Idealizing Identity-Based Encryption

Dennis Hofheinz[1], Christian Matt[2], and Ueli Maurer[2]

[1] Karlsruhe Institute of Technology (KIT), Germany
dennis.hofheinz@kit.edu
[2] Department of Computer Science, ETH Zurich, Switzerland
{mattc, maurer}@inf.ethz.ch

Abstract. We formalize the standard application of identity-based encryption (IBE), namely non-interactive secure communication, as realizing an ideal system which we call delivery controlled channel (DCC). This system allows users to be registered (by a central authority) for an identity and to send messages securely to other users only known by their identity. Quite surprisingly, we show that existing security definitions for IBE are not sufficient to realize DCC. In fact, it is impossible to do so in the standard model. We show, however, how to adjust any IBE scheme that satisfies the standard security definition IND-ID-CPA to achieve this goal in the random oracle model.

We also show that the impossibility result can be avoided in the standard model by considering a weaker ideal system that requires all users to be registered in an initial phase before any messages are sent. To achieve this, a weaker security notion, which we introduce and call IND-ID1-CPA, is actually sufficient. This justifies our new security definition and might open the door for more efficient schemes. We further investigate which ideal systems can be realized with schemes satisfying the standard notion and variants of selective security.

As a contribution of independent interest, we show how to model features of an ideal system that are potentially available to dishonest parties but not guaranteed, and which such features arise when using IBE.

Keywords: identity-based encryption, definitions, impossibility results, composability.

1 Introduction

1.1 Motivation

Identity-based encryption (IBE) is a generalization of public-key encryption where messages can be encrypted using a master public key and the *identity* of a user, which can be an arbitrary bit string, such as the user's e-mail address. Ciphertexts can be decrypted with a user secret key for the corresponding identity, where user secret keys are derived from a master secret key, which is generated together with the master public key.

The apparent standard application of IBE is non-interactive secure communication. More specifically, we assume a setting with many parties, and the goal is to enable each party to send any other party (known only by his/her identity) messages in a secure way. This secure communication should be non-interactive (or "one-shot") in the sense that the sending party should not be required to, e.g., look up a public key of the receiving party, or to communicate in any other way (beyond of course sending one message to the receiver). In fact, our requirements and expectations can be described as follows. We define a "resource" (or "ideal functionality" [10,1,15,5,19,13,12]) that provides the following basic services (via appropriate calls to the resource):

Registration. Each party is able to *register* his/her identity id. (Intuitively, an identity could be an email address or telephone number, that—presumably uniquely—identifies the registering party.)

Communication. Each party is able to *send* a message m to another party with identity id.

While an IBE scheme can be used in an obvious way to syntactically realize this functionality, the application is only secure if the IBE scheme satisfies a suitable security definition. Investigating the suitability of different security definitions for this task is the purpose of this paper.

The semantics of security definitions. We point out that security definitions for cryptographic primitives can serve two entirely different purposes, which are often not clearly distinguished. The first is to serve as a (technical) reference point, on one hand for devising schemes provably satisfying the definition based on a weak assumption, and on the other hand for building more sophisticated primitives from any scheme satisfying the definition. For instance, the one-way function definition serves this purpose excellently.

In this work, we are interested in a second purpose of security definitions, namely assuring the security of a certain type of application when a scheme satisfying the (technical) security definition is used. While definitions are usually devised with much intuition for what is needed in a certain application, a conventional technical security definition for a cryptographic primitive generally cannot directly imply the security of an associated application. Guaranteeing the security of an application can be seen as giving an application-semantics to a security definition.

1.2 Identity-Based Encryption and its Security

The concept of identity-based encryption has been conceived as early as 1984 [20]. A first candidate of an IBE scheme was presented in 1991 in [14], although without a detailed security model. In the 2000s, however, both a detailed security model [3] and a number of concrete IBE schemes (with security proofs under various assumptions) emerged, e.g., [3,7,21,9].

Both standard IBE security notions (IND-ID-CPA and IND-ID-CCA) are formalized as a security game. In this game, a hypothetical adversary $\mathcal{A}$ chooses an identity id^*, and messages m_0^* and m_1^*, and tries to distinguish an encryption of m_0^* from an encryption of m_1^* (both prepared for receiver identity id^*). Besides,

$\mathcal{A}$ may (adaptively) ask for arbitrary user secret keys for identities $id \neq id^*$. (In case of IND-ID-CCA security, $\mathcal{A}$ additionally gets access to a decryption oracle for arbitrary identities.) If no efficient $\mathcal{A}$ can successfully distinguish these ciphertexts, we consider the system secure.

At this point, we note that these game-based notions of security do allow for a form of adaptivity (in the sense that $\mathcal{A}$ may adaptively ask for user secret keys), but do not directly consider a concrete communication scenario.

1.3 Contributions

In this work, we investigate the goal of non-interactive communication, and in particular the use of IBE schemes to achieve that goal. Perhaps surprisingly, it turns out that the standard notions of IBE security do *not* imply non-interactive communication in the standard model. However, we prove that standard IBE security notions do imply non-interactive communication in the random oracle model and also weaker forms of non-interactive communication in the standard model. (Loosely speaking, standard IBE security notions achieve non-interactive communication in a setting in which registrations always occur *before* any attempt is made to send messages to the respective receiving party.) Furthermore, we introduce a new security notion that is weaker than the standard notion, but still implies a very natural weaker notion of non-interactive communication in the standard model.

To formalize our results, we use the constructive cryptography (CC) framework due to Maurer and Renner [13,12]. We stress, however, that our results do not depend on that particular formal model. Specifically, the reason that standard IBE security does not imply non-interactive communication is not tied to the specifics of CC. (We give a more detailed explanation of this reason below, and we will hint at the differences to a potential formulation in Canetti's universal composability framework [5] where appropriate.)

A more technical view. A little more technically, we model non-interactive communication as a "delivery controlled channels" resource DCC.[3] This resource has a number of interfaces, called A, $B_1, \ldots, B_n$, and C, to the involved users. Intuitively, interface C is used to register parties, A is used to send messages[4], and the interfaces B_i are used to receive messages by different parties.

More specifically, our resource admits the following types of queries:

- *Registration* queries (made at interface C) register an interface B_i for receiving messages sent to an identity id. (Depending on the envisioned physical registration process, the *fact* that B_i was registered under identity id may become public. We model this by leaking the pair (id, i) at all interfaces B_j.)

[3] The name "delivery controlled channels" indicates that a user can specify (or, control) to which recipient the message should be delivered.

[4] In this work, we focus on passive attacks (i.e., on eavesdropping adversaries). In particular, we will not consider adversarially sent messages. Thus, for simplicity, we will assume that all incoming requests to *send* a message arrive at a single interface A.

- *Send* queries (at interface A) send a message m to a given identity id. (The message will then be delivered to all interfaces which have been registered for this identity. Besides, any interface B_i which is *later* registered for that identity id will also receive m upon registration.)
- When thinking of an IBE scheme as realizing DCC, we cannot prevent dishonest parties from sharing their keys in the real world. As a result, also the messages sent to that party are shared with every party that got the key. Our ideal system DCC has to make this explicit, so we admit *share* queries (at any interface B_i) that cause all messages sent to this interface to be *potentially*[5] published at all other interfaces B_j that have also made a *share* query.

Furthermore, all parties (i.e., all interfaces B_i) at the beginning (potentially) receive an honestly generated random string (that corresponds to the randomness in the public master key of an IBE scheme that can potentially be extracted). We deem an IBE scheme secure if it implements this resource (when used in the straightforward way) in the sense of constructive cryptography. (In particular, this means that the view of any given party using the real IBE scheme can be simulated efficiently with access to the ideal non-interactive communication resource only.) We note that we do not model secret keys or ciphertexts in our ideal resource.

We remark that a possible ideal functionality in the UC setting would not use interfaces, but instead restrict the registration, send, and share queries to different parties. That is, only a designated "master party" could *register* other parties for receiving messages under certain identities. Every party P could *send* messages, and also issue a *share* query (with the same consequences as in our CC-based formulation).

Why current game-based definitions do not realize DCC. Our first observation is that existing game-based definitions of IBE security (such as IND-ID-CPA or IND-ID-CCA) do not appear to realize the above resource. To explain the reason, suppose that one party P performs its own registration (under an arbitrary identity and at an arbitrary interface B_i) *after* messages are sent to P. (Naturally, P will not be able to receive these messages before obtaining his/her own user secret key during registration.) Now we claim that P's view in that scenario cannot be simulated efficiently. Concretely, observe that P's view with a real IBE scheme essentially consists of two elements: first, a ciphertext c of a yet-unknown message m sent by another party; and second, a user secret key usk that allows to decrypt c to m. In order to simulate P's view, a simulator must thus first produce a ciphertext c at a point at which P is not registered as a receiving party. Since at that point, m is not yet known to P, c must in fact be simulated without knowledge of m. Later on, however, the simulator must also produce a user secret key usk that opens c as an encryption of m.

[5] Sharing is not guaranteed because our real system does not include channels between the B_i (since they are not needed). When composed with other systems, it might however be the case that such channels become available, so sharing cannot be excluded in a composable framework.

Put differently, the simulation thus faces a commitment problem: first, it has to commit to a ciphertext c, and later explain this ciphertext as an encryption of an arbitrary message m. For technically very similar reasons, public-key encryption cannot be simulated in the face of *adaptive* corruptions [17]. (However, we stress that in our case, no adaptive corruptions occur; see also the remark below.) As a consequence, we can show that non-interactive communication (as formalized by our resource DCC) cannot be achieved in the standard model. (We also note that this argument applies verbatim to the potential UC-based formulation sketched above.)

Weaker notions of non-interactive communication. Our negative result for the above resource DCC raises the question what we can do to achieve *some* form of non-interactive communication and also what existing, game-based IBE security notions actually achieve.

Recall that the commitment problem that arises with DCC occurs only when identities are registered *after* messages have been sent to this identity. A natural way to avoid this scenario is to assume first a registration phase (in which no message transmissions are allowed), and second a transmission phase (in which no registrations are allowed). This separation into two phases can be modeled as a resource $\mathsf{st2DCC}$ that only allows message transmissions (and from then on ignores registration attempts) after a specific input at the "registration" interface C.[6] We can show that $\mathsf{st2DCC}$ *can* be achieved by IND-ID-CPA secure IBE schemes. In that sense, the commitment problem of DCC is the *only* reason why we cannot achieve that resource. Interestingly, achieving $\mathsf{st2DCC}$ actually corresponds to a game-based notion of IBE security that we introduce and call IND-ID1-CPA security and that is weaker than IND-ID-CPA security.

