappa)\text{-ind-cca}\star}$ for $\mathbb{A}$. Counters q_i are initialized to 0, corruption queries are forwarded directly. $\mathbb{B}$ halts with the same output as $\mathbb{A}$.

```
$\mathsf{TXEM.Kg}$
    $(\mathsf{pk}', \mathsf{sk}') \leftarrow\!\!\$\ \mathsf{KEM.Kg}$
    $\mathsf{pk} \leftarrow \mathsf{pk}'$
    $\mathsf{sk} \leftarrow \langle \mathsf{pk}', \mathsf{sk}' \rangle$
    return $(\mathsf{pk}, \mathsf{sk})$

$\mathsf{TXEM.Check}(\mathsf{pk}, \mathsf{sk})$
    $\langle \mathsf{pk}', \mathsf{sk}' \rangle \leftarrow \mathsf{sk}$
    if $\mathsf{pk} \neq \mathsf{pk}'$ then return $0$
    return $\mathsf{KEM.Check}(\mathsf{pk}', \mathsf{sk}')$

$\mathsf{TXEM.Enc}(\sigma, \tau)$
    $\langle c, K^{\mathrm{mac}} \rangle \leftarrow \sigma$
    $\mathsf{mac} \leftarrow \mathsf{MAC}_{K^{\mathrm{mac}}}(\tau)$
    return $\langle c, \mathsf{mac} \rangle$
```

```
$\mathsf{TXEM.Key}_{\mathsf{pk}}(\ell)$
    $(K^{\mathrm{kem}}, c) \leftarrow\!\!\$\ \mathsf{KEM.Enc}_{\mathsf{pk}}$
    $\ell' \leftarrow \ell + \ell_{\mathsf{mackey}}$
    $K^{\mathrm{mac}} \| K^{\mathrm{xem}} \leftarrow F(\mathsf{pk}, c, K^{\mathrm{kem}}, \ell')$
    $\sigma \leftarrow \langle c, K^{\mathrm{mac}} \rangle$
    return $K^{\mathrm{xem}}$

$\mathsf{TXEM.Dec}_{\mathsf{sk}}(\langle c, \mathsf{mac} \rangle, \tau, \ell)$
    $\langle \mathsf{pk}', \mathsf{sk}' \rangle \leftarrow \mathsf{sk}$
    $K^{\mathrm{kem}} \leftarrow \mathsf{KEM.Dec}_{\mathsf{sk}'}(c)$
    if $K^{\mathrm{kem}} = \perp$ then return $\perp$
    $\ell' \leftarrow \ell + \ell_{\mathsf{mackey}}$
    $K^{\mathrm{mac}} \| K^{\mathrm{xem}} \leftarrow F(\mathsf{pk}', c, K^{\mathrm{kem}}, \ell')$
    if $\mathsf{MAC}_{K^{\mathrm{mac}}}(\tau) \neq \mathsf{mac}$ then return $\perp$
    return $K^{\mathrm{xem}}$
```

Fig. 13. A TagXEM from a KEM, a MAC, and an XOF F.

We leave it to the reader to verify that as before, employing a ROP-CPA XEM in place of the TagXEM yields IND-CPA security for the PKE of Fig. 9 (top row), by adapting the proof of Thm. 6 to the above. We again stress that using an information-theoretically CCA-secure DEM together with a CCA XEM does not seem to yield a proof of CCA inheritance to the PKE (see Sect. 4.3).

4.6 TagXEM from a KEM, a MAC, and a Random Oracle

With Thm. 8, we achieved what we set out to do: demonstrating tight MI inheritance from a TagXEM to an IND-CCA PKE. However, AGK only showed how to construct an IND-CCA KEM, providing a reduction to the MI-GapCDH assumption in the programmable random oracle model. Without the crucial support of tags, our construction only achieves CPA security. Furthermore, Thm. 5 does *not* easily transfer to the ROP setting: it is not clear how to combine a ROP-CCA KEM with a XOF to yield a ROP-CCA XEM.

We complete the picture by providing a TagXEM construction from a KEM, a MAC, and a XOF. Our construction (Fig. 13) is inspired by Abe et al.'s TagKEM construction [2] and we show that with an information-theoretic MAC, if the KEM is perfectly correct, has unique encapsulations [25] and is multi-instance one-way secure under plaintext-checking attacks (OW-PCA), then the TagXEM is ROP-CCA secure in the programmable random oracle model (to model the XOF). Before stating our concrete security result (Thm. 9), we will define the relevant concepts and advantages below.

Preliminaries. One-wayness for KEMs tasks an adversary to retrieve the ephemeral key that has been encapsulated, given the public key and the encapsulation. In the multi-instance setting, an adversary has access to many public keys and various encapsulations per public key and endeavours to find ephemeral keys for encapsulations for as many different public keys as possible (no reward for breaking multiple encapsulations under the same public key).

```
$\mathsf{Exp}^{(n,\kappa)\text{-ow-pca}\star}_{\mathrm{KEM}}(\mathbb{A})$
    $(\mathsf{pk}_1,\mathsf{sk}_1),\ldots,(\mathsf{pk}_\kappa,\mathsf{sk}_\kappa) \leftarrow\!\!\$\ \mathsf{KEM.Kg}$
    $(\mathtt{I},(j_i,\hat{K}_i)_{i\in\mathtt{I}}) \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{P},\mathcal{K}}(\mathsf{pk}_1,\ldots,\mathsf{pk}_\kappa)$
    $\mathbf{if}\ |\mathtt{I}| \neq n \vee \mathtt{I}\cap\mathtt{K}\neq\emptyset\ \mathbf{then\ return}\ 0$
    $\mathbf{return}\ \bigwedge_{i\in\mathtt{I}} \mathtt{P}_i[j_i] = \hat{K}_i$

$\mathcal{P}(i,c,K)$
    $K' \leftarrow \mathsf{KEM.Dec}_{\mathsf{sk}_i}(c)$
    $\mathbf{return}\ K = K'$

$\mathcal{E}(i)$
    $(K,c) \leftarrow\!\!\$\ \mathsf{KEM.Enc}_{\mathsf{pk}_i}$
    $\mathtt{P}_i \overset{\frown}{\longleftarrow} K$
    $\mathbf{return}\ c$

$\mathcal{K}(i)$
    $\mathtt{K} \overset{\cup}{\longleftarrow} i$
    $\mathbf{return}\ \mathsf{sk}_i$
```

Fig. 14. Multi-instance one-way security in the presence of plaintext checking attacks.

Plaintext-checking attacks (PCA) were introduced by Okamoto and Pointcheval [36, Definition 8] in a single-user public key encryption setting. Intuitively, PCA provides the adversary access to an oracle that, on input a pair (m, c) determines whether c encrypts m or not; more formally [1], the oracle checks whether c decrypts to m or not. In the context of KEMs, the PCA oracle takes a pair (K^{kem}, c) as input and determines whether c decapsulates to K^{kem} or not. The multi-user or multi-instance generalization is straightforward and the definition (in its modern decryption incarnation) inherently deals with imperfect correctness in the decryption.

Definition 7 considers one-wayness under plaintext checking attacks. For standard ElGamal KEM, where a (multiplicative) discrete-log group with generator g and of prime order q is given as part of the parameters, a public key consists of $h = g^x$ with $x \leftarrow\!\!\$\ \mathbb{Z}_q$ the private key, and an encapsulation outputs $(K^{\text{kem}}, c) = (h^r, g^r)$ for random $r \leftarrow\!\!\$\ \mathbb{Z}_q$, the one-wayness problem (in the single-user case) is equivalent to the computational Diffie–Hellman (CDH) problem. The plaintext checking oracle allows an adversary to learn, for group elements (k, c) of its choice, whether $k = c^x$ or not. The corresponding hardness assumption for OW-PCA is known as the Strong CDH assumption. An even stronger assumption is the GapCDH assumption, where an adversary instead can use an oracle that determines whether a quadruple of group elements is a Diffie–Hellman tuple or not.

Definition 7 (OW-PCA). *Let* KEM *be a key encapsulation mechanism. Then the one-way advantage under plaintext-checking attacks of an adversary* $\mathbb{A}$ *is*

$$\mathsf{Adv}^{(n,\kappa)\text{-ow-pca}\star}_{\mathrm{KEM}}(\mathbb{A}) = \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-ow-pca}\star}_{\mathrm{KEM}}(\mathbb{A}) = 1\right],$$

where the experiment is defined in Fig. 14.

In addition to perfect correctness and OW-PCA security, the security reduction for our construction (Thm. 9) relies on two further properties of the underlying KEM. Unique encapsulation captures that for a fixed public key and ephemeral key, the encapsulation corresponding to that ephemeral key is unique (without saying anything about how to compute it). Unique encapsulations have been used before, for instance by Heuer et al. [25] (see also Remark 4 below).

Definition 8 (Unique Encapsulation). *Let* KEM *be a perfectly correct KEM. Then it has unique encapsulations iff*

$$\Pr\left[\begin{array}{l}(\mathsf{pk},\mathsf{sk}) \leftarrow\!\!\$\ \mathsf{KEM.Kg}\\ (K_0^{kem}, c_0) \leftarrow\!\!\$\ \mathsf{KEM.Enc}_{\mathsf{pk}}\\ (K_1^{kem}, c_1) \leftarrow\!\!\$\ \mathsf{KEM.Enc}_{\mathsf{pk}}\end{array} : K_0^{kem} = K_1^{kem} \wedge c_0 \neq c_1\right] = 0\,.$$

The second additional property we require from the KEM is that collisions amongst encapsulations (under a single randomly drawn public key) are suitably rare. Def. 9 captures the relevant probability of a k-way encapsulation collision. If a KEM is perfectly correct with unique encapsulations, then colliding encapsulations are equivalent to colliding ephemeral keys; if, as is usually the case, these ephemeral keys are furthermore chosen uniformly at random from a finite set $\mathcal{X}$, we can upper bound $\epsilon_k(q)$ by $q^k/|\mathcal{X}|^{k-1}$ using a standard bound on k-way collisions (see e.g. [37, Appendix B]).

Definition 9 (Encapsulation Multi-Collisions). *Let* KEM *be a KEM, and let* $q, k \in \mathbb{Z}_{>1}$ *be parameters. Then the k-out-of-q encapsulation multi-collision probability is*

$$\epsilon_k(q) = \Pr\left[\begin{array}{l}(\mathsf{pk},\mathsf{sk}) \leftarrow\!\!\$\ \mathsf{KEM.Kg}\\ \forall_{i\in[q]}(K_i^{kem}, c_i) \leftarrow\!\!\$\ \mathsf{KEM.Enc}_{\mathsf{pk}}\end{array} : \exists_{\mathtt{J}\subseteq[q],|\mathtt{J}|=k}\forall_{i,j\in\mathtt{J}}c_i = c_j\right].$$

For completeness, we also present definitions of a deterministic message authentication code, so we dispense with an explicit verification algorithm in Def. 10 (for concreteness, we restrict to bitstrings for both keys and tags, of length ℓ_{mackey} and ℓ_{mac} respectively), and an information-theoretic notion of forgeries (Def. 11) where we use the same parameter k as above (or rather $k-1$ in Thm. 9), but this time to denote the number of valid message–tag pairs available to an adversary. The usual choice is $k=1$, e.g. when considering strongly universal$_2$ hash functions, but Wegman and Carter [40] already investigated $k>1$. Provided ℓ_{mackey} is large enough (at least $k \cdot \ell_{\mathsf{mac}}$), one can achieve $\hat{\epsilon}_k = 2^{-\ell_{\mathsf{mac}}}$, which is optimal.

Definition 10 (Message Authentication Code (MAC)). *A message authentication code* MAC *is a pair of algorithms* $\mathsf{MAC.Kg}$ *and* $\mathsf{MAC.Mac}$*, where* $\mathsf{MAC.Kg}$ *randomly generates a* $K^{mac} \in \{0,1\}^{\ell_{\mathsf{mackey}}}$*, and the deterministic* $\mathsf{MAC.Mac}$ *takes a key* K^{mac} *and a message* $m \in \mathcal{M}$ *to output tag* $\mathsf{mac} \leftarrow \mathsf{MAC.Mac}_{K^{mac}}(m) \in \{0,1\}^{\ell_{\mathsf{mac}}}$.

Definition 11 (Information-Theoretic MAC Forgeries). *Let* MAC *be given and let* $k \in \mathbb{Z}_{\geq 0}$ *be a parameter, then the forging advantage after observing* k *valid message–tag pairs is defined as*

$$\hat{\epsilon}_k = \max_{\substack{\forall i \in \{0\} \cup [k] \\ (m_i, \mathsf{mac}_i)}} \Pr\left[\mathsf{MAC.Mac}_{K^{mac}}(m_0) = \mathsf{mac}_0 \;\middle|\; \forall_{i \in [k]} \mathsf{MAC.Mac}_{K^{mac}}(m_i) = \mathsf{mac}_i\right] .$$

Security Claim. With all elements in place, we can state the security of Fig. 13's TXEM, in Thm. 9. The security bound depends on a tuning parameter k that feeds into both the collision probability of the underlying KEM and the forgery advantage of the MAC, with opposite effects. The ability to tune the bound therefore allows some flexibility when instantiating the three underlying primitives KEM, MAC, and XOF: for fixed q_c, increasing k will result in a smaller upper bound on $\epsilon_k(q_c)$, but to ensure that $\hat{\epsilon}_{k-1}$ does not dominate, it might then be necessary to increase the key size ℓ_{mackey} (and possibly tag size ℓ_{mac}) of the information-theoretic MAC (see Cor. 5 for a concrete instantiation). Otherwise, instantiating the information-theoretic MAC and the XOF is relatively straightforward (with the usual ROM caveats for the latter).

Theorem 9. *Let* TXEM *be as in Fig. 13, let* KEM *be a perfectly correct KEM with unique encapsulations, and let* $k \in \mathbb{Z}_{>1}$*. Then there is an SFBB reduction* $\mathbb{B}$ *such that, for all* $\mathbb{A}$ *that makes* q_c *challenge and* q_d *decryption oracle queries,*

$$\mathsf{Adv}^{(n,\kappa)\text{-rop-cca}\star}_{\mathrm{TXEM}}(\mathbb{A}) \leq \mathsf{Adv}^{(n,\kappa)\text{-ow-pca}\star}_{\mathrm{KEM}}(\mathbb{B}) + 2\big(q_d \hat{\epsilon}_{k-1} + \epsilon_k(q_c)\big)$$

in the programmable random oracle model, where $\hat{\epsilon}_{k-1}$ *is the forging advantage after observing* $k-1$ *valid message–tag pairs (Def. 11) and* $\epsilon_k(q_c)$ *is the* k*-out-of-*q_c *encapsulation multi-collision probability of KEM (Def. 9). If* $\mathbb{A}$ *makes* q_f *queries to the random oracle, then* $\mathbb{B}$ *makes at most* q_f *queries to its plaintext checking oracle.*

Proof. For simplicity, we will in the following employ the convention $\mathsf{MAC}_{K^{\mathrm{mac}}}(m) := \mathsf{MAC.Mac}_{K^{\mathrm{mac}}}(m)$.

The game $\mathrm{G}_0(\mathbb{A})$, given in Fig. 15, is simply the ROP-CCA TagXEM experiment of Figs. 6 and 11 instantiated with the TagXEM construction of Fig. 13, with slightly simplified key management and including the XOF modelled as a lazily sampled random oracle $\mathcal{F}$. For the latter, we treat the bit string $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}})$ as a list and use $\overset{\frown}{\longleftarrow}\$$ to sample (in this case a uniform random bit) and append to that string. The while loop thus ensures the correct prefix behaviour, while keeping the code concise. As output we take the first ℓ bits of $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}})$, denoted $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}})[\![\ell]\!]$.

In G_1 (Fig. 16), we make the $\mathcal{C}$ oracle independent of the ephemeral key returned by $\mathsf{KEM.Enc}$. To achieve this, the random oracle is altered so that now, whenever $\mathcal{F}$ is called together with consistent K^{kem} and c, the value of K^{kem} gets stored in a list $\mathtt{K}$. In the case that the correct key has been called to $\mathcal{F}$ before, we then simply retrieve it and call $\mathcal{F}$ as usual. Otherwise, we pre-sample K^{mac} and K^{xem}, and store them in the list $\mathtt{D}_i(c)$. Then, once $\mathcal{F}$ is called on that ciphertext with the corresponding K^{kem}, we program the random oracle using the value stored in $\mathtt{D}_i(c)$, ensuring consistency. Note how the programming does not alter the output distribution of $\mathcal{F}$, as each pre-sampled value is sampled uniformly at random, and is only ever programmed to a single element.

Additionally, we introduce the counters $ctr(\cdot,\cdot)$ to detect whether a k-wise multi-collision is created by the $\mathcal{C}$ oracle and, if so, the flag $\mathsf{bad}_{\mathsf{mc}}$ is set. This multi-collision detection does not change the behaviour of the game—it will assist us bounding another bad event in the next hop. We therefore have that G_0 and G_1 behave identically and

$$\Pr[\mathrm{G}_0(\mathbb{A})] - \Pr[\mathrm{G}_1(\mathbb{A})] = 0 . \tag{1}$$

In $\mathrm{G}_2(\mathbb{A})$ (Fig. 17), we make the decryption oracle independent of the private keys by leveraging the lists introduced in $\mathrm{G}_1(\mathbb{A})$: if $\mathbb{A}$ calls for the decryption of c under sk_i and $\mathtt{K}_i(c)$ has already been defined, we simply retrieve K^{kem} and call the random oracle as usual. Otherwise, we can conclude that $\mathcal{F}$ is yet to be called on the

```
Game $\mathrm{G}_0(\mathbb{A})$
$(\mathsf{pk}_1, \mathsf{sk}_1), \ldots, (\mathsf{pk}_\kappa, \mathsf{sk}_\kappa) \leftarrow\!\!\$\ \mathsf{KEM.Kg}$
$b_1, \ldots, b_\kappa \leftarrow\!\!\$\ \{0,1\}$
$(\mathtt{I}, \hat{b}) \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{C},\mathcal{E},\mathcal{D},\mathcal{K},\mathcal{B},\mathcal{F}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
if $|\mathtt{I}| \neq n \vee \mathtt{I} \cap (\mathtt{K} \cup \mathtt{B}) \neq \emptyset$ then $\hat{b} \leftarrow\!\!\$\ \{0,1\}$
return $\oplus_{i \in \mathtt{I}}\, b_i = \hat{b}$

Oracle $\mathcal{D}(i, \langle\langle c, \mathsf{mac}\rangle, \tau\rangle, \ell)$
if $\langle\langle c, \mathsf{mac}\rangle, \tau\rangle \in \mathtt{C}_i$ then return $\lightning$
$K^{\mathrm{kem}} \leftarrow \mathsf{KEM.Dec}_{\mathsf{sk}_i}(c)$
if $K^{\mathrm{kem}} = \perp$ then return $\perp$
$\ell' \leftarrow \ell + \ell_{\mathsf{mackey}}$
$K^{\mathrm{mac}} \| K^{\mathrm{xem}} \leftarrow \mathcal{F}(\mathsf{pk}_i, c, K^{\mathrm{kem}}, \ell')$
if $\mathsf{MAC}_{K^{\mathrm{mac}}}(\tau) \neq \mathsf{mac}$ then return $\perp$
return $K^{\mathrm{xem}}$
```

```
Oracle $\mathcal{C}(i, \ell, \Pi)$
$(K^{\mathrm{kem}}, c) \leftarrow\!\!\$\ \mathsf{KEM.Enc}_{\mathsf{pk}_i}$
$\ell' \leftarrow \ell + \ell_{\mathsf{mackey}}$
$K^{\mathrm{mac}} \| K_0^{\mathrm{xem}} \leftarrow \mathcal{F}(\mathsf{pk}_i, c, K^{\mathrm{kem}}, \ell')$
$\mathtt{E}_i \overset{\frown}{\leftarrow} \langle c, K^{\mathrm{mac}}\rangle$
$K_1^{\mathrm{xem}} \leftarrow \Pi(K_0^{\mathrm{xem}})$
return $K_{b_i}^{\mathrm{xem}}$

Oracle $\mathcal{E}(i, j, \tau)$
if $\mathtt{E}_i[j] = \emptyset$ then return $\lightning$
$\langle c, K^{\mathrm{mac}}\rangle \leftarrow \mathtt{E}_i[j], \mathtt{E}_i[j] \leftarrow \emptyset$
$\mathsf{mac} \leftarrow \mathsf{MAC}_{K^{\mathrm{mac}}}(\tau)$
$\mathtt{C}_i \overset{\cup}{\leftarrow} \langle\langle c, \mathsf{mac}\rangle, \tau\rangle$
return $\langle c, \mathsf{mac}\rangle$

Oracle $\mathcal{F}(\mathsf{pk}, c, K^{\mathrm{kem}}, \ell)$
while $\left|\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}})\right| < \ell$
    $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}}) \overset{\frown}{\leftarrow}\!\!\$\ \{0,1\}$
return $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}})[\![\ell]\!]$
```

Fig. 15. The game $\mathrm{G}_0(\mathbb{A})$ with $\mathcal{F}$ modelled as a lazily sampled random oracle. Not shown are the corruption oracles, which are as in Fig. 6, and will remain unchanged over the course of the game hops.

relevant values, and thus use $\mathtt{D}_i(c)$ to retrieve/define K^{mac} and check the validity of mac. If the latter is invalid, we return $\perp$; otherwise, the event $\mathsf{bad}_{\mathsf{mf}}$ occurs (and we may abort $\mathrm{G}_2(\mathbb{A})$ with any output).

By design, G_1 and G_2 are identical-until-$\mathsf{bad}_{\mathsf{mf}}$, and $\mathsf{bad}_{\mathsf{mf}}$ only happens if the adversary is able to produce a valid MAC forgery, possibly after having seen a number of valid mac produced under the same key by the game's oracles. We use $\mathsf{bad}_{\mathsf{mc}}$ to bound the number of valid mac an adversary has seen: we claim that if $\mathsf{bad}_{\mathsf{mc}}$ has not been set, that number is at most $k-1$ (see the argument below), implying that the probability of a forgery at that point is at most $\hat{\epsilon}_{k-1}$ (Def. 11) and by a simple union bound $\Pr[\mathsf{bad}_{\mathsf{mf}} \mid \neg\mathsf{bad}_{\mathsf{mc}}] \leq q_d \hat{\epsilon}_{k-1}$ where q_d is the number of decryption calls.

Back to flag $\mathsf{bad}_{\mathsf{mc}}$: assume it has not yet been set, thus for all pk and c, the counters $ctr(\mathsf{pk}, c) \leq k-1$. As the KEM is perfectly correct and has unique encapsulations, the implication is that, for any input-triple, the number of "delayed" $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}})$ calls during $\mathcal{C}$ is at most $k-1$ and each such $\mathcal{C}$ call can lead to at most one valid mac to be output by $\mathcal{E}$, proving our claim from above. Finally, we note that $\Pr[\mathsf{bad}_{\mathsf{mc}}] \leq \epsilon_k(q_c)$ where q_c is the total number of $\mathcal{C}$ queries (here we use that $\epsilon_k(x) + \epsilon_k(y) \leq \epsilon_k(x+y)$).

The arguments above show that

$$\Pr[\mathrm{G}_1(\mathbb{A})] - \Pr[\mathrm{G}_2(\mathbb{A}) \mid \neg\mathsf{bad}_{\mathsf{mf}}] \leq \Pr[\mathrm{G}_2(\mathbb{A}) \ :\ \mathsf{bad}_{\mathsf{mf}}] \leq q_d \hat{\epsilon}_{k-1} + \epsilon_k(q_c)\,. \tag{2}$$

Now $\mathbb{B}$, given in Fig. 18, gives a faithful simulation of G_2 while playing $\mathsf{Adv}^{(n,\kappa)\text{-ow-pca}\star}(\mathbb{A})$, using its encryption oracle in place of KEM.Enc to produce the challenges in $\mathcal{C}$, and its plaintext checking oracle in place of KEM.Dec to check for consistent inputs to $\mathcal{F}$. At the end of its run, $\mathbb{A}$ returns a list of targets $\mathtt{I}$ and a guess $\hat{b}$. Assuming $\mathsf{bad}_{\mathsf{mf}}$ was not triggered, and that none of the i in $\mathtt{I}$ were corrupted, $\mathbb{B}$ proceeds to check whether correct decapsulations were queried to the random oracle together with a challenge ciphertext produced under each targeted key, and collects one such decapsulation per key from $\mathtt{K}_i$. It finally halts with the collected keys, together with $\mathtt{I}$ and a set of indices j_i specifying the targeted challenge.

Let Q be the event that $\mathbb{A}$ returns a list $\mathtt{I}$ of uncompromised instances, for which $\mathcal{F}$ was queried with each key pk_i in $\mathtt{I}$ *at least once* with a K^{kem} satisfying $K^{\mathrm{kem}} = \mathsf{KEM.Dec}_{sk_i}(c)$. Note how $\neg Q$ then implies that either at least one challenge bit b_i is information-theoretically hidden from $\mathbb{A}$, and therefore also $\oplus_{i \in \mathtt{I}} b_i$, or the $\mathtt{I}$ output

```
Oracle $\mathcal{C}(i, \ell, \Pi)$
$(\,\cdot\,, c) \leftarrow\!\!\$\ \mathsf{KEM.Enc}_{\mathsf{pk}_i}$
$ctr(\mathsf{pk}_i, c) \leftarrow ctr(\mathsf{pk}_i, c) + 1$
if $ctr(\mathsf{pk}_i, c) \geq k$ then $\mathsf{bad}_{\mathsf{mc}} \leftarrow \mathsf{true}$
$\ell' \leftarrow \ell + \ell_{\mathsf{mackey}}$
if $\mathtt{K}_i(c) \neq \emptyset$
  $K^{\mathrm{kem}} \leftarrow \mathtt{K}_i(c)$
  $K^{\mathrm{mac}} \| K_0^{\mathrm{xem}} \leftarrow \mathcal{F}(\mathsf{pk}_i, c, K^{\mathrm{kem}}, \ell')$
else
  while $|\mathtt{D}_i(c)| < \ell'$
    $\mathtt{D}_i(c) \overset{\frown}{\longleftarrow}\!\$\ \{0, 1\}$
  $K^{\mathrm{mac}} \| K_0^{\mathrm{xem}} \leftarrow \mathtt{D}_i(c)[\![\ell']\!]$
$\mathtt{E}_i \overset{\frown}{\longleftarrow} \langle c, K^{\mathrm{mac}} \rangle$
$K_1^{\mathrm{xem}} \leftarrow \Pi(K_0^{\mathrm{xem}})$
return $K_{b_i}^{\mathrm{xem}}$
```

```
Oracle $\mathcal{F}(\mathsf{pk}, c, K^{\mathrm{kem}}, \ell)$
if $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}}) = \varepsilon$ then
  $\forall_{i \in [\kappa]} \mathsf{pk}_i = \mathsf{pk}$
    if $\mathsf{KEM.Dec}_{\mathsf{sk}_i}(c) = K^{\mathrm{kem}}$ then
      $\mathtt{K}_i(c) \leftarrow K^{\mathrm{kem}}$
      if $\mathtt{D}_i(c) \neq \varepsilon$ then
        $\mathtt{F}(\mathsf{pk}_i, c, K^{\mathrm{kem}}) \leftarrow \mathtt{D}_i(c)$
while $\left|\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}})\right| < \ell$
  $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}}) \overset{\frown}{\longleftarrow}\!\$\ \{0, 1\}$
return $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}})[\![\ell]\!]$
```

Fig. 16. The oracles modified in game G_1, in which $\mathcal{C}$ is made independent of the sampled K^{kem}. The flag $\mathsf{bad}_{\mathsf{mc}}$ is initialized to false and the counters $ctr(\cdot, \cdot)$ to 0. Changes in blue.

```
Oracle $\mathcal{D}(i, \langle\langle c, \mathsf{mac}\rangle, \tau\rangle, \ell)$
if $\langle\langle c, \mathsf{mac}\rangle, \tau\rangle \in \mathtt{C}_i$ then return $\lightning$
if $\mathtt{K}_i(c) = \emptyset$
  while $|\mathtt{D}_i(c)| < \ell_{\mathsf{mackey}}$
    $\mathtt{D}_i(c) \overset{\frown}{\longleftarrow}\!\$\ \{0, 1\}$
  $K^{\mathrm{mac}} \leftarrow \mathtt{D}_i(c)[\![\ell_{\mathsf{mackey}}]\!]$
  if $\mathsf{MAC}_{K^{\mathrm{mac}}}(\tau) \neq \mathsf{mac}$ then return $\perp$
  else $\mathsf{bad}_{\mathsf{mf}} \leftarrow$ true and abort
$K^{\mathrm{kem}} \leftarrow \mathtt{K}_i(c)$
$\ell' \leftarrow \ell + \ell_{\mathsf{mackey}}$
$K^{\mathrm{mac}} \| K^{\mathrm{xem}} \leftarrow \mathcal{F}(\mathsf{pk}_i, c, K^{\mathrm{kem}}, \ell')$
if $\mathsf{MAC}_{K^{\mathrm{mac}}}(\tau) \neq \mathsf{mac}$ then return $\perp$
return $K^{\mathrm{xem}}$
```

Fig. 17. The modified $\mathcal{D}$ oracle in $\mathrm{G}_2(\mathbb{A})$ (changes in blue) is made independent of the private keys. The $\mathsf{bad}_{\mathsf{mf}}$ flag, initialized to false, is set when a valid forgery is made without $\mathcal{F}$ having been called on the relevant values.

by $\mathbb{A}$ is ineligible; in either case, $\mathbb{A}$ gets advantage 0. This gives us,

$$\begin{aligned}
\Pr[\mathrm{G}_2(\mathbb{A}) \mid \neg\mathsf{bad}_{\mathsf{mf}}] &= \Pr[\mathrm{G}_2(\mathbb{A}) \mid Q \wedge \neg\mathsf{bad}_{\mathsf{mf}}] \cdot \Pr[Q \mid \neg\mathsf{bad}_{\mathsf{mf}}] \\
&\quad + \Pr[\mathrm{G}_2(\mathbb{A}) \mid \neg Q \wedge \neg\mathsf{bad}_{\mathsf{mf}}] \cdot \Pr[\neg Q \mid \neg\mathsf{bad}_{\mathsf{mf}}] \\
&\leq \Pr[Q \mid \neg\mathsf{bad}_{\mathsf{mf}}] + \frac{1}{2}\left(1 - \Pr[Q \mid \neg\mathsf{bad}_{\mathsf{mf}}]\right) \qquad (3) \\
&= \frac{1}{2}\Pr[Q \mid \neg\mathsf{bad}_{\mathsf{mf}}] + \frac{1}{2}\,.
\end{aligned}$$

Then note that

$$\begin{aligned}
\Pr[Q \mid \neg\mathsf{bad}_{\mathsf{mf}}] &\leq \Pr\left[\mathsf{Exp}_{\mathrm{KEM}}^{(n,\kappa)\text{-ow-pca}\star}(\mathbb{B})\right], \\
\implies \Pr[\mathrm{G}_2(\mathbb{A}) \mid \neg\mathsf{bad}_{\mathsf{mf}}] &\leq \frac{1}{2}\Pr\left[\mathsf{Exp}_{\mathrm{KEM}}^{(n,\kappa)\text{-ow-pca}\star}(\mathbb{B})\right] + \frac{1}{2}\,. \qquad (4)
\end{aligned}$$

```
$\mathbb{B}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
$b_1, \ldots, b_\kappa \leftarrow \{0,1\}$
$(\mathtt{I}, \hat{b}) \leftarrow \mathbb{A}^{\mathcal{C},\mathcal{E},\mathcal{D},\mathcal{K},\mathcal{B},\mathcal{F}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
$i_1, \ldots, i_n \leftarrow \mathtt{I}$
for $k \in [n]$ :
  for $c$ s.th. $\mathtt{K}_{i_k}(c) \neq \emptyset$ :
    for $j$ s.th. $c = \mathtt{P}_{i_k}[j]$ :
      $j_{i_k} \leftarrow j, \hat{K}_{i_k} \leftarrow \mathtt{K}_{i_k}(c)$
return $(\mathtt{I}, (j_i, \hat{K}_i)_{i \in \mathtt{I}})$
```

```
If $\mathbb{A}$ calls oracle $\mathcal{C}(i, \Pi, \ell)$
$c \leftarrow \mathcal{E}_{\mathbb{B}}(i), \mathtt{P}_i \xleftarrow{\frown} c$
$\ell' \leftarrow \ell + \ell_{\mathsf{mackey}}$
if $\mathtt{K}_i(c) \neq \emptyset$ :
  $K^{\mathrm{kem}} \leftarrow \mathtt{K}_i(c)$
  $K^{\mathrm{mac}} \| K_0^{\mathrm{xem}} \leftarrow \mathcal{F}(\mathsf{pk}_i, c, K^{\mathrm{kem}}, \ell')$
else :
  while $|\mathtt{D}_i(c)| < \ell'$ :
    $\mathtt{D}_i(c) \xleftarrow{\frown}_{\$} \{0,1\}$
  $K^{\mathrm{mac}} \| K_0^{\mathrm{xem}} \leftarrow \mathtt{D}_i(c)[\![\ell']\!]$
$\mathtt{E}_i \xleftarrow{\frown} \langle c, K^{\mathrm{mac}} \rangle$
$K_1^{\mathrm{xem}} \leftarrow \Pi(K_0^{\mathrm{xem}})$
return $K_{b_i}^{\mathrm{xem}}$
```

```
Oracle $\mathcal{F}(\mathsf{pk}, c, K^{\mathrm{kem}}, \ell)$
if $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}}) = \emptyset$ :
  $\forall_{i \in [\kappa]} \mathsf{pk}_i = \mathsf{pk}$ :
    if $\mathcal{P}_{\mathbb{B}}(i, c, K^{\mathrm{kem}})$ :
      $\mathtt{K}_i(c) \leftarrow K^{\mathrm{kem}}$
      if $\mathtt{D}_i(c) \neq \varepsilon$ :
        $\mathtt{F}(\mathsf{pk}_i, c, K^{\mathrm{kem}}) \leftarrow \mathtt{D}_i(c)$
while $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}}) < \ell$ :
  $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}}) \xleftarrow{\frown}_{\$} \{0,1\}$
return $\mathtt{F}(\mathsf{pk}, c, K^{\mathrm{kem}})[\![\ell]\!]$
```

Fig. 18. The reduction $\mathbb{B}$, simulating G_2 while playing $\mathsf{Exp}_{\mathrm{KEM}}^{(n,\kappa)\text{-ow-pca}\star}$. Changes to the oracles are highlighted in blue; omitted oracles are simulated exactly as in G_2.

Combining eqs. (1)–(4), we get

$$\begin{aligned}
\mathsf{Adv}_{\mathrm{TXEM}}^{(n,\kappa)\text{-rop-cca}\star}(\mathbb{A}) &= 2\Pr[\mathrm{G}_0(\mathbb{A})] - 1 \\
&= 2\big(\Pr[\mathrm{G}_1(\mathbb{A})] - \Pr[\mathrm{G}_2(\mathbb{A}) \mid \neg\mathsf{bad}_{\mathsf{mf}}] + \Pr[\mathrm{G}_2(\mathbb{A}) \mid \neg\mathsf{bad}_{\mathsf{mf}}]\big) - 1 \\
&= 2\big(\Pr[\mathrm{G}_1(\mathbb{A})] - \Pr[\mathrm{G}_2(\mathbb{A}) \mid \neg\mathsf{bad}_{\mathsf{mf}}]\big) + 2\Pr[\mathrm{G}_2(\mathbb{A}) \mid \neg\mathsf{bad}_{\mathsf{mf}}] - 1 \\
&\leq 2\big(q_d \hat{\epsilon}_{k-1} + \epsilon_k(q_c)\big) + 2\left(\frac{1}{2}\Pr\left[\mathsf{Exp}_{\mathrm{KEM}}^{(n,\kappa)\text{-ow-pca}\star}(\mathbb{B})\right] + \frac{1}{2}\right) - 1 \\
&= 2\big(q_d \hat{\epsilon}_{k-1} + \epsilon_k(q_c)\big) + \Pr\left[\mathsf{Exp}_{\mathrm{KEM}}^{(n,\kappa)\text{-ow-pca}\star}(\mathbb{B})\right] \\
&= 2\big(q_d \hat{\epsilon}_{k-1} + \epsilon_k(q_c)\big) + \mathsf{Adv}_{\mathrm{KEM}}^{(n,\kappa)\text{-ow-pca}\star}(\mathbb{B}) \; .
\end{aligned}$$

□

The proof borrows some ideas already used to prove AGK's Thm. 2. In fact, it is relatively straightforward to recast AGK's Thm. 2 as the multi-instance version of a OW-PCA KEM plus a programmable random oracle yielding an IND-CCA KEM, although the presence of the error terms $\hat{\epsilon}_{k-1}$ and especially $\epsilon_k(q_c)$ render recovery of AGK's Thm. 2 as a special case of our Thm. 9 not immediate.

Combining Thm. 8 and 9 in Cor. 4, we can finally conclude that our construction yields a PKE inheriting the multi-instance security of the underlying KEM (for parameter regimes where the loss term does not dominate).

Corollary 4. *Let PKE' be as in Fig. 9, let the underlying TagXEM be as in Fig 13, let KEM be a perfectly correct KEM with unique encapsulations, and let $k \in \mathbb{Z}_{>1}$. Then, there is an SFBB reduction $\mathbb{B}$ such that, for all $\mathbb{A}$ that makes q_c challenge and q_d decryption oracle queries,*

$$\mathsf{Adv}_{\mathrm{PKE}'}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{A}) \leq \mathsf{Adv}_{\mathrm{KEM}}^{(n,\kappa)\text{-ow-pca}\star}(\mathbb{B}) + 2\big(q_d \hat{\epsilon}_{k-1} + \epsilon_k(q_c)\big)$$

in the programmable random oracle model, where $\hat{\epsilon}_{k-1}$ is the forging advantage after observing $k-1$ valid message–tag pairs (Def. 11) and $\epsilon_k(q_c)$ is the k-out-of-q_c encapsulation multi-collision probability of KEM (Def. 9). If $\mathbb{A}$ makes q_f queries to the random oracle, then $\mathbb{B}$ makes at most q_f queries to its plaintext checking oracle.

Remark 4. The resulting construction is remarkably similar to the PKE studied by Heuer et al. [25] in the context of selective opening attacks (and to a lesser extent its predecessor by Steinfeld et al. [39] and successor by Lai et al. [32]). They too use a random oracle to derive a MAC key and a one-time pad from an ephemeral KEM key. The only two differences are that Heuer et al. do not consider arbitrary length messages and that their random oracle outputs $K^{\mathrm{xem}} \| K^{\mathrm{mac}}$, i.e. the opposite order from what we do.

For fixed length messages, the order in which those two keys are output does not matter. However, when moving to arbitrary length-messages, the order of the XOF output does matter. Outputting $K^{\mathrm{xem}} \| K^{\mathrm{mac}}$ instead would allow a length extension attack enabling the adversary to recover the MAC key, at which point producing forgeries would be trivial.

In a way, the construction is quite brittle that these small details matter. Another example of brittleness is that our reduction for Theorem 9 requires $\perp$ produced from a KEM decryption error to be indistinguishable from a failed MAC verification. In implementations, a timing attack might well break this requirement.

Remark 5. The proof of Thm. 9 does rely on perfect correctness of the underlying KEM, thus excluding many popular post-quantum KEMs based on the hardness of LWE. Having said that, establishing the post-quantum security of TXEM would require a proof in the quantum random oracle model [15]. We leave the construction of a post-quantum TagXEM as an enticing open problem.

A Concrete Instantiation. We conclude by providing a concrete bound for the construction when instantiating with low granularity ElGamal KEM on groups of size $\geq p$. ElGamal KEM satisfies perfect correctness and unique encapsulation (ensuring compatibility with Thm. 9) and produces uniformly random group elements as ephemeral keys, so $\epsilon_k(q_c) \leq q^k/p^{k-1}$. Furthermore, the relevant multi-instance OW-PCA security can be linked to the low granularity MI-GapCDH problem with corruptions (Thm. 12). By extending AGK's low granularity bound [4, Thm. 6] to include corruptions (Thm. 11) and combining with Cor. 4, we arrive at a clean information-theoretic bound (Cor. 5) in the generic group and programmable random oracle model. To keep the bound easier to interpret, we assume that the adversary makes at most $\sqrt{p}$ queries to the encryption and decryption oracles; realistically, an adversary will be able to make far more offline queries q to its generic group and for $q \approx \sqrt{p}$ a single discrete logarithm instance can already be broken. In a similar vein, the requirement that each group instance receive at least $\max\{60 \log_2 p, \sqrt{q_f}/2\}$ group operation calls (allowing some simplifications in the MI-GapCDH bound) is a reasonable one, as already argued by AGK, given that the number of group operations performed by an ElGamal adversary is "typically large".

