Anonymity-preserving Public-Key Encryption: A Constructive Approach

Markulf Kohlweiss[1], Ueli Maurer[2], Cristina Onete[3], Björn Tackmann[2], and Daniele Venturi[4]

[1] Microsoft Research, Cambridge, England
[2] ETH Zürich, Switzerland
[3] Darmstadt University of Technology, CASED, Germany
[4] Aarhus University, Denmark

Abstract. A receiver-anonymous channel allows a sender to send a message to a receiver without an adversary learning for whom the message is intended. Wireless broadcast channels naturally provide receiver anonymity, as does multi-casting one message to a receiver population containing the intended receiver. While anonymity and confidentiality appear to be orthogonal properties, making anonymous communication confidential is more involved than one might expect, since the ciphertext might reveal which public key has been used to encrypt. To address this problem, public-key cryptosystems with enhanced security properties have been proposed.

We investigate constructions as well as limitations for preserving receiver anonymity when using public-key encryption (PKE). We use the constructive cryptography approach by Maurer and Renner and interpret cryptographic schemes as constructions of a certain ideal resource (e.g. a confidential anonymous channel) from given real resources (e.g. a broadcast channel). We define appropriate anonymous communication resources and show that a very natural resource can be constructed by using a PKE scheme which fulfills three properties that appear in cryptographic literature (IND-CCA, key-privacy, weak robustness). We also show that a desirable stronger variant, preventing the adversary from selective "trial-deliveries" of messages, is unfortunately unachievable by any PKE scheme, no matter how strong. The constructive approach makes the guarantees achieved by applying a cryptographic scheme explicit in the constructed (ideal) resource; this specifies the exact requirements for the applicability of a cryptographic scheme in a given context. It also allows to decide which of the existing security properties of such a cryptographic scheme are adequate for the considered scenario, and which are too weak or too strong. Here, we show that weak robustness is necessary but that so-called strong robustness is unnecessarily strong in that it does not construct a (natural) stronger resource.

Keywords: public-key encryption, key privacy, robust encryption, anonymity, constructive cryptography

1 Introduction

Protocols and other mechanisms for protecting privacy often use cryptographic schemes in non-standard ways, sometimes requiring such schemes to have cryptographic properties that go beyond data authenticity and confidentiality. It is important that new cryptographic schemes take these requirements into account and that designers of privacy protocols are aware which cryptographic properties are needed in which situation. Several types of cryptographic schemes have been investigated with a focus on anonymity. In "key-private" public-key encryption, the ciphertext does not reveal information about the intended receiver [5,2], in private key exchange [8,1] two parties can exchange a key without revealing their identities, and "anonymous" signatures protect the signer's identity at least as long as parts of the signed plaintext remain hidden [27]. In this paper, we focus on public-key encryption and receiver anonymity.

The cryptographic community traditionally defines security notions for cryptographic schemes such as encryption from a game-based perspective, i.e., one defines properties of schemes by means of theoretical experiments, usually referred to as games. Though often used, well-studied, and improved over the years, game-based definitions have two major shortcomings. First, the models simplify the use of a scheme to the interaction between an *adversary* and a *challenger* (both somewhat artificial); thus, it is not clear what level of security is attained when a provably secure scheme is used in a specific context. Second, if a larger protocol employs such a provably secure (encryption) scheme, the security of the larger protocol is proved explicitly by reductions: one shows that breaking the security of the protocol leads to breaking the security of the (underlying) scheme. The reduction must in principle be tailor-made for each protocol.

A fundamentally different approach to defining the security of cryptographic schemes was proposed in [18]. In their *constructive cryptography* paradigm, one models both the resources assumed by a protocol and the desired functionality explicitly; the goal of the protocol is to *construct* (in a well-defined sense) the desired resource from the assumed resources. For a public-key encryption scheme, for instance, this means that one assumes an authenticated communication channel from the receiver to the sender to transmit the public key, and an insecure channel from the sender to the receiver to transmit the ciphertext. The goal is to construct, from the assumed channels, a confidential communication channel (from sender to receiver, cf. [20]). The assumed channels can be either physically realized or constructed cryptographically; the resulting channel can be used directly in any application requiring such a channel. Furthermore, as the constructive approach explicitly states the guarantees of both the assumed and the constructed resources, it allows to capture the *exact* cryptographic assumptions required for security.

Anonymity in constructive cryptography. In constructive cryptography, a network is a resource that can be accessed by multiple (honest or dishonest) parties. The parties use interfaces provided by the resource (and specific to each party) to access it; the interfaces specify exactly how each party can access the resource. In this context, anonymity is an explicit guarantee of the resource (e.g., a network). Since adversarial interaction with the network is also modeled by an interface (the attacker is a dishonest party), the security (or privacy) guarantees of the underlying network are described by the (absence of) capabilities of the adversary.

For the case of receiver-anonymous communication, we model a network resource with multiple receiver interfaces. Whenever a sender inputs a message at its interface and chooses a receiver, the network may leak some information (the length of the message, even the entire plaintext) at the adversary's interface; however, the resource will *not* reveal the receiver. In this context, a PKE scheme aims to construct a resource that still hides the receiver, while leaking no information on the message contents apart from (potentially) the length.

We consider the case where PKE schemes are used for end-to-end encryption between senders and receivers; in this case anonymity cannot be created through a cryptographic primitive. In fact, a constructive approach shows that schemes can only *preserve* the anonymity guaranteed by the underlying network, but never *produce* it.[1] If Alice sends a message to Bob over the Internet using Bob's publicly known IP address, then no encryption scheme (or key exchange protocol) can hide the fact that Bob is the intended receiver of Alice's message. In fact, encrypting messages can make the problem worse: Even if the transmission of the ciphertext is itself anonymous, the ciphertext might still reveal under which public key it was encrypted.

Hence, in a constructive analysis of the end-to-end use of cryptographic schemes, we always consider the *preservation* of anonymity. If the underlying network (one of the initial resources) is insecure, but guarantees some anonymity, then an "anonymity-preserving" scheme will improve security while retaining as much anonymity as possible. The obtained guarantees are strong in that they hold *regardless of the context*, i.e., of any prior knowledge the adversary might have and of any protocols executed in parallel.

Our contributions. We give a treatment of receiver anonymity in the context of public-key encryption (PKE) schemes from the perspective of constructive cryptography. Concretely, we describe anonymity as a feature of a communication resource, and we prove which security properties of the underlying encryption scheme are necessary and/or sufficient to achieve a confidential receiver-anonymous communication resource from a non-confidential, but also receiver-anonymous one. (Schemes with these properties exist in the literature.) Specifically, we consider the following network resources, where our notations extend the $\bullet$-notation of [20]. See Section 3.1 for details.

[1] This observation does not hold, however, for active or overlay networks that can implement their own multi-hop anonymous routing strategy (here, encryption is crucial). Buses [4] is an example of this, while TOR [10] is the most widely-used anonymization system based on this principle.

– The insecure broadcast network $\prec$, allowing a single sender to broadcast messages to multiple receivers. The adversary may learn the entire message and may remove, change, or inject messages;
– The confidential receiver-anonymous channel $-\!\Diamond\!\!\prec$, preserving both message confidentiality and receiver anonymity, leaking only the length of the message and allowing the adversary only to delete or honestly deliver messages, and to inject arbitrary messages to chosen recipients.

We show that $-\!\Diamond\!\!\prec$ can be constructed from $\prec$ and authenticated channels $\longleftarrow\!\!\bullet$ (in an initial step), by employing a secure (IND-CCA), key-private (IK-CCA), and weakly robust (WROB-CCA) PKE scheme. We prove that constructing $-\!\Diamond\!\!\prec$ does *not* require strong robustness (SROB-CCA, a stronger property for anonymous, secure encryption proposed in [2]). Of course, using SROB-CCA encryption also constructs $-\!\Diamond\!\!\prec$; however, this property is not *required*. Thus, the treatment in [2] relies on slightly too strong assumptions. Using SROB-CCA security, however, *does* yield a tighter security reduction.

We also show that one (the only natural) channel providing stronger anonymity than we achieve with IND-CCA, IK-CCA, and WROB-CCA encryption *cannot* be achieved by *any* PKE scheme at all (see Section 3.3). Thus, using e.g. the stronger SROB-CCA property does *not* construct this stronger channel. This does not mean that SROB-CCA is not useful in other scenarios; however, our results indicate that improving the properties of $-\!\Diamond\!\!\prec$ in a natural way cannot be done by using SROB-CCA, or any other type of encryption.

