Security Notions for Bidirectional Channels

Giorgia Azzurra Marson and Bertram Poettering

Ruhr University Bochum, Germany
{giorgia.marson,bertram.poettering}@rub.de

Abstract. This paper closes a definitional gap in the context of modeling cryptographic two-party channels. We note that, while most security models for channels consider exclusively unidirectional communication, real-world protocols like TLS and SSH are rather used for bidirectional interaction. The motivational question behind this paper is: Can analyses conducted with the unidirectional setting in mind—including the current ones for TLS and SSH—also vouch for security in the case of bidirectional channel usage? And, in the first place, what does security in the bidirectional setting actually mean?
After developing confidentiality and integrity notions for bidirectional channels, we analyze a standard way of combining two unidirectional channels to realize one bidirectional channel. Although it turns out that this construction is, in general, not as secure as commonly believed, we confirm that for many practical schemes security is provided also in the bidirectional sense.

Keywords: cryptographic channels, bidirectional communication, security models, TLS

1 Introduction

Communication Channels. Suppose that two parties, Alice and Bob, wish to exchange messages reliably. This means that Alice can send a sequence of messages to Bob, that Bob eventually receives these messages, and that their delivery occurs according to the sending order. The same is required if Bob sends and Alice receives. Protocols like TCP/IP realize exactly this functionality. However, such network protocols guarantee nothing in the presence of adversaries that tamper with the transmission.

One of the most widely-deployed applications of cryptography is to establish *secure* connections over the Internet, allowing secure transmission of data between two endpoints over an unprotected network. Prominent examples include the Transport Layer Security (TLS) protocol [DR08] and the Secure Shell (SSH) protocol [YL06], which both add a layer of protection on top of reliable network protocols. A secure connection is established by running an initial key exchange step through which the two endpoints establish a shared secret, and a subsequent cryptographic channel (a.k.a. secure channel) which uses the secret to protect the actual communication. In this work we focus on the second cryptographic component: the cryptographic channel.

Security: State of the Art. Several research efforts have been devoted to channel security, the result being a generally undisputed understanding of which properties a cryptographic channel should provide. The main features expected from cryptographic channels are data confidentiality and integrity—ensuring that the transmitted messages can only be accessed by the intended recipient and cannot be modified along the way without detection, respectively. Confidentiality and integrity are usually, but not necessarily, required simultaneously. Beyond these properties, in most situations it is desired that out-of-order delivery and replays of messages also be detected. Bellare, Kohno, and Namprempre [BKN02] were the first to formalize the above security goals and for this they introduced the notion of stateful authenticated encryption (stateful AE) as the primitive that meets them. They then used the stateful AE security model as a reference to analyze the SSH Binary Packet protocol. Later work by the same and other authors [KPB03,PRS11,JKSS12,KPW13,BMM+15], particularly in the context of analyzing the TLS Record Protocol [DR08], either confirms or refines the stateful AE notion. All in all, stateful AE is considered a reasonable abstraction of a secure channel.

We note, however, that although stateful AE has been introduced to analyze *bidirectional* channel protocols (concretely, SSH in [BKN02]), this primitive idealizes a *unidirectional* channel. Indeed, [BKN02] and its follow-ups all consider a restricted scenario where Alice sends messages but never receives and, conversely, Bob receives messages but never sends. Thus, existing work assessing the cryptographic

security of prominent protocols like TLS and SSH (as done in the above-mentioned papers) only accounts for the much simpler scenario in which the communication takes place *in one direction*, from the sender to the receiver. Thus, there is an evident gap between how secure channels are modeled in theory and how they are meant to be in practice. This paper fills this gap.

Towards Defining Bidirectional Security. Our first objective is to understand what it means to protect bidirectional communication. Since we are interested in security properties, henceforth we may refer to *cryptographic channels* simply as 'channels'. Intuitively, we expect that a bidirectional channel ensures confidentiality and integrity of data for both directions of communication.

A first attempt to define bidirectional security may be to require that (unidirectional) security holds in each direction independently of the other direction. According to this notion, a bidirectional channel would be deemed secure if 'it behaves as a secure unidirectional channel' when used to protect either direction, from Alice to Bob ($\rightarrow$) or from Bob to Alice ($\leftarrow$). Adopting such a notion would immediately allow to extend the existing analyses of the SSH and TLS channel protocols to the bidirectional case. This notion is, however, completely flawed. Indeed, we can design bidirectional channels that achieve the strongest confidentiality and integrity properties as long as the communication is restricted to one direction but become vulnerable as soon as a second direction of communication is available. We present one such scheme in Appendix B.

As a second attempt we may try to repair the notion above by requiring that (unidirectional) security holds in each direction even if both directions are available simultaneously. That is, according to this stronger notion a bidirectional channel would be declared as secure if each direction enjoys (unidirectional) security against adversaries attacking that specific direction. Let us validate this notion against a widely-deployed channel design that realizes a bidirectional channel by running two unidirectional channels in opposite directions. We name this construction the *canonic composition* of unidirectional channel, where 'canonic' indicates that it follows a common design of real-world channel protocols—including TLS and SSH—that combines two independent unidirectional channels to realize a bidirectional channel. For reference, we give the details of the canonic composition in Section 5 (see Figure 5), however, for now an intuitive understanding is sufficient. Assume that the two unidirectional channels offer confidentiality against active adversaries (a.k.a. indistinguishability against chosen-ciphertext attacks, IND-CCA). Then their canonic composition would be considered a confidential bidirectional channel according to our second-attempt notion. However, the latter notion misses an important point: it ignores the possibility that attacking one direction may indirectly harm the other direction. We clarify this with a practical example.

Consider an instant messaging service that allows registered users, after authenticating with a password, to chat with any other user of the service. Alice and Bob engage in a conversation and, since they care about confidentiality, they run the service over a bidirectional cryptographic channel that offers confidentiality against active attacks. If Alice and Bob follow the canonic composition paradigm and communicate using two independent, IND-CCA-secure channels, do they achieve the desired level of security? They do not, even if the underlying unidirectional channels are secure against active attacks. Indeed, assume the channel is such that the adversary is able to inject ciphertexts that decrypt to messages of its choice.[1] Under this condition, here is how the adversary proceeds. It delivers in the $B \rightarrow A$ direction a ciphertext that Alice decrypts to 'please authenticate'; Alice answers by sending her password over the $A \rightarrow B$ channel; as Alice's message comes unexpected and Bob cannot make sense out of it, he puts the password on public display; the adversary learns it from there. See Figure 1 for an illustration of the attack.

Intuitively, a bidirectional channel with confidentiality against *active* adversaries should prevent this attack from working (more precisely: it does not have to identify and report the attack but ensure that any information that Bob recovers under attack and potentially makes public be independent of what Alice sent). Evidently, the canonic composition falls short in providing this kind of protection.

As the described attack involves tampering with ciphertexts, one could come to the conclusion that requiring the unidirectional channels to provide integrity in addition to confidentiality would solve the problem. Is this change sufficient? Is it necessary? We do not question that demanding integrity protection

[1] This assumption does not contradict a pure confidentiality notion: IND-CCA security only requires that the outputs of the decryption algorithm in case of an active attack be *independent* of the encrypted messages. For a concrete example, see the proof of Theorem 5.

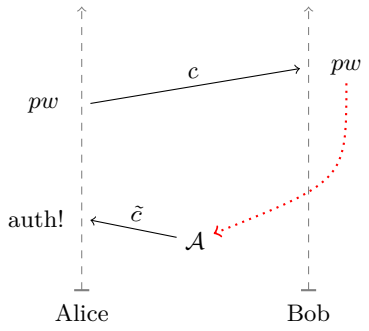

Fig. 1. A confidentiality attack against the canonic composition of two IND-CCA-secure unidirectional channels. In the figure time evolves bottom-up (dashed lines).

from a cryptographic channel is a good idea in general. However, making integrity a necessary part of the model also obstructs the view on the core of its security properties.

In this work we propose a security model for bidirectional communication that naturally extends the idea of ensuring 'unidirectional security' in the two directions but also captures the intuition that attacking one direction may affect the other direction. The two naive notions presented above fail because they both consider a bidirectional channel as 'built' from unidirectional channels. Our model instead sees a bidirectional channel 'as a whole'. After defining appropriate notions of confidentiality and integrity for bidirectional channels we will formally recast the attack from Figure 1. Using our framework, it will be evident that adding integrity to the unidirectional components does prevent the attack. However, requiring integrity is not necessary, in general, to achieve bidirectional confidentiality.

Contribution and Organization. In the first part of the paper we develop security notions for bidirectional channels. Following a long tradition (e.g., [KPB03,PRS11,JKSS12,KPW13,BMM+15]) our definitions are game based. In Section 3 we introduce a joint syntax for unidirectional and bidirectional channels. For bidirectional channels we then propose two flavors of integrity, INT-2PTXT and INT-2CTXT, as well as two flavors of confidentiality, IND-2CPA and IND-2CCA, in Section 4. Our models generalize the confidentiality and integrity notions for unidirectional channels by BKN [BKN02] (the latter we denote in the rest of the paper with INT-1PTXT, INT-1CTXT, IND-1CPA and IND-1CCA to avoid confusion with the bidirectional setting; for reference, we reproduce details of the four BKN models in Appendix A). In Section 4 we also study the relations among the newly defined notions and show that standard implications also hold in the bidirectional setting. Most notably, we prove a generalized version of the classic result that ciphertext integrity leverages confidentiality against passive attacks to confidentiality against active attacks, shortly IND-2CPA + INT-2CTXT $\Longrightarrow$ IND-2CCA. This result can be seen as a benchmark for the soundness of our notions. In the second part of the paper, in Section 5, we apply our model to scrutinize the canonic composition, an important real-world channel design that realizes a bidirectional channel from two unidirectional channels. More specifically, we study how security scales from the unidirectional components to the composed bidirectional channel. We particularly prove that the resulting bidirectional channel inherits both plaintext integrity and ciphertext integrity of its building blocks. We also prove that confidentiality against passive attacks can be lifted. The same does not hold for confidentiality against chosen-ciphertext attacks. We show the latter by giving an explicit counter-example (which formalizes the confidentiality attack from Figure 1).

Further Related Work. It is fair to say that the seminal work of Bellare, Kohno, and Namprempre [BKN02] is considered the reference for channel models in the game-based tradition. Black *et al.* [KPB03] extend the notions from [BKN02] to capture further confidentiality and authenticity goals, e.g., for protecting against combinations of packet loss, replay, and reordering attacks. Boldyreva *et al.* [BDPS12] refine the model of [BKN02] and pioneer the study of symmetric encryption in the presence of ciphertext fragmentation where the decryption processes ciphertexts in a byte-by-byte fashion. In different work [BDPS14] the same authors extend the security model of [BKN02] to allow for multiple decryption errors which occur in some implementations. More recently, Fischlin *et al.* [FGMP15] introduce security notions for channels that transport a stream of bytes rather than a sequence of (atomic) messages. The security

models of [BKN02] and their numerous successors have been employed to prove the security of the full TLS suite (key exchange and channel protocol) and other protocols, e.g., in [KPW13,JKSS12].