Corollary 5. *Let PKE' be as in Fig. 9, let the underlying TagXEM be as in Fig 13, let KEM be instantiated as low granularity ElGamal (see App. C) and let p be a lower bound on the generated groups. Let $k \in \mathbb{Z}_{>1}$, let MAC be an information-theoretic MAC with key length ℓ_{mackey} and output length ℓ_{mac} and satisfying $\hat{\epsilon}_{k-1} = 2^{-\ell_{\mathsf{mac}}}$. Then, for any information-theoretic $\mathbb{A}$ that makes at most $\sqrt{p}$ challenge oracle queries, at most $\sqrt{p}$ decryption oracle queries, q_f queries to the random oracle, and a total of q queries to the group-operation oracles with at least $\max\{60 \log_2 p, \sqrt{q_f}/2\}$ queries per group instance, it holds that*

$$\mathsf{Adv}_{\mathrm{PKE}'}^{(n,\kappa)\text{-ind-cca}\star}(\mathbb{A}) \leq \left(\frac{4 \cdot e \cdot q^2}{n^2 \cdot p}\right)^n + 2\left(\frac{\sqrt{p}}{2^{\ell_{\mathsf{mac}}}} + \frac{1}{p^{\frac{k}{2}-1}}\right)$$

in the programmable random oracle and generic group model.

For the construction to exhibit meaningful multi-instance security, we want the upper bound on the adversary's advantage to diminish with increasing n. Since the second term on the right hand side of Cor. 5 is independent of n, the first term has to dominate for advantages of interest. Thus, for a fixed p, we want to set ℓ_{mac} and k so that, irrespective of n, we do not really care about the other two terms, where ℓ_{mac} directly corresponds to the PKE's ciphertext expansion and increasing k will require longer ephemeral keys as output by the XOF to ensure that $\ell_{\mathsf{mackey}} \geq k \cdot \ell_{\mathsf{mac}}$. To minimize overhead, having both terms equal is optimal, corresponding to $2\ell_{\mathsf{mac}} = (k-1)\log_2 p$. Some reasonable options are then $(\ell_{\mathsf{mac}}, k) = (\log_2 p, 3)$ or $(\ell_{\mathsf{mac}}, k) = (3/2 \log_2 p, 4)$.

Alternatively, the bound can be interpreted in terms of the scaling factor, which focuses on the minimum resources needed to achieve an overwhelming advantage (see App. C for details). In that case, the second term, being independent of n, is manifestly of little interest for either of our suggested parameter choices.

Acknowledgement

The authors would like to thank Joseph Jaeger for identifying a bug in our definition of xor-indistinguishability in the published version of the article [17].

References

1. Abdalla, M., Benhamouda, F., Pointcheval, D.: Public-key encryption indistinguishable under plaintext-checkable attacks. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 332–352. Springer, Heidelberg (Mar / Apr 2015). https://doi.org/10.1007/978-3-662-46447-2_15
2. Abe, M., Gennaro, R., Kurosawa, K.: Tag-KEM/DEM: A new framework for hybrid encryption. Journal of Cryptology **21**(1), 97–130 (Jan 2008). https://doi.org/10.1007/s00145-007-9010-x
3. Auerbach, B., Giacon, F., Kiltz, E.: Everybody's a target: Scalability in public-key encryption. Cryptology ePrint Archive, Report 2019/364 (2019), https://eprint.iacr.org/2019/364
4. Auerbach, B., Giacon, F., Kiltz, E.: Everybody's a target: Scalability in public-key encryption. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020, Part III. LNCS, vol. 12107, pp. 475–506. Springer, Heidelberg (May 2020). https://doi.org/10.1007/978-3-030-45727-3_16
5. Bader, C., Jager, T., Li, Y., Schäge, S.: On the impossibility of tight cryptographic reductions. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 273–304. Springer, Heidelberg (May 2016). https://doi.org/10.1007/978-3-662-49896-5_10
6. Baecher, P., Brzuska, C., Fischlin, M.: Notions of black-box reductions, revisited. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013, Part I. LNCS, vol. 8269, pp. 296–315. Springer, Heidelberg (Dec 2013). https://doi.org/10.1007/978-3-642-42033-7_16
7. Bellare, M., Boldyreva, A., Micali, S.: Public-key encryption in a multi-user setting: Security proofs and improvements. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 259–274. Springer, Heidelberg (May 2000). https://doi.org/10.1007/3-540-45539-6_18
8. Bellare, M., Desai, A., Pointcheval, D., Rogaway, P.: Relations among notions of security for public-key encryption schemes. In: Krawczyk, H. (ed.) CRYPTO'98. LNCS, vol. 1462, pp. 26–45. Springer, Heidelberg (Aug 1998). https://doi.org/10.1007/BFb0055718
9. Bellare, M., Hofheinz, D., Kiltz, E.: Subtleties in the definition of IND-CCA: When and how should challenge decryption be disallowed? Journal of Cryptology **28**(1), 29–48 (Jan 2015). https://doi.org/10.1007/s00145-013-9167-4
10. Bellare, M., Kohno, T.: A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and applications. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 491–506. Springer, Heidelberg (May 2003). https://doi.org/10.1007/3-540-39200-9_31
11. Bellare, M., Paterson, K.G., Rogaway, P.: Security of symmetric encryption against mass surveillance. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 1–19. Springer, Heidelberg (Aug 2014). https://doi.org/10.1007/978-3-662-44371-2_1
12. Bellare, M., Ristenpart, T., Tessaro, S.: Multi-instance security and its application to password-based cryptography. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 312–329. Springer, Heidelberg (Aug 2012). https://doi.org/10.1007/978-3-642-32009-5_19
13. Bellare, M., Ristenpart, T., Tessaro, S.: Multi-instance security and its application to password-based cryptography. Cryptology ePrint Archive, Report 2012/196 (2012), https://eprint.iacr.org/2012/196
14. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Denning, D.E., Pyle, R., Ganesan, R., Sandhu, R.S., Ashby, V. (eds.) ACM CCS 93. pp. 62–73. ACM Press (Nov 1993). https://doi.org/10.1145/168588.168596
15. Boneh, D., Dagdelen, Ö., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 41–69. Springer, Heidelberg (Dec 2011). https://doi.org/10.1007/978-3-642-25385-0_3
16. Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Schwabe, P., Seiler, G., Stehle, D.: Crystals - kyber: A cca-secure module-lattice-based kem. In: 2018 IEEE European Symposium on Security and Privacy (EuroS P). pp. 353–367 (2018). https://doi.org/10.1109/EuroSP.2018.00032
17. Brunetta, C., Heum, H., Stam, M.: Multi-instance secure public-key encryption (2023), to appear at PKC'23, also available as https://eprint.iacr.org/archive/2022/909/20230404:114420
18. Cramer, R., Shoup, V.: A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: Krawczyk, H. (ed.) CRYPTO'98. LNCS, vol. 1462, pp. 13–25. Springer, Heidelberg (Aug 1998). https://doi.org/10.1007/BFb0055717
19. Cramer, R., Shoup, V.: Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. SIAM Journal on Computing **33**(1), 167–226 (2003)
20. Farshim, P., Tessaro, S.: Password hashing and preprocessing. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part II. LNCS, vol. 12697, pp. 64–91. Springer, Heidelberg (Oct 2021). https://doi.org/10.1007/978-3-030-77886-6_3
21. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. In: Wiener, M.J. (ed.) CRYPTO'99. LNCS, vol. 1666, pp. 537–554. Springer, Heidelberg (Aug 1999). https://doi.org/10.1007/3-540-48405-1_34
22. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. Journal of Cryptology **26**(1), 80–101 (Jan 2013). https://doi.org/10.1007/s00145-011-9114-1
23. Giacon, F., Kiltz, E., Poettering, B.: Hybrid encryption in a multi-user setting, revisited. In: Abdalla, M., Dahab, R. (eds.) PKC 2018, Part I. LNCS, vol. 10769, pp. 159–189. Springer, Heidelberg (Mar 2018). https://doi.org/10.1007/978-3-319-76578-5_6

24. Halevi, S., Krawczyk, H.: Security under key-dependent inputs. In: Ning, P., De Capitani di Vimercati, S., Syverson, P.F. (eds.) ACM CCS 2007. pp. 466–475. ACM Press (Oct 2007). `https://doi.org/10.1145/1315245.1315303`
25. Heuer, F., Jager, T., Kiltz, E., Schäge, S.: On the selective opening security of practical public-key encryption schemes. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 27–51. Springer, Heidelberg (Mar / Apr 2015). `https://doi.org/10.1007/978-3-662-46447-2_2`
26. Heum, H., Stam, M.: Tightness subtleties for multi-user pke notions. In: Paterson, M.B. (ed.) Cryptography and Coding. pp. 75–104. Springer International Publishing, Cham (2021). `https://doi.org/10.1007/978-3-030-92641-0_5`
27. Hoeffding, W.: Probability inequalities for sums of bounded random variables. Journal of the American Statistical Association **58**, 13–30 (1963)
28. Hofheinz, D., Hövelmanns, K., Kiltz, E.: A modular analysis of the Fujisaki-Okamoto transformation. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 341–371. Springer, Heidelberg (Nov 2017). `https://doi.org/10.1007/978-3-319-70500-2_12`
29. Jager, T., Kiltz, E., Riepel, D., Schäge, S.: Tightly-secure authenticated key exchange, revisited. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part I. LNCS, vol. 12696, pp. 117–146. Springer, Heidelberg (Oct 2021). `https://doi.org/10.1007/978-3-030-77870-5_5`
30. Kiltz, E., Pan, J., Riepel, D., Ringerud, M.: Multi-user CDH problems and the concrete security of NAXOS and HMQV. In: Rosulek, M. (ed.) CT-RSA 2023 (to appear). Springer, Heidelberg (2023), available as `https://eprint.iacr.org/2023/115`.
31. Kurosawa, K., Desmedt, Y.: A new paradigm of hybrid encryption scheme. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 426–442. Springer, Heidelberg (Aug 2004). `https://doi.org/10.1007/978-3-540-28628-8_26`
32. Lai, J., Yang, R., Huang, Z., Weng, J.: Simulation-based bi-selective opening security for public key encryption. In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021, Part II. LNCS, vol. 13091, pp. 456–482. Springer, Heidelberg (Dec 2021). `https://doi.org/10.1007/978-3-030-92075-3_16`
33. Lee, Y., Lee, D.H., Park, J.H.: Tightly cca-secure encryption scheme in a multi-user setting with corruptions. Des. Codes Cryptogr. **88**(11), 2433–2452 (2020)
34. Lewko, A.B., Waters, B.: Why proving HIBE systems secure is difficult. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 58–76. Springer, Heidelberg (May 2014). `https://doi.org/10.1007/978-3-642-55220-5_4`
35. NIST: SHA-3 standard: Permutation-based hash and extendable-output functions. Federal Information Processing Standards Publication 202, NIST (Aug 2015)
36. Okamoto, T., Pointcheval, D.: REACT: Rapid Enhanced-security Asymmetric Cryptosystem Transform. In: Naccache, D. (ed.) CT-RSA 2001. LNCS, vol. 2020, pp. 159–175. Springer, Heidelberg (Apr 2001). `https://doi.org/10.1007/3-540-45353-9_13`
37. Preneel, B.: Analysis and Design of Cryptographic Hash Functions. Ph.D. thesis, KU Leuven (Feb 1993)
38. Reingold, O., Trevisan, L., Vadhan, S.P.: Notions of reducibility between cryptographic primitives. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 1–20. Springer, Heidelberg (Feb 2004). `https://doi.org/10.1007/978-3-540-24638-1_1`
39. Steinfeld, R., Baek, J., Zheng, Y.: On the necessity of strong assumptions for the security of a class of asymmetric encryption schemes. In: Batten, L.M., Seberry, J. (eds.) ACISP 02. LNCS, vol. 2384, pp. 241–256. Springer, Heidelberg (Jul 2002). `https://doi.org/10.1007/3-540-45450-0_20`
40. Wegman, M.N., Carter, L.: New hash functions and their use in authentication and set equality. Journal of Computer and System Sciences **22**, 265–279 (1981)

A Deferred Proofs

A.1 Proof of Theorem 1

Theorem 1 (MKU $\to$ UKU). *Let $0 < n \leq \kappa$ be integer parameters and let PKE be a (γ, δ)-correct encryption scheme. Then, there is a type-preserving SFBB reduction $\mathbb{B}_{\mathrm{mku}}$, such that for every adversary $\mathbb{A}_{\mathrm{uku}}$,*

$$\mathsf{Adv}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}_{\mathrm{uku}}) \leq \mathsf{Adv}^{(n,\kappa)\text{-mku-cca}\star}_{\mathrm{PKE}}(\mathbb{B}_{\mathrm{mku}}) + \kappa\gamma\,.$$

Proof. $\mathbb{B}_{\mathrm{mku}}$ forwards all the experiment's messages until it receives from $\mathbb{A}_{\mathrm{uku}}$ the output $(\mathtt{I}, \vec{\hat{\mathsf{sk}}})$. Observe that these secret keys allow $\mathbb{A}_{\mathrm{uku}}$ to win if they were generated in the game setup, $(\mathsf{pk}_i, \hat{\mathsf{sk}}_i) \leftarrow\$\ \mathsf{PKE.Kg}$, while in order for $\mathbb{B}_{\mathrm{mku}}$ to win, the secret keys must pass the key validation algorithm, $\mathsf{PKE.Check}(\mathsf{pm}, \mathsf{pk}_i, \hat{\mathsf{sk}}_i) = \mathsf{true}$. Briefly, $\mathbb{B}_{\mathrm{mku}}$ wins if, at least, $\mathbb{A}_{\mathrm{uku}}$ wins and the secret keys pass the validation algorithm. Formally,

$$\begin{aligned}
&\Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-mku-cca}\star}_{\mathrm{PKE}}(\mathbb{B}_{\mathrm{mku}}) = 1\right] \\
&\quad \geq \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = 1 \wedge \bigwedge_{i\in\mathtt{I}} \mathsf{PKE.Check}(\mathsf{pm}, \mathsf{pk}_i, \hat{\mathsf{sk}}_i) = \mathsf{true}\right]
\end{aligned}$$

$$= \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = 1 \wedge \bigwedge_{i \in \mathtt{I}} \mathsf{PKE.Check}(\mathsf{pm}, \mathsf{pk}_i, \mathsf{sk}_i) = \mathsf{true}\right]$$

where, since $\mathbb{A}_{\mathrm{uku}}$ wins, it holds that $\hat{\mathsf{sk}}_i = \mathsf{sk}_i$;

$$\begin{aligned}
&\geq \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = 1 \wedge \bigwedge_{i \in [\kappa]} \mathsf{PKE.Check}(\mathsf{pm}, \mathsf{pk}_i, \mathsf{sk}_i) = \mathsf{true}\right] \\
&= \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = 1\right] \\
&\quad - \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = 1 \wedge \bigvee_{i \in [\kappa]} \mathsf{PKE.Check}(\mathsf{pm}, \mathsf{pk}_i, \mathsf{sk}_i) = \mathsf{false}\right] \\
&\geq \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = 1\right] - \Pr\left[\bigvee_{i \in [\kappa]} \mathsf{PKE.Check}(\mathsf{pm}, \mathsf{pk}_i, \mathsf{sk}_i) = \mathsf{false}\right] \\
&= \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = 1\right] - \sum_{i \in [\kappa]} \Pr[\mathsf{PKE.Check}(\mathsf{pm}, \mathsf{pk}_i, \mathsf{sk}_i) = \mathsf{false}] \\
&= \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-uku-cca}\star}_{\mathrm{PKE}}(\mathbb{A}) = 1\right] - \kappa\gamma\,.
\end{aligned}$$

Applying Def. 2 then yields the stated bound. □

B The Relationship between ROP and IND Security for KEMs

The following lemma shows that ROP is at least as strong as IND, in the sense that ROP tightly implies IND.

Lemma 4 (ROP $\implies$ IND)**.** *There is a type-preserving SFBB reduction $\mathbb{B}$ such that, for any adversary $\mathbb{B}$,*

$$\mathsf{Adv}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{TXEM}}(\mathbb{A}) \leq \mathsf{Adv}^{(n,\kappa)\text{-rop-cca}\star}_{\mathrm{TXEM}}(\mathbb{B})\,.$$

If $\mathbb{A}$ calls its challenge oracle q times, then $\mathbb{B}$ draws q keys uniformly at random.

Proof. Whenever $\mathbb{A}$ calls $\mathcal{C}(i, \ell)$, let $\mathbb{B}$ define the permutation Π that is simply the xor of the input with a freshly sampled key, and then call $\mathcal{C}_{\mathrm{ROP}}(i, \ell, \Pi)$. The resulting K_1 will look like a uniformly random key, with no relation to the encapsulated key K_0. □

In the other direction, a loss of $\binom{\kappa}{n} \cdot 2^n$ is inferred, as can be seen by adapting the proof techniques that led to Cor. 2.

Theorem 10 (IND $\implies$ ROP)**.** *There is an SFBB reduction $\mathbb{B}$ such that, for any adversary $\mathbb{B}$,*

$$\mathsf{Adv}^{(n,\kappa)\text{-rop-cca}\star}_{\mathrm{TXEM}}(\mathbb{A}) \leq \binom{\kappa}{n} \cdot 2^n \cdot \mathsf{Adv}^{(n,\kappa)\text{-ind-cca}\star}_{\mathrm{TXEM}}(\mathbb{B})\,.$$

$\mathbb{B}$'s overhead consists of generating $\kappa - n$ fresh keypairs, sampling κ bits, and a choosing a subset of $[\kappa]$ of cardinality n uniformly at random.

C On the MI-GapCDH Problem with Corruptions

In this section we study the hardness of the MI-GapCDH problem, as introduced by AGK, when enhanced with corruptions. AGK considered three settings: the (realistic) high granularity setting, with one global (standardized) group and group generator; mid granularity, where the group remains global but generators are independently sampled per user; and low granularity, where the groups themselves are independently sampled among users. AGK went on to provide useful bounds linking the hardness of each to standard assumptions (in the algebraic and generic group models).

```
Experiment $\mathsf{Exp}^{(n,\kappa)\text{-gapcdh}\star}_{\text{low-gran}}(\mathbb{A})$
for $i \in [\kappa]$ do
    $(\mathbb{G}_i, g_i, p_i) \leftarrow\!\!\$\ \mathsf{GGen}$
    $\mathtt{G}[i] \leftarrow (\mathbb{G}_i, g_i, p_i)$
    $\mathtt{x}[i] \leftarrow\!\!\$\ \mathbb{Z}_{p_i}$ ;   $\mathtt{X}[i] \leftarrow g_i^{\mathtt{x}[i]}$
    $\mathtt{y}[i] \leftarrow\!\!\$\ \mathbb{Z}_{p_i}$ ;   $\mathtt{Y}[i] \leftarrow g_i^{\mathtt{y}[i]}$
    $\mathtt{Z}[i] \leftarrow g_i^{\mathtt{x}[i]\cdot\mathtt{y}[i]}$
$(\mathtt{I}, \hat{\mathtt{Z}}) \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{DDH},\mathcal{K}}(\mathtt{G}, \mathtt{X}, \mathtt{Y})$
if $|\mathtt{I}| \neq n$ then return $0$
if $\mathtt{I} \cap \mathtt{K} \neq \emptyset$ then return $0$
return $\bigwedge_{i \in \mathtt{I}} \mathtt{Z}[i] = \hat{\mathtt{Z}}[i]$
```

```
Oracle $\mathcal{DDH}(i, X, Y, Z)$
parse $(X, Y) \to (g_i^x, g_i^y)$
return $g_i^{x \cdot y} = Z$

Oracle $\mathcal{K}(i)$
$\mathtt{K} \xleftarrow{\cup} i$
return $\mathtt{x}[i]$
```

Fig. 19. Multi-instance gap CDH with corruption for the low granularity case, i.e. each challenge is sampled for a completely independent group and generator. Highlighted in blue, the differences introduced concerning the uncorrupted version of Auerbach et al. [4].

Restricting ourselves for now to low granularity MI-GapCDH in the generic group model, we can directly adapt AGK's bound [4, Thm. 6]. Denote by $\mathsf{Exp}^{(n,\kappa)\text{-gapcdh}\star}_{\text{low-gran}}$ the low granularity n-out-of-κ MI-GapCDH experiment with corruptions (Fig. 19); the advantage of an adversary $\mathbb{A}$ against the experiment is $\mathsf{Adv}^{(n,\kappa)\text{-gapcdh}\star}_{\text{low-gran}}(\mathbb{A}) := \Pr\left[\mathsf{Exp}^{(n,\kappa)\text{-gapcdh}\star}_{\text{low-gran}}(\mathbb{A}) = 1\right]$.

Compared to AGK, we make one more change to the experiment by defining eligible lists $\mathtt{I}$, in addition to containing no compromised indices, to be of size exactly n rather than greater-or-equal n; this happens without loss of generality for computational problems (see also Sec. 3.3).

Theorem 11. *Let* GGen *be a group-generating algorithm generating generic groups of size at least* p. *Let* n, κ, q, q_D *and* q_i *for* $i \in [\kappa]$ *such that* $1 \leq n \leq \kappa$, $q = \sum_{i=1}^{n} q_i$, *and* q_i *are such that* $q_i \geq 60 \log_2 p$ *and* $4q_i^2 \geq q_D$. *Then, for any adversary* $\mathbb{A}$ *making at most* q_i *queries to the* i*-th group-operation oracle and* q_D *queries to the gap oracle,*

$$\mathsf{Adv}^{(n,\kappa)\text{-gapcdh}\star}_{\text{low-gran}}(\mathbb{A}) \leq \left(\frac{4 \cdot e \cdot q^2}{n^2 \cdot p}\right)^n .$$

Proof (sketch). The proof is identical to AGK's Theorem 6 up to the additional observation that, since instances are completely independent, knowing the solution to one cannot help in solving others. This does not alter any of the steps of their proof: as AGK point out, all the generated groups are independent and, to win, $\mathbb{A}$ must output an eligible list $\mathtt{I}$ of indices and a corresponding list $\hat{\mathtt{Z}}$ of solutions; all we have done is alter what constitutes an eligible list $\mathtt{I}$. □

AGK showed that low granularity schemes achieve optimal scaling, as apparent in the exponential nature of the above bound. Thus, Theorem 11 allows the multi-instance security of our constructions to lean on an explicit bound with optimal scaling; all that remains is to observe that for ElGamal KEM, its multi-instance OW-PCA security is easily achieved from the low granularity MI-GapCDH assumption. Writing X for g^x and Y for g^y, let KEM be such that $\mathsf{KEM.Kg}$ samples a group $(\mathbb{G}, g, p) \leftarrow\!\!\$\ \mathsf{GGen}$, private exponent $x \leftarrow\!\!\$\ \mathbb{Z}_p$ and sets $(\mathsf{pk}, \mathsf{sk}) \leftarrow (((\mathbb{G}, g, p), X), x)$, $\mathsf{KEM.Enc}_{\mathsf{pk}}$ samples $y \leftarrow\!\!\$\ \mathbb{Z}_p$ and sets $(K, c) \leftarrow (X^y, Y)$, and $\mathsf{KEM.Dec}_{\mathsf{sk}}$, on input $c = Y$, returns $K \leftarrow Y^x$ (where we assume $\mathsf{KEM.Dec}$ also implicitly has access to the group description).

Theorem 12. *Let* KEM *be as above. Then there is an SFBB reduction* $\mathbb{B}$ *such that, for all* $\mathbb{A}$,

$$\mathsf{Adv}^{(n,\kappa)\text{-ow-pca}\star}_{\text{KEM}}(\mathbb{A}) \leq \mathsf{Adv}^{(n,\kappa)\text{-gapcdh}\star}_{\text{low-gran}}(\mathbb{B}) .$$

If $\mathbb{A}$ *calls its challenge oracle* q *times, then* $\mathbb{B}$ *draws* q *uniformly random group elements.*

Proof (sketch). The reduction matches public keys to the received $\mathtt{X}[i]$ and forwards them to $\mathbb{A}$. Given that the multi-instance OW-PCA game (Fig. 14) is multi-challenge, the reduction relies on random self-reducibility to provide multiple challenges per instance: for each challenge oracle call to instance i, $\mathbb{B}$ samples $w \leftarrow\!\!\$\ \mathbb{Z}_{p_i}$ and sets $c \leftarrow \mathtt{Y}[i] \cdot g_i^w$, saving w in list $\mathtt{W}_i$ for later. Calls to the oracle $\mathcal{P}$ are forwarded to oracle $\mathcal{DDH}$, and likewise for corruption calls.

Once $\mathbb{A}$ halts with output $(\mathtt{I}, (j_i, \hat{K}_i)_{i\in\mathtt{I}})$, for each $i \in \mathtt{I}$, the reduction retrieves $w \leftarrow \mathtt{W}_i[j_i]$ and sets $\hat{\mathtt{Z}}[i] \leftarrow \hat{K}_i/\mathtt{X}[i]^w$, finally halting with output $\hat{\mathtt{Z}}$. If $\mathbb{A}$ returned correct ephemeral keys, then for each i we have $\hat{K}_i = c^x = g_i^{xy} g_i^{xw}$, so that dividing by $\mathtt{X}[i]^w = g_i^{xw}$ produces $\hat{\mathtt{Z}}[i] = g_i^{xy}$, as desired.

As the group is cyclic and g_i^w are independently sampled, uniformly random group elements, the $\mathtt{Y}[i] \cdot g_i^w$ are independently uniform too; thus, the simulation is faithful, and $\mathbb{B}$ wins iff $\mathbb{A}$ wins. □

For high and mid granularity, an immediate adaptation of AGK's bounds to include corruptions does not seem possible. Take their Theorem 5, relating the (uncorrupted) high granularity n-out-of-κ MI-GapCDH to a (n, n) multi-instance gap discrete logarithm problem, for which they give a useful bound. The proof constructs a reduction $\mathbb{B}$ that obtains n discrete logarithm challenges Z and injects them into the κ CDH challenges, which are then given to the adversary. This injection is done in such a way that possession of any n-sized subset of solutions to the κ challenges allows for the reconstruction of the original n DL solutions.

This strategy breaks down in the presence of corruptions, as answering a corrupting query on any GapCDH instance would necessitate to compromise one of the DL challenges the reduction seeks to break. (It is possible that this could be patched by having the adversary guess beforehand the instances that will be corrupted, although this would likely lead to a looser bound.) We therefore leave the hardness of mid and high granularity MI-GapCDH as interesting open problems, with high granularity being of particular relevance, as it most closely matches typical deployment practice.

D On the Scaling Factor

As argued in the introduction, we want a guarantee that breaking n PKE instances can not be done much more efficiently than breaking each independently. Ideally, then, the advantages should experience an exponential dampening with increasing n, as achieved by the KEM of Sect. C; our main results can then be interpreted to say that for any given n, our constructions inherits this smallness of the underlying KEM.

Another way to capture this intuition is to say that any adversary should have to spend n times as much computing time to break n instances, than that needed to break 1 system (for some suitable notion of "breaking"). This intuition was formalized asymptotically by AGK as the Scaling Factor (SF). Informally:

$$\mathsf{SF}_{\mathrm{PKE}}^{(n,\kappa)} = \frac{\text{resources necessary to break } n \text{ out of } \kappa \text{ instances}}{\text{resources necessary to break 1 out of 1 instance}} .$$

Assume that $\mathsf{SF}_{\mathrm{KEM}}^{(n,\kappa)} \geq \eta$ for some $\eta \geq 1$; another way to interpret our results would then be to show that it follows that $\mathsf{SF}_{\mathrm{PKE}}^{(n,\kappa)} \geq \eta$ for each of our constructions. As our results are formulated in the concrete-security setting, we do not give a formal definition of the scaling factor and a proof of inheritance, although we sketch the argument below.

In order to conclude that a construction inherits the scaling factor of the underlying KEM, we would have to show two things. Firstly, that

$$\begin{aligned}&\text{resources necessary to break } n \text{ out of } \kappa \text{ PKE instances}\\ &\qquad\qquad \geq \text{resources necessary to break } n \text{ out of } \kappa \text{ KEM instances};\end{aligned}$$

this follows (with a small loss) for our main construction from Corollary 4. Secondly, that

$$\begin{aligned}&\text{resources necessary to break 1 KEM instance}\\ &\qquad\qquad \geq \text{resources necessary to break 1 out of 1 PKE instance}.\end{aligned}$$

Then, it follows from

$$\begin{aligned}&\text{resources necessary to break } n \text{ out of } \kappa \text{ KEM instances}\\ &\qquad\qquad \geq \eta\,(\text{resources necessary to break 1 out of 1 KEM instance})\end{aligned}$$

that

$$\begin{aligned}&\text{resources necessary to break } n \text{ out of } \kappa \text{ PKE instances}\\ &\qquad\qquad \geq \eta\,(\text{resources necessary to break 1 out of 1 PKE instance}) \,,\end{aligned}$$

which, when dividing both sides by the single-instance resources (and omitting negligible terms) yields $\mathsf{SF}^{(n,\kappa)}_{\mathrm{PKE}} \geq \eta$.

However, the second point is somewhat counterintuitive as it seemingly requires showing a "reverse reduction", namely that a break against the KEM results in a break against the PKE. In other words, if the PKE inherits the multi-instance security of the KEM but itself is already harder to break than the KEM, the scaling factor might be reduced. Another subtlety is that the security notion used for the KEM might also cause unexpected complications.

Specifically for our constructions and proofs, the reverse reduction for Theorem 6 is straightforward. On the other hand, the reverse reduction for Theorem 9 already takes a bit more work: a reduction could simply forward the first part of a challenge ciphertext to the KEM adversary and use its decryption oracle to simulate the KEM's plaintext checking oracle.

In other words, we are confident that for our main construction, the scaling factor is essentially preserved from KEM to PKE. Nonetheless, as the scaling factor remains underdeveloped both in the concrete security setting and in the context of primitive-to-construction inheritance, we refrain from providing a full formal definition and analysis of the scaling factor and instead opt for a more classical interpretation of the advantages we achieve (Corr. 5).

Article III

SoK: Public Key Encryption with Openings

Carlo Brunetta, Hans Heum and Martijn Stam

In Cryptology ePrint Archive, Report 2023/1337, 2023. https://eprint.iacr.org/2023/1337

The candidate served as lead author of the work.

SoK: Public Key Encryption with Openings

Carlo Brunetta[1], Hans Heum[2], and Martijn Stam[1]

[1] Simula UiB, Bergen, Norway.
carlob,martijn@simula.no
[2] NTNU - Norwegian University of Science and Technology, Trondheim, Norway.
hans.heum@ntnu.no[⋆]

Abstract. When modelling how public key encryption can enable secure communication, we should acknowledge that secret information, such as private keys or the randomness used for encryption, could become compromised. Intuitively, one would expect unrelated communication to remain secure, yet formalizing this intuition has proven challenging. Several security notions have appeared that aim to capture said scenario, ranging from the multi-user setting with corruptions, via selective opening attacks (SOA), to non-committing encryption (NCE). Remarkably, how the different approaches compare has not yet been systematically explored.

We provide a novel framework that maps each approach to an underlying philosophy of confidentiality: indistinguishability versus simulatability based, each with an a priori versus an a posteriori variant, leading to four distinct philosophies. In the absence of corruptions, these notions are largely equivalent; yet, in the presence of corruptions, they fall into a hierarchy of relative strengths, from IND-CPA and IND-CCA at the bottom, via indistinguishability SOA and simulatability SOA, to NCE at the top.

We provide a concrete treatment for the four notions, discuss subtleties in their definitions and asymptotic interpretations and identify limitations of each. Furthermore, we re-cast the main implications of the hierarchy in a concrete security framework, summarize and contextualize other known relations, identify open problems, and close a few gaps.

We end on a survey of constructions known to achieve the various notions. We identify and name a generic random-oracle construction that has appeared in various guises to prove security in seemingly different contexts. It hails back to Bellare and Rogaway's seminal work on random oracles (CCS'93) and, as previously shown, suffices to meet one of the strongest notions of our hierarchy (single-user NCE with bi-openings).

Keywords: Selective Opening Attacks · Multi-User Security · Non-Committing Encryption · Corruptions

[⋆] Work by Hans Heum partially performed as part of his PhD studies at Simula UiB.

Table of Contents

1 Introduction . . . 3
2 Preliminaries . . . 6
2.1 Notation . . . 6
2.2 PKE Syntax . . . 7
2.3 Security Notions . . . 7
3 Confidentiality with Openings . . . 9
3.1 Four Kinds of Opening . . . 9
3.2 Four Philosophies of Confidentiality . . . 10
3.3 A Priori Indistinguishability with Selective Openings (IND) . . . 12
3.4 A Posteriori Indistinguishability with Selective Opening (ISO) . . . 17
3.5 A Posteriori Simulatability with Selective Opening (SSO) . . . 22
3.6 A Priori Simulatability with Selective Opening (NCE) . . . 27
4 Relations . . . 31
4.1 Hybridization . . . 32
4.2 Implications . . . 33
4.3 Separations . . . 35
4.4 Further Relations . . . 36
4.5 Selected Related Notions . . . 37
5 Constructions . . . 39
5.1 Bellare–Rogaway Encryption . . . 39
5.2 Other Constructions . . . 40
References . . . 45
A ISO Tightly Implies IND with Bi-openings . . . 50

1 Introduction

A group of crypto friends want to exchange their latest ideas with each other. Obviously, they want to do so confidentially and, being slightly old-fashioned, imagine they use public key encryption to secure their communication. Yet, an adversary, mistakenly reckoning our friends are all about the other crypto, sets out to break into the devices used by some of the cryptographers, recovering a number of private keys in the hope of a big score. Ideally, the communication that wasn't intended for the compromised bunch remains secure, so our somewhat disillusioned adversary cannot scoop their research ideas.

Scenarios similar to the one above, involving public key encryption (PKE) with multiple, say κ, users some of whom may be corrupted, have been used to motivate a range of different security notions for PKE above and beyond the now classical left-or-right indistinguishability under chosen ciphertext attacks (IND-CCA). These novel notions range from multi-user indistinguishability with corruptions, via various flavours of selective opening attacks (SOA), to (non-interactive) non-committing encryption (NCE). Given the plethora of theoretical notions based on very similar, practical motivations, the question arises what are the pros and cons of the various notions, and whether some notions should be preferred over others.

In addition to leaking keys, similar notions have also studied security when the randomness used for encryption leaks, or when both keys and randomness leak. Thus we end up with a host of notions, and a sizeable literature exploring how they relate and how each may be achieved. Navigating this literature can be a daunting task for the uninitiated, particularly given diverging formalisms, subtle variations in security definitions (which may or may not turn out consequential to any particular result), and even contradicting claims.

This is the situation that the present Systematization of Knowledge (SoK) aims to remedy. Our goal is *not* to provide a complete classification of all possible variations, but rather to provide a roadmap of the high level choices, and highlight possible pitfalls when designing notions of security with openings. Our generalized definitions, given in Sect. 3.3–Sect. 3.6, may serve as inspiration, yet we stress that for applications, utility of the definitions should be the guiding principle.

Our systematization identifies four philosophies underlying the notions of confidentiality of messages, using two orthogonal considerations: whether a notion is based on indistinguishability or simulation, and whether it is an a priori or a posteriori variant. Each notion then falls into one of the four categories: *a priori indistinguishability* for left-or-right indistinguishability-based (multi-user) notions (IND); *a posteriori indistinguishability* for indistinguishability SOA (ISO); *a posteriori simulatability* for simulation SOA (SSO); and a *priori simulatability* for non-committing encryption (NCE). Each notion furthermore comes in four variants depending on whether only keys leak (receiver opening, denoted $\star$), only messages and randomness leak (sender opening, denoted $\odot$), just messages (transmission opening, denoted $\diamond$), or all of the above (bi-opening, denoted $\circledast$); the exception is NCE, for which only sender, receiver, and bi-opening is defined. Finally, each notion comes in a CPA and a CCA variant, making for 30 notions of security in total—some of which are studied here for the first time. An overview of our notation is given in Table 1.

To begin with, we observe that the four philosophies of confidentiality, while polynomially equivalent sans openings, seem to fall into a strict hierarchy of strength whenever receiver and/or sender openings are accounted for, see Fig. 1. Transmission openings on the other hand are significantly weaker, and all but one of the main notions are known to be polynomially equivalent to IND-CPA/IND-CCA when only transmission openings are included (with the remaining equivalence being conjectured, see Open Problem 11).

Findings. We give generalized definitions of the four main philosophies that all include multiple users, challenges, full adaptivity, and bi-openings; message samplers and simulators are all stateful, which simplifies (and in the case of message samplers, strengthens) the formalizations. In addition, we provide novel a priori indistinguishability notions of security in the presence of transmission, sender and bi-openings (see Def. 3), which are notable for being equivalent to IND-CPA resp. IND-CCA, at a loss for sender openings (Thm. 1) and bi-openings (Thm. 3), but almost completely tightly in the case of transmission openings (Thm. 2).

While ideally we would give definite formalizations of each of the notions considered herein, there are a number of subtle definitional choices available with often no clear best one (for all contexts). We have leaned towards generality wherever possible, but some choices are less clear in terms of benefits and generality. For example, which inputs to a distinguisher should be provided by the game and which may be simulated (for SSO and NCE notions); specifically, whether the simulator may generate the

Table 1. Overview of the different formalizations of capturing confidentiality, the auxiliary adversarial power, and the adversarial opening prowess, together with their relevant associated oracle(s): here, $\mathcal{E}$ are (challenge) encryption oracles, $\mathcal{C}$ is a challenge oracle, $\mathcal{D}$ is the decryption oracle, and $\mathcal{T}$, $\mathcal{S}$, and $\mathcal{R}$ are the transmission, sender, and receiver opening oracles, respectively (see Sect. 3.1).

Shorthand	Associated oracle(s)	Name
κ		number of users (receivers)
β		number of challenge bits
IND	$\mathcal{E}$	indistinguishability
ISO	$\mathcal{E}$ $\mathcal{C}$	indistinguishability-based selective opening
SSO	$\mathcal{E}$	simulation-based selective opening
NCE	$\mathcal{E}$	non-committing encryption
CPA		chosen plaintext attack
CCA	$\mathcal{D}$	chosen ciphertext attack
$\diamond$	$\mathcal{T}$	transmission opening
$\odot$	$\mathcal{S}$	sender opening
$\star$	$\mathcal{R}$	receiver opening
$\circledast$	$\mathcal{S}$ $\mathcal{R}$	bi-opening (sender + receiver)

parameters and public keys itself. We opted for a notion where the simulator generates the public keys but not the parameters, see Def. 6.

As another contribution, we provide a concrete-security treatment of topics that have traditionally been studied asymptotically, recasting several known implications in a concrete light, with asymptotic interpretations that connect to the literature (see Thm. 4–Thm. 6).

Our systematization allowed us to uncover a number of open problems, which we highlight in Open Problem 1–Open Problem 21. Of particular interest are the many relations known for the CPA setting that remain open in the CCA setting (see Fig. 16). Surprisingly, it is unclear whether notions of SSO-CCA imply notions of ISO-CCA in the presence of openings, as a straightforward reduction runs into trouble with simulating the decryption oracle, see Open Problem 5. Slightly more technical, we expand an earlier CPA result by showing an equivalence between IND-CCA and ISO-CCA$\diamond$ (Thm. 7), but a similar expansion demonstrating equivalence of SSO-CCA$\diamond$ and IND-CCA runs into trouble simulating decryptions (Open Problem 11).

Message samplers play a central role in definitions of SOA (the “a posteriori” philosophies), and here again we find a number of open problems. For example, for which (restricted) classes of message samplers do κ-ISO-CPA$\star$ and κ-ISO-CCA$\star$, like their sender opening counterparts ($\odot$), become equivalent to IND-CPA resp. IND-CCA (Open Problem 10)?

