Broadcast Steganography*

Nelly Fazio[1,3], Antonio R. Nicolosi[2], and Irippuge Milinda Perera[3]

[1]The City College of CUNY
fazio@cs.ccny.cuny.edu
[2]Stevens Institute of Technology
nicolosi@cs.stevens.edu
[3]The Graduate Center of CUNY
{nfazio,iperera}@gc.cuny.edu

March 4, 2014

Abstract

We initiate the study of broadcast steganography (BS), an extension of steganography to the multi-recipient setting. BS enables a sender to communicate covertly with a dynamically designated set of receivers, so that the recipients recover the original content, while unauthorized users and outsiders remain *unaware* of the covert communication. One of our main technical contributions is the introduction of a new variant of anonymous broadcast encryption that we term *outsider-anonymous broadcast encryption with pseudorandom ciphertexts* (oABE$). Our oABE$ construction achieves sublinear ciphertext size and is secure in the standard model. Besides being of interest in its own right, oABE$ enables an efficient construction of BS secure in the standard model against adaptive adversaries with sublinear communication complexity.

Keywords: Steganography, Broadcast Encryption, Receiver Anonymity.

1 Introduction

Point-to-point encryption schemes are effective at concealing the *meaning* of the communication between two parties. If the parties additionally desire that the very *existence* of their communication over a public channel remains concealed, then the required tool is *steganography*. Conventional steganography allows *two* parties to communicate covertly, even in the presence of an adversary, by *hiding* the intended content within other, seemingly harmless messages. After its initial formalization in the information-theoretic [11] and complexity-theoretic [29, 31, 44] settings, steganography has received regular attention by the cryptographic community. To a first approximation, existing solutions differ mostly in the degree of adversarial control that they can tolerate, and in the specific trade-off that they achieve among the main efficiency measures of transmission overhead, public/secret key storage, and encryption/decryption complexity.

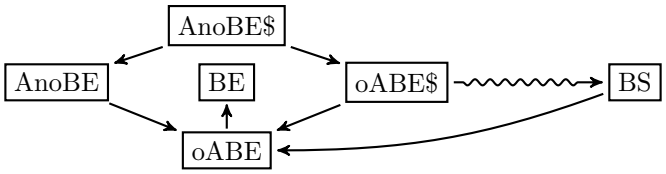

Figure 1: Relations between broadcast encryption (BE), (outsider) anonymous broadcast encryption (AnoBE and oABE), and broadcast steganography (BS). A straight arrow means that one notion implies the other, while the curly arrow denotes our black-box construction from oABE$ to BS (*cf.* Sect. 5). (To avoid cluttering the figure, relations implied by transitivity are omitted.)

Steganography. Simmons [42] introduced the cryptographic community to the problem of hidden communication with his famous *prisoners' dilemma*: Alice and Bob are in jail and can only talk in the presence of the jail warden Ward. Ward will not allow any encrypted communication, so Alice and Bob must hide their messages about an escape plan (the *hiddentext*) into innocent-looking communication (the *stegotext*) that Ward cannot distinguish from casual chatter (the *covertext*). Modern cryptographic treatment of steganography began with Cachin's formalization in the information-security setting [11] and Hopper *et al.*'s in the complexity-theoretic one [29]. Kiayias *et al.* [32] improve the efficiency of the steganographic protocol of [29] by replacing the use of a pseudorandom function family with the combination of a pseudorandom generator and a t-wise independent hash function. This approach was further refined in [33] to obtain a key-efficient steganographic system, where the gain stems from employing a novel *rejection sampling* method based on extractors.

In 2004, von Ahn and Hopper [44] extended the notion of steganography to the public-key setting, but mostly focused on security against passive adversaries. A stronger security model (steganographic secrecy against adaptive chosen-covertext attacks, or SS-CCA) was defined by Backes and Cachin [4], but their constructions attained only an intermediate security notion, termed steganographic secrecy against publicly-detectable, replayable adaptive chosen-covertext attacks (SS-PDR-CCA). Building upon the work of [4], Hopper [28] attained full SS-CCA security under the Decisional Diffie-Hellman (DDH) assumption, in the standard model. Le and Kurosawa [35] suggested a weaker generalization of the model of [4], but with better efficiency than [28].

All steganographic constructions mentioned above assume that the communication channel can be modeled by an efficient covertext sampler that can be queried adaptively, in a black-box manner. Dedic *et al.* [14, 41] looked into communication bounds for stegosystems of this kind, while Lysyanskaya and Meyerovich [37] dealt with the case of imperfect channel oracle samplers.

Work of von Ahn *et al.* [45] and Chandran *et al.* [13] introduced stealthiness to the setting of secure function evaluation, originating the notion of *covert two-party/multi-party computation*. Covert protocols allow parties to carry out distributed computations in a way that hides their very *intent* of taking part in the protocol: that is, unless *all* parties actively participate, nobody can detect that protocol messaging had been initiated (and aborted). This capability supports stealthy coordination between mutually mistrustful parties and enables fascinating applications like covert authentication [45] and co-spy detection [13]. However, it does not imply efficient covert dissemination of information to a chosen subset of (mostly passive) receivers, which is the main focus of this paper.

Broadcast Steganography (BS). In this work, we extend steganography to the broadcast setting. Intuitively, *broadcast steganography* enables a sender to communicate covertly with a dynamically

Table 1: Comparison of the parameters of (outsider) anonymous broadcast encryption schemes. Each scheme is CCA-secure and requires only one decryption attempt. Only our scheme provides pseudorandom ciphertexts ($c \approx \$$:Yes). N is the total number of users and r is the number of revoked users.

Scheme	Length of MPK	Length of sk	Length of c	Security Model	Anonymity	$c \approx \$$
BBW06 [5]	$O(N)$	$O(1)$	$O(N-r)$	Static, RO	Full	No
LPQ12 [36]	$O(N)$	$O(1)$	$O(N-r)$	Adaptive, Standard	Full	No
FP12a [20]	$O(N)$	$O(\log N)$	$O\left(r \log\left(\frac{N}{r}\right)\right)$	Adaptive, Standard	Outsider	No
FP12b [21]	$O(N \log N)$	$O(N)$	$O(r)$	Adaptive, Standard	Outsider	No
oABE$ [ours]	$O(N)$	$O(\log N)$	$O\left(r \log\left(\frac{N}{r}\right)\right)$	Adaptive, Standard	Outsider	Yes

designated set of receivers, so that authorized recipients correctly recover the original content, while unauthorized users and outsiders remain *unaware* of the covert communication. To construct broadcast steganography, we employ the "encrypt-then-embed" paradigm that underpins most steganographic constructions [4, 28, 29, 44] (*cf.* Sect. 2). Realizing this approach, however, requires solving several technical problems.

The first issue is that, in broadcast encryption, the receiver set is included explicitly in the ciphertext as part of its header (*e.g.*, [6, 7, 9, 15–18, 22, 23, 25, 26, 39]). This is a non-starter for steganography, which intrinsically requires that the existence of any data in the channel be concealed. To address this issue, we turn to *private* broadcast encryption, a notion introduced by Barth *et al.* [5] with the goal of keeping the identities of the authorized receivers anonymous (Sect. 2).

The second hurdle is that the "encrypt-then-embed" paradigm requires the underlying encryption functionality to have *pseudorandom* ciphertexts. This property so far had not been considered in the broadcast encryption literature, and none of the existing constructions support it natively. Interestingly, attaining pseudorandom ciphertexts requires implicitly that the identities of the recipients be unintelligible *in the view of outsiders* (pseudorandomness of the ciphertext clearly cannot hold in the view of the recipients). This condition ties back directly to the previous issue, but in a weaker form, as recipient anonymity is only required to hold against outsiders. As it turns out, Fazio and Perera [20] recently proposed a relaxation of full anonymity of exactly this sort: *outsider-anonymous broadcast encryption* (oABE). This notion trades some degree of anonymity for better efficiency: whereas all known fully-anonymous broadcast encryption schemes [5, 36] have ciphertexts *linear* in the number of receivers, the constructions of [20] obtain *sublinear* ciphertext length, though they do not necessarily guarantee that authorized users will learn no information about other members of the receiver set.

In light of the above observations, we put forth and realize (Sect. 4) a new broadcast encryption variant that we term *outsider-anonymous broadcast encryption with pseudorandom ciphertexts* (oABE$). oABE$ enables a black-box construction of BS (*cf.* Sect. 5). Realizing an efficient oABE$ scheme requires non-trivial enhancements to the oABE construction of [20], for it entails resolving the apparent tension between our ciphertext pseudorandom property and the ciphertext redundancy introduced by common approaches to CCA security [8, 19]. Our solution harmonizes these requirements using a novel Pedersen-like encapsulation mechanism discussed in Sect. 4.2. A comparison of our oABE$ construction with existing ones is reported in Table 1, whereas Fig. 1 shows how oABE$ relates to other anonymous broadcast communication tools.

Applications. The combination of stealth and revocation capabilities offered by broadcast steganography enables defenses against insider threats in anti-censorship systems, intelligence scenarios, and

Table 2: The parameters of our black-box broadcast steganography schemes. Type-1 channels are the most general, and are modeled as stateful probabilistic oracles whose output distribution *may* depend on past samples. Type-2 channels are slightly more restrictive as they assume history independence, and can then be modeled as efficiently sampleable document distributions, *i.e.*, efficiently computable randomized functions. N is the total number of users and r is the number of revoked users. The notion of BS-CHA (resp. BS-CCA) captures passive (resp. active) security for the BS setting (*cf.* Sect. 3.2).

Scheme	Length of MPK	Length of sk	Length of s	Security Model	Channel Type
BS-CHA	$O(N)$	$O(\log N)$	$O\big(r \log\big(\frac{N}{r}\big)\big)$	Adaptive, Standard	1
BS-CCA	$O(N)$	$O(\log N)$	$O\big(r \log\big(\frac{N}{r}\big)\big)$	Adaptive, Standard	2

other domains that rely on covert communication [38, 43].

For a military example, consider a camp where each soldier has an army smartphone, on which they receive weather forecast, unclassified news and other information in the clear. Suppose that headquarters suspect that a group of officials are conspiring to commit treachery, and decides to carry out an undercover investigation to confirm the identities of the traitors. Conventional broadcast encryption does not suffice to protect the transmission channel to the soldiers involved in the investigation of the traitors, because the selective exclusion of the conspirators from the communication would already put them on notice. Broadcast steganography, instead, would allow delivery of instructions to the investigating parties without risking alerting the traitors to the investigation.