An approach towards cryptographic channels from the perspective of composability with other primitives is pursued in [CK01,CK02,MRT12]. For instance, Canetti and Krawczyk [CK02] consider secure channels in the UC framework. They define an ideal functionality for secure channels that lets users communicate over a bidirectional link. While their model in principle does consider bidirectional communication, the concept of attacking one direction by manipulating the other direction is not reflected in their work. Prior work [CK01] by the same authors has a slightly more restricted model but receives a closer look by Namprempre [Nam02] who characterizes (game-based) notions that suffice to achieve a UC secure channel as per [CK01]. Recent works by Maurer *et al.* [MRT12,BMM+15] consider cryptographic channels, from the point of view of Constructive Cryptography (CC), as a unidirectional primitive.

2 Notation

Our security definitions are based on games played between a challenger and an adversary. These games are expressed using program code and terminate when a 'Stop' instruction is executed; the argument of the latter is the output of the game. We write $\Pr[G \Rightarrow 1]$ for the probability that game G terminates by running into a 'Stop with 1' instruction.

In game definitions, we distinguish the following operators for assigning values to variables: We use symbol '$\leftarrow$' when the assigned value results from a constant expression (including the output of a deterministic algorithm), and we write '$\leftarrow_{\$}$' when the value is either sampled uniformly at random from a finite set or is the output of a randomized algorithm.

We use bracket notation to denote associative arrays (a data structure that implements a 'dictionary'). For instance, for an associative array A the instruction $A[7] \leftarrow 3$ assigns value 3 to memory position 7, and the expression $A[2] = 5$ tests whether the value at position 2 is equal to 5. Associative arrays can be indexed with elements from arbitrary sets.

We denote the Boolean constants True and False with T and F, respectively. We sometimes use the ternary operator known from the C programming language: If C is a Boolean condition and e_1, e_2 are arbitrary expressions, the expression "C ? e_1 : e_2" evaluates to e_1 if $C = \mathrm{T}$ and to e_2 if $C = \mathrm{F}$.

If A, B are sets, with $A \cup B$ we denote their disjoint union.

3 Cryptographic Channels

We give a syntax definition that covers both unidirectional and bidirectional channels. Security notions for unidirectional channels are standard and reproduced in Appendix A. Security notions for bidirectional channels, and the relations among them, are studied in Section 4.

Our concept of cryptographic channel assumes two participants that we routinely refer to as Alice (A) and Bob (B). In the unidirectional setting, Alice invokes the *send* algorithm to transform messages into ciphertexts and Bob invokes the *receive* algorithm to translate ciphertexts back into messages. In the bidirectional setting, both parties can send and receive. In our formalization, the send and receive algorithms also take associated data [Rog02] that is assumed to match on both sides. Further, we assume both participants keep state between invocations of their algorithms.

Definition 1 (Syntax of channels). *A* (cryptographic) channel $\mathsf{Ch} = (\mathsf{init}, \mathsf{snd}, \mathsf{rcv})$ *for associated data space $\mathcal{AD}$ and message space $\mathcal{M}$ consists of a key space $\mathcal{K}$, a ciphertext space $\mathcal{C}$, a state space $\mathcal{S}$, a distinguished rejection symbol $\perp \notin (\mathcal{M} \cup \mathcal{S})$, and three efficient deterministic algorithms as follows:*

- *The* initialization *algorithm takes a key $K \in \mathcal{K}$ and outputs initial states $st_A, st_B \in \mathcal{S}$. We write $(st_A, st_B) \leftarrow \mathsf{init}(K)$. Overloading notation, we sometimes write $(st_A, st_B) \leftarrow_{\$} \mathsf{init}$ as an abbreviation for $K \leftarrow_{\$} \mathcal{K}$ followed by $(st_A, st_B) \leftarrow \mathsf{init}(K)$, i.e., the initialization of a channel with a uniform (but anonymous) key.*
- *The* sending *algorithm takes a state $st \in \mathcal{S}$, associated data $ad \in \mathcal{AD}$, and a message $m \in \mathcal{M}$, and outputs an updated state $st' \in \mathcal{S}$ together with a ciphertext $c \in \mathcal{C}$. We write $(st', c) \leftarrow \mathsf{snd}(st, ad, m)$.*
- *The* receiving *algorithm takes a state $st \in \mathcal{S}$, associated data $ad \in \mathcal{AD}$, and a ciphertext $c \in \mathcal{C}$, and outputs an updated state $st' \in \mathcal{S}$ or $\perp$, and a message $m \in \mathcal{M}$ or $\perp$. We write $(st', m) \leftarrow \mathsf{rcv}(st, ad, c)$. If $st' = \perp$ or $m = \perp$ we say the channel* rejects*. We require $st' = \perp$ iff $m = \perp$.*

The rcv algorithm uses symbol '$\perp$' as an explicit error indicator. Note that since $\perp \notin \mathcal{S}$, once rcv outputs $st' = \perp$ our syntax does not allow any further invocation of the snd and rcv algorithms on input 'state' st'. This reflects the reasonable behavior of (cryptographic) applications which, upon being notified of an error, erase all current state information and refuse to process any further input.

We proceed with definitions of correctness. Naturally, unidirectional and bidirectional channels offer different guarantees. We start with unidirected communication.

Correctness of unidirectional channels. Cryptographic unidirectional channels were first studied by BKN [BKN02], and the following definitions are in line with their work. For unidirectional channels we require that if Alice invokes the snd algorithm on a sequence of messages, the resulting ciphertexts are transmitted to Bob without modification (and without changing their order), and Bob plugs the ciphertexts into his rcv algorithm, then Bob recovers the messages that Alice sent. Formally we require that for all sequences $ad_1, \ldots, ad_l \in \mathcal{AD}$ of associated data and all sequences $m_1, \ldots, m_l \in \mathcal{M}$ of messages, if $K \in \mathcal{K}$ and $\mathsf{init}(K)$ outputs (st_A^0, st_B^0), and if $c_1, \ldots, c_l$ and $st_A^1, \ldots, st_A^l$ and $st_B^1, \ldots, st_B^l$ and $m'_1, \ldots, m'_l$ are such that $(st_A^i, c_i) = \mathsf{snd}(st_A^{i-1}, ad_i, m_i)$ and $(st_B^i, m'_i) = \mathsf{rcv}(st_B^{i-1}, ad_i, c_i)$ for all i, then Bob's rcv invocations do not reject and it holds that $(m'_1, \ldots, m'_l) = (m_1, \ldots, m_l)$. A different way to formalize exactly the same is via the FUNC^1 game in Figure 2 (left). Here, an adversary $\mathcal{A}$ schedules any number of send operations for Alice and receive operations for Bob and it wins (lines 14,15,16) if it delivers associated data and ciphertexts in the right order and without modification, but either the channel rejects or Bob recovers a wrong message. Game-internal variables $s, r, h, AD\text{-}C, M$ keep track of this winning condition: s and r are send and receive counters, h is a Boolean flag that indicates whether Bob is still honest (or 'clean', i.e., was not yet exposed to a manipulated or out-of-order ciphertext), and $AD\text{-}C, M$ are associative arrays storing associated data and ciphertexts, and messages, respectively. Observe that once Bob is flagged as exposed (line 19), because of line 13 the adversary cannot win the game any more, meaning no particular behavior of the channel is expected from this moment on.

For any channel Ch and any adversary $\mathcal{A}$ playing in the described game we define the advantage of $\mathcal{A}$ as $\mathbf{Adv}_{\mathsf{Ch}}^{\text{func1}}(\mathcal{A}) = \Pr[\text{FUNC}^1(\mathcal{A}) \Rightarrow 1]$, where the probability is over the choice of $K \in \mathcal{K}$ and over $\mathcal{A}$'s randomness. Throughout this paper we require perfect correctness, i.e., $\mathbf{Adv}_{\mathsf{Ch}}^{\text{func1}}(\mathcal{A}) = 0$ for all $\mathcal{A}$. Under this condition the two above correctness definitions for unidirectional channels are equivalent.

Correctness of bidirectional channels. We define the functionality of bidirectional channels by extending the game based approach from above. The corresponding game FUNC^2 is in Figure 2 (right). The working principles of FUNC^1 and FUNC^2 are quite similar. Besides the fact that in the bidirectional case Alice and Bob have independent send and receive counters, and flags indicating their honesty, the main difference is the update logic of the latter: Recall that in the unidirectional case Bob's h-flag was cleared when he was exposed to an associated-data field or ciphertext that was not authentic, i.e., not generated by Alice. In the bidirectional case, Bob's h-flag is cleared in addition when receiving an (authentic) ciphertext that Alice crafted after her own h-flag was cleared. (The h-flag of Alice is managed correspondingly.) This behavior is implemented, somewhat indirectly, in the FUNC^2 game via the conditional execution of lines 39–41.

For any channel Ch and any adversary $\mathcal{A}$ playing in the described game we define the advantage of $\mathcal{A}$ as $\mathbf{Adv}_{\mathsf{Ch}}^{\text{func2}}(\mathcal{A}) = \Pr[\text{FUNC}^2(\mathcal{A}) \Rightarrow 1]$, where the probability is over the choice of $K \in \mathcal{K}$ and over $\mathcal{A}$'s randomness. Again we require perfect correctness, i.e., $\mathbf{Adv}_{\mathsf{Ch}}^{\text{func2}}(\mathcal{A}) = 0$ for all $\mathcal{A}$.

We finally note that, in line with intuition, constructions of bidirectional channels in particular also serve as unidirectional channels: Alice would only send but never receive (although she could) and Bob would only receive but never send (although he could). In this sense, observe that bidirectional correctness implies unidirectional correctness. (This immediately follows from the specifications of the FUNC games: in FUNC^2 the rcv oracle of Alice would never be invoked, thus flag h_A would remain set throughout the game, thus lines 39–41 would always be executed for Alice; this is precisely the FUNC^1 game.)

4 Security of Bidirectional Channels

We give game based security definitions for bidirectional channels, formalizing two flavors of integrity protection and two flavors of indistinguishability. (See Appendix A on notions for unidirectional channels.) Our notions and naming conventions extend the ones from BKN [BKN02].