We also show that IND-ID-CPA security exactly corresponds to a resource stDCC which only allows registrations of identities to which no message has been sent so far. (In that sense, stDCC implements a "local" version of the two-phase separation of $\mathsf{st2DCC}$. Again, we stress that it is the responsibility of the implementation to enforce such a local separation.)

Finally, we provide relaxed resources preDCC and $\mathsf{pre2DCC}$ that are "selective" versions of stDCC and $\mathsf{st2DCC}$, respectively. (Here, "selective" means that the set of identities id that can be registered has to be specified initially, over interface A.) We proceed to show that resource preDCC is achieved precisely by selective IND-ID-CPA secure IBE schemes. Similarly, the resource $\mathsf{pre2DCC}$ is equivalent to a selective version of the game-based notion associated with the resource $\mathsf{st2DCC}$. The relations among security definitions and the achieved constructions are summarized in Fig. 1.

[6] While this separation is easily modeled as a resource, we stress that it is the responsibility of the (designer of the) implementation to physically enforce this separation. For instance, in face of a passive adversary, such a separation into phases could be enforced simply by telling honest parties not to send any messages until the second phase.

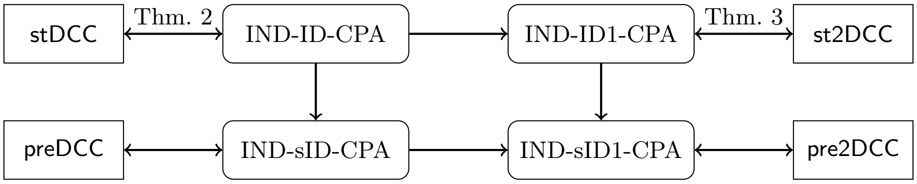

Fig. 1. Implications among security definitions and the constructed resources. Security definitions are drawn in boxes with rounded corners and resources are shown in rectangular boxes. The figure says for example that by Theorem 2, an IBE scheme can be used to construct the resource stDCC if and only if it is IND-ID-CPA secure, while IND-ID-CPA security implies IND-sID-CPA security and IND-ID1-CPA security. The equivalences of the selective security variants and the corresponding constructions are shown in the full version.

Relevance of the impossibility result. While it perhaps appears natural to process all registrations before messages for the corresponding identities are sent, this restriction substantially weakens the usefulness of IBE. For example, if IBE is used in a large context to encrypt emails where the encryption service is independent of the email providers, it seems desirable to be able to send encrypted emails to anyone with a valid email address, without knowing whether they have already registered for the encryption service. In fact, if one has to "ask" whether a user has already received his key before being able to send him a message, one gives up non-interactivity and does not gain much compared to standard public-key encryption.

Moreover, an interesting application, which was suggested in [3], is impossible: Assume the key authority every day publishes a key for the identity that corresponds to the current date. One should now be able to send a message "to the future" by encrypting it for the identity corresponding to, e.g., the following day. We are here precisely in the situation where a ciphertext is received before the corresponding key, so standard IBE does not guarantee the security of this application[7] (our construction in the random oracle model, however, does provide this guarantee).

On dishonest senders. The results in this paper only consider passive attacks, i.e., we assume only honest parties send messages. This makes our impossibility

[7] One can give a less technical argument why standard definitions are insufficient for this application than the inability to simulate: It is not excluded by IND-ID-CPA or IND-ID-CCA that first providing a ciphertext and later the user secret key for the corresponding identity yields a binding commitment (maybe only for some specific subset of the message space). In this case, a dishonest recipient Bob of a ciphertext for the following day can use this ciphertext to commit himself (to some third party) to the encrypted value, and open the commitment on the next day. Note that Bob committed himself to a value *he did not know*, misleading the third party into believing he knew it, which is not possible when an ideal "sending-to-the-future" functionality is used.

result only stronger, and all positive results can in principle be lifted to a setting with potentially dishonest senders by replacing the CPA-definitions with their (R)CCA-counterparts. However, this leads to some subtleties in the modeling. For example, one needs to simulate a dishonest sender sending some nonsensical bit string (which does not constitute a valid ciphertext) to a dishonest receiver. Furthermore, the two phases in the results with a separate registration and transmission phase become intermixed, because only honest parties are prevented from sending during the registration phase. To avoid such technicalities and simplify the presentation, we formulate all results only for honest senders.

1.4 Related Work

On the difference to the IBE ideal functionality of Nishimaki et al. We note that an ideal functionality for IBE has already been presented by Nishimaki et al. [18] in the UC framework. However, unlike our resources (when interpreted as UC functionalities as sketched above), their functionality was constructed directly along the IBE *algorithms*, and not to model the *goal* of non-interactive communication. Besides, their functionality does not guarantee secrecy for ciphertexts generated before the respective receiver has been initialized. (This relaxed guarantee corresponds to our relaxed resource stDCC that disallows registrations after communication attempts.)

As a consequence, [18] could indeed show that the standard game-based definition of security for IBE schemes is equivalent to realizing their ideal functionality. Specifically, their IBE abstraction thus compares differently from ours to game-based IBE security notions.

Relation to functional encryption. Identity-based encryption is known to be a special case of functional encryption [4], which has already been modeled in the constructive cryptography framework [11]. However, the results from that paper cannot directly be applied to the context of non-interactive communication as studied in our paper. One reason is that a different goal was modeled in [11] (namely adding access control to a public repository), where only three parties are considered. More importantly, we analyze security definitions which are specific to IBE, while [11] only considers (simulation based) security definitions for general functional encryption, which are more involved. We note, however, that the same commitment problem arises in the context of functional encryption [4].

Relation to adaptive corruptions in the public-key setting. As noted, *technically*, the commitment problem we encounter is very similar to the commitment problem faced in adaptively secure public-key encryption [17]. There, a simulation would have to first produce a ciphertext (without knowing the supposed plaintext). Later, upon an adaptive corruption of the respective receiver, the simulation would have to provide a secret key that opens that ciphertext suitably.

However, in our case, the actual *setting* in which the problem occurs is not directly related to corruptions. Namely, in our setting, a similar commitment problem occurs because messages may be sent to an identity prior to an “activation” of the corresponding communication channel. (In fact, since the mapping of receiving parties to identities may not be clear beforehand, prior to such an activation it is not even clear where to route the corresponding sent messages.) Hence, we can argue that the commitment problem we face is inherent to the IBE setting, independently of adaptive corruptions (all results in this paper are actually formulated for static corruptions).

2 Preliminaries

Constructive Cryptography The results in this paper are formulated using a simulation-based notion of security. There are many protocol frameworks based on such a simulation-based security notion (e.g., [10,1,15,5,19,13,12]). However, in this work, we use the constructive cryptography (CC) framework [13,12].

Briefly, CC makes statements about *constructions* of *resources* from other resources. A resource is a system with interfaces via which the resource interacts with its environment and which can be thought of as being assigned to parties. *Converters* are systems that can be attached to an interface of a resource to change the inputs and outputs at that interface, which yields another resource. The protocols of honest parties and simulators correspond to converters. Dishonest behavior at an interface is captured by *not* applying the protocol (instead of modeling an explicit adversary). An ideal resource is *constructed* from a real resource by a protocol, if the real resource with the protocol converters attached at the honest interfaces is indistinguishable from the ideal resource with the simulators attached at the dishonest interfaces.

We introduce the relevant concepts in more detail, following [13], in the following subsections. For readers more familiar with the Universal Composability (UC) framework [5], we also include explanations of how the presented concepts relate to similar concepts in UC.

Efficiency and Security Parameters. Negligibility and efficiency is defined with respect to a security parameter and the complexity of all algorithms and systems in this paper is polynomial in this security parameter. Thus, distinguishing advantages and advantages in winning a game are functions of this parameter. To simplify notation, we will omit security parameters and not provide them as additional inputs.

Notation for Algorithms and Systems. The algorithms and systems in this paper are described by pseudocode using the following conventions: For variables x and y, $x \leftarrow y$ denotes the assignment after which x has the value of y. For a finite set $\mathcal{S}$, $x \leftarrow \mathcal{S}$ denotes the assignment of a uniformly random element in $\mathcal{S}$ to x. If $\mathtt{A}$ is an algorithm, $x \leftarrow \mathtt{A}(\dots)$ denotes executing $\mathtt{A}(\dots)$ and assigning the returned value to x. For a probabilistic algorithm $\mathtt{A}$ and a (sufficiently long) bit string r, $\mathtt{A}(r; \ldots)$ denotes the execution of $\mathtt{A}$ with randomness r. We denote the length of a bit string s by $|s|$ and for s_1, s_2, $|(s_1, s_2)|$ denotes the bit length of (some fixed) unique encoding of (s_1, s_2).

2.1 Resources, Converters, and Distinguishers

We consider different types of *systems*, which are objects with *interfaces* via which they interact with their environment. Interfaces are denoted by uppercase letters. One can compose two systems by connecting one interface of each system. The composed object is again a system.

Two types of systems we consider here are *resources* and *converters*. Resources are denoted by bold uppercase letters or sans serif fonts and have a finite set $\mathcal{I}$ of interfaces. Resources with interface set $\mathcal{I}$ are called *$\mathcal{I}$-resources*. Converters have one *inside* and one *outside interface* and are denoted by lowercase Greek letters or sans serif fonts. The inside interface of a converter α can be connected to interface $I \in \mathcal{I}$ of a resource $\mathbf{R}$. The outside interface of α then serves as the new interface I of the composed resource, which is denoted by $\alpha^I \mathbf{R}$. We also write $\alpha_I \mathbf{R}$ instead of $\alpha_I^I \mathbf{R}$ for a converter α_I. For a vector of converters $\alpha = (\alpha_{I_1}, \ldots, \alpha_{I_n})$ with $I_1, \ldots, I_n \in \mathcal{I}$ and a set $\mathcal{P} \subseteq \{I_1, \ldots, I_n\}$ of interfaces, $\alpha_{\mathcal{P}} \mathbf{R}$ denotes the $\mathcal{I}$-resource that results from connecting α_I to interface I of $\mathbf{R}$ for every $I \in \mathcal{P}$. Moreover, $\alpha_{\overline{\mathcal{P}}} \mathbf{R}$ denotes the $\mathcal{I}$-resource one gets when α_I is connected to interface I of $\mathbf{R}$ for every $I \in \{I_1, \ldots, I_n\} \setminus \mathcal{P}$. For $\mathcal{I}$-resources $\mathbf{R}_1, \ldots, \mathbf{R}_m$, the *parallel composition* $[\mathbf{R}_1, \ldots, \mathbf{R}_m]$ is defined as the $\mathcal{I}$-resource where each interface $I \in \mathcal{I}$ allows to access the corresponding interfaces of all sub-systems $\mathbf{R}_i$ as sub-interfaces. Similarly, for converters $\alpha_1, \ldots, \alpha_m$, we define the *parallel composition* $[\alpha_1, \ldots, \alpha_m]$ via $[\alpha_1, \ldots, \alpha_m]^I [\mathbf{R}_1, \ldots, \mathbf{R}_m] \coloneqq [\alpha_1^I \mathbf{R}_1, \ldots, \alpha_m^I \mathbf{R}_m]$.

