A preliminary version of this work appears in Advances in Cryptology—ASIACRYPT 2013. This is the full version.

Constructing Confidential Channels from Authenticated Channels—Public-Key Encryption Revisited

Sandro Coretti, Ueli Maurer, and Björn Tackmann

Department of Computer Science
ETH Zürich

Sunday 3rd November, 2013

Abstract

The security of public-key encryption (PKE), a widely-used cryptographic primitive, has received much attention in the cryptographic literature. Many security notions for PKE have been proposed, including several versions of CPA-security, CCA-security, and non-malleability. These security notions are usually defined in terms of a certain game that an efficient adversary cannot win with non-negligible probability or advantage.

If a PKE scheme is used in a larger protocol, then the security of this protocol is proved by showing a reduction of breaking a certain security property of the PKE scheme to breaking the security of the protocol. A major problem is that each protocol requires in principle its own tailor-made security reduction. Moreover, which security notion of the PKE should be used in a given context is a priori not evident; the employed games model the use of the scheme abstractly through oracle access to its algorithms, and the sufficiency for specific applications is neither explicitly stated nor proven.

In this paper we propose a new approach to investigating the application of PKE, following the constructive cryptography paradigm of Maurer and Renner (ICS 2011). The basic use of PKE is to enable confidential communication from a sender A to a receiver B, assuming A is in possession of B's public key. One can distinguish two relevant cases: The (non-confidential) communication channel from A to B can be authenticated (e.g., because messages are signed) or non-authenticated. The application of PKE is shown to provide the construction of a secure channel from A to B from two (assumed) authenticated channels, one in each direction, or, alternatively, if the channel from A to B is completely insecure, the construction of a confidential channel without authenticity. Composition then means that the assumed channels can either be physically realized or can themselves be constructed cryptographically, and also that the resulting channels can directly be used in any applications that require such a channel. The composition theorem shows that several construction steps can be composed, which guarantees the soundness of this approach and eliminates the need for separate reduction proofs.

We also revisit several popular game-based security notions (and variants thereof) and give them a constructive semantics by demonstrating which type of construction is achieved by a PKE scheme satisfying which notion. In particular, the necessary and sufficient security notions for the above two constructions to work are CPA-security and a variant of CCA-security, respectively.

1 Introduction

Public-key encryption (PKE) is a cryptographic primitive devised to achieve confidential communication in a context where only authenticated (but not confidential) communication channels are available [12, 37]. The cryptographic security of PKE is traditionally defined in terms of a certain distinguishing game in which no efficient adversary is supposed to achieve a non-negligible advantage. There exists quite a wide spectrum of security notions and variants thereof. These notions are motivated by clearly captured attacks (e.g., a chosen-ciphertext attack) that should be prevented, but in some cases they seem to have been proposed mainly because they are stronger than previous notions or can be shown to be incomparable.

This raises the question of which security notion for PKE is suitable or necessary for a certain higher-level protocol (using PKE) to be secure. The traditional answer to this question is that for each protocol one (actually, a cryptography expert) needs to identify the right security notion and provide a reduction proof to show that a PKE satisfying this notion yields a secure protocol.[1]

An alternative approach is to capture the semantics of a security notion by characterizing directly what it achieves, making explicit in which applications it can be used securely. The constructive cryptography paradigm [26, 27] was proposed with this general goal in mind. Resources such as different types of communication channels are modeled explicitly, and the goal of a cryptographic protocol or scheme π is to *construct* a stronger or more useful resource S from an assumed resource R, denoted as $R \xmapsto{\pi} S$. Two such construction steps can then be composed, i.e., if we additionally consider a protocol ψ that assumes the resource S and constructs a resource T, the composition theorem states that

$$R \xmapsto{\pi} S \quad \wedge \quad S \xmapsto{\psi} T \qquad \Longrightarrow \qquad R \xmapsto{\psi \circ \pi} T,$$

where $\psi \circ \pi$ denotes the composed protocol.

Following the constructive paradigm, a protocol is built in a modular fashion from isolated construction steps. A security proof guarantees the soundness of one such step, and each proof is independent of the remaining steps. The composition theorem then guarantees that several such steps can be composed. While the general approach to protocol design based on reduction proofs is in principle sound, it is substantially more complex, more error-prone, and not suitable for re-use. This is part of the reason why it is generally not applied to the design of real-world protocols (e.g., TLS), which in turn is the main reason for the large number of protocol flaws discovered in the past. A major goal in cryptography must be to break the cycle of flaw discovery and fixes by providing solid proofs. Modularity appears to be the key in achieving this goal.

In this spirit, we treat the use of PKE as such a construction step. The contributions of this paper are two-fold. First, we show how one can construct, using PKE, confidential channels from authenticated and insecure channels (cf. Section 1.1 and Section 3). Second, we revisit several known game-based security notions (and variants thereof) and give them a constructive semantics, providing an explicit understanding of the application contexts for which a given notion is suitable (cf. Section 1.2 and Section 4). In Section 1.3 we describe how our results, although stated in a simpler setting, capture settings with multiple senders and the notion of corruption that exists in other frameworks, and in Section 1.4 we contrast the constructive paradigm with the approach of idealizing the properties of cryptographic schemes. Related work is discussed in Section 1.5.

1.1 Constructing Confidential Channels using Public-Key Encryption

From the perspective of constructive cryptography [26, 27], the purpose of a public-key encryption scheme is to construct a confidential channel from non-confidential channels. Here, a channel is a resource (or functionality) that involves a sender, a receiver, and—to model channels with different levels of security—an attacker. A channel generally allows the sender to transmit a message to the receiver; the security properties of a particular channel are captured by the capabilities available to the attacker, which might, e.g., include reading or modifying the messages in transmission.

The parties access the channel through interfaces that the channel provides and that are specific for each party. For example, the sender's interface allows to input messages, and the receiver's interface allows to receive them. We refer to the interfaces by labels A, B, and E, where A and B are the sender's and the receiver's interfaces, respectively, and E is the adversary's

[1]Note that this work is orthogonal to the foundational problem of designing practical PKE schemes provably satisfying certain security notions, based on realistic hardness assumptions. The seminal CCA-secure PKE scheme based on the DDH-assumption by Cramer and Shoup [10, 11] falls into this category, as do, e.g., [14, 35, 21, 23, 38].

interface. In this work, we consider the following four types of channels (from A to B; channels in the opposite direction are defined analogously), using a notation based on [28]:[2]

- An *insecure channel*, denoted $- \twoheadrightarrow$, allows the adversary to read, deliver, and to delete all messages input at A, as well as to inject his own messages.
- An *authenticated channel*, denoted $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow$, still allows to read all messages, but the adversary is limited to forwarding or deleting messages input at interface A.
- A *confidential channel*, denoted $-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$, only leaks the length of the messages but does not necessarily prevent injections.
- A *secure channel*, denoted $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$, also only leaks the message length, and only allows the adversary to forward or delete messages input at A.

To use public-key encryption, the receiver initially generates a key pair and transmits the public key to the sender. The sender needs to obtain the correct public key, which corresponds to assuming that the channel from B to A is authenticated ($\longleftarrow\!\!\bullet$[3]). To transmit a message confidentially, the sender then encrypts the message under the received public key and sends the ciphertext to the receiver over a channel that could be authenticated or completely insecure.

The exact type of channel that is constructed depends on the type of assumed channel used to transmit the ciphertext to the receiver: We show that if the assumed channel is authenticated ($\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow$) and the PKE scheme is ind-cpa-secure, the constructed channel is a secure channel ($\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$). If the assumed channel is insecure ($- \twoheadrightarrow$) and the PKE scheme is ind-cca-secure, the constructed channel is only confidential ($-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$). Using the above notation, for protocols π and π' based on ind-cpa and ind-cca encryption schemes, respectively, these constructions can be written as

$$[\longleftarrow\!\!\bullet, \bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow] \quad \stackrel{\pi}{\Longmapsto} \quad \bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet \qquad \text{and} \qquad [\longleftarrow\!\!\bullet, - \twoheadrightarrow] \quad \stackrel{\pi'}{\Longmapsto} \quad -\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet,$$

where the bracket notation means that both resources in the brackets are available.

The notion of constructing the confidential (or secure) channel from the two assumed non-confidential ones is made precise in a simulation-based sense [27, 26], where the simulator can be interpreted as translating all attacks on the protocol into attacks on the constructed (ideal) channel. As the constructed channel is secure by definition, there are no attacks on the protocol.

The composability of the construction notion then means that the constructed channel can again be used as an assumed resource (possibly along with additional assumed or constructed resources) in other protocols. For instance, if a higher-level protocol uses the confidential channel to transmit a message together with a shared secret value in order to achieve an additionally authenticated (and hence fully secure) transmission of the message, then the proof of this protocol is based on the "idealized" confidential channel and does not (need to) include a reduction to the security of the encryption scheme. In the same spirit, the authenticated channel from B to A could be a physically authenticated channel, but it could also be constructed by using, for instance, a digital signature scheme to authenticate the transmission of the public key (which is done by certificates in practice).

1.2 Constructive Semantics of Game-Based Security Notions

Security properties for PKE are often formalized via a game between a hypothetical challenger and an attacker. We assign constructive semantics to several existing game-based definitions by first characterizing the appropriate assumed and constructed resources and then showing that the "standard use" of a PKE scheme over those channels (as illustrated in Section 1.1) achieves the construction if (and sometimes only if) it has the considered property.[4]

[2] The "$\bullet$" in the notation signifies that the capabilities at the marked interface, i.e., sending or receiving, are exclusive to the respective party. If the "$\bullet$" is missing, the adversary also has these capabilities. The $\diamond$-symbol is explained in Section 2.4, and the "double heads" of the arrows indicate that multiple messages can be transmitted.

[3] The simple arrow indicates that $\longleftarrow\!\!\bullet$ is a single-use channel, i.e., only one message can be transmitted.

[4] We point out that our negative results do *not* rule out the existence of other protocols that are derived from the scheme in some possibly more complicated way; those could still achieve the respective construction.

In particular, we show that ind-cpa-security is not only sufficient but also necessary for constructing a secure channel from two authenticated channels. For the construction of a confidential channel from an authenticated and an insecure channel, it turns out that ind-cca-security, while sufficient, is unnecessarily strong. The transformation only requires the weaker notion of ind-rcca-security, which was introduced by Canetti et al. [9] to avoid the artificial strictness of ind-cca. We continue the analysis of ind-cca-security and follow up on work by Bellare et al. [6], where several non-equivalent definitional variants are considered. We show that only the stricter notions they consider are sufficient for the channel construction, leaving the exact semantics of the weaker notions unclear.

We also consider non-adaptive chosen-ciphertext security (ind-cca1) and non-malleability (nm-cpa). We show that both notions correspond to transformations between somewhat artificial channels, but might still be useful for specific applications.

1.3 Capturing Settings with Multiple, Potentially Corrupted Senders

Although our security definitions for public-key encryption are phrased in a setting where there is only one legitimate sender (at the A-interface), our treatment captures the setting with multiple senders. What is needed to formalize this more general case explicitly is a lifting of the setting with interfaces A, B, and E into the multi-party setting with many senders. In the case where all senders in the multi-sender setting faithfully follow the protocol, this lifting simply relates all those sender interfaces to the single sender interface in the setting with three interfaces.

In a scenario with multiple senders, it is important to formulate the guarantees that are maintained if one or more of the senders deviate from the protocol because their machines are controlled by some attacker (or virus). This is captured in most security frameworks by considering an external adversary that has the capability of corrupting some of the parties. In the context of PKE and secure communication, the goal is to still provide confidentiality guarantees to non-corrupted senders. (If the receiver is corrupted, then no security can be guaranteed.)