Game $\text{FUNC}^1(\mathcal{A})$
00 $s \leftarrow 0$
01 $r \leftarrow 0$
02 $h \leftarrow \text{T}$
03 $AD\text{-}C \leftarrow \emptyset$; $M \leftarrow \emptyset$
04 $(st_A, st_B) \leftarrow_{\$} \mathsf{init}$
05 $\mathcal{A}^{\text{snd,rcv}}$
06 Stop with 0

Oracle $\text{snd}(ad, m)$
07 $(st_A, c) \leftarrow \mathsf{snd}(st_A, ad, m)$
08 $AD\text{-}C[s] \leftarrow (ad, c)$
09 $M[s] \leftarrow m$
10 $s \leftarrow s + 1$
11 Return c

Oracle $\text{rcv}(ad, c)$
12 $(st_B, m) \leftarrow \mathsf{rcv}(st_B, ad, c)$
13 If h:
14 If $r < s \wedge (ad, c) = AD\text{-}C[r]$:
15 If $(st_B, m) = (\bot, \bot) \vee m \neq M[r]$:
16 Stop with 1
17 $r \leftarrow r + 1$
18 Else:
19 $h \leftarrow \text{F}$
20 Return

Game $\text{FUNC}^2(\mathcal{A})$
30 $s_A \leftarrow 0$; $s_B \leftarrow 0$
31 $r_A \leftarrow 0$; $r_B \leftarrow 0$
32 $h_A \leftarrow \text{T}$; $h_B \leftarrow \text{T}$
33 $AD\text{-}C \leftarrow \emptyset$; $M \leftarrow \emptyset$
34 $(st_A, st_B) \leftarrow_{\$} \mathsf{init}$
35 $\mathcal{A}^{\text{snd,rcv}}$
36 Stop with 0

Oracle $\text{snd}(u, ad, m)$
37 $(st_u, c) \leftarrow \mathsf{snd}(st_u, ad, m)$
38 If h_u:
39 $AD\text{-}C[u, s_u] \leftarrow (ad, c)$
40 $M[u, s_u] \leftarrow m$
41 $s_u \leftarrow s_u + 1$
42 Return c

Oracle $\text{rcv}(u, ad, c)$
43 $v \leftarrow \{A, B\} \setminus \{u\}$
44 $(st_u, m) \leftarrow \mathsf{rcv}(st_u, ad, c)$
45 If h_u:
46 If $r_u < s_v \wedge (ad, c) = AD\text{-}C[v, r_u]$:
47 If $(st_u, m) = (\bot, \bot) \vee m \neq M[v, r_u]$:
48 Stop with 1
49 $r_u \leftarrow r_u + 1$
50 Else:
51 $h_u \leftarrow \text{F}$
52 Return

Fig. 2. Functionality game for unidirectional (left) and bidirectional (right) channels. We assume that once an oracle query for a participant results in the participant's state being set to $\bot$, then no further query for that participant is accepted. We further assume $u \in \{A, B\}$, $ad \in \mathcal{AD}$, $m \in \mathcal{M}$, and $c \in \mathcal{C}$ for all such values provided by the adversary.

4.1 Integrity

The first type of integrity that we formalize is INT-2PTXT which ensures the (bidirectional) integrity of plaintexts. The corresponding security experiment is in Figure 3 (left). Plaintext integrity means that the adversary cannot arrange that messages (plaintexts) recovered by the receiving algorithm differ from those priorly fed into the sending algorithm (by the peer). In the game this is tracked via the send and receive counters s_A, s_B, r_A, r_B, and the associative array $AD\text{-}M$. The test that the recovered messages are the right ones (and also the provided associated data is consistent) is in line 13; in case the requirement is violated, the adversary wins (line 16). If the receive algorithm detects a manipulation and decides to torn down the channel, this is explicitly communicated to the adversary (line 12). Unless the adversary manages to let one party accept a forged message, the game terminates indicating a loss (line 05).

The second notion of integrity is INT-2CTXT which ensures the (bidirectional) integrity of ciphertexts. The notion is similar to INT-2PTXT, but the focus is on preventing manipulations of ciphertexts rather than manipulations of messages. The corresponding security experiment is in Figure 3 (right), and the relevant changes are in lines 37 and 43.

For a channel Ch, we define the INT-2PTXT advantage of an adversary $\mathcal{A}$ as $\mathbf{Adv}^{\text{int-2ptxt}}_{\mathsf{Ch}}(\mathcal{A}) = \Pr[\text{INT}^{\text{2ptxt}}(\mathcal{A}) \Rightarrow 1]$ and we define its INT-2CTXT advantage as $\mathbf{Adv}^{\text{int-2ctxt}}_{\mathsf{Ch}}(\mathcal{A}) = \Pr[\text{INT}^{\text{2ctxt}}(\mathcal{A}) \Rightarrow 1]$. The probabilities are over the choice of $K \in \mathcal{K}$ and over $\mathcal{A}$'s randomness. Intuitively, bidirectional channel Ch offers plaintext integrity if $\mathbf{Adv}^{\text{int-2ptxt}}_{\mathsf{Ch}}(\mathcal{A})$ is small for all efficient adversaries $\mathcal{A}$; similarly, it offers ciphertext integrity if $\mathbf{Adv}^{\text{int-2ctxt}}_{\mathsf{Ch}}(\mathcal{A})$ is small for all efficient $\mathcal{A}$.

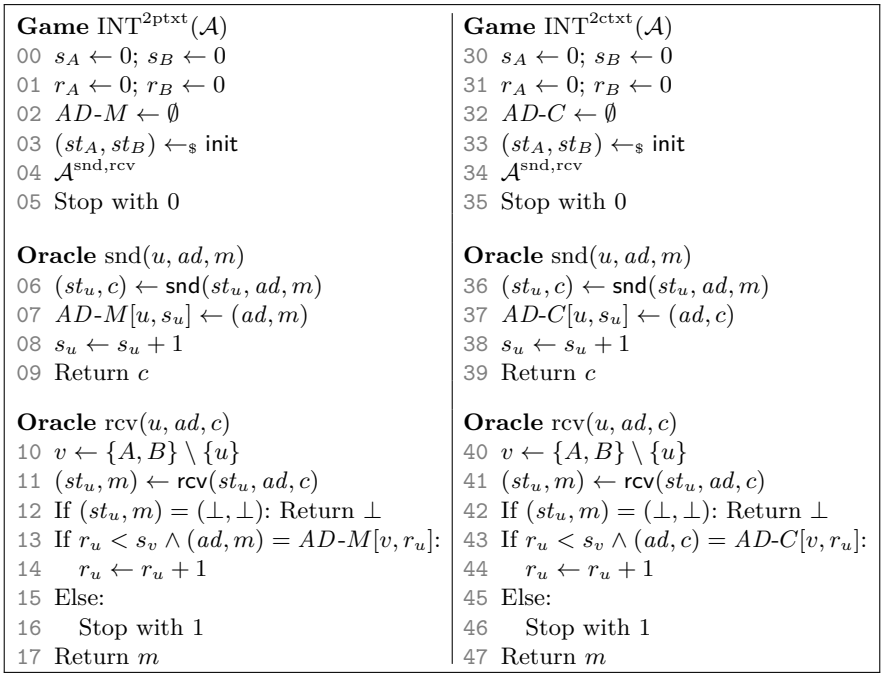

Fig. 3. Games for plaintext integrity (left) and ciphertext integrity (right) for bidirectional channels. We assume that once an oracle query for a participant results in the participant's state being set to $\perp$, then no further query for that participant is accepted. We further assume $u \in \{A, B\}$, $ad \in \mathcal{AD}$, $m \in \mathcal{M}$, and $c \in \mathcal{C}$ for all such values provided by the adversary.

4.2 Confidentiality

We define two confidentiality notions for bidirectional channels: The first, IND-2CPA, models (passive) chosen-plaintext attacks and the second, IND-2CCA, models (active) chosen-ciphertext attacks. For both we give game based definitions. We start with discussing the second notion.

Consider the game for IND-2CCA in Figure 4 (right). The counters s_A, s_B, r_A, r_B, the Boolean flags h_A, h_B, and the associative array $AD\text{-}C$ have the same function as in games FUNC^2 and $\text{INT}^{2\text{ctxt}}$. In particular, the h-flags indicate the cleanliness of participants, i.e., whether they were exposed to non-authentic ciphertexts. From the moment on that a party's h-flag is cleared, ciphertexts created by the party are considered poisoned and their delivery to the peer renders also the latter unclean. As in Figure 2, this logic is implemented via the conditional execution of lines 39,40. Concerning line 49, observe that as long as a participant is clean, the message m recovered in line 43 is equal to the peer's message m^b (from line 37), by the functionality of the channel. Also the adversary knows this, so to disallow trivial attacks, instead of letting the oracle return m, for honest participants the rcv oracle returns the suppression symbol $\diamond$.

Consider next the game for IND-2CPA in Figure 4 (left). The chosen-plaintext setting assumes a passive adversary, i.e., one where participants remain clean. Correspondingly, the game for IND-2CPA is the simplified version of the game for IND-2CCA where $h_A = h_B = \text{T}$ is assumed throughout the execution and the game is aborted if this assumption is violated (line 16).

For a channel Ch, we define the IND-2CPA advantage of an adversary $\mathcal{A}$ as $\mathbf{Adv}^{\text{ind-2cpa}}_{\mathsf{Ch}}(\mathcal{A}) = |\Pr[\text{IND}^{2\text{cpa},1}(\mathcal{A}) \Rightarrow 1] - \Pr[\text{IND}^{2\text{cpa},0}(\mathcal{A}) \Rightarrow 1]|$ and we define its IND-2CCA advantage as $\mathbf{Adv}^{\text{ind-2cca}}_{\mathsf{Ch}}(\mathcal{A}) = |\Pr[\text{IND}^{2\text{cca},1}(\mathcal{A}) \Rightarrow 1] - \Pr[\text{IND}^{2\text{cca},0}(\mathcal{A}) \Rightarrow 1]|$. The probabilities are over the choice of $K \in \mathcal{K}$ and over $\mathcal{A}$'s randomness. Intuitively, bidirectional channel Ch offers confidentiality against passive attacks if $\mathbf{Adv}^{\text{ind-2cpa}}_{\mathsf{Ch}}(\mathcal{A})$ is small for all efficient adversaries $\mathcal{A}$; similarly, it offers confidentiality against active attacks if $\mathbf{Adv}^{\text{ind-2cca}}_{\mathsf{Ch}}(\mathcal{A})$ is small for all efficient $\mathcal{A}$.

We conclude with two technical notes on our definitions.

Note 1. For unidirectional channels, BKN [BKN02] give confidentiality definitions considering passive (CPA) and active (CCA) attacks where the difference between then CPA and CCA security games is

Game $\text{IND}^{\text{2cpa},b}(\mathcal{A})$
00 $s_A \leftarrow 0$; $s_B \leftarrow 0$
01 $r_A \leftarrow 0$; $r_B \leftarrow 0$
02 $AD\text{-}C \leftarrow \emptyset$
03 $(st_A, st_B) \leftarrow_{\$} \mathsf{init}$
04 $b' \leftarrow_{\$} \mathcal{A}^{\text{snd,rcv}}$
05 Stop with b'

Oracle $\text{snd}(u, ad, m^0, m^1)$
06 $(st_u, c) \leftarrow \mathsf{snd}(st_u, ad, m^b)$
07 $AD\text{-}C[u, s_u] \leftarrow (ad, c)$
08 $s_u \leftarrow s_u + 1$
09 Return c

Oracle $\text{rcv}(u, ad, c)$
10 $v \leftarrow \{A, B\} \setminus \{u\}$
11 $(st_u, m) \leftarrow \mathsf{rcv}(st_u, ad, c)$
12 If $(st_u, m) = (\bot, \bot)$: Abort
13 If $r_u < s_v \wedge (ad, c) = AD\text{-}C[v, r_u]$:
14 $\quad r_u \leftarrow r_u + 1$
15 Else:
16 $\quad$ Abort
17 Return $\diamond$