A *distinguisher* $\mathbf{D}$ for resources with n interfaces is a system with $n+1$ interfaces, where n of them connect to the interfaces of a resource and a bit is output at the remaining one. We write $\Pr[\mathbf{DR} = 1]$ to denote the probability that $\mathbf{D}$ outputs the bit 1 when connected to resource $\mathbf{R}$. The goal of a distinguisher is to distinguish two resources by outputting a different bit when connected to a different resource. Its success is measured by the distinguishing advantage.

Definition 1. *The* distinguishing advantage *of a distinguisher* $\mathbf{D}$ *for resources* $\mathbf{R}$ *and* $\mathbf{S}$ *is defined as*

$$\varDelta^{\mathbf{D}}(\mathbf{R}, \mathbf{S}) \coloneqq |\Pr[\mathbf{DR} = 1] - \Pr[\mathbf{DS} = 1]|.$$

If $\varDelta^{\mathbf{D}}(\mathbf{R}, \mathbf{S}) = 0$ *for all distinguishers* $\mathbf{D}$*, we say* $\mathbf{R}$ *and* $\mathbf{S}$ *are* equivalent*, denoted as* $\mathbf{R} \equiv \mathbf{S}$*. If the distinguishing advantage is negligible for all efficient distinguishers, we say* $\mathbf{R}$ *and* $\mathbf{S}$ *are* computationally indistinguishable*, denoted as* $\mathbf{R} \approx \mathbf{S}$*.*

We introduce two special converters $\mathbf{1}$ and $\perp$. The converter $\mathbf{1}$ forwards all inputs at one of its interfaces to the other one. We thus have for all $\mathcal{I}$-resources $\mathbf{R}$ and all $I \in \mathcal{I}$

$$\mathbf{1}^I \mathbf{R} \equiv \mathbf{R}.$$

One can equivalently understand connecting $\mathbf{1}$ to interface I of a resource as not connecting any converter to that interface. Moreover, the converter $\bot$ blocks all inputs at the connected interface. That is, interface I of $\bot^I\mathbf{R}$ does not accept any inputs and there are no outputs at this interface.

Relation to UC concepts. In UC, systems as above can correspond to protocols, ideal functionalities, or simulators that interact with the protocol environment. More specifically, resources correspond to ideal functionalities, while converters can correspond to real or hybrid protocols, or to simulators. Namely, a UC protocol can be viewed as a way to convert calls to that protocol to calls to an underlying communication infrastructure (or hybrid functionality). Conversely, a UC simulator can be viewed as a way to convert the network interface of one protocol into that of another one. (In CC, there is no a-priori distinction between I/O and network interfaces; hence, both UC protocols and UC simulators correspond to converters.) Distinguishers as above correspond to the UC protocol environments.

2.2 Filtered Resources

In some situations, specific interactions with a resource might not be guaranteed but only potentially available. To model such situations, we extend the concept of a resource. Let $\mathbf{R}$ be an $\mathcal{I}$-resource and let $\phi = (\phi_I)_{I\in\mathcal{I}}$ be a vector of converters. We define the *filtered resource* $\mathbf{R}_\phi$ as a resource with the same set of interfaces $\mathcal{I}$. For a party connected to interface I of $\mathbf{R}_\phi$, interactions through the converter ϕ_I are guaranteed to be available, while interactions with $\mathbf{R}$ directly are only potentially available to dishonest parties. The converter ϕ_I can be seen as a filter shielding specific functionality of interface I. Dishonest parties can potentially remove the filter to get access to all features of the resource $\mathbf{R}$. Formally, $\mathbf{R}_\phi$ is defined as the set of all resources that allows all interactions allowed $\phi_\mathcal{I}\mathbf{R}$ but not more than allowed by $\mathbf{R}$; see [13] for more details.

2.3 Communication Resources

An important example of resources are communication channels, which allow the sender A to send messages from the message space $\mathcal{M} := \{0,1\}^*$ to the receiver B. We define two such channels, which differ in the capabilities of the adversary E. If a channel is used in a context with several potentially dishonest parties, all of them have access to interface E.

Definition 2. *An* authenticated channel from A to B*, denoted as* $\mathsf{AUT}^{A,B}$*, and a* secure channel from A to B*, denoted as* $\mathsf{SEC}^{A,B}$*, are resources with three interfaces A, B, and E. On input a message $m \in \mathcal{M}$ at interface A, they both output the same message m at interface B. Additionally,* $\mathsf{AUT}^{A,B}$ *outputs m at interface E and* $\mathsf{SEC}^{A,B}$ *outputs the length $|m|$ of the message at interface E. Other inputs are ignored. Both channels allow arbitrarily many messages to be sent.*

Remark 1. Alternatively, one could define authenticated and secure channels such that E also has the ability to delete messages. The results in this paper can be adapted to such a setting, but our assumption that sent messages are always delivered allows to simplify the presentation.

For authenticated channels, we do not want to guarantee that an adversary learns the message, it is rather not excluded. Similarly, secure channels should not guarantee that the length of the message leaks. To model this, we introduce filters that block all outputs at interface E. We then have that a secure channel is also authenticated, i.e., the set of (filtered) secure channels is a subset of the set of (filtered) authenticated channels.

Definition 3. *Let* $\phi^{\mathsf{AUT}} = \phi^{\mathsf{SEC}} := (\mathbf{1}, \mathbf{1}, \bot)$. *We will consider the filtered resources* $\mathsf{AUT}^{A,B}_{\phi^{\mathsf{AUT}}}$ *and* $\mathsf{SEC}^{A,B}_{\phi^{\mathsf{SEC}}}$.

Note that

$$\phi^{\mathsf{AUT}}_{\{A,B,E\}}\mathsf{AUT}^{A,B} = \mathbf{1}^A\mathbf{1}^B\bot^E\mathsf{AUT}^{A,B} \equiv \mathbf{1}^A\mathbf{1}^B\bot^E\mathsf{SEC}^{A,B} = \phi^{\mathsf{SEC}}_{\{A,B,E\}}\mathsf{SEC}^{A,B}$$

accepts messages at interface A and outputs them at interface B where interface E is inactive.

We finally introduce a more advanced communication resource that has many interfaces and allows a sender to send messages to all other interfaces. It is authenticated in the sense that the messages cannot be modified and everyone receives the same message.

Definition 4. *The* broadcast *resource* $\mathsf{BCAST}^{A,\mathcal{B}}$ *for a set* $\mathcal{B}$ *has interface set* $\{A\} \cup \mathcal{B}$. *On input a message* $m \in \mathcal{M}$ *at interface* A, *the same message is output at all interfaces* $B \in \mathcal{B}$. *Other inputs are ignored.*

Relation to UC concepts. The presented resources directly correspond to UC ideal functionalities for authenticated, secure, or broadcast channels. The different interfaces of the presented resources correspond to what different parties in UC could send or receive. (Here we note a common design difference in UC and CC: in UC, typically one would assume parties as fixed entities, and model communication and interfaces around them. In CC, one would typically start with the interfaces that reflect the semantic types of in- and outputs of a resource, and only later think of connecting entities like parties.)

2.4 Construction of Resources

A *protocol* is a vector of converters with the purpose of constructing a so-called ideal resource from an available real resource. Depending on which parties are considered potentially dishonest, we get a different notion of construction.

As an example from [8], consider the setting for public-key encryption with honest A and B where we want to construct a secure channel $\mathsf{SEC}^{A,B}_{\phi^{\mathsf{SEC}}}$ from authenticated channels $\mathsf{AUT}^{B,A}_{\phi^{\mathsf{AUT}}}$ and $\mathsf{AUT}^{A,B}_{\phi^{\mathsf{AUT}}}$ in presence of a dishonest eavesdropper E. Here, the real resource is $\left[\mathsf{AUT}^{B,A}_{\phi^{\mathsf{AUT}}}, \mathsf{AUT}^{A,B}_{\phi^{\mathsf{AUT}}}\right]$ and the ideal resource is $\mathsf{SEC}^{A,B}_{\phi^{\mathsf{SEC}}}$. In this setting, a protocol $\pi = (\pi_A, \pi_B, \pi_E)$ constructs $\mathbf{S}$ from $\mathbf{R}$ with potentially dishonest E if there exists a converter σ_E (called *simulator*) such that

$$\begin{aligned} \pi_A\pi_B\pi_E \left[\phi_E^{\mathsf{AUT}}\mathsf{AUT}^{B,A}, \phi_E^{\mathsf{AUT}}\mathsf{AUT}^{A,B}\right] &\approx \phi_E^{\mathsf{SEC}}\mathsf{SEC}^{A,B} \\ \text{and} \qquad \pi_A\pi_B \left[\mathsf{AUT}^{B,A}, \mathsf{AUT}^{A,B}\right] &\approx \sigma_E\mathsf{SEC}^{A,B}, \end{aligned}$$

where σ_E provides a sub-interface to the distinguisher for each channel that constitutes the real resource. The first condition ensures that the protocol implements the required functionality and the second condition ensures that whatever Eve can do when connected to the real resource without necessarily following the protocol, she could do as well when connected to the ideal resource by using the simulator σ_E. Since Eve is here only a hypothetical entity, we typically have $\pi_E = \bot$.

In this paper, we consider the more general setting that includes several potentially dishonest parties that (in contrast to Eve in the above example) also get certain guarantees if they are honest while unable to do more than specified by the ideal resource even if they are dishonest. We define a secure construction as follows.