For a civil rights scenario, an activist/blogger may want to hide her commentary into innocent-looking image postings to social media services (*e.g.*, Instagram or Weibo). Because censorship authorities may infiltrate among the activist's followers, the ability of broadcast steganography to authorize/deauthorize recipients at a fine grain would enable the blogger to revoke the infiltrator and prevent him from recovering the hiddentext, *without him noticing that he has been singled out.*

Our Contributions. This work initiates the study of broadcast steganography. After introducing a suitable security framework, we highlight the connections with the issue of recipient-anonymity in broadcast encryption. One of our main technical contributions is the introduction of a new variant of anonymous broadcast encryption that we term outsider-anonymous broadcast encryption with pseudorandom ciphertexts. Our oABE$ construction achieves sublinear ciphertext size and is secure in the standard model against adaptive adversaries, which required circumventing multiple technical hurdles and is thus of independent interest. Finally, we devise efficient oABE$-based BS schemes at varying security levels (*cf.* Table 2), including a construction with sublinear stegotexts secure in the standard model against adaptive adversaries.

2 Background

Documents & Covertexts. Let $\Sigma = \{0,1\}^\sigma$ be a finite set of bit-strings with length σ. Denote by Σ^* the set of sequences of finite length over Σ. We call the strings $u \in \Sigma$ *documents* and the strings $s \in \Sigma^*$ *covertexts*.

Channels. A *channel* $\mathfrak{C}_h$ is a function that takes as input a *channel history* $h \in \Sigma^*$ and produces a probability distribution on Σ. A channel history $h = s_1 \| \ldots \| s_l \in \Sigma^*$ is called *legal* if for all $i \in [1, l]$, $\Pr_{\mathfrak{C}_{s_1\|\ldots\|s_{i-1}}}[s_i] > 0$. A sampling of l documents in succession from a channel is denoted

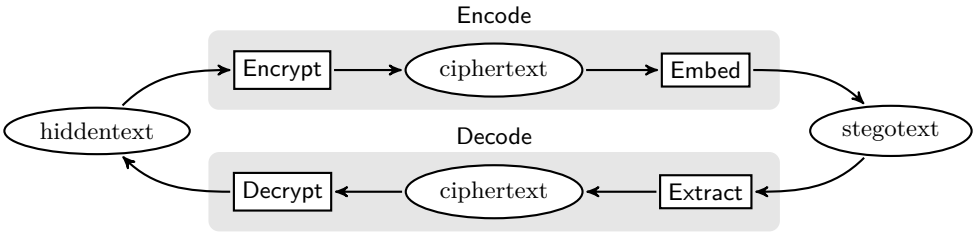

Figure 2: The "encrypt-then-embed" paradigm underlying (broadcast) steganography.

by $s = s_1 \| \ldots \| s_l \leftarrow \mathfrak{C}_h^l$ (shorthand notation for $s_1 \leftarrow \mathfrak{C}_h, s_2 \leftarrow \mathfrak{C}_{h\|s_1}, \ldots, s_l \leftarrow \mathfrak{C}_{h\|s_1\|\ldots\|s_{l-1}}$). A channel is called *always informative* if for every legal history $h \in \Sigma^*$, $H_\infty(\mathfrak{C}_h^l) = \Omega(l)$, where H_∞ is the min-entropy function. A channel can be modeled either as an oracle or as an efficiently computable randomized function $\mathsf{Channel}(h; r)$ (where r denotes the random coins). While the latter is a stronger assumption on the channel, [28] shows it to be necessary for secure steganography. Efficiently computable channels also enable broadcast steganographic constructions with stronger security guarantees (*cf.* Sect. 5).

Public-Key Steganography. From an operational standpoint, public-key steganography resembles the setting of asymmetric encryption: a participant with a public/secret key pair is able to receive covert messages (the *hiddentexts*) from another party, who only knows the public key. Unlike the case of public-key cryptography, however, the encoded hiddentexts, termed *stegotexts*, are required to be indistinguishable from the covertexts of the communication channel.

A common approach to realize public-key stegosystems is the "encrypt-then-embed" paradigm [4, 28, 29, 44], depicted in Fig. 2. At a high level, encoding is accomplished by first encrypting the hiddentext using a public-key cryptosystem, and then implanting the resulting ciphertext in the stegotext using an embedding function. The decoding process develops similarly, but in the reverse direction. Based on the security properties of the underlying cryptosystem and embedding function, one obtains stegosystems with a variety of security guarantees (*cf.* Sect. 1).

Outsider-Anonymous Broadcast Encryption (oABE). The notion of *private* broadcast encryption was initially introduced in [5], with the aim of providing explicit protection for identities of the receivers during each transmission. As a proof-of-concept, therein the authors suggested both generic and number-theoretic public-key constructions that do not leak any information about the list of authorized receivers, and are secure in the standard model and in the random oracle model, respectively. The proposed schemes, however, have communication complexity linear in the number of recipients. In [36], Libert *et al.* suggested proof techniques to argue the security of (a variant of) the number-theoretic construction of [5] without reliance on random oracles, thus attaining anonymous broadcast encryption with efficient decryption in the standard model. Still, ciphertexts in the resulting construction have length linear in the number of recipients. In [34], Kiayias and Samari put forth lower bounds on the ciphertext size of private broadcast encryption schemes and showed, among other results, that fully anonymous broadcast encryption schemes with a certain "atomicity" property (satisfied, *e.g.*, by the schemes of [5, 36]) must have $\Omega(s \cdot \lambda)$ ciphertext size, where s is the number of authorized receivers and λ is the security parameter.

Fazio and Perera [20] formalized the notion of *outsider-anonymous broadcast encryption*, which lies between the complete lack of protection that characterizes traditional broadcast encryption schemes as introduced in [22], and the full anonymity provided by [5, 36]. In an oABE scheme, an

attacker who intercepts a ciphertext of which she is not a legal recipient will be unable to learn anything about the identities of the legal recipients (let alone the contents of the ciphertext). Still, for those ciphertexts for which the adversary is in the authorized set of recipients, she might also garner information about the identities of the other receivers. This seems a natural relaxation, since often the *contents* of the communication already reveals something about the recipient set. Moreover, it enables schemes that achieve *sublinear* ciphertexts size and are secure against adaptive adversaries in the standard model. We observe that, in light of the lower bounds of [34], the trade-off proposed in [20] may be unavoidable.

Entropy Smoothing Hash. A family of hash functions $\mathcal{H}_{es} = \{H : X \to Y\}$ is "entropy smoothing" [30] if it is hard to distinguish $(H, H(x))$ from (H, y), where H is a random element of $\mathcal{H}_{es}$, x is a random element of X, and y is a random element of Y. More formally, $\mathcal{H}_{es}$ is called (t, ϵ)-entropy smoothing if for every t-time adversary $\mathcal{A}$,

$$\Big|\Pr\Big[\mathcal{A}(H, H(x)) = 1 \mid H \leftarrow\$\ \mathcal{H}_{es}, x \leftarrow\$\ X\Big] - \Pr\Big[\mathcal{A}(H, y) = 1 \mid H \leftarrow\$\ \mathcal{H}_{es}, y \leftarrow\$\ Y\Big]\Big| \leq \epsilon,$$

where the probability is over the choice of H, x, y and over the random coins used by $\mathcal{A}$.[1]

3 Broadcast Steganography (BS)

3.1 The Setting

Definition 3.1: A broadcast steganography scheme, associated with a universe of users $U = [1, N]$, a message space $\mathcal{MSP}$, and a channel $\mathfrak{C}_h$ on a set of documents Σ, is a tuple of probabilistic polynomial-time (PPT) algorithms $(\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encode}, \mathsf{Decode})$ such that:

$(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, N)$: Setup takes the security parameter 1^λ and the number of users in the system N as inputs and outputs the master public key MPK and the master secret key MSK.

$sk_i \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$: Given the master public key MPK, the master secret key MSK, and a user $i \in U$, KeyGen generates a secret key sk_i for user i.

$s \leftarrow \mathsf{Encode}(\mathsf{MPK}, S, h, m)$: Encode takes the master public key MPK, a set of receivers $S \subseteq U$, a channel history $h \in \Sigma^*$, and a message $m \in \mathcal{MSP}$ as inputs and outputs a stegotext $s \in \Sigma^*$ from the support of $\mathfrak{C}_h^l$ for some $l = poly(|m|)$.

$m/\perp \coloneqq \mathsf{Decode}(\mathsf{MPK}, sk_i, s)$: Given the master public key MPK, a secret key sk_i, and a stegotext $s \in \Sigma^*$, Decode either outputs a message $m \in \mathcal{MSP}$ or the failure symbol $\perp$. We assume that Decode is deterministic.

Correctness. For every $S \subseteq U$, $i \in S$, legal channel history $h \in \Sigma^*$, and $m \in \mathcal{MSP}$, if $(\mathsf{MPK}, \mathsf{MSK})$ is output by $\mathsf{Setup}(1^\lambda, N)$ and sk_i is generated by $\mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$, then $\mathsf{Decode}(\mathsf{MPK}, sk_i, \mathsf{Encode}(\mathsf{MPK}, S, h, m)) = m$ except with negligible probability in the security parameter λ. $\Diamond$

[1] Entropy smoothing is related to strong randomness extraction [46], but it is a much less stringent (and hence easier to realize) notion, as it seeks only computational (rather than information-theoretic) guarantees, and it is specific to *one* entropy source (the uniform distribution over the domain X), whereas strong extractors are applicable to any source of a given min-entropy.

Remark 3.1. In contrast to the definition from [28], our definition requires that the Decode algorithm works without receiving the channel history h corresponding to the stegotext s as an input. This is crucial for an efficient broadcast steganography scheme, because requiring that authorized users feed the Decode algorithm with the same h that was used by the sender entails a level of coordination that is unrealistic in a broadcast setting. Our definition also applies to channels whose samples do not depend on h at all, as Encode may simply ignore h.

3.2 The Security Models

In broadcast encryption (BE), the adversary's goal is to learn something about the message encrypted within a given ciphertext despite not having a valid decryption key. In broadcast steganography, the adversary's goal is to detect the *presence* of a message in a given covertext without a valid decoding key. In either case, one may consider multiple levels of security, according to the amount of power afforded to the attacker. We discuss below three models of security for broadcast steganography schemes, followed by formal definitions later in this section.

Chosen-Hiddentext Attack (BS-CHA). This is the weakest model of security for a broadcast steganography scheme. Analogous to the chosen-plaintext attack in broadcast encryption, the adversary in this context is only allowed to corrupt users by gaining their secret keys.