Game $\text{IND}^{\text{2cca},b}(\mathcal{A})$
30 $s_A \leftarrow 0$; $s_B \leftarrow 0$
31 $r_A \leftarrow 0$; $r_B \leftarrow 0$
32 $h_A \leftarrow \text{T}$; $h_B \leftarrow \text{T}$
33 $AD\text{-}C \leftarrow \emptyset$
34 $(st_A, st_B) \leftarrow_{\$} \mathsf{init}$
35 $b' \leftarrow_{\$} \mathcal{A}^{\text{snd,rcv}}$
36 Stop with b'

Oracle $\text{snd}(u, ad, m^0, m^1)$
37 $(st_u, c) \leftarrow \mathsf{snd}(st_u, ad, m^b)$
38 If h_u:
39 $\quad AD\text{-}C[u, s_u] \leftarrow (ad, c)$
40 $\quad s_u \leftarrow s_u + 1$
41 Return c

Oracle $\text{rcv}(u, ad, c)$
42 $v \leftarrow \{A, B\} \setminus \{u\}$
43 $(st_u, m) \leftarrow \mathsf{rcv}(st_u, ad, c)$
44 If $(st_u, m) = (\bot, \bot)$: Return $\bot$
45 If $r_u < s_v \wedge (ad, c) = AD\text{-}C[v, r_u]$:
46 $\quad r_u \leftarrow r_u + 1$
47 Else:
48 $\quad h_u \leftarrow \text{F}$
49 Return $h_u \ ? \diamond : m$

Fig. 4. Games for confidentiality of bidirectional channels against chosen-plaintext (left) and chosen-ciphertext (right) attacks. We assume that once an oracle query for a participant results in the participant's state being set to $\bot$, then no further query for that participant is accepted. We further assume $u \in \{A, B\}$, $ad \in \mathcal{AD}$, $m^0, m^1 \in \mathcal{M}$, and $c \in \mathcal{C}$ for all such values provided by the adversary. We write 'Abort' as an abbreviation for 'Stop with 0'.

precisely the existence of a rcv oracle. This is in line with security definitions for many other encryption primitives (e.g., public key encryption). Our formalizations for bidirectional channels, however, equip the adversary also in the CPA case with a rcv oracle. This discrepancy comes from the fact that in unidirectional channels (and similarly in public key encryption), if ciphertexts are delivered faithfully, the messages obtained by invoking the rcv algorithm are known a priori, namely by the requirement of (perfect) correctness. That is, in these cases the rcv oracle is redundant and can be removed without loss of generality. In contrast, in the setting of bidirectional channels where participants are both senders and receivers, the rcv oracle cannot be removed from the CPA game as it allows the adversary to advance the state of participants in a more general way. Indeed, the following example illustrates that the rcv oracle is indispensable for properly modeling the security of bidirectional channels against passive adversaries: Assume a channel construction in which the first rcv invocation of each participant flips an internal bit in the participant's state that makes all later snd invocations of the participant append vital key material to its ciphertext output. Such a scheme is clearly not secure against passive adversaries, but in a confidentiality model that lacks a rcv oracle the corresponding attack could not be expressed.

Note 2. We comment on a further restriction one might want to impose on the IND^{2cpa} and IND^{2cca} experiments. Most security definitions for stateless or stateful encryption, AEAD, etc. require that the snd oracle aborts if the lengths of m^0 and m^1 do not match (technically, a line saying "If $|m^0| \neq |m^1|$: Abort" would be inserted before lines 06 and 37). This is because most practical encryption schemes do not hide the length of the encrypted message, so if this requirement is not added the games could be distinguished by submitting m^0, m^1 of different lengths. Observe that our understanding of channels assumes an arbitrary abstract message space $\mathcal{M}$ (see Definition 1) which is not required to be a set of strings. As at our level of generality expressions like $|m|$ are not even defined, we did not add them to the games. Clearly, in the moment a specific message space is assumed, e.g., $\mathcal{M} = \{0, 1\}^*$, the corresponding restrictions could, and likely should, be added. As no formal argument in this paper depends on the

presence or absence of such a length check, all claims we make can be adapted to versions of the security notions that have length checks.

4.3 Relations Among Notions

We defined two notions of integrity and two notions of confidentiality. In the following we clarify on three relations between these notions, where the first two are immediate.

INT-2CTXT $\Longrightarrow$ INT-2PTXT. The security requirement that ciphertexts are delivered without modification is stronger than the requirement that plaintexts are. The argument is standard and leverages on the correctness definition: The latter precisely says that if ciphertexts are delivered faithfully, then also messages are transported without modification. That is, whenever the condition in line 43 in Figure 3 is fulfilled, then the condition in line 13 would be fulfilled in particular. We conclude that if no adversary succeeds in reaching line 46 of $\text{INT}^{\text{2ctxt}}$, then also no adversary succeeds in reaching line 16 of $\text{INT}^{\text{2ptxt}}$.

Standard arguments further show that the INT-2CTXT $\Longrightarrow$ INT-2PTXT implication is strict. Observe that if we would relax our syntax and correctness definitions towards allowing randomized rcv algorithms and small correctness errors, the named implication would *not* hold.

IND-2CCA $\Longrightarrow$ IND-2CPA. Also this implication is standard, but the argument does not build on the perfect correctness of the channel. Here the observation is simply that IND-2CPA adversaries are more restricted than IND-2CCA adversaries. In particular, any adversary for game IND^{2cpa} that is run in IND^{2cca} would achieve at least the same advantage.

IND-2CPA + INT-2CTXT $\Longrightarrow$ IND-2CCA. A channel that simultaneously is confidential against passive adversaries (eavesdroppers) and rejects all non-authentic ciphertexts, also provides confidentiality against active adversaries. This statement makes intuition formal: the INT-2CTXT notion degrades active adversaries (that in principle could manipulate ciphertexts on the wire) to passive observers, and for the latter the IND-2CPA notion ensures that nothing is learned about transmitted message contents. Corresponding results are well-known for stateless encryption [BN00] and unidirectional stateful encryption [BKN02].

As the claimed implication does not follow as directly as the relations above, we give a formal proof. Note that, as the theorem statement is in line with intuition, the proof also serves as a confirmation that our definitions of integrity and confidentiality are well chosen.

Theorem 1 (IND-2CPA + INT-2CTXT $\Longrightarrow$ IND-2CCA). *Let* Ch *be a bidirectional channel that offers indistinguishability under chosen-plaintext attacks (IND-2CPA) and integrity of ciphertexts (INT-2CTXT). Then* Ch *also offers indistinguishability under chosen-ciphertext attacks (IND-2CCA). More precisely, for every adversary* $\mathcal{A}$ *there exist adversaries* $\mathcal{B}$ *and* $\mathcal{C}$ *such that*

$$\mathbf{Adv}^{\text{ind-2cca}}_{\mathsf{Ch}}(\mathcal{A}) \leq 2 \cdot \mathbf{Adv}^{\text{int-2ctxt}}_{\mathsf{Ch}}(\mathcal{B}) + \mathbf{Adv}^{\text{ind-2cpa}}_{\mathsf{Ch}}(\mathcal{C}) \enspace .$$

The running times of $\mathcal{B}$ *and* $\mathcal{C}$ *are about that of* $\mathcal{A}$*. Moreover,* $\mathcal{B}$ *poses the same number of* snd *and* rcv *queries as* $\mathcal{A}$*, and* $\mathcal{C}$ *poses the same number of* snd *queries and at most the same number of* rcv *queries as* $\mathcal{A}$*.*

Proof. For $b \in \{0,1\}$ let $\text{G}^{0,b}$ denote the $\text{IND}^{\text{2cca},b}$ game (from Figure 4) for channel Ch against adversary $\mathcal{A}$, and let $\Pr[\text{G}^{0,b}]$ be a shortcut for the probability $\Pr[\text{G}^{0,b} \Rightarrow 1]$. We proceed via game hopping. Let $\text{G}^{1,b}$ be the game derived from $\text{G}^{0,b}$ by replacing the instruction of line 48 with 'Return $\bot$'. The newly added instruction forces termination of the game if the condition of line 45 is not satisfied, i.e., if $\mathcal{A}$ causes participant u to accept a pair (ad, c) that deviates from the sequence of associated data and ciphertexts sent by its peer v. For $b \in \{0,1\}$ let bad^b denote the event that, during an execution of either $\text{G}^{0,b}$ or $\text{G}^{1,b}$, one of the adversary's receiving queries does not fulfill the condition of line 45. As the two games $\text{G}^{0,b}$ and $\text{G}^{1,b}$ execute exactly the same instructions as long as the event bad^b does not occur, we have $\Pr[\text{G}^{0,b} \wedge \neg bad^b] = \Pr[\text{G}^{1,b} \wedge \neg bad^b]$, and thus $|\Pr[\text{G}^{0,b}] - \Pr[\text{G}^{1,b}]| \leq \Pr[bad^b]$.

Now we build two INT adversaries, $\mathcal{B}^0$ and $\mathcal{B}^1$, whose advantages are related to the probability that $\mathcal{A}$ triggers events bad^0 and bad^1, respectively. For $b \in \{0,1\}$, adversary $\mathcal{B}^b$ emulates the left-or-right oracle of the IND game for $\mathcal{A}$, using the snd oracle provided to it by the INT game. More specifically, whenever

$\mathcal{A}$ poses a query $\mathrm{snd}(u, ad, m^0, m^1)$ then $\mathcal{B}^b$ asks $\mathrm{snd}(u, ad, m^b)$ to its own oracle and forwards the answer to $\mathcal{A}$; similarly, $\mathcal{B}^b$ forwards to its own rcv oracle any query $\mathrm{rcv}(u, ad, c)$ that $\mathcal{A}$ poses, and releases the oracle answer only if the condition of line 45 (in Figure 4) is not fulfilled; otherwise it gives back the suppression symbol $\diamond$. Observe that $\mathcal{B}^b$ performs a perfect simulation of the games (both $\mathrm{G}^{0,b}$ and $\mathrm{G}^{1,b}$) as long as event bad^b does not occur; however, if bad^b happens then $\mathcal{B}^b$ breaks ciphertext integrity of Ch (indeed, the event would trigger line 46 from Figure 3), thus $\Pr[bad^b] \leq \mathbf{Adv}^{\text{int-2ctxt}}_{\mathsf{Ch}}(\mathcal{B}^b)$. Consider now an adversary $\mathcal{B}$ which tosses a random coin $d \in \{0,1\}$ and then runs $\mathcal{B}^0$ or $\mathcal{B}^1$ according to the outcome. By construction, $\mathcal{B}$'s advantage is the average of $\mathcal{B}^0$'s and $\mathcal{B}^1$'s advantages: $\mathbf{Adv}^{\text{int-2ctxt}}_{\mathsf{Ch}}(\mathcal{B}) \geq \Pr[bad^0 \wedge d = 0] + \Pr[bad^1 \wedge d = 1] = \frac{1}{2} \cdot \Pr[bad^0] + \frac{1}{2} \cdot \Pr[bad^1]$. We can now derive the following bound for $\mathcal{A}$'s advantage in the original game:

$$\begin{aligned}
\mathbf{Adv}^{\text{ind-2cca}}_{\mathsf{Ch}}(\mathcal{A}) &= \left|\Pr[\mathrm{G}^{0,1}] - \Pr[\mathrm{G}^{0,0}]\right| \\
&\leq \left|\Pr[\mathrm{G}^{0,1}] - \Pr[\mathrm{G}^{1,1}]\right| + \left|\Pr[\mathrm{G}^{1,1}] - \Pr[\mathrm{G}^{1,0}]\right| + \left|\Pr[\mathrm{G}^{1,0}] - \Pr[\mathrm{G}^{0,0}]\right| \\
&\leq \Pr[bad^1] + \left|\Pr[\mathrm{G}^{1,1}] - \Pr[\mathrm{G}^{1,0}]\right| + \Pr[bad^0] \\
&\leq 2 \cdot \mathbf{Adv}^{\text{int-2ctxt}}_{\mathsf{Ch}}(\mathcal{B}) + \left|\Pr[\mathrm{G}^{1,1}] - \Pr[\mathrm{G}^{1,0}]\right| \quad .
\end{aligned}$$