The ability of an attacker to act on behalf of corrupted senders means that it can directly send (potentially bogus) ciphertexts to the receiver, even if the communication to the receiver is authenticated. This capability corresponds exactly to the case of assuming only an unauthenticated channel, where the messages are injected via the E-interface. Hence, our treatment extends to the case of (static) sender corruption by considering the lifting that relates the interfaces of the senders in the multi-party scenario to the A-interface in the three-party setting, and provides the capabilities of the statically corrupted parties also at the E-interface. The lifting mappings described above are generic for constructive cryptography and not specific to public-key encryption, and hence formalizing them is not in the scope of the current paper.

In summary, the security of public-key encryption in the presence of potentially (statically) corrupted senders corresponds exactly to the construction of a confidential channel $-\!\diamond\!\!\twoheadrightarrow\!\bullet$ from one insecure channel $- \twoheadrightarrow$ and one authenticated channel $\longleftarrow\!\bullet$ in the opposite direction, as discussed in Section 1.1. This implies that in the presence of (static) corruption, ind-rcca security is required and sufficient both in the case where the channel from the sender to the receiver is authenticated, and also where it is not authenticated.

1.4 Idealizing the Properties of Cryptographic Schemes vs. Constructing Resources

The security guarantees that one requires from a cryptographic scheme can be modeled in fundamentally different ways, even within a single formal security framework. One approach, which underlies the public-key encryption functionality $\mathcal{F}_{\text{PKE}}$ in [9], is to idealize the properties of the algorithms that comprise the scheme. Such a functionality corresponds to a cryptographic scheme, and its interfaces closely resemble the interfaces of the algorithms (although, e.g., the private key is never output by $\mathcal{F}_{\text{PKE}}$). In such a treatment, elements that are essential for using the scheme, such as the ciphertext or the public key, will still appear in the functionality, but they are idealized in that, e.g., the ciphertext is independent of the corresponding plaintext; the idealized scheme is unbreakable by definition.

Another—fundamentally different—approach is to explicitly model *resources* that are available to one or more parties. The communication channels we describe in Section 1.1 can be considered *network resources*; there are also functionalities in the UC framework, such as $\mathcal{F}_{\text{AUTH}}$ or $\mathcal{F}_{\text{SC}}$ in [8], that can be interpreted in this way. More generally, one can also think of randomness, memory, or even computation as resources of this type. Following the constructive paradigm, the guarantees of a cryptographic scheme are *not* a resource, but modeled as the guarantee that the scheme transforms one (assumed) resource into another (constructed) resource.[5] Compared to ideal functionalities of the above type, the description of resources tends to be simpler and easier to understand. For example, in the case of public-key encryption, the confidential channel does not need to specify implementation artifacts such as ciphertexts or public keys.

While both approaches allow to divide the security proof of a composite protocol into several separate steps that can be proven independently, only the second approach enables a fully modular protocol design. Each sub-protocol achieves a well-defined construction step transforming a resource R into a resource S, which abstracts from how S is achieved. A higher-level protocol can thus use such a resource S independently of how it is obtained, and the construction of S can be replaced with a different one without affecting the design or proof of the higher-level protocol. Concretely, a protocol using the resource $-\!\diamond\!\!\twoheadrightarrow\!\!\bullet$ does not depend on whether or not the channel is constructed by a public-key encryption scheme, whereas a protocol using the functionality $\mathcal{F}_{\text{PKE}}$ will always be specific to this step.

1.5 Related Work

Game-Based Security. The study of PKE security was initiated by Goldwasser and Micali [19], who introduced the notions of indistinguishability (of encryptions) and semantic security. Yao [39] introduced another definition, based on computational entropy, which Micali et al. [33] proved equivalent to variants of the definitions by [19]. Goldreich [15, 16] made important modifications to the definitions and also dealt with uniform adversaries. Today's standard notion is *indistinguishability* under chosen-plaintext attack, ind-cpa. This definition has been strengthened by considering more powerful attackers that can additionally obtain decryptions of arbitrary ciphertexts. This lead to the notions of ind-cca1 and ind-cca2 (e.g., [34, 40]). Different variants of ind-cca2-security were analyzed by Bellare et al. [6]. Canetti et al. [9] introduced the weaker notion ind-rcca that suffices for many applications of PKE.

A second important security property is *non-malleability*, introduced by Dolev et al. [13]. Informally, it requires that an adversary cannot change a ciphertext into one that decrypts to a related message. The original definition of [13] was partially simulation-based. A purely game-based variant of this definition was considered in [5]. The two definitions were proven equivalent to yet another, simpler one by Bellare and Sahai [3]. Similarly to the indistinguishability notions, considering various attack models leads to the (standard) notions of nm-cpa, nm-cca1, and nm-cca2. Relations between these notions and those related to indistinguishability were investigated by Bellare et al. [5].

Real-World/Ideal-World Security. The idea of defining protocol security with respect to an ideal execution was first proposed by Goldreich et al. [17], where a simulator was used to formalize that whatever the adversary can achieve in an attack on the protocol he can also achieve in the ideal execution. First formal treatments of this approach were by Goldwasser and Levin [18], Micali and Rogaway [32], and Beaver [2] in the context of multi-party computation. The concept of a simulator can be traced back to the seminal work by Goldwasser et al. [20], who introduced it in the context of zero-knowledge proofs.

General security frameworks that allow the formalization of arbitrary functionalities to be realized by cryptographic protocols have been introduced by Canetti [7] as Universal Composability

[5]By contrast, a typical UC security statement is that a cryptographic scheme implements some functionality. While statements about *hybrid* protocols in UC appear similar to constructive statements, they are less expressive since, e.g., the UC framework technically does not allow to make statements about assuming only *bounded* resources, as protocols that use hybrid functionalities can always instantiate arbitrarily many functionalities of a given type.

(UC) as well as by Pfitzmann and Waidner [36] and Backes et al. [1] as Reactive Simulatability (RSIM). Treatments of PKE exist in both frameworks. As mentioned in Section 1.4, the treatment in UC is with respect to an "ideal PKE" functionality. Realizing this functionality is equivalent to ind-cca2-security [9].

Canetti and Krawczyk [8] formulate UC functionalities that model different types of communication channels and can be interpreted as network resources; they show that their secure channels functionality can be realized by key exchange and symmetric encryption. They do not treat public-key encryption (beyond what is implied by viewing the above scheme as KEM-DEM), and the model of computation underlying the UC framework does not even allow to directly formulate the security condition for the non-authenticated case as this would require to instantiate the authenticated channels functionality in a "directed" way (but a hybrid protocol in the UC framework can always instantiate the functionality in both directions).

The formalization of the functionalities in [36] is closer to our approach, but less modular and hence more complicated since they immediately analyze the schemes in a multi-party scenario; the treatment is restricted to and directly proves the case where the authenticated transmission of the ciphertexts is achieved by digital signatures instead of using a generic composition statement. More generally, both frameworks [7] and [36] are designed from a bottom-up perspective (starting from a selected machine model), whereas we follow the top-down approach of [27], which leads to simpler, more abstract definitions and statements.

Maurer et al. [30] described symmetric encryption as the construction of confidential channels from non-confidential channels and shared keys, and compared the security definitions they obtained with previous game-based definitions. The goal of this work is to provide a comparable treatment for the case of public-key encryption. In the same spirit, specific anonymity-related properties of public-key encryption and their relation to the construction of receiver-anonymous channels have been discussed by Kohlweiss et al. [24].

2 Preliminaries

2.1 Systems: Resources and Converters, Distinguishers, and Reductions

Resources and converters (see below) are modeled as systems. At the highest level of abstraction (following the hierarchy in [27]), systems are objects with interfaces by which they connect to (interfaces of) other systems; each interface is labeled with an element of a label set and connects to only a single other interface. This concept of *abstract systems* captures the topological structures that result when multiple systems are connected in this manner.

The abstract systems concept, however, does not model the behavior of systems, i.e., *how* the systems interact via their interfaces. Consequently, statements about cryptographic protocols are statements at the next (lower) abstraction level. In this work, we describe all systems in terms of (probabilistic) discrete systems, which we explain in Section 2.2.

Resources and Converters. *Resources* in this work are systems with three interfaces labeled by A, B, and E. A protocol is modeled is a pair of two so-called *converters* (one for each honest party), which are directed in that they have an *inside* and an *outside* interface, denoted by `in` and `out`, respectively. As a notational convention, we generally use upper-case, bold-face letters (e.g., $\mathbf{R}$, $\mathbf{S}$) or channel symbols (e.g., $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow$) to denote resources and lower-case Greek letters (e.g., α, β) or sans-serif fonts (e.g., enc, dec) for converters. We denote by Φ the set of all resources and by Σ the set of all converters.

The topology of a composite system is described using a term algebra, where each expression starts from one (or more) resources on the right-hand side and is subsequently extended with further terms on the left-hand side. An expression is interpreted in the way that all interfaces of the system it describes can be connected to interfaces of systems which are appended on the left. For instance, for a single resource $\mathbf{R} \in \Phi$, all its interfaces A, B, and E are accessible.

For $I \in \{A, B, E\}$, a resource $\mathbf{R} \in \Phi$, and a converter $\alpha \in \Sigma$, the expression $\alpha^I \mathbf{R}$ denotes the composite system obtained by connecting the inside interface of α to interface I of $\mathbf{R}$; the outside interface of α becomes the I-interface of the composite system. The system $\alpha^I\mathbf{R}$ is again a resource (cf. Figure 2 on page 11).

For two resources $\mathbf{R}$ and $\mathbf{S}$, $[\mathbf{R},\mathbf{S}]$ denotes the parallel composition of $\mathbf{R}$ and $\mathbf{S}$. For each $I \in \{A, B, E\}$, the I-interfaces of $\mathbf{R}$ and $\mathbf{S}$ are merged and become the *sub-interfaces* of the I-interface of $[\mathbf{R},\mathbf{S}]$, which we denote by $I.1$ and $I.2$.[6] A converter α that connects to the I-interface of $[\mathbf{R},\mathbf{S}]$ has two inside sub-interfaces, denoted by $\mathtt{in}.1$ and $\mathtt{in}.2$, where the first one connects to $I.1$ and the second one connects to $I.2$.

Any two converters α and β can be composed sequentially by connecting the inside interface of β to the outside interface of α, written $\beta\circ\alpha$, with the effect that $(\beta\circ\alpha)^I\mathbf{R} = \beta^I\alpha^I\mathbf{R}$. Moreover, converters can also be taken in parallel, denoted by $[\alpha,\beta]$, with the effect that $[\alpha,\beta]^I[\mathbf{R},\mathbf{S}] = [\alpha^I\mathbf{R},\beta^I\mathbf{S}]$.

We assume the existence of an identity converter $\mathsf{id} \in \Sigma$ with $\mathsf{id}^I\mathbf{R} = \mathbf{R}$ for all resources $\mathbf{R} \in \Phi$ and interfaces $I \in \{A, B, E\}$ and of a special converter $\bot \in \Sigma$ with an inactive outside interface.

Distinguishers. A *distinguisher* is a special type of system $\mathbf{D}$ that connects to all interfaces of a resource $\mathbf{U}$ and outputs a single bit at the end of its interaction with $\mathbf{U}$. In the term algebra, this appears 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[7]

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

The advantage of a class $\mathcal{D}$ of distinguishers is defined as $\Delta^{\mathcal{D}}(\mathbf{U},\mathbf{V}) := \sup_{\mathbf{D}\in\mathcal{D}} \Delta^{\mathbf{D}}(\mathbf{U},\mathbf{V})$.