Definition 5. *Let $\mathbf{R}_\phi$ and $\mathbf{S}_\psi$ be filtered $\mathcal{I}$-resources and let $\pi = (\pi_I)_{I\in\mathcal{I}}$ be a protocol. Further let $\mathcal{U} \subseteq \mathcal{I}$ be the set of interfaces with potentially dishonest behavior. We say π* constructs $\mathbf{S}_\psi$ *from* $\mathbf{R}_\phi$ with potentially dishonest $\mathcal{U}$*, denoted by*

$$\mathbf{R}_\phi \overset{\pi}{\underset{\mathcal{U}}{\Longmapsto}} \mathbf{S}_\psi,$$

if there exist converters $\sigma = (\sigma_U)_{U\in\mathcal{U}}$ such that

$$\forall \mathcal{P} \subseteq \mathcal{U} : \pi_{\overline{\mathcal{P}}}\phi_{\overline{\mathcal{P}}}\mathbf{R} \approx \sigma_{\mathcal{P}}\psi_{\overline{\mathcal{P}}}\mathbf{S}.$$

The converters σ_U are called simulators*.*

For $\mathcal{U} = \mathcal{I}$, this definition corresponds to the abstraction notion from [13], which considers all parties as potentially dishonest. The construction notion is composable in the following sense:

$$\mathbf{R}_\phi \overset{\pi}{\underset{\mathcal{U}}{\Longmapsto}} \mathbf{S}_\psi \;\wedge\; \mathbf{S}_\psi \overset{\pi'}{\underset{\mathcal{U}}{\Longmapsto}} \mathbf{T}_\tau \quad\Longrightarrow\quad \mathbf{R}_\phi \overset{\pi'\pi}{\underset{\mathcal{U}}{\Longmapsto}} \mathbf{T}_\tau,$$

where $\pi'\pi$ is the protocol that corresponds to first applying π and then π' to the resource.

To apply the above definition to an unfiltered resource $\mathbf{R}$, one can formally introduce trivial filters $\phi_I := \mathbf{1}$ for $I \in \mathcal{I}$ and consider the filtered resource $\mathbf{R}_\phi$ which is identical to $\mathbf{R}$. In such cases, we will omit the filters. We refer the reader to [13] for more details.

Relation to UC concepts. The "constructs" notion presented above directly corresponds to the UC notion of secure realization. (The combination of π and $\mathbf{R}$ corresponds to the real protocol in UC, while $\mathbf{S}$ matches the UC ideal protocol.) The "constructs" notion does not consider an explicit adversary on the real protocol. (Instead, in UC terms, a dummy adversary is considered without loss of generality.) There is a difference, however, in the modeling of corruptions. Generally, in UC, adaptive corruptions are considered. In the CC modeling above, only static corruptions of parties are considered. Moreover, instead of modeling corruptions through special "corrupt" messages sent from the adversary or environment, in CC corruptions are modeled simply be letting the distinguisher connect to the interfaces of corrupted parties.

Finally, a subtle difference between CC and UC security is that CC security requires "local" simulators for each interface, whereas in UC, one simulator is required that handles all parties (resp. interfaces) at once. While this makes CC security a stricter notion than UC security, this difference will not be relevant to our results. (In particular, our negative result has nothing to do with the fact that CC security requires local simulation.)

3 Delivery Controlled Channels

A broadcast channel allows a sender A to send messages to recipients $B_1, \ldots, B_n$. One can understand the application of an IBE scheme to add some form of delivery control to such a channel. More specifically, the enhanced channel allows A to send a message for some identity id in an identity space $\mathcal{ID}$ such that only the B_i that are registered for this identity receive the message, even if several other B_i are dishonest. We assume this registration is managed by a central authority C. We formalize this by a *delivery controlled channel* DCC. This resource also allows the registration of identities after messages have been sent for this identity. In this case, the corresponding user after registration learns all such messages.

Because the public key and each ciphertext contain randomness, during initialization and for each sent message, all parties (potentially) receive common randomness. Moreover, when someone gets registered for an identity, this identity together with a corresponding user secret key is sent to this party over a secure channel. By definition, a secure channel can leak the length of the transmitted messages. Since the length of user secret keys can depend on the identity for which the key has been generated and also on the used randomness, dishonest users potentially learn which identity has just been registered for whom and potentially even which randomness was used to generate the corresponding secret key. Furthermore, dishonest recipients can share their secret keys with others in the real world, which has the effect in the ideal world that the other recipients also learn the messages sent for an identity that has been registered for the user who shared his keys. We model this by a special symbol `share` that B_i can input. A message sent for identity id is then received by B_i if id has been registered for B_i or if there is a B_j such that B_i and B_j have input `share` and id has been registered for B_j.

Definition 6. *Let $n, \rho \in \mathbb{N}$, $\mathcal{M} \coloneqq \{0,1\}^*$, and let $\mathcal{ID}$ be a nonempty set. The resource $\mathsf{DCC}^{n,\mathcal{ID},\rho}$ has the interfaces A, C, and B_i for $i \in \{1, \ldots, n\}$. The resource internally manages the set $S \subseteq \{B_1, \ldots, B_n\}$ of interface names that want to share their identities and for each $i \in \{1, \ldots, n\}$, the set $I_i \subseteq \mathcal{ID}$ of identities registered for interface B_i. Initially, both sets are empty. The resource works as follows:*

Initialization

```
$j \leftarrow 1$
$r \leftarrow \{0,1\}^\rho$
for all $i \in \{1, \ldots, n\}$ do
    output $r$ at interface $B_i$
```

Interface *A*

```
Input: $(id_j, m_j) \in \mathcal{ID} \times \mathcal{M}$
$r_j \leftarrow \{0,1\}^\rho$
for all $i \in \{1, \ldots, n\}$ do
    if $id_j \in I_i$ or $\big(B_i \in S$ and $id_j \in \bigcup_{k \in S} I_k\big)$ then
        output $(id_j, m_j, r_j)$ at interface $B_i$
    else
        output $(id_j, |m_j|, r_j)$ at interface $B_i$
$j \leftarrow j + 1$
```

Interface B_i

```
Input: share
$S \leftarrow S \cup \{B_i\}$
```

Interface *C*

```
Input: $(id, i) \in \mathcal{ID} \times \{1, \ldots, n\}$
$I_i \leftarrow I_i \cup \{id\}$
$r \leftarrow \{0,1\}^\rho$
for all $k \in \{1, \ldots, n\}$ do
    output $(id, i, r)$ at interface $B_k$
    if $k = i$ or $\{B_i, B_k\} \subseteq S$ then
        for all $l \in \{1, \ldots, j - 1\}$ such that $id_l = id$ do
            output $m_l$ at interface $B_k$
```

All inputs not matching the given format are ignored.

The randomness that the B_i get corresponds to randomness one can potentially extract from the public key, the ciphertexts, and the length of the user secret keys of an IBE scheme. Honest users are not guaranteed to receive this randomness, we rather cannot exclude that dishonest parties do so. Similarly, we cannot exclude that dishonest parties share their identities, that they learn the identity for which a message is designated and the length of the message without being registered for that identity, and that they learn who gets registered for which identity. To model that these interactions are not guaranteed, we introduce filters to block inputs and outputs at interfaces B_i for honest parties: For $i \in \{1, \ldots, n\}$, let $\phi^{\mathsf{DCC}}_{B_i}$ be the converter that on input $(id, m, r) \in \mathcal{ID} \times \mathcal{M} \times \{0,1\}^\rho$ at its inside interface, outputs (id, m) at its outside interface, on input $m \in \mathcal{M}$ at its inside interface, outputs m at its outside interface, and on input $(id, k, r) \in \mathcal{ID} \times \{1, \ldots, n\} \times \{0,1\}^\rho$ with $k = i$ at its inside interface, outputs id at its outside interface. All other inputs at any of its interfaces are ignored and thereby blocked. Further let $\phi^{\mathsf{DCC}}_A = \phi^{\mathsf{DCC}}_C := \mathbf{1}$ be the converter that forwards all inputs at one of its interfaces to the other one and let $\phi^{\mathsf{DCC}} := (\phi^{\mathsf{DCC}}_A, \phi^{\mathsf{DCC}}_C, \phi^{\mathsf{DCC}}_{B_1}, \ldots, \phi^{\mathsf{DCC}}_{B_n})$. We will consider the filtered resource $\mathsf{DCC}^{n,\mathcal{ID},\rho}_{\phi^{\mathsf{DCC}}}$.

Remark 2. The resource defined above assumes that a central authority C registers all identities and allows one party to have more than one identity and one identity to be registered for several users. That resource can now be used in larger context where this registration process is regulated. For example, one can have a protocol programmed on top of DCC that requires B_i to send his identity together with a copy of his passport to C. Moreover, C could ensure that each identity is registered for at most one user. In such an application, the resource DCC could directly be used without considering how it was constructed. Due to composition of the constructive cryptography framework, we can thus focus on the construction of DCC and decouple confidentiality from the actual registration process.

Static identity management. We now define a more restricted resource that only allows the registration of an identity as long as no message has been sent for this identity.

Definition 7. *Let $n, \rho \in \mathbb{N}$, $\mathcal{M} := \{0,1\}^*$, and let $\mathcal{ID}$ be a nonempty set. The resource $\mathsf{stDCC}^{n,\mathcal{ID},\rho}$ is identical to $\mathsf{DCC}^{n,\mathcal{ID},\rho}$ except that inputs $(id, i) \in \mathcal{ID} \times \{1, \ldots, n\}$ at interface C are ignored if $id \in \bigcup_{k=1}^{j-1}\{id_k\}$. We will use the same filters as above and consider the resource $\mathsf{stDCC}^{n,\mathcal{ID},\rho}_{\phi^{\mathsf{DCC}}}$.*

The above resource prevents identities for which messages have been sent to be registered, but other identities can still be registered. The following resource restricts the registration process further and operates in two phases: Initially, only registrations are allowed and no messages can be sent. At any point, C can end the registration phase and enable A to send messages.

Definition 8. *Let $n, \rho \in \mathbb{N}$, $\mathcal{M} := \{0,1\}^*$, and let $\mathcal{ID}$ be a nonempty set. The resource $\mathsf{st2DCC}^{n,\mathcal{ID},\rho}$ behaves as $\mathsf{DCC}^{n,\mathcal{ID},\rho}$ except that it initially ignores all inputs at interface A. On input the special symbol `end registration` at interface C, the resource outputs `registration ended` at interfaces $B_1, \ldots, B_n$,*[8]

[8] Note that ϕ^{DCC} blocks this output for honest users, i.e., it is not necessarily guaranteed that everyone learns that the registration has ended. It is not excluded by our protocol since C there informs A that messages may now be sent, and this communication could be observed by dishonest users. If it is desirable in an application that everyone learns that the registration has ended, one can still use $\mathsf{st2DCC}^{n,\mathcal{ID},\rho}$ by letting C explicitly send that information to all B_i via an additional channel. This would happen outside of the resource $\mathsf{st2DCC}^{n,\mathcal{ID},\rho}$ as a separate construction.

and from then on ignores all inputs at interface C and allows inputs at interface A. We will consider the filtered resource $\mathsf{st2DCC}^{n,\mathcal{ID},\rho}_{\phi^{\mathsf{DCC}}}$.