We finally show how to bound the difference in probability between games $\mathrm{G}^{1,1}$ and $\mathrm{G}^{1,0}$ with the IND-2CPA advantage of an adversary $\mathcal{C}$. Briefly, $\mathcal{C}$ relays $\mathcal{A}$'s sending queries to its left-or-right oracle (the snd oracle is essentially the same in the IND-2CPA game and the IND-2CCA game) and registers sent pairs (ad, c) as entries of an associative array $AD\text{-}C$ corresponding to each sending query. Answering $\mathcal{A}$'s queries to rcv is in principle more challenging, since the IND-2CPA game provides $\mathcal{C}$ with a receiving oracle that expects only 'in-sync' queries (i.e., queries (u, ad, c) that match the sequence $AD\text{-}C$ that u's peer sent) while the game $\mathrm{G}^{1,b}$ also allows $\mathcal{A}$ to submit 'out-of-sync' queries and see the corresponding output of rcv on those. However, the modification made in game $\mathrm{G}^{1,b}$ makes the answers of rcv predictable. Indeed, there are only two ways for the receiving oracle to answer $\mathcal{A}$'s queries. If $\mathcal{A}$ poses an 'in-sync' query (i.e., for which the condition of line 45 is fulfilled), the oracle returns the suppression symbol $\diamond$. If instead $\mathcal{A}$ queries rcv with an 'out-of-sync' query, the oracle outputs the rejection symbol $\perp$, because either the query leads to an actual rejection in line 44, or the test of line 45 fails and hence the newly added 'Return $\perp$' instruction is executed in line 48. Given this, algorithm $\mathcal{C}$ can emulate the rcv oracle by simply checking if for each given query (u, ad, c) the condition of line 45 is fulfilled (note that such test can be performed using public information): If it is, $\mathcal{C}$ forwards the query to its own rcv oracle (note that in this case the query is 'in-sync', hence the instructions in lines 12 and 16 do not play a role) and returns $\diamond$ to $\mathcal{A}$; otherwise, it simply returns $\perp$ to $\mathcal{A}$ without querying its oracle. By inspecting the code of games $\mathrm{G}^{1,b}$ we see that $\mathcal{C}$ provides a sound simulation. This leads to the desired bound:

$$\begin{aligned}
\mathbf{Adv}^{\text{ind-2cca}}_{\mathsf{Ch}}(\mathcal{A}) &\leq 2 \cdot \mathbf{Adv}^{\text{int-2ctxt}}_{\mathsf{Ch}}(\mathcal{B}) + \left|\Pr[\mathrm{G}^{1,1}] - \Pr[\mathrm{G}^{1,0}]\right| \\
&\leq 2 \cdot \mathbf{Adv}^{\text{int-2ctxt}}_{\mathsf{Ch}}(\mathcal{B}) + \mathbf{Adv}^{\text{ind-2cpa}}_{\mathsf{Ch}}(\mathcal{C}) \quad .
\end{aligned}$$

□

5 The Canonic Composition

We study a classic construction paradigm that realizes a bidirectional channel from two independent instances of a unidirectional channel operated in opposite directions. Many real-world channel protocols, including SSH and TLS, have been designed with this strategy in mind. Due to its widespread deployment we call the paradigm the *canonic composition* (of two unidirectional channels). As already mentioned in the introduction, unfortunately, most security analyses of channel protocols based on the canonic composition consider security aspects separately for each direction of communication; see [BKN02,JKSS12,KPW13] for some examples. We complete the picture by studying how the security of the canonic composition relates to the security of the underlying (unidirectional) building blocks.

5.1 The Construction

Let $\mathsf{Ch} = (\mathsf{init}, \mathsf{snd}, \mathsf{rcv})$ be a unidirectional channel for associated data space $\mathcal{AD}$ and message space $\mathcal{M}$, with key space $\mathcal{K}$, ciphertext space $\mathcal{C}$, and state space $\mathcal{S}$. The canonic composition paradigm employs two

independent instances of Ch: one protects the communication in the direction from Alice to Bob ($\rightarrow$), the other protects the direction from Bob to Alice ($\leftarrow$).

Let $\mathcal{K}^* = \mathcal{K} \times \mathcal{K}$, $\mathcal{C}^* = \mathcal{C}$, and $\mathcal{S}^* = \mathcal{S} \times \mathcal{S}$, and let $\mathsf{Ch}^* = (\mathsf{init}^*, \mathsf{snd}^*, \mathsf{rcv}^*)$ denote the bidirectional channel obtained from Ch by applying the transform specified in Figure 5. Each instance of Ch^* is seeded with a key of the form $K = (K^{\rightarrow}, K^{\leftarrow})$. The idea is that Alice uses $K^{\rightarrow}$ to send and Bob uses the same key to receive; similarly, Bob uses $K^{\leftarrow}$ to send and Alice uses it to receive. The initialization algorithm init^* thus prepares initial states for Alice and Bob by running init twice, on input keys $K^{\rightarrow}$ and $K^{\leftarrow}$, obtaining state pairs $(st_S^{\rightarrow}, st_R^{\rightarrow})$ and $(st_S^{\leftarrow}, st_R^{\leftarrow})$; it then sets Alice's and Bob's initial states to $st_A = (st_S^{\rightarrow}, st_R^{\leftarrow})$ and $st_B = (st_S^{\leftarrow}, st_R^{\rightarrow})$. When a party wishes to send a message m, it extracts from its state $st = (st_S, st_R)$ the part st_S dedicated to sending and invokes (unidirectional) algorithm snd on input st_S and m. Similarly, for processing a ciphertext c, the party extracts part st_R from its state and invokes rcv on input st_R and c, in order to recover m. Importantly, if the latter operation rejects outputting $st_R = \bot$, also the rcv^* algorithm outputs $st = \bot$.

Proc $\mathsf{init}^*(K)$	**Proc** $\mathsf{snd}^*(st, ad, m)$	**Proc** $\mathsf{rcv}^*(st, ad, c)$
00 $(K^{\rightarrow}, K^{\leftarrow}) \leftarrow K$	06 $(st_S, st_R) \leftarrow st$	10 $(st_S, st_R) \leftarrow st$
01 $(st_S^{\rightarrow}, st_R^{\rightarrow}) \leftarrow \mathsf{init}(K^{\rightarrow})$	07 $(st_S, c) \leftarrow \mathsf{snd}(st_S, ad, m)$	11 $(st_R, m) \leftarrow \mathsf{rcv}(st_R, ad, c)$
02 $(st_S^{\leftarrow}, st_R^{\leftarrow}) \leftarrow \mathsf{init}(K^{\leftarrow})$	08 $st \leftarrow (st_S, st_R)$	12 If $(st_R, m) \neq \bot$:
03 $st_A \leftarrow (st_S^{\rightarrow}, st_R^{\leftarrow})$	09 Return (st, c)	13 $st \leftarrow (st_S, st_R)$
04 $st_B \leftarrow (st_S^{\leftarrow}, st_R^{\rightarrow})$		14 Else:
05 Return (st_A, st_B)		15 $st \leftarrow \bot$
		16 Return (st, m)

Fig. 5. A bidirectional channel $\mathsf{Ch}^* = (\mathsf{init}^*, \mathsf{snd}^*, \mathsf{rcv}^*)$ built from a unidirectional channel $\mathsf{Ch} = (\mathsf{init}, \mathsf{snd}, \mathsf{rcv})$ using the canonic composition paradigm.

5.2 Security Analysis

For a unidirectional channel Ch and the bidirectional channel Ch^* obtained from it via the canonic composition, we investigate the relationship between the security of Ch and the security of Ch^*. More concretely, for the (unidirectional) security notions INT-1PTXT, INT-1CTXT, IND-1CPA, and IND-1CCA (see Appendix A) that channel Ch might achieve, we study whether channel Ch^* achieves the corresponding bidirectional notions. In a nutshell our results are as follows:

$$\begin{array}{ccccc} \overbrace{\text{INT-1PTXT}}^{\text{direction } \rightarrow} & + & \overbrace{\text{INT-1PTXT}}^{\text{direction } \leftarrow} & \Longrightarrow & \overbrace{\text{INT-2PTXT}}^{\text{directions } \overrightarrow{\leftarrow}} \\ \text{INT-1CTXT} & + & \text{INT-1CTXT} & \Longrightarrow & \text{INT-2CTXT} \\ \text{IND-1CPA} & + & \text{IND-1CPA} & \Longrightarrow & \text{IND-2CPA} \\ \text{IND-1CCA} & + & \text{IND-1CCA} & \not\Longrightarrow & \text{IND-2CCA} \end{array}$$

That is, while channel Ch^* inherits guarantees on plaintext integrity, ciphertext integrity, and confidentiality against passive attacks from Ch, for confidentiality against active attacks a similar implication does not hold. We provide formal statements and proofs for the above implications and separations in Theorems 2–5.