The distinguishing advantage measures how much the output distribution of $\mathbf{D}$ differs when it is connected to either $\mathbf{U}$ or $\mathbf{V}$. There is an equivalence notion on systems (which is defined on the discrete systems level), denoted by $\mathbf{U} \equiv \mathbf{V}$, which implies that $\Delta^{\mathbf{D}}(\mathbf{U},\mathbf{V}) = 0$ for all distinguishers $\mathbf{D}$.

Note that the distinguishing advantage is a pseudo-metric.[8] 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 distinguishers $\mathbf{D}$. There is an equivalence notion on systems (which is defined on the discrete systems level), denoted by $\mathbf{U} \equiv \mathbf{V}$, which means that $\Delta^{\mathbf{D}}(\mathbf{U},\mathbf{V})$ for all distinguishers $\mathbf{D}$.

Games. We capture games defining security properties as distinguishing problems in which an adversary $\mathbf{A}$ tries to distinguish between two *game systems* $\mathbf{G}_0$ and $\mathbf{G}_1$. Game systems (or simply *games*) are single-interface systems, which appear, similarly to resources, on the right-hand side of the expressions in the term algebra. The adversary is a distinguisher that connects to a game (instead of a resource). We denote by $\mathcal{A}$ the class of *all* adversaries for games.

Reductions. When relating two distinguishing problems, it is convenient to use a special type of system $\mathbf{C}$ that translates one setting into the other. Formally, $\mathbf{C}$ is a converter that has an *inside* and an *outside* interface. When it is connected to a system $\mathbf{S}$, which is denoted by $\mathbf{CS}$,[9] the inside interface of $\mathbf{C}$ connects to the (merged) interface(s) of $\mathbf{S}$ and the outside interface of $\mathbf{C}$ is the interface of the composed system. $\mathbf{C}$ is called a *reduction system* (or simply *reduction*).

To reduce distinguishing two systems $\mathbf{S},\mathbf{T}$ to distinguishing two systems $\mathbf{U},\mathbf{V}$, one exhibits a reduction $\mathbf{C}$ such that $\mathbf{CS} \equiv \mathbf{U}$ and $\mathbf{CT} \equiv \mathbf{V}$.[10] Then, for all distinguishers $\mathbf{D}$, we have $\Delta^{\mathbf{D}}(\mathbf{U},\mathbf{V}) = \Delta^{\mathbf{D}}(\mathbf{CS},\mathbf{CT}) = \Delta^{\mathbf{DC}}(\mathbf{S},\mathbf{T})$. The last equality follows from the fact that $\mathbf{C}$ can also be thought of as being part of the distinguisher.

[6]Hence, the parallel composition is not commutative.

[7]Note that two random experiments are involved in the definition of $\Delta^{\mathbf{D}}(\mathbf{U},\mathbf{V})$. The first probability is over (the randomness of) $\mathbf{D}$ and $\mathbf{U}$ and the second one over $\mathbf{D}$ and $\mathbf{V}$.

[8]A pseudo-metric $\delta : X \times X \to \mathbb{R}_0^+$ satisfies (a) $\delta(x,x) = 0$, (b) $\delta(x,y) = \delta(y,x)$, and (c) $\delta(x,z) \leq \delta(x,y) + \delta(y,z)$ for all $x,y,z \in X$.

[9]For readability we sometimes write $\mathbf{C}(\mathbf{S})$.

[10]For instance, we consider reductions from distinguishing game systems to distinguishing resources. Then, $\mathbf{C}$ connects to a game on the inside and provides interfaces A, B, and E on the outside.

2.2 Discrete Systems

Protocols that communicate by passing messages and the respective resources are described as (probabilistic) discrete systems. Their behavior can be formalized by random systems as in [25], i.e., as families of conditional probability distributions of the outputs (as random variables) given all previous inputs and outputs of the system. For systems with multiple interfaces, the interface to which an input or output is associated is explicitly specified as part of the input or output. For the restricted (but here sufficient) class of systems that for each input provide (at most) a single output, an execution of a collection of systems is defined as the consecutive evaluation of the respective random systems (similarly to the models in [7, 22]).

2.3 The Notion of Construction

Recall that we consider resources with interfaces A, B, and E, where A and B are interfaces of honest parties and E is the interface of the adversary. We formalize the security of protocols via the following notion of *construction*, which was introduced in [26] (and is a special case of the abstraction notion from [27]):

Definition 1. *Let* Φ *and* Σ *be as in Section 2.1. A protocol* $\pi = (\pi_1, \pi_2) \in \Sigma^2$ constructs resource $\mathbf{S} \in \Phi$ from resource $\mathbf{R} \in \Phi$ within ε and with respect to distinguisher class $\mathcal{D}$, *denoted*

$$\mathbf{R} \quad \overset{(\pi,\varepsilon)}{\Longmapsto} \quad \mathbf{S},$$

if

$$\begin{cases} \qquad\qquad \Delta^{\mathcal{D}}(\pi_1^A \pi_2^B \bot^E \mathbf{R}, \bot^E \mathbf{S}) \leq \varepsilon & (\textit{availability}) \\ \exists \sigma \in \Sigma : \quad \Delta^{\mathcal{D}}(\pi_1^A \pi_2^B \mathbf{R}, \sigma^E \mathbf{S}) \leq \varepsilon & (\textit{security}). \end{cases}$$

The availability condition captures that a protocol must correctly implement the functionality of the constructed resource in the absence of the adversary. The security condition models the requirement that everything the adversary can achieve in the setting with the assumed resource and the protocol, he can also accomplish in the setting with the constructed resource (using the simulator to translate the behavior).

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 $\pi_1^A \pi_2^B \mathbf{R}$ without affecting the security of the composed system. Security and availability are preserved under composition. More formally, if for some resources $\mathbf{R}$, $\mathbf{S}$, and $\mathbf{T}$ and protocols π and ϕ,

$$\mathbf{R} \quad \overset{(\pi,\varepsilon)}{\Longmapsto} \quad \mathbf{S} \qquad \text{and} \qquad \mathbf{S} \quad \overset{(\phi,\varepsilon')}{\Longmapsto} \quad \mathbf{T},$$

then

$$\mathbf{R} \quad \overset{(\phi \circ \pi, \varepsilon+\varepsilon')}{\Longmapsto} \quad \mathbf{T},$$

as well as

$$[\mathbf{R}, \mathbf{U}] \quad \overset{([\pi,\mathsf{id}],\varepsilon)}{\Longmapsto} \quad [\mathbf{S}, \mathbf{U}] \quad \text{and} \quad [\mathbf{U}, \mathbf{R}] \quad \overset{([\mathsf{id},\pi],\varepsilon)}{\Longmapsto} \quad [\mathbf{U}, \mathbf{S}]$$

for any resource $\mathbf{U}$. More details can be found in [26].

2.4 Channels

We consider the types of channels shown in Figure 1. Each channel initially expects a special cheating bit $b \in \{0, 1\}$ at interface E, indicating whether the adversary is present and intends to interfere with the transmission of the messages. The special converter $\bot$ (cf. Section 2.1) always sets $b = 0$. For simplicity, we will assume that whenever $\bot$ is not present, all cheating bits are set to 1.

Channel Name	Symbol	$\ell(m)$	inj
Insecure Channel	$-\twoheadrightarrow$	m	✓
Confidential Channel	$-\!\diamond\!\!\twoheadrightarrow\!\bullet$	$\vert m\vert$	✓
Authenticated Channel	$\bullet\!\!-\!\diamond\!\!\twoheadrightarrow$	m	×
Secure Channel	$\bullet\!\!-\!\diamond\!\!\twoheadrightarrow\!\bullet$	$\vert m\vert$	×

Figure 1: *Channel resources considered in this work.*

A channel from A to B with leakage ℓ and message space $\mathcal{M} \subseteq \{0,1\}^*$ is a resource with interfaces A, B, and E and behaves as follows:[11] When the i^{th} message $m \in \mathcal{M}$ is input at interface A, it is recorded as (i, m) and $(i, \ell(m))$ is output at interface E. When (dlv, i') is input at interface E and if (i', m') has been recorded, m' is delivered at interface B. If injections are permissible, when (inj, m') is input at interface E, m' is output at interface B.[12,13]

The security statements in this work are parameterized by the number of messages that are transmitted over the channels. More precisely, for each of the above channels and each $n \in \mathbb{N}$, we define the *n-bounded channel* as the one that processes (only) the first n queries at the A-interface and the first n queries at the E-interface (as described above) and ignores all further queries at these interfaces. We then require from a protocol that it constructs, for all $n \in \mathbb{N}$, the n-bounded "ideal" channel from the n-bounded assumed channel.[14] Wherever the number n is significant, such as in the theorem statements, we denote the n-bounded versions of channels by writing the n on top of the channel symbol (e.g., $\overset{n}{-\!\diamond\!\!\twoheadrightarrow\!\bullet}$); we omit it in places that are of less formal nature.

Finally, a simple-arrow symbol (e.g., $\bullet\!\!\longrightarrow$) denotes a *single-use* channel. That is, only one message may be transmitted.[15]

2.5 Public-Key Encryption Schemes

A public-key encryption (PKE) scheme with message space $\mathcal{M} \subseteq \{0,1\}^*$ and ciphertext space $\mathcal{C}$ is defined as three algorithms $\Pi = (K, E, D)$, where the key-generation algorithm K outputs a key pair $(\mathsf{pk}, \mathsf{sk})$, the (probabilistic) encryption algorithm E takes a message $m \in \mathcal{M}$ and a public key pk and outputs a ciphertext $c \leftarrow E_{\mathsf{pk}}(m)$, and the decryption algorithm takes a ciphertext $c \in \mathcal{C}$ and a secret key sk and outputs a plaintext $m \leftarrow D_{\mathsf{sk}}(c)$. The output of the decryption algorithm can be the special symbol $\diamond$, indicating an invalid ciphertext.

A PKE scheme is correct if $m = D_{\mathsf{sk}}(E_{\mathsf{pk}}(m))$ (with probability 1 over the randomness in the encryption algorithm) for all messages m and all key pairs $(\mathsf{pk}, \mathsf{sk})$ generated by K.

It will be more convenient to phrase bit-guessing games used in definitions of PKE security properties as a distinguishing problem between two game systems (cf. Section 2.1). We consider the following games, which correspond to the (standard) notions of ind-cpa (cpa for short), ind-cca2 (cca), ind-cca1 (cca1), ind-rcca (rcca), and nm-cpa (nm).[16] Informally, a scheme is secure in the sense of a notion if efficient adversaries have negligible advantage in distinguishing the two corresponding game systems.

CPA Game. Consider the systems $\mathbf{G}_0^{\mathsf{cpa}}$ and $\mathbf{G}_1^{\mathsf{cpa}}$ defined as follows: For a PKE scheme Π, both initially run the key-generation algorithm to obtain $(\mathsf{pk}, \mathsf{sk})$ and output pk. Upon (the

[11] If the cheating bit is set to $b = 0$, all messages input at the sender interface A are immediately delivered to B.

[12] Note that none of the channels prevents the adversary from reordering or replaying messages sent over the channel. The $\diamond$-symbol suggests the "internal buffer" in which the channel stores messages input at A.

[13] Note that the dlv-instruction is redundant with $-\twoheadrightarrow$, and we will ignore it throughout this work.

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

[15] The reason for not writing $\overset{1}{\bullet\!\!-\!\diamond\!\!\twoheadrightarrow}$ is that such a channel will be single-use independently of the parametrization mentioned above.