Publicly-Detectable Replayable Chosen-Covertext Attack (BS-PDR-CCA). In this model of security, the adversary is additionally given access to a decoding oracle through which they can obtain the hiddentext (if any) in any covertext s of their choice, as recovered by any honest user i of their choice, subject to the following restriction: After receiving the challenge covertext s^* for the set of recipients S^*, the adversary is not allowed to query the decoding oracle with a user index i and a covertext s such that $i \in S^*$ and $s \equiv_{\mathsf{MPK}} s^*$, where $\equiv_{\mathsf{MPK}}$ is an arbitrary *compatible relation*:

Definition 3.2: Let $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encode}, \mathsf{Decode})$ be a BS scheme. A binary relation on stegotexts of Π induced by a master public key MPK of Π is called a *compatible relation* (denoted by $\equiv_{\mathsf{MPK}}$) if for any two stegotexts s_1, s_2 encoded under sets of receivers S_1, S_2 respectively, we have

1. If $s_1 \equiv_{\mathsf{MPK}} s_2$ then for any $i_1 \in S_1$ and $i_2 \in S_2$, $\mathsf{Decode}(\mathsf{MPK}, sk_{i_1}, s_1) = \mathsf{Decode}(\mathsf{MPK}, sk_{i_2}, s_2)$ except with negligible probability in the security parameter λ.

2. There exists a PPT algorithm that takes MPK, s_1, s_2 and determines whether $s_1 \equiv_{\mathsf{MPK}} s_2$. $\Diamond$

Chosen-Covertext Attack (BS-CCA). A BS-CCA adversary has the same capabilities from the BS-PDR-CCA model of security, but the restriction for the decoding queries is now lifted. Specifically, the only covertext that the adversary is not allowed to submit to the decoding oracle with a user index $i \in S^*$ is the challenge covertext s^* itself.

We now formally define the BS-CCA security model via the following security game.

Definition 3.3: For a given BS scheme $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encode}, \mathsf{Decode})$, the BS-IND-CCA game, played between a PPT adversary $\mathcal{A}$ and a challenger $\mathcal{C}$, is defined as follows:

Setup: $\mathcal{C}$ runs $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, N)$ and gives $\mathcal{A}$ the resulting master public key MPK, keeping the master secret key MSK to itself. $\mathcal{C}$ also initializes the set of revoked users R to be empty.

Phase 1: $\mathcal{A}$ adaptively issues queries $q_1, \ldots, q_m$ of one of the following types:

- Secret-key query i: $\mathcal{A}$ requests the secret key of a user $i \in U$. $\mathcal{C}$ runs $sk_i \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$, adds i to R, and sends sk_i to $\mathcal{A}$.
- Decoding query (i, s): $\mathcal{A}$ issues a decoding query on a user index $i \in U$ and a covertext $s \in \Sigma^*$. $\mathcal{C}$ computes $\mathsf{Decode}(\mathsf{MPK}, \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i), s)$ and gives the result to $\mathcal{A}$.

Challenge: $\mathcal{A}$ gives $\mathcal{C}$ a message $m^* \in \mathcal{MSP}$, a legal history $h \in \Sigma^*$, and a set of user identities $S^* \subseteq U$ with the restriction that $S^* \cap R = \emptyset$. $\mathcal{C}$ picks a random bit $b^* \in \{0, 1\}$ and generates the challenge s^* depending on it as follows. If $b^* = 0$, then $\mathcal{C}$ encodes m^* into a stegotext s^* for the receiver set S^*, *i.e.*, $s^* \leftarrow \mathsf{Encode}(\mathsf{MPK}, S^*, h, m^*)$. Otherwise, $\mathcal{C}$ sample s^* as a covertext of equal length, *i.e.*, $s^* \leftarrow\!\!\$\ \mathfrak{C}_h^{l^*}$ for $l^* = |\mathsf{Encode}(\mathsf{MPK}, S^*, h, m^*)|/\sigma$. At the end, $\mathcal{C}$ gives s^* to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ adaptively issues additional queries $q_{m+1}, \ldots, q_n$ where each q_i is one of the following:

- Secret-key query i such that $i \notin S^*$.
- Decoding query (i, s) such that, if $i \in S^*$, then $s \neq s^*$.

Guess: $\mathcal{A}$ outputs a guess $b \in \{0, 1\}$ and wins if $b = b^*$.

The adversary $\mathcal{A}$ is called a BS-IND-CCA adversary and $\mathcal{A}$'s advantage is defined as

$$\mathsf{Adv}_{\mathcal{A},\Pi}^{\text{BS-IND-CCA}} := \left|\Pr[b = b^*] - \tfrac{1}{2}\right|,$$

where the probability is over the random coins used by the adversary $\mathcal{A}$ and the challenger $\mathcal{C}$. $\Diamond$

Definition 3.4: A BS scheme Π is $(t, Q_{sk}, Q_d, \epsilon)$-BS-CCA-secure if for any t-time BS-IND-CCA adversary making at most Q_{sk} adaptive secret-key queries and at most Q_d adaptive decoding queries, it is the case that $\mathsf{Adv}_{\mathcal{A},\Pi}^{\text{BS-IND-CCA}} \leq \epsilon$. $\Diamond$

By restricting the kind of decoding queries allowed in *Phase 2* of the BS-IND-CCA game above, we can obtain the BS-IND-PDR-CCA game. Specifically, the adversary now cannot issue any decoding query (i, s) such that $i \in S^*$ and $s \equiv_{\mathsf{MPK}} s^*$ for some compatible relation $\equiv_{\mathsf{MPK}}$. The adversary $\mathcal{A}$ in this game is called a BS-IND-PDR-CCA adversary and $\mathcal{A}$'s advantage is defined as

$$\mathsf{Adv}_{\mathcal{A},\Pi}^{\text{BS-IND-PDR-CCA}} := \left|\Pr[b = b^*] - \tfrac{1}{2}\right|.$$

Definition 3.5: A BS scheme Π is $(t, Q_{sk}, Q_d, \epsilon)$-BS-PDR-CCA-secure with respect to some compatible relation $\equiv_{\mathsf{MPK}}$ if for any t-time BS-IND-PDR-CCA adversary making at most Q_{sk} adaptive secret-key queries and at most Q_d adaptive decoding queries, it holds that $\mathsf{Adv}_{\mathcal{A},\Pi}^{\text{BS-IND-PDR-CCA}} \leq \epsilon$.$\Diamond$

The BS-IND-CHA game is defined similar to the BS-IND-CCA game, with the restriction that the adversary is not allowed to issue any decoding queries during *Phase 1* and *Phase 2*. The adversary is still allowed to issue secret-key queries.

Definition 3.6: A BS scheme Π is (t, Q_{sk}, ϵ)-BS-CHA-secure if Π is $(t, Q_{sk}, 0, \epsilon)$-BS-CCA-secure.$\Diamond$

4 Anonymity and Pseudorandomness in Broadcast Encryption

In Sect. 2, we briefly discussed the notion of outsider-anonymous broadcast encryption [20], a security model for BE whose goal is to hide the identities of the intended receivers of a broadcast ciphertext from unauthorized users. As outlined in Sect. 1, a crucial technical step to realize broadcast steganography is combining receiver anonymity with pseudorandomness of broadcast ciphertexts (*cf.* Sect. 5). This section develops the notion of *outsider-anonymous broadcast encryption with pseudorandom ciphertexts* (oABE$), and presents an efficient construction secure in the standard model under a stronger security model, *outsider anonymity and ciphertext pseudorandomness against chosen-ciphertext attacks* (oABE$-CCA).

4.1 The Security Models of oABE$

We now present three oABE$ security models: oABE$-CPA, oABE$-PDR-CCA, and oABE$-CCA. In Sect. 4.2, we present an oABE$-CCA-secure construction. At a high level, these security models require that for any message m^* and set of recipients S^*, no PPT adversary $\mathcal{A}$ can distinguish between an actual encryption of m^* intended for the set S^*, and a truly random string of the same length as an encryption of m^* for S^*, so long as $\mathcal{A}$ does not possess the secret key of any user in S^*.

Definition 4.1: For a given oABE$ scheme $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encrypt}, \mathsf{Decrypt})$, the oABE$-IND-CCA game, played between a PPT adversary $\mathcal{A}$ and a challenger $\mathcal{C}$, is defined as follows:

Setup: $\mathcal{C}$ runs $(\mathsf{MPK}, \mathsf{MSK}) \leftarrow \mathsf{Setup}(1^\lambda, N)$ and gives $\mathcal{A}$ the resulting master public key MPK, keeping the master secret key MSK to itself. $\mathcal{C}$ also initializes the set of revoked users R to be empty.

Phase 1: $\mathcal{A}$ adaptively issues queries $q_1, \ldots, q_m$ where each q_i is one of the following:

- Secret-key query i: $\mathcal{A}$ requests the secret key of a user $i \in U$. $\mathcal{C}$ runs $sk_i \leftarrow \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$, adds i to R, and sends sk_i to $\mathcal{A}$.
- Decryption query (i, c): $\mathcal{A}$ sends a decryption query on a user $i \in U$ and a ciphertext $c \in \mathcal{CSP}$. $\mathcal{C}$ computes $\mathsf{Decrypt}(\mathsf{MPK}, \mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i), c)$ and gives the result to $\mathcal{A}$.

Challenge: $\mathcal{A}$ gives $\mathcal{C}$ a message $m^* \in \mathcal{MSP}$ and a set of user identities $S^* \subseteq U$ with the restriction that $S^* \cap R = \emptyset$. $\mathcal{C}$ picks a random bit $b^* \in \{0,1\}$ and generates the challenge ciphertext c^* depending on it: if $b^* = 0$, then $c^* \leftarrow \mathsf{Encrypt}(\mathsf{MPK}, S^*, m^*)$, else $c^* \leftarrow\$ \{0,1\}^{l^*}$ for $l^* = |\mathsf{Encrypt}(\mathsf{MPK}, S^*, m^*)|$. The challenge ciphertext c^* is then given to $\mathcal{A}$.

Phase 2: $\mathcal{A}$ adaptively issues additional queries $q_{m+1}, \ldots, q_n$ where each q_i is one of the following:

- Secret-key query i such that $i \notin S^*$.
- Decryption query (i, c) such that, if $i \in S^*$, then $c \neq c^*$.

Guess: $\mathcal{A}$ outputs a guess $b \in \{0,1\}$ and wins if $b = b^*$.

The adversary $\mathcal{A}$ is called an oABE$-IND-CCA adversary and $\mathcal{A}$'s advantage is defined as

$$\mathsf{Adv}^{\text{oABE\$-IND-CCA}}_{\mathcal{A},\Pi} := \left|\Pr[b = b^*] - \tfrac{1}{2}\right|,$$

where the probability is over the random coins used by the adversary $\mathcal{A}$ and the challenger $\mathcal{C}$. $\Diamond$

Observe that the key difference of the above definition from the oABE notion defined in [20] is in the *Challenge* phase, where the challenger either returns the encryption of m^* or a random bit-string with appropriate length.