Related work. The first definition of key-private public-key encryption appears in [5]; the goal of the primitive was to attain receiver anonymity. Abdalla et al. [2] noted that also robustness is needed for the PKE scheme to achieve this property, since otherwise an honest receiver is unable to detect whether he is the intended recipient of a given ciphertext and could obtain a bogus decryption. We explicitly describe the guarantees achieved without robustness in the resource $-\!\Diamond_{\$}\!\!\prec$ in Section 4.1. Mohassel [22] analyzed game-based security and anonymity notions for KEM-DEM encryption schemes, showing that, for this particular type of composition, weak robustness together with the key privacy of the KEM (key-encapsulation mechanism) and DEM (data-encapsulation mechanism) components is sufficient to obtain a key-private hybrid public-key encryption scheme. Our result implies that weak robustness is sufficient even for universal composition; a *constructive* formulation of KEM-DEM schemes is currently being developed. However, as shown recently by Farshim et al. [11] (even strong) robustness is insufficient in certain contexts, such as Sako's auction protocol. The same concept (i.e., that only the intended recipient must be able to decrypt a ciphertext to a meaningful plaintext) lies at the core of *incomparable* public keys in [26].

More general (game-based) frameworks that mix the analysis of cryptographic schemes and traffic-analysis resistance have been proposed in [14] and [24]. Independently, different cryptographic [7,3] and traffic-analysis models [12,13] have been developed for variants of onion routing. Whereas our work here does not consider traffic analysis explicitly, our in-depth results can be composed with meaningful models of traffic analysis. We discuss implications of our results for traffic analysis in Section 5.

An early treatment of anonymity in networks (including receiver anonymity) was given in [25]. They explicitly considered the idea of using public-key encryption towards realizing receiver-anonymous networks. However, our treatment in this paper gives a more thorough, formal assessment of receiver anonymity and investigates necessary and sufficient resources that are necessary to achieve different levels of it. Nagao et al. [23] describe a similar resource for two sender-anonymous channels and show that such channels can be related by reductions to other types of channels, such as secure channels and direction hiding channels. Ishai et al. [15] provide a broader investigation on how to bootstrap cryptographic functionalities using anonymity. The resource we construct here provides receiver, rather than sender anonymity, and we also require confidentiality for our ideal resource (which is not the case for [15]).

2 Preliminaries

Notation. We use the symbol $\Diamond$ to denote an "error" output of an algorithm. Moreover, for an integer $n \in \mathbb{N}$, we let $[n] \coloneqq \{1, \dots, n\}$. We generally use typewriter fonts such as `enc` or `dec` to denote algorithms.

2.1 Systems: Resources and Converters, Distinguishers, and Games

We model objects like resources and protocols in terms of systems. At the highest level of abstraction—following the hierarchy in [18]—systems are objects with interfaces by which they connect to (interfaces of) other systems. Each interface is labeled with an element of a given label set and connects to only a single other interface. This concept, which we refer to as *abstract systems*, captures the topological structures that result when multiple systems are connected in this manner. In the following, we describe the basic types of systems that appear in this work at this level (of abstraction), and we introduce a notation for describing the structure in which multiple such systems are composed.

The abstract systems concept however does not model the behavior of systems, i.e., *how* the systems interact via their interfaces. As statements about cryptographic protocols are statements about behavior, they are formalized at the next (lower) abstraction level. In this respect, all systems in this work are (probabilistic) discrete systems, similar to [17].

Resources and converters. A *resource* for a multi-party setting is a system that provides one interface for each party. In our setting, resources have one interface labeled A for the sender, n interfaces labeled $B_1, \dots, B_n$ for the n receivers, and one interface labeled E associated with the attacker. Resources are usually denoted either by special symbols such as $\prec\!\!\!\lhd$ or by bold-face upper-case letters like $\mathbf{R}$ or $\mathbf{S}$. Protocols are formalized as tuples of so-called *converters*, one for each honest party; converters are systems that have two interfaces: one *inside* and one *outside* interface. Standard notations for converters are small Greek letters or special identifiers such as enc or dec; the set of all converters is denoted as Σ. A complete protocol (i.e., a tuple of converters) is denoted by a bold-face Greek letter, such as $\boldsymbol{\pi}$.

Converters can be attached to resources by connecting the inside interface of the converter to one interface of the resource. Notationally, if we attach the inside interface of the converter $\phi \in \Sigma$ to interface I of the resource $\mathbf{R}$, we write $\phi^I\mathbf{R}$. The resulting system $\phi^I\mathbf{R}$ is again a resource which provides all the interfaces of $\mathbf{R}$ (apart from I) as the respective interfaces, and the outside interface of the converter as the I-interface. If multiple parties use a protocol $\boldsymbol{\pi}$, then all converters that together form $\boldsymbol{\pi}$, one for each (honest) party, are attached to the resource in this manner. This is then denoted as $\boldsymbol{\pi}\mathbf{R}$.

Multiple resources $\mathbf{R}_1, \ldots, \mathbf{R}_m$ can be composed in parallel. This is denoted $[\mathbf{R}_1, \ldots, \mathbf{R}_m]$ and is again a resource, such that each interface $I \in \mathcal{I}$ of $[\mathbf{R}_1, \ldots, \mathbf{R}_m]$ allows to access the corresponding interfaces of $\mathbf{R}_1, \ldots, \mathbf{R}_m$.

Distinguishers. A *distinguisher* $\mathbf{D}$ is a special type of system that connects to all interfaces of a resource $\mathbf{U}$ and outputs a single bit at the end of its interaction with $\mathbf{U}$. We write this as the expression $\mathbf{DU}$, which defines a binary random variable. The *distinguishing advantage of a distinguisher* $\mathbf{D}$ *on two systems* $\mathbf{U}$ *and* $\mathbf{V}$ is defined as

$$\Delta^{\mathbf{D}}(\mathbf{U}, \mathbf{V}) \coloneqq |\mathsf{P}(\mathbf{DU} = 1) - \mathsf{P}(\mathbf{DV} = 1)|,$$

and we define $\Delta^{\mathcal{D}}(\mathbf{U}, \mathbf{V}) \coloneqq \sup_{\mathbf{D} \in \mathcal{D}} \Delta^{\mathbf{D}}(\mathbf{U}, \mathbf{V})$ as the advantage of a class $\mathcal{D}$ of distinguishers. The distinguishing advantage measures how much the output distribution of $\mathbf{D}$ differs when it is connected to either $\mathbf{U}$ or $\mathbf{V}$. Intuitively, if no distinguisher differentiates between $\mathbf{U}$ and $\mathbf{V}$, they can be used interchangeably in any environment (otherwise the environment can serve as a distinguisher).

The distinguishing advantage is a pseudo-metric. In particular, it satisfies the triangle inequality, i.e., $\Delta^{\mathbf{D}}(\mathbf{U}, \mathbf{W}) \leq \Delta^{\mathbf{D}}(\mathbf{U}, \mathbf{V}) + \Delta^{\mathbf{D}}(\mathbf{V}, \mathbf{W})$ for all resources $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{W}$, and for all distinguishers $\mathbf{D}$. Two systems are *equivalent*, denoted by $\mathbf{U} \equiv \mathbf{V}$, if they have the same behavior, which is the same as requiring that $\Delta^{\mathbf{D}}(\mathbf{U}, \mathbf{V}) = 0$ for *all* distinguishers $\mathbf{D}$.

The notion of construction. The formalization of constructive security definitions follows the ideal-world/real-world paradigm. The "real world" corresponds to an execution of the protocol $\boldsymbol{\pi}$ in which all honest parties have their converter attached to the assumed resource $\mathbf{R}$; more formally, we consider the *real-world system* $\boldsymbol{\pi}\mathbf{R}$. The "ideal world" corresponds to the constructed resource $\mathbf{S}$ with a simulator σ connected to the E-interface of $\mathbf{S}$, written $\sigma^E\mathbf{S}$ and referred to as *ideal-world system.* The purpose of σ is to adapt the E-interface of $\mathbf{S}$ such that it resembles the corresponding interface of $\boldsymbol{\pi}\mathbf{R}$.[2] If the two systems $\boldsymbol{\pi}\mathbf{R}$ and $\sigma^E\mathbf{S}$

[2] Indeed, the adversary can emulate the behavior of any efficient simulator σ; thus, using $\sigma^E\mathbf{S}$ instead of $\mathbf{S}$ can only restrict the adversary's power, so using $\sigma^E\mathbf{S}$ and hence $\boldsymbol{\pi}\mathbf{R}$ instead of $\mathbf{S}$ is safe.

are indistinguishable, then this roughly means that "whatever an attacker can do in the real world, he can also do in the ideal world".