[16] We consider the so-called real-or-random versions of these games, which are equivalent to the more popular left-or-right formulations (as shown in [4] for symmetric encryption). For non-malleability, we use an indistinguishability-based version by [3].

first) query (chall, m), $\mathbf{G}_0^{\mathsf{cpa}}$ outputs an encryption $c \leftarrow E_{\mathsf{pk}}(m)$ of m and $\mathbf{G}_1^{\mathsf{cpa}}$ an encryption $c \leftarrow E_{\mathsf{pk}}(\bar{m})$, called the *challenge*, of a randomly chosen message $\bar{m}$ of length $|m|$.

CCA Games. For $b \in \{0,1\}$, system $\mathbf{G}_b^{\mathsf{cca1}}$ proceeds as $\mathbf{G}_b^{\mathsf{cpa}}$ but additionally answers decryption queries (dec, c') before the challenge is output by returning $m' \leftarrow D_{\mathsf{sk}}(c')$. $\mathbf{G}_b^{\mathsf{cca}}$ answers decryption queries at any time unless c' equals the challenge c (if defined), in which case the answer is test.

RCCA Game. Consider the systems $\mathbf{G}_0^{\mathsf{rcca}}$ and $\mathbf{G}_1^{\mathsf{rcca}}$ defined as follows: Initially, both run the key-generation algorithm to obtain $(\mathsf{pk}, \mathsf{sk})$ and output pk. Upon (the first) query (chall, m), *both* choose a random message $\bar{m}$ of length $|m|$. $\mathbf{G}_0^{\mathsf{rcca}}$ outputs $c \leftarrow E_{\mathsf{pk}}(m)$ and $\mathbf{G}_1^{\mathsf{rcca}}$ outputs $c \leftarrow E_{\mathsf{pk}}(\bar{m})$. Both systems answer decryption queries (dec, c'), but if $D_{\mathsf{sk}}(c') \in \{m, \bar{m}\}$ (if m and $\bar{m}$ are defined), the answer is test.

For more details about RCCA-security, see Section 4.2 or consult [9], where the notion was introduced.

NM Game. Consider the systems $\mathbf{G}_0^{\mathsf{nm}}$ and $\mathbf{G}_1^{\mathsf{nm}}$ defined as follows: Both initially run the key-generation algorithm to obtain $(\mathsf{pk}, \mathsf{sk})$ and output pk. Upon (the first) query (chall, m), $\mathbf{G}_0^{\mathsf{nm}}$ outputs an encryption $c \leftarrow E_{\mathsf{pk}}(m)$ of m and $\mathbf{G}_1^{\mathsf{nm}}$ an encryption $c \leftarrow E_{\mathsf{pk}}(\bar{m})$ of a randomly chosen message $\bar{m}$ of length $|m|$. When a query $(\mathsf{dec}, c_1, \ldots, c_\ell)$ is input, both systems decrypt $c_1, \ldots, c_\ell$, return the resulting plaintexts (if any of the ciphertexts equal c, the corresponding plaintexts are replaced by test), and terminate the interaction.

2.6 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 $\{\mathbf{S}_\kappa\}_{\kappa \in \mathbb{N}}$ and letting the distinguishing advantage be a real-valued function of κ. Then, for a given notion of efficiency, one can consider security w.r.t. classes of efficient distinguishers and a suitable negligibility notion. All reductions in this work are efficient with respect to the standard polynomial-time notions.

3 Constructing Confidential Channels with Public-Key Encryption

The main purpose of public-key encryption (PKE) is to achieve confidential communication. As a constructive statement, this means that we view a PKE scheme Π as a protocol, a pair of converters $(\mathsf{enc}, \mathsf{dec})$, whose goal is to construct a confidential channel from non-confidential channels. Differentiating between the two cases where the communication from the sender to the receiver is authenticated and unauthenticated, respectively, this is written as

$$[\longleftarrow\!\bullet, \bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow] \stackrel{(\mathsf{enc},\mathsf{dec})}{\Longrightarrow} \bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet \quad (1) \qquad \text{and} \qquad [\longleftarrow\!\bullet, -\!\!\twoheadrightarrow] \stackrel{(\mathsf{enc},\mathsf{dec})}{\Longrightarrow} -\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet. \quad (2)$$

In both cases, the *single-use* channel $\longleftarrow\!\bullet$ captures the ability of the sender to obtain the receiver's public key in an authenticated fashion. In construction (1), the communication from the sender A to the receiver B is authenticated, which is modeled by the channel $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow$. The goal is to achieve a secure channel $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$, which only leaks the length of the messages sent at interface A. In construction (2), the communication from A to B is completely insecure, which is captured by the insecure channel $-\!\!\twoheadrightarrow$. Here, the goal is to achieve a confidential channel $-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$, which still hides messages input at the A-interface but also allows to inject arbitrary messages at E.

In the following, we first show how a PKE scheme Π can be seen as a converter pair $(\mathsf{enc}, \mathsf{dec})$. We then prove that $(\mathsf{enc}, \mathsf{dec})$ achieves construction (1) if the underlying PKE scheme is cpa-secure, and construction (2) if the underlying PKE scheme is cca-secure.

3.1 PKE Schemes as Protocols

Let $\Pi = (K, E, D)$ be a PKE scheme. Based on Π, we define a pair of protocol converters $(\mathsf{enc}, \mathsf{dec})$ for constructions (1) and (2). Both converters have two sub-interfaces $\mathtt{in}.1$ and $\mathtt{in}.2$ on the inside, as we connect them to a resource that is a parallel composition of two other resources (cf. Section 2.1).

Converter enc works as follows: It initially expects a public key pk at the $\mathtt{in}.1$. When a message m is input at the outside interface $\mathtt{out}$, enc outputs $c \leftarrow E_{\mathsf{pk}}(m)$ at $\mathtt{in}.2$. Converter dec initially generates a key pair $(\mathsf{pk}, \mathsf{sk})$ using key-generation algorithm K and outputs pk at $\mathtt{in}.1$. When dec receives c' at $\mathtt{in}.2$, it computes $m' \leftarrow D_{\mathsf{sk}}(c')$ and, if $m' \neq \diamond$, outputs m' at the outside interface $\mathtt{out}$.

3.2 Constructing a Secure from Two Authenticated Channels

Towards proving that the protocol $(\mathsf{enc}, \mathsf{dec})$ indeed achieves construction (1), note first that the correctness of Π implies that the *availability* condition of Definition 1 is satisfied. To prove *security*, we need to exhibit a simulator σ such that the assumed resource $[\longleftarrow\!\bullet, \bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow]$ with the protocol converters is indistinguishable from the constructed resource $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$ with the simulator (cf. Figure 2).

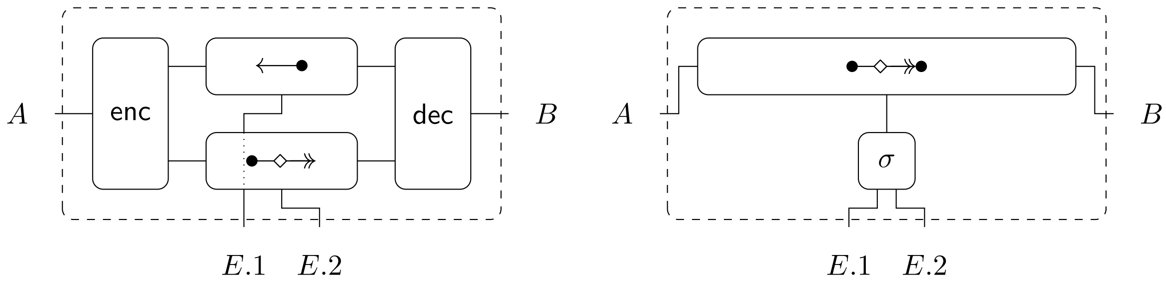

Figure 2: *Left: The assumed resource (two authenticated channels) with protocol converters* enc *and* dec *attached to interfaces A and B, denoted* $\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\bullet, \bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow]$. *Right: The constructed resource (a secure channel) with simulator σ attached to the E-interface, denoted* $\sigma^E \bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$. *In particular, σ must simulate the E-interfaces of the two authenticated channels. The protocol is secure if the two systems are indistinguishable.*

Theorem 1 implies that $(\mathsf{enc}, \mathsf{dec})$ realizes (1) if the underlying PKE scheme is cpa-secure.

Theorem 1. *There exists a simulator σ and for any $n \in \mathbb{N}$ there exists a (efficient) reduction $\mathbf{C}$ such that for every $\mathbf{D}$,*[17]

$$\Delta^{\mathbf{D}}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\bullet, \overset{n}{\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow}], \sigma^E\ \overset{n}{\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet}) \leq n \cdot \Delta^{\mathbf{DC}}(\mathbf{G}_0^{\mathsf{cpa}}, \mathbf{G}_1^{\mathsf{cpa}}).$$

Proof. First, consider the following simulator σ for interface E of $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$, which has two sub-interfaces denoted by $\mathtt{out}.1$ and $\mathtt{out}.2$ on the outside (since the real-world system has two sub-interfaces at E): Initially, σ generates a key pair $(\mathsf{pk}, \mathsf{sk})$ and outputs $(1, \mathsf{pk})$ at $\mathtt{out}.1$.[18] When it receives (i, l) at the inside interface $\mathtt{in}$, σ generates an encryption $c \leftarrow E_{\mathsf{pk}}(\bar{m})$ of a randomly chosen message $\bar{m}$ of length l and outputs (i, c) at $\mathtt{out}.2$. When (dlv, i') is input at $\mathtt{out}.2$, σ simply outputs (dlv, i') at $\mathtt{in}$. Consider the two systems

$$\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\bullet, \overset{1}{\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow}] \qquad \text{and} \qquad \sigma^E\ \overset{1}{\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet}\ .$$

Distinguishing $\mathbf{G}_0^{\mathsf{cpa}}$ from $\mathbf{G}_1^{\mathsf{cpa}}$ can be reduced to distinguishing these two systems via the following reduction system $\mathbf{C}'$, which connects to a game on the inside and provides interfaces A,

---

[17] Recall that $\overset{n}{\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow}$ denotes the channel that processes the first n messages input at interfaces A and E only.

[18] For simplicity, we assume that the public key is always delivered, i.e., that dlv is input at interface E of $\longleftarrow\!\bullet$.

B, and E on the outside (cf. Section 2.1 for details on reduction systems): Initially, $\mathbf{C}'$ takes a value pk from the game (on the inside) and outputs $(1, \mathsf{pk})$ at the (outside) $E.1$-interface. When a message m is input at the A-interface of $\mathbf{C}'$, it is passed as (chall, m) to the game. The resulting challenge c is output as $(1, c)$ at the $E.2$-interface. When $(\mathsf{dlv}, 1)$ is input at the $E.2$-interface, $\mathbf{C}'$ outputs m at interface B. We have

$$\mathbf{C}'\mathbf{G}_0^{\mathsf{cpa}} \equiv \mathsf{enc}^A\mathsf{dec}^B[\leftarrow\!\!\bullet, \bullet\!\!\overset{1}{-\!\diamond\!\!\twoheadrightarrow}] \qquad \text{and} \qquad \mathbf{C}'\mathbf{G}_1^{\mathsf{cpa}} \equiv \sigma^E \; \bullet\!\!\overset{1}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet,$$

and thus

$$\begin{aligned}\Delta^{\mathbf{D}}(\mathsf{enc}^A\mathsf{dec}^B[\leftarrow\!\!\bullet, \bullet\!\!\overset{n}{-\!\diamond\!\!\twoheadrightarrow}], \sigma^E \; \bullet\!\!\overset{n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet) &\leq n \cdot \Delta^{\mathbf{D}\mathbf{C}''}(\mathsf{enc}^A\mathsf{dec}^B[\leftarrow\!\!\bullet, \bullet\!\!\overset{1}{-\!\diamond\!\!\twoheadrightarrow}], \sigma^E \; \bullet\!\!\overset{1}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet) \\ &= n \cdot \Delta^{\mathbf{D}\mathbf{C}''}(\mathbf{C}'\mathbf{G}_0^{\mathsf{cpa}}, \mathbf{C}'\mathbf{G}_1^{\mathsf{cpa}}) \\ &= n \cdot \Delta^{\mathbf{D}\mathbf{C}}(\mathbf{G}_0^{\mathsf{cpa}}, \mathbf{G}_1^{\mathsf{cpa}}),\end{aligned}$$

where $\mathbf{C} := \mathbf{C}''\mathbf{C}'$ and the first inequality follows from a standard hybrid argument for a reduction system $\mathbf{C}''$ (cf. Lemma 7 in Appendix A). □

3.3 Constructing a Confidential from an Authenticated and an Insecure Channel

To prove that the protocol $(\mathsf{enc}, \mathsf{dec})$ achieves construction (2), we need to exhibit a simulator σ such that the assumed resource $[\leftarrow\!\!\bullet, -\twoheadrightarrow]$ with the protocol converters is indistinguishable from the constructed resource $-\!\diamond\!\!\twoheadrightarrow\!\!\bullet$ with the simulator. Theorem 2 implies that $(\mathsf{enc}, \mathsf{dec})$ realizes (2) if the underlying PKE scheme is cca-secure.