Definition 4.2: An oABE\$ scheme Π is $(t, Q_{sk}, Q_d, \epsilon)$-oABE\$-CCA-secure if for any t-time oABE\$-IND-CCA adversary making at most Q_{sk} (resp. Q_d) adaptive secret-key (resp. decryption) queries we have $\mathsf{Adv}^{\text{oABE\$-IND-CCA}}_{\mathcal{A},\Pi} \leq \epsilon$. $\Diamond$

The oABE\$-IND-PDR-CCA game is obtained by restricting the adversary during *Phase 2* of the oABE\$-IND-CCA game from submitting any decoding query (i, c) such that $i \in S^*$ and $c \equiv_{\mathsf{MPK}} c^*$, where $\equiv_{\mathsf{MPK}}$ is an arbitrary compatible relation of the oABE\$ scheme.[2] The adversary $\mathcal{A}$ in this game is called an oABE\$-IND-PDR-CCA adversary and $\mathcal{A}$'s advantage is defined as

$$\mathsf{Adv}^{\text{oABE\$-IND-PDR-CCA}}_{\mathcal{A},\Pi} := \left|\Pr[b = b^*] - \tfrac{1}{2}\right|.$$

Definition 4.3: An oABE\$ scheme Π is $(t, Q_{sk}, Q_d, \epsilon)$-oABE\$-PDR-CCA-secure with respect to a compatible relation $\equiv_{\mathsf{MPK}}$ if for any t-time oABE\$-IND-PDR-CCA adversary making at most Q_{sk} adaptive secret-key queries and at most Q_d adaptive decoding queries $\mathsf{Adv}^{\text{oABE\$-IND-PDR-CCA}}_{\mathcal{A},\Pi} \leq \epsilon$.$\Diamond$

By restricting the adversary in the oABE\$-IND-CCA game from submitting any decoding queries during *Phase 1* and *Phase 2*, we obtain the oABE\$-IND-CPA game. The adversary is still allowed to issue secret-key queries.

Definition 4.4: An oABE\$ scheme Π is (t, Q_{sk}, ϵ)-oABE\$-CPA-secure if Π is $(t, Q_{sk}, 0, \epsilon)$-oABE\$-CCA-secure. $\Diamond$

4.2 An oABE\$-CCA-Secure Construction

Our construction builds on the one of [20], so we start with a brief review of the latter. At a high level, the approach of [20] is to: (1) "bundle" multiple ciphertexts of an anonymous identity-based encryption scheme (AIBE, *e.g.*, [1, 10, 24]) into a single oABE ciphertext; (2) "tag" each AIBE ciphertext to enable the decryptor to efficiently locate the component compatible with her decryption key; and (3) "seal" everything together with a one-time signature to thwart CCA attacks. To attain pseudorandom oABE ciphertexts, we will start with an anonymous identity-based encryption scheme with *pseudorandom ciphertexts* (AIBE\$) like the one of [2]. Additionally, we will use an *entropy-smoothing* hash function [30] to hide the structure in the ciphertext tags.

These adjustments do not suffice because the presence of the one-time signature introduces additional structure in the oABE ciphertext of [20]. To get around this, we substitute one-time signatures with MACs (implemented via pseudorandom functions) and employ a variant of an *encapsulation mechanism* [8, 19] with an additional pseudorandom property. In short, an encapsulation mechanism is a "relaxed" commitment scheme consisting of a triplet of algorithms $(\mathsf{SetupCom}, \mathsf{Commit}, \mathsf{Open})$: $\mathsf{SetupCom}(1^\lambda)$ produces a commitment public key PK''; $\mathsf{Commit}(\mathsf{PK}'')$ samples a random bit string $\hat{k}$ together with associated commitment and decommitment information com and decom; and $\mathsf{Open}(\mathsf{PK}'', \mathsf{com}, \mathsf{decom})$ recovers $\hat{k}$. For *hiding*, triples of the form $(\mathsf{PK}'', \mathsf{com}, \hat{k})$ ought to be statistically indistinguishable from those of the form $(\mathsf{PK}'', \mathsf{com}, r)$ for random r. For

[2] The definition of a compatible relation for an oABE\$ scheme follows analogously to Definition 3.2.

Algorithm: $\mathsf{Commit}(\mathsf{PK}'')$

1 $\hat{k} \leftarrow\$ \{0,1\}^\lambda$
2 **repeat**
3 $\quad \tilde{k} \leftarrow\$ \mathbb{Z}_q$, $\mathsf{com} \coloneqq \mathsf{mp}(g_{\mathsf{com}}^{\hat{k}} h_{\mathsf{com}}^{\tilde{k}})$
4 **until** $\mathsf{com} < 2^\lambda$
5 $\mathsf{decom} \coloneqq (\hat{k}, \tilde{k})$
6 **return** $(\hat{k}, \mathsf{com}, \mathsf{decom})$

Algorithm: $\mathsf{Open}(\mathsf{PK}'', \mathsf{com}, \mathsf{decom})$

1 **parse** decom **as** $(\hat{k}, \tilde{k})$
2 **if** $\mathsf{com} = \mathsf{mp}(g_{\mathsf{com}}^{\hat{k}} h_{\mathsf{com}}^{\tilde{k}})$ **then**
3 $\quad$ **return** $\hat{k}$
4 **return** $\perp$

Figure 3: Our Pedersen-like encapsulation mechanism.

relaxed binding, given a random output $(\hat{k}, \mathsf{com}, \mathsf{decom})$ of $\mathsf{Commit}(\mathsf{PK}'')$, it should be hard to produce decom' such that $\mathsf{Open}(\mathsf{PK}'', \mathsf{com}, \mathsf{decom}') \notin \{\hat{k}, \perp\}$.

Let p, q be primes such that $2^\lambda < q < 2^{\lambda+1}$ and $p = 2q + 1$, and g be a square modulo p. Denote by $\mathbb{G} = \langle g \rangle$ the group of quadratic residues modulo p. To "pack" quadratic residues into λ bits, we will use rejection sampling along with the following well-known $\mathbb{G}$–$\mathbb{Z}_q$ bijection (*cf. e.g.*, [28]):

$$\mathsf{mp}(a) = \begin{cases} a & \text{if } a \leq q \\ p - a & \text{otherwise} \end{cases} \qquad \mathsf{mp}^{-1}(b) = \begin{cases} b & \text{if } b^{\frac{p-1}{2}} \equiv 1 \bmod p \\ p - b & \text{otherwise} \end{cases}$$

Figure 3 shows the Commit and Open functionalities of our Pedersen-like [40] encapsulation mechanism over $\mathbb{G}$, whose commitment public keys are random pairs $(g_{\mathsf{com}}, h_{\mathsf{com}})$ of generators of $\mathbb{G}$. The hiding requirement follows from the hiding properties of standard Pedersen commitments, coupled with the observation that $\mathsf{mp}(\cdot)$ is a bijection. Relaxed binding follows from the discrete logarithm assumption in $\mathbb{G}$, again similarly to standard Pedersen commitments. A novel feature of our encapsulation mechanism is that the distribution of commitments com induced by the $\mathsf{Commit}(\mathsf{PK}'')$ algorithm is *uniform over* $\{0,1\}^\lambda$, and hence the relaxed commitment scheme of Fig. 3 has *pseudorandom commitments*.

Let $\Pi' = (\mathsf{Init}, \mathsf{Ext}, \mathsf{Enc}, \mathsf{Dec})$ be an AIBE\$-CCA-secure AIBE\$ scheme with expansion ℓ (*i.e.*, $|\mathsf{Enc}(\mathsf{MPK}', \mathsf{ID}, m)| = \ell(|m|)$). Let $F : \{0,1\}^\lambda \times \{0,1\}^* \to \{0,1\}^\lambda$ be a PRF and let $\mathcal{H}_{es} = \{\mathbb{G}^2 \to \{0,1\}^\lambda\}$ be an entropy smoothing hash function family. Below we describe at a high level how we combine these primitives into an oABE\$-CCA-secure scheme Π; Fig. 4 reports the details.

To attain sublinear ciphertexts, we follow the approach of [20], which is based on the *Subset Cover Framework* [15, 39] (*cf.* also App. A). We arrange the $N = 2^n$ users in a perfect binary tree with N leaves, and assign to each user (using AIBE\$) $n + 1$ decryption keys, corresponding to all the nodes in the path to its designated leaf (Line 4 of KeyGen). Each oABE\$ ciphertexts consists of multiple AIBE\$ components. For efficient decryption, AIBE\$ components are tagged using a twin-DH-based [12] technique reminiscent of [21, 36] (Line 10 of $\mathsf{Encrypt}$) so that recipients can single out which AIBE\$ component to decrypt, and with which key (Lines 5–8 and 9 of $\mathsf{Decrypt}$). Throughout $\mathsf{Encrypt}$, we make sure that each piece in an oABE\$ ciphertext looks random, with the use of rejection sampling (Lines 3–5), entropy smoothing (Line 10), dummy components (Line 13), and pseudorandom MACs (Line 15) in place of one-time signature. Forgoing signatures introduce a complication, as the input to the PRF appears to depend on the PRF key $\hat{k}$: the $\overline{c}_j$ values and the oABE\$ components c_j's computed in Lines 10 and 11 are derived from com and decom, which correlate with $\hat{k}$. We solve this circularity by mediating the occurrence of $\hat{k}$ in the ciphertext via the encapsulation scheme of Fig. 3 (*cf.* App. B for more details).