Note that when using stDCC, A can prevent the registration of an identity by sending a message for this identity. On the other hand, $\mathsf{st2DCC}$ gives C full control over the registration process while being less dynamic. Depending on the application, one of these resources might be preferable.

Predetermined identities. We finally introduce two resources that additionally require all identities that are used be determined at the beginning. This allows us to capture the guarantees provided by selectively secure IBE schemes (see Definition 11).

Definition 9. *Let $n, \rho \in \mathbb{N}$, $\mathcal{M} := \{0,1\}^*$, and let $\mathcal{ID}$ be a nonempty set. The resources $\mathsf{preDCC}^{n,\mathcal{ID},\rho}$ and $\mathsf{pre2DCC}^{n,\mathcal{ID},\rho}$ have the interfaces A, C, and B_i for $i \in \{1,\dots,n\}$. Before the resources output anything or accept any input, they wait for the input of a finite set $\mathcal{S} \subseteq \mathcal{ID}$ (encoded as a list of its elements) at interface A. On this input, they output `ok` at interfaces $B_1,\dots,B_n$. Afterwards, $\mathsf{preDCC}^{n,\mathcal{ID},\rho}$ behaves identically to $\mathsf{stDCC}^{n,\mathcal{ID},\rho}$ and $\mathsf{pre2DCC}^{n,\mathcal{ID},\rho}$ behaves identically to $\mathsf{st2DCC}^{n,\mathcal{ID},\rho}$ with the exception that they only accept inputs $(id_j, m_j) \in \mathcal{S} \times \mathcal{M}$ at interface A (there is no restriction on inputs at interface C). We will again consider the filtered resources $\mathsf{preDCC}^{n,\mathcal{ID},\rho}_{\phi^{\mathsf{DCC}}}$ and $\mathsf{pre2DCC}^{n,\mathcal{ID},\rho}_{\phi^{\mathsf{DCC}}}$.*[9]

4 IBE Schemes and Protocols

4.1 IBE Schemes and Their Security

Identity-based encryption. An *identity-based encryption (IBE) scheme $\mathcal{E}$* with message space $\mathcal{M}$ and identity space $\mathcal{ID}$ consists of four PPT algorithms. Key generation $\texttt{Gen}()$ outputs a master public key mpk and a master secret key msk. Extraction $\texttt{Ext}(msk, id)$ (for a master secret key msk and an identity $id \in \mathcal{ID}$) outputs a user secret key usk_{id}. Encryption $\texttt{Enc}(mpk, id, m)$ (for a master public key mpk, an identity $id \in \mathcal{ID}$, and a message $m \in \mathcal{M}$) outputs a ciphertext c. Decryption $\texttt{Dec}(usk_{id}, id, c)$ (for a user secret key usk_{id}, an identity $id \in \mathcal{ID}$, and a ciphertext c) outputs a message $m \in \mathcal{M} \cup \{\perp\}$. For correctness, we require that for all $(mpk, msk) \leftarrow \texttt{Gen}()$, all $id \in \mathcal{ID}$, all $m \in \mathcal{M}$, all $c \leftarrow \texttt{Enc}(mpk, id, m)$, and all $usk_{id} \leftarrow \texttt{Ext}(msk, id)$, we always have $\texttt{Dec}(usk_{id}, id, c) = m$.

Standard security definitions for IBE schemes. We first provide the standard security definition for IBE schemes against passive attacks:

Definition 10 (IND-ID-CPA security). *Consider the experiment $\mathsf{Exp}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}}$ in Fig. 2 for an IBE scheme $\mathcal{E} = (\texttt{Gen}, \texttt{Ext}, \texttt{Enc}, \texttt{Dec})$ and an algorithm $\mathcal{A}$. In this experiment, $\mathcal{A}$ is not allowed to output an identity id that it has queried to*

[9] Again, the filter ϕ^{DCC} blocks the outputs `ok` and `registration ended` at interfaces B_i.

```
**Experiment** $\mathsf{Exp}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}}$:
  $(mpk, msk) \leftarrow \mathtt{Gen}()$
  $(st, id, m_0, m_1) \leftarrow \mathcal{A}^{\mathtt{Ext}(msk,\cdot)}(mpk)$
  $b \leftarrow \{0,1\}$
  $c^* \leftarrow \mathtt{Enc}(mpk, id, m_b)$
  $b' \leftarrow \mathcal{A}^{\mathtt{Ext}(msk,\cdot)}(st, c^*)$
  Return 1 if $b' = b$, else return 0
```

```
**Experiment** $\mathsf{Exp}^{\mathsf{ind\text{-}sid\text{-}cpa}}_{\mathcal{E},\mathcal{A}}$:
  $(st, id) \leftarrow \mathcal{A}()$
  $(mpk, msk) \leftarrow \mathtt{Gen}()$
  $(st', m_0, m_1) \leftarrow \mathcal{A}^{\mathtt{Ext}(msk,\cdot)}(st, mpk)$
  $b \leftarrow \{0,1\}$
  $c^* \leftarrow \mathtt{Enc}(mpk, id, m_b)$
  $b' \leftarrow \mathcal{A}^{\mathtt{Ext}(msk,\cdot)}(st', c^*)$
  Return 1 if $b' = b$, else return 0
```

Fig. 2. The IND-(s)ID-CPA experiment with scheme $\mathcal{E}$ and adversary $\mathcal{A}$.

its ***Ext*** *oracle, or to later query id to* ***Ext****. Furthermore, $\mathcal{A}$ must output m_0, m_1 of equal length. Let*

$$\mathsf{Adv}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}} := \Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = 1\right] - 1/2.$$

We say that $\mathcal{E}$ has indistinguishable ciphertexts under chosen-plaintext attacks (is IND-ID-CPA secure) if $\mathsf{Adv}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}}$ is negligible for all PPT $\mathcal{A}$.

We further consider a weaker security notion introduced in [6] where the adversary has to specify the identity he wants to attack at the beginning of the experiment.

Definition 11 (IND-sID-CPA security). *Consider experiment $\mathsf{Exp}^{\mathsf{ind\text{-}sid\text{-}cpa}}_{\mathcal{E},\mathcal{A}}$ in Fig. 2 for an IBE scheme $\mathcal{E} = (\mathtt{Gen}, \mathtt{Ext}, \mathtt{Enc}, \mathtt{Dec})$ and an algorithm $\mathcal{A}$. In this experiment, $\mathcal{A}$ is not allowed to query id to* ***Ext*** *and has to output m_0, m_1 of equal length. Let*

$$\mathsf{Adv}^{\mathsf{ind\text{-}sid\text{-}cpa}}_{\mathcal{E},\mathcal{A}} := \Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}sid\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = 1\right] - 1/2.$$

We say that $\mathcal{E}$ has indistinguishable ciphertexts under selective identity, chosen-plaintext attacks (is IND-sID-CPA secure) if $\mathsf{Adv}^{\mathsf{ind\text{-}sid\text{-}cpa}}_{\mathcal{E},\mathcal{A}}$ is negligible for all PPT $\mathcal{A}$.

Non-adaptive security. We introduce two novel security notions for IBE schemes that loosely correspond to variants of the standard definitions under "lunchtime attacks" [16]. While CCA1 in contrast to CCA allows the adversary only to ask decryption queries in an initial phase, our definitions restrict the adversary to ask Ext queries only in an initial phase.

Definition 12 (IND-(s)ID1-CPA security). *Consider the two experiments $\mathsf{Exp}^{\mathsf{ind\text{-}id1\text{-}cpa}}_{\mathcal{E},\mathcal{A}}$ and $\mathsf{Exp}^{\mathsf{ind\text{-}sid1\text{-}cpa}}_{\mathcal{E},\mathcal{A}}$ for an IBE scheme $\mathcal{E} = (\mathtt{Gen}, \mathtt{Ext}, \mathtt{Enc}, \mathtt{Dec})$ and an algorithm $\mathcal{A}$ in Fig. 3. In these experiments, $\mathcal{A}$ is only considered valid if all queries to its* ***Ext*** *oracle are different from id and if $|m_0| = |m_1|$. Let*

$$\mathsf{Adv}^{\mathsf{ind\text{-}id1\text{-}cpa}}_{\mathcal{E},\mathcal{A}} := \Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}id1\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = 1\right] - 1/2 \quad \textit{and}$$

$$\mathsf{Adv}^{\mathsf{ind\text{-}sid1\text{-}cpa}}_{\mathcal{E},\mathcal{A}} := \Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}sid1\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = 1\right] - 1/2.$$

```
Experiment $\mathsf{Exp}^{\text{ind-id1-cpa}}_{\mathcal{E},\mathcal{A}}$:
  $(mpk, msk) \leftarrow \texttt{Gen}()$
  $st \leftarrow \mathcal{A}^{\texttt{Ext}(msk,\cdot)}()$
  $(st', id, m_0, m_1) \leftarrow \mathcal{A}(st, mpk)$
  $b \leftarrow \{0,1\}$
  $c^* \leftarrow \texttt{Enc}(mpk, id, m_b)$
  $b' \leftarrow \mathcal{A}(st', c^*)$
  Return 1 if $b' = b$, else return 0
```

```
Experiment $\mathsf{Exp}^{\text{ind-sid1-cpa}}_{\mathcal{E},\mathcal{A}}$:
  $(st, id) \leftarrow \mathcal{A}()$
  $(mpk, msk) \leftarrow \texttt{Gen}()$
  $st' \leftarrow \mathcal{A}^{\texttt{Ext}(msk,\cdot)}(st)$
  $(st'', m_0, m_1) \leftarrow \mathcal{A}(st', mpk)$
  $b \leftarrow \{0,1\}$
  $c^* \leftarrow \texttt{Enc}(mpk, id, m_b)$
  $b' \leftarrow \mathcal{A}(st'', c^*)$
  Return 1 if $b' = b$, else return 0
```

Fig. 3. The IND-(s)ID1-CPA experiment with scheme $\mathcal{E}$ and adversary $\mathcal{A}$.

We say that $\mathcal{E}$ has indistinguishable ciphertexts under non-adaptive chosen-plaintext attacks (is IND-ID1-CPA secure) if $\mathsf{Adv}^{\text{ind-id1-cpa}}_{\mathcal{E},\mathcal{A}}$ is negligible for all valid PPT $\mathcal{A}$ and $\mathcal{E}$ has indistinguishable ciphertexts under selective identity, non-adaptive chosen-plaintext attacks (is IND-sID1-CPA secure) if $\mathsf{Adv}^{\text{ind-sid1-cpa}}_{\mathcal{E},\mathcal{A}}$ is negligible for all valid PPT $\mathcal{A}$.