Yet other open problems concern achievability: for example, can κ-ISO-CPA$\circledast$ be achieved in the standard model (Open Problem 19)? For achievable notions, a secondary question becomes how tightly, for instance for our novel notion of (κ, β)-IND-CPA$\circledast$ (Open Problem 21)?

We hope that highlighting these open problems can serve as inspiration.

Applications. With the hierarchy in hand, one may ask what notion is really needed for a given application: going higher in the hierarchy broadens applicability at the cost of achievability, with several impossibility results sitting towards the top. Take for instance NCE with receiver openings under chosen plaintext attacks (NCE-CPA$\star$): impossible to achieve in the plain model, yet easily achieved in the programmable random oracle model; and, once achieved, it allows one to show security of a plethora of multi-party computation protocols that were, before the appearance of the notion, only known to be secure against adaptive adversaries assuming information-theoretically secure channels between the parties [6, 100].

For some protocols, for instance involving secret sharing with encrypted shares, the more achievable ISO may suffice. For yet other applications, e.g. building authenticated key exchange (AKE), the multi-user with corruptions notion—the only notion of the hierarchy equivalent to IND-CCA—suffices [2].

As we present the various notions in Sect. 3, we also discuss their pros and cons in terms of usability and achievability. A summary follows (for references, see the relevant sections):

- The main challenge of a priori indistinguishability (IND) is that these notions traditionally do not allow an adversary to open challenge ciphertexts, making them unsuitable for modelling sender and

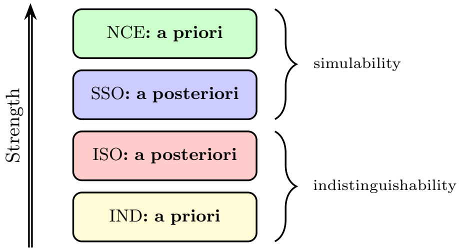


Fig. 1. The main hierarchy of philosophies, and their associated security notion with opening. The notions become stronger as we go up the hierarchy in the presence of sender or receiver opening (or both).

transmission openings, and for primitives such as secret sharing that rely on the adversary learning a subset of the challenge plaintexts. As we show, the use of multiple challenge bits allows also sender and transmission openings to be modelled in the a priori indistinguishability setting, though at the expense of composition challenges and tightness losses. On the positive side, notions of a priori indistinguishability are all implied by IND-CPA/IND-CCA, meaning many well-established constructions are known to achieve them, although not necessarily tightly so (see Sect. 3.3). Furthermore, if plaintexts are interrelated in only a limited way, it is possible that IND-CPA/ IND-CCA already suffices for ISO$\odot$-security, see Sect. 4.2.

- A posteriori indistinguishability (ISO) does allow for the opening of challenge ciphertexts, but at the cost of messages now being sampled rather than simply chosen. Thus, a heavier formalism involving message samplers becomes necessary, and, particular to a posteriori indistinguishability, involve conditional resampling, which may or may not be efficient for a particular application (see Sect. 3.4).
- A posteriori simulatability (SSO) does away with the resampling phase, leading to yet broader applicability, but at the cost of a significantly stronger notion: one that, in the presence of receiver openings (SSO$\star$), is even known to be unachievable in the standard model (see Sect. 3.5).
- Finally, a priori simulatability (NCE) does away with the need for message samplers, making for an arguably simpler notion, but at the cost of being the strongest and thus hardest to achieve. In particular, in the presence of receiver openings, it cannot be achieved in the standard model. On a more technical level, we disallow challenging a corrupted receiver in notions of a priori simulatability, and so if this restriction clashes with a particular use case then the lower three levels of the hierarchy might be more suitable (see Sect. 3.6).

One central question remains: when modelling openings, which notion of security is the right one? While there may not exist a definite answer, a few general recommendations come to mind:

- Firstly, assuming the goal is to model realistic settings, consider choosing a notion with bi-openings, as messages, encryption randomness, and keys all leak in the real world.
- Secondly, when proving your construction secure in the presence of openings, aim for the highest possible notion in the hierarchy, and (assuming it falls short of the top), consider providing some intuition as to why your construction does not reach higher (e.g. it is a standard model construction, and thus can not achieve simulatability notions with receiver openings; or it is binding, and thus can not achieve simulatability notions with sender openings; or simply that attempts at a proof ran into trouble).
- Thirdly, if you are applying PKE as a primitive in a larger protocol, stick to the lower notions unless you have a good reason to go higher, such as a need to open challenges while staying single-challenge-bit (ISO), or the need to support message distributions that are not efficiently conditionally resampleable (SSO). This ensures that the broadest possible set of PKE constructions can be used to instantiate your protocol.

Ultimately, we hope that this work can provide a helpful framework for comparing and contextualizing alternative notions of security with openings, and encourage their application in protocol design by untangling the surrounding literature. For example, as the main usefulness of notions of SOA lie in their ability to open a subset of challenge ciphertexts to unveil sampled plaintexts to the adversary—even when said plaintexts are arbitrarily interrelated—we expect that such notions have the potential to be particularly useful in the context of threshold cryptography. And yet we are aware of only a single work to date that applies a primitive achieving a SOA-like notion of security to prove a higher level protocol secure [17]. We hope the present systematization will help expand this list.

Future directions. We limit our investigation to perfect correctness, as most of the literature cited does not consider imperfectly correct schemes. However, with the rise of post-quantum cryptography, and lattice-based schemes in particular, the study of imperfect correctness is of increasing importance, and we view it as an important open problem to re-establish the relations studied herein in such a setting.

On a related note: several results considered herein rely heavily on (programmable) random oracles, or similar idealized models, which necessitate quantum upgrades such as the quantum random oracle model (QROM) [20] in order to be able to claim post-quantum security; thus re-establishing results in the post-quantum setting provides an intriguing double challenge.

Besides confidentiality, related goals have also been studied in the presence of openings, such as non-malleability [81] and anonymity (key privacy) [80]. While beyond our present scope, we find it would be interesting and worthwhile to re-establish our hierarchy in these settings, and investigate the notions of security the four philosophies give rise to when combined with the various openings.

Roadmap. After laying some groundwork in Sect. 2, we discuss the four kinds of openings in Sect. 3.1, and the four philosophies in Sect. 3.2. We then move our way up the hierarchy, from multi-user indistinguishability with openings (IND) at the bottom in Sect. 3.3, via indistinguishability-SOA (ISO) in Sect. 3.4 and simulation-SOA (SSO) in Sect. 3.5, to non-committing encryption (NCE) in Sect. 3.6. For each, we give a generalized definition stated in a unified formalism, survey other possible choices in the definitions, discuss the pros and cons in terms of applicability, highlight open problems, give a concrete-security proof that the notion in question implies the notion below it in the hierarchy, and provide some historical remarks. Additionally, in Sect. 3.3, we study for the first time notions of transmission, sender and bi-openings in the a priori indistinguishability setting, which can be modelled with security games employing more than one challenge bit, and show them to be poly-equivalent to the single- and multi-user notions of IND-CPA/IND-CCA at a loss linear in the number of challenge bits (tighter in the case of transmission openings).

In Sect. 4, we map out known relations, in terms of hybridization in the number of users (Sect. 4.1), implications between notions (Sect. 4.2), and separations between notions (Sect. 4.3). We identify a large number of open problems along the way, and we close a few. In Sect. 4.5, we discuss a selection of related notions that have appeared over the years, and where they sit in relation to our main hierarchy.

In Sect. 5 we move on to constructions. We identify in Sect. 5.1 a proof technique, which we name Bellare–Rogaway Encryption after its (first) inventors [12], which has appeared repeatedly in constructions aiming at the various notions, and with which one can in fact achieve the notion sitting at the very top of the hierarchy: non-committing encryption with bi-openings (the caveat being that the proof, by necessity, relies on programming a random oracle). Finally, in Sect. 5.2 we survey what other constructions are known to achieve the respective notions, both in idealized models like the programmable random oracle model (ROM), and in the standard model; Table 2 provides a helpful menu of options.

2 Preliminaries

2.1 Notation

For a positive integer n, we let $[n]$ denote the set $\{1, \ldots, n\}$. For a bit string $x \in \{0,1\}^*$, $|x|$ denotes its length. For a finite set $\mathcal{X}$, $|\mathcal{X}|$ is the cardinality of $\mathcal{X}$. For a list $\mathtt{X}$, $\mathtt{X}[i]$ retrieves the ith element of the list, and, by convention for an index set $\mathcal{I}$, $\mathtt{X}[\mathcal{I}]$ indicates the list $\mathtt{X}$ restricted to the elements whose indices are contained in $\mathcal{I}$. For $n \in \mathbb{Z}_{>0}$, we write $\mathtt{X}[\![n]\!]$ for $\mathtt{X}$ restricted to its first n elements, so it equals $\mathtt{X}[\mathcal{I}]$ with $\mathcal{I} = [n]$.

We use code-based experiments, where by convention all sets, lists, and lazily sampled functions are initialized empty. We use $\Pr[\mathsf{Code} : \mathsf{Event} \mid \mathsf{Condition}]$ to denote the conditional probability of Event occurring when Code is executed, conditioned on $\mathsf{Condition}$. We omit Code when it is clear from the context and $\mathsf{Condition}$ when it is not needed. In our experiments, we will assume that no oracle calls are made (by the adversary) with evidently out-of-bounds inputs (e.g. list indices or key handles).

In our (pseudo)code, we denote with $x \leftarrow y$ the deterministic assignment of y to the variable x and use the shorthand $\mathcal{X} \xleftarrow{\cup} x$ for the operation $\mathcal{X} \leftarrow \mathcal{X} \cup \{x\}$ and $\mathtt{X} \xleftarrow{\frown} x$ for appending the element x to a list $\mathtt{X}$. Furthermore, we use the shorthand $x \leftarrow\!\!\$\ \mathcal{X}$ to denote uniform sampling from the finite set $\mathcal{X}$ and $x \leftarrow\!\!\$\ Y(\cdot)$ for assigning the output of the probabilistic algorithm Y to x. We can make the randomness r of Y explicit by writing $x \leftarrow Y(\,\cdot\,; r)$, for previously sampled r.

We will also consider stateful algorithms, for instance when we write $m \leftarrow\!\!\$\ \mathsf{M}_{\langle s\rangle}(\alpha)$ the algorithm M takes as state s and as input the value α and outputs m, but it may change its state s as part of its processing (code-snippet taken from Fig. 8).

2.2 PKE Syntax

A public-key encryption scheme PKE consists of five algorithms. The probabilistic parameter generation algorithm $\mathsf{PKE.Pm}$ on input a security parameter λ outputs shared, public system parameters pm (these might for instance be the description of an elliptic curve group with generator for an ECDLP-based system); we use the shorthand $\mathsf{PKE}[\lambda]$ for a concrete instantiation of the scheme for the given security parameter. The probabilistic key generation algorithm $\mathsf{PKE.Kg}$ on input pm outputs a public/private key pair $(\mathsf{pk}, \mathsf{sk})$; without loss of generality, we assume that any algorithm that receives pk implicitly receives pm as well, and any algorithm receiving sk also receives pk. The probabilistic algorithm $\mathsf{PKE.Rnd}$ on input pm samples random coins r for encryption. Subsequently, the deterministic encryption algorithm $\mathsf{PKE.Enc}$ on input a public key pk, a message $m \in \mathcal{M}$ and randomness r outputs a ciphertext c. We can also treat $\mathsf{PKE.Enc}$ as a probabilistic algorithm by folding in the randomness generation by $\mathsf{PKE.Rnd}$, simply writing $c \leftarrow\!\!\$\ \mathsf{PKE.Enc}_{\mathsf{pk}}(m)$. Finally, the typically deterministic decryption algorithm $\mathsf{PKE.Dec}$ on input a private key sk and a ciphertext c outputs either a message $m \in \mathcal{M}$ or some failure symbol $\perp$. Henceforth, we assume that the message space $\mathcal{M}$ consists of (a subset of) arbitrary, finite length bit strings, i.e. $\mathcal{M} \subseteq \{0,1\}^*$ (which includes fixed-length binary representations of an algebraic structure). In addition, ciphertexts are typically bit strings whose length depends on, and hence leaks, the message length.

We typically assume the schemes to be perfectly correct, so

$$\Pr[r \leftarrow\!\!\$\ \mathsf{PKE.Rnd}(\mathsf{pm}) \,:\, \mathsf{PKE.Dec}_{\mathsf{sk}}(\mathsf{PKE.Enc}_{\mathsf{pk}}(m; r)) = m] = 1$$

for all parameters pm, all key pairs $(\mathsf{pk}, \mathsf{sk})$ generated by $\mathsf{PKE.Kg}(\mathsf{pm})$, and all messages $m \in \mathcal{M}$.

Remark 1. Some modern schemes, like LWE-based ones, allow a small probability of incorrectness, where decryption of an honestly generated ciphertext may occasionally return a wrong message or fail. Some classical schemes, such as hybrid KEM–DEM ones, may return distinct error messages (e.g. a KEM failure $\perp_{\mathsf{KEM}}$ versus a DEM failure $\perp_{\mathsf{DEM}}$). Other schemes might loosen the link between message length and ciphertext length to partially hide the former [51, 110]. To deal with these more general scenarios, some of the security definitions might require some subtle changes (beyond pure syntactical ones) to best capture the changed reality; moreover, even when the definitions remain the same, security proofs might rely on perfect correctness. While we occasionally highlight specific challenges when faced with a more general scenario, we stress that results shown to hold for single-error, length-regular, perfectly correct schemes do not automatically port to either of those more general scenarios. Specifically, whether known results on SOA and NCE still hold in the imperfect correctness scenario is unclear and we leave open the challenge of re-establishing relations in a setting with imperfect correctness.

Open Problem 1. *How are known security definitions and their relations affected when lifted to a setting with imperfect correctness, multiple error messages, or deviations from length regularity?*

2.3 Security Notions

Concrete advantages. Most security notions can be phrased in terms of an adversarial goal and the adversary's powers. This separation in goals and powers is reflected in the notation GOAL-POWER, such as IND-CPA for indistinguishability under chosen plaintext attacks.

We are primarily concerned with indistinguishability-style notions that task the adversary with guessing a single bit. These notions are modelled by a distinguishing experiment $\mathsf{Exp}^{\text{goal-power}}_{\mathsf{PKE}[\lambda],\ldots}(\mathbb{A})$ that, at the end of the game, either outputs true (if the adversary guesses correctly) or false (if not), leading to a distinguishing advantage (Def. 1). We will occasionally conflate the Boolean values true and false with the bits 1 and 0, respectively, and some games may manipulate the adversary's output prior to checking its guess, for instance to ensure that "bad" adversarial behaviour cannot be advantageous.

Definition 1 (Distinguishing advantage). *Let* $\mathsf{PKE}[\lambda]$ *be given and let* $\mathsf{Exp}^{\text{goal-power}}_{\mathsf{PKE}[\lambda],\dots}(\mathbb{A})$ *be a distinguishing experiment for the notion* GOAL-POWER*, where the dots represent other possible dependencies of the notion (such as simulators or message samplers). Then an adversary* $\mathbb{A}$ *'s distinguishing advantage is defined as*

$$\mathsf{Adv}^{\text{goal-power}}_{\mathsf{PKE}[\lambda],\dots}(\mathbb{A}) \coloneqq 2 \cdot \Pr\Big[\mathsf{Exp}^{\text{goal-power}}_{\mathsf{PKE}[\lambda],\dots}(\mathbb{A})\Big] - 1\,.$$

Unless otherwise stated, all our notions, including single-user notions, are multi-challenge, modelled by one or more challenge oracles (that depend on the bit to be guessed). Their precise formalizations depend on the underlying philosophy related to capturing "nothing leaks" (see Sect. 3).

Many powers are associated with a helper oracle, like the decryption oracle $\mathcal{D}$ used to model chosen ciphertext attacks (CCA), or the various opening oracles (see Sect. 3.1). As part of the goal, we prefix notions by the number of users κ and/or challenge bits β, and as part of the powers we postfix notions by the opening oracles available to them ($\diamond$, $\odot$, $\star$ or $\circledast$), see Table 1. If κ resp. β is set to one we typically omit the prefix, and likewise the postfix absent openings.

Although our focus will be on CPA and CCA security, there are many other powers sitting in between in strength. Without going into details, these include plaintext checking attacks (PCA) [101], ciphertext verification attacks (CVA) [39, 88], constrained chosen ciphertext attacks (CCCA) [74], and replayable chosen ciphertext attacks (RCCA) [32]. In the context of selective opening attacks, the CPA and CCA worlds display some remarkable divergence, raising the question where the various intermediate notions would sit.

Whenever possible, we state results using the concrete security approach. Thus, advantages are studied directly without a clear concept of what entails "security". Furthermore, advantages are well-defined without having to specify whether, say, a simulator may depend on the adversary or vice versa. Of course, when providing concrete reductions, the order of quantifiers does matter and speaks to the generality of a reduction rather than to the strength of a notion. A good, concrete reduction will result in a bound that, when combined with contemporary cryptanalytic hardness estimates of any underlying problem, results in concrete parameter choices for the scheme, targeting any desired security level. A good, concrete reduction should also be instrumental to derive asymptotic security statements as corollaries.

For simplicity, we will assume that adversaries will always terminate and do so with correctly formatted output; similarly, we assume adversaries respect the input–output interface of their oracles. These conventions are without loss of generality as the format checks could easily be added explicitly without yielding the adversary any additional advantage; the termination is implicitly captured by bounding the number of oracle calls an adversary may make and, if need be, modelling the adversary as a random system instead of an algorithm.

The asymptotic alternative. For asymptotic security, algorithms are modelled as uniform, probabilistic interactive Turing machines (ITM) that should work for all λ. An algorithm is deemed efficient if its ITM runs in worst-case time polynomial in λ; a scheme is defined secure for a given security notion if, for all efficient adversaries, its advantage is negligible in the security parameter λ [53]. For more complicated notions, we also need to resolve further dependencies (the "..." in Def. 1) and here, the order of quantifiers—whether a simulator may depend on the adversary or vice versa—does play a crucial role in defining the strength of a particular definition of security. Moreover, for security notions defined relative to message samplers, these samplers will be restricted to (efficient) classes for which security holds.

As a scheme can be secure or not in the asymptotic setting, the conditions under which a notion is achievable becomes a topic of study. Relatedly, we will also meet several impossibility results, saying that security under a notion is unachievable for all schemes (relative to some general conditions).

Comparing security notions. Both in the concrete and asymptotic settings, security notions can be compared with one another, but often with slightly different purposes. In the concrete setting, bounds are explicit and what they look like matters. For instance, if a reduction has a security loss factor 1 with little to no computational overhead, then the reduction is tight and, if there are tight reductions in both directions, then the notions are tightly equivalent. If a bound, tight or otherwise, cannot be improved upon (for instance there is a matching attack or metareduction), then the bound is sharp.

In the asymptotic setting, the emphasis is usually on implications, which is shown by the existence of a polynomial reduction, i.e. one whose security loss and computational overhead are polynomial in the security parameter. If there are polynomial reductions in both directions, then the notions are (polynomially) equivalent. If a security notion is not implied by another, then a separation may be shown, for instance by providing a (conditional) counterexample: a scheme that fulfils the definition of a PKE and can be proven secure under one notion, but for which there is an efficient adversary with a significant (typically overwhelming) advantage under the other notion.

If there is an implication in one direction and a separation in the other, then one notion is strictly stronger than the other; if there are separations in both directions, then the notions are incomparable. If there is neither an implication nor a separation between two notions, there is work to do!

Concrete security, specifically tightness and sharpness of bounds, only really becomes relevant when implications are known to exist. For instance, a priori indistinguishability with openings (Sect. 3.3) is polynomially equivalent to IND-CCA, but not tightly so, and is therefore only of interest in a concrete security setting. The remaining philosophies with openings have on the other hand been almost exclusively studied in the asymptotic setting, and the map of relations given in Fig. 16 is built on polynomial implications and separations.

Several of the security notions are furthermore defined relative to message samplers and/or simulators. In the concrete setting, reductions are then given relative to concrete samplers/simulators (see e.g. Thm. 4). In the asymptotic setting, they are instead quantified over as part of the definition of security, and implications between security notions are given relative to classes of samplers/simulators. Thus the asymptotic setting facilitates more general reductions, at the cost of concreteness.

We generally prefer the concrete approach over the asymptotic approach where possible. We state our results accordingly, but revert to the asymptotic mindset for the purpose of recalling many known results. We strive to make it clear from context and language usage which mindset we are working in at any time (e.g. tightness/lossiness in the concrete setting, implications/separations in the asymptotic setting).

Most of the reductions we consider are black-box, in the sense that they treat the adversary they build upon as a black-box. Furthermore, for generality, we often state that our reductions are type-preserving, which means that the type of queries the reduction makes, matches those of the underlying adversary. Type-preserving reductions are convenient to show simultaneously that, for instance, both CCA security of one flavour implies CCA security of another flavour and CPA security of that one flavour implies CPA security of that other flavour.

3 Confidentiality with Openings

3.1 Four Kinds of Opening

In a system with many users, one would like a guarantee that uncompromised traffic remains confidential even if a subset of the users are compromised. We will address different confidentiality guarantees in the next section and first explore user compromises themselves. Typical deployment of PKE involves two distinct user roles: that of sender and that of receiver. Their compromises need to be modelled separately, leading to four distinct flavours of openings.

Transmission openings. The weakest form of opening allows an adversary to open challenge ciphertexts to retrieve the underlying message. It differs from a chosen ciphertext attack as it specifically targets challenge ciphertexts, which are explicitly prohibited for a CCA-style decryption oracle. The notion is relatively rare, but we include it for completeness. We use the name transmission openings, let $\mathcal{T}$ denote the transmission opening oracle, and indicate its presence with the suffix $\diamond$.

Transmission openings can model partial compromises on both sender and receiver end: a sender might still have (a copy of) the message lying around, but have erased the ephemeral randomness used to encrypt; a receiver might take strong measures not to leak its long-term private key but might not care too much about the contents of a single message leaking.

The added power of the transmission opening oracle to an adversary appears minimal, in contrast to the stronger sender and receiver openings that we will discuss next. Indeed, Bellare and Yilek [16] (henceforth BY12) showed transmission-SOA equivalent to IND-CPA in the context of simulation-based selective opening attacks (which we will consider as a posteriori simulatability in the next section), see Sect. 4.2. Similarly, for a priori indistinguishability, we show that the added power is minimal (see Thm. 2), while for a priori simulatability, a transmission opening oracle would not add anything to the notion (as the adversary already knows the plaintext, see Def. 7).

Sender openings. Here an adversary can open any challenge ciphertext to receive both the message and underlying randomness; it models the compromise of a sender incapable of securely erasing said randomness. The study of SOA started out as a study of security in the presence of randomness reveals [41], as motivated by the setting of multi-party computation, where erasures are notoriously tricky [10].

Sender openings can be considered in a multiple-senders/single-receiver setting, so there is only one public–private key pair, yet the opening is per ciphertext. We let $\mathcal{S}$ denote the sender opening oracle, and indicate its presence with the suffix $\odot$.

Compared to transmission openings, and depending on the formalism, sender openings are much harder to deal with formally: the core technical difficulty sits with the committing property of most encryption schemes, as any adversary receiving the randomness and message corresponding to a challenge ciphertext can re-encrypt to verify that challenges and openings are consistent.

Receiver openings. An adversary who fully corrupts a receiver will obtain that user's private key. These kind of openings are found in both the multi-user literature [2,3,85] and the SOA literature [9,63,91].

For the latter, it is customary to reveal not just the private key, but also all the challenge messages that were encrypted under the corresponding public key. For us, receiver openings will only reveal the private key. For perfectly correct schemes, this choice is without loss of generality, as evidently, an adversary with access to both a ciphertext and the private key can simply run the decryption algorithm to obtain the originally encrypted message. When perfect correctness is not guaranteed, having one oracle that only reveals the private key and another that reveals the messages (cf. transmission openings) should result in a finer-grained notion.

We let $\mathcal{R}$ denote the receiver opening oracle, and indicate its presence with the suffix $\star$. For receiver openings to be meaningful, one should consider multiple receivers; we let κ indicate the number of receivers (used as prefix for security notions). As for sender openings, the core difficulty of receiver openings is the adversary's ability to verify that challenges and openings are consistent.

Bi-openings. Finally, an adversary might be granted access to both sender and receiver opening oracles (and thus also transmission openings), as indicated by the suffix $\circledast$. Bi-openings have been the standard in the NCE setting from the get-go [6,29], but only recently appeared in the SOA literature [91].

3.2 Four Philosophies of Confidentiality

To contrast and compare different confidentiality notions with openings, we first revisit four different philosophies to formalize confidentiality without. All four approaches aim to capture that an adversary "learns nothing" and hark back to Shannon's concept of perfect secrecy and its (near) equivalent notions [107]. The notions split in two, depending on whether they are indistinguishability versus simulatability based, and for both we consider an *a priori* and an *a posteriori* variant (see Fig. 1). We next describe each, going roughly in order of increasing strength.

A priori indistinguishability. In the information-theoretic, symmetric setting (where $\mathcal{M}$ consists of fixed-length bistrings) the idea is that, given any two messages, the same distribution over ciphertexts is induced, which can be formalized by stating that, for all m_0 and m_1 (and c),

$$\Pr[C = c \mid M = m_0] = \Pr[C = c \mid M = m_1]\,,$$

where the probability is primarily over the choice of the secret key. In the computational, public key setting, the formalization differs, leading to classic left-or-right indistinguishability: an adversary, given access to a single public key pk, selects two (equal length) messages m_0 and m_1, receives the encryption of one of them under pk, and needs to figure out which one. Generalizations to allow multiple challenges and multiple users [7] form the basis for multi-user indistinguishability with corruptions (κ-IND-CPA$\star$ and κ-IND-CCA$\star$, see Sect. 3.3).

A posteriori indistinguishability. Here the concept is that, given a ciphertext, any two messages are equally likely. In the information-theoretic setting, it can be formalized as

$$\Pr[M = m_0 \mid C = c] = \Pr[M = m_1 \mid C = c]\,,$$

which hides a dependency on the message distribution underlying M: the notion can only be satisfied iff the a priori distribution on the messages is uniform (in which case it is equivalent to a priori indistinguishability via Bayes's theorem).

For a computational PKE version, consider an experiment where an adversary specifies a message distribution M over $\mathcal{M} \subseteq \{0,1\}^*$. The game would sample a message m_0 according to M and encrypt m_0 to obtain ciphertext c. It would then sample a second message m_1 according to M, conditioned on $|m_0| = |m_1|$. Finally, the game returns (m_b, c) to the adversary, who has to guess b.

A posteriori indistinguishability is rather a rare notion for PKE, nonetheless it is the route taken for indistinguishability-based notions of selective opening attacks (ISO for short, see Sect. 3.4). In order for the notion to imply IND-CPA, however, the adversary should be allowed adaptive control over the distribution. In the concrete setting, we define the notion relative to a conditional resampler. In the asymptotic setting, this conditional resampler causes a bifurcation of the notion depending on whether said resampler can be efficiently implemented (effectively restricting the message samplers) or not.

Remark 2. While rare for PKE, a posteriori indistinguishability is reminiscent of standard notions of indistinguishability for KEMs, in which the adversary, given a ciphertext, must distinguish between the encapsulated symmetric key and a freshly drawn key [37]. With transmission openings, a posteriori indistinguishability closely matches Enhanced IND-CPA/IND-CCA for KEMs, in which the adversary receives challenge ciphertexts and is given the choice of whether to open it, revealing the encapsulated key, or to challenge it to receive either the encapsulated key, or a freshly drawn one [57].

A posteriori simulatability. Shannon's definition of perfect secrecy captured that, given a ciphertext c, each message m is as likely as it was before seeing c, or more formally

$$\Pr[M = m \mid C = c] = \Pr[M = m]\,;$$

again there is a dependency on the message distribution, but this time it can be arbitrary (so it is less troublesome than for a posteriori indistinguishability).

Perfect secrecy captures that, given a ciphertext, nothing should be leaked about the message. Goldwasser and Micali [54] famously captured this concept in a computational PKE setting as semantic security (SEM). Goldreich [53] later refined the notion by introducing a simulator: an adversary outputs a message distribution M, the game samples m according to M, encrypts m and returns the resulting ciphertext c to the adversary. Whatever the adversary subsequently computes, possibly using additional oracles, a simulator should be able to simulate. Some definitional variations are possible [111], based for instance on whether the adversary outputs only a single bit or an arbitrary output (to be simulated).

Like perfect secrecy, semantic security is arguably the most intuitive notion. Compared to a posteriori indistinguishability, there is no need to put any onerous restrictions on M, that is on how messages are sampled (although in the computational setting, it does need to be efficient). Simulation-based notions for selective opening attacks (SSO) follow this philosophy, see Sect. 3.5 for details.

A priori simulatability. Finally, we can require that the likelihood of observing a ciphertext c is independent of m, or

$$\Pr[C = c \mid M = m] = \Pr[C = c]\,.$$

In the information-theoretic setting, Shannon already showed this formalization equivalent to perfect secrecy.

The concept can be reinterpreted to say that, given a message, nothing is learned about the ciphertexts that might result from encrypting it, which can be captured by saying one can produce, or simulate, fully convincing ciphertexts without access to the message (before an adversary even gets involved). Hence, a priori simulatability.

Restricting to chosen-plaintext attacks for PKE, a priori simulatability can be defined by allowing an adversary to obtain the public key, select a message m, and then either receive an encryption of m or a simulated ciphertext, with the simulator only given access to the public key and the length of the message. The most common incarnation of a priori simulatability in the PKE setting fixes the simulator to simply select a random message of matching length and encrypting it, and the resulting notion is known as real-or-random indistinguishability (ROR) [8].

Upgrading to chosen-ciphertext attacks, the decryption oracle might have to be simulated as well; precise formalizations of a priori simulatability can be found in the universal composability (UC) framework [27, 90]. When allowing opening oracles, these need to be simulated as well; the resulting notion is known as (non-interactive) non-committing encryption (NCE), as discussed in Sect. 3.6.

```
Experiment $\mathsf{Exp}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{A})$
    $b \leftarrow\!\!\$\ \{0,1\}$
    $\mathsf{pm} \leftarrow\!\!\$\ \mathsf{PKE.Pm}(\lambda)$
    $\forall_{i\in[\kappa]}(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow\!\!\$\ \mathsf{PKE.Kg}(\mathsf{pm})$
    $\hat{b} \leftarrow\!\!\$\ \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{R}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
    if $\mathcal{K} \cap \mathcal{I} \neq \emptyset : \hat{b} \leftarrow\!\!\$\ \{0,1\}$
    return $b = \hat{b}$
```

```
Oracle $\mathcal{E}(i, m_0, m_1)$
    if $|m_0| \neq |m_1|$ : return $\lightning$
    $\mathcal{K} \xleftarrow{\cup} i$
    $c \leftarrow\!\!\$\ \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m_b)$
    $\mathcal{C}_i \xleftarrow{\cup} c$
    return $c$
```

```
Oracle $\mathcal{D}(i, c)$
    if $c \in \mathcal{C}_i$ : return $\lightning$
    $m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
    return $m$

Oracle $\mathcal{R}(i)$
    $\mathcal{I} \xleftarrow{\cup} i$
    return $\mathsf{sk}_i$
```

Fig. 2. A priori indistinguishability with receiver openings, also known as multi-user indistinguishability with corruptions.

Discussion. If we exclude a posteriori indistinguishability, then the remaining three notions are all equivalent in the information-theoretic setting, as already proven by Shannon. The same is true in the asymptotic computational setting: again excluding a posteriori indistinguishability, IND-CPA, SEM-CPA, and ROR-CPA are all polynomially equivalent [8, 54], as are IND-CCA, SEM-CCA, and UC-CCA [75, 111].

A posteriori indistinguishability appears not to have been studied for PKE without openings, although equivalence for message samplers satisfying conditional resamplability follows from Sect. 4.

3.3 A Priori Indistinguishability with Selective Openings (IND)

The multi-user setting [7] fits within the framework of a priori indistinguishability. Originally, openings were not considered, yet modelling multi-user security with receiver openings, also known as corruptions has seen an uptick [2, 58, 68, 92]. Several formalizations are possible, depending for instance on whether a single challenge bit is used across all κ public keys or whether each public key is allocated its own challenge bit. For receiver openings, we concentrate on the former, more standard approach (see Def. 2).

With only a single challenge bit, the opening of challenges would enable a trivial win and so must be disallowed. Thus, there is no meaningful notion of transmission or sender opening in such experiments. Generalizing to multiple challenge bits alleviates this issue: in the following, after recalling the canonical single-bit multi-user notion with receiver openings, we study a single-user multi-bit notion with sender openings, which to the best of our knowledge is studied here for the first time. We then collect the pieces and present the first notion of a priori indistinguishability with bi-openings, which, by allowing both multiple users and multiple challenge bits, strictly generalizes the prior notions.

Receiver openings. As explained, our formalization of a priori indistinguishability with receiver openings matches canonical notions of multi-user indistinguishability with corruptions, and we adopt a recent formalization [68].

Definition 2. *The* κ-IND-CCA$\star$ *advantage* $\mathsf{Adv}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{A})$ *of an adversary* $\mathbb{A}$ *against public key encryption scheme* $\mathsf{PKE}[\lambda]$ *is the distinguishing advantage against the game* $\mathsf{Exp}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{A})$ *(see Fig. 2).*

Uses and Limitations. By a straightforward hybrid argument, the multi-user setting with receiver openings is implied by the single-user setting with a κ tightness loss, which entails polynomial equivalence [7].

For concrete instantiations, there are schemes known to be tightly secure in the multi-user setting with corruptions in the programmable random oracle model (see Sect. 5.2). The notion furthermore benefits from ease of composition, e.g. in constructing tightly secure hybrid encryption from a KEM and a DEM [92].

The main limitation of the notion is its segregation of opening versus challenging: adversaries cannot gain any advantage through both challenging and opening a user, as enforced by overwriting the adversary's output with a uniform value in the final stage of $\mathsf{Exp}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{A})$. This makes the notion inadequate for e.g. threshold security [105], for which security should hold as long as not too many challenges are opened.

Opening challenges. Employing multiple challenge bits, opening challenges becomes a viable strategy, provided the adversary outputs an uncompromised bit handle and guess at the end. For receiver openings, each user may for instance be associated a challenge bit; so that users can now be both challenged and

```
Experiment $\mathsf{Exp}^{\beta\text{-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{A})$
    $\forall_{i \in [\beta]} b_i \leftarrow\!\!\$ \{0,1\}$
    $\mathsf{pm} \leftarrow\!\!\$ \mathsf{PKE.Pm}(\lambda)$
    $(\mathsf{pk}, \mathsf{sk}) \leftarrow\!\!\$ \mathsf{PKE.Kg}(\mathsf{pm})$
    $(i, \hat{b}_i) \leftarrow\!\!\$ \mathbb{A}^{\mathcal{E},\mathcal{D},(\mathcal{T},)\mathcal{S}}(\mathsf{pk})$
    if $i \in \mathcal{I} : \hat{b} \leftarrow\!\!\$ \{0,1\}$
    return $b_i = \hat{b}_i$
```

```
Oracle $\mathcal{E}(i, m_0, m_1)$
    if $|m_0| \neq |m_1|$ : return $\lightning$
    $r \leftarrow\!\!\$ \mathsf{PKE.Rnd}(\mathsf{pm})$
    $c \leftarrow \mathsf{PKE.Enc}_{\mathsf{pk}}(m_{b_i}; r)$
    $\mathtt{E} \xleftarrow{\frown} (i, m_{b_i}, r)$
    $\mathcal{C} \xleftarrow{\cup} c$
    return $c$
```

```
Oracle $\mathcal{D}(c)$
    if $c \in \mathcal{C}$ : return $\lightning$
    $m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}}(c)$
    return $m$
```

```
Oracle $\mathcal{T}(j)$
    $(i, m, r) \leftarrow \mathtt{E}[j]$
    $\mathcal{I} \xleftarrow{\cup} i$
    return $m$
```

```
Oracle $\mathcal{S}(j)$
    $(i, m, r) \leftarrow \mathtt{E}[j]$
    $\mathcal{I} \xleftarrow{\cup} i$
    return $(m, r)$
```

Fig. 3. A multiple-challenge-bit a priori indistinguishability game with transmission and sender openings.

corrupted, as long as at least one uncompromised user remains by the end. A KEM version of this multiple-challenge-bit security notion suffices for a construction achieving tightly (multi-challenge-bit) secure authenticated key exchange [2].

However, having multiple challenge bits comes with its own set of challenges: in particular, it typically leads to lossy composition theorems [68, 84]. The notion might also remain inadequate for e.g. threshold schemes, as having multiple challenge bits makes it hard to keep challenges consistent with each other.

Nonetheless, it facilitates the study of transmission and sender (and therefore also bi-) opening in the a priori indistinguishability setting.

Sender openings. We consider a single-receiver notion without receiver openings and define $\mathsf{Adv}^{\beta\text{-ind-cca}\odot}_{\mathsf{PKE}}(\mathbb{A})$ in the vein of Def. 2 using Fig. 3. At the start of the game, β challenge bits are drawn, representing β senders. The adversary can learn the value of a challenge bit by opening any challenge for which $m_0 \neq m_1$, which will no longer make for a valid guess. The notion is implied by the single-user notion (Thm. 1), essentially through a guessing argument, leading to a tightness loss linear in the number of challenge bits.

A comparable loss, namely the κ security loss from IND-CCA to κ-IND-CCA⋆ is known to be sharp, both in the sense that there are schemes that meet the bound [7], and through meta-reduction showing that no black-box reduction can achieve a better bound [3]. One can wonder whether Thm. 1 is similarly sharp, as expressed in Open Problem 2.

Open Problem 2. *How sharp is the bound of Thm. 1?*

Theorem 1. *Let* $\mathsf{PKE}[\lambda]$ *be given. Then there is a type-preserving black-box reduction* $\mathbb{B}_{\text{ind}}$ *such that, for all* $\mathbb{A}_{\odot}$,

$$\mathsf{Adv}^{\beta\text{-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\odot}) \leq \beta \cdot \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ind}}) .$$

The runtime of $\mathbb{B}_{\text{ind}}$ *is upper bounded by that of* $\mathbb{A}_{\odot}$ *plus* q_e *encryptions and* q_d *decryptions, where* q_e *and* q_d *are* $\mathbb{A}_{\odot}$*'s number of challenge oracle calls and decryption oracle calls respectively, and some small overhead.*

Proof. The proof relies on two lemmas (stated and proved below), that together imply the stated result. The first one, Lemma 1, shows that 1-IND-CCA⊙ (Fig. 3 with $\beta = 1$) implies β-IND-CCA⊙ with a β tightness loss. The second one, Lemma 2, shows that when there is only one challenge bit, transmission/sender opening oracles do not help the adversary at all; in that case IND-CCA tightly implies 1-IND-CCA⊙. □

One could of course chain the reductions from the proofs of Lemma 1 and Lemma 2 in an attempt to create a more direct proof of Thm. 1. In essence, such a combined reduction would still make an initial guess i' on which sender i the adversary will end up attacking, forwarding challenges to its own challenge oracle whenever a challenge is asked on that handle and otherwise simulating the oracles off-line. The question though is what the reduction should do in the event that a challenge constructed under the guessed bit handle is requested to be opened. It cannot simulate either opening oracle honestly, and

```
Reduction $\mathbb{C}_{1\odot}(\mathsf{pk})$
$i' \leftarrow\!\!\$\ [\beta], k \leftarrow 0$
$\forall_{i \in [\beta] \setminus i'} b_i \leftarrow\!\!\$\ \{0,1\}$
$(i, \hat{b}_i) \leftarrow\!\!\$\ \mathbb{A}_{\odot}^{\mathcal{E},\mathcal{D},\mathcal{T},\mathcal{S}}(\mathsf{pk})$
**if** $i = i' : \hat{b} \leftarrow \hat{b}_i$
**else** $: \hat{b} \leftarrow\!\!\$\ \{0,1\}$
**return** $\hat{b}$
```

```
If $\mathbb{A}_\odot$ calls $\mathcal{D}(c)$
**if** $c \in \mathcal{C}$ : **return** $\lightning$
$m \leftarrow \mathcal{D}_{\mathbb{C}}(c)$
**return** $m$
```

```
If $\mathbb{A}_\odot$ calls $\mathcal{E}_{\mathbb{A}}(i, m_0, m_1)$
**if** $i = i'$ :
    $k \leftarrow k + 1$
    $c \leftarrow \mathcal{E}_{\mathbb{C}}(m_0, m_1)$
    $\mathtt{E} \xleftarrow{\frown} (i, \perp, \perp, k)$
**else** :
    **if** $|m_0| \neq |m_1|$ : **return** $\lightning$
    $r \leftarrow\!\!\$\ \mathsf{PKE.Rnd}(\mathsf{pm})$
    $c \leftarrow \mathsf{PKE.Enc}_{\mathsf{pk}}(m_{b_i}; r)$
    $\mathcal{C} \xleftarrow{\cup} c$
    $\mathtt{E} \xleftarrow{\frown} (i, m_{b_i}, r, \perp)$
**return** $c$
```

```
If $\mathbb{A}_\odot$ calls $\mathcal{T}(j)$
$(i, m, r, k) \leftarrow \mathtt{E}[j]$
**if** $i = i'$ :
    $m \leftarrow \mathcal{T}_{\mathbb{C}}(k)$
**return** $m$
```

```
If $\mathbb{A}_\odot$ calls $\mathcal{S}(j)$
$(i, m, r, k) \leftarrow \mathtt{E}[j]$
**if** $i = i'$ :
    $(m, r) \leftarrow \mathcal{S}_{\mathbb{C}}(k)$
**return** $(m, r)$
```

Fig. 4. The reduction $\mathbb{C}_{1\odot}$ simulating $\mathsf{Exp}_{\mathsf{PKE}[\lambda]}^{\beta\text{-ind-cca}\odot}$ for $\mathbb{A}_\odot$.

an inaccurate sender opening oracle is easily noticed by the adversary, so aborting the simulation (as the combined reduction would do) seems the natural option. Analysing the advantage of this guess-and-occassionally-abort reduction is doable with a sufficiently fine-grained case-analysis, but splitting the analysis in two Lemmas with their own reductions seemed the more modular and cleaner approach.