Algorithm: $\mathsf{Setup}(1^\lambda, N)$

$(\mathsf{MPK}', \mathsf{MSK}') \leftarrow \mathsf{Init}(1^\lambda)$
$\mathsf{PK}'' \leftarrow \mathsf{SetupCom}(1^\lambda), H \leftarrow\$\ \mathcal{H}_{es}$
$\triangleright$ Fam – the set of all the subtrees in $\mathcal{T}$
for $j \coloneqq 1$ **to** $|\mathsf{Fam}|$ **do**
    $\triangleright$ T_j – the subtree in Fam indexed by j
    $\triangleright$ HID_j – the HID of T_j's root
    $a_{1,\mathsf{HID}_j}, a_{2,\mathsf{HID}_j}, b_{1,\mathsf{HID}_j}, b_{2,\mathsf{HID}_j} \leftarrow\$\ \mathbb{Z}_q$
    $A_{1,\mathsf{HID}_j} \coloneqq g^{a_{1,\mathsf{HID}_j}}$, $A_{2,\mathsf{HID}_j} \coloneqq g^{a_{2,\mathsf{HID}_j}}$
    $B_{1,\mathsf{HID}_j} \coloneqq g^{b_{1,\mathsf{HID}_j}}$, $B_{2,\mathsf{HID}_j} \coloneqq g^{b_{2,\mathsf{HID}_j}}$
$\mathsf{MPK} \coloneqq (\mathsf{MPK}', \mathsf{PK}'', H, N, \mathbb{G}, g, \{A_{i,\mathsf{HID}_j}, B_{i,\mathsf{HID}_j}\}_{i\in\{1,2\}, j\in[1,|\mathsf{Fam}|]})$
$\mathsf{MSK} \coloneqq (\mathsf{MSK}', \{a_{i,\mathsf{HID}_j}, b_{i,\mathsf{HID}_j}\}_{i\in\{1,2\}, j\in[1,|\mathsf{Fam}|]})$
return $(\mathsf{MPK}, \mathsf{MSK})$

Algorithm: $\mathsf{Encrypt}(\mathsf{MPK}, S, m)$

$r \coloneqq N - |S|, L \coloneqq \lfloor r \log(\frac{N}{r}) \rfloor$
$(\hat{k}, \mathsf{com}, \mathsf{decom}) \leftarrow \mathsf{Commit}(\mathsf{PK}'')$
repeat
    $s \leftarrow\$\ \mathbb{Z}_q$, $\overline{c}_0 \coloneqq \mathsf{mp}(g^s)$
until $\overline{c}_0 < 2^\lambda$
$\triangleright$ Cov – the subtrees covering S in $\mathcal{T}$
for $j \coloneqq 1$ **to** $|\mathsf{Cov}|$ **do**
    $\triangleright$ T_j – a subtree in Cov
    $\triangleright$ HID_j – the HID of T_j's root
    $\overline{c}_j \coloneqq H((A^{\mathsf{com}}_{1,\mathsf{HID}_j} A_{2,\mathsf{HID}_j})^s, (B^{\mathsf{com}}_{1,\mathsf{HID}_j} B_{2,\mathsf{HID}_j})^s)$
    $c_j \leftarrow \mathsf{Enc}(\mathsf{MPK}', \mathsf{HID}_j, \mathsf{com}\|m\|\mathsf{decom})$
for $j \coloneqq |\mathsf{Cov}| + 1$ **to** L **do**
    $\overline{c}_j \leftarrow\$\ \{0,1\}^\lambda$, $c_j \leftarrow\$\ \{0,1\}^{\ell(3\lambda+1+|m|)}$
$\hat{c} \coloneqq \overline{c}_0\|\overline{c}_1\|c_1\| \ldots \|\overline{c}_L\|c_L$
$\sigma \coloneqq F(\hat{k}, \hat{c}), c \coloneqq \sigma\|\hat{c}\|\mathsf{com}$
return c

Algorithm: $\mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$

$\triangleright$ HID_i – the HID of leaf i in $\mathcal{T}$
for $z \coloneqq 1$ **to** $n+1$ **do**
    $\overline{sk}_{i,z} \coloneqq (a_{1,\mathsf{HID}_{i|z}}, a_{2,\mathsf{HID}_{i|z}}, b_{1,\mathsf{HID}_{i|z}}, b_{2,\mathsf{HID}_{i|z}})$
    $sk_{i,z} \leftarrow \mathsf{Ext}(\mathsf{MPK}', \mathsf{MSK}', \mathsf{HID}_{i|z})$
$sk_i \coloneqq ((\overline{sk}_{i,1}, sk_{i,1}), \ldots, (\overline{sk}_{i,n+1}, sk_{i,n+1}))$
return sk_i

Algorithm: $\mathsf{Decrypt}(\mathsf{MPK}, sk_i, c)$

parse sk_i **as** $((\overline{sk}_{i,1}, sk_{i,1}), \ldots, (\overline{sk}_{i,n+1}, sk_{i,n+1}))$
parse c **as** $\sigma\|\hat{c}\|\mathsf{com}$
parse $\hat{c}$ **as** $\overline{c}_0\|\overline{c}_1\|c_1\| \ldots \|\overline{c}_L\|c_L$
$\tilde{c}_0 \coloneqq \mathsf{mp}^{-1}(\overline{c}_0)$
for $z \coloneqq 1$ **to** $n+1$ **do**
    parse $\overline{sk}_{i,z}$ **as** $(\tilde{a}_{1,z}, \tilde{a}_{2,z}, \tilde{b}_{1,z}, \tilde{b}_{2,z})$
    $tag_z \coloneqq H(\tilde{c}_0^{\tilde{a}_{1,z}\mathsf{com}+\tilde{a}_{2,z}}, \tilde{c}_0^{\tilde{b}_{1,z}\mathsf{com}+\tilde{b}_{2,z}})$
if $\exists z \in [1, n+1]\ \exists j \in [1, L] : tag_z = \overline{c}_j$ **then**
    $m' \coloneqq \mathsf{Dec}(\mathsf{MPK}', sk_{i,z}, c_j)$
    if $m' \neq \perp$ **then**
        parse m' **as** $\overline{\mathsf{com}}\|m\|\mathsf{decom}$
        if $\overline{\mathsf{com}} = \mathsf{com}$ **then**
            $\hat{k} \coloneqq \mathsf{Open}(\mathsf{PK}'', \mathsf{com}, \mathsf{decom})$
            if $\hat{k} \neq \perp \wedge \sigma = F(\hat{k}, \hat{c})$ **then**
                return m
return $\perp$

Figure 4: The oABE\$-CCA-secure construction. $\mathcal{T}$ is the perfect binary tree with $N = 2^n$ leaves, which represent the users in the system. $\mathsf{HID}_{i|z}$ denotes a prefix of the hierarchical identifier HID_i with length z.

Function: $\mathsf{Sample}(\lambda, h, H, c)$
Input: parameter λ, history h, function H, bit-string c
Output: stegotext s

$l \coloneqq |c|$
for $i \coloneqq 1$ **to** l **do**
  $j \coloneqq 0$
  repeat
    $j \coloneqq j + 1$, $s_i \leftarrow \mathfrak{C}_h$
  until $H(s_i) = c_i \vee j = \lambda$
  $h \coloneqq h \| s_i$
$s \coloneqq s_1 \| \ldots \| s_l$
return s

(a) Regular

Function: $\mathsf{DSample}(\lambda, H, c, r)$
Input: parameter λ, function H, bit-string c, randomness r
Output: stegotext s

$l \coloneqq |c|$
for $i \coloneqq 1$ **to** l **do**
  $j \coloneqq 0$
  repeat
    $j \coloneqq j + 1$, $s_i \coloneqq \mathsf{Channel}(r^{\lambda}_{\lambda(i-1)+j})$
  until $H(s_i) = c_i \vee j = \lambda$
$s \coloneqq s_1 \| \ldots \| s_l$
return s

(b) Deterministic

Figure 5: The rejection-sampler functions.

Theorem 4.5 (Proof in App. B): *If F is a (t_1, ϵ_1)-hard PRF, Π' is $(t_2, Q_{sk}, Q_d, \epsilon_2)$-AIBE\$-CCA-secure, $\mathcal{H}_{es}$ is a (t_3, ϵ_3)-entropy smoothing hash function, and DDH is (t_4, ϵ_4)-hard in $\mathbb{G}$, then the construction given in Fig. 4 is $(t_1 + t_2 + t_3 + t_4, Q_{sk}, Q_d, (\epsilon_1 + \epsilon_2 + \epsilon_3 + 2(\epsilon_4 + \frac{Q_d}{q}))r \log(\frac{N}{r}))$-oABE\$-CCA-secure, where N is the total number of users and r is the number of revoked users.*□

5 Constructions of Public-Key Broadcast Steganography

We now present three constructions of broadcast steganography: one for each model of security defined in Sect. 3.2. Our constructions employ the encrypt-then-embed paradigm depicted in Fig. 2, using oABE\$ (Sect. 4) for encryption and rejection-sampling [3, 29, 44] for embedding. In what follows, s_i^{σ} denotes the i^{th} leftmost non-overlapping substring with length σ of a given bit-string s.

5.1 A BS-CHA-Secure Construction

The rejection-sampler function used in our first construction is given in Fig. 5a. Sample takes as input a security parameter λ, a channel history $h \in \Sigma^*$, a function $H : \Sigma \to \{0, 1\}$, and a bit-string $c \in \{0, 1\}^*$, and outputs a covertext $s \in \Sigma^*$. Internally, for every bit c_i, Sample attempts to find a covertext $s_i^{\sigma} \in \Sigma$ such that $H(s_i^{\sigma}) = c_i$ by repeatedly querying the channel oracle up to λ number of times.[3] This mechanism allows a simple method to extract c from s: compute $c = H(s_1^{\sigma}) \| \ldots \| H(s_l^{\sigma})$ where $l = |s|/\sigma$. As shown in [4, 44], if the channel is always informative, H is a strongly universal hash function, and c is uniformly random, then the maximum statistical distance between $s_1 \leftarrow \mathsf{Sample}(\lambda, h, H, c)$ and $s_2 \leftarrow \mathfrak{C}_h^{|c|}$ for any valid $h \in \Sigma^*$ is negligible in the security parameter λ. For simplicity, we denote this statistical distance when $|c| = 1$ by ϵ_1 in the reminder of the paper.

We obtain our BS-CHA-secure scheme by combining the rejection-sampler function from Fig. 5a with our oABE\$ scheme (*cf.* Sect. 4). Formally, given a strongly universal hash function family $\mathcal{H}_{su} = \{H : \Sigma \to \{0, 1\}\}$ and an oABE\$-CPA-secure oABE\$ scheme $\Pi' = (\mathsf{Setup}', \mathsf{KeyGen}', \mathsf{Encrypt}',$

[3] Sample may fail to find a valid s_i during the λ iterations, but only with negligible probability in the parameter λ.

Algorithm: $\mathsf{Setup}(1^\lambda, N)$
1 $(\mathsf{MPK}', \mathsf{MSK}') \leftarrow \mathsf{Setup}'(1^\lambda, N)$
2 $H \leftarrow\!\!\$\ \mathcal{H}_{su}$
3 $\mathsf{MPK} \coloneqq (\mathsf{MPK}', H)$
4 $\mathsf{MSK} \coloneqq \mathsf{MSK}'$
5 **return** $(\mathsf{MPK}, \mathsf{MSK})$

Algorithm: $\mathsf{Encode}(\mathsf{MPK}, S, h, m)$
1 $c \leftarrow \mathsf{Encrypt}'(\mathsf{MPK}', S, m)$
2 $s \leftarrow \mathsf{Sample}(\lambda, h, H, c)$
3 **return** s

Algorithm: $\mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$
1 $sk_i \leftarrow \mathsf{KeyGen}'(\mathsf{MPK}', \mathsf{MSK}', i)$
2 **return** sk_i

Algorithm: $\mathsf{Decode}(\mathsf{MPK}, sk_i, s)$
1 $l \coloneqq |s|/\sigma$
2 **for** $j \coloneqq 1$ **to** l **do**
3 $\quad c_j \coloneqq H(s_j^\sigma)$
4 $c \coloneqq c_1 \| \ldots \| c_l$
5 $m \coloneqq \mathsf{Decrypt}'(\mathsf{MPK}', sk_i, c)$
6 **return** m

Figure 6: The BS-CHA-secure construction.