The confidential channel $-\!\diamond\!\!\twoheadrightarrow\!\!\bullet$ is the best channel one can construct from the two assumed channels. As the E-interface has the same capabilities as the A-interface at both the authenticated (from B to A) and the insecure channels, it will necessarily also be possible to inject messages to the receiver via the E-interface by simply applying the sender's protocol converter.

Theorem 2. *There exists a simulator σ and for any $n \in \mathbb{N}$ there exists a (efficient) reduction $\mathbf{C}$ such that for every $\mathbf{D}$,*

$$\Delta^{\mathbf{D}}(\mathsf{enc}^A\mathsf{dec}^B[\leftarrow\!\!\bullet, \overset{n}{-\twoheadrightarrow}], \sigma^E \; \overset{n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet) \leq n \cdot \Delta^{\mathbf{D}\mathbf{C}}(\mathbf{G}_0^{\mathsf{cca}}, \mathbf{G}_1^{\mathsf{cca}}).$$

Proof. First, consider the following simulator σ for interface E of $-\!\diamond\!\!\twoheadrightarrow\!\!\bullet$, which again has two outside sub-interfaces $\mathtt{out}.1$ and $\mathtt{out}.2$: Initially, it generates a key pair $(\mathsf{pk}, \mathsf{sk})$ and outputs $(1, \mathsf{pk})$ at $\mathtt{out}.1$. When it receives (i, l) at the inside interface $\mathtt{in}$, it generates an encryption $c \leftarrow E_{\mathsf{pk}}(\bar{m})$ of a randomly chosen message $\bar{m}$ of length l, outputs (i, c) at the $\mathtt{out}.2$, and records (c, i). When (inj, c') is input at $\mathtt{out}.2$, σ proceeds as follows: If (c', i') has been recorded for some i', it outputs (dlv, i') at $\mathtt{in}$. Otherwise, it computes $m' \leftarrow D_{\mathsf{sk}}(c')$ and, if $m' \neq \diamond$, outputs (inj, m') at $\mathtt{in}$.

Denote by $\overset{n,q}{-\twoheadrightarrow}$ the insecure channel that processes the first n inputs at interface A and the first q inputs at interface E (and similarly for $\overset{n,q}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet$). Consider now the problem of distinguishing the two systems

$$\mathbf{U} := \mathsf{enc}^A\mathsf{dec}^B[\leftarrow\!\!\bullet, \overset{1,n}{-\twoheadrightarrow}] \qquad \text{and} \qquad \mathbf{V} := \sigma^E \; \overset{1,n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet,$$

which are depicted in Figure 3. A distinguisher $\mathbf{D}$ connected to the real-world system $\mathbf{U}$ initially sees a public key at interface $E.1$. If $\mathbf{D}$ inputs a message m at interface A, an encryption of m (created by enc) is output at interface $E.2$. When $\mathbf{D}$ inputs a ciphertext c' at E, it sees a decryption of c' (by dec) at B. The ideal-world system $\mathbf{V}$ behaves differently: Initially, $\mathbf{D}$ also sees a public key at $E.1$. But when it inputs a message m at A, an encryption c of a randomly chosen message is output at interface $E.2$ (by simulator σ). When c is input at interface $E.2$, m is output at B (as σ issues a dlv-instruction to the channel). When $c' \neq c$ is input at $E.2$, a decryption of c' (injected by σ) is output at B.

The translation between the channel setting and the game setting is achieved by the following reduction system $\mathbf{C}'$: Initially, $\mathbf{C}'$ takes a value pk from the game (on the inside) and outputs

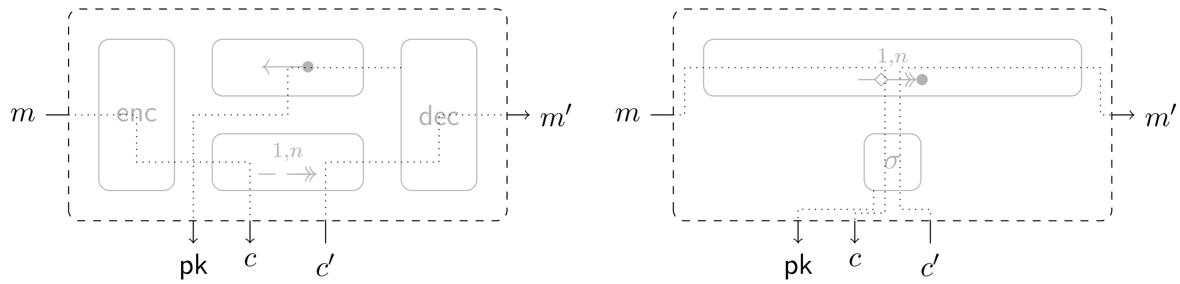

Figure 3: *The systems* $\mathbf{U}$ *and* $\mathbf{V}$ *with the "message flow" from the perspective of a distinguisher: Initially, a public-key* pk *is output at interface* E*. Inputting a message* m *at interface* A *causes a ciphertext* c *to be output at the* E*-interface. Note that* c *is the challenge in the* cca*-game. Inputting a ciphertext* c' *at interface* E *results in a message* m' *being output at* B*. This corresponds to the decryption oracle in the* cca*-game.*

it as $(1, \mathsf{pk})$ at the (outside) $E.1$-interface. When a message m is input at interface A of $\mathbf{C}'$, (chall, m) is output to the game. The resulting challenge c is output as $(1, c)$ at interface $E.2$. When (inj, c) is input at interface $E.2$, $\mathbf{C}'$ outputs m at interface B. When (inj, c') with $c' \neq c$ is input at interface $E.2$, $\mathbf{C}'$ passes (dec, c') to the game's decryption oracle and outputs the answer m' at interface B, provided $m' \neq \diamond$. We have

$$\mathbf{C}'\mathbf{G}_0^{\mathsf{cca}} \equiv \mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\bullet, -\overset{1,n}{\twoheadrightarrow}] \qquad \text{and} \qquad \mathbf{C}'\mathbf{G}_1^{\mathsf{cca}} \equiv \sigma^E \overset{1,n}{-\!\diamond\!\twoheadrightarrow\bullet},$$

and thus

$$\begin{aligned}\Delta^{\mathbf{D}}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\bullet, -\overset{n}{\twoheadrightarrow}], \sigma^E \overset{n}{-\!\diamond\!\twoheadrightarrow\bullet}) &\leq n \cdot \Delta^{\mathbf{DC}''}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\bullet, -\overset{1,n}{\twoheadrightarrow}], \sigma^E \overset{1,n}{-\!\diamond\!\twoheadrightarrow\bullet}) \\ &= n \cdot \Delta^{\mathbf{DC}''}(\mathbf{C}'\mathbf{G}_0^{\mathsf{cca}}, \mathbf{C}'\mathbf{G}_1^{\mathsf{cca}}) \\ &= n \cdot \Delta^{\mathbf{DC}}(\mathbf{G}_0^{\mathsf{cca}}, \mathbf{G}_1^{\mathsf{cca}}),\end{aligned}$$

where $\mathbf{C} := \mathbf{C}''\mathbf{C}'$ and the first inequality follows from a standard hybrid argument for a reduction system $\mathbf{C}''$ (cf. Lemma 8 in Appendix A). □

3.4 Replay-Protected Channels from CCA-Security

As pointed out in Section 4.2, cca-security is overly strict in that only the weaker rcca-security is necessary to achieve construction (2). In fact, using a cca-secure PKE scheme one can construct a replay-protected confidential channel $\overset{\mathsf{RP}}{-\!\diamond\!\twoheadrightarrow\bullet}$, which works as $-\!\diamond\!\twoheadrightarrow\bullet$ with the exception that for any index i', the query (dlv, i') is processed at most once (cf. Section 2.4). The protocol converters $(\mathsf{enc}', \mathsf{dec}')$ are built as $(\mathsf{enc}, \mathsf{dec})$ in Section 3.1, except that dec' processes every ciphertext received at `in`.2 only once. Similarly, the corresponding simulator σ' also processes every ciphertext received at `out`.2 only once.[19]

3.5 Applicability of the Constructed Channels

The plain use of PKE yields constructions (1) and (2), i.e., one obtains the resources $\bullet\!-\!\diamond\!\twoheadrightarrow\bullet$ and $-\!\diamond\!\twoheadrightarrow\bullet$. Both channels allow the adversary to reorder or replay the messages sent by A. In practice, where PKE is often used to encapsulate symmetric keys, it is important, however, that keys used in various protocols by different users are independent. Thus, it is more useful to obtain independent single-use channels

$$[\bullet\!\!\longrightarrow\!\!\bullet, \ldots, \bullet\!\!\longrightarrow\!\!\bullet] \quad \text{and} \quad [\longrightarrow\!\!\bullet, \ldots, \longrightarrow\!\!\bullet]$$

[19] Note that, in fact, an sd-rcca-secure PKE scheme suffices (cf. [9] for more details). In this case, dec' and σ' process only one ciphertext per equivalence class.

instead of $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$ and $-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$, respectively.

In the authenticated setting, given independent authenticated channels, protocol $(\mathsf{enc}, \mathsf{dec})$ (with only formal modifications) achieves the construction

$$[\bullet\!\!\longrightarrow, \ldots, \bullet\!\!\longrightarrow] \quad \xRightarrow{(\mathsf{enc},\mathsf{dec})} \quad [\bullet\!\!\longrightarrow\!\!\!\!\rightarrowtail\!\!\bullet, \ldots, \bullet\!\!\longrightarrow\!\!\!\!\rightarrowtail\!\!\bullet].$$

In the unauthenticated setting, however, the analogous construction

$$[\longrightarrow, \ldots, \longrightarrow] \quad \xRightarrow{(\mathsf{enc},\mathsf{dec})} \quad [\longrightarrow\!\!\!\!\rightarrowtail\!\!\bullet, \ldots, \longrightarrow\!\!\!\!\rightarrowtail\!\!\bullet]$$

is not achieved by $(\mathsf{enc}, \mathsf{dec})$ since, due to the absence of authenticity, the adversary can freely take a ciphertext it observes on any of the insecure channels $\longrightarrow$ and insert it into another one. Thus, the ideal resource cannot consist of independent channels. This issue can be taken care of by (explicitly) introducing session identifiers (SIDs). A systematic treatment of SIDs and handling multiple sessions and senders can be found in [31].