Lemma 1. *Let $\mathsf{PKE}[\lambda]$ be given. Then there is a type-preserving black-box reduction $\mathbb{C}_{1\odot}$ such that, for all $\mathbb{A}_\odot$,*

$$\mathsf{Adv}_{\mathsf{PKE}[\lambda]}^{\beta\text{-ind-cca}\odot}(\mathbb{A}_\odot) = \beta \cdot \mathsf{Adv}_{\mathsf{PKE}[\lambda]}^{1\text{-ind-cca}\odot}(\mathbb{C}_{1\odot})\,.$$

The runtime of $\mathbb{C}_{1\odot}$ is upper bounded by that of $\mathbb{A}_\odot$ plus q_e encryptions and q_d decryptions, where q_e and q_d are $\mathbb{A}_\odot$'s number of challenge oracle calls and decryption oracle calls respectively, and some small overhead.

Proof. The reduction $\mathbb{C}_{1\odot}$ (Fig. 4) makes a guess i' at the bit handle that $\mathbb{A}_\odot$ will end up attacking, forwarding challenge and opening oracle calls relating to that bit handle to its own oracles, and simulating the remaining challenge and opening oracle calls off-line. Decryption oracle calls are forwarded except if they involve a ciphertext issued as a challenge, in which case $\lightning$ is returned instead, as usual. Once $\mathbb{A}_\odot$ halts with a guess $(i, \hat{b}_i)$, $\mathbb{C}_{1\odot}$ checks whether the handle matches its guess, halting with $\hat{b}_i$ if yes and with a uniform guess otherwise.

As the simulation is perfect even if $\mathbb{A}_\odot$ asks to open a challenge produced under the guessed bit handle, $\mathbb{C}_{1\odot}$'s guess i' is information-theoretically hidden from $\mathbb{A}_\odot$, thus

$$\Pr[i = i'] = \frac{1}{\beta}\,.$$

Furthermore, if $\mathbb{C}_{1\odot}$ guessed incorrectly then its random bit $\hat{b}$ will be correct half of the time, so

$$\Pr\Big[\mathsf{Exp}_{\mathsf{PKE}[\lambda]}^{1\text{-ind-cca}\odot}(\mathbb{C}_{1\odot}) \;\Big|\; i \neq i'\Big] = \frac{1}{2}\,;$$

whereas if $\mathbb{C}_{1\odot}$ guessed correctly, its winning probability matches that of $\mathbb{A}_\odot$, regardless of whether or not $b_{i'}$ got compromised (if it did, the respective game mechanisms force the experiments of both $\mathbb{A}_\odot$ and $\mathbb{C}_{1\odot}$ to a comparison with a uniform random bit):

$$\Pr\Big[\mathsf{Exp}_{\mathsf{PKE}[\lambda]}^{1\text{-ind-cca}\odot}(\mathbb{C}_{1\odot}) \;\Big|\; i = i'\Big] = \Pr\Big[\mathsf{Exp}_{\mathsf{PKE}[\lambda]}^{\beta\text{-ind-cca}\odot}(\mathbb{A}_\odot)\Big]\,.$$

Using these observations, we can express $\mathbb{C}_{1\odot}$'s winning probability as

$$\begin{aligned}
\Pr\Big[\mathsf{Exp}^{\text{1-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{C}_{1\odot})\Big] &= \Pr[i = i']\Pr\Big[\mathsf{Exp}^{\text{1-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{C}_{1\odot}) \;\Big|\; i = i'\Big] \\
&\quad + \Pr[i \neq i']\Pr\Big[\mathsf{Exp}^{\text{1-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{C}_{1\odot}) \;\Big|\; i \neq i'\Big] \\
&= \frac{1}{\beta}\Pr\Big[\mathsf{Exp}^{\text{1-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{C}_{1\odot}) \;\Big|\; i = i'\Big] + (1 - \frac{1}{\beta})\frac{1}{2} \\
&= \frac{1}{\beta}\left(\Pr\Big[\mathsf{Exp}^{\beta\text{-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\odot})\Big] - \frac{1}{2}\right) + \frac{1}{2}\,;
\end{aligned}$$

and thus, after applying Def. 1, its advantage satisifies

$$\begin{aligned}
\mathsf{Adv}^{\text{1-ind-cca}\odot}(\mathbb{C}_{1\odot}) &= 2\left(\frac{1}{\beta}\left(\Pr\Big[\mathsf{Exp}^{\beta\text{-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\odot})\Big] - \frac{1}{2}\right) + \frac{1}{2}\right) - 1 \\
&= \frac{1}{\beta}\left(2\Pr\Big[\mathsf{Exp}^{\beta\text{-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\odot})\Big] - 1\right) \\
&= \frac{1}{\beta}\mathsf{Adv}^{\beta\text{-ind-cca}\odot}(\mathbb{A}_{\odot})\,.
\end{aligned}$$

□

Lemma 2. *Let* $\mathsf{PKE}[\lambda]$ *be given. Then there is a type-preserving black-box reduction* $\mathbb{B}_{\text{ind}}$ *such that, for all* $\mathbb{C}_{1\odot}$*,*

$$\mathsf{Adv}^{\text{1-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{C}_{1\odot}) = \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ind}})\,.$$

The runtime of $\mathbb{C}_{1\odot}$ *is upper bounded by that of* $\mathbb{B}_{\text{ind}}$*, plus some small overhead.*

Proof. The reduction $\mathbb{B}_{\text{ind}}$ simulates the game honestly by forwarding oracles up until the point that one of the opening oracles is called, at which point $\mathbb{B}_{\text{ind}}$ aborts the simulation and outputs a uniform guess. Let bad denote the event that an opening oracle is called. We have that

$$\Pr\Big[\mathsf{Exp}^{\text{ind-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ind}}) \;\Big|\; \neg\mathsf{bad}\Big] = \Pr\Big[\mathsf{Exp}^{\text{1-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{C}_{1\odot}) \;\Big|\; \neg\mathsf{bad}\Big]$$

and

$$\Pr\Big[\mathsf{Exp}^{\text{ind-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ind}}) \;\Big|\; \mathsf{bad}\Big] = \frac{1}{2}\,.$$

If $\mathbb{C}_{1\odot}$ calls one of its opening oracles, the real game will overwrite its guess with a uniform guess, and so

$$\Pr\Big[\mathsf{Exp}^{\text{1-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{C}_{1\odot}) \;\Big|\; \mathsf{bad}\Big] = \frac{1}{2}\,,$$

allowing us to conclude

$$\begin{aligned}
&\Pr\Big[\mathsf{Exp}^{\text{ind-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ind}})\Big] = \Pr\Big[\mathsf{Exp}^{\text{1-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{C}_{1\odot})\Big] \\
\Longrightarrow\;& \mathsf{Adv}^{\text{ind-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ind}}) = \mathsf{Adv}^{\text{1-ind-cca}\odot}_{\mathsf{PKE}[\lambda]}(\mathbb{C}_{1\odot})\,.
\end{aligned}$$

□

Transmission openings. Fig. 3 simultaneously serves to define advantages for the β-IND-CCA◇ and β-IND-CCA notions (by omitting $\mathcal{S}$ for the former and both $\mathcal{S}$ and $\mathcal{T}$ for the latter). The notions β-IND-CCA and IND-CCA are tightly equivalent absent opening [68, Thm. 1]. Thus it is the reveal of messages and randomness, as opposed to the additional challenge bits, that gives the notions their strength, as the proof of tight equivalence fails in the presence of sender or transmission openings.

For transmission openings, we are able to show an almost tight equivalence to IND-CCA, losing only a factor 2 in one direction, while being trivially tight in the other direction. We conclude that transmission openings are of little interest in the a priori indistinguishability setting, adding at most a completely marginal strength.

Thm. 2 is inspired by BY12's proof of equivalence of IND-CPA and SSO-CPA◇ and shows that β-IND-CCA◇ is implied by ROR-CCA within a factor 2. Recall that ROR-CCA also implies IND-CCA within a factor 2 [8], leaving open the possibility that β-IND-CCA◇ and IND-CCA are in fact tightly equivalent (Open Problem 3, see also Fig. 5).

$$\begin{tikzcd}
 & \beta\text{-IND-CCA}\odot \arrow[dl, Rightarrow] \arrow[dr, Rightarrow] & \\
\beta\text{-IND-CCA} \arrow[r, Rightarrow, shift left] & \text{IND-CCA} \arrow[l, Rightarrow, shift left, "{[68]}"] \arrow[u, "\cdot\beta\text{ (Thm. 1)}"] \arrow[d, Rightarrow, shift left] & \beta\text{-IND-CCA}\diamond \arrow[l, Rightarrow] \\
 & \text{ROR-CCA} \arrow[u, shift left, "\cdot 2\ {[8]}"] \arrow[ur, "\cdot 2\text{ (Thm. 2)}"'] &
\end{tikzcd}$$

Fig. 5. Relations among the single-user single- and multi-bit notions of indistinguishability, with and without sender/transmission opening. (Double arrows = tight.)

```
Reduction $\mathbb{B}_{\text{ror}}(\mathsf{pk})$
$\forall_{i\in[\beta]} d_i \leftarrow\!\!\$\ \{0,1\}$
$(i,\hat{d}_i) \leftarrow\!\!\$\ \mathbb{A}_\diamond^{\mathcal{E},\mathcal{D},\mathcal{T}}(\mathsf{pk})$
if $i \in \mathcal{I} : \hat{d}_i \leftarrow\!\!\$\ \{0,1\}$
$\hat{b} \leftarrow \neg(d_i = \hat{d}_i)$
return $\hat{b}$
```

```
If $\mathbb{A}_\diamond$ calls $\mathcal{E}_{\mathbb{A}}(i, m_0, m_1)$
if $|m_0| \neq |m_1|$ : return $\lightning$
$c \leftarrow \mathcal{E}_{\mathbb{B}}(m_{d_i})$
$\mathtt{E} \overset{\frown}{\longleftarrow} (i, m_{d_i})$
return $c$
```

```
If $\mathbb{A}_\diamond$ calls $\mathcal{T}(j)$
$(i, m) \leftarrow \mathtt{E}[j]$
$\mathcal{I} \overset{\cup}{\longleftarrow} i$
return $m$
```

Fig. 6. The reduction $\mathbb{B}_{\text{ror}}$ simulating $\mathsf{Exp}^{\kappa\text{-ind-cca}\diamond}_{\mathsf{PKE}[\lambda]}$ for $\mathbb{A}_\diamond$. (The decryption oracle is simply forwarded.)

Open Problem 3. *Are* IND-CCA *and* β-IND-CCA$\diamond$ *tightly equivalent?*

Theorem 2. *Let* $\mathsf{PKE}[\lambda]$ *be given. Then there is a type-preserving black-box reduction* $\mathbb{B}_{\text{ror}}$ *such that, for all* $\mathbb{A}_\diamond$,

$$\mathsf{Adv}^{\beta\text{-ind-cca}\diamond}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_\diamond) \leq 2 \cdot \mathsf{Adv}^{\text{ror-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ror}}) .$$

The runtime of $\mathbb{B}_{\text{ror}}$ *is upper bounded by that of* $\mathbb{A}_\diamond$ *plus some small overhead.*

Proof. Reduction $\mathbb{B}_{\text{ror}}$ is given in Fig. 6 (the $\mathcal{D}$ oracle is simply forwarded). Denote by b the challenge bit that $\mathbb{B}_{\text{ror}}$ is tasked with guessing, with $b = 0$ corresponding to "real" and $b = 1$ to "random". We will bound its advantage next.

Intuitively, if $\mathbb{A}_\diamond$ returns a compromised bit handle, then its β-IND-CCA$\diamond$ advantage becomes 0; in this case $\mathbb{B}_{\text{ror}}$ overwrites $\mathbb{A}_\diamond$'s guess with a uniform value, thus also gaining advantage 0 and equality holds. Otherwise there are two cases: either $b = 0$, in which case the simulation is completely faithful, or $b = 1$, in which case all uncompromised d_i are information-theoretically hidden from $\mathbb{A}_\diamond$. The reduction makes the guess $\hat{b} = 0$ (real) if $\mathbb{A}_\diamond$ makes a correct guess and $\hat{b} = 1$ (random) if not.

Let us look at each case separately: for $b = 0$,

$$\begin{aligned}
\Pr\left[\mathsf{Exp}^{\text{ror-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ror}}) \;\middle|\; b = 0\right] &= \Pr\left[\hat{b} = 0 \;\middle|\; b = 0\right] \\
&= \Pr\left[\hat{d}_i = d_i \;\middle|\; b = 0\right] \\
&= \Pr\left[\mathsf{Exp}^{\kappa\text{-ind-cca}\diamond}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_\diamond)\right] ,
\end{aligned}$$

where the final equality holds due to the simulation being completely faithful. For $b = 1$,

$$\begin{aligned}
\Pr\left[\mathsf{Exp}^{\text{ror-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ror}}) \;\middle|\; b = 1\right] &= \Pr\left[\hat{b} = 1 \;\middle|\; b = 1\right] \\
&= \Pr\left[\hat{d}_i \neq d_i \;\middle|\; b = 1\right] = \frac{1}{2} ,
\end{aligned}$$

where the final equality follows from unopened d_i being information-theoretically hidden from $\mathbb{A}_\diamond$ and opened $\hat{d}_i$ being supplanted by a uniform bit by the reduction.

Inserting the distinguishing advantage (Def. 1) and reordering yields the statement. □

```
Experiment $\mathsf{Exp}^{(\kappa,\beta)\text{-ind-cca}\circledast}_{\mathsf{PKE}[\lambda]}(\mathbb{A})$
    $\forall_{j\in[\beta]} b_j \leftarrow\!\!\$ \{0,1\}$
    $\mathsf{pm} \leftarrow\!\!\$ \mathsf{PKE.Pm}(\lambda)$
    $\forall_{i\in[\kappa]}(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow\!\!\$ \mathsf{PKE.Kg}(\mathsf{pm})$
    $(j, \hat{b}_j) \leftarrow\!\!\$ \mathbb{A}^{\mathcal{E},\mathcal{D},(\mathcal{T},)\mathcal{S},\mathcal{R}}(\mathsf{pk}_1, \dots, \mathsf{pk}_\kappa)$
    if $\exists i \in \mathcal{K}_j \cap \mathcal{I} : \hat{b}_j \leftarrow\!\!\$ \{0,1\}$
    if $j \in \mathcal{J} : \hat{b}_j \leftarrow\!\!\$ \{0,1\}$
    return $b_j = \hat{b}_j$
```

```
Oracle $\mathcal{E}(i, j, m_0, m_1)$
    if $|m_0| \neq |m_1|$ : return $\lightning$
    $\mathcal{K}_j \xleftarrow{\cup} i$
    $r \leftarrow\!\!\$ \mathsf{PKE.Rnd}(\mathsf{pm})$
    $c \leftarrow \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m_{b_j}; r)$
    $\mathtt{E} \xleftarrow{\frown} (j, m_{b_j}, r)$
    $\mathcal{C}_i \xleftarrow{\cup} c$
    return $c$

Oracle $\mathcal{D}(i, c)$
    if $c \in \mathcal{C}_i$ : return $\lightning$
    $m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
    return $m$
```

```
Oracle $\mathcal{T}(k)$
    $(j, m, r) \leftarrow \mathtt{E}[k]$
    $\mathcal{J} \xleftarrow{\cup} j$
    return $m$

Oracle $\mathcal{S}(k)$
    $(j, m, r) \leftarrow \mathtt{E}[k]$
    $\mathcal{J} \xleftarrow{\cup} j$
    return $(m, r)$

Oracle $\mathcal{R}(i)$
    $\mathcal{I} \xleftarrow{\cup} i$
    return $\mathsf{sk}_i$
```

Fig. 7. A priori indistinguishability with multiple challenge bits and bi-opening.

Bi-openings. With multiple users and multiple challenge bits, it is possible to model bi-opening in the a priori indistinguishability setting. For full generality, users and challenge bits should be decoupled; otherwise, if each user is allocated a separate challenge bit, it is unclear if the resulting notion is really stronger than single-bit indistinguishability with receiver opening [68]. Thus, the adversary is now free to choose which user to challenge and under which challenge bit the challenge should be constructed. What we end up with is a so-called "free-bit" notion, matching recent formalizations [68, 85] except for the addition of a sender and a transmission opening oracle.

Definition 3. *The* (κ, β)-IND-CCA$\circledast$ *advantage* $\mathsf{Adv}^{(\kappa,\beta)\text{-ind-cca}\circledast}_{\mathsf{PKE}[\lambda]}(\mathbb{A})$ *of an adversary* $\mathbb{A}$ *against public key encryption scheme* $\mathsf{PKE}[\lambda]$ *is the distinguishing advantage against the game* $\mathsf{Exp}^{(\kappa,\beta)\text{-ind-cca}\circledast}_{\mathsf{PKE}[\lambda]}(\mathbb{A})$ *(see Fig. 7).*

The notion is implied by IND-CCA with a $\kappa \cdot \beta$ tightness loss, which follows from it being implied by κ-IND-CCA$\star$ with a β tightness loss as we show next.

Theorem 3. *Let* $\mathsf{PKE}[\lambda]$ *be given. Then there is a type-preserving black-box reduction* $\mathbb{B}_\star$ *such that, for all* $\mathbb{A}_\circledast$*,*

$$\mathsf{Adv}^{(\kappa,\beta)\text{-ind-cca}\circledast}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_\circledast) \leq \beta \cdot \mathsf{Adv}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_\star)\,.$$

The runtime of $\mathbb{B}_\star$ *is upper bounded by that of* $\mathbb{A}_\circledast$ *plus* q_e *encryptions and* q_d *decryptions, where* q_e *and* q_d *are* $\mathbb{A}_\circledast$*'s number of challenge oracle calls and decryption calls respectively, and some small overhead.*

Proof (sketch). The proof is essentially the same as that of Thm. 1, except with multiple users, and it is again useful to split it up in the equivalent of Lemma 1 and Lemma 2: at the outset the reduction, playing $(\kappa, 1)$-IND-CCA$\circledast$, guesses a bit handle j' and simulates (κ, β)-IND-CCA$\circledast$ by forwarding challenge calls using that bit handle to its own challenge oracle; it simulates non-matching oracle calls using the public keys and encrypting honestly. Once again, the simulation is perfect, even in the event that a call to an opening oracle would compromise $b_{j'}$ and, as before, the probability that the guess matches the returned handle is $\Pr[j = j'] = {}^1\!/\!_\beta$, leading to the stated tightness loss. Finally, we can show that a reduction playing κ-IND-CCA$\star$ gains the advantage of any adversary playing $(\kappa, 1)$-IND-CCA$\circledast$ by simply aborting the simulation and outputting a uniform guess in the case that the challenge bit is compromised through opening. □

3.4 A Posteriori Indistinguishability with Selective Opening (ISO)

A posteriori indistinguishability, as described in Sect. 3.2, is defined relative to a message sampler M: rather than choosing challenge messages m_0 and m_1, the adversary is allowed to affect the sampling through input α. During a subsequent challenge phase, the adversary receives either the originally sampled message(s) or a resampled version thereof. The concept lends itself well to modelling opening attacks: the resampling can be refined to conditional resampling, thus ensuring consistency with the opening and

```
Experiment $\mathsf{Exp}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{S}}(\mathbb{A})$
$b \leftarrow\!\!\$ \{0,1\}$
$\mathsf{challenged} \leftarrow \mathsf{false}$
$q \leftarrow 0, s \leftarrow \epsilon$
$\mathsf{pm} \leftarrow\!\!\$ \mathsf{PKE.Pm}(\lambda)$
$\forall_{i\in[\kappa]}(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow\!\!\$ \mathsf{PKE.Kg}(\mathsf{pm})$
$\hat{b} \leftarrow\!\!\$ \mathbb{A}^{\mathcal{E},\mathcal{C},\mathcal{D},(\mathcal{T},)\mathcal{S},\mathcal{R}}(\mathsf{pk}_1,\ldots,\mathsf{pk}_\kappa)$
return $b = \hat{b}$
```

```
Oracle $\mathcal{D}(i,c)$
if $c \in \mathcal{C}_i$ : return $\lightning$
$m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
return $m$
```

```
Oracle $\mathcal{E}(i,\alpha)$
if $\mathsf{challenged}$ : return $\lightning$
$q \leftarrow q+1$
$\mathtt{K} \xleftarrow{\frown} i$
$\mathtt{A} \xleftarrow{\frown} \alpha$
$m \leftarrow\!\!\$ \mathsf{M}_{\langle s\rangle}(\alpha)$
$\mathtt{L} \xleftarrow{\frown} |m|, \mathtt{M}^0 \xleftarrow{\frown} m$
$r \leftarrow\!\!\$ \mathsf{PKE.Rnd}(\mathsf{pm})$
$\mathtt{R} \xleftarrow{\frown} r$
$c \leftarrow \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m;r)$
$\mathcal{C}_i \xleftarrow{\cup} c$
return $c$
```

```
Oracle $\mathcal{C}$
if $\mathsf{challenged}$ : return $\lightning$
$\mathsf{challenged} \leftarrow \mathsf{true}$
for $j \in [q]$
    if $\mathtt{K}[j] \in \mathcal{I}$ : $\mathcal{J} \xleftarrow{\cup} j$
$\mathtt{M}^1 \leftarrow\!\!\$ \mathsf{S}(\mathtt{A},\mathtt{L},\mathcal{J},\mathtt{M}^0[\mathcal{J}])$
return $\mathtt{M}^b$
```

```
Oracle $\mathcal{T}(j)$
if $\mathsf{challenged}$ : return $\lightning$
$\mathcal{J} \xleftarrow{\cup} j$
return $\mathtt{M}^0[j]$
```

```
Oracle $\mathcal{S}(j)$
if $\mathsf{challenged}$ : return $\lightning$
$\mathcal{J} \xleftarrow{\cup} j$
return $(\mathtt{M}^0[j], \mathtt{R}[j])$
```

```
Oracle $\mathcal{R}(i)$
if $\mathsf{challenged}$ : return $\lightning$
$\mathcal{I} \xleftarrow{\cup} i$
return $\mathsf{sk}_i$
```

Fig. 8. A posteriori indistinguishability, also known as indistinguishability SOA, with bi-opening.

avoiding that the challenge bit leaks trivially. Moreover, unlike a priori indistinguishability, the experiment poses no limitations on which ciphertexts may be opened, while simultaneously retaining the preferred single-challenge-bit structure.

When formalizing an ISO notion, the way in which messages get sampled (and resampled) plays an important role and, as we will survey shortly, different abstractions are possible. Our notion of ISO-CCA$\circledast$ uses BY12's idea of a fixed, stateful sampling algorithm M which, on adversarial input α, outputs a single message. We generalize BY12's notion by, in addition to sender and transmission opening, also allowing receiver opening and chosen ciphertext attacks. Furthermore, we make a syntactical distinction between the message sampler M and its resampler S. An adversary $\mathbb{A}$'s advantage (against a given PKE) will be relative to both this message sampler M and resampler S, as made explicit in Def. 4 below. This definition simultaneously serves to define weaker notions such as κ-ISO-CPA$\diamond$, κ-ISO-CCA$\star$, etc., by changing which oracles the adversary has access to.

Definition 4. *The κ-ISO-CCA$\circledast$ advantage $\mathsf{Adv}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{S}}(\mathbb{A})$ of an adversary $\mathbb{A}$ against public key encryption scheme $\mathsf{PKE}[\lambda]$, relative to message sampler M and resampler S, is the distinguishing advantage against the game $\mathsf{Exp}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{S}}(\mathbb{A})$ (see Fig. 8).*

A run of the game proceeds in two stages. In the first stage, the adversary has access to its encryption oracle $\mathcal{E}$, as well as to any of its auxiliary oracles (to open and decrypt). Each encryption query will result in a single challenge ciphertext and the game keeps track of the corresponding encrypted messages (across queries) in the list $\mathtt{M}^0$. The opening oracle(s) will reveal some of $\mathtt{M}^0$, either directly through $\mathcal{T}$ or $\mathcal{S}$ or indirectly through $\mathcal{R}$; the shorthand $\mathtt{M}^0[\mathcal{J}]$, for $(\mathtt{M}^0[j])_{j\in\mathcal{J}}$, indicates the opened messages.

The second stage commences once the adversary calls its challenge oracle $\mathcal{C}$, which blocks access to all oracles apart from $\mathcal{D}$; the flag $\mathsf{challenged}$ enforces the access control. The challenge oracle itself creates a full list of resampled messages $\mathtt{M}^1$ using resampler S and returns either the real or resampled list, depending on the challenge bit.

To avoid trivial wins, $\mathtt{M}^1$ needs to be consistent with $\mathtt{M}^0$ relative to what an adversary trivially learns about the latter: the opening oracles reveal $\mathtt{M}^0[\mathcal{J}]$ (and $\mathcal{J}$); the queries α, collected in $\mathtt{A}$, carry information about the distribution; and we usually assume that ciphertext lengths leak message lengths, collected in the list $\mathtt{L}$. Hence, the resampler S is given the input $(\mathtt{A},\mathtt{L},\mathcal{J},\mathtt{M}^0[\mathcal{J}])$ to facilitate conditional resampling.

Ideally, the resampler S samples exactly from the same distribution as M, conditioned on S's input. Let $\mathring{\mathsf{S}}$ be this ideal, not necessarily efficient, resampler.

Definition 5 (Resampling error). *Let* M *be a stateful sampling algorithm with ideal resampler* $\mathring{\mathsf{S}}$*, and let* S *be a resampling algorithm. Let* $\ell \in \mathbb{Z}_{>0}$ *correspond to the number of* M *calls, and define the support* $\mathrm{Supp}_{\ell,\lambda}(\mathsf{M})$ *as the set of all tuples* $(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$ *subject to* $|\mathtt{A}| = |\mathtt{L}| = \ell$ *that may occur, i.e. for which there exists an adversary* $\mathbb{A}$ *and PKE scheme* PKE *such that the probability that* $\mathsf{Exp}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{S}}(\mathbb{A})$ *results in* $(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$ *being input to* S *is non-zero. For* $(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}]) \in \mathrm{Supp}_{\ell,\lambda}(\mathsf{M})$*, let* $\delta_{\mathring{\mathsf{S}},\mathsf{S}}(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$ *be the statistical distance between* $\mathring{\mathsf{S}}(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$ *and* $\mathsf{S}(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$*. Then the* resampling error *of* S *is*

$$\epsilon^{\ell}_{\mathring{\mathsf{S}},\mathsf{S}}(\lambda) = \max_{(\mathtt{A},\mathtt{L},\mathcal{J},\mathtt{M}^0[\mathcal{J}]) \in \mathrm{Supp}_{\ell,\lambda}(\mathsf{M})} \delta_{\mathring{\mathsf{S}},\mathsf{S}}(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}]) \,.$$

Remark 3. Although BY12 mention the requirement that resampling should result in a distribution statistically close to the ideal conditional resampler, their formalization differs in a number of aspects. Firstly, they define ideal resampling algorithmically with respect to the coins of the original samplers; our approach is more abstract and simply accepts that the relevant conditional distribution is well-defined. Secondly, they define ideal resampling on inputs outside $\mathrm{Supp}_{\ell,\lambda}(\mathsf{M})$ to yield $\perp$ and expect a resampler to behave the same; we do not pose any demands on the resampler in that case (consequently, their resampler must be able to distinguish $\mathrm{Supp}_{\ell,\lambda}(\mathsf{M})$ whereas ours does not). Thirdly, they define the resampling error in terms of a game played by an unbounded adversary, thus hiding the dependency on ℓ. As any statistical distance can be realized as distinguishing advantage by an unbounded adversary—and no unbounded adversary can do any better—their advantage would be more akin to taking the supremum over $\ell \in \mathbb{Z}_{>0}$ of $\epsilon^{\ell}_{\mathring{\mathsf{S}},\mathsf{S}}(\lambda)$. Finally, their distinguishing game randomly samples parameters according to the public key encryption scheme at hand, making their advantage essentially an expectation of the resampling error over the choice of said parameters.

Compared to BY12, our definition of resampling error Def. 5 has the benefit of being easier to work with, by avoiding unbounded adversaries that interact in a resampling experiment, and connecting instead to the intuitive language of statistical distance. Due to the slightly different choices made, somewhat surprisingly the definitions appear technically incomparable; we leave open the task of convincing consolidation of the concept of conditional resamplability, especially in relation to the notion of efficient conditional resamplability (see the "Asymptotic interpretation" paragraph later in this section).

Lemma 3. *Let* $\mathsf{PKE}[\lambda]$ *and* κ*-ISO-CCA*$\circledast$ *adversary* $\mathbb{A}$*, making at most* ℓ $\mathcal{E}$*-queries, be given. Let* M *be a message sampler with ideal resampler* $\mathring{\mathsf{S}}$ *and let* S *be an arbitrary resampler. Then*

$$\left| \mathsf{Adv}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{S}}(\mathbb{A}) - \mathsf{Adv}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}) \right| \leq \epsilon^{\ell}_{\mathring{\mathsf{S}},\mathsf{S}}(\lambda)$$

Proof. The games $\mathsf{Exp}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{S}}(\mathbb{A})$ (using S) and $\mathsf{Adv}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A})$ (using $\mathring{\mathsf{S}}$) are identical if $b = 0$ for both, and, if $b = 1$, they are identical until $\mathbb{A}$ makes its single call to the $\mathcal{C}$ oracle. Moreover, by definition of the resampling error, the statistical distance between $\mathcal{C}$'s output in the two experiments (using S versus $\mathring{\mathsf{S}}$) is at most $\epsilon^{\ell}_{\mathring{\mathsf{S}},\mathsf{S}}(\lambda)$, which therefore bounds the computational distinguishing advantage of any $\mathbb{A}$. □

Alternative samplers. BY12's stateful sampler differs from more common formulations of selective opening attacks, where the adversary can directly specify a stateless, multi-message sampler (or distribution), so $\mathcal{E}$ would return a vector of challenge ciphertexts. For convenience, we include some works from an a posteriori simulation-based perspective in the discussion below, see also Sect. 3.5.

Historically, sender openings and receiver openings had been studied separately, which was reflected in the samplers as well. For sender openings, typically only a single public key is created (so $\kappa = 1$), a vector of ℓ messages is sampled once and subsequently encrypted under this lone public key [10]; in contrast, for receiver openings, there are $\kappa > 1$ public keys and a vector of exactly κ messages is sampled and encrypted one-each under the κ public keys (receivers) of the system [63].

Definitional choices on how messages may be sampled, affect the strength of the resulting notion. There are two main choices on how sampling is modelled, leading to four possible styles of sampler.

The first choice is between stateless and stateful samplers, where stateful sampling allows message dependencies across calls to the $\mathcal{E}$-oracle. Intuitively, stateless sampling allows a hybrid argument to reduce (non-tightly) the number of $\mathcal{E}$-queries to 1 [87], whereas for stateful sampling such a hybrid

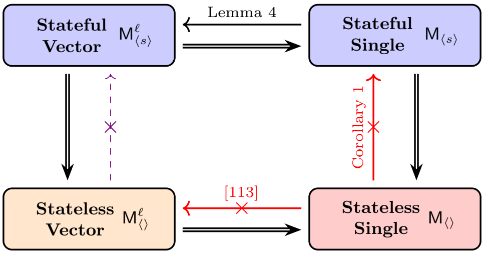

Stateful Vector $\mathsf{M}^\ell_{\langle s\rangle}$
Lemma 4
Stateful Single $\mathsf{M}_{\langle s\rangle}$
Corollary 1
[113]
Stateless Vector $\mathsf{M}^\ell_{\langle\rangle}$
Stateless Single $\mathsf{M}_{\langle\rangle}$

Fig. 9. Relations between notions of SOA employing different styles of samplers. (Double arrows = trivial; purple-dashed = conjectured).

argument looks challenging (and is in fact not possible, see below). We can make the distinction between stateless and stateful samplers explicit by their subscript, writing M respectively $\mathsf{M}_{\langle s\rangle}$.

The second choice asks whether, for each sampling, a single message should be returned, or a message vector of length ℓ. Returning a vector allows for a joint distribution over the messages even when sampling is stateless. We can make the distinction between single-message and vector samplers explicit by their superscript, writing $\mathsf{M}_{\langle s\rangle}$ respectively $\mathsf{M}^\ell_{\langle s\rangle}$. For vector samplers, Fig. 8's challenge oracle $\mathcal{E}$ takes as its first input a list $\mathtt{I}$ of key handles that the adversary wishes to challenge ($\mathcal{E}$ then samples $\ell = |\mathtt{I}|$ messages and encrypts each sampled message under the corresponding public key). Existing vector sample notions often put restrictions on which $\mathtt{I}$ are allowed, for instance each key handle exactly once [63] or an ℓ-fold repetition of a single key handle [113].

The four resulting notions and how they relate are presented in Fig. 9. We will argue that stateful samplers yield potentially stronger notions of security than the more common (stateless) vector sampling.

Our first observation is that for stateful samplers it is irrelevant whether one message is sampled per oracle call or a fixed number ℓ: the two resulting notions are tightly equivalent. Showing that stateful vector sampling is at least as general as stateful single-message sampling is of course trivial when considering vectors of length $\ell = 1$ (and for $\ell > 1$ one can define $\mathsf{M}^\ell_{\langle s\rangle}$ using its first element to encode $\mathsf{M}_{\langle s\rangle}$ and use some fixed, known message for the remaining $\ell - 1$ elements). We show the converse, that stateful single-message sampling is as general as stateful vector sampling, next.

Lemma 4. *Let $\mathsf{PKE}[\lambda]$ be given, let $\mathsf{M}^\ell_{\langle s\rangle}$ be a stateful message sampler (with ideal resampler $\mathring{\mathsf{S}}$) returning ℓ messages per call, and let $\mathsf{M}_{\langle s'\rangle}$ be the stateful message sampler sampling from the same distribution as $\mathsf{M}^\ell_{\langle s\rangle}$ by calling $\mathsf{M}^\ell_{\langle s\rangle}$ initially and then after every ℓth call, yet returning only a single message per call (so for $\ell - 1$ of its calls, $\mathsf{M}_{\langle s'\rangle}$ ignores its input). Let $\mathring{\mathsf{S}}'$ be $\mathsf{M}_{\langle s'\rangle}$'s ideal resampler. Then there is a type-preserving black-box reduction $\mathbb{B}$ such that, for all $\mathbb{A}$,*

$$\mathsf{Adv}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M}^\ell_{\langle s\rangle},\mathring{\mathsf{S}}}(\mathbb{A}) \le \mathsf{Adv}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M}_{\langle s'\rangle},\mathring{\mathsf{S}}'}(\mathbb{B})\,.$$

The runtime of $\mathbb{B}$ is upper bounded by that of $\mathbb{A}$, except that if $\mathbb{A}$ makes q encryption oracle calls, then $\mathbb{B}$ makes $\ell \cdot q$ encryption oracle calls.

Proof (sketch). Whenever $\mathbb{A}$ calls $\mathcal{E}(\mathtt{I}, \alpha)$, $\mathbb{B}$ calls its own encryption oracle ℓ times sequencing through $\mathtt{I}$ for the i inputs and using the same input α throughout, returning the resulting ℓ-length ciphertext vector. This simulation is perfect as any relations between the sampled messages will be preserved using the sampler's internal state. □

As stateful vector-sampling $\mathsf{M}^\ell_{\langle s\rangle}$ trivially implies stateless vector-sampling $\mathsf{M}^\ell_{\langle\rangle}$ (by ignoring the state), as a corollary stateful single-message sampling $\mathsf{M}_{\langle s\rangle}$ implies stateless vector sampling $\mathsf{M}^\ell_{\langle\rangle}$ (top-right to bottom-left in Fig. 9).

In contrast, stateless single-message sampling does not imply vector-sampling, which intuitively follows from a hybrid argument (see Sect. 4.1).

ISO implies IND. We next present a concrete variation of BY12's result that ISO security implies IND security [16, Thm. 4.3]. Specifically, we show that, for a suitably chosen message sampler (see Lemma 5),

κ-ISO-CCA$\star$ implies κ-IND-CCA$\star$ with only a factor 2 loss (the CPA case follows from the reduction's type-preservation). In contrast, BY12 only showed that single-sample κ-IND-CPA$\diamond$ implies single challenge IND-CPA (with a factor 2 loss), thus our result is both tighter for multi-challenge situations and more general by allowing additional oracles.

Conversely, general separations in the form of counterexamples are known (see Sect. 4 for details), indicating that a posteriori indistinguishability is a strictly stronger notion than a priori indistinguishability in the presence of receiver openings.

Lemma 5. *Let $\mathsf{M}_{\langle s\rangle}$ be the sampler that as input α only takes message pairs (m_0, m_1) subject to both $|m_0| = |m_1|$ and $m_0 \neq m_1$. On first invocation (when $s = \varepsilon$), it draws $s \leftarrow\!\!\$ \{0,1\}$ and, on all invocations, on input $\alpha = (m_0, m_1)$, it returns m_s. Let $\mathring{\mathsf{S}}$ be its ideal resampler.*

Consider S that on input $(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$, first checks whether $\mathtt{M}^0[\mathcal{J}]$ is non-empty. If so, it contains at least one opened m_s drawn from (m_0, m_1) satisfying $m_0 \neq m_1$ and S sets $s' \leftarrow s$; otherwise it draws $s' \leftarrow\!\!\$ \{0,1\}$. Finally, S sets and returns $\mathtt{M}^1 \leftarrow \mathtt{A}_{s'}$, where $\mathtt{A}$ is interpreted as $(\mathtt{A}_0, \mathtt{A}_1)$ based on the special form of the α.

Then $\mathsf{S} = \mathring{\mathsf{S}}$.

Proof. By inspection.

Theorem 4. *Let $\mathsf{PKE}[\lambda]$ be given and let $\mathsf{M}_{\langle s\rangle}$ be as given in Lemma 5. Then there is a type-preserving black-box reduction $\mathbb{B}_{\text{iso}}$ such that, for all $\mathbb{A}_{\text{ind}}$,*

$$\mathsf{Adv}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\text{ind}}) \leq 2 \cdot \mathsf{Adv}^{\kappa\text{-iso-cca}\star}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{B}_{\text{iso}}) .$$

The runtime of $\mathbb{B}_{\text{iso}}$ is upper bounded by that of $\mathbb{A}_{\text{ind}}$.