4.2 Using IBE Schemes in Constructions

In this section, we define the real resources we assume to be available and describe the protocol converters that are designed to construct the resources defined in Sect. 3 using an IBE scheme. Whether these constructions are achieved according to Definition 5 depends on the security properties of the IBE scheme, which we analyze in Sect. 5.

Delivery Controlled Channels. To construct a delivery controlled channel from a broadcast channel[10], we use an IBE scheme in a straightforward way: The party at interface C generates all keys, sends the public key authentically to A and the user secret keys securely to the corresponding B_i. To send a message, A broadcasts an encryption thereof and the B_i with matching identity decrypt it. Hence, we need in addition to the broadcast channel an authenticated channel from C to A to transmit the public key and secure channels from C to each B_i. We abbreviate the network consisting of these channels as

$$\mathsf{NW} := \left[\mathsf{BCAST}^{A,\{B_1,\ldots,B_n\}}, \mathsf{AUT}^{C,A}, \mathsf{SEC}^{C,B_1}, \ldots, \mathsf{SEC}^{C,B_n}\right].$$

The real resource in our construction corresponds to the filtered resource $\mathsf{NW}_{\phi^{\mathsf{NW}}}$ where $\phi^{\mathsf{NW}} := (\phi^{\mathsf{NW}}_A, \phi^{\mathsf{NW}}_C, \phi^{\mathsf{NW}}_{B_1}, \ldots, \phi^{\mathsf{NW}}_{B_n})$ with $\phi^{\mathsf{NW}}_I := [\mathbf{1}, \phi^{\mathsf{AUT}}_I, \phi^{\mathsf{SEC}}_I, \ldots, \phi^{\mathsf{SEC}}_I]$ for $I \in \{A, C, B_1, \ldots, B_n\}$.[11]

[10] Note that we consider the sender to be honest in this paper. Hence, assuming a broadcast channel to be available is not a strong assumption.

[11] In this context, the channel SEC^{C,B_i} is a resource with $n+2$ interfaces where interface C corresponds to interface A of the resource in Definition 2, interface B_i

For an IBE scheme $\mathcal{E}$, we define protocol converters enc, dec, and reg as follows and let $\mathsf{IBE} := (\mathsf{enc}, \mathsf{reg}, \mathsf{dec}, \dots, \mathsf{dec})$: The converter enc first expects to receive a master public key mpk at its inside interface and stores it internally. On input a message and identity $(id, m) \in \mathcal{ID} \times \mathcal{M}$ at its outside interface, it computes $c \leftarrow \mathtt{Enc}(mpk, id, m)$ and outputs (id, c) at its inside sub-interface to $\mathsf{BCAST}^{A,\{B_1,\dots,B_n\}}$. The converter dec on input an identity and a corresponding user secret key (id, usk_{id}) at its inside interface, stores this tuple internally and outputs id at its outside interface. For all pairs (id_j, c_j) with $id_j = id$ stored internally, dec computes $m_j \leftarrow \mathtt{Dec}(usk_{id}, id, c_j)$ and outputs m_j at its outside interface. On input an identity and a ciphertext (id, c) at its inside interface, it stores (id, c) internally and if it has stored a user secret key for the identity id, computes $m \leftarrow \mathtt{Dec}(usk_{id}, id, c)$ and outputs (id, m) at its outside interface. The converter reg initially computes $(mpk, msk) \leftarrow \mathtt{Gen}()$, stores msk internally, and outputs mpk at its inside sub-interface to $\mathsf{AUT}^{C,A}_{\phi^{\mathsf{AUT}}}$. On input (id, i) at its outside interface, it computes $usk_{id} \leftarrow \mathtt{Ext}(msk, id)$ and outputs (id, usk_{id}) at its inside sub-interface to $\mathsf{SEC}^{C,B_i}_{\phi^{\mathsf{SEC}}}$.

Static identity management. To construct stDCC, the protocol at interface C has to reject registration requests for identities for which messages have already been sent. To be able to do so, it needs to know for which identities this is the case. We thus assume there is an additional authenticated channel from A to C that is used to inform C about usage of identities. The real resource is then $\mathsf{NW}^{+}_{\phi^{\mathsf{NW}+}}$ for

$$\mathsf{NW}^{+} := \left[\mathsf{BCAST}^{A,\{B_1,\dots,B_n\}}, \mathsf{AUT}^{A,C}, \mathsf{AUT}^{C,A}, \mathsf{SEC}^{C,B_1}, \dots, \mathsf{SEC}^{C,B_n}\right]$$

and $\phi^{\mathsf{NW}^+} := (\phi_A^{\mathsf{NW}^+}, \phi_C^{\mathsf{NW}^+}, \phi_{B_1}^{\mathsf{NW}^+}, \dots, \phi_{B_n}^{\mathsf{NW}^+})$ where for $I \in \{A, C, B_1, \dots, B_n\}$, $\phi_I^{\mathsf{NW}} := [\mathbf{1}, \phi_I^{\mathsf{AUT}}, \phi_I^{\mathsf{AUT}}, \phi_I^{\mathsf{SEC}}, \dots, \phi_I^{\mathsf{SEC}}]$.

We define the protocol $\mathsf{IBE}^{\mathsf{s}} := (\mathsf{enc}^{\mathsf{s}}, \mathsf{reg}^{\mathsf{s}}, \mathsf{dec}^{\mathsf{s}}, \dots, \mathsf{dec}^{\mathsf{s}})$ by describing the differences from IBE as follows: On input $(id, m) \in \mathcal{ID} \times \mathcal{M}$ at its outside interface, $\mathsf{enc}^{\mathsf{s}}$ additionally outputs id at its inside interface to $\mathsf{AUT}^{A,C}_{\phi^{\mathsf{AUT}}}$. The converter $\mathsf{reg}^{\mathsf{s}}$ on input id at its inside interface, stores this identity internally. It subsequently ignores inputs (id, i) at its outside interface if it has stored id.

Note that it is crucial for this construction that $\mathsf{AUT}^{A,C}$ cannot be interrupted or delayed. Otherwise an attacker could prevent C from learning that some identity has already been used to send messages and this identity could still be registered. In practice, one could realize such channel by letting C acknowledge the receipt while A sends the message only after receiving this acknowledgment. This would, however, contradict the goal of non-interactivity.

corresponds to interface B, and interfaces B_j for $j \neq i$ correspond to copies of interface E. Similarly, ϕ_C^{SEC} corresponds to ϕ_A^{SEC} in Definition 3, $\phi_{B_i}^{\mathsf{SEC}}$ corresponds to ϕ_B^{SEC}, and $\phi_{B_j}^{\mathsf{SEC}}$ to ϕ_E^{SEC} for $j \neq i$. For simplicity, we do not introduce a different notation for the different filters.

If such reliable channel is not available, we can still construct $\mathsf{st2DCC}$ from NW using the protocol $\mathsf{IBE}^{\mathsf{2s}} := (\mathsf{enc}^{\mathsf{2s}}, \mathsf{reg}^{\mathsf{2s}}, \mathsf{dec}^{\mathsf{2s}}, \ldots, \mathsf{dec}^{\mathsf{2s}})$ defined as follows: It works as IBE, except that $\mathsf{reg}^{\mathsf{2s}}$ initially does not send mpk to A. On input `end registration` at its outside interface, $\mathsf{reg}^{\mathsf{2s}}$ sends mpk to A and ignores further inputs. The converter $\mathsf{enc}^{\mathsf{2s}}$ ignores all inputs until it receives mpk at its inside interface and from then on handles all inputs as enc.

Remark 3. Note that sending mpk is here used to signal A that it can now start sending messages. Since we assume that the sender is always honest, we do not need to require, e.g., that mpk cannot be computed from user secret keys; as long as mpk has not been sent, A will not send any messages.

Predetermined identities. To construct $\mathsf{preDCC}_{\phi^{\mathsf{DCC}}}$ from $\mathsf{NW}^{+}_{\phi^{\mathsf{NW+}}}$, we define the protocol $\mathsf{IBE}^{\mathsf{p}} = (\mathsf{enc}^{\mathsf{p}}, \mathsf{reg}^{\mathsf{p}}, \mathsf{dec}^{\mathsf{p}}, \ldots, \mathsf{dec}^{\mathsf{p}})$ that uses a selectively secure IBE scheme. The protocol is almost identical to $\mathsf{IBE}^{\mathsf{s}}$ with the difference that $\mathsf{enc}^{\mathsf{p}}$ initially expects a finite set $\mathcal{S} \subseteq \mathcal{ID}$ (encoded as a list of its elements) as input at its outside interface. On this input, it stores $\mathcal{S}$ internally, sends `ok` to C via $\mathsf{AUT}^{A,C}_{\phi^{\mathsf{AUT}}}$, and subsequently ignores all inputs (id, m) for $id \notin \mathcal{S}$. The converter $\mathsf{reg}^{\mathsf{p}}$ initially waits and ignores all inputs at its outside interface until it receives the input `ok` at its inside interface. It then sends mpk to A and from then on behaves identically to $\mathsf{reg}^{\mathsf{2s}}$.

Similarly, we define a protocol $\mathsf{IBE}^{\mathsf{2p}} = (\mathsf{enc}^{\mathsf{2p}}, \mathsf{reg}^{\mathsf{2p}}, \mathsf{dec}^{\mathsf{2p}}, \ldots, \mathsf{dec}^{\mathsf{2p}})$ to construct $\mathsf{pre2DCC}_{\phi^{\mathsf{DCC}}}$ from $\mathsf{NW}^{+}_{\phi^{\mathsf{NW+}}}$. It works as IBE except that $\mathsf{enc}^{\mathsf{2p}}$ initially expects a finite set $\mathcal{S} \subseteq \mathcal{ID}$ (encoded as a list of its elements) as input at its outside interface. On this input, it stores $\mathcal{S}$ internally, sends `ok` to C via $\mathsf{AUT}^{A,C}_{\phi^{\mathsf{AUT}}}$, and ignores all further inputs until it receives mpk over $\mathsf{AUT}^{C,A}_{\phi^{\mathsf{AUT}}}$. From then on, it handles all inputs as enc, but ignores inputs (id, m) for $id \notin \mathcal{S}$. The converter $\mathsf{reg}^{\mathsf{2p}}$ initially waits and ignores all inputs at its outside interface until it receives the input `ok` at its inside interface. It then accepts registration requests at its outside interface as reg. On input `end registration` at its outside interface, $\mathsf{reg}^{\mathsf{2p}}$ sends mpk to A and ignores further inputs.

Remark 4. While both $\mathsf{IBE}^{\mathsf{p}}$ and $\mathsf{IBE}^{\mathsf{2p}}$ need $\mathsf{AUT}^{A,C}_{\phi^{\mathsf{AUT}}}$, $\mathsf{IBE}^{\mathsf{2p}}$ uses this channel only once in the beginning to let A send `ok` to C. The availability of such channel only at the beginning might be easier to guarantee in practice.