Theorem 2 (Integrity of plaintexts). *If* Ch *offers integrity of plaintexts (INT-1PTXT) then also* Ch^* *offers integrity of plaintexts (INT-2PTXT). More precisely, for every adversary* $\mathcal{A}$ *against* Ch^* *there exists an adversary* $\mathcal{B}$ *against* Ch *such that*

$$\mathbf{Adv}^{\text{int-2ptxt}}_{\mathsf{Ch}^*}(\mathcal{A}) \leq 2 \cdot \mathbf{Adv}^{\text{int-1ptxt}}_{\mathsf{Ch}}(\mathcal{B}) \ .$$

The running time of $\mathcal{B}$ *is about that of* $\mathcal{A}$ *plus the time to run* init, snd *and* rcv *to answer all of* $\mathcal{A}$*'s queries in one direction. Moreover,* $\mathcal{B}$ *poses at most the same number of* snd *and* rcv *queries as* $\mathcal{A}$*.*

Proof. The overall idea is that any integrity violation (in the bidirectional sense) in Ch^* translates to an integrity violation (in the unidirectional sense) in one of the two instances of Ch, either in the direction from Alice to Bob ($\rightarrow$) or in the direction from Bob to Alice ($\leftarrow$). Let G^0 denote the $\mathrm{INT}^{\mathrm{2ptxt}}$ game (from Figure 3) for channel Ch^* against adversary $\mathcal{A}$, and let $\Pr[\mathrm{G}^0]$ be a shortcut for the probability $\Pr[\mathrm{G}^0 \Rightarrow 1]$. Let $bad^{\rightarrow}$ (respectively, $bad^{\leftarrow}$) be the event that $\mathcal{A}$ causes termination of G^0 with output 1 by posing a query of the form $\mathrm{rcv}(B,\cdot,\cdot)$ (resp., $\mathrm{rcv}(A,\cdot,\cdot)$). We proceed with game hopping. We define game G^1 by modifying game G^0 by enforcing termination with output 0 (making $\mathcal{A}$ lose) whenever event $bad^{\rightarrow}$ occurs. Similarly, we define game G^2 as the modification of G^1 that enforces termination with output 0 if event $bad^{\leftarrow}$ occurs. Games G^0 and G^1 execute the same instructions as long as event $bad^{\rightarrow}$ does not occur, and G^1 and G^2 execute the same instructions as long as event $bad^{\leftarrow}$ does not occur. We thus have $|\Pr[\mathrm{G}^0] - \Pr[\mathrm{G}^1]| \leq \Pr[bad^{\rightarrow}]$ and $|\Pr[\mathrm{G}^1] - \Pr[\mathrm{G}^2]| \leq \Pr[bad^{\leftarrow}]$. Further we have $\Pr[\mathrm{G}^2] = 0$, leading to the inequality

$$\begin{aligned}\mathbf{Adv}^{\text{int-2ptxt}}_{\mathsf{Ch}^*}(\mathcal{A}) &\leq \big|\Pr[\mathrm{G}^0] - \Pr[\mathrm{G}^1]\big| + \big|\Pr[\mathrm{G}^1] - \Pr[\mathrm{G}^2]\big| + \Pr[\mathrm{G}^2] \\ &\leq \Pr[bad^{\rightarrow}] + \Pr[bad^{\leftarrow}] + 0 \ .\end{aligned}$$

It remains to bound the probabilities of events $bad^{\rightarrow}$ and $bad^{\leftarrow}$. Intuitively, if $bad^{\rightarrow}$ occurs $\mathcal{A}$ wins against the integrity game of the Alice-to-Bob direction, and analogously for event $bad^{\leftarrow}$. Correspondingly, from $\mathcal{A}$ we construct adversaries $\mathcal{B}^{\rightarrow}, \mathcal{B}^{\leftarrow}$ against the INT-1PTXT security of channel Ch that achieve advantages $\Pr[bad^{\rightarrow}]$ and $\Pr[bad^{\rightarrow}]$, respectively. Briefly, $\mathcal{B}^{\rightarrow}$ answers $\mathcal{A}$'s Bob-to-Alice send and receive queries by picking a key $\widetilde{K} \leftarrow_{\$} \mathcal{K}$, initializing fresh states $(\widetilde{st_S}, \widetilde{st_R}) \leftarrow \mathsf{init}(\widetilde{K})$, and thus running its own instance of channel Ch in the Bob-to-Alice direction. Further, it answers $\mathcal{A}$'s Alice-to-Bob queries (of the form $\mathrm{snd}(A,\cdot,\cdot)$ and $\mathrm{rcv}(B,\cdot,\cdot)$) by relaying them to its own oracles, provided by the INT-1PTXT game. Now, observe that if $\mathcal{A}$ triggers event $bad^{\rightarrow}$ then in the INT-1PTXT game (from Figure 7), line 14 would be executed, thus causing $\mathcal{B}^{\rightarrow}$ to win the integrity game against the unidirectional channel Ch. By using a similar argument in the opposite direction we obtain a reduction $\mathcal{B}^{\leftarrow}$ that wins the INT-1PTXT game as soon as event $bad^{\leftarrow}$ happens. Finally, let $\mathcal{B}$ be an adversary that tosses a random coin $d \in \{0,1\}$ and runs $\mathcal{B}^{\rightarrow}$ if $d = 0$ and $\mathcal{B}^{\leftarrow}$ if $d = 1$. By construction we have $\mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{B}) \geq \frac{1}{2} \cdot \mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{B}^{\rightarrow}) + \frac{1}{2} \cdot \mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{B}^{\leftarrow})$, from which the claimed inequality follows. □

The above proof strategy is easily adapted to show that also the ciphertext integrity of channel Ch can be lifted to that of channel Ch^*. We thus omit an explicit proof and just give the theorem statement.

Theorem 3 (Integrity of ciphertexts). *If* Ch *offers integrity of ciphertexts (INT-1CTXT) then also* Ch^* *offers integrity of ciphertexts (INT-2CTXT). More precisely, for every adversary* $\mathcal{A}$ *against* Ch^* *there exists an adversary* $\mathcal{B}$ *against* Ch *such that*

$$\mathbf{Adv}^{\text{int-2ctxt}}_{\mathsf{Ch}^*}(\mathcal{A}) \leq 2 \cdot \mathbf{Adv}^{\text{int-1ctxt}}_{\mathsf{Ch}}(\mathcal{B}) \ .$$

The running time of $\mathcal{B}$ *is about that of* $\mathcal{A}$ *plus the time to run* init, snd *and* rcv *to answer all of* $\mathcal{A}$*'s queries in one direction. Moreover,* $\mathcal{B}$ *poses at most the same number of* snd *and* rcv *queries as* $\mathcal{A}$*.*

Theorem 4 (Confidentiality against passive adversaries). *If* Ch *offers indistinguishability against chosen-plaintext attacks (IND-1CPA) then also* Ch^* *offers indistinguishability against chosen-plaintext attacks (IND-2CPA). More precisely, for every adversary* $\mathcal{A}$ *against* Ch^* *there exists an adversary* $\mathcal{B}$ *against* Ch *such that*

$$\mathbf{Adv}^{\text{ind-2cpa}}_{\mathsf{Ch}^*}(\mathcal{A}) \leq 2 \cdot \mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{B}) \ .$$

The running time of $\mathcal{B}$ *is about that of* $\mathcal{A}$ *plus the time to run* init, snd *and* rcv *to answer all of* $\mathcal{A}$*'s queries in one direction. Moreover,* $\mathcal{B}$ *poses at most the same number of* snd *and* rcv *queries as* $\mathcal{A}$*.*

Proof. We prove the theorem statement with an argument similar to the one used in the proof of Theorem 2: If adversary $\mathcal{A}$ asks only in-sync queries (i.e., never triggers the execution of lines 12,16 in Figure 4) then, by looking at each communication direction individually, $\mathcal{A}$'s queries are also in-sync according to the unidirectional IND-1CPA game (i.e., no query triggers the execution of lines 10,14 in Figure 8). To formalize this intuition we define some intermediate games. The first game, that we denote by $\mathrm{G}^{0,0}$, is the game $\mathrm{IND}^{\mathrm{2cpa},0}(\mathcal{A})$ from Figure 4. As in the previous proofs, we use the shortcut $\Pr[\mathrm{G}^{0,0}]$ for the

probability $\Pr[\mathrm{G}^{0,0} \Rightarrow 1]$. Now define $\mathrm{G}^{0,1}$ from $\mathrm{G}^{0,0}$ by modifying the left-or-right oracle as follows: Whenever $\mathcal{A}$ poses a query $\mathrm{snd}(A, ad, m^0, m^1)$, invoke algorithm snd on message m^0 (as in the original game); if the query is $\mathrm{snd}(B, ad, m^0, m^1)$, invoke algorithm snd on message m^1. In other words, $\mathrm{G}^{0,1}$ selects the 'left' message if the sender is Alice while it sends the 'right' message if the sender is Bob. Finally, derive $\mathrm{G}^{1,1}$ from game $\mathrm{G}^{0,1}$ by letting the left-or-right oracle always select message m^1. Note that $\mathrm{G}^{1,1} = \mathrm{IND}^{2\mathrm{cpa},1}(\mathcal{A})$. Given the games we can bound $\mathcal{A}$'s advantage in the original game as follows:

$$\mathbf{Adv}^{\text{ind-2cpa}}_{\mathsf{Ch}^*}(\mathcal{A}) \leq |\Pr[\mathrm{G}^{1,1}] - \Pr[\mathrm{G}^{0,1}]| + |\Pr[\mathrm{G}^{0,1}] - \Pr[\mathrm{G}^{0,0}]| \ .$$

We show next that the difference in probability between games $\mathrm{G}^{1,1}$ and $\mathrm{G}^{0,1}$, and between games $\mathrm{G}^{0,1}$ and $\mathrm{G}^{0,0}$, can be upper bounded by the IND-1CPA advantage of two adversaries $\mathcal{B}^{\rightarrow}$ and $\mathcal{B}^{\leftarrow}$ against the unidirectional channel Ch. Note that either of the above combinations of games fixes one of the two selection bits. For instance, both games $\mathrm{G}^{1,1}$ and $\mathrm{G}^{0,1}$ make Bob send the 'left' message. This combination of games implicitly defines a new indistinguishability game $\mathrm{G}^{b,1}$ where $\mathcal{A}$ has to tell apart $\mathrm{G}^{1,1}$ and $\mathrm{G}^{0,1}$. In fact, the latter observation is the basic working principle of the reduction $\mathcal{B}^{\rightarrow}$, which answers $\mathcal{A}$'s queries in the direction '$\rightarrow$' using the oracles provided by the IND-1CPA game against channel Ch, and answers the queries in the direction '$\leftarrow$' by running an independent instance of channel Ch. It is immediate to see that $\mathcal{B}^{\rightarrow}$ provides a perfect simulation of game $\mathrm{G}^{b,1}$. To bound $\mathcal{A}$'s distinguishing advantage in game $\mathrm{G}^{b,1}$ with $\mathcal{B}^{\rightarrow}$'s advantage it suffices to show that if all of $\mathcal{A}$'s queries are in-sync, i.e., do not cause premature termination of the game, then the corresponding queries that $\mathcal{B}^{\rightarrow}$ poses in the outer IND-1CPA game are in-sync, too. Let $q = (u, ad, c)$ be any of $\mathcal{A}$'s receiving queries and suppose that q does not trigger the execution of lines 12,16 (in Figure 4). If $u = B$ there is nothing to show: $\mathcal{B}^{\rightarrow}$ answers the query on its own by invoking algorithm rcv (recall that $\mathcal{B}^{\rightarrow}$ runs an independent instance of Ch, so in particular it is in control of the states st_A and st_B for this instance). In the opposite case, i.e., $u = A$, adversary $\mathcal{B}^{\rightarrow}$ asks a query $\mathrm{rcv}(ad, c)$ to its own receiving oracle, and this may in principle cause abrupt termination of the game. However, by inspection of the IND-1CPA game it is immediate to see that this is not the case, as any out-of-sync query would also be considered out-of-sync in the IND-2CPA game from Figure 4. This allows us to derive the bound $|\Pr[\mathrm{G}^{1,1}] - \Pr[\mathrm{G}^{0,1}]| \leq \mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{B}^{\rightarrow})$. Using a similar strategy we can construct a reduction $\mathcal{B}^{\leftarrow}$ which, symmetrically to $\mathcal{B}^{\rightarrow}$, emulates game $\mathrm{G}^{0,b}$ using the oracles provided by the IND-1CPA game and maintaining its own instance of the unidirectional channel Ch in the direction Alice-to-Bob ($\rightarrow$), and attacks the unidirectional channel Ch in the Bob-to-Alice direction ($\leftarrow$). This leads to a second inequality: $|\Pr[\mathrm{G}^{0,1}] - \Pr[\mathrm{G}^{0,0}]| \leq \mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{B}^{\leftarrow})$. Now consider an IND-adversary $\mathcal{B}$ which tosses a random coin $d \in \{0, 1\}$ and runs $\mathcal{B}^{\rightarrow}$ if $d = 0$ and $\mathcal{B}^{\leftarrow}$ if $d = 1$. We obtain $\mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{B}) \geq \frac{1}{2} \cdot \mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{B}^{\rightarrow}) + \frac{1}{2} \cdot \mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{B}^{\leftarrow})$, which implies the claimed bound. □

To prove the next theorem we revisit the attack from Figure 1 in light of our formalisms, showing a successful chosen-ciphertext attack (IND-2CCA) against the canonic composition of two instances of a unidirectional channel which is indistinguishable against chosen-ciphertext attacks (IND-1CCA).