Proof. Without loss of generality, we may assume that $\mathbb{A}_{\text{ind}}$ does not call $\mathcal{E}_{\mathbb{A}}(i, m_0, m_1)$ with either $|m_0| \neq |m_1|$ or $m_0 = m_1$, nor does it corrupt and challenge on the same key. Technically, one can create an intermediate reduction $\mathbb{B}_{\text{ind}}$ that runs $\mathbb{A}_{\text{ind}}$ and, playing the same game, forwards everything to its own oracles but those pointless $\mathcal{E}$ calls, which it can easily simulate by responding with $\lightning$ and $\mathsf{PKE.Enc}(m_0)$, respectively; if $\mathbb{A}_{\text{ind}}$ makes a call that would trigger $\mathcal{K} \cap \mathcal{I} \neq \emptyset$ then $\mathbb{B}_{\text{ind}}$ samples $\hat{b} \leftarrow\!\!\$ \{0,1\}$ and terminates with that bit. By inspection, $\mathbb{B}_{\text{ind}}$'s advantage is at least $\mathbb{A}_{\text{ind}}$'s (it could be larger for instance when $\mathbb{A}_{\text{ind}}$ correctly guesses the bit, while having both corrupted a key and challenged it with $m_0 = m_1$).

Let $\mathbb{B}_{\text{iso}}$ be such that if $\mathbb{A}_{\text{ind}}$ calls $\mathcal{E}_{\mathbb{A}}(i, m_0, m_1)$ subject to both $|m_0| = |m_1|$ and $m_0 \neq m_1$, it sets $\alpha = (m_0, m_1)$ and calls $\mathcal{E}_{\mathbb{B}}(i, \alpha)$, returning the resulting c; if $\mathbb{A}_{\text{ind}}$ calls any other oracles, $\mathbb{B}_{\text{iso}}$ forwards the call and returns the result. When $\mathbb{A}_{\text{ind}}$ halts with a guess $\hat{s}$, $\mathbb{B}_{\text{iso}}$ calls $\mathcal{C}$ and receives $\mathtt{M}^b$. Since $\mathbb{B}_{\text{iso}}$ can maintain its own perfect copy of $\mathtt{A}$, it can check whether $\mathtt{M}^b = \mathtt{A}_{\hat{s}}$. If so, $\mathbb{B}_{\text{iso}}$ halts with output $\hat{b} = 0$ (indicating a guess that the returned messages were the real ones), otherwise $\mathbb{B}_{\text{iso}}$ halts with output $\hat{b} = 1$.

We can rephrase $\mathbb{B}_{\text{iso}}$'s distinguishing advantage

$$\mathsf{Adv}^{\kappa\text{-iso-cca}\star}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{S}}(\mathbb{B}_{\text{iso}}) = \Pr\Big[\hat{b} = 0 \;\Big|\; b = 0\Big] - \Pr\Big[\hat{b} = 0 \;\Big|\; b = 1\Big]$$

and analyse each term individually. Based on $\mathbb{B}_{\text{iso}}$'s description, the event $\hat{b} = 0$ is equivalent to the event $\mathtt{M}^b = \mathtt{A}_{\hat{s}}$.

If $b = 0$, then $\mathtt{M}^0 = \mathtt{A}_s$, so the first term is equivalent to $\Pr[\mathtt{A}_{\hat{s}} = \mathtt{A}_s \mid b = 0]$. Given that $\mathtt{A}_0 \neq \mathtt{A}_1$ (by our assumption on $\mathbb{A}_{\text{ind}}$ not making queries with $m_0 = m_1$) we can simplify further to $\Pr[\hat{s} = s \mid b = 0]$. At this point, the conditional $b = 0$ becomes irrelevant as it is independent of both $\hat{s}$ and s (jointly). Finally, $\Pr[\hat{s} = s]$ equals $\Pr\Big[\mathsf{Exp}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\text{ind}})\Big]$ as, by design of $\mathbb{B}_{\text{iso}}$ and M, $\mathbb{A}_{\text{ind}}$ is provided with an environment that perfectly matches that of $\mathsf{Exp}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\cdot)$, where M's bit s plays the role of the bit $\mathbb{A}_{\text{ind}}$ has to guess. Thus,

$$\Pr\Big[\hat{b} = 0 \;\Big|\; b = 0\Big] = \Pr\Big[\mathsf{Exp}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\text{ind}})\Big] .$$

If $b = 1$, then $\mathtt{M}^1 = \mathtt{A}_{s'}$, so the second term is equivalent to $\Pr[\mathtt{A}_{\hat{s}} = \mathtt{A}_{s'} \mid b = 1]$, which simplifies to $\Pr[\hat{s} = s' \mid b = 1]$. As we assumed $\mathbb{A}_{\text{ind}}$ maintained $\mathcal{K} \cap \mathcal{I} = \emptyset$ as invariant (for its game), it follows that, for $\mathbb{B}_{\text{iso}}$'s game, $\mathcal{J} = \emptyset$ and hence $\mathring{\mathsf{S}}$ will have drawns s' uniformly at random, independently of $\hat{s}$. Thus,

$$\Pr\Big[\hat{b} = 0 \;\Big|\; b = 1\Big] = \frac{1}{2} .$$

Putting the pieces together, we obtain

$$2 \cdot \mathsf{Adv}^{\kappa\text{-iso-cca}\star}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{S}}(\mathbb{B}_{\text{iso}}) = 2 \cdot \Pr\Big[\mathsf{Exp}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\text{ind}})\Big] - 1 \geq \mathsf{Adv}^{\kappa\text{-ind-cca}\star}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\text{ind}}),$$

where the final inequality takes into account the intermediate $\mathbb{B}_{\text{ind}}$ reduction to justify our assumption on $\mathbb{A}_{\text{ind}}$'s behaviour (from the beginning of the proof).

□

Asymptotic interpretation. The advantage specified in Def. 4 leads to two potential asymptotic interpretations: weak ISO and "full" ISO. Full ISO security is achieved for a scheme PKE if, for all PPT $\mathbb{A}$ and all PPT M, the advantage $\mathsf{Adv}^{\text{iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A})$ is negligible in λ (where of course security for CPA and other openings are defined analogously). Importantly, the quantification over the message samplers M is irrespective of how efficient $\mathring{\mathsf{S}}$ might be implemented. For weak ISO security, the class of samplers is restricted to those for which there exists an efficient resampler S, that is a PPT S such that for all polynomials poly the maximum resampling error $\max_{\ell \leq \mathsf{poly}(\lambda)} \epsilon^{\ell}_{\mathring{\mathsf{S}},\mathsf{S}}(\lambda)$ is negligible (in λ).

Clearly, restricting the class of samplers M as for weak ISO results in a significantly weaker notion, as further explored by Böhl et al. [19]. There are in fact further gradations in the security notion depending on how "efficient resampling" is formalized. We already explained the subtle differences between our formalization of the sampling error and BY12's. Other works [19, 63] require the efficient resampler to have zero statistical distance; such a stricter requirement on the resampler leads to a potentially weaker notion of security.

How ISO is formalized asymptotically (full versus weak) affects how the alternative types of sampling relate to each other, specifically when interpreting Lemma 4. For Full ISO, an almost immediate consequence of Lemma 4 is that stateful vector sampling and stateful single-message sampling are equivalent: if $\mathsf{M}^{\ell}_{\langle s \rangle}$ is efficient (PPT), then so is the derived $\mathsf{M}_{\langle s' \rangle}$ (and vice versa), without having to worry about the efficiency of the ideal resampler. However, for weak ISO, resampling efficiency comes into play and even if $\mathsf{M}^{\ell}_{\langle s \rangle}$'s ideal resampler $\mathring{\mathsf{S}}$ has an efficient resampler S, there is no guaranteee that the same holds for $\mathsf{M}_{\langle s' \rangle}$'s ideal resampler $\mathring{\mathsf{S}}'$. The main technical difficulty here is that $\mathring{\mathsf{S}}'$ also needs to deal with message vectors that are not a multiple of ℓ messages and it is possible that "partial" vectors are harder to resample efficiently (for instance, if the lengths of the missing messages might assist the resampling).

We conclude that, in an asymptotic setting, additional definitional choices increase the number of possible notions and, lacking clear use cases, determining which notion makes most sense is tricky. For instance, Full-ISO has mostly fallen out of favour, as allowing inefficient resampling seems to yield an artificially strong notion of security, with currently no known schemes achieving it (see also Sect. 4.5).

Luckily, even the weakest version of weak ISO allows us to conclude from Thm. 4 that ISO security implies IND security. Both the message sampler and its ideal resampler used in that theorem (as specified in Lemma 5) are clearly efficient and hence included in any reasonably class of message samplers (used to define a flavour of weak ISO security).

Further remarks. ISO-CPA was initially defined relative to message distributions independent of the public key [10]. Böhl et al. [19] noted that as a result, the notion did not imply IND-CPA; with the converse being open, the notions seemed incomparable. Allowing message samplers an input α resolves this issue [16].

Given that (κ, β)-IND-CCA⊛ is implied by κ-IND-CCA⋆ with a β loss (Thm. 3), we may conclude that κ-ISO-CCA⋆ implies (κ, β)-IND-CCA⊛ with a $2 \cdot \beta$ security loss. In fact, a tighter reduction that loses only a factor 2 is possible by starting from κ-ISO-CCA⊛ (instead of κ-ISO-CCA⋆ as in Thm. 4), see Thm. 8 (App. A) for details.

3.5 A Posteriori Simulatability with Selective Opening (SSO)

As explained in Sect. 3.2, the main idea of a posteriori simulatability is to have a simulator Sim simulate the computations of $\mathbb{A}$ without seeing the ciphertexts, thus capturing the idea that the ciphertexts leak nothing about the plaintexts. Unlike the ISO notion from the previous section, for SSOthe relevant security game does not contain a conditional resampling phase, making the notion suitable when such resampling is problematic. A potential downside is that the presence of simulators and distinguishers can complicate reductions compared to indistinguishability-based alternatives.

```
Experiment $\mathsf{Exp}^{\kappa\text{-sso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}}(\mathbb{A},\mathbb{D})$
$b \xleftarrow{\$} \{0,1\}$
$s \leftarrow \varepsilon$
$\mathsf{pm} \xleftarrow{\$} \mathsf{PKE.Pm}(\lambda)$
$\forall_{i\in[\kappa]}(\mathsf{pk}_i,\mathsf{sk}_i) \xleftarrow{\$} \mathsf{PKE.Kg}(\mathsf{pm})$
if $b = 0$ :
    $\mathsf{out} \xleftarrow{\$} \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{S},\mathcal{R},\mathcal{T}}(\mathsf{pk}_1,\ldots,\mathsf{pk}_\kappa)$
    for $j \in [|\mathtt{K}|]$
        if $\mathtt{K}[j] \in \mathcal{I} : \mathcal{J} \xleftarrow{\cup} j$
else :
    $\mathsf{out} \xleftarrow{\$} \mathsf{Sim}^{\mathcal{E},\mathcal{T}}(\mathsf{pm})$
$\hat{b} \xleftarrow{\$} \mathbb{D}(\mathsf{pm},\mathtt{A},\mathtt{M},\mathcal{J},\mathsf{out})$
return $b = \hat{b}$
```

```
Oracle $\mathcal{E}(i,\alpha)$
$\mathtt{K} \xleftarrow{\frown} i, \mathtt{A} \xleftarrow{\frown} \alpha$
$m \xleftarrow{\$} \mathsf{M}_{\langle s\rangle}(\alpha)$
$\mathtt{M} \xleftarrow{\frown} m$
$r \xleftarrow{\$} \mathsf{PKE.Rnd}(\mathsf{pm})$
$c \leftarrow \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m;r)$
$\mathtt{R} \xleftarrow{\frown} r, \mathcal{C}_i \xleftarrow{\cup} c$
if $b = 0$ : return $c$
else : return $|m|$
```

```
Oracle $\mathcal{D}(i,c)$
if $c \in \mathcal{C}_i$ : return $\lightning$
$m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
return $m$
```

```
Oracle $\mathcal{T}(j)$
$\mathcal{J} \xleftarrow{\cup} j$
return $\mathtt{M}[j]$
```

```
Oracle $\mathcal{S}(j)$
$\mathcal{J} \xleftarrow{\cup} j$
return $(\mathtt{M}[j],\mathtt{R}[j])$
```

```
Oracle $\mathcal{R}(i)$
$\mathcal{I} \xleftarrow{\cup} i$
return $\mathsf{sk}_i$
```

Fig. 10. κ-SSO-CCA⊛ security game, for which a distinguisher $\mathbb{D}$ is tasked with guessing whether it received the view of adversary $\mathbb{A}$ playing the real game or a simulated view (by Sim).

In the CPA setting, with either sender or receiver opening SSO-CPA is known to be strictly stronger (and therefore harder to achieve) than ISO-CPA [63], and so we place a posteriori simulatability above a priori indistinguishability in our hierarchy (Fig. 1). With only transmission openings present, the notion is tightly implied by a priori indistinguishability [16], see Sect. 4.2.

On the other hand, in the CCA setting the relationship between a posteriori simulatability and a posteriori indistinguishability remains largely open: in particular, while we conjecture that SSO-CCA implies ISO-CCA with any (matching) openings (see Fig. 16), a proof thereof has to the best of our knowledge yet to appear, see Open Problem 5.

Our formalization (Fig. 10) is based on BY12's SSO-CPA⊙ notion, where we added multiple users, receiver and thus bi-openings, as well as a CCA oracle. The joint advantage of adversary $\mathbb{A}$ and distinguisher $\mathbb{D}$ is therefore relative to the stateful, single-message sampler M (see Sect. 3.4) as well as the simulator Sim. (The equivalent of Lemma 4, extending to vector samplers, holds also in the current SSO setting.)

Definition 6. *The κ-SSO-CCA⊛ advantage $\mathsf{Adv}^{\kappa\text{-sso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}}(\mathbb{A},\mathbb{D})$ of an adversary $\mathbb{A}$ and distinguisher $\mathbb{D}$ against public key encryption scheme $\mathsf{PKE}[\lambda]$, relative to message sampler M and simulator Sim, is the distinguishing advantage against the game $\mathsf{Exp}^{\kappa\text{-sso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}}(\mathbb{A},\mathbb{D})$ (see Fig. 10).*

And so our formalization of SSO comes with three players: the adversary $\mathbb{A}$, the simulator Sim, and the distinguisher $\mathbb{D}$. In the real game ($b = 0$), $\mathbb{A}$ gets access to the public keys and all oracles, and the encryption oracle $\mathcal{E}$ returns encryptions of the sampled messages. Once the adversary is content, it halts with some output out. Intuitively, $\mathbb{A}$'s goal is to make it as easy as possible for $\mathbb{D}$ to guess correctly, and without loss of generality $\mathbb{A}$ will simply output its view, i.e. the transcript of its interactions with the game as well as its internal randomness.

In the ideal game ($b = 1$), the game instead calls Sim who does not get access to the ciphertexts, and whose goal is to fool the distinguisher, and so Sim will want to do everything in its power to make out look like it originated from an adversary who did have access to the real ciphertexts. Since we usually assume that ciphertexts leak message lengths, the simulator does receive message lengths (in place of ciphertexts) to facilitate its job; additionally, it can open individual messages through the transmission opening oracle. The sender and receiver oracles are not present as, in the ideal game, there are no keys to be opened, nor is there any randomness sampled by the encryption oracle.

Eventually, $\mathbb{D}$ makes a decision on whether out was produced by someone with access to the real ciphertexts and opening oracles or not, halting with a guess $\hat{b} = 0$ for "real", or $\hat{b} = 1$ for "ideal". Crucially, the distinguisher receives all the sampled messages $\mathtt{M}$ directly from the experiment, in addition to the real parameters pm, sampler inputs $\mathtt{A}$ and list of opened challenges $\mathcal{J}$.

Remark 4. The strength of the notion is governed by which of these additional inputs $\mathbb{D}$ receives directly from the game, as those inputs effectively bind the simulator to be honest. For instance, denying the

```
Experiment $\mathsf{Exp}^{\kappa\text{-sso}'\text{-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}}(\mathbb{A},\mathbb{D})$
$b \leftarrow\$ \{0,1\}$
$s \leftarrow \varepsilon$
$\mathsf{pm} \leftarrow\$ \mathsf{PKE.Pm}(\lambda)$
$\forall_{i\in[\kappa]}(\mathsf{pk}_i,\mathsf{sk}_i) \leftarrow\$ \mathsf{PKE.Kg}(\mathsf{pm})$
$\textbf{if } b = 0:$
    $\mathsf{out} \leftarrow\$ \mathbb{A}^{\mathcal{E},\mathcal{D},\mathcal{S},\mathcal{R},\mathcal{T}}(\mathsf{pk}_1,\ldots,\mathsf{pk}_\kappa)$
$\textbf{else}\ :$
    $\mathsf{out} \leftarrow\$ \mathsf{Sim}^{\mathcal{E},\mathcal{R}',\mathcal{T}}(\mathsf{pm})$
$\hat{b} \leftarrow\$ \mathbb{D}(\mathsf{pm},\mathtt{K},\mathtt{A},\mathtt{M},\mathcal{I},\mathcal{J},\mathsf{out})$
$\textbf{return } b = \hat{b}$
```

```
Oracle $\mathcal{E}(i,\alpha)$
$\mathtt{K} \xleftarrow{\frown} i, \mathtt{A} \xleftarrow{\frown} \alpha$
$m \leftarrow\$ \mathsf{M}_{\langle s\rangle}(\alpha)$
$\mathtt{M} \xleftarrow{\frown} m$
$r \leftarrow\$ \mathsf{PKE.Rnd}(\mathsf{pm})$
$c \leftarrow \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m;r)$
$\mathtt{R} \xleftarrow{\frown} r, \mathcal{C}_i \xleftarrow{\cup} c$
$\textbf{if } b = 0 : \textbf{return } c$
$\textbf{else if } i \in \mathcal{I} : \textbf{return } m$
$\textbf{else}\ : \textbf{return } |m|$
```

```
Oracle $\mathcal{D}(i,c)$
$\textbf{if } c \in \mathcal{C}_i : \textbf{return } \lightning$
$m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
$\textbf{return } m$
```

```
Oracle $\mathcal{T}(j)$
$\mathcal{J} \xleftarrow{\cup} j$
$\textbf{return } \mathtt{M}[j]$
```

```
Oracle $\mathcal{S}(j)$
$\mathcal{J} \xleftarrow{\cup} j$
$\textbf{return } (\mathtt{M}[j],\mathtt{R}[j])$
```

```
Oracle $\mathcal{R}(i)$
$\mathcal{I} \xleftarrow{\cup} i$
$\textbf{return } \mathsf{sk}_i$
```

```
Oracle $\mathcal{R}'(i)$
$\mathcal{I} \xleftarrow{\cup} i$
$\textbf{for } j \in [|\mathtt{K}|]$
    $\textbf{if } \mathtt{K}[j] = i : \mathtt{L} \xleftarrow{\frown} \mathtt{M}[j]$
$\textbf{return } \mathtt{L}$
```

Fig. 11. An alternative κ-SSO$'$-CCA$\circledast$ security game, where the simulator Sim's behaviour is bound by additional direct inputs (by the game) to the distinguisher $\mathbb{D}$. The simulator has access to its $\mathcal{T}$ oracle whenever the adversary has access to $\mathcal{S}$ or $\mathcal{T}$ (so the $\diamond$, $\odot$, and $\circledast$ notions) and to $\mathcal{R}'$ whenever $\mathbb{A}$ has access to $\mathcal{R}$ (so the $\star$ and $\circledast$ notions).

distinguisher access to the list $\mathcal{J}$ of opened challenges yields a vacuous notion: a simulator could run a copy of the experiment with $\mathbb{A}$ and, whenever $\mathbb{A}$ makes an $\mathcal{E}$-query, Sim would call its own $\mathcal{E}$-oracle, immediately open that challenge using $\mathcal{T}$ to receive the underlying message, and then simply encrypt that message to obtain a ciphertext to return to $\mathbb{A}$ (and eventually Sim uses $\mathbb{A}$'s out as output). Conditioned on only pm, A, and M, this simulator's out will be identically distributed to that of a real adversary.

For receiver openings, our mechanism only provides the distinguisher with the indices of the messages that were opened as a logical consequence of revealing private keys. Instead, one could provide the distinguisher directly with the list $\mathcal{I}$ of the opened keys [63], plus the information (K) needed to identify which ciphertexts were encrypted under which key (for past single-shot, stateless vector sampling formalizations, restrictions on $\mathcal{E}$ typically made K superfluous). In the case of bi-openings, one would then provide both the list $\mathcal{I}$ and $\mathcal{J}$ to an adversary [91]. For completeness, we have included the alternative notion κ-SSO$'$-CCA$\circledast$ in Fig. 11, that captures this finer-grained mechanism in the context of adaptive, stateful sampling.

Considering sender openings only, Bellare, Hofheinz and Yilek [10] opt for another mechanism instead: their advantage statement is parameterized by the number of openings allowed and both the adversary and simulator are restricted to making at most that many openings (the restriction on the number of openings made by the simulator is not made explicit in the published versions [10, 69], but follows from one of the full versions [14]). Inspired by this mechanism, one could in our formalism replace the distinguisher's input $\mathcal{J}$ by only the cardinality $|\mathcal{J}|$ of said list; effectively, it allows the simulator a bit more freedom to deviate from what an adversary is doing, but not enough to render the notion vacuous as above.

Not providing $\mathbb{D}$ with the parameters pm gives an alternative, weaker notion of SSO [113]: since the simulator is now free to produce the parameters itself, it opens for strategies that e.g. involve inserting trapdoors in pm. Conversely, providing the public keys pk_i to the distinguisher takes away the ability for a simulator to use 'fake' keys in its output out; in that case, for the notion to make sense, Sim would have to be provided the pk_i as input, as well as oracle access to $\mathcal{R}$, furthermore the list $\mathcal{I}$ of corrupted parties would then be a secondary additional input to the distinguisher. We let our notion of a posteriori simulatability be one in which $\mathbb{D}$, and thus also Sim, are given pm (matching BY12), but not the pk_i.

Remark 5. A further weakening of SSO$\circledast$ (and SSO$\star$) is possible by restricting adversarial access to the challenge oracle $\mathcal{E}$ on already corrupted key handles, by adding a line to the top of $\mathcal{E}$ that, whenever $i \in \mathcal{I}$, immediately return $\lightning$; we denote this notion by SSO*. Past definitions of SSO with receiver openings often implicitly included this restriction by virtue of being staged and using stateless vector sampling: for

```
Distinguisher $\mathbb{D}_{\text{sso}}(\mathtt{A}, \mathtt{M}, \mathcal{J}, \mathsf{out})$
$\mathtt{M}^0 \leftarrow \mathtt{M},\ \forall_{i \in |\mathtt{M}|} \mathtt{L}[i] \leftarrow |\mathtt{M}[i]|$
$\mathtt{M}^1 \leftarrow\!\!\$\ \mathring{\mathsf{S}}(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$
$d \leftarrow\!\!\$\ \{0,1\}$
$\mathsf{state} \leftarrow \mathsf{out}$
Run $\mathbb{A}^2_{\text{iso}}(\mathtt{M}^d)$ using $\mathsf{state}$
if $\mathbb{A}^2_{\text{iso}}$ terminates within time $T$ with output $\hat{d}$
  $\hat{b} \leftarrow \neg(d = \hat{d})$
else
  $\hat{b} \leftarrow 1$
return $\hat{b}$
```

Fig. 12. The distinguisher $\mathbb{D}_{\text{sso}}$ of Thm. 5.

instance, an adversary would have a single shot to receive a vector of challenge ciphertexts after which it could non-adaptively corrupt a set of keys [63].

There is no obvious reason for such a restriction, although intuitively challenging on an already corrupted key handle seems of little benefit to an adversary as the messages involved are not considered confidential: the call's corresponding index j is guaranteed to end up in $\mathcal{J}$, so a simulator Sim can access the message as well. Yet, the newly sampled message can depend on the sampler's state, and corrupting a private key possibly allows an adversary to trigger a subsequent call to the sampler that reveals information about its state relating to past, unopened messages.

Curiously, for NCE we will soon see (Def. 7) that the restriction is inevitable and, as a consequence, we can only show that NCE implies this weaker, restricted version of SSO.

Open Problem 4. *How do notions of* SSO, SSO$'$, *and* SSO* *relate?*

SSO implies ISO. A posteriori simulatability with selective opening implies a posteriori indistinguishability with selective opening in the CPA-setting, as shown by BY12 for transmission and sender openings [16, Theorem 3.3]. We next provide a concrete, updated statement to include receiver and bi-openings. Our proof corrects a subtle mistake in BY12's original, as explained inline.

Theorem 5. *Let* $\mathsf{PKE}[\lambda]$ *be given, and let* M *be a sampler with ideal resampler* $\mathring{\mathsf{S}}$*. Then, for any adversary* $\mathbb{A}_{\text{iso}}$ *playing* $\mathsf{Exp}^{\kappa\text{-iso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\cdot)$ *and making at most q challenge queries, there exist (non black-box) type-preserving reduction* $\mathbb{B}_{\text{sso}}$ *and distinguisher* $\mathbb{D}_{\text{sso}}$ *such that, for all simulators* Sim*,*

$$\mathsf{Adv}^{\kappa\text{-iso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}_{\text{iso}}) \leq 2 \cdot \mathsf{Adv}^{\kappa\text{-sso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}}(\mathbb{B}_{\text{sso}}, \mathbb{D}_{\text{sso}}),$$

The combined runtime of $\mathbb{B}_{\text{sso}}$ *and* $\mathbb{D}_{\text{sso}}$ *is upper bounded by twice that of* $\mathbb{A}_{\text{iso}}$ *plus that of one call to* $\mathring{\mathsf{S}}$ *and some small overhead.*

Proof. Let T be an upper bound on the time $\mathbb{A}_{\text{iso}}$ can take in the experiment $\mathsf{Exp}^{\kappa\text{-iso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\cdot)$. Without loss of generality, we assume $\mathbb{A}_{\text{iso}}$ always calls $\mathcal{C}$ exactly once, so we can split $\mathbb{A}_{\text{iso}}$ in two: let $\mathbb{A}^1_{\text{iso}}$ be $\mathbb{A}_{\text{iso}}$ up until the point it calls $\mathcal{C}$ and denote with state its state at that point; furthermore, let $\mathbb{A}^2_{\text{iso}}$ be $\mathbb{A}_{\text{iso}}$ after it receives the response $\mathtt{M}^b$ of calling $\mathcal{C}$, continuing from state state.

Define $\mathbb{B}_{\text{sso}}$ as running $\mathbb{A}^1_{\text{iso}}$, forwarding all oracles appropriately and, once $\mathbb{A}^1_{\text{iso}}$ terminates (by making its $\mathcal{C}$ call) with state state, $\mathbb{B}_{\text{sso}}$ sets $\mathsf{out} \leftarrow \mathsf{state}$ and terminates with output out.

The distinguisher (Fig. 12) receives the real message vector $\mathtt{M}^0$ as its input and uses the ideal resampling algorithm $\mathring{\mathsf{S}}$ to produce a resampled message vector $\mathtt{M}^1$, draws a bit d, and, depending on its value, runs $\mathbb{A}^2_{\text{iso}}$ on either the real or resampled message vector, using its own input out as $\mathbb{A}^2_{\text{iso}}$'s initial state. As there is no a priori guarantee that the distinguisher's input out is a valid state (namely one that could have been generated by $\mathbb{A}^1_{\text{iso}}$), the runtime of $\mathbb{A}^2_{\text{iso}}$ on $\mathsf{state} = \mathsf{out}$ is not bound by that of $\mathbb{A}_{\text{iso}}$ itself, which is why the distinguisher checks $\mathbb{A}^2_{\text{iso}}$'s runtime explicitly. (BY12, using an asymptotic framework, implicitly and erroneously assume that $\mathbb{A}^2_{\text{iso}}$ inherits $\mathbb{A}_{\text{iso}}$'s polynomial runtime, even when run on simulated states.) If $\mathbb{A}^2_{\text{iso}}$ does finish in time then $\mathbb{D}_{\text{sso}}$'s output depends on whether $\mathbb{A}^2_{\text{iso}}$ guessed d correctly or not.

Let b denote the challenge bit of the $\mathsf{Exp}^{\kappa\text{-sso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}}(\cdot,\cdot)$ game, then $\mathbb{B}_{\text{sso}}$ and $\mathbb{D}_{\text{sso}}$ win with the following probability.

$$\Pr\left[\mathsf{Exp}^{\kappa\text{-sso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}}(\mathbb{B}_{\text{sso}},\mathbb{D}_{\text{sso}})\right] = \frac{1}{2}\left(\Pr\left[b=\hat{b}\;\middle|\; b=0\right] + \Pr\left[b=\hat{b}\;\middle|\; b=1\right]\right) .$$

For the first term, $\mathbb{B}_{\text{sso}}$ and $\mathbb{D}_{\text{sso}}$ essentially run $\mathbb{A}_{\text{iso}}$ start to finish with $\mathbb{A}^1_{\text{iso}}$'s finishing state equaling $\mathbb{A}^2_{\text{iso}}$ starting state. Thus, $\mathbb{D}_{\text{sso}}$ will never have to time-cap $\mathbb{A}^2_{\text{iso}}$ and (with some logic deciphering) $\Pr\left[b=\hat{b}\;\middle|\; b=0\right] = \Pr\left[d=\hat{d}\;\middle|\; b=0\right]$. Moreover, $\mathbb{A}_{\text{iso}}$ is given a faithful simulation of $\mathsf{Exp}^{\kappa\text{-iso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\cdot)$, thus $\Pr\left[d=\hat{d}\;\middle|\; b=0\right] = \Pr\left[\mathsf{Exp}^{\kappa\text{-iso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}_{\text{iso}})\right]$.

For the second term, let bad_T denote the event that the distinguisher time-caps $\mathbb{A}^2_{\text{iso}}$ and hence sets $\hat{b} \leftarrow 1$. Then we can rewrite

$$\Pr\left[b=\hat{b}\;\middle|\; b=1\right] = \Pr[\mathsf{bad}_T] + (1-\Pr[\mathsf{bad}_T])\Pr\left[d\neq\hat{d}\;\middle|\; b=1\wedge\neg\mathsf{bad}_T\right] ,$$

where we also used that $\Pr[\mathsf{bad}_T \mid b=0]=0$. For the final conditional probability, out and hence state was produced by Sim without access to unopened messages, so that the challenge bit d is information-theoretically hidden from $\mathbb{A}^2_{\text{iso}}$ and the probability that $d\neq\hat{d}$ equals a half.

Finally, collecting the pieces and some algebraic manipulation yields the theorem statement:

$$\begin{aligned}
2\cdot\mathsf{Adv}^{\kappa\text{-sso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}}(\mathbb{B}_{\text{sso}},\mathbb{D}_{\text{sso}}) &= 2\cdot\Pr\left[b=\hat{b}\;\middle|\; b=0\right] + 2\cdot\Pr\left[b=\hat{b}\;\middle|\; b=1\right] - 2\\
&= \left(2\cdot\Pr\left[\mathsf{Exp}^{\kappa\text{-iso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}_{\text{iso}})\right]-1\right)\\
&\quad + 2\cdot\Pr[\mathsf{bad}_T] + 2\cdot(1-\Pr[\mathsf{bad}_T])\cdot\frac{1}{2} - 1\\
&= \mathsf{Adv}^{\kappa\text{-iso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}_{\text{iso}}) + \Pr[\mathsf{bad}_T]\\
&\geq \mathsf{Adv}^{\kappa\text{-iso-cpa}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}_{\text{iso}}) .
\end{aligned}$$

□

The proof of Thm. 5 does not carry over to the CCA setting, as the distinguisher $\mathbb{D}_{\text{sso}}$ would have to provide a simulation of the decryption oracle $\mathcal{D}$ to $\mathbb{A}^2_{\text{iso}}$ without access to the private keys or any oracles itself (indeed, already for weaker notions like PCA, CVA, etc., the proof breaks down). Whether SSO-CCA implies ISO-CCA with either kind of opening remains open, see Open Problem 5.

Open Problem 5. *Does* SSO-CCA *imply* ISO-CCA *in the presence of sender, receiver, or transmission opening?*

Asymptotic interpretation. As we are primarily concerned with concrete security, our security definition is indifferent as to whether the distinguisher may depend on the simulator or vice versa. When moving to asymptotic security, the choice of quantifiers leads to a weak and strong version of SSO security. A given scheme PKE is deemed weak SSO secure (for either opening, CPA or CCA) if, for all PPT $\mathbb{A}$, PPT M, and PPT $\mathbb{D}$ there exists a PPT simulator Sim such that the relevant advantage is negligible in λ. Alternatively, PKE is deemed to be strong SSO secure (again, for either opening, CPA or CCA) if for every PPT $\mathbb{A}$ and PPT M there is a PPT simulator Sim such that all PPT distinguishers $\mathbb{D}$ have negligible distinguishing advantage. Both options have appeared in the literature, for instance BY12 opted for the weaker variant, whereas more recent work went for the stronger one [91].

Similar to ISO, the mechanism on how to sample can affect the definition of SSO security (whether weak or strong). The equivalent of Lemma 4 holds for SSO, but without any need for ideal resamplers. Consequently, for both weak and strong SSO security, stateful (fixed-length) vector samplers and stateful single-message samplers are equivalent.

For an asymptotic interpretation of Thm. 5, we observe that the runtime of the derived distinguisher $\mathbb{D}$ includes that of the ideal sampler $\mathring{\mathsf{S}}$, thus this $\mathbb{D}$ might not be efficient. To ensure $\mathbb{D}$ is efficient, we could replace its call to $\mathring{\mathsf{S}}$ with that of an efficient resampler S and rely on Lemma 3 to argue that this change only affects negligible change in the distinguisher's advantage.

SSO⋆ is unachievable (without programming). Our formalization of SSO is multi-challenge, allowing non-trivial relations between sampled messages (through stateful or vector sampling, see Sect. 3.4). Yang et al. [113] showed κ-SSO-CPA⋆ to be unachievable in the non-programmable random oracle model, in the sense that private keys would have to be at least as long as the total number of plaintext bits to be encrypted [113, Thm. 3.1]. Thus, κ-SSO-CPA⊛, κ-SSO-CCA⋆, and κ-SSO-CCA⊛ must all be similarly unachievable. This mirrors Nielsen's earlier impossibility of non-interactive NCE in the non-programmable oracle model [100] (see Sect. 3.6).

Intuitively, the impossibility works as follows: the adversary $\mathbb{A}$, who wants to help distinguisher $\mathbb{D}$, commits to the public keys and challenge ciphertexts using the (non-programmable) random oracle. Then, it communicates the commitment, i.e. the digest of the random oracle, to $\mathbb{D}$, through the set of opened key handles, $\mathcal{I}$ (as $\mathcal{I}$ is given to $\mathbb{D}$ in their experiment, cf. Fig. 11). This is done by opening key number i iff the digest at position i has bit value 1. Then, $\mathbb{D}$ can recompute the commitment and check that it matches the form of $\mathcal{I}$. This strategy is hard to simulate, as it would require a simulator to commit to the ciphertexts before opening the corresponding messages. With the ability to program the random oracle, the commitment can be delayed until after opening.

The analysis relies on the uniformity of the random oracle and combinatorial bounds on the possible ways to choose the various cryptographic objects: for private keys larger than the number of bits encrypted, the analysis fails, leading to the stated requirement. Also note how the strategy puts a lower bound on the number of users in the system: concretely, if the random oracle output length is h, Yang et al. set $\kappa = h + 1$.

Yang et al.'s impossibility involved a notion of SSO in which the simulator was allowed to produce the parameters itself, cf. Remark 4; it implies impossibility for stronger notions where the simulator has less freedom.

Historical remarks. When Dwork et al. [41] first formalized selective opening attacks (for commitment schemes), they provided three definitions based on the framework of semantic security: one simulator based one, one based on a relation-predicate and one based on a function-predicate. The first two are equivalent, whereas the final one appeared weaker. Their simulation-based definition was then adapted to the public key setting with sender openings [10]. As with ISO, message sampling initially happened independently of the public key, meaning the notion did not imply IND-CPA; as detailed in Sect. 3.4, this shortcoming was later patched by BY12, who simultaneously introduced the notion of stateful samplers.

An SSO notion with receiver openings was subsequently developed [9]. Recently bi-openings were considered, including a "weak" version, for which an adversary must choose between access to either $\mathcal{S}$ or $\mathcal{R}$, but not both; it already has to make this choice after seeing the public keys, so before seeing any of the challenge ciphertexts. This weak version already turned out strictly stronger than either SSO⊙ and SSO⋆ individually [91].

As a final observation, the seeming difficulty in showing an implication from a posteriori simulatability to a posteriori indistinguishability with opening in the CCA setting as compared to in the CPA setting (see Open Problem 5) echoes the great time gap between establishing the equivalence of a posteriori simulatability (semantic security) and a priori indistinguishability without openings in the CPA setting (1986) [98] versus the CCA setting (2003) [111].

3.6 A Priori Simulatability with Selective Opening (NCE)

As explained in Sect. 3.2, a priori simulatability captures the idea that knowledge of a message should not be a prerequisite for producing ciphertexts that can pass off as encryptions of it, or in other words: irrespective of m, a simulator (without access to m) should be able to create a ciphertext c that cannot be distinguished from a real encryption of m.

Our formalization tasks a simulator Sim with simulating the view of adversary $\mathbb{A}$, taking on the role of game rather than player. This has the benefit of involving no message samplers: single messages are simply chosen by the adversary and given to the encryption oracle, who receives a (real or simulated) encryption in return. This mechanism makes a transmission oracle superfluous, as $\mathbb{A}$ always knows what message a challenge was supposed to encrypt; we therefore exclude the $\mathcal{T}$ oracle from Fig. 13.

Definition 7. *The* κ-NCE-CCA⊛ *advantage* $\mathsf{Adv}^{\kappa\text{-nce-cca}⊛}_{\mathsf{PKE}[\lambda],\mathsf{Sim}}(\mathbb{A})$ *of an adversary* $\mathbb{A}$ *against public key encryption scheme* $\mathsf{PKE}[\lambda]$*, relative to simulator* Sim*, is the distinguishing advantage against the game* $\mathsf{Exp}^{\kappa\text{-nce-cca}⊛}_{\mathsf{PKE}[\lambda],\mathsf{Sim}}(\mathbb{A})$ *(see Fig. 13).*

```
Experiment $\mathsf{Exp}^{\kappa\text{-nce-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{Sim}}(\mathbb{A})$
$b \leftarrow\$ \{0,1\}$
$\mathcal{O}_b \leftarrow (\mathcal{E}_b, \mathcal{D}_b, \mathcal{S}_b, \mathcal{R}_b)$
$s \leftarrow \varepsilon, q \leftarrow 0$
$\mathsf{pm} \leftarrow\$ \mathsf{PKE.Pm}(\lambda)$
$\textbf{if } b = 0:$
    $\forall_{i\in[\kappa]}(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow\$ \mathsf{PKE.Kg}(\mathsf{pm})$
$\textbf{else }:$
    $\forall_{i\in[\kappa]}\mathsf{pk}_i \leftarrow\$ \mathsf{Sim}_{\langle s\rangle}(\mathsf{Ini}, \mathsf{pm})$
$\hat{b} \leftarrow\$ \mathbb{A}^{\mathcal{O}_b}(\mathsf{pm}, \mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
$\textbf{return } b = \hat{b}$
```

```
Oracle $\mathcal{E}_0(i,m)$
$\textbf{if } i \in \mathcal{I} : \textbf{return } \lightning$
$r \leftarrow\$ \mathsf{PKE.Rnd}(\mathsf{pm})$
$c \leftarrow \mathsf{PKE.Enc}_{\mathsf{pk}_i}(m; r)$
$\mathtt{R} \xleftarrow{\frown} r$
$\mathcal{C}_i \xleftarrow{\cup} c$
$\textbf{return } c$
```

```
Oracle $\mathcal{D}_0(i,c)$
$\textbf{if } c \in \mathcal{C}_i : \textbf{return } \lightning$
$m \leftarrow \mathsf{PKE.Dec}_{\mathsf{sk}_i}(c)$
$\textbf{return } m$
```

```
Oracle $\mathcal{S}_0(j)$
$r \leftarrow \mathtt{R}[j]$
$\textbf{return } r$
```

```
Oracle $\mathcal{R}_0(i)$
$\mathcal{I} \xleftarrow{\cup} i$
$\textbf{return } \mathsf{sk}_i$
```

```
Oracle $\mathcal{E}_1(i,m)$
$\textbf{if } i \in \mathcal{I} : \textbf{return } \lightning$
$q \leftarrow q + 1$
$c \leftarrow\$ \mathsf{Sim}_{\langle s\rangle}(\mathsf{Enc}, i, |m|)$
$\mathtt{M} \xleftarrow{\frown} m, \mathtt{M}_i \xleftarrow{\frown} (q, m)$
$\mathcal{C}_i \xleftarrow{\cup} c$
$\textbf{return } c$
```

```
Oracle $\mathcal{D}_1(i,c)$
$\textbf{if } c \in \mathcal{C}_i : \textbf{return } \lightning$
$m \leftarrow\$ \mathsf{Sim}_{\langle s\rangle}(\mathsf{Dec}, i, c)$
$\textbf{return } m$
```

```
Oracle $\mathcal{S}_1(j)$
$m \leftarrow \mathtt{M}[j]$
$r \leftarrow\$ \mathsf{Sim}_{\langle s\rangle}(\mathsf{Sen}, j, m)$
$\textbf{return } r$
```

```
Oracle $\mathcal{R}_1(i)$
$\mathcal{I} \xleftarrow{\cup} i$
$\mathsf{sk}_i \leftarrow\$ \mathsf{Sim}_{\langle s\rangle}(\mathsf{Rec}, i, \mathtt{M}_i)$
$\textbf{return } \mathsf{sk}_i$
```

Fig. 13. The NCE experiment. Middle column = real, right column = ideal.