4 Constructive Semantics of Game-Based Security Notions

We analyze several game-based security notions from a constructive viewpoint. We complete the analysis of cpa-security from Section 3.2 by showing that it is also necessary to achieve construction (1). Moreover, we explain why the notion of cca is unnecessarily strict for construction (2) and that the construction in fact only requires the weaker notion of rcca introduced in [9].

Then, we follow up on work by Bellare et al. [6], who compared several variants of defining cca-security, and show that only the stricter notions they consider are sufficient for construction (2). We also provide constructive semantics for non-adaptive chosen-ciphertext security ($\mathsf{ind\text{-}cca1}$) and non-malleability ($\mathsf{nm\text{-}cpa}$).

4.1 Chosen-Plaintext Security is Necessary for Construction (1)

We prove in Section 3.2 that indistinguishability under chosen-plaintext attacks, cpa-security, suffices to construct a secure channel from two authenticated channels. Here, we show that it is also necessary. That is, if protocol $(\mathsf{enc}, \mathsf{dec})$, based on a PKE scheme Π as shown in Section 3.1, achieves the construction, then Π must be cpa-secure.

In the following, let

$$\mathbf{U} := \mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow] \qquad \text{and} \qquad \mathbf{V} := \sigma^E \bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet,$$

where σ is an *arbitrary* simulator.

Theorem 3. *There exist (efficient) reductions $\mathbf{C}_0$ and $\mathbf{C}_1$ such that for all adversaries $\mathbf{A}$,*

$$\Delta^{\mathbf{A}}(\mathbf{G}_0^{\mathsf{cpa}}, \mathbf{G}_1^{\mathsf{cpa}}) \leq \Delta^{\mathbf{AC}_0}(\mathbf{U}, \mathbf{V}) + \Delta^{\mathbf{AC}_1}(\mathbf{U}, \mathbf{V}).$$

Proof. Consider the following reduction systems $\mathbf{C}_0$ and $\mathbf{C}_1$, both connecting to an $\{A, B, E\}$-resource on the inside and providing a single interface on the outside (for the adversary): Initially, both obtain $(1, \mathsf{pk})$ at the inside $E.1$-interface and output pk at the outside interface. When (chall, m) is received on the outside, $\mathbf{C}_0$ outputs m at the inside A-interface and $\mathbf{C}_1$ a randomly chosen message $\bar{m}$ of length $|m|$. Subsequently, $(1, c)$ is received at the inside $E.2$-interface, and c is output (as the challenge) on the outside by both systems. We have

$$\mathbf{C}_0\mathbf{U} \equiv \mathbf{G}_0^{\mathsf{cpa}} \qquad \text{and} \qquad \mathbf{C}_1\mathbf{U} \equiv \mathbf{G}_1^{\mathsf{cpa}} \qquad \text{and} \qquad \mathbf{C}_0\mathbf{V} \equiv \mathbf{C}_1\mathbf{V},$$

where the last equivalence follows from the fact that, in $\mathbf{V}$, the input from $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$ to σ is the same in both systems (the length of the message input at the A-interface of $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$), and therefore they behave identically. Hence,

$$\begin{aligned}
\Delta^{\mathbf{A}}(\mathbf{G}_0^{\mathsf{cpa}}, \mathbf{G}_1^{\mathsf{cpa}}) &= \Delta^{\mathbf{A}}(\mathbf{C}_0\mathbf{U}, \mathbf{C}_1\mathbf{U}) \\
&\leq \Delta^{\mathbf{A}}(\mathbf{C}_0\mathbf{U}, \mathbf{C}_0\mathbf{V}) + \Delta^{\mathbf{A}}(\mathbf{C}_0\mathbf{V}, \mathbf{C}_1\mathbf{V}) + \Delta^{\mathbf{A}}(\mathbf{C}_1\mathbf{V}, \mathbf{C}_1\mathbf{U}) \\
&= \Delta^{\mathbf{AC}_0}(\mathbf{U}, \mathbf{V}) + \Delta^{\mathbf{AC}_1}(\mathbf{U}, \mathbf{V}).
\end{aligned}$$

□

4.2 Relaxed Chosen-Ciphertext Security is Necessary for Construction (2)

Indistinguishability under chosen-ciphertext attacks, cca-security, suffices to construct a confidential channel from an authenticated and an insecure one (cf. Section 3.3). It is, however, unnecessarily strict, as can be seen from the following example, adapted from [9]: Let Π be a PKE scheme and assume it is cca-secure. Consider a modified scheme Π' that works exactly as Π, except that a 0-bit is appended to every encryption, which is ignored during decryption. It is easily seen that Π' is not cca-secure, since the adversary can obtain a decryption of the challenge ciphertext by flipping its last bit and submitting the result to the decryption oracle. PKE scheme Π' can, however, still be used to achieve construction (2) using a simulator that also issues the dlv-instruction to $-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$ when flipping the last bit of a ciphertext received at the outside interface results in a recorded ciphertext (but otherwise works like σ from Theorem 2).

Canetti et al. [9] introduced the notion of *replayable chosen ciphertext* security, rcca, which is more permissive in that it allows the adversary to transform a ciphertext into one that decrypts to the same message. Below we show that if protocol $(\mathsf{enc}, \mathsf{dec})$, based on a PKE scheme Π as shown in Section 3.1, achieves construction (2), then Π must be rcca-secure. Note that rcca is also sufficient for the construction if the message space of Π is sufficiently large (cf. Appendix B).

In the following, let

$$\mathbf{U} := \mathsf{enc}^A \mathsf{dec}^B [\longleftarrow\!\!\bullet, -\!\!\twoheadrightarrow] \qquad \text{and} \qquad \mathbf{V} := \sigma^E -\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet,$$

where σ is an *arbitrary* simulator.

Theorem 4. *There exist (efficient) reductions* $\mathbf{C}_0$ *and* $\mathbf{C}_1$ *such that for all adversaries* $\mathbf{A}$*,*

$$\Delta^{\mathbf{A}}(\mathbf{G}_0^{\mathsf{rcca}}, \mathbf{G}_1^{\mathsf{rcca}}) \leq \Delta^{\mathbf{A}\mathbf{C}_0}(\mathbf{U}, \mathbf{V}) + \Delta^{\mathbf{A}\mathbf{C}_1}(\mathbf{U}, \mathbf{V}).$$

Proof. Consider the following reductions $\mathbf{C}_0$ and $\mathbf{C}_1$. Again, both connect to an $\{A, B, E\}$-resource on the inside and provide a single interface on the outside: Initially, both obtain $(1, \mathsf{pk})$ at the inside $E.1$-interface and output pk at the outside interface. When (chall, m) is received on the outside, *both* systems choose a random message $\bar{m}$. $\mathbf{C}_0$ outputs m at the inside A-interface and $\mathbf{C}_1$ outputs $\bar{m}$. Subsequently, $(1, c)$ is received at the inside E-interface, and c is output on the outside by both systems. When a decryption query (dec, c') is received on the outside, both systems output (inj, c') at the inside $E.2$-interface. A subsequently received message m' at B is output on the outside by both systems (as answer to the decryption query) unless $m' \in \{m, \bar{m}\}$, in which case test is returned. We have

$$\mathbf{C}_0\mathbf{U} \equiv \mathbf{G}_0^{\mathsf{rcca}} \qquad \text{and} \qquad \mathbf{C}_1\mathbf{U} \equiv \mathbf{G}_1^{\mathsf{rcca}} \qquad \text{and} \qquad \mathbf{C}_0\mathbf{V} \equiv \mathbf{C}_1\mathbf{V},$$

where the last equivalence follows from the fact that, in $\mathbf{V}$, the input from $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$ to σ is the same in both systems (the length of the message input at the A-interface of $\bullet\!\!-\!\!\diamond\!\!\twoheadrightarrow\!\!\bullet$) and that decryption queries causing m or $\bar{m}$ to be output at the B-interface are answered by test. Hence,

$$\begin{aligned}
\Delta^{\mathbf{A}}(\mathbf{G}_0^{\mathsf{rcca}}, \mathbf{G}_1^{\mathsf{rcca}}) &= \Delta^{\mathbf{A}}(\mathbf{C}_0\mathbf{U}, \mathbf{C}_1\mathbf{U}) \\
&\leq \Delta^{\mathbf{A}}(\mathbf{C}_0\mathbf{U}, \mathbf{C}_0\mathbf{V}) + \Delta^{\mathbf{A}}(\mathbf{C}_0\mathbf{V}, \mathbf{C}_1\mathbf{V}) + \Delta^{\mathbf{A}}(\mathbf{C}_1\mathbf{V}, \mathbf{C}_1\mathbf{U}) \\
&= \Delta^{\mathbf{A}\mathbf{C}_0}(\mathbf{U}, \mathbf{V}) + \Delta^{\mathbf{A}\mathbf{C}_1}(\mathbf{U}, \mathbf{V}).
\end{aligned}$$

□

4.3 Variants of Chosen-Ciphertext Security

Bellare et al. [6] analyze several ways of enforcing the condition that the adversary must not query the challenge ciphertext c to the decryption oracle. They consider modifications along two axes: First, the condition can be enforced during the entire game (b for *both* phases) or only in the second phase (s for *second* phase), i.e., after the c has been given to the adversary. Second, one can either exclude adversaries with a non-zero probability of violating the condition from consideration (e for *exclusion*) or penalize an adversary (by declaring the game lost) whenever he asks the challenge c (p for *penalty*). The combination of these choices yields four *non-equivalent* notions ind-cca-sp, ind-cca-se, ind-cca-bp, ind-cca-be. The s-notions are equivalent to each other and to our formulation of cca-security (cf. Section 2.5). The e-notions are strictly weaker and do in fact not even imply cca1-security [6]. Since cca1-security is weaker than rcca-security and rcca is needed for construction (2), they are not sufficient for (2).

4.4 Non-Malleability

Informally, a non-malleable PKE scheme is such that the adversary cannot transform a ciphertext into one that decrypts to a related message. We consider the notion of non-malleability under chosen-plaintext attacks, nm-cpa, and show that from a PKE scheme with this property we can build a protocol $(\mathsf{enc}'', \mathsf{dec}'')$ that achieves the construction

$$[\longleftarrow\!\!\bullet, - \twoheadrightarrow\!\!|] \quad \overset{(\mathsf{enc}'',\mathsf{dec}'')}{\Longmapsto} \quad -\!\!\diamond\!\!\twoheadrightarrow\!\!|\!\!\bullet, \tag{3}$$

where $- \twoheadrightarrow\!\!|$ works like $- \twoheadrightarrow$ but halts when halt is input at B and where the batch channel $-\!\!\diamond\!\!\twoheadrightarrow\!\!|\!\!\bullet$ is defined as follows: It internally keeps an initially empty list $\mathcal{L}$ of messages. When the i^{th} message m is input at interface A, it is recorded as (i, m) and $(i, |m|)$ is output at interface E. When (dlv, i') is input at interface E and if (i', m') has been recorded, m' is appended to $\mathcal{L}$. When (inj, m') is input at interface E, m' is appended to $\mathcal{L}$. When dlv-all is input at B, all messages in $\mathcal{L}$ are output at B, and the channel halts.

The protocol converters $(\mathsf{enc}'', \mathsf{dec}'')$ are built as $(\mathsf{enc}, \mathsf{dec})$ in Section 3.1, except that dec'' only outputs the messages it received once dlv-all is input at the outside interface, at which time it also outputs halt at its inside interface and halts. Theorem 5 below implies that $(\mathsf{enc}'', \mathsf{dec}'')$ achieves construction (3) if Π is nm-cpa-secure.