Theorem 5 (No confidentiality against active adversaries). *If IND-1CCA secure unidirectional channels exist, then there exists one such channel* Ch *such that its canonic composition* Ch^* *is not IND-2CCA secure. More precisely, there exists an efficient adversary* $\mathcal{A}$ *that breaks the confidentiality of* Ch^* *achieving* $\mathbf{Adv}^{\text{ind-2cca}}_{\mathsf{Ch}^*}(\mathcal{A}) = 1$.

Proof. We prove the statement in two steps. We first argue that an IND-1CCA secure unidirectional channel Ch exists where the rcv algorithm never rejects an incoming ciphertext but always outputs a message. We then show that the canonic composition Ch^* of two instances of Ch is not IND-2CCA secure.

Let Ch' be any IND-1CCA secure unidirectional channel. Construct $\mathsf{Ch} = (\mathsf{init}, \mathsf{snd}, \mathsf{rcv})$ from $\mathsf{Ch}' = (\mathsf{init}', \mathsf{snd}', \mathsf{rcv}')$ by having init and init' be the same algorithms, snd and snd' be the same algorithms, and having rcv such that (1) when rcv' outputs a message m then also rcv outputs m, and (2) when rcv' rejects then rcv switches to a mode where on each invocation it outputs an a priori fixed message $\tilde{m} \in \mathcal{M}$. Clearly, if Ch' is IND-1CCA secure, then so is Ch. (Of course Ch does not offer any reasonable kind of integrity, but this does not contradict its IND-1CCA security.)

Consider now the canonic composition Ch^* of two instances of Ch. We describe an adversary $\mathcal{A}$ against the IND-2CCA security of Ch^* that achieves an advantage of 1. As the attack does not rely on the associated data input of the snd and rcv algorithms, for simplicity in the following we do not

annotate it. Adversary $\mathcal{A}$ fixes an arbitrary ciphertext $\tilde{c}$ and two messages $m^0 \neq m^1$. It then poses three queries: (1) a query $\mathrm{rcv}(A, \tilde{c})$, which makes Alice output message $\tilde{m}$ (that plays no further role in the attack), (2) a query $\mathrm{snd}(A, m^0, m^1)$, which makes Alice produce a ciphertext c for either m^0 or m^1, and (3) a query $\mathrm{rcv}(B, c)$, which asks Bob for a decryption of c. Adversary $\mathcal{A}$ outputs $b' = 1$ if Bob answers with m^1; otherwise it outputs $b' = 0$. See Figure 6 for an illustration of the attack.

We analyze $\mathcal{A}$'s advantage as follows: By the rules of the IND-2CCA experiment (see Figure 4), query (1) is identified as active (Alice receives although nothing has been sent by Bob; formally, condition $r_A < s_B$ from line 45 is not satisfied) and $h_A \leftarrow \mathrm{F}$ is set in line 48. This means that query (2) does not increase counter s_A in line 40. Thus also query (3) is identified as active (because condition $r_B < s_A$ is not satisfied), thus $h_B \leftarrow \mathrm{F}$ is set in line 48 and the oracle returns c's decryption m^b in line 49. This allows for recovering bit b with probability 1. □

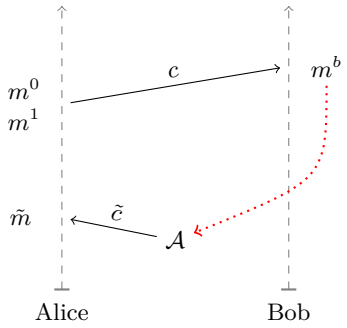

Fig. 6. An IND-2CCA attack against a bidirectional channel Ch^* obtained from an IND-1CCA secure unidirectional channel Ch via the canonic composition. In the figure time evolves bottom-up (dashed lines).

5.3 The Canonic Composition in Practice

In Theorem 5 we saw that operating two IND-CCA secure unidirectional channels in opposite directions does not necessarily result in one IND-CCA secure bidirectional channel. What are the implications of this on real-world protocols like TLS and SSH? Fortunately, combining results from prior work, our results from Section 4, and the results from the current section, leads to the conclusion that the above mentioned protocols *do* achieve the strongest security notions proposed in this paper. The sequence of argumentation is as follows.

Works like [Kra01,BKN02,PRS11,BMM+15] on the security of SSH and TLS indicate that unidirectional versions of the latter fulfill the notion of stateful authenticated encryption, or, in our words, INT-1PTXT, INT-1CTXT, IND-1CPA, and IND-1CCA (see Appendix A). Thus, by Theorems 3 and 4, the full (bidirectional) protocols achieve INT-2CTXT and IND-2CPA security. Finally, Theorem 1 tells us that SSH and TLS achieve IND-2CCA security. The following corollary makes this formal.

Corollary 1. *If* Ch *offers unidirectional integrity of ciphertexts (INT-1CTXT) and indistinguishability against chosen-plaintext attacks (IND-1CPA) then* Ch^* *offers bidirectional integrity of ciphertexts (INT-2CTXT) and bidirectional indistinguishability against chosen-ciphertext attacks (IND-2CCA).*

Acknowledgments

We thank Marc Fischlin, Felix Günther, Martijn Stam, and anonymous reviewers for helpful discussions and insightful comments on early versions of this work. Giorgia Marson was supported by the DFG as part of projects P2 within the CRC 1119 CROSSING and the research training group 1817/1 Ubicrypt. Bertram Poettering was supported by the ERC Project ERCC (FP7/615074).

References

BDPS12. Alexandra Boldyreva, Jean Paul Degabriele, Kenneth G. Paterson, and Martijn Stam. Security of symmetric encryption in the presence of ciphertext fragmentation. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 682–699, Cambridge, UK, April 15–19, 2012. Springer, Heidelberg, Germany.

BDPS14. Alexandra Boldyreva, Jean Paul Degabriele, Kenneth G. Paterson, and Martijn Stam. On symmetric encryption with distinguishable decryption failures. In Shiho Moriai, editor, *FSE 2013*, volume 8424 of *LNCS*, pages 367–390, Singapore, March 11–13, 2014. Springer, Heidelberg, Germany.

BKN02. Mihir Bellare, Tadayoshi Kohno, and Chanathip Namprempre. Authenticated encryption in SSH: Provably fixing the SSH binary packet protocol. In Vijayalakshmi Atluri, editor, *ACM CCS 02*, pages 1–11, Washington D.C., USA, November 18–22, 2002. ACM Press.

BMM+15. Christian Badertscher, Christian Matt, Ueli Maurer, Phillip Rogaway, and Björn Tackmann. Augmented secure channels and the goal of the TLS 1.3 record layer. In Man Ho Au and Atsuko Miyaji, editors, *ProvSec 2015*, volume 9451 of *LNCS*, pages 85–104, Kanazawa, Japan, November 24–26, 2015. Springer, Heidelberg, Germany.

BN00. Mihir Bellare and Chanathip Namprempre. Authenticated encryption: Relations among notions and analysis of the generic composition paradigm. In Tatsuaki Okamoto, editor, *ASIACRYPT 2000*, volume 1976 of *LNCS*, pages 531–545, Kyoto, Japan, December 3–7, 2000. Springer, Heidelberg, Germany.

CK01. Ran Canetti and Hugo Krawczyk. Analysis of key-exchange protocols and their use for building secure channels. In Birgit Pfitzmann, editor, *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 453–474, Innsbruck, Austria, May 6–10, 2001. Springer, Heidelberg, Germany.

CK02. 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, Amsterdam, The Netherlands, April 28 – May 2, 2002. Springer, Heidelberg, Germany.

DR08. T. Dierks and E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246 (Proposed Standard), August 2008. Updated by RFCs 5746, 5878, 6176.

FGMP15. Marc Fischlin, Felix Günther, Giorgia Azzurra Marson, and Kenneth G. Paterson. Data is a stream: Security of stream-based channels. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015, Part II*, volume 9216 of *LNCS*, pages 545–564, Santa Barbara, CA, USA, August 16–20, 2015. Springer, Heidelberg, Germany.

JKSS12. Tibor Jager, Florian Kohlar, Sven Schäge, and Jörg Schwenk. On the security of TLS-DHE in the standard model. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 273–293, Santa Barbara, CA, USA, August 19–23, 2012. Springer, Heidelberg, Germany.

KPB03. Tadayoshi Kohno, Adriana Palacio, and John Black. Building secure cryptographic transforms, or how to encrypt and MAC. Cryptology ePrint Archive, Report 2003/177, 2003. http://eprint.iacr.org/2003/177.

KPW13. Hugo Krawczyk, Kenneth G. Paterson, and Hoeteck Wee. On the security of the TLS protocol: A systematic analysis. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 429–448, Santa Barbara, CA, USA, August 18–22, 2013. Springer, Heidelberg, Germany.

Kra01. Hugo Krawczyk. The order of encryption and authentication for protecting communications (or: How secure is SSL?). In Joe Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 310–331, Santa Barbara, CA, USA, August 19–23, 2001. Springer, Heidelberg, Germany.

MRT12. 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, Taormina, Sicily, Italy, March 19–21, 2012. Springer, Heidelberg, Germany.

Nam02. Chanathip Namprempre. Secure channels based on authenticated encryption schemes: A simple characterization. In Yuliang Zheng, editor, *ASIACRYPT 2002*, volume 2501 of *LNCS*, pages 515–532, Queenstown, New Zealand, December 1–5, 2002. Springer, Heidelberg, Germany.

PRS11. Kenneth G. Paterson, Thomas Ristenpart, and Thomas Shrimpton. Tag size does matter: Attacks and proofs for the TLS record protocol. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 372–389, Seoul, South Korea, December 4–8, 2011. Springer, Heidelberg, Germany.

Rog02. Phillip Rogaway. Authenticated-encryption with associated-data. In Vijayalakshmi Atluri, editor, *ACM CCS 02*, pages 98–107, Washington D.C., USA, November 18–22, 2002. ACM Press.

YL06. T. Ylonen and C. Lonvick. The Secure Shell (SSH) Protocol Architecture. RFC 4251 (Proposed Standard), January 2006.

A Unidirectional Channels

In the cryptographic literature, secure channels are often modeled as stateful encryption primitives (as in [Nam02,BKN02,KPB03], to just name a few). Importantly for this paper, these channel models consider a restricted scenario in which one party only sends and the other only receives, thus providing a *unidirectional* channel. Our syntax in Definition 1 contains that of stateful (authenticated) encryption as a special case. For completeness, in this section we reproduce some established security definitions for unidirectional channels. Precisely, we translate the ideas of BKN [BKN02] to our notation.