The oracles of Fig. 13 each come in two variants: one "real" (for $b = 0$), and one simulated, or "ideal" ($b = 1$). The real encryption, decryption, and opening oracles behave as expected: in particular, the real challenge encryption oracle simply encrypts the chosen message and returns the ciphertext.

Ideal oracles, meanwhile, call the corresponding subroutine of Sim. For the encryption oracle, Sim is asked to produce a ciphertext (under the relevant key handle) seeing only the length of the message. The message (or messages) is later revealed to Sim in the event that an opening oracle is called. To avoid trivial wins by the adversary, in the case of an $\mathcal{S}$ call Sim must come up with randomness such that re-encrypting the message produces the same ciphertext; similarly, in the case of $\mathcal{R}$ it should provide a private key such that decrypting *any* of the ciphertexts previously provided as a challenge under the corresponding key handle yields the correct message.

In our previous notions (Sect. 3.3–3.5), $\mathbb{A}$ was free to continue challenging a key handle i after the key had been opened. For NCE, such behaviour must be restricted, lest the notion becomes trivially unachievable. To see how, consider an adversary that calls $\mathcal{R}_b(i)$, receiving private key $\widetilde{\mathsf{sk}}_i$, followed by $\mathcal{E}_b(i, m)$ for a uniformly at random chosen message m of pre-determined length, receiving ciphertext c, and subsequently the adversary outputs $\hat{b} = 0$ iff $m = \mathsf{PKE.Dec}_{\widetilde{\mathsf{sk}}_i}(c)$. In the $b = 0$ world, the decryption check is guaranteed to succeed (assuming perfect correctness) so $\hat{b} = 0$ is guaranteed; yet, in the $b = 1$ world, m is information-theoretically hidden from Sim beyond its length $|m|$, thus the decryption check will hold with probability at most $2^{-|m|}$, yielding a significant distinguishing advantage of $1 - 2^{-|m|}$.

Even when a simulator would be allowed to program a random oracle, the adversary above is troublesome. Realistically, Sim's only hope to fool $\mathbb{A}$ is to program the random oracle for the calls that $\mathbb{A}$ makes to perform the check $m = \mathsf{PKE.Dec}_{\widetilde{\mathsf{sk}}_i}(c)$. In order to do so successfully, Sim would somehow have to learn m based on the queries $\mathbb{A}$ makes, but herein lies the rub: consider the sequence of oracle calls that honest decryption would make, and suppose there is a first oracle call whose input allows Sim to extract non-trivial information about m. Then that non-trivial information is already extractable from the answers Sim itself has provided so far, creating a complication (as m's contents would still have been information-theoretically hidden up to then based on prior calls). Essentially, even when Sim is allowed to program, it still ends up having to bootstrap its own knowledge of m (which is impossible).

```
Reduction $\mathbb{B}_{\text{nce}}(\mathsf{pm}, \mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
$s \leftarrow \varepsilon$
$\mathsf{out} \leftarrow\!\!\$\ \mathbb{A}_{\text{sso}}^{\mathcal{E},\mathcal{D},(\mathcal{T},)\mathcal{S},\mathcal{R}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
for $j \in [|\mathtt{K}|]$ :
    if $\mathtt{K}[j] \in \mathcal{I} : \mathcal{J} \overset{\cup}{\longleftarrow} j$
$\hat{b} \leftarrow\!\!\$\ \mathbb{D}_{\text{sso}}(\mathsf{pm}, \mathtt{A}, \mathtt{M}, \mathcal{J}, \mathsf{out})$
return $\hat{b}$
```

```
If $\mathbb{A}_{\text{sso}}$ calls $\mathcal{E}(i, \alpha)$
$\mathtt{K} \overset{\frown}{\longleftarrow} i, \mathtt{A} \overset{\frown}{\longleftarrow} \alpha$
$m \leftarrow\!\!\$\ \mathsf{M}_{\langle s \rangle}(\alpha)$
$\mathtt{M} \overset{\frown}{\longleftarrow} m$
$c \leftarrow \mathcal{E}_{\mathbb{B}}(i, m)$
return $c$
```

```
If $\mathbb{A}_{\text{sso}}$ calls $\mathcal{T}(j)$
$\mathcal{J} \overset{\cup}{\longleftarrow} j$
return $\mathtt{M}[j]$
```

```
If $\mathbb{A}_{\text{sso}}$ calls $\mathcal{S}(j)$
$\mathcal{J} \overset{\cup}{\longleftarrow} j$
$r \leftarrow \mathcal{S}_{\mathbb{B}}(j)$
return $(\mathtt{M}[j], r)$
```

```
If $\mathbb{A}_{\text{sso}}$ calls $\mathcal{R}(i)$
$\mathcal{I} \overset{\cup}{\longleftarrow} i$
$\mathsf{sk}_i \leftarrow \mathcal{R}_{\mathbb{B}}(i)$
return $\mathsf{sk}_i$
```

Fig. 14. The reduction $\mathbb{B}_{\text{nce}}$, simulating κ-SSO*-CCA$\circledast$ for $\mathbb{A}_{\text{sso}}$ and $\mathbb{D}_{\text{sso}}$ (the decryption oracle is simply forwarded).

As in previous sections, Sim is stateful, and we again allow it (in the ideal game) to produce the public keys but not the parameters: as with SSO (cf. Remark 4), the notion is meaningful even when the parameters are generated by Sim. Restricting the simulator by disallowing the generation of its own pm yields a potentially stronger notion and, as it matches the formalism of SSO better, is our preferred option.

Remark 6. Our formalization employs stateful simulators; earlier formalizations of NCE often consider NCE schemes to be tuples of algorithms extended to include a faking and an opening algorithm. For example, Hazay et al. [63] define the algorithms $\mathsf{PKE.Enc}^*$ and $\mathsf{PKE.Open}$ such that any ciphertext produced by $\mathsf{PKE.Enc}^*$ (on input the public key and message length) may be opened using $\mathsf{PKE.Open}$ (on input the message, the public and private keys, a trapdoor produced by $\mathsf{PKE.Enc}^*$, and the ciphertext to be opened). In our nomenclature, this corresponds to a simulator with precisely prescribed state and behaviour, potentially strengthening the notion without clear benefits. (Hazay et al. further strengthen their notion by insisting the simulator uses an externally, honestly generated public key of which it only learns the private key when running $\mathsf{PKE.Open}$, but not yet when running $\mathsf{PKE.Enc}^*$.)

NCE implies SSO. We next show (Thm. 6) how a priori simulatability (NCE) tightly implies a posteriori simulatability (SSO) in the presence of bi-openings. The reduction comes with a crucial caveat though: due to our NCE notion's restriction that opened keys cannot subsequently be challenged, the notion of SSO implied by NCE is weakened to one that makes the same restriction.

Theorem 6. *Let* SSO* *be defined as* SSO*(Fig. 10), except that a query $\mathcal{E}(i, \alpha)$ with $i \in \mathcal{I}$ leads to the immediate return of $\lightning$. Let $\mathsf{PKE}[\lambda]$ be given, then there exists a type-preserving black-box reduction $\mathbb{B}_{\text{nce}}$ (with black-box access to $\mathbb{A}_{\text{sso}}, \mathbb{D}_{\text{sso}}$, and M to be quantified later) such that for all* NCE *simulators $\mathsf{Sim}_{\text{nce}}$, there is a (black-box)* SSO *simulator $\mathsf{Sim}_{\text{sso}}$ such that for all adversaries $\mathbb{A}_{\text{sso}}$, message samplers M, and distinguishers $\mathbb{D}_{\text{sso}}$,*

$$\mathsf{Adv}^{\kappa\text{-sso}^*\text{-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}_{\text{sso}}}(\mathbb{A}_{\text{sso}}, \mathbb{D}_{\text{sso}}) = \mathsf{Adv}^{\kappa\text{-nce-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{Sim}_{\text{nce}}}(\mathbb{B}_{\text{nce}})\,.$$

The runtime of $\mathbb{B}_{\text{nce}}$ is upper bounded by that of $\mathbb{A}_{\text{sso}}$ and $\mathbb{D}_{\text{sso}}$ combined, plus that of running M q_e times, where q_e is the number of oracle calls made by $\mathbb{A}_{\text{sso}}$ to $\mathcal{E}$. The runtime of $\mathsf{Sim}_{\text{sso}}$ is upper bounded by q times that of $\mathsf{Sim}_{\text{nce}}$, where q is the total number of oracle calls made by $\mathbb{A}_{\text{sso}}$ to $\mathcal{E}$, $\mathcal{D}$, $\mathcal{T}$, $\mathcal{S}$, and $\mathcal{R}$, and $\mathsf{Sim}_{\text{sso}}$ additionally calls its $\mathcal{T}$ oracle $|\mathcal{J}|$ times, where $|\mathcal{J}|$ is the total number of opened challenge ciphertexts.

Proof. Let $\mathbb{A}_{\text{sso}}$ be an adversary playing the SSO* game; without loss of generality, we may assume that $\mathbb{A}_{\text{sso}}$ does not make any (pointless) $\mathcal{E}(i, \alpha)$ queries for which $i \in \mathcal{I}$. Let $\mathbb{D}_{\text{sso}}$ be a distinguisher. As the games $\mathsf{Exp}^{\kappa\text{-sso}^*\text{-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}_{\text{sso}}}(\cdot)$ and $\mathsf{Exp}^{\kappa\text{-nce-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{Sim}_{\text{nce}}}(\cdot)$ are both independent of their simulators $\mathsf{Sim}_{\ldots}$ conditioned on their respective bits $b = 0$, we can create a reduction $\mathbb{B}_{\text{nce}}$ (Fig. 14) that calls $\mathbb{A}_{\text{sso}}$ and $\mathbb{D}_{\text{sso}}$, simulating their $\mathsf{Exp}^{\kappa\text{-sso}^*\text{-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}_{\text{sso}}}(\cdot)$ environment such that, if $\mathbb{B}_{\text{nce}}$'s oracles and inputs are real ($b = 0$), then so are those of $\mathbb{A}_{\text{sso}}$ and $\mathbb{D}_{\text{sso}}$. Thus

$$\Pr\left[\mathsf{Exp}^{\kappa\text{-nce-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{Sim}_{\text{nce}}}(\mathbb{B}_{\text{nce}})\ \middle|\ b = 0\right] = \Pr\left[\mathsf{Exp}^{\kappa\text{-sso}^*\text{-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}_{\text{sso}}}(\mathbb{A}_{\text{sso}}, \mathbb{D}_{\text{sso}})\ \middle|\ b = 0\right]$$

```
Simulator $\mathsf{Sim}_{\mathrm{sso}}(\mathsf{pm})$
$q \leftarrow 0, s \leftarrow \varepsilon$
$\forall_{i \in [\kappa]} \mathsf{pk}_i \leftarrow \mathsf{Sim}_{\mathrm{nce}\langle s \rangle}(\mathsf{Ini}, \mathsf{pm})$
$\mathsf{out} \leftarrow\!\!\$\ \mathbb{A}_{\mathrm{sso}}^{\mathcal{E},\mathcal{D},(\mathcal{T},)\mathcal{S},\mathcal{R}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
return $\mathsf{out}$
```

```
If $\mathbb{A}_{\mathrm{sso}}$ calls $\mathcal{D}(i, c)$
if $c \in \mathtt{C}_i$ : return $\lightning$
$m \leftarrow\!\!\$\ \mathsf{Sim}_{\mathrm{nce}\langle s \rangle}(\mathsf{Dec}, i, c)$
return $m$
```

```
If $\mathbb{A}_{\mathrm{sso}}$ calls $\mathcal{E}(i, \alpha)$
$q \leftarrow q + 1$
$\mathtt{M}_i \xleftarrow{\frown} (q, \bot)$
$|m| \leftarrow \mathcal{E}_{\mathsf{Sim}}(i, \alpha)$
$c \leftarrow\!\!\$\ \mathsf{Sim}_{\mathrm{nce}\langle s \rangle}(\mathsf{Enc}, i, |m|)$
$\mathtt{C}_i \xleftarrow{\cup} c$
return $c$
```

```
If $\mathbb{A}_{\mathrm{sso}}$ calls $\mathcal{T}(j)$
$m \leftarrow \mathcal{T}_{\mathsf{Sim}}(j)$
return $m$
```

```
If $\mathbb{A}_{\mathrm{sso}}$ calls $\mathcal{S}(j)$
$m \leftarrow \mathcal{T}_{\mathsf{Sim}}(j)$
$r \leftarrow \mathsf{Sim}_{\mathrm{nce}\langle s \rangle}(\mathsf{Sen}, j, m)$
return $(m, r)$
```

```
If $\mathbb{A}_{\mathrm{sso}}$ calls $\mathcal{R}(i)$
for $j \in |\mathtt{M}_i|$ :
    $(q, \bot) \leftarrow \mathtt{M}_i[j]$
    $m \leftarrow \mathcal{T}_{\mathsf{Sim}}(q)$
    $\mathtt{M}_i[j] \leftarrow (q, m)$
$\mathsf{sk}_i \leftarrow\!\!\$\ \mathsf{Sim}_{\mathrm{nce}\langle s \rangle}(\mathsf{Rec}, i, \mathtt{M}_i)$
return $\mathsf{sk}_i$
```

Fig. 15. An SSO simulator mimicking the behaviour of $\mathbb{A}_{\mathrm{sso}}$ in the $b = 1$ case of $\mathbb{B}_{\mathrm{nce}}$'s simulation (Fig. 14). Implicit in the figure is $\mathsf{Sim}_{\mathrm{sso}}$'s state s, which it uses to keep track of global variables.

(where we slightly abused notation by conflating the b bits from the two distinct games).

If $b = 1$, both $\mathsf{Exp}^{\kappa\text{-sso}^*\text{-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}_{\mathrm{sso}}}(\cdot)$ and $\mathsf{Exp}^{\kappa\text{-nce-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{Sim}_{\mathrm{nce}}}(\cdot)$ do depend on their simulator. We claim that for any simulator $\mathsf{Sim}_{\mathrm{nce}}$, the simulator $\mathsf{Sim}_{\mathrm{sso}}$ (Fig. 15) perfectly mimics $\mathbb{A}_{\mathrm{sso}}$'s behaviour in the ideal ($b = 1$) world. We may therefore conclude that

$$\Pr\left[\mathsf{Exp}^{\kappa\text{-sso}^*\text{-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathsf{Sim}_{\mathrm{sso}}}(\mathbb{A}_{\mathrm{sso}}, \mathbb{D}_{\mathrm{sso}}) \;\middle|\; b = 1\right] = \Pr\left[\mathsf{Exp}^{\kappa\text{-nce-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{Sim}_{\mathrm{nce}}}(\mathbb{B}_{\mathrm{nce}}) \;\middle|\; b = 1\right].$$

Combining the $b = 0$ and $b = 1$ cases with the definition of a distinguishing advantage yields the theorem statement. □

Remark 7. The restriction on $i \notin \mathcal{I}$ for $\mathcal{E}(i, \alpha)$-calls appears unfortunate and is largely an artefact of our modelling choice to allow an adversary adaptive oracle access to challenge encryptions. Clearly, our theorem suffices to show that (our notion of) NCE tightly implies the more customary staged, single-shot version of SSO (with stateless vector-sampling), as there the game only allows corruptions after the challenge.

For instance, implications similar to ours were previously shown for sender openings [45, Lemma 1] and receiver openings [63, Theorem 4.4] (based on variants of NCE⊙ and NCE⋆, respectively). However, both those results use a staged SSO notion; moreover, both results use a single-key single-shot NCE notion, resulting in a reduction loss linear in the number of challenge ciphertexts (i.e. equal to the length of the vector of messages being sampled). In comparison, our result is tighter and more general and seems to resolve an open problem identified by BY12, who imagine that the general NCE⊛ (referred to as "MPC definition" by BY12) implies both SSO⊙ and SSO⋆.

The proof of Thm. 6 can readily be adapted to use the ($i \notin \mathcal{I}$)-restricted variant of SSO′ (Fig. 11) as target notion, if so desired. However, whether NCE suffices to imply (unrestricted) SSO we leave open (Open Problem 6).

Open Problem 6. *Does* NCE *imply* SSO *(without restriction) and if so, how tightly?*

Asymptotic interpretation. Our concrete definition of an NCE advantage (Def. 7) leads to two possible asymptotic definitions of security, again depending on the order of quantifiers (cf. SSO's asymptotic interpretation). For the "weak" option, a scheme PKE is deemed secure if, for all PPT $\mathbb{A}$ there exists a PPT simulator Sim resulting in a negligible advantage, whereas for the "strong" option, a scheme is only deemed secure if there exists a universal PPT simulator Sim that works for all PPT $\mathbb{A}$ (i.e. will result in a negligible advantage for all PPT $\mathbb{A}$).

The way our concrete implication is stated (Thm. 6) allows us to conclude that weak NCE implies the corresponding weak SSO* and strong NCE implies the corresponding strong SSO*.

NCE⋆ is unachievable (without programming). Just like κ-SSO-CPA⋆, κ-NCE-CPA⋆ is unachievable in the non-programmable random oracle model (in the sense that private keys would have to be at least as long as the total number of plaintext bits to be encrypted), as first shown by Nielsen [100]. Since κ-NCE-CPA$\star$ implies κ-SSO-CPA$\star$, this impossibility also follows from the (later) impossibility of κ-SSO-CPA$\star$ [113] (see Sect. 3.5).

Intuitively, Nielsen's proof is combinatorial in nature. There are at most $2^{|\mathsf{sk}|}$ possible decryptions of each ciphertext (one for each key); since chosen messages are independent of the private keys, if the messages are chosen from a set significantly larger than $2^{|\mathsf{sk}|}$, then it becomes likely that the chosen message is not in the set of possible decryptions for the simulated ciphertext.

The unachievability of κ-NCE-CPA$\circledast$, κ-NCE-CCA$\star$, and κ-NCE-CCA$\circledast$ in the non-programmable random oracle model follows.

Historical remarks. The formulation of NCE arose from the study of multi-party computation (MPC), whose protocols typically rely on suitably secure channels between any pair of parties. Before NCE, many protocols were only shown secure against adaptive adversaries in the information-theoretic setting; in the computational setting, it was only known how to achieve security against static adversaries, i.e. lacking the ability to adaptively choose which parties to open (after seeing the challenges). NCE arose naturally as a cryptographic security notion for secure peer-to-peer channels, permitting MPC schemes secure against adaptive adversaries.

Beaver and Haber [6] were the first to achieve such adaptively secure channels, but their solution crucially relied on secure erasures, meaning their scheme was not secure in the presence of (receiver) openings. Later schemes achieved security against adaptive adversaries in the presence of openings, at the cost of being interactive (three-round) protocols [5,29,38]; these are all proven secure in the standard model. Nielsen then completed the picture by constructing non-interactive NCE (sometimes known as NINCE) in the programmable random oracle model, and proving that programming is necessary to achieve NCE-CPA$\circledast$ non-interactively [100].

Camenisch et al. [25] gave an updated definition of 1-NCE-CCA$\circledast$ ("FULL–SIM" [25, Def. 6]) that allowed the adversary adaptive access to a sender and a receiver opening oracle, and featuring a stateful simulator. Thus their formalization closely resembles ours (Def. 7), with the exception that $\kappa = 1$, and that they allowed for public key encryption with labels [26]. Moreover, they allow for challenges to be issued even after corrupting the (one) user by altering the mechanism so that, post-compromise, the full message is fed to the simulator, enabling simulation (and bypassing the counterexample we used to argue to not allow post-compromise challenges). One can update Def. 7 to include such a mechanism.

Jaeger [82] recently introduced a generalized definition of NCE, referred to as SIM*, in which access to a programmable random oracle is part of the security notion (rather than the construction). Several composition results not known to hold in the a priori simulatability setting, such as the composition of a KEM and a DEM to form a PKE, were shown to hold for notions of SIM*. Intuitively, as simulator, adversary, and reduction alike are all allowed to program the same random oracle, programming can be made consistent across several game hops. He furthermore showed that κ-SIM*-CCA$\circledast$ hybridizes in the number of users κ, a result not known to hold for notions of NCE. (To see that SIM* is a strict generalization of NCE, observe that the adversary is strengthened compared to NCE adversaries due to its ability to program the random oracle, while the simulator is restricted compared to NCE simulators due to only being given the ability to program one specific oracle; thus, notions of SIM* trivially imply the corresponding notion of NCE.)

4 Relations

The road to understand the relations between the various notions presented in Sect. 3 has been long and winding, and as we have seen with the various open problems, the journey is still ongoing. We provide overviews of known relations in the CPA and CCA settings in Fig. 16. Here, bold arrows highlight results new to the current work, while purple dashed arrows represent relations that have, to the best of our knowledge, yet to be formally established, but that mirror other known results.

We stress that Fig. 16 reflects asymptotic interpretations of the various notions as, historically, these notions and their relations have primarily been considered asymptotically. As we mentioned in Sect. 2.3, the relevant implications and separations are given relative to classes of samplers/simulators in that case. For instance, an implication SSO-CPA$\odot$ $\Rightarrow$ ISO-CPA$\odot$ should be interpreted that, for all efficient PKE, if for the class of all efficient samplers SSO-CPA$\odot$ security holds, then ISO-CPA$\odot$ security also holds for the class of all samplers with efficient resamplability. For this particular implication, one can instead show the more concrete statement that for any efficient PKE and any sampler with efficient resamplability, SSO-CPA$\odot$ security with respect to that particular sampler implies ISO-CPA$\odot$ security with respect to that same sampler. In addition to the trivial, 'drop-an-oracle' concrete implications, we provided concrete counterparts for the main "downwards" implications in Sect. 3.

In contrast, we leave the remaining (non-trivial) results using their original asymptotic phrasing, unless explicitly stated otherwise. In addition to (full) implications as explained above, these results include partial implications and various kinds of separations. A partial implication IND-CPA $\Rrightarrow$ SSO-CPA$\odot$ states that for all efficient PKE, if IND-CPA security holds, then for all samplers in some class, SSO-CPA$\odot$ security holds (the partiality of the implication may also restrict the class of PKE).

Separations show that implications cannot be proven, for instance IND-CPA $\not\Rightarrow$ SSO-CPA$\odot$ would indicate that there exists an efficient IND-CPA-secure scheme and some efficient sampler for which SSO-CPA$\odot$ security does not hold. From a strict, logical perspective, such a separation has to be conditional on the existence of an efficient IND-CPA-secure scheme to begin with. Often separations only have a partial scope, for instance by showing that they only hold for efficient PKE with certain properties, or when considering black-box reductions only. Although we will always indicate the rough scope, we routinely omit the precise details and conditions under which a separation has been shown and refer to the relevant source for details instead.

Finally, we will encounter some semi-separations, in the sense that partial security of one notion is insufficient to establish an impliciation. For instance, Cor. 1's semi-separation ISO-CCA$\odot$ $\not\Rrightarrow$ SSO-CPA$\odot$ indicates that there plausibly exists an efficient PKE for which ISO-CCA$\odot$ security holds with respect to some strict subclass of message samplers, yet SSO-CPA$\odot$ security with respect to some relevant class of message samplers does not.

Remark 8. Formalizations often differ between works in subtle but important ways (for instance the order of quantifiers or the exact interfaces of adversaries, simulators and distinguishers), and the implications and separations given in Fig. 16 should therefore be interpreted as going between families of notions, in the sense that there is for instance a formalization of SSO-CPA$\odot$ known to imply a formalization of ISO-CPA$\odot$; or the other way around, that some formalization of ISO-CPA$\odot$ cannot imply some formalization of SSO-CPA$\odot$.Consequently, one should take some care when interpreting the figures, as arrows may not trivially compose. To alleviate some of these complications, we provided concrete, more systematic versions of several known implications in Sect. 3 with updated proofs, and our asymptotic interpretations clarify which implications do hold and between which formalizations subtleties arise. One striking example where composition is not straightforward is our reduction from NCE to SSO (Thm. 6), as it requires challenging compromised key handles to be disallowed also in the SSO experiment, as it is in the NCE experiment. Thus, implications from NCE to notions further down the hierarchy also do not immediately follow without putting similar restrictions on each notion.

4.1 Hybridization

A natural question when presented with any fully adaptive, multi-user security notion is whether the notion hybridizes in the number of users and challenges (i.e. the number and type of calls to the challenge encryption oracle $\mathcal{E}$), as is the case for multi-user indistinguishability without openings [7]. Indeed, in the closely related a priori indistinguishability setting the answer is yes as we noted in Sect. 3.3 already: the original hybrid proof works equally well with receiver openings (i.e. for κ-IND-CPA$\star$/κ-IND-CCA$\star$) [68], and is unaffected by the presence of sender or transmission openings. Of more interest are the other three settings, especially as historically they were often presented in a single challenge setting and, in the case of sender openings only, with only a single user. Our first observation is that in all three settings the availability of openings does seem to, at the very least, complicate any hybrid argument.

Users. On the positive side, ISO$\diamond$ and SSO-CPA$\diamond$ should hybridize due to their equivalence with IND (see Sect. 4.2 below); we conjecture that so does SSO-CCA$\diamond$.

On the other hand, for receiver openings, corrupting a single user does not benefit an adversary, thus IND $\Rightarrow$ 1-SSO$\star$ $\Rightarrow$ 1-ISO$\star$. As hybridization in the number of users would mean that 1-ISO$\star$ $\Rightarrow$ κ-ISO$\star$, resp. 1-SSO$\star$ $\Rightarrow$ κ-SSO$\star$ and neither κ-ISO$\star$ nor κ-SSO$\star$ can be equivalent to IND-CPA (see Sect. 4.3 below), hybridization is not possible.

Jaeger [82] noted that providing a proof of hybridization for NCE$\star$ appears difficult, however a proof of impossibility remains elusive.

Open Problem 7. *Determine whether* NCE$\star$ *hybridizes in the number of users.*

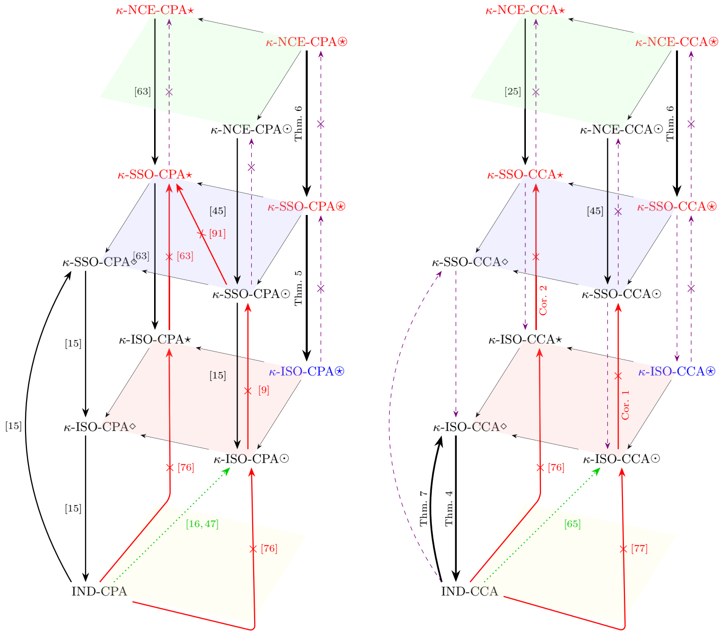


Fig. 16. Known and conjectured relations in the CPA and CCA settings. Highlighted in red are notions known to be unachievable in the standard model, and in blue notions for which standard model achievability remains open. Slim arrows (without references) are trivial, bold arrows highlight implications shown for the first time here, and violet dashed arrows are open problems (shown as the relation one would expect based on known results). Finally, green dotted arrows represent conditional implications.

In the presence of sender openings only, the question of hybridization appears mostly unexplored.

Open Problem 8. *Do notions of* ISO$\odot$, SSO$\odot$, *and/or* NCE$\odot$ *hybridize in the number of users?*

Challenges. NCE does not hybridize in the number of challenges in general; this follows from the impossibility result: there are standard model schemes secure against single-challenge NCE (as long as $|\mathsf{sk}| \geq |m|$) that will be insecure against q-challenge NCE (for insufficient $|\mathsf{sk}|$).

In the ISO and SSO settings, hybridization in the number of challenges depends on the sampler (see "Alternative samplers" in Sect. 3.4). For stateless samplers, hybridization is possible [16, 87]. However, in the stateless setting, hybridization in the vector length ℓ is not always possible: specifically, for κ-SSO-CCA$\star$ vector message sampling $\mathsf{M}^{\ell}_{\langle\rangle}$ where multiple messages per key are sampled simultaneously (so ℓ is a multiple of κ) is strictly stronger than the setting where for each challenge only a single message is sampled per key (so $\ell = \kappa$), at least without the ability to program oracles, such as a random oracle [113]. Consequently, for stateful samplers in that same κ-SSO-CCA$\star$ setting, no generic hybrid argument is possible for the number of challenges. On the other hand, for (stateless) product samplers a hybrid argument is possible in the length of the messages vectors which was initially shown for commitment schemes [41] and which is similar to the hybrid argument used for the number of challenges; in general, the lack of hybridization in one setting (κ-SSO-CCA$\star$) need not imply the lack thereof in other ones.

Open Problem 9. *When is hybridization in the number of challenges/length of the sampling vector (im)possible in the* ISO *and* SSO *settings?*

4.2 Implications

Trivial implications from ignoring oracles. Almost any time a security notion comes with a helper oracle, such as the decryption oracle $\mathcal{D}$ of CCA or the various opening oracles (see Table 1), there is a trivial reduction to a security notion without the oracle that simply ignores the oracle in question. Possible, non-trivial complications may arise for simulator-based notions where the simulator also loses oracle access and hence becomes less powerful; for our two simulator-based notions no such complications can arise as NCE's simulator has no oracle access whatsoever and SSO's simulator access to $\diamond$ is effectively restricted (technically, security with some opening includes universal quantification over all adversaries that do not open and, for those adversaries, the simulator cannot call its $\diamond$ oracle without being immediately noticed by a distinguisher checking its $\mathcal{J}$ input). So for instance, any CCA notion implies the CPA equivalent, and any notion with bi-openings implies any notion with transmission, sender, or receiver openings only. In Fig. 16, these trivial implications are represented by slim arrows that outline each plane of the hierarchy, going from more powerful openings to weaker ones. Likewise, any notion in the rightmost column (CCA) trivially implies the corresponding notion on the left (CPA).

Downward implications. The hierarchical structure between the different philosophies presented in Fig. 1 (from a priori indistinguishability up to a priori simulatability) is well-established in the CPA setting, as follows from the asymptotic interpretations of Thm. 6 (NCE implies SSO [45,60,63]), Thm. 5 (SSO implies ISO [15]), and Thm. 4 (ISO implies IND [15]), respectively. The main caveat that we uncovered is that NCE only implies a slightly restricted version of our more general SSO notion, although it still suffices to imply the older SSO version (with a stateless vector sampler). Various separation results, to be discussed in Sect. 4.3, reinforce the hierarchy by ruling out that 'lower' notions are in fact equivalent to the corresponding 'higher' one (with the noticeable exception of CPA$\diamond$, for which IND, ISO and SSO are all equivalent). By contrast, the CCA hierarchy currently contains a glaring hole as far as implications go, namely whether SSO-CCA implies ISO-CCA in the presence of openings, see Open Problem 5.

A priori indistinguishability and tightness. As mentioned in Sect. 4.1, the multi-user-with-corruptions notions of the a priori indistinguishability setting hybridize, meaning they are all implied by the single-user IND-CPA/IND-CCA notions with a tightness loss linear in the number of users [7], and the number of challenge bits in the case of β-IND-CCA$\odot$ and (κ, β)-IND-CCA$\circledast$ (Thm. 3), and so these notions are all asymptotically equivalent. Since Fig. 16 maps asymptotic implications and separations, it therefore does not distinguish between single- and multi-user notions of a priori indistinguishability. (We will revisit tightness in Sect. 5.2.)

IND partially implies ISO$\odot$. There are classes of message samplers for which IND-CPA does imply κ-ISO-CPA$\odot$: Fuchsbauer et al. [47] showed that these include samplers inducing product distributions (i.e. independent message sampling), Markov distributions, and more generally any graph-induced message distribution for which the underlying directed graph can be traversed in polynomial time (for a certain definition of "traversed"). Heuer later showed that the result transfers to the CCA setting [65]; extending to receiver openings remains open. These partial implications appear as green dotted arrows in Fig. 16.

Open Problem 10. *For which classes of message samplers does* IND-CPA *security imply* κ-ISO-CPA$\star$ *security, or even* κ-ISO-CPA$\circledast$ *security? How about in the* CCA *setting?*

IND implies notions of SOA with transmission openings. As shown by BY12, IND-CPA implies κ-SSO-CPA$\diamond$, and when starting from κ-IND-CPA, the reduction is furthermore tight [16, Thm. 4.1]. As we will show momentarily, for ISO a similar implication holds also in the CCA setting, which follows from a reduction to multi-user real-or-random indistinguishability (Thm. 7).

For SSO, it is unclear whether a similar implication holds in the CCA setting; a straightforward upgrade of BY12's proof runs into technical difficulties with simulating access to a decryption oracle (intuitively, the simulator would not have access to a decryption oracle, and we cannot rule out the possibility that an adversary could somehow convince the distinguisher that it does have access to one).

Open Problem 11. *Does* IND-CCA *security imply* SSO-CCA$\diamond$ *security?*

Theorem 7. *Let* $\mathsf{PKE}[\lambda]$ *be given, let* $q \in \mathbb{Z}_{>0}$*, and let* M *be a sampler with ideal resampler* $\mathring{\mathsf{S}}$*. Then there is a reduction* $\mathbb{B}_{\text{ror}}$ *such that for any adversary* $\mathbb{A}_{\text{iso}}$ *making at most q challenge oracle calls,*

$$\mathsf{Adv}^{\kappa\text{-iso-cca}\diamond}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}_{\text{iso}}) = 2 \cdot \mathsf{Adv}^{\kappa\text{-ror-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ror}})\,.$$

The runtime of $\mathbb{B}_{\text{ror}}$ *is upper bounded by that of* $\mathbb{A}_{\text{iso}}$*, plus that of running* M *q times and* $\mathring{\mathsf{S}}$ *once, and some small overhead.*

```
Reduction $\mathbb{B}_{\text{ror}}(\mathsf{pk}_1, \dots, \mathsf{pk}_\kappa)$
    $d \leftarrow\$ \{0,1\}$
    $\mathsf{challenged} \leftarrow \mathsf{false}$
    $\hat{d} \leftarrow\$ \mathbb{A}_{\text{iso}}^{\mathcal{E},\mathcal{C},\mathcal{D},\mathcal{T}}(\mathsf{pk}_1, \dots, \mathsf{pk}_\kappa)$
    $\hat{b} \leftarrow \neg(d = \hat{d})$
    return $\hat{b}$

If $\mathbb{A}_{\text{iso}}$ calls $\mathcal{D}(i, c)$
    $m \leftarrow \mathcal{D}_{\mathbb{B}}(i, c)$
    return $m$

If $\mathbb{A}_{\text{iso}}$ calls $\mathcal{E}(i, \alpha)$
    if $\mathsf{challenged}$ : return $\lightning$
    $\mathtt{A} \xleftarrow{\frown} \alpha$
    $m \leftarrow\$ \mathsf{M}_{\langle s \rangle}(\alpha)$
    $\mathtt{L} \xleftarrow{\frown} |m|$
    $\mathtt{M}^0 \xleftarrow{\frown} m$
    $c \leftarrow \mathcal{E}_{\mathbb{B}}(i, m)$
    return $c$

If $\mathbb{A}_{\text{iso}}$ calls $\mathcal{C}$
    if $\mathsf{challenged}$ : return $\lightning$
    $\mathsf{challenged} \leftarrow \mathsf{true}$
    $\mathtt{M}^1 \leftarrow\$ \mathring{\mathsf{S}}(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$
    return $\mathtt{M}^d$

If $\mathbb{A}_{\text{iso}}$ calls $\mathcal{T}(j)$
    if $\mathsf{challenged}$ : return $\lightning$
    $\mathcal{J} \xleftarrow{\cup} j$
    return $\mathtt{M}[j]$
```

Fig. 17. The reduction $\mathbb{B}_{\text{ror}}$ of Thm. 7 simulating κ-ISO-CCA◇ for $\mathbb{A}_{\text{iso}}$.

Proof. The reduction $\mathbb{B}_{\text{ror}}$ (Fig. 17) simulates κ-ISO-CCA◇ for $\mathbb{A}_{\text{iso}}$. Let b denote the challenge bit of the $\mathsf{Exp}^{\kappa\text{-ror-cca}}_{\mathsf{PKE}[\lambda]}(\cdot, \cdot)$ game, then $\mathbb{B}_{\text{ror}}$ wins with the following probability.

$$\begin{aligned}\Pr\Big[\mathsf{Exp}^{\kappa\text{-ror-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ror}})\Big] &= \frac{1}{2}\left(\Pr\Big[\hat{b} = 0 \;\Big|\; b = 0\Big] + \Pr\Big[\hat{b} = 1 \;\Big|\; b = 1\Big]\right)\\ &= \frac{1}{2}\left(\Pr\Big[d = \hat{d} \;\Big|\; b = 0\Big] + \Pr\Big[d \neq \hat{d} \;\Big|\; b = 1\Big]\right) ,\end{aligned}$$

as follows from $\mathbb{B}_{\text{ror}}$'s description (Fig. 17). The first term represents the probability that $\mathbb{A}_{\text{iso}}$ wins a faithful simulation of the ISO-CCA◇ game, and so

$$\Pr\Big[d = \hat{d} \;\Big|\; b = 0\Big] = \Pr\Big[\mathsf{Exp}^{\kappa\text{-iso-cca}\diamond}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}_{\text{iso}})\Big] .$$

For the second term, the challenge bit is information-theoretically hidden from $\mathbb{A}_{\text{iso}}$ (as resampling is ideal), so that

$$\Pr\Big[d \neq \hat{d} \;\Big|\; b = 1\Big] = \frac{1}{2} .$$

Recombining the two cases gives the theorem statement:

$$\begin{aligned}2 \cdot \mathsf{Adv}^{\kappa\text{-ror-cca}}_{\mathsf{PKE}[\lambda]}(\mathbb{B}_{\text{ror}}) &= 2 \cdot \Pr\Big[d = \hat{d} \;\Big|\; b = 0\Big] + 2 \cdot \Pr\Big[d \neq \hat{d} \;\Big|\; b = 1\Big] - 2\\ &= 2 \cdot \Pr\Big[\mathsf{Exp}^{\kappa\text{-iso-cca}\diamond}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}_{\text{iso}})\Big] + 2 \cdot \frac{1}{2} - 2\\ &= \mathsf{Adv}^{\kappa\text{-iso-cca}\diamond}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{A}_{\text{iso}}) .\end{aligned}$$

□

Asymptotic interpretation. For an asymptotic interpretation of Thm. 7, we observe that the runtime of the reduction $\mathbb{B}_{\text{ror}}$ includes that of the ideal sampler $\mathring{\mathsf{S}}$, and thus the reduction might not be efficient. Like for Thm. 5, we can ensure $\mathbb{B}_{\text{ror}}$ is efficient by replacing its call to $\mathring{\mathsf{S}}$ with that of an efficient resampler S, and rely on Lemma 3 to argue that this change only effects a negligible change in the reduction's advantage.