5 Constructing Delivery Controlled Channels

5.1 Impossibility of Construction

We now show that there is no IBE scheme that can be used to construct $\mathsf{DCC}_{\phi^{\mathsf{DCC}}}$ from $\mathsf{NW}_{\phi^{\mathsf{NW}}}$.

Theorem 1. *Let $n > 0$, $\mathcal{ID}$ a nonempty set, and let $\rho \in \mathbb{N}$. Then there is no IBE scheme such that we have for the corresponding protocol* IBE

$$\mathsf{NW}_{\phi^{\mathsf{NW}}} \overset{\mathsf{IBE}}{\underset{\{B_1,\dots,B_n\}}{\Longmapsto}} \mathsf{DCC}^{n,\mathcal{ID},\rho}_{\phi^{\mathsf{DCC}}}.$$

Proof. This proof closely resembles Nielsen's impossibility proof of non-committing public-key encryption [17]. Assume $\mathsf{IBE} = (\mathsf{enc}, \mathsf{reg}, \mathsf{dec}, \dots, \mathsf{dec})$ achieves the construction and let $\mathcal{P} := \{B_1\}$. Then there exists a converter σ_{B_1} such that $\mathsf{IBE}_{\overline{\mathcal{P}}}\phi^{\mathsf{NW}}_{\overline{\mathcal{P}}}\mathsf{NW} \approx \sigma_{\mathcal{P}}\phi^{\mathsf{DCC}}_{\overline{\mathcal{P}}}\mathsf{DCC}^{n,\mathcal{ID},\rho}$. Let $id \in \mathcal{ID}$, let ν be an upper bound on the length of the output of $\mathtt{Ext}(\cdot, id)$, and consider the following distinguisher: The distinguisher $\mathbf{D}$ chooses $m \in \{0,1\}^{\nu+1}$ uniformly at random and inputs (id, m) at interface A. Let (id, c) be the resulting output at interface B_1 (if there is no such output, $\mathbf{D}$ returns 0). Then, $\mathbf{D}$ inputs $(id, 1)$ at interface C. Let (id, usk) be the resulting output at interface B_1 and return 0 if there is no such output or if $|usk| > \nu$. Finally, $\mathbf{D}$ inputs first (id, c) and then (id, usk) at the inside interface of dec and returns 1 if dec outputs id and m at its outside interface, and 0 otherwise.

Correctness of the IBE scheme implies that $\mathbf{D}$ always outputs 1 if connected to the real resource. In the ideal world, c is generated independently of m only given $|m|$ because σ_{B_1} does not learn m until $(id, 1)$ is input at interface C. Moreover, there are at most 2^ν possible values for usk such that $|usk| \leq \nu$. Hence, there are at most 2^ν values of m such that there exists a usk that decrypts c to m with probability more than $\frac{1}{2}$. Since m was chosen uniformly from $\{0,1\}^{\nu+1}$, $\mathbf{D}$ outputs 1 with probability at most $\frac{1}{2} + \frac{1}{2} \cdot \frac{1}{2} = \frac{3}{4}$ when connected to the ideal resource. Thus, the distinguishing advantage is at least $\frac{1}{4}$, which is a contradiction. □

5.2 Equivalence of IND-ID-CPA Security and Construction of Statically Delivery Controlled Channels

While no IBE scheme constructs $\mathsf{DCC}_{\phi^{\mathsf{DCC}}}$ from $\mathsf{NW}_{\phi^{\mathsf{NW}}}$, we show that IND-ID-CPA security is sufficient to construct $\mathsf{stDCC}_{\phi^{\mathsf{DCC}}}$ from $\mathsf{NW}^{+}_{\phi^{\mathsf{NW}+}}$. See the full version for a proof.

Lemma 1. *Let ρ be an upper bound on the randomness used in one invocation of **`Gen`**, **`Ext`**, and **`Enc`**. Then, there exist efficient converters $\sigma_{B_1}, \dots, \sigma_{B_n}$ such that for all $\mathcal{P} \subseteq \{B_1, \dots, B_n\}$ and for all efficient distinguishers $\mathbf{D}$ that input at most q messages at interface A, there exists an efficient algorithm $\mathcal{A}$ such that*

$$\Delta^{\mathbf{D}}\left(\mathsf{IBE}^{\mathsf{s}}_{\overline{\mathcal{P}}}\phi^{\mathsf{NW}^+}_{\overline{\mathcal{P}}}\mathsf{NW}^+, \sigma_{\mathcal{P}}\phi^{\mathsf{DCC}}_{\overline{\mathcal{P}}}\mathsf{stDCC}^{n,\mathcal{ID},\rho}\right) = 2q \cdot \left|\mathsf{Adv}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}}\right|.$$

We now prove conversely that IND-ID-CPA security is also necessary for the construction:

Lemma 2. *Let $\rho \in \mathbb{N}$ and $\mathcal{P} \subseteq \{B_1, \dots, B_n\}, \mathcal{P} \neq \emptyset$. Then, for all valid IND-ID-CPA adversaries $\mathcal{A}$ and for all efficient converters σ_{B_i} for $B_i \in \mathcal{P}$, there exists an efficient distinguisher $\mathbf{D}$ such that*

$$\left|\mathsf{Adv}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}}\right| = \Delta^{\mathbf{D}}\left(\mathsf{IBE}^{\mathsf{s}}_{\overline{\mathcal{P}}}\phi^{\mathsf{NW}^+}_{\overline{\mathcal{P}}}\mathsf{NW}^+, \sigma_{\mathcal{P}}\phi^{\mathsf{DCC}}_{\overline{\mathcal{P}}}\mathsf{stDCC}^{n,\mathcal{ID},\rho}\right).$$

Proof. Let $\mathcal{A}$ be a valid IND-ID-CPA adversary and let σ_{B_i} be efficient converters for $B_i \in \mathcal{P}$. Further let $B_i \in \mathcal{P}$. We now define two distinguishers, $\mathbf{D}_0$ and $\mathbf{D}_1$. Let *mpk* be the initial output at interface B_i of the resource connected to the distinguisher (if nothing is output, let *mpk* be some default value[12]). Both distinguishers then invoke $\mathcal{A}(mpk)$. The oracle query id' of $\mathcal{A}$ is answered as follows by both distinguishers: They input (id', i) at interface C and let the answer to the query be $usk_{id'}$ where $(id', usk_{id'})$ is the resulting output of the resource at interface B_i (and let $usk_{id'}$ be some default value if there is no such output). If $\mathcal{A}$ returns (st, id, m_0, m_1), $\mathbf{D}_0$ and $\mathbf{D}_1$ input (id, m_0) and (id, m_1) at interface A, respectively. Now let (id, c^*) be the resulting output at the sub-interface of B_i corresponding to $\mathsf{BCAST}^{A,\{B_1,\dots,B_n\}}$ (and let c^* be some default value if there is no such output). Both distinguishers then invoke $\mathcal{A}$ on input (st, c^*). Oracle queries are answered as above. Note that *id* will not be queried since $\mathcal{A}$ is a valid IND-ID-CPA adversary and therefore inputs at interface C will be handled as before. Finally, $\mathbf{D}_0$ and $\mathbf{D}_1$ output the bit returned by $\mathcal{A}$.

Note that for all $\beta \in \{0,1\}$

$$\Pr\left[\mathbf{D}_\beta\left(\mathsf{IBE}^{\mathsf{s}}_{\mathcal{P}}\phi^{\mathsf{NW}^+}_{\mathcal{P}}\mathsf{NW}^+\right) = 1\right] = \Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = \beta \,\middle|\, b = \beta\right]$$

because the outputs of the real system are precisely generated as the corresponding values in the IND-ID-CPA experiment. Further note that we have

$$\Pr\left[\mathbf{D}_0\left(\sigma_{\mathcal{P}}\phi^{\mathsf{DCC}}_{\mathcal{P}}\mathsf{stDCC}^{n,\mathcal{ID},\rho}\right) = 1\right] = \Pr\left[\mathbf{D}_1\left(\sigma_{\mathcal{P}}\phi^{\mathsf{DCC}}_{\mathcal{P}}\mathsf{stDCC}^{n,\mathcal{ID},\rho}\right) = 1\right]$$

since $\mathbf{D}_0$ and $\mathbf{D}_1$ only differ in the message they input and σ_{B_i} only learns the length of that message, which is the same for the two messages (since $\mathcal{A}$ is a valid IND-ID-CPA adversary), so its output does not depend on the choice of the message. Now let $\mathbf{D}$ be the distinguisher that chooses $\beta \in \{0,1\}$ uniformly at random, runs $\mathbf{D}_\beta$, and outputs the XOR of $\mathbf{D}_\beta$'s output and β. We conclude

$$\begin{aligned}
&\left|\mathsf{Adv}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}}\right| = \left|\Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = 1\right] - \frac{1}{2}\right| \\
&= \frac{1}{2}\left|\Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = 1 \,\middle|\, b = 0\right] + \Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = 1 \,\middle|\, b = 1\right] - 1\right| \\
&= \frac{1}{2}\left|\Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = 0 \,\middle|\, b = 0\right] - \Pr\left[\mathsf{Exp}^{\mathsf{ind\text{-}id\text{-}cpa}}_{\mathcal{E},\mathcal{A}} = 1 \,\middle|\, b = 1\right]\right| \\
&= \frac{1}{2}\left|\Pr\left[\mathbf{D}_0\left(\mathsf{IBE}^{\mathsf{s}}_{\mathcal{P}}\phi^{\mathsf{NW}^+}_{\mathcal{P}}\mathsf{NW}^+\right) = 1\right] - \Pr\left[\mathbf{D}_1\left(\mathsf{IBE}^{\mathsf{s}}_{\mathcal{P}}\phi^{\mathsf{NW}^+}_{\mathcal{P}}\mathsf{NW}^+\right) = 1\right]\right| \\
&= \frac{1}{2}\Bigg|\Pr\left[\mathbf{D}_0\left(\mathsf{IBE}^{\mathsf{s}}_{\mathcal{P}}\phi^{\mathsf{NW}^+}_{\mathcal{P}}\mathsf{NW}^+\right) = 1\right] + \Pr\left[\mathbf{D}_1\left(\mathsf{IBE}^{\mathsf{s}}_{\mathcal{P}}\phi^{\mathsf{NW}^+}_{\mathcal{P}}\mathsf{NW}^+\right) = 0\right] \\
&\quad - \Pr\left[\mathbf{D}_0\left(\sigma_{\mathcal{P}}\phi^{\mathsf{DCC}}_{\mathcal{P}}\mathsf{stDCC}^{n,\mathcal{ID},\rho}\right) = 1\right] - \Pr\left[\mathbf{D}_1\left(\sigma_{\mathcal{P}}\phi^{\mathsf{DCC}}_{\mathcal{P}}\mathsf{stDCC}^{n,\mathcal{ID},\rho}\right) = 0\right]\Bigg| \\
&= \Delta^{\mathbf{D}}\left(\mathsf{IBE}^{\mathsf{s}}_{\mathcal{P}}\phi^{\mathsf{NW}^+}_{\mathcal{P}}\mathsf{NW}^+, \sigma_{\mathcal{P}}\phi^{\mathsf{DCC}}_{\mathcal{P}}\mathsf{stDCC}^{n,\mathcal{ID},\rho}\right).
\end{aligned}$$

□

[12] Note that this is only possible in the ideal system if σ_{B_i} is flawed. Hence, one could distinguish better in this case, but we do not need that for the proof.