Apart from the *security* condition described above, we also require an *availability* condition,[3] which excludes trivial protocols: If no attacker is present, the protocol must implement the specified functionality. In the definition, we use the special converter "$\bot$" that, when attached to a certain interface of a system, blocks this interface for the distinguisher.[4]

Definition 1 (Construction). *The protocol* π constructs $\mathbf{S}$ from the resource $\mathbf{R}$ within ε and with respect to the class $\mathcal{D}$ of distinguishers *if*

$$\exists\, \sigma \in \Sigma: \ \Delta^{\mathcal{D}}(\boldsymbol{\pi}\mathbf{R}, \sigma^E \mathbf{S}) \leq \varepsilon \qquad \text{and} \qquad \Delta^{\mathcal{D}}(\bot^E \boldsymbol{\pi}\mathbf{R}, \bot^E \mathbf{S}) \leq \varepsilon.$$

An important property of Definition 1 is its composability. Intuitively, if a resource $\mathbf{S}$ is used in the construction of a larger system, then the composability implies that $\mathbf{S}$ can be replaced by a construction $\boldsymbol{\pi}\mathbf{R}$ without requiring an explicit security reduction. For completeness, we include the composition theorem (which is adapted from [21]) in the full version of the paper [16].

Public-key encryption schemes. A public-key encryption (PKE) scheme with message space $\mathcal{M}$, ciphertext space $\mathcal{C}$, and public-key space $\mathcal{PK}$ is typically described as three algorithms $\mathsf{PKE} = (\mathtt{kgen}, \mathtt{enc}, \mathtt{dec})$. The key-generation algorithm $\mathtt{kgen}$ outputs a key pair (pk, sk), the (probabilistic) encryption algorithm $\mathtt{enc}$ takes a message $m \in \mathcal{M}$ and a public key pk and outputs a ciphertext $c = \mathtt{enc}(pk; m)$, and the decryption algorithm takes a ciphertext $c \in \mathcal{C}$ and a secret key sk and outputs a plaintext $m = \mathtt{dec}(sk; c)$. The decryption algorithm may also output the special symbol $\Diamond$ (for an invalid input c).

In constructive cryptography, using PKE in a setting with only one sender and one receiver can be described as deploying converters enc_1 (associated with the sender) and dec_1 (associated with the receiver) as follows. The receiver (within dec_1) initially runs the key-generation algorithm $\mathtt{kgen}$ to obtain a key pair (sk, pk), stores the private key sk locally, and sends the public key pk via an authenticated channel (denoted $\longleftarrow\!\!\bullet$, the first assumed resource). Upon receiving a ciphertext $\tilde{c}$ at the inside interface (via an a priori insecure communication channel $\longrightarrow$, the second assumed resource), dec_1 computes $\tilde{m} = \mathtt{dec}(sk; \tilde{c})$ and outputs $\tilde{m}$. The encryption converter enc_1 initially obtains the public key pk (via $\longleftarrow\!\!\bullet$) and, for each message m obtained at the outside interface, enc_1 computes $c = \mathtt{enc}(pk; m)$ and sends c over the insecure channel $\longrightarrow$. As pointed out already in [20], this constructs a confidential channel $\longrightarrow\!\!\!\!\bullet$.

In this paper, we consider PKE schemes deployed in a setting with one sender A and n receivers $B_1, \ldots, B_n$, corresponding to a tuple $(\mathsf{enc}, \mathsf{dec}, \ldots, \mathsf{dec})$ of $n+1$ converters. Each converter dec is defined similarly to dec_1 above, but if the decryption algorithm $\mathtt{dec}$ outputs $\Diamond$, then the converter dec outputs nothing. The encryption converter enc connects at its inside interface to $n+1$ resources.

[3] This corresponds to the completeness or correctness properties in some contexts.

[4] The $\bot$-converter also signals to the resource that no attacker is there.

By using the first n resources (here instantiated by $\longleftarrow\!\!\bullet^n$, i.e. for each receiver B_i there is one authenticated channel from B_i to A), enc expects to obtain public keys $pk_1, \ldots, pk_n$. Upon receiving $(m, i) \in \mathcal{M} \times [n]$ at the outside interface, enc computes $c = \mathtt{enc}(pk_i; m)$ and sends (c, i) via the $(n+1)$st resource (instantiated by an insecure broadcast network $\prec\!\!\triangleleft$) at the inside interface.

Games and security properties. Game-based definitions specify a property of a cryptographic scheme based on an interaction between two (hypothetical) entities: the game (or challenger) and the adversary. During the interaction, the adversary may issue "oracle queries" to the challenger, the responses of which model what information may be leaked to the adversary. The adversary's goal is specified by the game, and could be, e.g., forging a message or distinguishing encryptions of different messages. If this game cannot be won by any (efficient) adversary, then the scheme is secure against the considered type of attack.

We formalize the adversary and the game as systems that are connected by their interfaces. The game, often denoted as $\mathbf{G}$ with additional super- and subscripts, allows the adversary $\mathbf{A}$ to issue "oracle queries" via that interface. Whether or not the game is won is signaled by a special (monotone) output bit of $\mathbf{G}$ (this can be considered as an additional interface) that is initially 0 but switches to 1 as soon as the winning condition is fulfilled. This bit is denoted Output. For a game $\mathbf{G}$ and an adversary $\mathbf{A}$, we define the *game-winning probability* after q steps (queries) as

$$\Gamma_q^{\mathbf{A}}(\mathbf{G}) \coloneqq \mathsf{P}^{\mathbf{AG}}(\mathsf{Output}_q = 1).$$

For an adversary $\mathbf{A}$ that halts after (at most) q steps, we write $\Gamma^{\mathbf{A}}(\mathbf{G}) \coloneqq \Gamma_q^{\mathbf{A}}(\mathbf{G})$.

Many games considered in the context of encryption schemes, including most games considered here, are *bit-guessing games.* These games can often be described by a pair of systems $\mathbf{G}_0$ and $\mathbf{G}_1$; in the beginning of the game, a bit $B \in \{0, 1\}$ is chosen uniformly at random and the adversary is given access to $\mathbf{G}_B$. The goal is to find the bit B; thus, the adversary has a probability $\frac{1}{2}$ to simply guess this value. Hence, we measure the adversary's success in terms of his *advantage*, that is, the (absolute) difference between $\mathbf{A}$'s probability of winning $\mathbf{G}$ and the success probability for these "trivial" strategies, formally $\varPhi^{\mathbf{A}}(\mathbf{G}) = 2 \cdot \left|\Gamma^{\mathbf{A}}(\mathbf{G}) - \frac{1}{2}\right|$. Note also that $\varPhi^{\mathbf{A}}(\mathbf{G}) = \Delta^{\mathbf{A}}(\mathbf{G}_0, \mathbf{G}_1)$.

For a security property that is defined by means of $\mathbf{G}$, we say that the scheme is secure within ε and with respect to a class $\mathcal{A}$ of adversaries if the advantage $\mathbf{A} \in \mathcal{A}$ has in winning $\mathbf{G}$ is bounded by ε.

Asymptotics. To allow for asymptotic security definitions, cryptographic protocols are often equipped with a so-called *security parameter.* We formulate all statements in this paper in a non-asymptotic fashion, but asymptotic statements can be obtained by treating systems $\mathbf{S}$ as asymptotic families $\{\mathbf{S}_k\}_{k \in \mathbb{N}}$ and letting the distinguishing advantage be a real-valued function of k. Then, for a given notion of efficiency, one can consider security with respect to classes of efficient distinguishers and a suitable negligibility notion. All reductions in this work are efficient with respect to the standard polynomial-time notions.

2.2 Games for Key Privacy and Robustness

We describe the queries that an adversary can ask in a game formally as *procedures* that he can *call*; the specific game structure is enforced by the order in which they are called. This is not a technically new approach (see for instance [6]); however, it integrates smoothly with the security statements we aim for in this work. The most important properties for our work are IND-CCA-security, key privacy, and robustness.

Key privacy. In a key-private PKE scheme, the adversary, given two public keys pk_0 and pk_1, must be unable to tell which key was used to generate a given ciphertext [5]. This definition is similar in spirit to the standard "left-or-right" IND-CCA definition, where the adversary is given the public key, but does not know which of two messages is encrypted under it. In the key-privacy game the message is known, but not the public key. The standard notion of key privacy, i.e. key privacy for chosen ciphertext attacks (IK-CCA) is recalled in the full version [16] together with the two variants we use in our reductions.