We indicate unidirectional flavors of security notions by prefixing their name with a "1", obtaining integrity of plaintexts (INT-1PTXT), integrity of ciphertexts (INT-1CTXT), indistinguishability against chosen-plaintext attacks (IND-1CPA), and indistinguishability against chosen-ciphertext attacks (IND-1CCA).

Consider the integrity games in Figure 7. For a channel Ch, we define the INT-1PTXT advantage of an adversary $\mathcal{A}$ as $\mathbf{Adv}^{\text{int-1ptxt}}_{\mathsf{Ch}}(\mathcal{A}) = \Pr[\text{INT}^{\text{1ptxt}}(\mathcal{A}) \Rightarrow 1]$ and we define its INT-1CTXT advantage as $\mathbf{Adv}^{\text{int-1ctxt}}_{\mathsf{Ch}}(\mathcal{A}) = \Pr[\text{INT}^{\text{1ctxt}}(\mathcal{A}) \Rightarrow 1]$. The probabilities are over the choice of $K \in \mathcal{K}$ and over $\mathcal{A}$'s randomness. Intuitively, unidirectional channel Ch offers plaintext integrity if $\mathbf{Adv}^{\text{int-1ptxt}}_{\mathsf{Ch}}(\mathcal{A})$ is small for all efficient adversaries $\mathcal{A}$; similarly, it offers ciphertext integrity if $\mathbf{Adv}^{\text{int-1ctxt}}_{\mathsf{Ch}}(\mathcal{A})$ is small for all efficient $\mathcal{A}$.

Game $\text{INT}^{\text{1ptxt}}(\mathcal{A})$	**Game** $\text{INT}^{\text{1ctxt}}(\mathcal{A})$
00 $s \leftarrow 0$; $r \leftarrow 0$	30 $s \leftarrow 0$; $r \leftarrow 0$
01 $AD\text{-}M \leftarrow \emptyset$	31 $AD\text{-}C \leftarrow \emptyset$
02 $(st_S, st_R) \leftarrow_{\$} \mathsf{init}$	32 $(st_S, st_R) \leftarrow_{\$} \mathsf{init}$
03 $\mathcal{A}^{\text{snd,rcv}}$	33 $\mathcal{A}^{\text{snd,rcv}}$
04 Stop with 0	34 Stop with 0
Oracle $\text{snd}(ad, m)$	**Oracle** $\text{snd}(ad, m)$
05 $(st_S, c) \leftarrow \mathsf{snd}(st_S, ad, m)$	35 $(st_S, c) \leftarrow \mathsf{snd}(st_S, ad, m)$
06 $AD\text{-}M[s] \leftarrow (ad, m)$	36 $AD\text{-}C[s] \leftarrow (ad, c)$
07 $s \leftarrow s + 1$	37 $s \leftarrow s + 1$
08 Return c	38 Return c
Oracle $\text{rcv}(ad, c)$	**Oracle** $\text{rcv}(ad, c)$
09 $(st_R, m) \leftarrow \mathsf{rcv}(st_R, ad, c)$	39 $(st_R, m) \leftarrow \mathsf{rcv}(st_R, ad, c)$
10 If $(st_R, m) = (\bot, \bot)$: Return $\bot$	40 If $(st_R, m) = (\bot, \bot)$: Return $\bot$
11 If $r < s \wedge (ad, m) = AD\text{-}M[r]$:	41 If $r < s \wedge (ad, c) = AD\text{-}C[r]$:
12 $r \leftarrow r + 1$	42 $r \leftarrow r + 1$
13 Else:	43 Else:
14 Stop with 1	44 Stop with 1
15 Return m	45 Return m

Fig. 7. Games for plaintext integrity (left) and ciphertext integrity (right) for unidirectional channels. (See also Figure 3 for an explanation of the notation.)

Consider next the confidentiality games in Figure 8. We define the IND-1CPA advantage of an adversary $\mathcal{A}$ as $\mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{A}) = |\Pr[\text{IND}^{\text{1cpa},1}(\mathcal{A}) \Rightarrow 1] - \Pr[\text{IND}^{\text{1cpa},0}(\mathcal{A}) \Rightarrow 1]|$ and we define its IND-1CCA advantage as $\mathbf{Adv}^{\text{ind-1cca}}_{\mathsf{Ch}}(\mathcal{A}) = |\Pr[\text{IND}^{\text{1cca},1}(\mathcal{A}) \Rightarrow 1] - \Pr[\text{IND}^{\text{1cca},0}(\mathcal{A}) \Rightarrow 1]|$. The probabilities are over the choice of $K \in \mathcal{K}$ and over $\mathcal{A}$'s randomness. Intuitively, unidirectional channel Ch offers confidentiality against passive attacks if $\mathbf{Adv}^{\text{ind-1cpa}}_{\mathsf{Ch}}(\mathcal{A})$ is small for all efficient adversaries $\mathcal{A}$; similarly, it offers confidentiality against active attacks if $\mathbf{Adv}^{\text{ind-1cca}}_{\mathsf{Ch}}(\mathcal{A})$ is small for all efficient $\mathcal{A}$.

B Unidirectional Security ≠ Bidirectional Security

As already discussed in the introduction, a naive way to define channel security in the bidirectional setting could be to require that both directions are protected independently of each other (each in a unidirectional sense). A bidirectional channel offering this kind of security would, when used as a

Game $\mathrm{IND}^{\mathrm{1cpa},b}(\mathcal{A})$	**Game** $\mathrm{IND}^{\mathrm{1cca},b}(\mathcal{A})$
00 $s \leftarrow 0$; $r \leftarrow 0$	30 $s \leftarrow 0$; $r \leftarrow 0$; $h \leftarrow \mathrm{T}$
01 $AD\text{-}C \leftarrow \emptyset$	31 $AD\text{-}C \leftarrow \emptyset$
02 $(st_S, st_R) \leftarrow_{\$} \mathsf{init}$	32 $(st_S, st_R) \leftarrow_{\$} \mathsf{init}$
03 $b' \leftarrow_{\$} \mathcal{A}^{\mathrm{snd,rcv}}$	33 $b' \leftarrow_{\$} \mathcal{A}^{\mathrm{snd,rcv}}$
04 Stop with b'	34 Stop with b'
Oracle $\mathrm{snd}(ad, m^0, m^1)$	**Oracle** $\mathrm{snd}(ad, m^0, m^1)$
05 $(st_S, c) \leftarrow \mathsf{snd}(st_S, ad, m^b)$	35 $(st_S, c) \leftarrow \mathsf{snd}(st_S, ad, m^b)$
06 $AD\text{-}C[s] \leftarrow (ad, c)$	36 If h:
07 $s \leftarrow s + 1$	37 $\quad AD\text{-}C[s] \leftarrow (ad, c)$
08 Return c	38 $\quad s \leftarrow s + 1$
	39 Return c
Oracle $\mathrm{rcv}(ad, c)$	**Oracle** $\mathrm{rcv}(ad, c)$
09 $(st_R, m) \leftarrow \mathsf{rcv}(st_R, ad, c)$	40 $(st_R, m) \leftarrow \mathsf{rcv}(st_R, ad, c)$
10 If $(st_R, m) = (\bot, \bot)$: Abort	41 If $(st_R, m) = (\bot, \bot)$: Return $\bot$
11 If $r < s \wedge (ad, c) = AD\text{-}C[r]$:	42 If $r < s \wedge (ad, c) = AD\text{-}C[r]$:
12 $\quad r \leftarrow r + 1$	43 $\quad r \leftarrow r + 1$
13 Else:	44 Else:
14 $\quad$ Abort	45 $\quad h \leftarrow \mathrm{F}$
15 Return $\diamond$	46 Return $h \,?\, \diamond : m$

Fig. 8. Games for confidentiality of unidirectional channels against chosen-plaintext (left) and chosen-ciphertext (right) attacks. (See also Figure 4 for an explanation of the notation.) Note the rcv oracle of game $\mathrm{IND}^{\mathrm{1cpa}}$ is redundant and could be removed.

unidirectional channel, be secure in either direction. We argue that this notion of security would be too weak. Concretely, we describe a bidirectional channel that is obviously insecure, yet it fulfills the naive security notion considered above.

Let Ch be a unidirectional channel with key space $\mathcal{K}$, let $\mathcal{K}^* = \mathcal{K} \times \mathcal{K}$, and let Ch^* be the bidirectional channel obtained from Ch as specified in Figure 9. The constructed channel essentially protects the two directions of communication using independent keys, $K^{\leftarrow}$ and $K^{\rightarrow}$. However, all ciphertexts sent from Alice to Bob (generated using key $K^{\rightarrow}$) will carry the Bob-to-Alice key $K^{\leftarrow}$ in the clear, and vice versa.

Note that if we restrict the attention to the (traditional) unidirectional case by letting Alice only send and Bob only receive, channel Ch^* provides as much confidentiality as Ch does (e.g., IND-2CCA security). However, in a bidirectional setting channel Ch^* is blatantly insecure, even against fully passive attacks (IND-2CPA).

Proc $\mathsf{init}^*(K)$	**Proc** $\mathsf{snd}^*(st, ad, m)$	**Proc** $\mathsf{rcv}^*(st, ad, c)$
00 $(K^{\rightarrow}, K^{\leftarrow}) \leftarrow K$	06 $(k, st_S, st_R) \leftarrow st$	11 $(k, st_S, st_R) \leftarrow st$
01 $(st_S^{\rightarrow}, st_R^{\rightarrow}) \leftarrow \mathsf{init}(K^{\rightarrow})$	07 $(st_S, c) \leftarrow \mathsf{snd}(st_S, k \,\|\, ad, m)$	12 $(k', c) \leftarrow c$
02 $(st_S^{\leftarrow}, st_R^{\leftarrow}) \leftarrow \mathsf{init}(K^{\leftarrow})$	08 $st \leftarrow (k, st_S, st_R)$	13 $(st_R, m) \leftarrow \mathsf{rcv}(st_R, k' \,\|\, ad, c)$
03 $st_A \leftarrow (K^{\leftarrow}, st_S^{\rightarrow}, st_R^{\leftarrow})$	09 $c \leftarrow (k, c)$	14 If $(st_R, m) \neq (\bot, \bot)$:
04 $st_B \leftarrow (K^{\rightarrow}, st_S^{\leftarrow}, st_R^{\rightarrow})$	10 Return (st, c)	15 $\quad st \leftarrow (k, st_S, st_R)$
05 Return (st_A, st_B)		16 Else:
		17 $\quad st \leftarrow \bot$
		18 Return (st, m)

Fig. 9. A bidirectional channel $\mathsf{Ch}^* = (\mathsf{init}^*, \mathsf{snd}^*, \mathsf{rcv}^*)$ built from two instances of a unidirectional channel $\mathsf{Ch} = (\mathsf{init}, \mathsf{snd}, \mathsf{rcv})$. It falls prey to a purely passive (IND-2CPA) attack, even if Ch is secure against active adversaries (IND-1CCA).