4.3 Separations

IND does not imply ISO (with sender or receiver openings). Standard IND-CCA security implies neither ISO-CPA⊙ nor κ-ISO-CPA⋆ in general: Hofheinz et al. [76, 77] showed that IND-CCA does not imply a closely related notion of threshold ISO-CPA⊙ security, for which the adversary gets a number of encrypted shares of a secret and must uncover the secret. Meanwhile, ISO-CPA⊙ does imply threshold ISO-CPA⊙, establishing a separation. Adapting the strategy to the receiver opening setting via an analogous notion of receiver-threshold security (in which each share is encrypted under a random public key rather than a single, fixed public key), they were able to likewise establish a separation from IND-CCA to κ-ISO-CPA$\star$ [76].

As, as mentioned in the introduction, actually developed use cases of SOA-notions are rare, the separations above simultaneously support our belief in the notions' usefulness in the context of threshold security, where IND-CCA is insufficient.

ISO does not imply SSO (with sender or receiver openings). Bellare et al. [9], generalizing an earlier result due to Hofheinz [10, 69], showed that no *committing* PKE scheme (i.e. one that implies a computationally binding non-interactive commitment scheme) can achieve SSO-CPA$\odot$, even when message samplers are restricted to independently sampled uniform bitstrings. (Hofheinz only showed that there exists a sampler relative to which such schemes cannot be proven SSO-CPA$\odot$-secure via black-box reductions to standard cryptographic assumptions.) Since there are IND-CCA schemes that are committing, for instance the Cramer–Shoup encryption scheme [35], they concluded that IND-CCA cannot imply SSO-CPA$\odot$.

As explained in Sect. 4.2, Heuer [65] lifted an earlier result in the CPA setting [47] to the CCA setting, and thereby showed that for many message samplers, including those that sample independently (i.e. that infer product distributions), IND-CCA does imply ISO-CCA$\odot$.

Combining the two results, we see that for a large class of message samplers, IND-CCA implies ISO-CCA$\odot$ while being separated from SSO-CPA$\odot$ (even when restricted to that same class of message samplers). Thus we reach a semi-separation (previously a full separation was observed in the CPA setting only [9]):

Corollary 1. *Class-restricted* ISO-CCA$\odot$ *security does not imply (class-restricted)* SSO-CPA$\odot$ *security.*

Turning to receiver openings, Bellare et al. [9] also identified a feature called decryption verifiability, which captures that anyone given a tuple consisting of public key, private key, ciphertext and message, should be able to verify that the ciphertext is an honest encryption of the message (for some definition of "honest"). They showed that no scheme that is decryption verifiable can be SSO-CPA$\star$ secure, even for independent, uniform sampling. Since many natural IND-CPA schemes are decryption verifiable, for instance ElGamal [43], a separation is established.

Both their results (committing schemes do not achieve SSO-CPA$\odot$; decryption verifiable schemes do not achieve κ-SSO-CPA$\star$) build on the same underlying proof idea as the impossibility of SSO$\star$, already discussed in Sect. 3.5: the adversary computes the hash of the challenge ciphertexts, and communicates the hash value to the distinguisher using the set of opened indices ($\mathcal{J}$ for $\odot$; $\mathcal{I}$ for $\star$); for any scheme that satisfies the respective property, no efficient simulator can provide a convincing simulation of this strategy.

However, with the standard model impossibility of κ-SSO-CPA$\star$ in hand, combined with the existence of schemes achieving κ-ISO-CCA$\star$ in the standard model (under reasonable computational assumptions, see Sect. 5.2), we can again reach a stronger conclusion (previously only shown for the CPA setting [63]):

Corollary 2. κ-ISO-CCA$\star$ *security does not imply* κ-SSO-CPA$\star$ *security.*

The separation above leaves room for partial implications, where κ-SSO-CPA$\star$ is implied for restricted classes of message samplers (that are not captured by the impossibility). Furthermore, that impossibility result itself relies on the ability of an adversary to use the index set of corrupted challenges ($\mathcal{J}$) or keys ($\mathcal{I}$) to 'communicate' with the distinguisher. Using a formalization of κ-SSO-CPA$\star$ where only the cardinality of those sets is passed directly to the distinguisher would require re-examination of those impossibility results (to determine whether they can be ported to the new formalization).

Technically, the situation is still partly open for bi-opening, as no scheme has yet been shown to achieve κ-ISO-CPA$\circledast$ in the standard model (see Open Problem 19); any such scheme would immediately lead to a separation, namely that κ-ISO-CPA$\circledast$ security does not imply κ-SSO-CPA$\circledast$ security, either in part (if the scheme only achieves CPA) or in full (if it also achieves CCA).

4.4 Further Relations

SSO and NCE. Remarkably, whether SSO and NCE are separated or equivalent appears largely open, beyond the logical consequences of various unachievability results for both SSO$\star$ and NCE$\star$ in the standard model.

Camenisch et al. [25] claim that NCE$\star$ is strictly stronger than notions of SSO$\star$, however this claim was based on the former being unachievable in the standard model, and the belief that the latter was achievable in the standard model. With neither achievable in the standard model, their relation becomes less relevant from a logical perspective (unless perhaps in an idealized model, such as the programmable random oracle).

Hazay et al. [63] gave a separation from κ-SSO-CPA$\star$ to κ-NCE-CPA$\star$, however their algorithmic formalization of NCE was stronger than our Def. 7, cf. Remark 6: in particular, and unlike our formalization, their NCE simulator was not allowed to produce the public keys (while their SSO simulator was). This technicality turns out to be central to their separation result.

Open Problem 12. *Which, plausibly achievable notions of* SSO *imply notions of* NCE *in general and which are separated? For settings where neither notion is achievable in the standard model, which relations can be drawn in (specific) idealized models?*

Sender vs. receiver opening. Although sender openings and receiver openings model quite different scenarios, their formalizations have a lot in common; moreover, ideas for one scenario have subsequently often been adapted to the other. Yet, the question when one type of opening implies the other, or conversely, when no such implication can exist, has not achieved much direct attention (Open Problem 13). Of course, juxtaposing feasibility results for one with impossibility results for the other, does provide some answers.

Together, the unachievability of κ-SSO-CPA$\star$ in the standard model and combined with the plausible existence of a standard model construction achieving NCE-CCA$\odot$ [45] (see Sect. 5.2) seem to indicate that a priori simulatability with sender opening (NCE$\odot$) cannot imply a posteriori simulatability with receiver opening (SSO$\star$). However, the construction was only shown 1-NCE-CCA$\odot$ secure in the standard model. As it is unclear whether NCE$\odot$ hybridizes in the number of users (see Sect. 4.1), and SSO$\star$, unlike NCE$\star$, is only relevant for $\kappa > 1$ (for $\kappa = 1$ it becomes equivalent to a notion without openings), we can only formally conclude that 1-NCE-CCA$\odot$ does not imply 1-NCE-CPA$\star$. Of course, the achievability of 1-NCE-CCA$\odot$ feeds the impression that κ-NCE-CCA$\odot$ is achievable as well, which would restore a full separation.

BY12 gave a construction that achieves κ-SSO-CPA$\odot$ in the standard model for arbitrary κ; we may therefore conclude that κ-SSO-CPA$\odot$ does not imply κ-SSO-CPA$\star$.

The reverse direction is uninteresting for unrestricted message spaces in the standard model (as SSO$\star$ and NCE$\star$ are both unachievable then). Lai et al. [91] observed that the Cramer–Shoup scheme [35] achieves a single-message-bit variant of κ-SSO-CCA$\star$ in the standard model [78], but as the scheme is committing, it cannot achieve SSO-CPA$\odot$ [69]; we therefore conjecture that notions of SSO$\star$ and SSO$\odot$ are separated in both directions, and thus incomparable.

Implications between notions with various openings of course do hold for a priori indistinguishability (at a loss, see Sect. 3.3), while the situation appears open for ISO.

Open Problem 13. *How do notions with sender openings (only) and notions with receiver openings (only) relate to each other?*

4.5 Selected Related Notions

So far, we have concentrated on the main notions that can arise when formalizing openings in the context of public key encryption. Below we will briefly address a few additional notions (we already touched upon these in Sect. 3). Yet, many more related security notions have been proposed in the past, that we will not expand upon. Some of these further notions are subtle variants of what we have already seen, for instance weak bi-opening [91] or tweaked NCE [63], while other notions go beyond message confidentiality, for instance by considering non-malleability [81] or anonymity/key-privacy [80].

SIM*. As NCE$\star$ cannot be realized in the random oracle model without programming (let alone the standard model), studying the notion only makes sense in idealized models. One difficulty that surfaces when including programming directly in the security model is the need to keep said programming consistent, especially when considering the composition of various reductions that might all vie with the simulator in their programming.

When Jaeger [82] recently introduced a strengthened notion of a priori simulatability, named SIM*-AC (for "Adaptive Compromise"), hereafter SIM*, he baked the ideal object into the notion, allowing simulators, reductions and, crucially, adversaries alike to program the ideal object through the same oracle

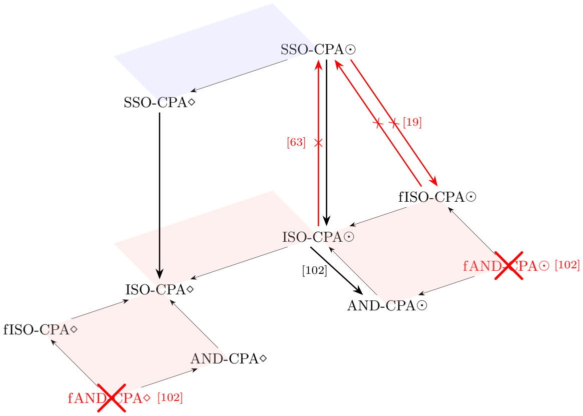


Fig. 18. Full ISO, full AND, and how they relate to the middle section of Fig. 16 (ISO-CPA⊙ and SSO-CPA⊙). A red cross means the notion is unachievable even in idealized models (under mild conditions).

interface. He showed that SIM* does hybridize in the number of users and that the security of several common constructions, such as the KEM/DEM hybrid PKE and the Fujisaki–Okamoto transform, carries over to the a priori simulatability setting. However, even for SIM* hybridization in the number of challenges appears tricky [83].

As SIM* security also covers adversaries that do not program, any SIM* notion tightly implies its NCE counterpart (in the programmable oracle model, where only the simulator and reduction may program).

Deniable encryption. Deniable public key encryption, as introduced by Cannetti et al. [28], shares many features with NCE: in particular, like NCE, one should be able to open a ciphertext to plaintexts other than the one originally encrypted. The main difference is that, for deniable encryption, this capability should be available to the end user, while for NCE we only require it to be accessible to the simulator (i.e. by programming ideal objects). Thus, non-interactive deniable PKE with bi-opening is strictly stronger than NCE, and thus it must also unachievable in the standard model. Unlike NCE however, non-interactive deniable PKE with receiver opening is unachievable *in general*: to see this, recall that programming random oracles is necessary to achieve NCE with receiver openings [100]; we have no way of granting the end user such powers.

On the other hand, non-interactive deniable encryption with sender opening *is* achievable, although the only known such scheme requires encryption to be a quantum algorithm [34]. Additionally there is a generic transform that takes any sender-deniable encryption scheme to a receiver-deniable encryption scheme by adding extra interaction (so it is no longer non-interactive, bypassing the impossibility result) [28], and there are interactive schemes achieving deniability with bi-opening [33].

If the requirement that a given ciphertext can be opened to any message is weakened to, for instance, only two predetermined messages (under two different keys), one can construct non-interactive schemes with this limited deniability functionality. Several common blockcipher modes-of-operation, including AES-GCM, sport such deniability-like properties [56], and prior works have rather focused on how to rid these schemes of that deniability, and thus achieve fully *committing* authenticated encryption [40, 56].

Full ISO-CPA and friends. Böhl et al. [19] noted how allowing resampling to be inefficient leads to a significantly stronger notion of ISO. They named the resulting notion "Full" ISO (fISO), and showed that fISO-CPA⊙ and SSO-CPA⊙ are incomparable. No scheme to date has been shown to achieve fISO-CPA (not even for transmission openings, though it has mainly been studied for sender openings, see Open Problem 14) and, as allowing inefficient resampling seems to yield an artificially strong notion of security, fISO has mostly fallen out of favour. fISO therefore sits outside of our main hierarchy (Fig. 16).

Open Problem 14. *Is* fISO-CPA *achievable in the presence of sender, receiver, or transmission opening?*

A closely related notion, which we will refer to as Full AND (fAND), works exactly as Full ISO, except that in the challenge phase (see oracle $\mathcal{C}$ in Fig. 8), the adversary gets to see both the original and the resampled messages [102]. Although (non-full) AND-CPA⊙ and ISO-CPA⊙ are equivalent, fAND turns out to be unachievable (for public key spaces supporting Σ-protocols) [102]. Furthermore, the proof that fAND-CPA is unachievable arlready holds in the presence of transmission opening (by inspection), in stark contrast to AND-CPA◇, which is equivalent to IND-CPA (Thm. 7). These results are summarized in Fig. 18, with the large red crosses representing unachievability.

The unachievability of fAND-CPA◇ lends support to our belief that fISO-CPA⊙ is an unachievable notion and that its abandonment is warranted.

5 Constructions

The literature is rich with constructions achieving the various notions of the hierarchy, from well-known constructions achieving IND-CPA at the bottom, to the unachievable-without-programming κ-NCE-CCA⊛ at the top.

In this section, we survey the state of the art, and group families of constructions based on their main underlying assumption. In addition, we identify a ROM construction technique due to Bellare and Rogaway [12], henceforth Bellare–Rogaway Encryption (BRE).

5.1 Bellare–Rogaway Encryption

Bellare–Rogaway Encryption combines a random oracle with one-time pad encryption, and the technique is as simple as it is elegant. It gives highly efficient schemes with very strong security guarantees, with two important caveats: first, the security proof relies on programming random oracles (which as we have seen is necessary to achieve simulatability with receiver opening, see Sect. 3.5–Sect. 3.6); and second, that the security reduces to one-wayness of a Key Encapsulation Mechanism (KEM) in the presence of plaintext checking attacks (OW-PCA), rather than just CPA. Such KEMs can be instantiated with the GapCDH assumption, or other such "gap" assumptions that exploit (conjectured) differences between computational and decisional hardness; there are also generic transformations taking OW-CPA to OW-PCA in the (Q)ROM [71]. OW-PCA has furthermore been shown to be necessary to prove CCA security of common BRE schemes [109] like REACT [101].

It goes as follows: let encryption, on input a public key pk and message m, encapsulate a seed K_{kem} as c_1, and let the seed be fed to an extendable output function (XOF) $\mathcal{F}$ to retrieve a pseudorandom bitstring K_{otp} of length m; then, use K_{otp} as key for one-time padding (OTP) the message, yielding c_2; the ciphertext c is the tuple (c_1, c_2). Decryption essentially repeats the process: decapsulate c_1 to recover seed K_{kem}, compute K_{otp}, and one-time pad it with c_2 to recover m. As indicated by the notation, key encapsulation mechanisms (KEMs) are optimal primitives for instantiating such schemes, provided they achieve OW-PCA. (Bellare and Rogaway originally employed trapdoor permutations in place of KEMs, for which plaintext checking comes for free [12].)

When modelling $\mathcal{F}$ as a random oracle, reprogramming enables a simulator: upon encryption, it simply chooses the message encryptions c_2 uniformly at random, and stores it for later. The properties of the OTP then ensure that, for any message m and ciphertext c_2, there exists a key K_{otp} such that $m \oplus K_{\text{otp}} = c_2$. Thus, when this message is later revealed to the simulator, it can simply reprogram $\mathcal{F}$ such that on input seed K_{kem}, the key $K_{\text{otp}} = m \oplus c_2$ is output. (If CCA-security is desired, the seed can be expanded to include a MAC key, with which the message can be authenticated.)

An adversary can notice the simulation by querying the random oracle on the correct seed before the message is revealed to the simulator, disrupting the planned programming. Thus, the security of the construction reduces to the one-wayness of the KEM: if a reduction can recognize such a query (through the use of the plaintext-checking oracle), then it can win the OW-PCA game.

To summarize: BRE is the combination of a KEM, a XOF modelled as an RO, the OTP, and possibly a MAC, with the proof relying on the ability of a simulator to make an OTP ciphertext decrypt to any message by programming the random oracle.

The BRE technique has appeared numerous times, including:

- to achieve IND-CPA and IND-CCA security (as originally conceived) [12];

- generalized as the REACT transformation, which takes any OW-PCA PKE to an IND-CCA PKE [101] and which also allows for other symmetric schemes in place of the OTP;
- modularized as a TagKEM hybrid PKE achieving tight multi-instance IND-CCA security [23];
- to achieve κ-SSO-CCA with sender [66], receiver [93], and bi-opening [91];
- to achieve (single-user) NCE with bi-opening in both CPA [100] and CCA [24,25] settings.

While terminology and implementation details often differ between the constructions (particularly in how exactly they achieve CCA security, for example by use of encrypt-then-MAC [23,66,79,91,101], or by use of MAC-and-encrypt [25]), these schemes all share the same underlying idea. As we see, the technique is very powerful, as it comes close to achieving the strongest notion of our hierarchy: Camenisch et al. [25] gave a construction achieving 1-NCE-CCA⊛, although generalizing to multiple users remains open.

Open Problem 15. *How can the BRE technique be used to achieve* κ-NCE-CCA⊛ *for* $\kappa > 1$*?*

On another note, as pointed out by Jaeger [82], the issue with modularity in the context of NCE lies with challenges in keeping random oracle programming consistent through multiple reductions. (And to paraphrase Krawczyk [89], modularity is simplicity's best friend.) As we have seen (Sect. 3.6), Jaeger resolved this issue by upgrading to notions of SIM*, i.e. notions of a priori simulatability in which all players—simulators, adversaries and reductions alike—are given the ability to reprogram the random oracle, facilitating consistent programming across several game hops. Such an approach could allow for a modularized BRE technique, for instance by first transforming the OW-PCA KEM into a suitably κ-SIM*-CCA⊛ secure TagKEM, and then show that combining it with the OTP yields a hybrid PKE that also achieves κ-SIM*-CCA⊛ (taking inspiration from Brunetta et al. [23]).

Open Problem 16. *To what extent can the BRE technique be used to achieve* κ-SIM*-CCA⊛ *in a modular manner?*

5.2 Other Constructions

Although the BRE technique gives simple and efficient constructions achieving one of the strongest notions of the hierarchy, modelling a primitive like the XOF as a programmable random oracle is, admittedly, a leap of faith; preferably, results are stated in the standard model, reducing to standard (falsifiable) assumptions, such as the pseudorandomness of the XOF's output or the intractability of finding colliding inputs (for a sufficiently long XOF output).

While neither NCE⋆ nor SSO⋆ is achievable in the standard model, there is a rich literature of constructions that do in ideal models, as well as those that achieve the various other notions in the standard model. Roughly speaking, moving down the hierarchy and/or from bi-openings to receiver or sender openings usually makes achievability easier in the sense of allowing simpler constructions based on weaker assumptions.

We provide an overview of the various constructions next: Table 2 groups constructions together in families based on their model, underlying assumption, or general approach, and places them in the hierarchy based on the strongest notion that a member of a family has been shown to achieve.

For the ideal models, we primarily consider the aforementioned programmable random oracle model [12], where specifically the simulator is allowed to program (cf. programming by a reduction [46]). An alternative to the programmable ROM is the programmable Ideal Cipher Model (ICM). In the ideal cipher model, a symmetric cipher is modelled as a family of random permutations; the ideal cipher model is primarily known from its non-programming incarnation [18,107,112], with programming relatively rare (whereas for random oracles, programming is fairly established). In Table 2, we concentrate on the ideal model being used by the various constructions. Obviously, there will still be a standard computational assumption needed; these will be explained in the accompanying text.

For constructions in the standard model, a wide variety of different primitives and hardness assumptions have been used to construct PKE schemes secure against various types and notions of opening. These underpinning principles are listed below.

- Lossy Encryption (LE) [10], a PKE scheme with an additional "lossy key generation" algorithm, such that 1) lossy keypairs are indistinguishable from normal ones, and 2) ciphertexts produced under lossy keys statistically hide the message. It follows that correctness cannot hold for lossy keypairs; informally speaking, encrypting with a lossy key should produce only garbage. It also follows that for any lossy keypair $(\mathsf{pk}, \mathsf{sk})$, ciphertext c, and message m, there exists (with high probability) randomness r such that encrypting m under lossy key pk and randomness r results in c, (thus revealing r opens c to m).

Table 2. An overview of the state of the art of constructions achieving the various notions (arrows point in the direction of superseding results). Highlighted in red are impossibility results, and in blue settings concerned with tightness (a priori indistinguishability); constructions that additionally achieve CCA security are highlighted in **bold**.

		Simulability		Indistinguishability	
		A Priori	A Posteriori	A Posteriori	A Priori
Ideal Model	⋆	↓	**ICM** [78]	←	**ROM** [92]
	⊛	**ROM** [25, 100]	←	←	↓
	⊙	↑	**ICM** [67]	←	Open Problem 21
Standard Model	⋆	× [100]	× [113]	**HPS** [87]	Open Problem 20, **MDDH** [58]
	⊛	↑	↑	Open Problem 19	Open Problem 21
	⊙	**HPS** [45]	LEEO [10], **TailKEM** [95, 97], **ABM-LTF** [70, 94]	LE [10]	Cor. 3

- Lossy Encryption with Efficient Opening (LEEO) [10], an LE scheme for which opening, as described above, can be done efficiently.
- All-But-Many Lossy Trapdoor Functions (ABM-LTF) [70], generalizations of Lossy Trapdoor Functions (LTFs) [104], themselves generalizations of trapdoor functions enhanced to two modes of operation: either, given access to a trapdoor, the entire pre-image can be recovered from the image, or a statistically significant amount of information about the pre-image is unrecoverable from the image and trapdoor alone; the two functionalities should furthermore be indistinguishable without access to the trapdoor. Thus they resemble lossy encryption as described above, and can indeed be used to instantiate LE schemes [10]. All-but-many LTFs are LTFs that are additionally parametrized by tags, which can be either injective or lossy, yielding injective or lossy trapdoor functions, respectively. A master trapdoor allows for the generation of an arbitrary number of lossy tags, while the production of lossy tags should be infeasible without access to the master trapdoor.
- Hash Proof Systems (HPS) [36], a primitive hailing back to the Cramer–Shoup encryption scheme [35] that resembles a non-interactive, designated verifier zero-knowledge proof system. HPS come in universal and weak [61] variants, with the latter resembling a KEM variant of lossy encryption [87].
- Tailored KEM (TailKEM) [95], KEMs with two additional properties: firstly, the support of the encapsulation algorithm should cover only a small subset of the full ciphertext space, and secondly, an ephemeral key and its encapsulation, as output by the encapsulation algorithm, should be indistinguishable from elements chosen uniformly at random from the respective spaces; additionally, the TailKEM should satisfy a "tailored" variant of CCCA security [74].
- the Matrix Decisional Diffie–Hellman assumption (MDDH) [44], which is a generalization of DDH and the n-linear assumption (nLin) [74].

As shown in Sect. 3, each layer of the hierarchy implies the layers below (conjectured in the case of SSO-CCA ⇒ ISO-CCA, see Open Problem 5). Therefore, any construction that achieves a notion in the hierarchy will also achieve the notions below it; and likewise, any construction achieving security under bi-opening trivially achieves security under receiver and sender (and transmission) openings. For brevity, we restrict Table 2 to show the most general placement(s) for each category only, with arrows pointing to stronger results. We make an exception for a priori indistinguishability, as asymptotically speaking these notions are all achieved by any IND-CPA/IND-CCA PKE scheme, and hence we focus on works explicitly targeting tightness under the notion in question (highlighted in blue in Table 2).

We will work our way through Table 2 next, first covering idealized settings, then achievability in the standard model for each of the notions going down the hierarchy (left to right in the table), before finishing with the quest for tightness in the a priori indistinguishability setting. Formal treatments of the various construction families are beyond our scope, as is comparing schemes in terms of efficiency and useability;

we will restrict ourselves to brief summaries of each, highlighting open questions of achievability as we go along.

On achievability. When we speak of a notion being achievable, we mean that, under some common computational assumption, there exists an efficient construction achieving (asymptotic) security under the notion, for arbitrary message spaces and for any number of challenges. As neither NCE$\star$ nor SSO$\star$ can be achieved in the standard model so generally (as private key lengths would have to exceed the total number of bits to be encrypted), several standard model constructions only support polynomially sized message spaces [10,60,63,73,78,86,91,96]. These constructions, while potentially useful for limited applications, do not appear in Table 2.

On tightness. So far, we have used the term "tight" to mean that the multiplicative loss of a concrete bound equals 1. In the following we will instead adopt the asymptotic convention of Han et al. [58] and say that a bound is "tight" if the multiplicative loss is independent of the number of oracle queries q, the number of users κ, and the security parameter λ, and "almost-tight" if the loss is independent of q and κ and at most linear in λ.

Ideal model constructions. We consider various ideal models that allow programming, such as the Random Oracle Model (ROM) [12] and the Ideal Cipher Model (ICM).

Further ROM constructions. Hofheinz et al. [77] showed that for a large class of group-based PKE constructions, IND-CPA implies ISO-CPA$\odot$ when some hash function in the PKE construction is modelled as a non-programmable random oracle and the underlying group is treated generically using a programmable interpretation of Shoup's Generic Group Model (GGM) [108].

Heuer et al. [66] showed that RSA-OAEP [13] achieves SSO-CCA$\odot$, and that the DHIES transformation [1], originally shown to take a OW-PCA KEM to a IND-CCA PKE in the ROM, actually yields PKE schemes achieving SSO-CCA$\odot$. Whether these constructions achieve even stronger notions remains open.

Open Problem 17. *Do the OAEP and DHIES transformations yield PKE schemes achieving κ-*NCE-CCA$\odot$*? What about κ-*NCE-CCA$\circledast$*?*

Heuer [65] showed that the Fujisaki–Okamoto (FO) transformation [48] turns a OW-CPA PKE satisfying a certain condition (γ-spreadness) into an ISO-CCA$\odot$ secure PKE. Pan et al. [103] gave three constructions that tightly (and compactly) achieve SSO-CCA$\odot$ in the ROM, based on the Computational Diffie–Hellman (CDH) assumption, the Strong CDH assumption, and the DDH assumption, respectively. They furthermore showed that the FO transformation tightly turns any LE scheme into an ISO-CCA$\odot$ secure scheme, and any LEEO scheme tightly into an SSO-CCA$\odot$ secure scheme. More recently, Jaeger [82] showed that the FO transformation turns any OW-PCA KEM into a κ-SIM*-CCA$\circledast$ secure PKE, which tightly implies κ-NCE-CCA$\circledast$ security (see Sect. 3.6).

ICM constructions. The programmable ICM allows for strong security guarantees for schemes that already see widespread use, such as KEM/DEM hybrid encryption (with the DEM based on the ideal cipher). The constructions below achieve SSO$\odot$ and SSO$\star$, respectively (neither construction seems to achieve SSO$\circledast$).

Heuer et al. [67] show that any hybrid PKE combining an IND-CCA secure KEM with an NCE-like data encapsulation mechanism ("simulatable" DEM) satisfies SSO-CCA$\odot$. They go on to show that common blockcipher modes of operation such as CTR, CBC, CCM, and GCM achieve this non-committing property in the programmable ideal cipher model.

Huang et al. [78] show that the Canetti–Halevi–Katz (CHK) transformation [30], originally taking any CPA-secure identity-based encryption (IBE) [106] to an IND-CCA secure PKE in the standard model, gives PKE schemes achieving κ-SSO-CCA$\star$ when instantiated with an IBE satisfying the IBE equivalent of SSO-CPA$\star$; such IBE schemes are then constructed in the programmable ideal cipher model. However, the CHK transformation does not seem to apply to the sender opening setting [64], making it a less promising route towards bi-opening resilience.

A priori simulatability (NCE) in the standard model. As already covered (Sect. 3.6), a priori simulatability is not achieveable in the standard model in the presence of receiver openings, nor, as a consequence, for bi-openings (see Sect. 3.6). For sender openings only however, the situation is different: Fehr et al. [45] gave constructions that achieve NCE-CPA$\odot$ and NCE-CCA$\odot$, respectively, in the standard model, from HPS for the former, and from HPS combined with a new primitive called a Cross-Authentication Code (XAC) for the latter. (Interestingly, neither scheme satisfies perfect correctness, cf. Remark 1.)

A variant of NCE has been studied under a restriction wherein each ciphertext need only be decryptable to one of ℓ pre-determined messages [49,62]. For ℓ sufficiently small, the standard model impossibility of NCE$\star$ is thus bypassed. We view the possibilities and limitations of such notions in the standard model as an interesting avenue for further exploration.

Canetti et al. [31] provided an alternative path to overcoming the standard-model impossibility of NCE$\circledast$, namely by letting the private key evolve over time (while keeping the public key fixed). However, their solution does rely on the secure erasure of older private key material, making it somewhat unsatisfactory.

A posteriori simulatability (SSO) in the standard model. Similarly to NCE, we have seen in Sect. 3.5 that a posteriori simulatability is unachievable in the standard model in the presence of receeiver openings, and so we will only discuss constructions for SSO$\odot$ below.

SSO$\odot$ *from LEEO.* Bellare et al. [10] defined lossy encryption, and showed that lossy encryption suffices to achieve ISO-CPA$\odot$ (without being necessary [102]); if opening is furthermore efficient, then the resulting LEEO scheme achieves SSO-CPA$\odot$. They go on to show that such schemes can be constructed in the standard model; indeed, they observe that the classical Goldwasser–Micali encryption scheme [55], based on the quadratic residuosity assumption, is a lossy encryption scheme with efficient opening.

BY12 later updated their definition of κ-SSO-CPA$\odot$ to include multiple challenges, users, and stateful samplers (as detailed in Sect. 3.4), and showed that LEEO achieves also this updated notion (at an additional security loss in the number of users and challenges) [16, Thm. 6.2].

Although LEEO on its own does not suffice to argue SSO-CCA$\odot$ security [45], there are specific LEEO schemes achieving SSO-CCA$\odot$ [94], see below.

SSO$\odot$ *from ABM-LTF.* Hofheinz [70] showed that SSO-CCA$\odot$ can be achieved from ABM-LTFs, and that ABM-LTFs can be constructed from the Decisional Composite Residuosity (DCR) assumption. (They do so by first showing that the scheme satisfies ISO-CCA$\odot$, and then by constructing an efficient opener, making the scheme a LEEO.)

Libert et al. [94] later achieved an almost-tight reduction from ABM-LTFs to SSO-CCA$\odot$, by the construction of a CCA-secure LEEO, while simultaneously constructing ABM-LTFs from Learning With Errors (LWE); however their construction relied on a non-standard PRF property, achieved through reduction to a stronger-than-usual LWE assumption (Non-uniform LWE [21]).

SSO$\odot$ *from TailKEM.* Generalizing the approach of Fehr et al. (see NCE in the standard model paragraph above), Liu et al. [95] introduced TailKEMs and showed how they could be used to construct PKE schemes that achieve SSO-CCA$\odot$ in the standard model. They furthermore showed that TailKEMs can be instantiated from HPS, indistinguishability obfuscation (i$\mathcal{O}$ [4], obfuscating programs so that an adversary with white-box access to the program learns nothing beyond its input–output behaviour, for some definition of "learning nothing"), or nLin.

Subsequently, Lyu et al. [97] showed that further refining the definition to cover multiple instances allows for a tight proof of security, and that such refined TailKEMs can be almost-tightly instantiated from MDDH.

A posteriori indistinguishability (ISO) in the standard model. While SSO$\star$ is unachievable in the standard model, ISO$\star$ has been achieved in the standard model, as detailed below; as has ISO$\odot$, though not ISO$\circledast$ (see Open Problem 19).

On a similar note, for sender opening, the impossibility for committing (binding) schemes to achieve SSO-CPA$\odot$ does not hold for ISO-CPA$\odot$ [9, 10].

Generic transformation from IND-CPA *to* κ-ISO-CCA$\star$*.* Jia et al. [87] gave a generic way to lift any κ-ISO-CPA$\star$ scheme to κ-ISO-CCA$\star$ in the standard model based on the Naor–Yung paradigm [99], combining the CPA scheme with an IND-CCA scheme and a non-interactive zero-knowledge proof. Combined with a κ-ISO-CPA$\star$ construction from any weak HPS, and Hazay et al.'s construction of weak HPS from any IND-CPA PKE [61], we get a transformation taking any IND-CPA secure PKE to a κ-ISO-CCA$\star$ secure PKE in the standard model.

Interestingly, the κ-ISO-CPA$\star$ construction [61] was originally aimed at achieving bounded leakage resilience [42], begging the question whether it hints to deeper connections between the two fields of study.

Open Problem 18. *How do notions of leakage resilience relate to notions of selective opening attacks?*

ISO$\star$ *from HPS.* In addition to the above-mentioned transformation, Jia et al. [87] showed that the HPS-based schemes due to Cramer and Shoup [36] achieve κ-ISO-CCA$\star$.

ISO$\odot$ *from LE.* As mentioned, Bellare et al. [10] showed that lossy encryption achieves ISO-CPA$\odot$ even if opening is not efficient. They furthermore showed that, aside from the Goldwasser–Micali encryption scheme (see LEEO paragraph above), LE schemes can be instantiated in the standard model from the DDH assumption, as well as from lossy trapdoor functions. As mentioned in Sect. 3.4, BY12 later updated their security notions to include multiple users and challenges, and showed that the same constructions achieve multi-user, multi-challenge ISO-CPA$\odot$.

ISO$\odot$ *from ABM-LTF.* Hofheinz [70] constructed an ABM-LTF not only from the DCR assumption, but also from pairings,which then yielded a PKE scheme achieving ISO-CCA$\odot$. As the pairing-based ABM-LTF is not known to support efficient opening, its SSO-CCA$\star$ security is left open (unlike the aforementioned DCR-based scheme).

Boyen et al. [22] constructed ABM-LTFs from LWE, yielding a PKE construction achieving ISO-CCA$\odot$. While their scheme achieves a potentially weaker notion of security from LWE compared to the concurrent work Libert et al. [94] (ISO-CCA$\odot$ vs. SSO-CCA$\odot$, see SSO section above), it does have a tight security reduction to standard hardness assumptions.

On the possibility of ISO$\circledast$*.* As both ISO$\odot$ and ISO$\star$ are achievable in the standard model, it stands to reason that a posteriori simulatability with bi-opening is as well, though such a scheme has yet to appear. Given that there are respective HPS schemes achieving ISO$\odot$ [87] and ISO$\star$ [45] in the standard model, possibly ISO$\circledast$ can be achieved from the same primitive in the standard model.

Indeed, Lai et al. [91] gave a standard model HPS construction targeting SSO-CCA under a variant of bi-opening called weak bi-opening (see Sect. 4.5); as their security notion implies κ-SSO-CCA$\star$, their construction is bound by that notion's impossibility result, and indeed their scheme requires private keys whose size is lower bounded by the total number of bits to be encrypted.

Open Problem 19. *How can a posteriori indistinguishability with bi-openings (*κ-ISO-CPA$\circledast$*/*κ-ISO-CCA$\circledast$*) be achieved in the standard model?*

A priori indistinguishability (IND) and tightness. There are of course many schemes that achieve a priori indistinguishability with openings: being polynomially equivalent to single-user, single-challenge IND-CPA (resp. IND-CCA) make the notions mainly interesting in the context of concrete security and tightness.

Early constructions of tightly secure PKE in the multi-user setting did not consider openings [50, 52, 72], yet results establishing that blackbox reductions cannot be tight do rely on receiver opening [3].

Nonetheless, only a few works have to date targeted tight a priori indistinguishability with receiver openings. The first one is a Naor–Yung [99] inspired hybrid PKE that tightly achieves κ-IND-CCA$\star$ in the programmable ROM [92]. The second is a PKE employing a generalization of HPS known as Quasi-Adaptive HPS [59], that achieves κ-IND-CCA$\star$ with an almost-tight reduction to MDDH in the standard model [58].

No scheme to date has tightly achieved IND$\star$, for either CPA or CCA, in the standard model.

Open Problem 20. *How can* κ-IND-CPA$\star$*/*κ-IND-CCA$\star$ *be achieved tightly in the standard model?*

(Note that any scheme tightly achieving ISO$\star$ would also achieve tightness under IND$\star$ via Thm. 4.)

Regarding IND$\odot$ *and* IND$\circledast$. We introduced a priori indistinguishability notions with sender and receiver opening in Sect. 3.3, and so, naturally, the degree to which tightness under these multiple-challenge-bit security notions can be achieved has not been much studied. However, given that there are schemes that tightly achieve ISO-CCA$\odot$ in the standard model [22], and given that ISO-CCA$\odot$ tightly implies β-IND-CCA$\odot$ as follows from Thm. 8 (for $\kappa = 1$, ignoring the receiver opening oracle), we conclude that the latter notion is also tightly achievable.

Corollary 3. *β-IND-CCA$\odot$ is tightly achievable in the standard model.*

Security under multi-challenge-bit indistinguishability notions has also been studied in the context of multi-instance security [11], where tightness was recently achieved in the programmable ROM by Brunetta et al. [23] from a BRE-like hybrid PKE (see Sect. 5.1). Multi-instance security notions fix one challenge bit per key (as opposed to the more general free-bit approach of our notions) and do not consider sender openings. We nonetheless conjecture that there are BRE schemes that tightly achieve (κ, β)-IND-CPA$\circledast$.

Open Problem 21. *Can we (almost-)tightly achieve (κ, β)-IND-CPA$\circledast$ in 1) the ROM? 2) the standard model? What about for* CCA*?*

Acknowledgement

The authors would like to thank Joseph Jaeger for many helpful comments and discussions.