$\mathsf{Decrypt}'$) with expansion ℓ (*i.e.*, $|\mathsf{Encrypt}'(\mathsf{MPK}', S, m)| = \ell(|m|)$), we construct a BS-CHA-secure broadcast steganography scheme $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encode}, \mathsf{Decode})$ as shown in Fig. 6.

Theorem 5.1 (Proof in App. C): *If the channel is always informative, $\mathcal{H}_{su}$ is a strongly universal hash function family, and Π' is $(t_2, Q_{sk}, \epsilon_2)$-oABE\$-CPA-secure, then the construction in Fig. 6 is $(t_2, Q_{sk}, \mu\epsilon_1 + \epsilon_2)$-BS-CHA-secure, where μ is the polynomial bound on the total message length.* □

Remark 5.1. If the oABE\$ scheme employed in Fig. 6 is oABE\$-PDR-CCA-secure, then the resulting broadcast steganography scheme is BS-PDR-CCA-secure.

5.2 A BS-CCA-Secure Construction

Unfortunately, our first construction fails to provide a BS-CCA-secure broadcast steganography scheme even if the oABE\$ scheme internally used provides oABE\$-CCA security. The problem is that the rejection-sampler function from Fig. 5a allows multiple covertexts corresponding to a given bit-string. However, this limitation can be overcome in the case of channels that are efficiently computable and whose samples are independently distributed. In fact, for channels of this type, Hopper [27] devised a *deterministic* rejection-sampler function $\mathsf{DSample}$ that maps a given bit-string to exactly one covertext.

As shown in Fig. 5b, $\mathsf{DSample}$ takes in input a security parameter λ, a predicate $H : \Sigma \to \{0,1\}$ along with a bit-string $c \in \{0,1\}^*$ to embed, and a random bit-string $r \in \{0,1\}^{|c| \cdot \lambda^2}$ that controls the embedding. To sample $s \in \Sigma^*$, for every bit c_i of c, $\mathsf{DSample}$ seeks $s_i^\sigma \in \Sigma$ such that $H(s_i^\sigma) = c_i$, by repeatedly drawing from the channel according to the random chunks specified in r. This approach requires that the channel be efficiently computable by a function $\mathsf{Channel}(\cdot)$ whose samples are independent of the history (hence we drop h from its input), but guarantees that an adversary who intercepts a stegotext is unable to tweak it meaningfully. Furthermore, as shown in [4, 28, 44], if H is a strongly universal hash function, and c and r are uniformly random, then the statistical distance between stegotexts produced by $\mathsf{DSample}$ and innocent covertexts sampled from $\mathsf{Channel}(\cdot)$ is a negligible function ϵ_1 of λ.

Figure 7 reports the details of our BS-IND-CCA-secure scheme $\Pi = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{Encode}, \mathsf{Decode})$, based on a strongly universal hash function family $\mathcal{H}_{su}$, a variable-length pseudorandom generator (vPRG) $G : \{0,1\}^\lambda \times \mathbb{Z} \to \{0,1\}^*$ (whose second input sets the output length), and an oABE\$-IND-CCA-secure scheme $\Pi' = (\mathsf{Setup}', \mathsf{KeyGen}', \mathsf{Encrypt}', \mathsf{Decrypt}')$ with expansion ℓ.

Algorithm: $\mathsf{Setup}(1^\lambda, N)$
$(\mathsf{MPK}', \mathsf{MSK}') \leftarrow \mathsf{Setup}'(1^\lambda, N)$
$H \leftarrow\$ \mathcal{H}_{su}$
$\mathsf{MPK} \coloneqq (\mathsf{MPK}', H, G)$
$\mathsf{MSK} \coloneqq \mathsf{MSK}'$
return $(\mathsf{MPK}, \mathsf{MSK})$

Algorithm: $\mathsf{Encode}(\mathsf{MPK}, S, m)$
$\hat{r} \leftarrow\$ \{0,1\}^\lambda$
$c \leftarrow \mathsf{Encrypt}'(\mathsf{MPK}', S, \hat{r}\|m)$
$r \coloneqq G(\hat{r}, |c| \cdot \lambda^2)$
$s \coloneqq \mathsf{DSample}(\lambda, H, c, r)$
return s

Algorithm: $\mathsf{KeyGen}(\mathsf{MPK}, \mathsf{MSK}, i)$
$sk_i \leftarrow \mathsf{KeyGen}'(\mathsf{MPK}', \mathsf{MSK}', i)$
return sk_i

Algorithm: $\mathsf{Decode}(\mathsf{MPK}, sk_i, s)$
$l \coloneqq |s|/\sigma$
for $j \coloneqq 1$ **to** l **do**
    $c_j \coloneqq H(s_j^\sigma)$
$c \coloneqq c_1\| \ldots \|c_l$
$m' \coloneqq \mathsf{Decrypt}'(\mathsf{MPK}', sk_i, c)$
if $m' \neq \perp$ **then**
    parse m' **as** $\hat{r}\|m$
    $r \coloneqq G(\hat{r}, l \cdot \lambda^2)$
    if $\mathsf{DSample}(\lambda, H, c, r) = s$ **then**
        return m
return $\perp$

Figure 7: The BS-CCA-secure construction.

Theorem 5.2 (Proof in App. D): *If the channel is always informative, $\mathcal{H}_{su}$ is a strongly universal hash function family, G is a (t_2, ϵ_2)-hard vPRG, and Π' is $(t_3, Q_{sk}, Q_d, \epsilon_3)$-oABE\$-CCA-secure, then the construction in Fig. 7 is $(t_2 + t_3, Q_{sk}, Q_d, \mu\epsilon_1 + \epsilon_2 + \epsilon_3)$-BS-CCA-secure, where μ is the polynomial bound on the total message length.* □

6 Extensions and Future Work

As in the case of broadcast encryption, one may consider extensions of the notion of broadcast steganography that enhance the setting discussed in this paper with additional functionality or security properties. In particular, while broadcast steganography natively protects the recipients' identities from outsiders, it does not aim to prevent recipients from finding out about each other. The natural solution for that is *anonymous* broadcast steganography (AnoBS). By extending the anonymous broadcast encryption schemes of [5,36] to support ciphertext pseudorandomness, we can use them in place of our oABE\$ to achieve fully anonymous broadcast steganography. The resulting AnoBS scheme, however, would have ciphertexts with length *linear* in the number of receivers.

Acknowledgments

Nelly Fazio's research is sponsored in part by NSF CAREER award #1253927 and NSF award #1117675, and by PSC-CUNY award 64578-00 42 (jointly funded by the Professional Staff Congress and the City University of New York). Nelly Fazio and Irippuge Milinda Perera are supported in part by the U.S. Army Research Laboratory and the U.K. Ministry of Defence under Agreement Number W911NF-06-3-0001. Antonio Nicolosi's research is sponsored in part by NSF awards #1117679 and #1040784. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints

for Government purposes notwithstanding any copyright notation hereon.

References

[1] M. Abdalla, M. Bellare, D. Catalano, E. Kiltz, T. Kohno, T. Lange, J. Malone-Lee, G. Neven, P. Paillier, and H. Shi. Searchable encryption revisited: Consistency properties, relation to Anonymous IBE, and extensions. In *Advances in Cryptology—CRYPTO*, pages 205–222, 2005.

[2] S. Agrawal and X. Boyen. Identity-based encryption from lattices in the standard model. Manuscript, 2009. `http://www.cs.stanford.edu/~xb/ab09/`.

[3] R. Anderson and F. Petitcolas. On the limits of steganography. *IEEE Journal on Selected Areas in Communications*, 16(4):474–481, 1998.

[4] M. Backes and C. Cachin. Public-key steganography with active attacks. In *Theory of Cryptography—TCC*, pages 210–226, 2005.

[5] A. Barth, D. Boneh, and B. Waters. Privacy in encrypted content distribution using private broadcast encryption. In *Financial Cryptography and Data Security—FC*, pages 52–64, 2006.

[6] S. Berkovits. How to broadcast a secret. In *Advances in Cryptology—EUROCRYPT*, pages 535–541, 1991.

[7] D. Boneh, C. Gentry, and B. Waters. Collusion resistant broadcast encryption with short ciphertexts and private keys. In *Advances in Cryptology—CRYPTO*, pages 258–275, 2005.

[8] D. Boneh and K. Jonathan. Improved efficiency for CCA-secure cryptosystems built using identity-based encryption. In *Topics in Cryptology—CT-RSA*, pages 87–103, 2005.

[9] D. Boneh and B. Waters. A fully collusion resistant broadcast, trace, and revoke system. In *ACM Conference on Computer and Communications Security—CCS*, pages 211–220, 2006.

[10] X. Boyen and B. Waters. Anonymous hierarchical identity-based encryption (without random oracles). In *Advances in Cryptology—CRYPTO*, pages 290–307, 2006.

[11] C. Cachin. An information-theoretic model for steganography. *Information and Computation*, 192(1):41–56, 2004.

[12] D. Cash, E. Kiltz, and V. Shoup. The twin Diffie-Hellman problem and applications. In *Advances in Cryptology—EUROCRYPT*, pages 127–145, 2008.

[13] N. Chandran, V. Goyal, R. Ostrovsky, and A. Sahai. Covert multi-party computation. In *IEEE Symposium on Foundations of Computer Science—FOCS*, pages 238–248, 2007.

[14] N. Dedic, G. Itkis, L. Reyzin, and S. Russell. Upper and Lower Bounds on Black-Box Steganography. *Journal of Cryptology*, 22(3):365–394, 2009.

[15] Y. Dodis and N. Fazio. Public-key broadcast encryption for stateless receivers. In *Digital Rights Management—DRM*, pages 61–80, 2002.

[16] Y. Dodis and N. Fazio. Public-key trace and revoke scheme secure against adaptive chosen ciphertext attack. In *Public Key Cryptography—PKC*, pages 100–115, 2003.

[17] Y. Dodis, N. Fazio, A. Kiayias, and M. Yung. Scalable public-key tracing and revoking. In *ACM Symposium on Principles of Distributed Computing—PODC*, pages 190–199, 2003. Invited to the Special Issue of Journal of Distributed Computing PODC 2003.

[18] Y. Dodis, N. Fazio, A. Lysyanskaya, and D. Yao. ID-based encryption for complex hierarchies with applications to forward security and broadcast encryption. In *ACM Conference on Computer and Communications Security—CCS*, pages 354–363, 2004.

[19] Y. Dodis and J. Katz. Chosen-ciphertext security of multiple encryption. In *Theory of Cryptography—TCC*, pages 188–209, 2005.