Robustness. The notion of *robustness* in encryption was formalized by Abdalla et al. [2] in two flavors: *weak* and *strong* robustness. They consider both versions under both chosen plaintext and chosen ciphertext attacks. We focus here on weak, resp. strong robustness under chosen ciphertext attacks (WROB-CCA, resp. SROB-CCA), associated with the experiments in Figures 1, resp. 2, where the adversary may call the following oracles.

- On input an identifier ID, the oracle $\mathbf{GenUser}(\cdot)$ generates a public and a private key for the user ID and returns the public key. A set U keeps track of the honestly generated key pairs and identifiers.
- On input a valid identifier $\mathsf{ID} \in U$, the oracle $\mathbf{Corrupt}(\cdot)$ returns the private key corresponding to user ID and adds the identifier to a set V.
- On input a valid identifier $\mathsf{ID} \in U$ and a ciphertext c, the decryption oracle $\mathbf{Decrypt}(\cdot, \cdot)$ outputs the corresponding plaintext m.

$\mathbf{Init}()$	$\mathbf{GenUser}(\mathsf{ID})$	$\mathbf{Corrupt}(\mathsf{ID})$	$\mathbf{Decrypt}(\mathsf{ID}, c)$	$\mathbf{GameOutput}(m, \mathsf{ID}_0, \mathsf{ID}_1)$
$U \leftarrow \emptyset$ $V \leftarrow \emptyset$ $\mathsf{Output} \leftarrow 0$ **end.**	$(sk_{\mathsf{ID}}, pk_{\mathsf{ID}}) \leftarrow \mathtt{kgen}()$ $U \leftarrow U \cup \{(\mathsf{ID}; sk_{\mathsf{ID}}; pk_{\mathsf{ID}})\}$ return pk_{ID} **end.**	if $(\mathsf{ID}; \cdot; \cdot) \notin U$ return $\Diamond$ end if. $V \leftarrow V \cup \{\mathsf{ID}\}$ return sk_{ID} from U **end.**	if $(\mathsf{ID}; \cdot; \cdot) \notin U$ return $\Diamond$ end if. return $\mathtt{dec}(sk_{\mathsf{ID}}, c)$ **end.**	if $(\mathsf{ID}_0 = \mathsf{ID}_1) \vee \{\mathsf{ID}_0, \mathsf{ID}_1\} \cap V \neq \emptyset$ return $\Diamond$ end if. $c \leftarrow \mathtt{enc}(pk_{\mathsf{ID}_0}, m)$ $m_1 \leftarrow \mathtt{dec}(sk_{\mathsf{ID}_1}, c)$ $\mathsf{Output} \leftarrow (m \neq \Diamond) \wedge (m_1 \neq \Diamond)$ **end.**

Fig. 1: The weak robustness game, $\mathbf{G}^{\mathsf{w\text{-}rob}}$.

In the WROB-CCA game, the adversary chooses a plaintext and two identities. The plaintext is encrypted by the challenger (without tampering) for the first identity. The adversary wins if this ciphertext decrypts to a valid plaintext for the second identity. By contrast, for strong robustness (SROB-CCA), the adversary can manipulate ciphertexts and wins if a chosen ciphertext decrypts to valid plaintexts for two different public keys.

Init()	**GenUser**(ID)	**Corrupt**(ID)	**Decrypt**(ID, c)	**GameOutput**($c, \mathsf{ID}_0, \mathsf{ID}_1$)
$U \leftarrow \emptyset$	$(sk_{\mathsf{ID}}, pk_{\mathsf{ID}}) \leftarrow \mathtt{kgen}()$	if $(\mathsf{ID}; \cdot; \cdot) \notin U$	if $(\mathsf{ID}; \cdot; \cdot) \notin U$	if $(\mathsf{ID}_0 = \mathsf{ID}_1) \vee \{\mathsf{ID}_0, \mathsf{ID}_1\} \cap V \neq \emptyset$
$V \leftarrow \emptyset$	$U \leftarrow U \cup \{(\mathsf{ID}; sk_{\mathsf{ID}}; pk_{\mathsf{ID}})\}$	return $\Diamond$	return $\Diamond$	return $\Diamond$
$\mathsf{Output} \leftarrow 0$	return pk_{ID}	end if.	end if.	end if.
end.	**end.**	$V \leftarrow V \cup \{\mathsf{ID}\}$	return $\mathtt{dec}(sk_{\mathsf{ID}}, c)$	$\mathsf{Output} \leftarrow (\mathtt{dec}(sk_{\mathsf{ID}_0}, c) \neq \Diamond) \wedge$
		return sk_{ID} from U	**end.**	$(\mathtt{dec}(sk_{\mathsf{ID}_1}, c) \neq \Diamond)$
		end.		**end.**

Fig. 2: The strong robustness game, $\mathbf{G}^{\mathsf{s\text{-}rob}}$.

3 Receiver-Anonymous Communication

The main goal of this work is to model and achieve confidential and receiver-anonymous communication. We first formalize a useful anonymity guarantee by describing in Section 3.1 the resource , which *can* actually be constructed from a "broadcast" channel and several authenticated channels (to transmit the public keys). We then discuss in Section 3.2 in which (inefficient) way this construction can be achieved by vanilla public-key encryption, and, in Section 3.3, we argue that "much more" anonymity is impossible to achieve. Finally, in Section 3.4 we show how to achieve this construction more efficiently, by using a PKE scheme that is IND-CCA, IK-CCA [5], and WROB-CCA [2].

3.1 Resources for Receiver-Anonymous Communication

An n-receiver channel is a resource with an interface labeled A for the sender, interfaces labeled $B_1, \ldots, B_n$ for the receivers, and a third type of interface labeled E that captures potential adversarial access. The security properties of different n-receiver channels are described in the following; the symbolic notation for the channels extends that from [20].

The security statements in this work are parametrized by the number of messages that are transmitted over the channels. More precisely, for each of the following channels and each $q \in \mathbb{N}$, we define the *q-bounded channel* as the one that processes (only) the first q queries at the A-interface and the first q queries at the E-interface as described, and ignores all further queries at these interfaces. We then require from a protocol that it constructs, for all $q \in \mathbb{N}$, the q-bounded "ideal" channel from the q-bounded assumed channel.[5] Wherever the number q is significant, such as in the theorem statements, we denote the q-bounded versions of channels by writing the q on top of the channel symbol (e.g., q); we omit it in places that are of less formal nature.

[5] This condition is equivalent to considering an "unbounded" channel; the important feature is that *the protocol* is independent of the number q of messages.

Insecure broadcast communication. We base our constructions on a resource $-\!\!<$, which allows the sender to broadcast a given message to all receivers $B_1, \ldots, B_n$. Such a channel can be implemented, for example, by multi-sending the same message individually to each receiver over an insecure network; the channel models also what is achieved by wireless broadcast. The resource $-\!\!<$ leaks the complete message at the E-interface, and allows to delete, change, or inject messages destined for particular receivers via the E-interface. In more detail:

– If at the E-interface the $\perp$-converter is connected,[6] then on input the k-th message m_k at the A-interface, output m_k at B_j for all $j \in [n]$.
– Otherwise, on input the k-th message m_k at the A-interface, output m_k at the E-interface. Upon the query $(\texttt{inject}, j, \tilde{m})$ at the E-interface for $j \in [n]$ and $\tilde{m} \in \mathcal{M}$, deliver $\tilde{m}$ at interface B_j.

Confidential receiver-anonymous communication. The confidential receiver-anonymous channel $-\!\diamond\!\!<^{\bullet}_{\bullet}$ leaks neither the message contents nor the intended recipient to the adversary, just the message length. It allows, however, to "conditionally" deliver a message to a chosen user if and only if this chosen user was the originally intended recipient.

– If at the E-interface the $\perp$-converter is connected, then on the k-th input (m_k, i_k) at the A-interface, output m_k at B_{i_k}.
– Otherwise, on the k-th input (m_k, i_k) at the A-interface, output the message length $|m_k|$ at the E-interface. Furthermore, the E-interface allows the following queries:
  - $(\texttt{inject}, j, \tilde{m})$ for $j \in [n]$ and $\tilde{m} \in \mathcal{M}$: delivers $\tilde{m}$ at interface B_j;
  - $(\texttt{deliver}, j, \bar{k})$ for $j \in [n]$, $\bar{k} \in \mathbb{N}$: If at least $\bar{k}$ messages have been sent via A and $i_{\bar{k}} = j$, then it delivers the message $m_{\bar{k}}$ at B_j.