References

1. Abdalla, M., Bellare, M., Rogaway, P.: The oracle Diffie-Hellman assumptions and an analysis of DHIES. In: Naccache, D. (ed.) CT-RSA 2001. LNCS, vol. 2020, pp. 143–158. Springer, Heidelberg (Apr 2001). https://doi.org/10.1007/3-540-45353-9_12
2. Bader, C., Hofheinz, D., Jager, T., Kiltz, E., Li, Y.: Tightly-secure authenticated key exchange. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part I. LNCS, vol. 9014, pp. 629–658. Springer, Heidelberg (Mar 2015). https://doi.org/10.1007/978-3-662-46494-6_26
3. Bader, C., Jager, T., Li, Y., Schäge, S.: On the impossibility of tight cryptographic reductions. In: Fischlin, M., Coron, J.S. (eds.) EUROCRYPT 2016, Part II. LNCS, vol. 9666, pp. 273–304. Springer, Heidelberg (May 2016). https://doi.org/10.1007/978-3-662-49896-5_10
4. Barak, B., Goldreich, O., Impagliazzo, R., Rudich, S., Sahai, A., Vadhan, S.P., Yang, K.: On the (im)possibility of obfuscating programs. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 1–18. Springer, Heidelberg (Aug 2001). https://doi.org/10.1007/3-540-44647-8_1
5. Beaver, D.: Plug and play encryption. In: Kaliski Jr., B.S. (ed.) CRYPTO'97. LNCS, vol. 1294, pp. 75–89. Springer, Heidelberg (Aug 1997). https://doi.org/10.1007/BFb0052228
6. Beaver, D., Haber, S.: Cryptographic protocols provably secure against dynamic adversaries. In: Rueppel, R.A. (ed.) EUROCRYPT'92. LNCS, vol. 658, pp. 307–323. Springer, Heidelberg (May 1993). https://doi.org/10.1007/3-540-47555-9_26
7. Bellare, M., Boldyreva, A., Micali, S.: Public-key encryption in a multi-user setting: Security proofs and improvements. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 259–274. Springer, Heidelberg (May 2000). https://doi.org/10.1007/3-540-45539-6_18
8. Bellare, M., Desai, A., Pointcheval, D., Rogaway, P.: Relations among notions of security for public-key encryption schemes. In: Krawczyk, H. (ed.) CRYPTO'98. LNCS, vol. 1462, pp. 26–45. Springer, Heidelberg (Aug 1998). https://doi.org/10.1007/BFb0055718
9. Bellare, M., Dowsley, R., Waters, B., Yilek, S.: Standard security does not imply security against selective-opening. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 645–662. Springer, Heidelberg (Apr 2012). https://doi.org/10.1007/978-3-642-29011-4_38
10. Bellare, M., Hofheinz, D., Yilek, S.: Possibility and impossibility results for encryption and commitment secure under selective opening. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 1–35. Springer, Heidelberg (Apr 2009). https://doi.org/10.1007/978-3-642-01001-9_1
11. Bellare, M., Ristenpart, T., Tessaro, S.: Multi-instance security and its application to password-based cryptography. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 312–329. Springer, Heidelberg (Aug 2012). https://doi.org/10.1007/978-3-642-32009-5_19
12. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Denning, D.E., Pyle, R., Ganesan, R., Sandhu, R.S., Ashby, V. (eds.) ACM CCS 93. pp. 62–73. ACM Press (Nov 1993). https://doi.org/10.1145/168588.168596
13. Bellare, M., Rogaway, P.: Optimal asymmetric encryption. In: Santis, A.D. (ed.) EUROCRYPT'94. LNCS, vol. 950, pp. 92–111. Springer, Heidelberg (May 1995). https://doi.org/10.1007/BFb0053428

14. Bellare, M., Yilek, S.: Encryption schemes secure under selective opening attack. Cryptology ePrint Archive, Report 2009/101 (original full version) (2009), https://eprint.iacr.org/2009/101, version 20090302:083605
15. Bellare, M., Yilek, S.: Encryption schemes secure under selective opening attack. Cryptology ePrint Archive, Report 2009/101 (2009), https://eprint.iacr.org/2009/101
16. Bellare, M., Yilek, S.: Encryption schemes secure under selective opening attack. Cryptology ePrint Archive, Report 2009/101 (updated full version) (2012), https://eprint.iacr.org/2009/101, version 20120923:212424
17. Benhamouda, F., Gentry, C., Gorbunov, S., Halevi, S., Krawczyk, H., Lin, C., Rabin, T., Reyzin, L.: Can a public blockchain keep a secret? In: Pass, R., Pietrzak, K. (eds.) TCC 2020, Part I. LNCS, vol. 12550, pp. 260–290. Springer, Heidelberg (Nov 2020). https://doi.org/10.1007/978-3-030-64375-1_10
18. Black, J., Rogaway, P., Shrimpton, T., Stam, M.: An analysis of the blockcipher-based hash functions from PGV. Journal of Cryptology **23**(4), 519–545 (Oct 2010). https://doi.org/10.1007/s00145-010-9071-0
19. Böhl, F., Hofheinz, D., Kraschewski, D.: On definitions of selective opening security. In: Fischlin, M., Buchmann, J., Manulis, M. (eds.) PKC 2012. LNCS, vol. 7293, pp. 522–539. Springer, Heidelberg (May 2012). https://doi.org/10.1007/978-3-642-30057-8_31
20. Boneh, D., Dagdelen, Ö., Fischlin, M., Lehmann, A., Schaffner, C., Zhandry, M.: Random oracles in a quantum world. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 41–69. Springer, Heidelberg (Dec 2011). https://doi.org/10.1007/978-3-642-25385-0_3
21. Boneh, D., Lewi, K., Montgomery, H.W., Raghunathan, A.: Key homomorphic PRFs and their applications. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part I. LNCS, vol. 8042, pp. 410–428. Springer, Heidelberg (Aug 2013). https://doi.org/10.1007/978-3-642-40041-4_23
22. Boyen, X., Li, Q.: All-but-many lossy trapdoor functions from lattices and applications. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part III. LNCS, vol. 10403, pp. 298–331. Springer, Heidelberg (Aug 2017). https://doi.org/10.1007/978-3-319-63697-9_11
23. Brunetta, C., Heum, H., Stam, M.: Multi-instance secure public-key encryption. In: Boldyreva, A., Kolesnikov, V. (eds.) PKC 2023, Part II. LNCS, vol. 13941, pp. 336–367. Springer, Heidelberg (May 2023). https://doi.org/10.1007/978-3-031-31371-4_12
24. Camenisch, J., Lehmann, A., Neven, G., Samelin, K.: Virtual smart cards: How to sign with a password and a server. In: Zikas, V., De Prisco, R. (eds.) SCN 16. LNCS, vol. 9841, pp. 353–371. Springer, Heidelberg (Aug / Sep 2016). https://doi.org/10.1007/978-3-319-44618-9_19
25. Camenisch, J., Lehmann, A., Neven, G., Samelin, K.: UC-secure non-interactive public-key encryption. In: Köpf, B., Chong, S. (eds.) CSF 2017 Computer Security Foundations Symposium. pp. 217–233. IEEE Computer Society Press (2017). https://doi.org/10.1109/CSF.2017.14
26. Camenisch, J., Shoup, V.: Practical verifiable encryption and decryption of discrete logarithms. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 126–144. Springer, Heidelberg (Aug 2003). https://doi.org/10.1007/978-3-540-45146-4_8
27. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. Cryptology ePrint Archive, Report 2000/067 (2000), https://eprint.iacr.org/2000/067
28. Canetti, R., Dwork, C., Naor, M., Ostrovsky, R.: Deniable encryption. In: Kaliski Jr., B.S. (ed.) CRYPTO'97. LNCS, vol. 1294, pp. 90–104. Springer, Heidelberg (Aug 1997). https://doi.org/10.1007/BFb0052229
29. Canetti, R., Feige, U., Goldreich, O., Naor, M.: Adaptively secure multi-party computation. In: 28th ACM STOC. pp. 639–648. ACM Press (May 1996). https://doi.org/10.1145/237814.238015
30. Canetti, R., Halevi, S., Katz, J.: Chosen-ciphertext security from identity-based encryption. In: Cachin, C., Camenisch, J. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 207–222. Springer, Heidelberg (May 2004). https://doi.org/10.1007/978-3-540-24676-3_13
31. Canetti, R., Halevi, S., Katz, J.: Adaptively-secure, non-interactive public-key encryption. In: Kilian, J. (ed.) TCC 2005. LNCS, vol. 3378, pp. 150–168. Springer, Heidelberg (Feb 2005). https://doi.org/10.1007/978-3-540-30576-7_9
32. Canetti, R., Krawczyk, H., Nielsen, J.B.: Relaxing chosen-ciphertext security. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 565–582. Springer, Heidelberg (Aug 2003). https://doi.org/10.1007/978-3-540-45146-4_33
33. Canetti, R., Park, S., Poburinnaya, O.: Fully deniable interactive encryption. In: Micciancio, D., Ristenpart, T. (eds.) CRYPTO 2020, Part I. LNCS, vol. 12170, pp. 807–835. Springer, Heidelberg (Aug 2020). https://doi.org/10.1007/978-3-030-56784-2_27
34. Coladangelo, A., Goldwasser, S., Vazirani, U.V.: Deniable encryption in a quantum world. In: Leonardi, S., Gupta, A. (eds.) STOC'22. pp. 1378–1391. ACM (2022). https://doi.org/10.1145/3519935.3520019
35. Cramer, R., Shoup, V.: A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: Krawczyk, H. (ed.) CRYPTO'98. LNCS, vol. 1462, pp. 13–25. Springer, Heidelberg (Aug 1998). https://doi.org/10.1007/BFb0055717
36. Cramer, R., Shoup, V.: Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 45–64. Springer, Heidelberg (Apr / May 2002). https://doi.org/10.1007/3-540-46035-7_4

37. Cramer, R., Shoup, V.: Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. SIAM Journal on Computing **33**(1), 167–226 (2003). https://doi.org/10.1137/S0097539702403773
38. Damgård, I., Nielsen, J.B.: Improved non-committing encryption schemes based on a general complexity assumption. In: Bellare, M. (ed.) CRYPTO 2000. LNCS, vol. 1880, pp. 432–450. Springer, Heidelberg (Aug 2000). https://doi.org/10.1007/3-540-44598-6_27
39. Das, A., Dutta, S., Adhikari, A.: Indistinguishability against chosen ciphertext verification attack revisited: The complete picture. In: Susilo, W., Reyhanitabar, R. (eds.) ProvSec 2013. LNCS, vol. 8209, pp. 104–120. Springer, Heidelberg (Oct 2013). https://doi.org/10.1007/978-3-642-41227-1_6
40. Dodis, Y., Grubbs, P., Ristenpart, T., Woodage, J.: Fast message franking: From invisible salamanders to encryptment. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018, Part I. LNCS, vol. 10991, pp. 155–186. Springer, Heidelberg (Aug 2018). https://doi.org/10.1007/978-3-319-96884-1_6
41. Dwork, C., Naor, M., Reingold, O., Stockmeyer, L.J.: Magic functions. In: 40th FOCS. pp. 523–534. IEEE Computer Society Press (Oct 1999). https://doi.org/10.1109/SFFCS.1999.814626
42. Dziembowski, S.: Intrusion-resilience via the bounded-storage model. In: Halevi, S., Rabin, T. (eds.) TCC 2006. LNCS, vol. 3876, pp. 207–224. Springer, Heidelberg (Mar 2006). https://doi.org/10.1007/11681878_11
43. ElGamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. In: Blakley, G.R., Chaum, D. (eds.) CRYPTO'84. LNCS, vol. 196, pp. 10–18. Springer, Heidelberg (Aug 1984)
44. Escala, A., Herold, G., Kiltz, E., Ràfols, C., Villar, J.: An algebraic framework for Diffie-Hellman assumptions. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013, Part II. LNCS, vol. 8043, pp. 129–147. Springer, Heidelberg (Aug 2013). https://doi.org/10.1007/978-3-642-40084-1_8
45. Fehr, S., Hofheinz, D., Kiltz, E., Wee, H.: Encryption schemes secure against chosen-ciphertext selective opening attacks. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 381–402. Springer, Heidelberg (May / Jun 2010). https://doi.org/10.1007/978-3-642-13190-5_20
46. Fischlin, M., Lehmann, A., Ristenpart, T., Shrimpton, T., Stam, M., Tessaro, S.: Random oracles with(out) programmability. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 303–320. Springer, Heidelberg (Dec 2010). https://doi.org/10.1007/978-3-642-17373-8_18
47. Fuchsbauer, G., Heuer, F., Kiltz, E., Pietrzak, K.: Standard security does imply security against selective opening for Markov distributions. In: Kushilevitz, E., Malkin, T. (eds.) TCC 2016-A, Part I. LNCS, vol. 9562, pp. 282–305. Springer, Heidelberg (Jan 2016). https://doi.org/10.1007/978-3-662-49096-9_12
48. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. In: Wiener, M.J. (ed.) CRYPTO'99. LNCS, vol. 1666, pp. 537–554. Springer, Heidelberg (Aug 1999). https://doi.org/10.1007/3-540-48405-1_34
49. Garay, J.A., Wichs, D., Zhou, H.S.: Somewhat non-committing encryption and efficient adaptively secure oblivious transfer. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 505–523. Springer, Heidelberg (Aug 2009). https://doi.org/10.1007/978-3-642-03356-8_30
50. Gay, R., Hofheinz, D., Kohl, L.: Kurosawa-desmedt meets tight security. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part III. LNCS, vol. 10403, pp. 133–160. Springer, Heidelberg (Aug 2017). https://doi.org/10.1007/978-3-319-63697-9_5
51. Gellert, K., Jager, T., Lyu, L., Neuschulten, T.: On fingerprinting attacks and length-hiding encryption. In: Galbraith, S.D. (ed.) CT-RSA 2022. LNCS, vol. 13161, pp. 345–369. Springer, Heidelberg (Mar 2022). https://doi.org/10.1007/978-3-030-95312-6_15
52. Giacon, F., Kiltz, E., Poettering, B.: Hybrid encryption in a multi-user setting, revisited. In: Abdalla, M., Dahab, R. (eds.) PKC 2018, Part I. LNCS, vol. 10769, pp. 159–189. Springer, Heidelberg (Mar 2018). https://doi.org/10.1007/978-3-319-76578-5_6
53. Goldreich, O.: Foundations of Cryptography: Basic Tools, vol. 1. Cambridge University Press, Cambridge, UK (2001)
54. Goldwasser, S., Micali, S.: Probabilistic encryption and how to play mental poker keeping secret all partial information. In: 14th ACM STOC. pp. 365–377. ACM Press (May 1982). https://doi.org/10.1145/800070.802212
55. Goldwasser, S., Micali, S.: Probabilistic encryption. Journal of Computer and System Sciences **28**(2), 270–299 (1984)
56. Grubbs, P., Lu, J., Ristenpart, T.: Message franking via committing authenticated encryption. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part III. LNCS, vol. 10403, pp. 66–97. Springer, Heidelberg (Aug 2017). https://doi.org/10.1007/978-3-319-63697-9_3
57. Han, S., Liu, S., Gu, D.: Key encapsulation mechanism with tight enhanced security in the multi-user setting: Impossibility result and optimal tightness. In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021, Part II. LNCS, vol. 13091, pp. 483–513. Springer, Heidelberg (Dec 2021). https://doi.org/10.1007/978-3-030-92075-3_17
58. Han, S., Liu, S., Gu, D.: Almost tight multi-user security under adaptive corruptions & leakages in the standard model. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part III. LNCS, vol. 14006, pp. 132–162. Springer, Heidelberg (Apr 2023). https://doi.org/10.1007/978-3-031-30620-4_5

59. Han, S., Liu, S., Lyu, L., Gu, D.: Tight leakage-resilient CCA-security from quasi-adaptive hash proof system. In: Boldyreva, A., Micciancio, D. (eds.) CRYPTO 2019, Part II. LNCS, vol. 11693, pp. 417–447. Springer, Heidelberg (Aug 2019). https://doi.org/10.1007/978-3-030-26951-7_15
60. Hara, K., Kitagawa, F., Matsuda, T., Hanaoka, G., Tanaka, K.: Simulation-based receiver selective opening CCA secure PKE from standard computational assumptions. In: Catalano, D., De Prisco, R. (eds.) SCN 18. LNCS, vol. 11035, pp. 140–159. Springer, Heidelberg (Sep 2018). https://doi.org/10.1007/978-3-319-98113-0_8
61. Hazay, C., López-Alt, A., Wee, H., Wichs, D.: Leakage-resilient cryptography from minimal assumptions. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 160–176. Springer, Heidelberg (May 2013). https://doi.org/10.1007/978-3-642-38348-9_10
62. Hazay, C., Patra, A.: Efficient one-sided adaptively secure computation. Journal of Cryptology **30**(1), 321–371 (Jan 2017). https://doi.org/10.1007/s00145-015-9222-4
63. Hazay, C., Patra, A., Warinschi, B.: Selective opening security for receivers. In: Iwata, T., Cheon, J.H. (eds.) ASIACRYPT 2015, Part I. LNCS, vol. 9452, pp. 443–469. Springer, Heidelberg (Nov / Dec 2015). https://doi.org/10.1007/978-3-662-48797-6_19
64. Hemenway, B., Libert, B., Ostrovsky, R., Vergnaud, D.: Lossy encryption: Constructions from general assumptions and efficient selective opening chosen ciphertext security. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 70–88. Springer, Heidelberg (Dec 2011). https://doi.org/10.1007/978-3-642-25385-0_4
65. Heuer, F.: On the selective opening security of public-key encryption. Doctoral thesis, Ruhr-Universität Bochum, Universitätsbibliothek (2017)
66. Heuer, F., Jager, T., Kiltz, E., Schäge, S.: On the selective opening security of practical public-key encryption schemes. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 27–51. Springer, Heidelberg (Mar / Apr 2015). https://doi.org/10.1007/978-3-662-46447-2_2
67. Heuer, F., Poettering, B.: Selective opening security from simulatable data encapsulation. In: Cheon, J.H., Takagi, T. (eds.) ASIACRYPT 2016, Part II. LNCS, vol. 10032, pp. 248–277. Springer, Heidelberg (Dec 2016). https://doi.org/10.1007/978-3-662-53890-6_9
68. Heum, H., Stam, M.: Tightness subtleties for multi-user pke notions. In: Paterson, M.B. (ed.) Cryptography and Coding. pp. 75–104. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-92641-0_5
69. Hofheinz, D.: Possibility and impossibility results for selective decommitments. Journal of Cryptology **24**(3), 470–516 (Jul 2011). https://doi.org/10.1007/s00145-010-9066-x
70. Hofheinz, D.: All-but-many lossy trapdoor functions. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 209–227. Springer, Heidelberg (Apr 2012). https://doi.org/10.1007/978-3-642-29011-4_14
71. Hofheinz, D., Hövelmanns, K., Kiltz, E.: A modular analysis of the Fujisaki-Okamoto transformation. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 341–371. Springer, Heidelberg (Nov 2017). https://doi.org/10.1007/978-3-319-70500-2_12
72. Hofheinz, D., Jager, T.: Tightly secure signatures and public-key encryption. In: Safavi-Naini, R., Canetti, R. (eds.) CRYPTO 2012. LNCS, vol. 7417, pp. 590–607. Springer, Heidelberg (Aug 2012). https://doi.org/10.1007/978-3-642-32009-5_35
73. Hofheinz, D., Jager, T., Rupp, A.: Public-key encryption with simulation-based selective-opening security and compact ciphertexts. In: Hirt, M., Smith, A.D. (eds.) TCC 2016-B, Part II. LNCS, vol. 9986, pp. 146–168. Springer, Heidelberg (Oct / Nov 2016). https://doi.org/10.1007/978-3-662-53644-5_6
74. Hofheinz, D., Kiltz, E.: Secure hybrid encryption from weakened key encapsulation. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 553–571. Springer, Heidelberg (Aug 2007). https://doi.org/10.1007/978-3-540-74143-5_31
75. Hofheinz, D., Müller-Quade, J., Steinwandt, R.: On modeling IND-CCA security in cryptographic protocols. Cryptology ePrint Archive, Report 2003/024 (2003), https://eprint.iacr.org/2003/024
76. Hofheinz, D., Rao, V., Wichs, D.: Standard security does not imply indistinguishability under selective opening. In: Hirt, M., Smith, A.D. (eds.) TCC 2016-B, Part II. LNCS, vol. 9986, pp. 121–145. Springer, Heidelberg (Oct / Nov 2016). https://doi.org/10.1007/978-3-662-53644-5_5
77. Hofheinz, D., Rupp, A.: Standard versus selective opening security: Separation and equivalence results. In: Lindell, Y. (ed.) TCC 2014. LNCS, vol. 8349, pp. 591–615. Springer, Heidelberg (Feb 2014). https://doi.org/10.1007/978-3-642-54242-8_25
78. Huang, Z., Lai, J., Chen, W., Au, M.H., Peng, Z., Li, J.: Simulation-based selective opening security for receivers under chosen-ciphertext attacks. Des. Codes Cryptogr. **87**(6), 1345–1371 (2019)
79. Huang, Z., Lai, J., Chen, W., ul Haq, M.R., Jiang, L.: Practical public key encryption with selective opening security for receivers. Information Sciences **478**, 15–27 (2019)
80. Huang, Z., Lai, J., Han, S., Lyu, L., Weng, J.: Anonymous public key encryption under corruptions. In: Agrawal, S., Lin, D. (eds.) ASIACRYPT 2022, Part III. LNCS, vol. 13793, pp. 423–453. Springer, Heidelberg (Dec 2022). https://doi.org/10.1007/978-3-031-22969-5_15
81. Huang, Z., Liu, S., Mao, X., Chen, K.: Non-malleability under selective opening attacks: Implication and separation. In: Malkin, T., Kolesnikov, V., Lewko, A.B., Polychronakis, M. (eds.) ACNS 15. LNCS, vol. 9092, pp. 87–104. Springer, Heidelberg (Jun 2015). https://doi.org/10.1007/978-3-319-28166-7_5

82. Jaeger, J.: Let attackers program ideal models: Modularity and composability for adaptive compromise. In: Hazay, C., Stam, M. (eds.) EUROCRYPT 2023, Part III. LNCS, vol. 14006, pp. 101–131. Springer, Heidelberg (Apr 2023). https://doi.org/10.1007/978-3-031-30620-4_4
83. Jaeger, J.: Personal communication (2023)
84. Jager, T., Kiltz, E., Riepel, D., Schäge, S.: Tightly-secure authenticated key exchange, revisited. In: Canteaut, A., Standaert, F.X. (eds.) EUROCRYPT 2021, Part I. LNCS, vol. 12696, pp. 117–146. Springer, Heidelberg (Oct 2021). https://doi.org/10.1007/978-3-030-77870-5_5
85. Jager, T., Stam, M., Stanley-Oakes, R., Warinschi, B.: Multi-key authenticated encryption with corruptions: Reductions are lossy. In: Kalai, Y., Reyzin, L. (eds.) TCC 2017, Part I. LNCS, vol. 10677, pp. 409–441. Springer, Heidelberg (Nov 2017). https://doi.org/10.1007/978-3-319-70500-2_14
86. Jia, D., Lu, X., Li, B.: Receiver selective opening security from indistinguishability obfuscation. In: Dunkelman, O., Sanadhya, S.K. (eds.) INDOCRYPT 2016. LNCS, vol. 10095, pp. 393–410. Springer, Heidelberg (Dec 2016). https://doi.org/10.1007/978-3-319-49890-4_22
87. Jia, D., Lu, X., Li, B.: Constructions secure against receiver selective opening and chosen ciphertext attacks. In: Handschuh, H. (ed.) CT-RSA 2017. LNCS, vol. 10159, pp. 417–431. Springer, Heidelberg (Feb 2017). https://doi.org/10.1007/978-3-319-52153-4_24
88. Joye, M., Quisquater, J.J., Yung, M.: On the power of misbehaving adversaries and security analysis of the original EPOC. In: Naccache, D. (ed.) CT-RSA 2001. LNCS, vol. 2020, pp. 208–222. Springer, Heidelberg (Apr 2001). https://doi.org/10.1007/3-540-45353-9_16
89. Krawczyk, H.: The joy of cryptography: A personal journey (2023), https://crypto.iacr.org/2023/files/667slides.pdf, IACR Distinguished Lecture, presented at Crypto'23
90. Küsters, R., Tuengerthal, M.: Joint state theorems for public-key encryption and digital signature functionalities with local computation. In: Sabelfeld, A. (ed.) CSF 2008 Computer Security Foundations Symposium. pp. 270–284. IEEE Computer Society Press (2008). https://doi.org/10.1109/CSF.2008.18
91. Lai, J., Yang, R., Huang, Z., Weng, J.: Simulation-based bi-selective opening security for public key encryption. In: Tibouchi, M., Wang, H. (eds.) ASIACRYPT 2021, Part II. LNCS, vol. 13091, pp. 456–482. Springer, Heidelberg (Dec 2021). https://doi.org/10.1007/978-3-030-92075-3_16
92. Lee, Y., Lee, D.H., Park, J.H.: Tightly cca-secure encryption scheme in a multi-user setting with corruptions. Des. Codes Cryptogr. **88**(11), 2433–2452 (2020)
93. Lei, F., Chen, W., Chen, K.: A non-committing encryption scheme based on quadratic residue. In: International Symposium on Computer and Information Sciences. pp. 972–980. Springer (2006)
94. Libert, B., Sakzad, A., Stehlé, D., Steinfeld, R.: All-but-many lossy trapdoor functions and selective opening chosen-ciphertext security from LWE. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017, Part III. LNCS, vol. 10403, pp. 332–364. Springer, Heidelberg (Aug 2017). https://doi.org/10.1007/978-3-319-63697-9_12
95. Liu, S., Paterson, K.G.: Simulation-based selective opening CCA security for PKE from key encapsulation mechanisms. In: Katz, J. (ed.) PKC 2015. LNCS, vol. 9020, pp. 3–26. Springer, Heidelberg (Mar / Apr 2015). https://doi.org/10.1007/978-3-662-46447-2_1
96. Lu, Y., Hara, K., Tanaka, K.: Receiver selective opening CCA secure public key encryption from various assumptions. In: Nguyen, K., Wu, W., Lam, K.Y., Wang, H. (eds.) ProvSec 2020. LNCS, vol. 12505, pp. 213–233. Springer, Heidelberg (Nov / Dec 2020). https://doi.org/10.1007/978-3-030-62576-4_11
97. Lyu, L., Liu, S., Han, S., Gu, D.: Tightly SIM-SO-CCA secure public key encryption from standard assumptions. In: Abdalla, M., Dahab, R. (eds.) PKC 2018, Part I. LNCS, vol. 10769, pp. 62–92. Springer, Heidelberg (Mar 2018). https://doi.org/10.1007/978-3-319-76578-5_3
98. Micali, S., Rackoff, C., Sloan, B.: The notion of security for probabilistic cryptosystems. In: Odlyzko, A.M. (ed.) CRYPTO'86. LNCS, vol. 263, pp. 381–392. Springer, Heidelberg (Aug 1987). https://doi.org/10.1007/3-540-47721-7_27
99. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: 22nd ACM STOC. pp. 427–437. ACM Press (May 1990). https://doi.org/10.1145/100216.100273
100. Nielsen, J.B.: Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 111–126. Springer, Heidelberg (Aug 2002). https://doi.org/10.1007/3-540-45708-9_8
101. Okamoto, T., Pointcheval, D.: REACT: Rapid Enhanced-security Asymmetric Cryptosystem Transform. In: Naccache, D. (ed.) CT-RSA 2001. LNCS, vol. 2020, pp. 159–175. Springer, Heidelberg (Apr 2001). https://doi.org/10.1007/3-540-45353-9_13
102. Ostrovsky, R., Rao, V., Visconti, I.: On selective-opening attacks against encryption schemes. In: Abdalla, M., Prisco, R.D. (eds.) SCN 14. LNCS, vol. 8642, pp. 578–597. Springer, Heidelberg (Sep 2014). https://doi.org/10.1007/978-3-319-10879-7_33
103. Pan, J., Zeng, R.: Compact and tightly selective-opening secure public-key encryption schemes. In: Agrawal, S., Lin, D. (eds.) ASIACRYPT 2022, Part III. LNCS, vol. 13793, pp. 363–393. Springer, Heidelberg (Dec 2022). https://doi.org/10.1007/978-3-031-22969-5_13
104. Peikert, C., Waters, B.: Lossy trapdoor functions and their applications. In: Ladner, R.E., Dwork, C. (eds.) 40th ACM STOC. pp. 187–196. ACM Press (May 2008). https://doi.org/10.1145/1374376.1374406
105. Shamir, A.: How to share a secret. Communications of the Association for Computing Machinery **22**(11), 612–613 (Nov 1979)

106. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Blakley, G.R., Chaum, D. (eds.) CRYPTO'84. LNCS, vol. 196, pp. 47–53. Springer, Heidelberg (Aug 1984)
107. Shannon, C.E.: Communication theory of secrecy systems. Bell Systems Technical Journal **28**(4), 656–715 (1949)
108. Shoup, V.: Lower bounds for discrete logarithms and related problems. In: Fumy, W. (ed.) EUROCRYPT'97. LNCS, vol. 1233, pp. 256–266. Springer, Heidelberg (May 1997). `https://doi.org/10.1007/3-540-69053-0_18`
109. Steinfeld, R., Baek, J., Zheng, Y.: On the necessity of strong assumptions for the security of a class of asymmetric encryption schemes. In: Batten, L.M., Seberry, J. (eds.) ACISP 02. LNCS, vol. 2384, pp. 241–256. Springer, Heidelberg (Jul 2002). `https://doi.org/10.1007/3-540-45450-0_20`
110. Tezcan, C., Vaudenay, S.: On hiding a plaintext length by preencryption. In: Lopez, J., Tsudik, G. (eds.) ACNS 11. LNCS, vol. 6715, pp. 345–358. Springer, Heidelberg (Jun 2011). `https://doi.org/10.1007/978-3-642-21554-4_20`
111. Watanabe, Y., Shikata, J., Imai, H.: Equivalence between semantic security and indistinguishability against chosen ciphertext attacks. In: Desmedt, Y. (ed.) PKC 2003. LNCS, vol. 2567, pp. 71–84. Springer, Heidelberg (Jan 2003). `https://doi.org/10.1007/3-540-36288-6_6`
112. Winternitz, R.S.: A secure one-way hash function built from DES. In: IEEE Symposium on Security and Privacy. pp. 88–90. IEEE Computer Society (1984). `https://doi.org/10.1109/SP.1984.10027`
113. Yang, R., Lai, J., Huang, Z., Au, M.H., Xu, Q., Susilo, W.: Possibility and impossibility results for receiver selective opening secure PKE in the multi-challenge setting. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020, Part I. LNCS, vol. 12491, pp. 191–220. Springer, Heidelberg (Dec 2020). `https://doi.org/10.1007/978-3-030-64837-4_7`

A ISO Tightly Implies IND with Bi-openings

From the fact that (κ,β)-IND-CCA$\circledast$ is implied by κ-IND-CCA$\star$ with a β loss (Thm. 3), we concluded in Sect. 3.4 that κ-ISO-CCA$\star$ implies (κ,β)-IND-CCA$\circledast$ with a $2 \cdot \beta$ security loss. We next show that there is a message sampler such that the reduction only loses a factor 2.

Lemma 6. *Let $\mathsf{M}_{\langle s\rangle}$ be as given in Fig. 19 (left): its input α is interpeted as (j, m_0, m_1), where the message pair m_0, m_1 is subject to $|m_0| = |m_1|$, and index j is subject to $j \in [\beta]$. On first invocation (when $s = \varepsilon$), it draws $s \leftarrow\$ \{0,1\}^\beta$ and, on all invocations, on input $\alpha = (j, m_0, m_1)$, it returns $m_{s[j]}$. Let $\mathring{\mathsf{S}}$ be its ideal resampler.*

Consider S (Fig. 19, right) that on input $(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$, for each compromised call sets $s'[j] \leftarrow s[j]$ for the relevant j, and for uncompromised j draws $s'[j] \leftarrow\$ \{0,1\}$, and appends $m_{s'[j]}$ to the resampled message vector $\mathtt{M}^1$; and, once this process is completed for all challenge calls, returns $\mathtt{M}^1$.

Then $\mathsf{S} = \mathring{\mathsf{S}}$.

Proof. By inspection.

Theorem 8. *Let $\mathsf{PKE}[\lambda]$ be given, and let $\mathsf{M}_{\langle s\rangle}$ be as given in Lemma 6. Then there is a type-preserving black-box reduction $\mathbb{B}_{\text{iso}}$ such that, for all $\mathbb{A}_{\text{ind}}$,*

$$\mathsf{Adv}^{(\kappa,\beta)\text{-ind-cca}\circledast}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\text{ind}}) \leq 2 \cdot \mathsf{Adv}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{B}_{\text{iso}})\,,$$

where $\mathbb{B}_{\text{iso}}$ calls $\mathbb{A}_{\text{ind}}$ once.

The proof follows in the vein of that of Thm. 4, only updated to accomodate the message sampler given in Lemma 6 and the more general experiments.

Proof. Without loss of generality, we may assume that the adversary $\mathbb{A}_{\text{ind}}$ does not call $\mathcal{E}_{\mathbb{A}}(i, j, m_0, m_1)$ with either $|m_0| \neq |m_1|$ or $m_0 = m_1$, and that if $\mathbb{A}_{\text{ind}}$ outputs guess $(j, \hat{b}_j)$ then $\mathbb{A}_{\text{ind}}$ made a call to $\mathcal{E}$ with bit handle j at least once, and did not compromise b_j through calls to its opening oracles. Technically, one can create an intermediate reduction $\mathbb{B}_{\text{ind}}$ that runs $\mathbb{A}_{\text{ind}}$ and, playing the same game, forwards everything to its own oracles but those pointless calls to $\mathcal{E}$ (which it can easily simulate), and in the event that $\mathbb{A}_{\text{ind}}$ either halts with an index j that has not been challenged, or halts with an index j that has been compromised, or tries to compromise the last uncompromised challenge bit, $\mathbb{B}_{\text{ind}}$ chooses a bit handle j' that has been challenged at least once (making an extra call to $\mathcal{E}$ in the case that $\mathbb{A}_{\text{ind}}$ halts without ever calling $\mathcal{E}$), samples $\hat{b}_{j'} \leftarrow\$ \{0,1\}$, and terminates with the output $(j', \hat{b}_{j'})$. By inspection, $\mathbb{B}_{\text{ind}}$'s advantage is at least $\mathbb{A}_{\text{ind}}$'s.

```
Sampler $\mathsf{M}_{\langle s\rangle}(j, m_0, m_1)$
if $s = \varepsilon : s \leftarrow\$ \{0,1\}^{\beta}$
return $m_{s[j]}$
```

```
Resampler $\mathsf{S}(\mathtt{A}, \mathtt{L}, \mathcal{J}, \mathtt{M}^0[\mathcal{J}])$
for $k \in \mathcal{J}$ :
    $(j, m_0, m_1) \leftarrow \mathtt{A}[k]$
    if $s'[j] = \varepsilon \wedge m_0 \neq m_1$ :
        if $\exists b \in \{0,1\}$ s.t. $\mathtt{M}^0[k] = m_b : s'[j] \leftarrow b$
        else return ↯
for $k \in |\mathtt{A}|$ :
    $(j, m_0, m_1) \leftarrow \mathtt{A}[k]$
    if $s'[j] = \varepsilon : s'[j] \leftarrow\$ \{0,1\}$
    $\mathtt{M}^1 \overset{\frown}{\longleftarrow} m_{s'[j]}$
return $\mathtt{M}^1$
```

Fig. 19. The M and S of Lemma 6.

```
Reduction $\mathbb{B}_{\text{iso}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
$k \leftarrow 0$
$(j, \hat{s}[j]) \leftarrow\$ \mathbb{A}_{\text{ind}}^{\mathcal{E},\mathcal{D},(\mathcal{T},)\mathcal{S},\mathcal{R}}(\mathsf{pk}_1, \ldots, \mathsf{pk}_\kappa)$
if $k_j = \varepsilon : \hat{b} \leftarrow\$ \{0,1\}$
else : $\mathtt{M}^b \leftarrow \mathcal{C}$
    $(j, m_0, m_1) \leftarrow \mathtt{A}[k_j]$
    if $\mathtt{M}^b[k_j] = m_{\hat{s}[j]} : \hat{b} \leftarrow 0$
    else : $\hat{b} \leftarrow 1$
return $\hat{b}$
```

```
If $\mathbb{A}_{\text{ind}}$ calls $\mathcal{E}(i, j, m_0, m_1)$
if $|m_0| \neq |m_1|$ : return ↯
$k \leftarrow k + 1, k_j \leftarrow k$
$\alpha \leftarrow (j, m_0, m_1), \mathtt{A} \overset{\frown}{\longleftarrow} \alpha$
$c \leftarrow \mathcal{E}_{\mathbb{B}}(i, \alpha)$
return $c$
```

Fig. 20. The reduction $\mathbb{B}_{\text{iso}}$ simulating $\mathsf{Exp}_{\mathsf{PKE}[\lambda]}^{(\kappa,\beta)\text{-ind-cca}\circledast}$ for $\mathbb{A}_{\text{ind}}$. The decryption oracle and opening oracles are simply forwarded.

Let $\mathbb{B}_{\text{iso}}$ be as given in Fig. 20: if $\mathbb{A}_{\text{ind}}$ calls $\mathcal{E}_{\mathbb{A}}(i, j, m_0, m_1)$, subject to both $|m_0| = |m_1|$ and $m_0 \neq m_1$, it sets $\alpha = (j, m_0, m_1)$ and calls $\mathcal{E}_{\mathbb{B}}(i, \alpha)$, returning the resulting c; if $\mathbb{A}_{\text{ind}}$ calls any other oracles, $\mathbb{B}_{\text{iso}}$ forwards the call and returns the result. When $\mathbb{A}_{\text{ind}}$ halts with a guess $\hat{s}[j]$, $\mathbb{B}_{\text{iso}}$ calls $\mathcal{C}$ and receives $\mathtt{M}^b$. Since $\mathbb{B}_{\text{iso}}$ can maintain its own perfect copy of $\mathtt{A}$, it can check whether calls made to bit handle j are consistent with the guess $\hat{s}[j]$ (checking one such call suffices, as the rest are guaranteed to be consistent with it). If so, $\mathbb{B}_{\text{iso}}$ halts with output $\hat{b} = 0$ (indicating a guess that the returned messages were the real ones), otherwise $\mathbb{B}_{\text{iso}}$ halts with output $\hat{b} = 1$.

We can rephrase $\mathbb{B}_{\text{iso}}$'s distinguishing advantage

$$\mathsf{Adv}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}^{\kappa\text{-iso-cca}\circledast}(\mathbb{B}_{\text{iso}}) = \Pr\left[\hat{b} = 0 \;\middle|\; b = 0\right] - \Pr\left[\hat{b} = 0 \;\middle|\; b = 1\right]$$

and analyse each term individually. Based on $\mathbb{B}_{\text{iso}}$'s description, and given the assumptions that j was used as part of at least one challenge oracle call, the event $\hat{b} = 0$ is equivalent to the event $\mathtt{M}^b[k_j] = m_{\hat{s}[j]}$, where k_j represents the last call to $\mathcal{E}$ using bit handle j, and m_0, m_1 are the messages that were provided for that call.

If $b = 0$, then $\mathtt{M}^0[k_j] = m_{s[j]}$, and so (given our assumption that calls are not made with $m_0 = m_1$) the first term is equivalent to $\Pr[\hat{s}[j] = s[j] \mid b = 0]$. At this point the conditional becomes irrelevant, as it is independent of both $\hat{s}$ and s (jointly). Finally, $\Pr[\hat{s}[j] = s[j]]$ equals $\Pr\left[\mathsf{Exp}_{\mathsf{PKE}[\lambda]}^{(\kappa,\beta)\text{-ind-cca}\circledast}(\mathbb{A}_{\text{ind}})\right]$ as, by design of $\mathbb{B}_{\text{iso}}$ and M, $\mathbb{A}_{\text{iso}}$ is provided with an environment that perfectly matches that of $\mathsf{Exp}_{\mathsf{PKE}[\lambda]}^{(\kappa,\beta)\text{-ind-cca}\circledast}(\cdot)$, where the individual bits of M's bit string s play the role of the challenge bits, one free choice of which $\mathbb{A}_{\text{ind}}$ has to guess. Thus,

$$\Pr\left[\hat{b} = 0 \;\middle|\; b = 0\right] = \Pr\left[\mathsf{Exp}_{\mathsf{PKE}[\lambda]}^{(\kappa,\beta)\text{-ind-cca}\circledast}(\mathbb{A}_{\text{ind}})\right].$$

If $b = 1$, then $\mathtt{M}^1[k_j] = m_{s'[j]}$, so the second term is equivalent to $\Pr[\hat{s}[j] = s'[j] \mid b = 1]$. As we assumed $\mathbb{A}_{\text{ind}}$ did not make opening oracle calls that would compromise $s[j]$, $\mathring{\mathsf{S}}$ will have drawn $s'[j]$

uniformly at random, independently of $\hat{s}[j]$. Thus,

$$\Pr\left[\hat{b} = 0 \,\middle|\, b = 1\right] = \frac{1}{2}\,.$$

Putting the pieces together, we obtain

$$\begin{aligned} 2 \cdot \mathsf{Adv}^{\kappa\text{-iso-cca}\circledast}_{\mathsf{PKE}[\lambda],\mathsf{M},\mathring{\mathsf{S}}}(\mathbb{B}_{\text{iso}}) &= 2 \cdot \Pr\left[\mathsf{Exp}^{(\kappa,\beta)\text{-ind-cca}\circledast}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\text{ind}})\right] - 1 \\ &\geq \mathsf{Adv}^{(\kappa,\beta)\text{-ind-cca}\circledast}_{\mathsf{PKE}[\lambda]}(\mathbb{A}_{\text{ind}})\,, \end{aligned}$$

where the final inequality takes into account the intermediate $\mathbb{B}_{\text{ind}}$ reduction to justify our assumption on $\mathbb{A}_{\text{ind}}$'s behaviour (from the beginning of the proof). □

Graphic design: Communication Division, UIB / Print: Skipnes Kommunikasjon AS

UNIVERSITAS BERGENSIS

uib.no

ISBN: 9788230858417 (print)
9788230851319 (PDF)