Lemmata 1 and 2 together imply the following theorem:

Theorem 2. *Let ρ be an upper bound on the randomness used in one invocation of* Gen*,* Ext*, and* Enc*. We then have*

$$\mathsf{NW}^{+}_{\phi^{\mathsf{NW}+}} \overset{\mathsf{IBE}^{\mathsf{s}}}{\underset{\{B_1,\ldots,B_n\}}{\Longmapsto}} \mathsf{stDCC}^{n,\mathcal{ID},\rho}_{\phi^{\mathsf{DCC}}}$$
$$\iff \text{ the underlying IBE scheme is IND-ID-CPA-secure.}$$

The following theorem can be proven very similarly by observing that the reductions used to prove Theorem 2 translate queries to the Ext oracle by the adversary to inputs at interface C by the distinguisher and vice versa and that $\mathsf{NW}_{\phi^{\mathsf{NW}}}$ and $\mathsf{st2DCC}^{n,\mathcal{ID},\rho}_{\phi^{\mathsf{DCC}}}$ restrict such inputs exactly as $\mathcal{A}$ is restricted in $\mathsf{Exp}^{\mathsf{ind\text{-}id1\text{-}cpa}}_{\mathcal{E},\mathcal{A}}$.

Theorem 3. *Let ρ be an upper bound on the randomness used in one invocation of* Gen*,* Ext*, and* Enc*. We then have*

$$\mathsf{NW}_{\phi^{\mathsf{NW}}} \overset{\mathsf{IBE}^{\mathsf{2s}}}{\underset{\{B_1,\ldots,B_n\}}{\Longmapsto}} \mathsf{st2DCC}^{n,\mathcal{ID},\rho}_{\phi^{\mathsf{DCC}}}$$
$$\iff \text{ the underlying IBE scheme is IND-ID1-CPA-secure.}$$

Selective security. We similarly show the equivalence of IND-sID-CPA security and the construction of statically delivery controlled channels with predetermined identities in the full version.

6 Construction with Random Oracles

6.1 Random Oracles

We show how any IND-ID-CPA secure IBE scheme $\mathcal{E} = (\mathtt{Gen}, \mathtt{Ext}, \mathtt{Enc}, \mathtt{Dec})$ can be used to construct DCC from the resource $\mathsf{NW}^{\mathsf{RO}}$, which corresponds to our network together with a random oracle. A random oracle is a uniform random function $\{0,1\}^* \to \{0,1\}^k$ for some k to which all parties have access. The heuristic to model a hash function as a random oracle was proposed by Bellare and Rogaway [2]. Theorem 1 implies that no hash function can be used to instantiate the random oracle in this construction. However, if a random oracle is actually available, e.g., via a trusted party or secure hardware, the overall construction is sound. For our purpose, it is sufficient to consider random oracles with binary codomain.

Definition 13. *The resource* RO *has interfaces A, C, and $B_1, \ldots, B_n$. On input $x \in \{0,1\}^*$ at interface $I \in \{A, C, B_1, \ldots, B_n\}$, if x has not been input before (at any interface),* RO *chooses $y \in \{0,1\}$ uniformly at random and outputs y at interface I; if x has been input before and the resulting output was y,* RO *outputs y at interface I.*

Programmability. For our construction, we will assume that a random oracle is available as part of the real resource. Our protocol then constructs an ideal resource that does not give the honest parties access to the random oracle. Thus, the simulators in the ideal world can answer queries to the random oracle arbitrarily as long as they are consistent with previous answers and are indistinguishable from uniform bits. This gives the simulators additional power which allows us to overcome the impossibility result from Theorem 1. Since the simulators can in some sense "reprogram" the random oracle, we are in a scenario that is often referred to as *programmable random oracle model.*

6.2 Construction of Delivery Controlled Channels

Our protocol $\mathsf{IBE}^{\mathsf{ro}}$ uses the same idea as Nielsen's scheme [17] and essentially corresponds to the transformation from [4, Section 5.3] (see also [11]) applied to an IBE scheme. At a high level, it works as follows: To send a message m for identity id, choose a bit string r (of sufficient length, say λ) uniformly at random, input $(r, 1), \ldots, (r, |m|)$ to the random oracle to obtain a uniform value r' with $|r'| = |m|$. Finally encrypt r with the IBE scheme for identity id and send the resulting ciphertext together with $m \oplus r'$. The security proof exploits that the one-time pad is non-committing and the random oracle is programmable.

A detailed description of the protocol and the involved resources as well as a proof sketch of the following theorem can be found in the full version.

Theorem 4. *Let ρ be an upper bound on the randomness used in one invocation of* `Gen`*,* `Ext` *and* `Enc`*. If $\mathcal{E}$ is IND-ID-CPA secure, we have*

$$\mathsf{NW}^{\mathsf{RO}}_{\phi^{\mathsf{NW}^{\mathsf{RO}}}} \overset{\mathsf{IBE}^{\mathsf{ro}}}{\underset{\{B_1,\ldots,B_n\}}{\Longmapsto}} \left[\mathsf{DCC}^{n,\mathcal{ID},\rho+\lambda}_{\phi^{\mathsf{DCC}}}, \mathsf{RO}_{\phi^{\mathsf{RO}}}\right].$$

Acknowledgments. Ueli Maurer was supported by the Swiss National Science Foundation (SNF), project no. 200020-132794. Dennis Hofheinz was supported by DFG grants HO 4534/2-2 and HO 4534/4-1.

References

1. Beaver, D.: Foundations of secure interactive computing. In: Proceedings of CRYPTO 1991. pp. 377–391. No. 576 in Lecture Notes in Computer Science, Springer (1991)
2. Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Proceedings of the 1st ACM Conference on Computer and Communications Security. pp. 62–73. CCS '93, ACM, New York, NY, USA (1993)
3. Boneh, D., Franklin, M.K.: Identity-based encryption from the weil pairing. In: Proceedings of CRYPTO 2001. pp. 213–229. No. 2139 in Lecture Notes in Computer Science, Springer (2001)
4. Boneh, D., Sahai, A., Waters, B.: Functional encryption: Definitions and challenges. In: Ishai, Y. (ed.) Theory of Cryptography, Lecture Notes in Computer Science, vol. 6597, pp. 253–273. Springer Berlin / Heidelberg (2011)

5. Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: Proceedings of FOCS 2001. pp. 136–145. IEEE Computer Society (2001)
6. Canetti, R., Halevi, S., Katz, J.: A forward-secure public-key encryption scheme. In: Biham, E. (ed.) Advances in Cryptology — EUROCRYPT 2003, Lecture Notes in Computer Science, vol. 2656, pp. 255–271. Springer Berlin Heidelberg (2003)
7. Cocks, C.: An identity based encryption scheme based on quadratic residues. In: Proceedings of IMA Int. Conf. 2001. pp. 360–363. No. 2260 in Lecture Notes in Computer Science, Springer (2001)
8. Coretti, S., Maurer, U., Tackmann, B.: Constructing confidential channels from authenticated channels—public-key encryption revisited. In: Sako, K., Sarkar, P. (eds.) Advances in Cryptology - ASIACRYPT 2013, Lecture Notes in Computer Science, vol. 8269, pp. 134–153. Springer Berlin Heidelberg (2013)
9. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of STOC 2008. pp. 197–206. ACM (2008)
10. Goldreich, O., Micali, S., Wigderson, A.: How to play any mental game or a completeness theorem for protocols with honest majority. In: Proceedings of STOC 1987. pp. 218–229. ACM (1987)
11. Matt, C., Maurer, U.: A definitional framework for functional encryption. Cryptology ePrint Archive, Report 2013/559 (2013)
12. Maurer, U.: Constructive cryptography – a new paradigm for security definitions and proofs. In: Mödersheim, S., Palamidessi, C. (eds.) Theory of Security and Applications, Lecture Notes in Computer Science, vol. 6993, pp. 33–56. Springer Berlin Heidelberg (2012)
13. Maurer, U., Renner, R.: Abstract cryptography. In: Chazelle, B. (ed.) The Second Symposium on Innovations in Computer Science, ICS 2011. pp. 1–21. Tsinghua University Press (Jan 2011)
14. Maurer, U.M., Yacobi, Y.: Non-interactive public-key cryptography. In: Proceedings of EUROCRYPT 1991. pp. 498–507. No. 547 in Lecture Notes in Computer Science, Springer (1991)
15. Micali, S., Rogaway, P.: Secure computation (abstract). In: Proceedings of CRYPTO 1991. pp. 392–404. No. 576 in Lecture Notes in Computer Science, Springer (1991)
16. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: STOC. pp. 427–437. ACM (1990)
17. Nielsen, J.B.: Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case. In: Proceedings of CRYPTO 2002. pp. 111–126. No. 2442 in Lecture Notes in Computer Science, Springer (2002)
18. Nishimaki, R., Manabe, Y., Okamoto, T.: Universally composable identity-based encryption. In: Proceedings of VIETCRYPT 2006. pp. 337–353. No. 4341 in Lecture Notes in Computer Science, Springer (2006)
19. Pfitzmann, B., Waidner, M.: A model for asynchronous reactive systems and its application to secure message transmission. In: Proceedings of IEEE Symposium on Security and Privacy 2001. pp. 184–200. IEEE Computer Society (2001)
20. Shamir, A.: Identity-based cryptosystems and signature schemes. In: Proceedings of CRYPTO 1984. pp. 47–53. No. 196 in Lecture Notes in Computer Science, Springer (1984)
21. Waters, B.: Efficient identity-based encryption without random oracles. In: Proceedings of EUROCRYPT 2005. pp. 114–127. No. 3494 in Lecture Notes in Computer Science, Springer (2005)