Theorem 5. *There exists a simulator σ and for any $n \in \mathbb{N}$ there exists a (efficient) reduction $\mathbf{C}$ such that for every $\mathbf{D}$,*

$$\Delta^{\mathbf{D}}(\mathsf{enc}''^A \mathsf{dec}''^B [\longleftarrow\!\!\bullet, - \overset{n}{\twoheadrightarrow}\!\!|], \sigma^E \; -\!\!\overset{n}{\diamond}\!\!\twoheadrightarrow\!\!|\!\!\bullet) \le n \cdot \Delta^{\mathbf{DC}}(\mathbf{G}_0^{\mathsf{nm}}, \mathbf{G}_1^{\mathsf{nm}}).$$

Proof. Let σ be the simulator from Theorem 2. Consider the two systems

$$\mathsf{enc}''^A \mathsf{dec}''^B [\longleftarrow\!\!\bullet, - \overset{1}{\twoheadrightarrow}\!\!|] \qquad \text{and} \qquad \sigma^E \; -\!\!\overset{1}{\diamond}\!\!\twoheadrightarrow\!\!|\!\!\bullet\,.$$

Distinguishing $\mathbf{G}_0^{\mathsf{nm}}$ from $\mathbf{G}_1^{\mathsf{nm}}$ can be reduced to distinguishing these two systems via the following reduction system $\mathbf{C}'$. Initially, $\mathbf{C}'$ takes pk from the game and outputs it at the E-interface. When a message m is input at interface A of $\mathbf{C}'$, it is forwarded as (chall, m) to the game. The challenge c from the game is output as $(1, c)$ at interface E. When (inj, c') is input at interface E, $\mathbf{C}'$ records c'. When dlv-all is input at interface B, $\mathbf{C}'$ passes the vector of all recorded ciphertexts to the game. In the subsequently received vector of plaintexts from the game, it replaces all test-messages by m. Then, it outputs all the plaintexts at B and halts. We have

$$\mathbf{C}'\mathbf{G}_0^{\mathsf{nm}} \equiv \mathsf{enc}''^A \mathsf{dec}''^B [\longleftarrow\!\!\bullet, - \overset{1}{\twoheadrightarrow}\!\!|] \qquad \text{and} \qquad \mathbf{C}'\mathbf{G}_1^{\mathsf{nm}} \equiv \sigma^E \; -\!\!\overset{1}{\diamond}\!\!\twoheadrightarrow\!\!|\!\!\bullet,$$

and thus

$$\begin{aligned}
\Delta^{\mathbf{D}}(\mathsf{enc}''^A \mathsf{dec}''^B [\longleftarrow\!\!\bullet, - \overset{n}{\twoheadrightarrow}\!\!|], \sigma^E \; -\!\!\overset{n}{\diamond}\!\!\twoheadrightarrow\!\!|\!\!\bullet) &\le n \cdot \Delta^{\mathbf{DC}''}(\mathsf{enc}''^A \mathsf{dec}''^B [\longleftarrow\!\!\bullet, - \overset{1}{\twoheadrightarrow}\!\!|], \sigma^E \; -\!\!\overset{1}{\diamond}\!\!\twoheadrightarrow\!\!|\!\!\bullet) \\
&= n \cdot \Delta^{\mathbf{DC}''}(\mathbf{C}'\mathbf{G}_0^{\mathsf{nm}}, \mathbf{C}'\mathbf{G}_1^{\mathsf{nm}}) \\
&= n \cdot \Delta^{\mathbf{DC}}(\mathbf{G}_0^{\mathsf{nm}}, \mathbf{G}_1^{\mathsf{nm}}),
\end{aligned}$$

where $\mathbf{C} := \mathbf{C}''\mathbf{C}'$ and the first inequality follows from a standard hybrid argument for a reduction system $\mathbf{C}''$ (the proof is similar to that of Lemma 8). □

The assumed channel $-\twoheadrightarrow$ could itself be constructed in a setting where A and B have synchronized clocks and B buffers all messages until an agreed point in time, when A also stops sending. By the composition theorem, the channel that is constructed in this manner can then serve as the assumed channel in construction 3 to construct the channel $-\!\diamond\!\!\twoheadrightarrow\!\bullet$ using PKE. This channel may then for instance be useful for running a protocol implementing a blind auction.

4.5 Non-Adaptive Chosen-Ciphertext Security

Security against lunchtime attacks, or $\mathsf{ind\text{-}cca1}$-security, is defined via a corresponding game $\mathbf{G}^{\mathsf{cca1}}$, which works as $\mathbf{G}^{\mathsf{cca}}$ except that no decryption queries are answered once the adversary has been given the challenge ciphertext. The most natural way to translate this into a constructive statement is to consider the construction of a (type of) confidential channel $\circ\!\!-\!\diamond\!\!\twoheadrightarrow\!\bullet$ where the adversary can inject messages at interface E only as long as no message has been input at A from an insecure channel $\circ\!\!-\ \twoheadrightarrow$ with the same property.

Theorem 6, whose proof we omit the proof as it is very similar to the proof of Theorem 2, implies that protocol $(\mathsf{enc}, \mathsf{dec})$ built from a $\mathsf{cca1}$-secure PKE scheme Π as in Section 3.1 achieves

$$[\leftarrow\!\!\bullet, \circ\!\!-\ \twoheadrightarrow] \quad \stackrel{(\mathsf{enc}'',\mathsf{dec}'')}{\Longmapsto} \quad \circ\!\!-\!\diamond\!\!\twoheadrightarrow\!\bullet. \tag{4}$$

Theorem 6. *There exists a simulator σ and for any $n \in \mathbb{N}$ there exists a (efficient) reduction $\mathbf{C}$ such that for every $\mathbf{D}$,*

$$\Delta^{\mathbf{D}}(\mathsf{enc}^A \mathsf{dec}^B [\leftarrow\!\!\bullet, \circ\!\!-\stackrel{n}{\ }\!\twoheadrightarrow], \sigma^E\, \circ\!\!-\!\stackrel{n}{\diamond}\!\!\twoheadrightarrow\!\bullet) \leq n \cdot \Delta^{\mathbf{DC}}(\mathbf{G}_0^{\mathsf{cca1}}, \mathbf{G}_1^{\mathsf{cca1}}).$$

Although this construction seems somewhat artificial, as with construction (3), it can be used in any setting where the assumed channel is an appropriate modeling of an available physical channel (or can itself be constructed from such a channel).

5 Conclusions

We described the basic application of PKE as the construction of a confidential channel from non-confidential ones. This construction step can then be used within any larger protocol; the composability guarantee is essential for the modular design of complex protocols, thus taming the complexity of security-protocol design. To be ultimately applicable to full-fledged real-world protocols, other relevant cryptographic primitives also need to be modeled in the same way. While for symmetric encryption and MACs this was explained in [29, 30], and for commitments in [27], treating digital signatures and other cryptographic schemes and security mechanisms (sequence numbers, session identifiers, etc.) in constructive cryptography is left for follow-up work (e.g., [31]).

Acknowledgments. The work was supported by the Swiss National Science Foundation (SNF), project no. 200020-132794.

References

[1] Michael Backes, Birgit Pfitzmann, and Michael Waidner. The Reactive Simulatability (RSIM) Framework for Asynchronous Systems. *Information and Computation*, 205(12): 1685–1720, December 2007.

[2] Donald Beaver. Foundations of secure interactive computing. In Joan Feigenbaum, editor, *CRYPTO*, volume 576 of *LNCS*, pages 377–391, Heidelberg, 1991. Springer.

[3] Mihir Bellare and Amit Sahai. Non-malleable Encryption: Equivalence between Two Notions, and an Indistinguishability-Based Characterization. In Michael Wiener, editor, *CRYPTO 1999*, volume 1666 of *LNCS*, pages 519–536, Heidelberg, 1999. Springer.

[4] Mihir Bellare, Anand Desai, E. Jokipii, and Phillip Rogaway. A Concrete Security Treatment of Symmetric Encryption. In *38th FOCS*, pages 394–403, 1997.

[5] Mihir Bellare, Anand Desai, David Pointcheval, and Phillip Rogaway. Relations Among Notions of Security for Public-Key Encryption Schemes. In Hugo Krawczyk, editor, *CRYPTO 1998*, volume 1462 of *LNCS*, pages 26–45, Heidelberg, 1998. Springer.

[6] Mihir Bellare, Dennis Hofheinz, and Eike Kiltz. Subtleties in the Definition of IND-CCA: When and How Should Challenge-Decryption be Disallowed? Cryptology ePrint Archive 2009/418, 2009.

[7] Ran Canetti. Universally Composable Security: A New Paradigm for Cryptographic Protocols. Cryptology ePrint Archive, Report 2000/067, 2000.

[8] Ran Canetti and Hugo Krawczyk. Universally Composable Notions of Key Exchange and Secure Channels. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 337–351, Heidelberg, 2002. Springer.

[9] Ran Canetti, Hugo Krawczyk, and Jesper Buus Nielsen. Relaxing Chosen-Ciphertext Security. In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 565–582, Heidelberg, 2003. Springer.

[10] Ronald Cramer and Victor Shoup. A Practical Public Key Cryptosystem Provably Secure Against Adaptive Chosen Ciphertext Attack. In Hugo Krawczyk, editor, *CRYPTO 1998*, volume 1462 of *LNCS*, pages 13–25, Heidelberg, 1998. Springer.

[11] Ronald Cramer and Victor Shoup. Design and Analysis of Practical Public-Key Encryption Schemes Secure against Adaptive Chosen Ciphertext Attack. *SIAM Journal on Computing*, 33:167–226, 2001.

[12] Whitfield Diffie and Martin E. Hellman. New Directions in Cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976.

[13] Danny Dolev, Cynthia Dwork, and Moni Naor. Non-Malleable Cryptography (Extended Abstract). In *23rd ACM STOC*, pages 542–552, 1991.

[14] Eiichiro Fujisaki, Tatsuaki Okamoto, David Pointcheval, and Jacques Stern. RSA-OAEP Is Secure under the RSA Assumption. In Joe Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 260–274, Heidelberg, 2001. Springer.

[15] Oded Goldreich. Foundations of Cryptography. Class Notes, Spring 1989. Technion University.

[16] Oded Goldreich. A Uniform-Complexity Treatment of Encryption and Zero-Knowledge. *Journal of Cryptology*, 6(1):21–53, 1993.

[17] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to Play any Mental Game or A Completeness Theorem for Protocols with Honest Majority. In *19th ACM STOC*, pages 218–229, 1987.

[18] Shafi Goldwasser and Leonid A. Levin. Fair computation of general functions in presence of immoral majority. In *CRYPTO*, pages 77–93, 1990.

[19] Shafi Goldwasser and Silvio Micali. Probabilistic Encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[20] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The Knowledge Complexity of Interactive Proof-Systems (Extended Abstract). In *17th ACM STOC*, pages 291–304, 1985.

[21] Dennis Hofheinz and Eike Kiltz. Practical Chosen Ciphertext Secure Encryption from Factoring. In Antoine Joux, editor, *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 313–332, Heidelberg, 2009. Springer.

[22] Dennis Hofheinz and Victor Shoup. GNUC: A New Universal Composability Framework. Cryptology ePrint Archive, Report 2011/303, 2011.

[23] Eike Kiltz, Krzysztof Pietrzak, Martijn Stam, and Moti Yung. A New Randomness Extraction Paradigm for Hybrid Encryption. In Antoine Joux, editor, *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 590–609, Heidelberg, 2009. Springer.

[24] Markulf Kohlweiss, Ueli Maurer, Cristina Onete, Björn Tackmann, and Daniele Venturi. Anonymity-Preserving Public-Key Encryption: A Constructive Approach. In E. De Cristofaro and M. Wright, editors, *PETS 2013*, volume 7981 of *LNCS*, pages 19–39, Heidelberg, 2013. Springer.