This is also depicted in Figure 3. In the application of a public-key cryptosystem to a broadcast network such as $-\!\diamond\!\!<^{\bullet}_{\bullet}$, the capabilities at the E-interface correspond to trial deliveries of intercepted messages and to adversarial encryptions.

Authenticated channel. Each receiver uses one authenticated channel $\longleftarrow\!\!\bullet$ to send its public key to the sender; we use n parallel authenticated channels, denoted $\longleftarrow\!\!\bullet^n$ (one for each receiver), as assumed resources in our constructions. Formally, a single authenticated channel $\longleftarrow\!\!\bullet$ with message space $\mathcal{M}$ is a three-party resource with interfaces A, B_i (for some i), and E. On input a message $m \in \mathcal{M}$ at interface B_i, the channel outputs m at the E-interface. The channel outputs m at the A-interface only upon receiving an acknowledgement from the E-interface (the adversary controls message delivery).

[6] Formally, there is a special input that provokes this behavior, and the converter $\perp$ provides this input.

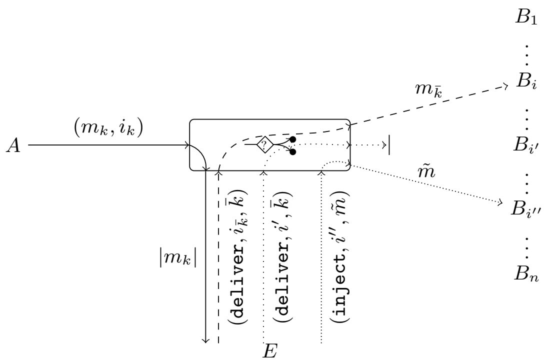


Fig. 3: The confidential receiver-anonymous channel.

3.2 Generic Construction using Public-Key Encryption

The channel $-\!\!\diamond\!\!\prec$ can be constructed from $\prec$ and $\longleftarrow\!\!\bullet^n$ using any secure public-key scheme: Each receiver generates a key pair and sends the public key through its authenticated channel $\longleftarrow\!\!\bullet$ to the sender; the sender transmits a message to a specific receiver by concatenating (in a fixed predetermined order): an encryption of this message under the intended receiver's public key and a "garbage" message encrypted with the appropriate key for each additional potential receiver; this composite message is then sent via the broadcast channel. Each receiver decrypts only "its" part of the composite ciphertext and checks whether or not the message was "garbage." (Typically, the "garbage" message can be set to a constant message $\bar{m} \in \mathcal{M}$ not otherwise used.) If the broadcast channel is achieved by multi-sending the same message to each receiver, then one can also send only the corresponding part to each receiver.

Yet, this approach has two main disadvantages. First, the computation and communication complexity is linear in the (potentially large) number of possible receivers. Second, the sender must *know* the public keys of all potential receivers, not just of the one intended receiver.

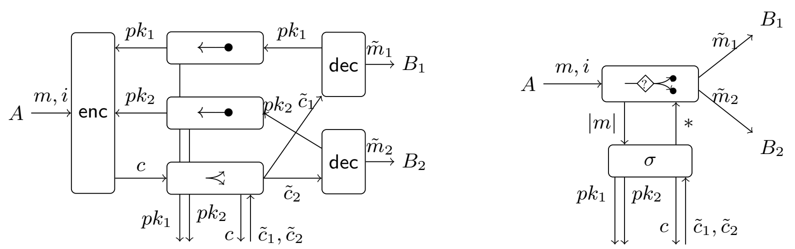
3.3 "Upper Bounds" on Anonymity

Anonymity beyond the guarantees of $-\!\!\diamond\!\!\prec$ seems unlikely to be achieved from the resources $\prec$ and $\longleftarrow\!\!\bullet^n$ which we assumed. Indeed, we show that a (minor and natural) extension of $-\!\!\diamond\!\!\prec$ cannot be achieved from our assumed resources. The extension, denoted by ANON, removes the "conditional delivery" capability provided at the E-interface in resource $-\!\!\diamond\!\!\prec$, and enables deliveries of the type $(\texttt{deliver}, \bar{k})$ for $\bar{k} \in \mathbb{N}$, where, if at least $\bar{k}$ messages have been sent via A, then the message $m_{\bar{k}}$ is delivered to $B_{i_{\bar{k}}}$. In particular, the distinguisher can use the E-interface of system $\prec$ to deliver the messages to, e.g., only one chosen receiver, which will output the message if and only if it is the intended recipient. We call this process a "trial delivery" and show that it allows the distinguisher to tell the real-world system apart from the ideal-world system with ANON, where trial deliveries are impossible by definition.

This result is formalized in in the full version [16]; the proof expands on the sketch we gave above. Note that the channel ANON is just one type of ideal resource providing stronger anonymity guarantees than $-\!\!\Diamond\!\!\prec$; however, our impossibility result extends easily to any resource without conditional deliveries.

3.4 Achieving Confidential Receiver-Anonymous Communication

A public-key encryption scheme constructs the resource $-\!\!\Diamond\!\!\prec$ from a broadcast channel if it has the properties IND-CCA, IK-CCA, and WROB-CCA. The property WROB-CCA (weak robustness) captures the guarantee that ciphertexts honestly generated for one user will not be successfully decrypted by another user. We show that weak robustness is sufficient for our construction. This may appear somewhat surprising since the adversary *can* inject arbitrary ciphertexts into the channel $\lessdot$, see [2]. The intuitive reason why WROB-CCA is sufficient is two-fold: First, preventing the adversary from generating a single "fresh" ciphertext that is accepted by two receivers is only helpful if injecting two different ciphertexts is impossible, or harder for the adversary than injecting a single one (cf. Section 4.3). Second, the non-malleability guarantees of IND-CCA exclude that the adversary can "maul" honestly generated ciphertexts such that unintended receivers decrypt "related" plaintexts (this is used in the reduction to IND-CCA in the proof of Theorem 1).



(a) Using public-key encryption over an (insecure) broadcast.

(b) The idealized setting, the value "$*$" depends on $\tilde{c}_1, \tilde{c}_2$.

Fig. 4: The security statement in a setting with two receivers.

The security statement we prove below is depicted in Figure 4, where we show how the scheme is used together with the assumed resources: Each sender transmits its public key authentically to the sender, who then uses the broadcast channel to transmit the ciphertext to both receivers. Figure 4b shows the idealized setting, where the message is transmitted via the resource $-\!\!\Diamond\!\!\prec$ (which guarantees confidentiality). The value "$*$" is determined by the simulator and depends on the values $\tilde{c}_1$ and $\tilde{c}_2$ given by the adversary; the symbol may stand for a query to deliver the message m or to inject unrelated messages.

Theorem 1 shows that if the public-key encryption scheme has the three assumed properties, then the two settings in Figure 4 are indistinguishable. Intuitively, whenever such a scheme is used to protect messages transmitted via a broadcast channel like $\lessdot$, one obtains the guarantees explicitly described by the "idealized" network resource $-\!\!\Diamond\!\!\prec\!\!:$. The proof of the theorem shows that every distinguisher for the two settings can be transformed into an adversary against (at least) one of the three properties IND-CCA, IK-CCA, and WROB-CCA with loss qn for q messages and n receivers. The games $\mathbf{G}^{\textsf{ind-cca}}$ and $\mathbf{G}^{\textsf{ik-cca}}$ referred to in the theorem are defined in the full version [16].