[20] N. Fazio and I. M. Perera. Outsider-anonymous broadcast encryption with sublinear ciphertexts. In *Public Key Cryptography—PKC*, pages 225–242, 2012.

[21] N. Fazio and I. M. Perera. Outsider-anonymous broadcast encryption with sublinear ciphertexts. Cryptology ePrint Archive, Report 2012/129, 2012. Full Version of [20].

[22] A. Fiat and M. Naor. Broadcast encryption. In *Advances in Cryptology—CRYPTO*, pages 480–491, 1993.

[23] J. A. Garay, J. Staddon, and A. Wool. Long-lived broadcast encryption. In *Advances in Cryptology—CRYPTO*, pages 333–352, 2000.

[24] C. Gentry. Practical identity-based encryption without random oracles. In *Advances in Cryptology—EUROCRYPT*, pages 445–464, 2006.

[25] C. Gentry and B. Waters. Adaptive security in broadcast encryption systems (with short ciphertexts). In *Advances in Cryptology—EUROCRYPT*, pages 171–188, 2009.

[26] D. Halevy and A. Shamir. The LSD broadcast encryption scheme. In *Advances in Cryptology—CRYPTO*, pages 47–60, 2002.

[27] N. J. Hopper. *Toward a Theory of Steganography.* PhD thesis, Carnegie Mellon University, 2004.

[28] N. J. Hopper. On steganographic chosen covertext security. In *Automata, Languages and Programming—ICALP*, pages 311–323, 2005.

[29] N. J. Hopper, J. Langford, and L. von Ahn. Provably Secure Steganography. In *Advances in Cryptology—CRYPTO*, pages 77–92, 2002.

[30] R. Impagliazzo and D. Zuckerman. How to recycle random bits. In *IEEE Symposium on Foundations of Computer Science—FOCS*, pages 248–253, 1989.

[31] S. Katzenbeisser and F. A. Petitcolas. Defining security in steganographic systems. In *Security and Watermarking of Multimedia Contents IV*, pages 50–56, 2002.

[32] A. Kiayias, Y. Raekow, and A. Russell. Efficient steganography with provable security guarantees. In *Information Hiding—IH*, pages 118–130, 2005.

[33] A. Kiayias, A. Russell, and N. Shashidhar. Key-efficient steganography with provable security guarantees. In *Information Hiding—IH*, pages 118–130, 2012.

[34] A. Kiayias and K. Samari. Lower bounds for private broadcast encryption. In *Information Hiding—IH*, pages 176–190, 2012.

[35] T. Le and K. Kurosawa. Efficient Public Key Steganography Secure Against Adaptive Chosen Stegotext Attacks. Cryptology ePrint Archive, Report 2003/244, 2003.

[36] B. Libert, K. G. Paterson, and E. A. Quaglia. Anonymous broadcast encryption. In *Public Key Cryptography—PKC*, pages 206–224, 2012.

[37] A. Lysyanskaya and M. Meyerovich. Provably Secure Steganography with Imperfect Sampling. In *Public Key Cryptography—PKC*, pages 123–139, 2006.

[38] W. Mazurczyk, M. Karas, and K. Szczypiorski. Skyde: A skype-based steganographic method, 2013. `arxiv.org/abs/1301.3632`.

[39] D. Naor, M. Naor, and J. Lotspiech. Revocation and tracing schemes for stateless receivers. In *Advances in Cryptology—CRYPTO*, pages 41–62, 2001.

[40] T. P. Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing. In *Advances in Cryptology—CRYPTO*, pages 129–140, 1991.

[41] L. Reyzin and S. Russell. Simple Stateless Steganography. Cryptology ePrint Archive, Report 2003/093, 2003.

[42] G. Simmons. The Prisoners' Problem and the Subliminal Channel. In *Advances in Cryptology—CRYPTO*, pages 51–67, 1983.

[43] The Economist. Speaking with silence, February 2013.

[44] L. von Ahn and N. J. Hopper. Public-key steganography. In *Advances in Cryptology—EUROCRYPT*, pages 323–341, 2004.

[45] L. von Ahn, N. J. Hopper, and J. Langford. Covert two-party computation. In *ACM Symposium on Theory of Computing—STOC*, pages 513–522, 2005.

[46] D. Zuckerman. General weak random sources. In *IEEE Symposium on Foundations of Computer Science—FOCS*, pages 534–543, 1990.

A Review of the Subset Cover Framework

The *subset cover* (SC) *framework* proposed by Naor *et al.* [39] is a system that abstracts a variety of revocation schemes in the private-key setting where only the Center can broadcast. In a nutshell, a revocation scheme belonging to the SC framework defines a collection of subsets $\mathcal{S}$ of the universe of users $U = [1, N]$ in the system. During the key generation phase, the Center assigns to each subset $S_i \in \mathcal{S}$ a long-lived key k_i, which is also given to each user belonging to S_i. When the Center wants to broadcast a message m, it generates a short-lived session key $\widehat{k}$, determines the set of revoked users R, finds a set of disjoint subsets $\widehat{\mathcal{S}}$ from $\mathcal{S}$ that contains or "covers" all the users in $U \backslash R$, encrypts $\widehat{k}$ using the long-lived keys corresponding to the subsets in $\widehat{\mathcal{S}}$, and finally broadcasts the encryption of m under $\widehat{k}$ and the encryptions of $\widehat{k}$ to all the users in the system. Upon receiving a broadcast ciphertext, a user can decrypt successfully and obtain m if and only if that user is part of the authorized set (*i.e.*, the user possesses a long-lived key corresponding to some subset of $\widehat{\mathcal{S}}$).

The authors in [39] also presented two concrete revocation schemes, namely the *complete subtree* (CS) method and the *subset difference* (SD) method. In the CS method, which is the simplest of the two, the ciphertext length is $O(r \log(\frac{N}{r}))$ and the secret key length at a receiver is $O(\log N)$, where r is the number of revoked users. In the SD method, the one with more involved computations, the ciphertext length reduces to $O(r)$ while the secret key length increases to $O(\log^2 N)$. Another crucial difference between the two schemes is that the assignment of the long-lived keys in the former is information-theoretic, whereas in the latter its computational. Below we provide a short description of the CS method, and we refer the reader to [39] for further details on the SD method.

Complete Subtree Method. In this scheme, the N users are represented as the leaves of a perfect binary tree $\mathcal{T}$ and the collection of subsets $\mathcal{S}$ contains all possible subtrees of $\mathcal{T}$. In case N is not a power of 2, some dummy users are added to the system. During the key generation phase, every subtree in $\mathcal{S}$ is assigned a long-lived secret key which is also made available to all the users belonging to that subtree. Since every user is a member of all the subtrees rooted at each node in the path from the root of $\mathcal{T}$ down to the leaf corresponding to that user, the secret key length at a user is $O(\log N)$. The ciphertext length becomes $O(r \log(\frac{N}{r}))$ due to the fact that it requires on average a logarithmic number of subtrees to revoke r users (see [39] for a formal analysis).

Extension of the SC Framework to the Public-Key Setting. The original SC framework was defined in the private-key setting. In [15], Dodis and Fazio extended the SC framework to the public-key setting by combining a novel assignment of hierarchical identifiers (HIDs) to the nodes in $\mathcal{T}$ with (hierarchical) identity-based encryption ((H)IBE). For completeness, we only explain below the extension of the CS method. We refer to [15] for the specifics regarding the SD method.

The assignment of HIDs to the nodes in $\mathcal{T}$ goes as follows. First, the root of $\mathcal{T}$ is assigned a special identifier denoted by ε. Next, each edge e of $\mathcal{T}$ is assigned the identifier $\mathsf{ID}_e \in \{0,1\}$ depending on whether the edge connects to the left child or to the right child. Then, the hierarchical identifier HID_v of any node v can be computed by concatenating all the identifiers starting from the root of $\mathcal{T}$ down to v (*i.e.*, $\mathsf{HID}_v := \varepsilon \| \mathsf{ID}_{e_1} \| \ldots \| \mathsf{ID}_{e_{\log N}}$). It is important to note that any prefix of HID_v represents a valid HID of an ancestor of v.

Once the HIDs of the nodes are assigned, the authors employ an IBE scheme in order to encrypt the short-lived session keys during broadcasts. The long-lived keys of the subsets in $\mathcal{S}$ now become the IBE keys corresponding to the HIDs of the nodes in $\mathcal{T}$. Since the structure of the $\mathcal{T}$ and the assignment of HIDs are publicly known to all the users, any user in the system can be a sender as well as a receiver. In the public-key setting, the Center becomes the trusted authority that provides each user with the required IBE keys.

B Proof of Theorem 4.5

Proof. We organize our proof as a sequence of games $(\mathrm{Game}_0, \overline{\mathrm{Game}}_1, \mathrm{Game}_1, \ldots, \overline{\mathrm{Game}}_l, \mathrm{Game}_l)$ between an oABE\$-IND-CCA adversary $\mathcal{A}$ and the challenger $\mathcal{C}$, where l denotes the cardinality of the coverset Cov induced by the set of authorized receivers S^* chosen by $\mathcal{A}$ during the *Challenge* phase of the oABE\$-IND-CCA game. In the first game (Game_0), $\mathcal{A}$ receives an encryption of m^* for S^* in the *Challenge* phase, and in the last game (Game_l), $\mathcal{A}$ receives a uniformly random bit-string of the appropriate length as the challenge ciphertext.