[25] Ueli Maurer. Indistinguishability of Random Systems. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 110–132, Heidelberg, 2002. Springer.

[26] Ueli Maurer. Constructive Cryptography—A New Paradigm for Security Definitions and Proofs. In S. Moedersheim and C. Palamidessi, editors, *TOSCA 2011*, volume 6993 of *LNCS*, pages 33–56, Heidelberg, April 2011. Springer.

[27] Ueli Maurer and Renato Renner. Abstract Cryptography. In Bernard Chazelle, editor, *The Second Symposium in Innovations in Computer Science, ICS 2011*, pages 1–21. Tsinghua University Press, January 2011.

[28] Ueli Maurer and Pierre E. Schmid. A Calculus for Security Bootstrapping in Distributed Systems. *Journal of Computer Security*, 4(1):55–80, 1996.

[29] Ueli Maurer and Björn Tackmann. On the Soundness of Authenticate-then-Encrypt: Formalizing the Malleability of Symmetric Encryption. In *ACM CCS*, pages 505–515, 2010.

[30] Ueli Maurer, Andreas Rüedlinger, and Björn Tackmann. Confidentiality and Integrity: A Constructive Perspective. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 209–229, Heidelberg, 2012. Springer.

[31] Ueli Maurer, Björn Tackmann, and Sandro Coretti. Key Exchange with Unilateral Authentication: Composable Security Definition and Modular Protocol Design. Cryptology ePrint Archive, Report 2013/555, 2013.

[32] Silvio Micali and Phillip Rogaway. Secure computation (abstract). In Joan Feigenbaum, editor, *CRYPTO*, volume 576 of *LNCS*, pages 392–404, Heidelberg, 1991. Springer.

[33] Silvio Micali, Charles Rackoff, and Bob Sloan. The Notion of Security for Probabilistic Cryptosystems. *SIAM Journal on Computing*, 17(2):412–426, 1988.

[34] Moni Naor and Moti Yung. Public-key Cryptosystems Provably Secure Against Chosen Ciphertext Attacks. In *22nd ACM STOC*, pages 427–437, 1990.

[35] Chris Peikert and Brent Waters. Lossy Trapdoor Functions and Their Applications. In *40th ACM STOC*, pages 187–196, 2008.

[36] Birgit Pfitzmann and Michael Waidner. A Model for Asynchronous Reactive Systems and its Application to Secure Message Transmission. In *IEEE Symposium on Security and Privacy*, pages 184–200, 2001.

[37] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[38] Alon Rosen and Gil Segev. Chosen-Ciphertext Security via Correlated Products. In Omer Reingold, editor, *TCC 2009*, volume 5444 of *LNCS*, pages 419–436, Heidelberg, 2009. Springer.

[39] Andrew Chi-Chih Yao. Theory and Applications of Trapdoor Functions (Extended Abstract). In *23rd FOCS*, pages 80–91, 1982.

[40] Yuliang Zheng and Jennifer Seberry. Practical Approaches to Attaining Security Against Adaptively Chosen Ciphertext Attacks (Extended Abstract). In Ernest F. Brickell, editor, *CRYPTO '92*, volume 740 of *LNCS*, pages 292–304, Heidelberg, 1992. Springer.

A Security for Many Messages

Let $(\mathsf{enc}, \mathsf{dec})$ be a protocol constructed from a PKE scheme as shown in Section 3.1.

Lemma 7. *For every $n \in \mathbb{N}$ there exists a (efficient) reduction $\mathbf{C}''$ such that*

$$\Delta^{\mathbf{D}}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \bullet\!\!\overset{n}{-\!\diamond\!\!\twoheadrightarrow}], \sigma^E\ \bullet\!\!\overset{n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet) \leq n \cdot \Delta^{\mathbf{D}\mathbf{C}''}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \bullet\!\!\overset{1}{-\!\diamond\!\!\twoheadrightarrow}], \sigma^E\ \bullet\!\!\overset{1}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet),$$

where σ is the simulator from Theorem 1.

Proof. Omitted (as similar to the proof of Lemma 8). $\square$

Recall that $\overset{n,q}{-\,-\!\!\twoheadrightarrow}$ denotes the insecure channel that processes the first n inputs at interface A and the first q inputs at interface E (and similarly for $\overset{n,q}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet$).

Lemma 8. *For every $n \in \mathbb{N}$ there exists a (efficient) reduction $\mathbf{C}''$ such that*

$$\Delta^{\mathbf{D}}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \overset{n}{-\,-\!\!\twoheadrightarrow}], \sigma^E\ \overset{n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet) \leq n \cdot \Delta^{\mathbf{D}\mathbf{C}''}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \overset{1,n}{-\,-\!\!\twoheadrightarrow}], \sigma^E\ \overset{1,n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet),$$

where σ is the simulator from Theorem 2.

Proof. Let $\mathbf{D}$ be an arbitrary distinguisher. For $i = 1, \ldots, n$, consider the following reduction system $\mathbf{C}''_i$ (which processes at most n inputs at the outside A and E-interfaces): Initially, $\mathbf{C}''_i$ forwards a public key pk from the inside $E.1$-interface to the outside $E.1$-interface. When the j^{th} message m is input at the outside A-interface, if $j < i$, $\mathbf{C}''_i$ randomly chooses a message $\bar{m}$ of length $|m|$ and computes $c \leftarrow E_{\mathsf{pk}}(\bar{m})$, if $j = i$, it outputs m at the inside A-interface and obtains c at the inside $E.2$-interface, and if $j > i$ it computes $c \leftarrow E_{\mathsf{pk}}(m)$. In all cases, it outputs (j, c) at the outside $E.2$-interface and records (c, m). When (inj, c') is input at the outside $E.2$-interface, if (c', m') has been recorded for some m', m' is output at the outside B-interface, and otherwise (inj, c') is output at the inside $E.2$-interface and the subsequently received message m' at the inside B-interface is output at the outside B-interface. Note that

$$\mathbf{C}''_1\left(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \overset{1,n}{-\,-\!\!\twoheadrightarrow}]\right) \equiv \mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \overset{n}{-\,-\!\!\twoheadrightarrow}]$$

and

$$\mathbf{C}''_n\left(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \overset{1,n}{-\,-\!\!\twoheadrightarrow}]\right) \equiv \sigma^E\ \overset{n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet\ .$$

Moreover, for $i = 1, \ldots, n-1$, we have

$$\mathbf{C}''_{i-1}\left(\sigma^E\ \overset{1,n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet\right) \equiv \mathbf{C}''_i\left(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \overset{1,n}{-\,-\!\!\twoheadrightarrow}]\right).$$

Thus for the reduction $\mathbf{C}''$ that chooses i uniformly at random from $\{1, \ldots, n\}$ and then implements $\mathbf{C}''_i$,

$$\Delta^{\mathbf{D}\mathbf{C}''}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \overset{1,n}{-\,-\!\!\twoheadrightarrow}], \sigma^E\ \overset{1,n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet) = \frac{1}{n}\Delta^{\mathbf{D}}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, \overset{n}{-\,-\!\!\twoheadrightarrow}], \sigma^E\ \overset{n}{-\!\diamond\!\!\twoheadrightarrow}\!\!\bullet).$$

$\square$

B RCCA and the Equivalence to Transformation (2)

To settle the question of equivalence between transformation (2) and rcca-security, it remains to see whether rcca-security suffices to achieve (2). It turns out that this is the case if the message space $\mathcal{M}$ of the underlying PKE is large. For simplicity, we assume that all messages in $\mathcal{M}$ have equal length.

Theorem 9. *There exist a simulator σ and for any $n \in \mathbb{N}$ there exists a (efficient) reduction $\mathbf{C}$ such that for every $\mathbf{D}$,*

$$\Delta^{\mathbf{D}}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, -\overset{n}{\twoheadrightarrow}], \sigma^E \overset{n}{-\!\diamond\!\!\twoheadrightarrow\!\!\bullet}) \leq n \cdot \Delta^{\mathbf{DC}}(\mathbf{G}_0^{\mathsf{rcca}}, \mathbf{G}_1^{\mathsf{rcca}}) + \frac{n^2}{|\mathcal{M}|}.$$

Proof (sketch). We show

$$\Delta^{\mathbf{D}}(\mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, -\overset{1,n}{\twoheadrightarrow}], \sigma^E \overset{1,n}{-\!\diamond\!\!\twoheadrightarrow\!\!\bullet}) \leq \Delta^{\mathbf{DC}}(\mathbf{G}_0^{\mathsf{rcca}}, \mathbf{G}_1^{\mathsf{rcca}}) + \frac{n}{|\mathcal{M}|}.$$

The proof can be generalized using a hybrid argument.

Consider the following simulator σ (with two sub-interfaces at E): Initially, σ generates a key pair $(\mathsf{pk}, \mathsf{sk})$ and outputs pk at the outside interface. When it receives (i, l) at the inside interface, it generates an encryption $c \leftarrow E_{\mathsf{pk}}(\bar{m})$ of a randomly chosen message $\bar{m}$ (of length l), outputs (i, c) at the outside interface, and records (m, i). When (inj, c') is input at the outside interface, σ proceeds as follows: It computes $m' \leftarrow D_{\mathsf{sk}}(c')$. If (m', i') has been recorded for some i', it outputs (dlv, i') at its inside interface. Otherwise, if $m' \neq \diamond$, it outputs (inj, m') at the inside interface. Set

$$\mathbf{U} := \mathsf{enc}^A\mathsf{dec}^B[\longleftarrow\!\!\bullet, -\overset{1,n}{\twoheadrightarrow}] \qquad \text{and} \qquad \mathbf{V} := \sigma^E \overset{1,n}{-\!\diamond\!\!\twoheadrightarrow\!\!\bullet}$$

The translation between the game and the channel setting is done by the reduction $\mathbf{C}$. Initially, $\mathbf{C}$ takes pk from the game and outputs it at the E-interface. When a message m is input at interface A of $\mathbf{C}$, it is forwarded to the game. The resulting challenge c is output as $(1, c)$ at interface E. When (inj, c') with $c' \neq c$ is input at interface E, $\mathbf{C}$ passes c' to the game's decryption oracle. If the answer is test, it outputs m at interface B. If the answer is a message $m' \neq \diamond$, it is output at B. We have

$$\mathbf{C}\mathbf{G}_1^{\mathsf{rcca}} \equiv \mathbf{V}.$$

Moreover, for any $\mathbf{D}$, $\Delta^{\mathbf{D}}(\mathbf{C}\mathbf{G}_0^{\mathsf{rcca}}, \mathbf{U}) \leq n/|\mathcal{M}|$, since the two systems behave identically until $\mathbf{D}$ inputs (inj, c') for a c' that decrypts to $\bar{m}$ (chosen by $\mathbf{G}_0^{\mathsf{rcca}}$) at the E-interface. Therefore,

$$\begin{aligned}\Delta^{\mathbf{D}}(\mathbf{U}, \mathbf{V}) &\leq \Delta^{\mathbf{D}}(\mathbf{U}, \mathbf{C}\mathbf{G}_0^{\mathsf{rcca}}) + \Delta^{\mathbf{D}}(\mathbf{C}\mathbf{G}_0^{\mathsf{rcca}}, \mathbf{C}\mathbf{G}_1^{\mathsf{rcca}}) \\ &\leq \frac{n}{|\mathcal{M}|} + \Delta^{\mathbf{DC}}(\mathbf{G}_0^{\mathsf{rcca}}, \mathbf{G}_1^{\mathsf{rcca}}),\end{aligned}$$

and the claim follows. $\square$