Theorem 1. *Let* $(\texttt{kgen}, \texttt{enc}, \texttt{dec})$ *be a public-key encryption scheme that has the three properties* IND-CCA-, IK-CCA-, *and* WROB-CCA. *Then, the protocol* $(\textsf{enc}, \textsf{dec}, \ldots, \textsf{dec})$ *defined as in Section 2.1 transforms* $\lessdot$ *and* $(\longleftarrow\!\bullet)^n$ *into* $-\!\!\Diamond\!\!\prec\!\!:$. *More formally, there are a simulator* σ *and for each* $q \in \mathbb{N}$ *four reductions* $\mathbf{A}_q(\cdot)$, $\mathbf{A}'_q(\cdot)$, $\mathbf{A}''_q(\cdot)$, $\mathbf{A}'''_q(\cdot)$ *such that*

$$\Delta^{\mathbf{D}}\left(\textsf{enc}^A\textsf{dec}^{B_1}\ldots\textsf{dec}^{B_n}\bot^E\left[\overset{q}{\lessdot}, \longleftarrow\!\bullet^n\right], \bot^E\overset{q}{-\!\!\Diamond\!\!\prec\!\!:}\right) \le qn\cdot\Gamma^{\mathbf{A}_q(\mathbf{D})}\left(\mathbf{G}^{\textsf{w-rob}}\right), \quad (1)$$

and

$$\Delta^{\mathbf{D}}\left(\textsf{enc}^A\textsf{dec}^{B_1}\ldots\textsf{dec}^{B_n}\left[\overset{q}{\lessdot}, \longleftarrow\!\bullet^n\right], \sigma^E\overset{q}{-\!\!\Diamond\!\!\prec\!\!:}\right)$$
$$\le qn\cdot\Phi^{\mathbf{A}'_q(\mathbf{D})}\left(\mathbf{G}^{\textsf{ind-cca}}\right) + 2qn\cdot\Phi^{\mathbf{A}''_q(\mathbf{D})}\left(\mathbf{G}^{\textsf{ik-cca}}\right) + qn\cdot\Gamma^{\mathbf{A}'''_q(\mathbf{D})}\left(\mathbf{G}^{\textsf{w-rob}}\right). \quad (2)$$

Proof (sketch). We sketch the proofs for conditions (1) and (2) independently.

Availability. We describe a reduction $\mathbf{A}_q(\cdot)$ that turns a distinguisher $\mathbf{D}$ between the real-world system $\textsf{enc}^A\textsf{dec}^{B_1}\ldots\textsf{dec}^{B_n}\bot^E[\lessdot, \longleftarrow\!\bullet^n]$ (which we denote $\mathbf{R}_\bot$) and the ideal-world system $\bot^E(-\!\!\Diamond\!\!\prec\!\!:)$ (denoted $\mathbf{S}_\bot$) into an adversary for the the WROB-CCA game. The idea of the proof is to construct a monotone event sequence (MES, see [17]), which becomes true once the distinguisher inputs a pair (m, i) at the A-interface such that a receiver B_j for some index $j \ne i$ outputs some plaintext $m_j \ne \Diamond$. If the encryption scheme has perfect correctness, the systems $\mathbf{R}_\bot$ and $\mathbf{S}_\bot$ are equivalent, conditioned on the MES remaining false (if the scheme is *not* perfectly correct, we alter the MES to take this into account). Yet, note that even isolating a query (m, i) that invokes the MES does not immediately imply that a new encryption of the same m and pk_i will yield another ciphertext (in the query of the WROB-CCA game) that decrypts to $m'_j \ne \Diamond$ by sk_j for the index $j \ne i$. Instead, for the reduction to be successful, the reduction $\mathbf{A}_q$ guesses the query and the receiver where this erroneous decryption will occur. Thus, the reduction loses a factor qn, as claimed.

Security. We first describe the simulator σ attached to the E-interface of the ideal resource. The role of σ is to simulate the interaction at the E-interface to a distinguisher. We then prove that σ is indeed a good simulator: in other words, we provide reductions that transform a given successful distinguisher into a successful adversary against one of the following games: IND-CCA, IK-CCA, or WROB-CCA. The simulator σ runs as follows:

- Generate n private-/public-key pairs (pk_i, sk_i) with $i \in [n]$ to simulate each pk_i that it is transmitted via the corresponding channel $\longleftarrow\!\bullet$. Furthermore, generate one auxiliary key pair $(\tilde{pk}, \tilde{sk})$.
- Upon the k-th message length ℓ_k from $-\!\!\diamond\!\!\!\prec^{\bullet}_{\bullet}$, generate a new ciphertext $c_k = \mathtt{enc}(\tilde{pk}; 0^{\ell_k})$ and simulate c_k as a message on $\prec$.
- When $\mathbf{D}$ delivers a message $\tilde{c}$ to some user $j \in [n]$:
  - In case $\tilde{c} = c_{\bar{k}}$ for some $\bar{k} \in \mathbb{N}$, issue $(\mathtt{deliver}, j, \bar{k})$ to $-\!\!\diamond\!\!\!\prec^{\bullet}_{\bullet}$.
  - In case $\tilde{c}$ is "fresh," compute $\tilde{m}_j \coloneqq \mathtt{dec}(sk_j; \tilde{c})$, and, if $\tilde{m}_j \neq \lozenge$, issue $(\mathtt{inject}, j, \tilde{m}_j)$ to $-\!\!\diamond\!\!\!\prec^{\bullet}_{\bullet}$.

Assume that there exists a distinguisher $\mathbf{D}$ that successfully distinguishes the real-world system $\mathsf{enc}^A\mathsf{dec}^{B_1}\ldots\mathsf{dec}^{B_n}\,[\prec, \longleftarrow\!\bullet^n]$ from the ideal-world system $\sigma^E -\!\!\diamond\!\!\!\prec^{\bullet}_{\bullet}$. We sketch the security reductions to the underlying games.

WROB-CCA. As a first intermediate step, we introduce a hybrid resource $\mathbf{H}_1$. This resource behaves like $-\!\!\diamond\!\!\!\prec^{\bullet}_{\bullet}$, except that it allows for the delivery of an arbitrary message to a party other than the intended recipient: namely, instead of the query $(\mathtt{deliver}, j, \bar{k})$, we allow to deliver a message $\tilde{m}$ to a user B_j for $j \neq i_{\bar{k}}$ (still $m_{\bar{k}}$ for $j = i_{\bar{k}}$) by means of $(\mathtt{deliver}, j, \bar{k}, \tilde{m})$. We use a modified simulator σ_1 that sends the decryption of the ciphertext simulated for message $\bar{k}$ under the key of user j. The systems $\sigma_1^E\mathbf{H}_1$ and $\sigma^E -\!\!\diamond\!\!\!\prec^{\bullet}_{\bullet}$ are equivalent unless, for some query, there is a user B_j, not the intended recipient of some ciphertext, that outputs a message upon receiving the ciphertext. A distinguisher that provokes this situation (i.e., it causes some unintended recipient to output a message from a ciphertext) can be used to win the WROB-CCA game. The reduction $\mathbf{A}_q'''(\cdot)$ obtains n generated keys from the WROB-CCA game, which correspond to the users, and an additional key, used to simulate ciphertexts. As in the availability proof, $\mathbf{A}_q'''$ has to guess on which query $(\tilde{m}_l, i_l)$ and with respect to which other index j the erroneous decryption will occur, for sending the appropriate $\tilde{m}_l$, i_l, and j as its challenge in the weak robustness game. In order to properly simulate the eavesdropper to the environment, we use a slightly tweaked version of weak robustness (equivalent to the original one) where we also obtain the generated ciphertext when running the **GameOutput** oracle.

IND-CCA. We introduce a second hybrid $\mathbf{H}_2$ that behaves as $\mathbf{H}_1$ but additionally leaks the receiver's identity (no anonymity). The suitable simulator σ_2 always encrypts the all-zero string of appropriate length for the respective user, and decrypts as needed. Two things must be shown: first, that $\sigma_2^E\mathbf{H}_2$ is indistinguishable from the real-world system; and second, that $\sigma_1^E\mathbf{H}_1$ and $\sigma_2^E\mathbf{H}_2$ are indistinguishable. We start with the former one, where the reduction $\mathbf{A}_q'(\cdot)$ uses a hybrid argument to employ a distinguisher for $\sigma_2^E\mathbf{H}_2$ and $\mathsf{enc}^A\mathsf{dec}^{B_1}\ldots\mathsf{dec}^{B_n}\,[\prec, \longleftarrow\!\bullet^n]$ to win the IND-CCA game. Technically, one defines a sequence of hybrid systems, where the i-th hybrid simulates "ideal" encryptions for the first $i-1$

receivers, uses the game to simulate for the i-th receiver, and "real" encryptions for the remaining receivers. The reduction $\mathbf{A}'_q(\cdot)$ then chooses $i \in [n]$ uniformly at random. Overall, the first hybrid with no simulated encryptions is equivalent to $\mathsf{enc}^A\mathsf{dec}^{B_1}\dots\mathsf{dec}^{B_n}\,[\prec, \longleftarrow\!\!\bullet^n]$, while the hybrid with only simulated encryptions is equivalent to the hybrid $\sigma_2\mathbf{H}_2$. As the IND-CCA game offers only a single challenge query, another hybrid argument must be employed to account for the number of encryptions; this adds a factor of q.

IK-CCA. The last step is to show a reduction $\mathbf{A}''_q(\cdot)$ that turns a distinguisher between $\sigma_1\mathbf{H}_1$ and $\sigma_2\mathbf{H}_2$ corresponding, respectively, to the first and second hybrid introduced in the proof, into an IK-CCA-adversary. Recall that $\mathbf{H}_2$ behaves just like $\mathbf{H}_1$ except that it does not grant anonymity. We again use a hybrid argument with qn "intermediate" systems between $\sigma_1^E\mathbf{H}_1$ and $\sigma_2^E\mathbf{H}_2$, similarly to the IND-CCA case, such that each intermediate system embeds the challenge at a different position (as above). All other keys, encryptions, or decryptions are either simulated as "real" or as "ideal," depending on their position. The system where only the queries are "real" is equivalent to $\sigma_2\mathbf{H}_2$, and the system where only $\tilde{pk}$ was used (all queries are "ideal") is equivalent to $\sigma_1\mathbf{H}_1$. $\square$

4 Relation to Notions of Robustness

While the confidential receiver-anonymous channel can be achieved using an encryption scheme that fulfills IND-CCA, IK-CCA, and WROB-CCA, anonymity without robustness is not sufficient. This was already noted by Abdalla et al. [2], who point out that if one receiver obtains a ciphertext that was intended for a different receiver, the decryption should yield this information—by producing an error symbol—instead of an arbitrary, but well-formed plaintext, because this undetected, but unintended plaintext message might "upset" higher level protocols. This "robustness," however, is not guaranteed by IND-CCA or IK-CCA.

This section formalizes and proves statements related to robustness. In Section 4.1 we describe the type of channel one obtains if the PKE scheme is only IND-CCA- and IK-CCA-secure; this confirms the intuition given in [2]. We then show in Section 4.2 that WROB-CCA is indeed formally *necessary* to construct the channel $\multimap\!\!\diamond\!\!\prec$: Every (IND-RCCA and IK-RCCA-secure) scheme that achieves the constructive notion *must* be weakly robust. Finally, in Section 4.3 we show that a strongly robust scheme will also *only* construct the resource $\multimap\!\!\diamond\!\!\prec$, though with a tighter reduction. We also explain why a strongly robust scheme does not help to construct a "qualitatively better" resource.

4.1 Anonymity with Erroneous Transmission

The channel one obtains from applying an IND-CCA and IK-CCA-secure scheme to $\prec$ and $\longleftarrow\!\!\bullet^n$ is the resource $\multimap\!\!\diamond_{\$}\!\!\prec$ which is parametrized by a family of distributions $\left(\mathsf{P}^{\ell}_{Y_1\dots Y_n}\right)_{\ell\in\mathbb{N}}$ and differs from $\multimap\!\!\diamond\!\!\prec$ only in the cases where honestly generated messages are transmitted to receivers other than the intended one (either during an honest transmission or because the adversary forwards an honestly sent message to such a receiver). Without weak robustness, the unintended receiver will output a message according to the (scheme-specific) distribution. A formal description of $-\!\!\diamondsuit\!\!\!\$\!\!\prec\!\!\bullet$ follows.

- If at the E-interface the $\perp$-converter is connected, then for the k-th input (m_k, i_k) at the A-interface, choose $m'_{k,1}, \ldots, m'_{k,n}$ according to $\mathsf{P}^{|m_k|}_{Y_1 \ldots Y_n}$, output m_k at B_{i_k} and $m'_{k,j}$ at B_j for $j \neq i_k$ (if $m'_{k,j} \neq \Diamond$, else nothing).
- Otherwise, on the k-th input (m_k, i_k) at the A-interface, output only the message length $|m_k|$ at the E-interface. Furthermore, the E-interface allows the following queries:
  - $(\texttt{inject}, j, \tilde{m})$ for $j \in \{1, \ldots, n\}$ and $\tilde{m} \in \mathcal{M}$: Delivers $\tilde{m}$ at B_j,
  - $(\texttt{deliver}, j, \bar{k}, \tilde{m})$ for $j \in \{1, \ldots, n\}$, $\bar{k} \in \mathbb{N}$, and $m \in \mathcal{M}$: If at least $\bar{k}$ messages have been sent via A, then delivers message $m_{\bar{k}}$ at B_j if $i_{\bar{k}} = j$, and delivers $\tilde{m}$ at B_j otherwise.

In the full version [16] we show that the channel $-\!\!\diamondsuit\!\!\!\$\!\!\prec\!\!\bullet$ is constructed from $\prec$ and n authenticated channels $\longleftarrow\!\bullet$ if the encryption scheme is IND-CCA- and IK-CCA-secure. In the proof, we instantiate the channel $-\!\!\diamondsuit\!\!\!\$\!\!\prec\!\!\bullet$ with a distribution $\mathsf{P}^{\ell}_{Y_1 \ldots Y_n}$ that we define by honestly choosing keys for the receivers and, whenever a message is sent to a party B_i, decrypting a "random ciphertext" of the correct length with respect to the keys of all parties B_j with $j \neq i$.

4.2 "Equivalence" with Weak Robustness

In Section 3.4 we showed that IND-CCA, IK-CCA, and WROB-CCA security are *sufficient* to construct $-\!\!\diamondsuit\!\!\prec\!\!\bullet$. Indeed, (slightly weaker variants of) these properties are also *necessary*: If a PKE scheme is sufficient for the construction, then it must also be weakly robust, IND-RCCA, and IK-RCCA. Note that "CCA"-notions are sufficient, but not necessary, as they also prohibit that a scheme allows for "trivial" modifications of the ciphertext, which do not have an impact on the actual security [9,19]. We explicitly describe the IND-RCCA-game and the IK-RCCA-game in the full version. These notions, compared to the original ones, have more "elaborate" decryption oracles that prevent decryptions of "trivially modified" ciphertexts.

The formal statement and the proof are deferred to the full version [16]. The basic idea is that for each of the three games for weak robustness, IND-RCCA, and IK-RCCA, we show that a successful adversary will also serve as a good distinguisher in the constructive security statement.

4.3 Anonymity with Strong Robustness

Strong robustness (SROB-CCA, [2]) is strictly stronger than weak robustness. Intuitively, whereas weak robustness states that honestly generated ciphertexts are not decryptable by two distinct receivers, strong robustness requires this even for adversarially generated ciphertexts. A strongly robust scheme will of course also be sufficient to achieve $-\!\!\Diamond\!\!\!\lessdot\!:$, in the full version we show that we even achieve better bounds in the reduction. Intuitively, due to the exact definition of the oracles in the games, the reduction to SROB-CCA can exploit *every* inconsistency in an emulated interaction with the distinguisher, whereas the reduction to WROB-CCA has to guess *when* the inconsistency will occur.

Somewhat surprisingly, strong robustness does not provide a "qualitatively" better security guarantee than weak robustness. ("Qualitative" refers to the properties of the resources, in contrast to the "quantitative" reduction tightness.) This is particularly relevant since obtaining a WROB-CCA secure scheme from a non-robust one is easier than obtaining an SROB-CCA one [2].

To some extent, the fact that the "qualitative" guarantees of weak and strong robustness coincide stems from the assumed resource $\lessdot$. Since $\lessdot$ allows the adversary to inject arbitrary ciphertexts to arbitrary receivers, there is no incentive to send *the same* (faked) ciphertext to two or more different users; the adversary could also send different ciphertexts. Technically, from a network that allows the adversary to inject one message to multiple receivers more "easily" than it allows him to inject different messages, a strongly robust scheme indeed achieves a "better" resource than a weakly robust one; in the weakly robust case the adversary can inject messages to *several* receivers "easily," in the strongly robust case *only to one*. We think, however, that such a network guarantee (injecting several different messages is more difficult) would have to be justified by a particular application and should not be the focus of a general-purpose discussion as ours.

Theorem 2. *An encryption scheme that is* IND-CCA-, 1-sided-IK-CCA-, *and* SROB-CCA-*secure will transform* $\lessdot$ *and* $(\longleftarrow\!\!\bullet)^n$ *into* $-\!\!\Diamond\!\!\!\lessdot\!:$. *More formally, there exist a simulator* σ' *and reductions* $\mathbf{A}_{nq}(\cdot)$, $\mathbf{A}'_q(\cdot)$, $\mathbf{A}''_q(\cdot)$, $\mathbf{A}'''_{nq}(\cdot)$ *such that*

$$\Delta^{\mathbf{D}}\left(\mathsf{enc}^A\mathsf{dec}^{B_1}\ldots\mathsf{dec}^{B_n}\bot^E\left[\overset{q}{\lessdot},\longleftarrow\!\!\bullet^n\right],\bot^E(\overset{q}{-\!\!\Diamond\!\!\!\lessdot\!:})\right)\le\Gamma^{\mathbf{A}_{nq}(\mathbf{D})}\left(\mathbf{G}^{\mathsf{s\text{-}rob}}\right),\quad(3)$$

and

$$\Delta^{\mathbf{D}}\left(\mathsf{enc}^A\mathsf{dec}^{B_1}\ldots\mathsf{dec}^{B_n}\left[\overset{q}{\lessdot},\longleftarrow\!\!\bullet^n\right],\sigma'^E(\overset{q}{-\!\!\Diamond\!\!\!\lessdot\!:})\right)$$
$$\le qn\cdot\Phi^{\mathbf{A}'_q(\mathbf{D})}\left(\mathbf{G}^{\mathsf{ind\text{-}cca}}\right)+qn\cdot\Phi^{\mathbf{A}''_q(\mathbf{D})}\left(\mathbf{G}^{\mathsf{1\text{-}sided\text{-}ik\text{-}cca}}\right)+\Gamma^{\mathbf{A}'''_{nq}(\mathbf{D})}\left(\mathbf{G}^{\mathsf{s\text{-}rob}}\right).\quad(4)$$

5 Conclusion and Possible Extensions

We analyzed the problem of achieving confidentiality for a receiver-anonymous channel; our results are the constructive counterpart of the notions discussed in [5,2]. In particular, we showed that confidentiality, key privacy, and weak robustness are indeed sufficient for such a scheme to be useful, and that (slightly relaxed versions of) these are indeed necessary. We have also discussed why strong robustness is not necessary in this context. Our results do not only support the trust in existing schemes and constructions; they also show that the simpler and more efficient weakly robust schemes (see [2]) can be used safely.

Our constructive statements help explore the boundary between cryptography and traffic analysis. For example, an (active) instance of the latter, conditional delivery, cannot be prevented by end-to-end encryption (even if it has all the properties we suggest); indeed countermeasures against such attacks at the application level are critical to provide any meaningful traffic analysis resistance. Our ideal resource, thus, does not yet correspond directly to the black-box system models used by traffic analysis research, but is a component upon which such a model could be based. In contrast to our model here, traffic analysis frameworks usually consider restricted attackers that observe only parts of the system and a probabilistic model for sender and receiver behavior.[7]

Protocols in which encrypted messages are processed by multiple parties can, to some extent, prevent conditional deliveries. In a MIX-network, for instance, the attacker cannot direct a multi-layered ciphertext at a particular recipient, as he will be unable to remove the outer ciphertext layers. Thus receiver-anonymous communication using onions, threshold decryption, or verifiable re-randomization bypasses our impossibility result, instead requiring additional (distributed) trust in third parties.

Our study of anonymity properties of end-to-end encryption is only a first step in the constructive modeling of resources with useful anonymity properties and constructions thereof. The general paradigm of examining the security of anonymity-preserving cryptographic schemes in a constructive manner can (and should) be applied to other schemes as well, including topics such as anonymous signature schemes and key-exchange protocols.

Acknowledgments

Ueli Maurer and Björn Tackmann were supported by the Swiss National Science Foundation (SNF), project no. 200020-132794. Daniele Venturi acknowledges support from the Danish National Research Foundation, the National Science Foundation of China (under the grant 61061130540), the Danish Council for Independent Research (under the DFF Starting Grant 10-081612) and also from the CFEM research center within which part of this work was performed.

References

1. Abadi, M., Fournet, C.: Private authentication. Theor. Comput. Sci. 322(3), 427–476 (2004)
2. Abdalla, M., Bellare, M., Neven, G.: Robust encryption. In: TCC. LNCS, vol. 5978, pp. 480–497. Springer (2010)
3. Backes, M., Goldberg, I., Kate, A., Mohammadi, E.: Provably secure and practical onion routing. In: Chong, S. (ed.) CSF. pp. 369–385. IEEE (2012)

[7] In this aspect, our analysis plays a role similar to the cryptographic analysis of an onion routing protocol in [3], which provides a formal foundation for the traffic analysis of onion routing in [13]. Our analysis targets a cryptographic primitive and not a protocol, and can thus be more detailed.

4. Beimel, A., Dolev, S.: Buses for anonymous message delivery. Journal of Cryptology 16(1), 25–39 (2003)
5. Bellare, M., Boldyreva, A., Desai, A., Pointcheval, D.: Key-privacy in public-key encryption. In: ASIACRYPT. LNCS, vol. 2248, pp. 566–582. Springer (2001)
6. Bellare, M., Rogaway, P.: The security of triple encryption and a framework for code-based game-playing proofs. In: EUROCRYPT. LNCS, vol. 4004, pp. 409–426. Springer (2006)
7. Camenisch, J., Lysyanskaya, A.: A formal treatment of onion routing. In: CRYPTO. LNCS, vol. 3621, pp. 169–187. Springer (2005)
8. Canetti, R., Krawczyk, H.: Security analysis of IKE's signature-based key-exchange protocol. In: CRYPTO. LNCS, vol. 2442, pp. 27–52. Springer (2002)
9. Canetti, R., Krawczyk, H., Nielsen, J.: Relaxing chosen-ciphertext security. In: CRYPTO. LNCS, vol. 2729, pp. 262–582. Springer (2003)
10. Dingledine, R., Mathewson, N., Syverson, P.: Tor: The second-generation onion router. In: Proceedings of the 13th USENIX Security Symposium (August 2004)
11. Farshim, P., Libert, B., Paterson, K.G., Quaglia, E.A.: Robust encryption, revisited. In: PKC. pp. 352–368. Springer (2013)
12. Feigenbaum, J., Johnson, A., Syverson, P.F.: A model of onion routing with provable anonymity. In: Financial Crypto. LNCS, vol. 4886, pp. 57–71. Springer (2007)
13. Feigenbaum, J., Johnson, A., Syverson, P.F.: Probabilistic analysis of onion routing in a black-box model. ACM Trans. Inf. Syst. Secur. 15(3), 14 (2012)
14. Hevia, A., Micciancio, D.: An indistinguishability-based characterization of anonymous channels. In: PETS. LNCS, vol. 5134, pp. 24–43. Springer (2008)
15. Ishai, Y., Kushilevitz, E., Ostrovsky, R., Sahai, A.: Cryptography from anonymity. In: FOCS. pp. 239–248. IEEE Computer Society (2006)
16. Kohlweiss, M., Maurer, U., Onete, C., Tackmann, B., Venturi, D.: Anonymity-preserving public-key encryption: A constructive approach. Cryptology ePrint Archive, Report 2013/238, `http://eprint.iacr.org/`
17. Maurer, U.: Indistinguishability of random systems. In: EUROCRYPT. LNCS, vol. 2332, pp. 110–132. Springer (2002)
18. Maurer, U., Renner, R.: Abstract cryptography. In: Innovations in Computer Science. Tsinghua University Press (2011)
19. Maurer, U., Rüedlinger, A., Tackmann, B.: Confidentiality and integrity: A constructive perspective. In: TCC. LNCS, Springer (2012)
20. Maurer, U., Schmid, P.: A calculus for security bootstrapping in distributed systems. Journal of Computer Security 4(1), 55–80 (1996)
21. Maurer, U., Tackmann, B.: On the soundness of Authenticate-then-Encrypt: Formalizing the malleability of symmetric encryption. In: ACM CCS. ACM (2010)
22. Mohassel, P.: A closer look at anonymity and robustness in encryption schemes. In: ASIACRYPT. LNCS, vol. 6477, pp. 501–518. Springer (2010)
23. Nagao, W., Manabe, Y., Okamoto, T.: Relationship of three cryptographic channels in the UC framework. In: ProvSec. LNCS, vol. 5324, pp. 268–282. Springer (2008)
24. Onete, C., Venturi, D.: Security & indistinguishability in the presence of traffic analysis. Cryptology ePrint Archive, Report 2011/260 (2011)
25. Pfitzmann, A., Waidner, M.: Networks without user observability. In: EUROCRYPT. pp. 245–253 (1985)
26. Waters, B.R., Felten, E.W., Sahai, A.: Receiver anonymity via incomparable public keys. In: ACM CCS. pp. 112–121 (2003)
27. Yang, G., Wong, D.S., Deng, X., Wang, H.: Anonymous signature schemes. In: PKC. LNCS, vol. 3958, pp. 347–363. Springer (2006)