Game_0: corresponds to the game given in Definition 4.1 when the challenge bit b^* is fixed to 0. The interaction between $\mathcal{A}$ and $\mathcal{C}$ during *Setup*, *Phase 1*, *Phase 2*, and *Guess* follows exactly as specified in Definition 4.1. During the *Challenge* phase, $\mathcal{A}$ gives $\mathcal{C}$ a message $m^* \in \mathcal{MSP}$ and a set of user identities $S^* \subseteq U$ with the restriction that $S^* \cap R = \emptyset$, where R is the set of users that $\mathcal{A}$ corrupted during *Phase 1*. $\mathcal{C}$ computes the challenge ciphertext c^*, which is subsequently sent to $\mathcal{A}$, as follows:

1 $r := N - |S^*|$, $L := \lfloor r \log(\frac{N}{r}) \rfloor$
2 $(\hat{k}, \mathsf{com}, \mathsf{decom}) \leftarrow \mathsf{Commit}(\mathsf{PK}'')$
3 **repeat** $s \leftarrow\$ \mathbb{Z}_q$, $\overline{c}_0 := \mathsf{mp}(g^s)$ **until** $\overline{c}_0 < 2^\lambda$
4 **for** $j := 1$ **to** l **do**
5 $\quad \overline{c}_j := H((A^{\mathsf{com}}_{1,\mathsf{HID}_j} A_{2,\mathsf{HID}_j})^s, (B^{\mathsf{com}}_{1,\mathsf{HID}_j} B_{2,\mathsf{HID}_j})^s)$
6 $\quad c_j \leftarrow \mathsf{Enc}(\mathsf{MPK}', \mathsf{HID}_j, \mathsf{com} \| m^* \| \mathsf{decom})$
7 **for** $j := l+1$ **to** L **do**
8 $\quad \overline{c}_j \leftarrow\$ \{0,1\}^\lambda$
9 $\quad c_j \leftarrow\$ \{0,1\}^{\ell(3\lambda+1+|m^*|)}$
10 $\hat{c} := \overline{c}_0 \| \overline{c}_1 \| c_1 \| \ldots \| \overline{c}_L \| c_L$
11 $\sigma := F(\hat{k}, \hat{c})$, $c^* := \sigma \| \hat{c} \| \mathsf{com}$

$\overline{\mathrm{Game}}_h$ ($1 \le h \le l$): is similar to Game_{h-1}, but, when creating c^*, $\mathcal{C}$ replaces Lines 4–9 with:

$1'$ **for** $j \coloneqq 1$ **to** $l-h$ **do**
$2'$ $\quad \overline{c}_j \coloneqq H((A^{\mathsf{com}}_{1,\mathsf{HID}_j} A_{2,\mathsf{HID}_j})^s, (B^{\mathsf{com}}_{1,\mathsf{HID}_j} B_{2,\mathsf{HID}_j})^s)$
$3'$ $\quad c_j \leftarrow \mathsf{Enc}(\mathsf{MPK}', \mathsf{HID}_j, \mathsf{com} \| m^* \| \mathsf{decom})$
$4'$ $\overline{c}_{l-h+1} \coloneqq H((A^{\mathsf{com}}_{1,\mathsf{HID}_{l-h+1}} A_{2,\mathsf{HID}_{l-h+1}})^s, (B^{\mathsf{com}}_{1,\mathsf{HID}_{l-h+1}} B_{2,\mathsf{HID}_{l-h+1}})^s)$
$5'$ $c_{l-h+1} \leftarrow\!\!\$\ \{0,1\}^{\ell(3\lambda+1+|m^*|)}$
$6'$ **for** $j \coloneqq l-h+2$ **to** L **do**
$7'$ $\quad \overline{c}_j \leftarrow\!\!\$\ \{0,1\}^{\lambda}$
$8'$ $\quad c_j \leftarrow\!\!\$\ \{0,1\}^{\ell(3\lambda+1+|m^*|)}$

$\mathbf{Game}_h$($1 \le h \le l$): is similar to $\overline{\mathrm{Game}}_h$, but, when creating c^*, $\mathcal{C}$ replaces Lines $4'$–$8'$ with:

$1''$ **for** $j \coloneqq l-h+1$ **to** L **do**
$2''$ $\quad \overline{c}_j \leftarrow\!\!\$\ \{0,1\}^{\lambda}$
$3''$ $\quad c_j \leftarrow\!\!\$\ \{0,1\}^{\ell(3\lambda+1+|m^*|)}$

For $0 \le i_1 \le l$ and $1 \le i_2 \le l$ let $\mathsf{Adv}^{i_1}_{\mathcal{A},\Pi}$ and $\overline{\mathsf{Adv}}^{i_2}_{\mathcal{A},\Pi}$ denote $\mathcal{A}$'s advantage in winning Game_{i_1} and $\overline{\mathrm{Game}}_{i_2}$ respectively. In Lemma B.1, we show that if the underlying PRF F is (t_1, ϵ_1)-hard and the AIBE\$ scheme Π' is $(t_2, Q_{sk}, Q_d, \epsilon_2)$-AIBE\$-CCA-secure, then $\mathcal{A}$'s advantage of distinguishing Game_{h-1} from $\overline{\mathrm{Game}}_h$ is at most $\epsilon_1 + \epsilon_2$. In Lemma B.2, we show that if $\mathcal{H}_{es}$ is an (t_2, ϵ_2)-entropy smoothing family of hash functions and DDH is (t_4, ϵ_4)-hard in $\mathbb{G}$, then $\mathcal{A}$ has at most $\epsilon_3 + 2(\epsilon_4 + \frac{Q_d}{q})$ advantage in distinguishing $\overline{\mathrm{Game}}_h$ from Game_h. Therefore,

$$\left|\mathsf{Adv}^0_{\mathcal{A},\Pi} - \mathsf{Adv}^l_{\mathcal{A},\Pi}\right| \le \left(\epsilon_1 + \epsilon_2 + \epsilon_3 + 2\left(\epsilon_4 + \frac{Q_d}{q}\right)\right) l \le \left(\epsilon_1 + \epsilon_2 + \epsilon_3 + 2\left(\epsilon_4 + \frac{Q_d}{q}\right)\right) L$$
$$\le \left(\epsilon_1 + \epsilon_2 + \epsilon_3 + 2\left(\epsilon_4 + \frac{Q_d}{q}\right)\right) r \log\left(\frac{N}{r}\right). \blacksquare$$

Lemma B.1: *For $1 \le h \le l$, if the underlying PRF F is (t_1, ϵ_1)-hard and the AIBE\$ scheme Π' is $(t_2, Q_{sk}, Q_d, \epsilon_2)$-AIBE\$-CCA-secure, then $\mathcal{A}$'s advantage of distinguishing Game_{h-1} from $\overline{\mathrm{Game}}_h$ is at most $\epsilon_1 + \epsilon_2$.* □

Proof. We build a PPT adversary $\mathcal{B}$ that internally runs the oABE\$-IND-CCA game with the adversary $\mathcal{A}$ in order to gain advantage in the AIBE\$-IND-CCA game with the challenger $\mathcal{C}'$. We denote the secret-key oracle and the decryption oracle of $\mathcal{C}'$ by $\mathcal{O}'_{sk}(\cdot)$ and $\mathcal{O}'_d(\cdot,\cdot)$ respectively. After receiving the master public key MPK' of the AIBE\$ scheme from $\mathcal{C}'$, $\mathcal{B}$ executes the oABE\$-IND-CCA game with $\mathcal{A}$ as follows:

Setup: $\mathcal{B}$ generate MPK, which he eventually sends to $\mathcal{A}$, by executing Lines 2–10 of the Setup algorithm of Fig. 4. $\mathcal{B}$ also keeps the exponents $\{a_{i,\mathsf{HID}_j}, b_{i,\mathsf{HID}_j}\}_{i\in\{1,2\}, j\in[1,|\mathsf{Fam}|]}$ to himself and initializes the set of revoked users R to be empty.

Phase 1: When $\mathcal{A}$ invokes a secret-key query for user i, $\mathcal{B}$ computes the secret key sk_i by executing lines Lines 1–6 of the KeyGen algorithm of Fig. 4 with one modification: during Line 4, $\mathcal{B}$ sets $sk_{i,z} \leftarrow \mathcal{O}'_{sk}(\mathsf{HID}_{i|z})$. Next, after adding i to R, $\mathcal{B}$ sends sk_i to $\mathcal{A}$.

When $\mathcal{A}$ invokes a decryption query (i, c), $\mathcal{B}$ computes the hierarchical identifier of leaf i in $\mathcal{T}\mathsf{HID}_i$ and proceeds as follows:

1 **parse** c **as** $\sigma \| \hat{c} \| \mathsf{com}$

parse $\hat{c}$ **as** $\overline{c}_0\|\overline{c}_1\|c_1\|\ldots\|\overline{c}_L\|c_L$
$\tilde{c}_0 \coloneqq \mathsf{mp}^{-1}(\overline{c}_0)$
for $z \coloneqq 1$ **to** $n+1$ **do**
    $\tilde{a}_{1,z} \coloneqq a_{1,\mathsf{HID}_{i|z}}$, $\tilde{a}_{2,z} \coloneqq a_{2,\mathsf{HID}_{i|z}}$ $\tilde{b}_{1,z} \coloneqq b_{1,\mathsf{HID}_{i|z}}$, $\tilde{b}_{2,z} \coloneqq b_{2\mathsf{HID}_{i|z}}$
    $tag_z \coloneqq H(\tilde{c}_0^{\tilde{a}_{1,z}\mathsf{com}+\tilde{a}_{2,z}}, \tilde{c}_0^{\tilde{b}_{1,z}\mathsf{com}+\tilde{b}_{2,z}})$
if $\exists z \in [1,n+1]\ \exists j \in [1,L] : tag_z = \overline{c}_j$ **then**
    $m' \coloneqq \mathcal{O}'_d(\mathsf{HID}_{i|k}, c_j)$
    if $m' \neq \perp$ **then**
        parse m' **as** $\overline{\mathsf{com}}\|m\|\mathsf{decom}$
        if $\overline{\mathsf{com}} = \mathsf{com}$ **then**
            $\hat{k} \coloneqq \mathsf{Open}(\mathsf{PK}'', \mathsf{com}, \mathsf{decom})$
            if $\hat{k} \neq \perp \wedge \sigma = F(\hat{k}, \hat{c})$ **then**
                return m
return $\perp$

Challenge: After receiving from $\mathcal{A}$ a message $m^* \in \mathcal{MSP}$ and a set of user identities $S^* \subseteq U$ with the restriction that $S^* \cap R = \emptyset$, $\mathcal{B}$ picks $(\hat{k}, \mathsf{com}, \mathsf{decom}) \leftarrow \mathsf{Commit}(\mathsf{PK}'')$ and sets

$$\mathsf{ID}' \coloneqq \mathsf{HID}_{l-h+1}, \quad m' \coloneqq \mathsf{com}\|m^*\|\mathsf{decom}.$$

Next, $\mathcal{B}$ sends the identity ID' and the messages m' as the challenge query to $\mathcal{C}'$. Then, $\mathcal{C}'$ picks a random bit $b' \in \{0,1\}$ and generates the challenge ciphertext c' depending on it: if $b' = 0$, then $c' \leftarrow \mathsf{Enc}(\mathsf{MPK}', \mathsf{ID}', \mathsf{com}\|m^*\|\mathsf{decom})$, else $c' \leftarrow\$ \{0,1\}^{\ell(|m'|)}$, and returns c' to $\mathcal{B}$. Finally, $\mathcal{B}$ computes the challenge ciphertext c^*, which is eventually sent to $\mathcal{A}$, as follows:

$r \coloneqq N - |S^*|$, $L \coloneqq \lfloor r \log(\frac{N}{r}) \rfloor$
repeat $s \leftarrow\$ \mathbb{Z}_q$, $\overline{c}_0 \coloneqq \mathsf{mp}(g^s)$ **until** $\overline{c}_0 < 2^\lambda$
for $j \coloneqq 1$ **to** $l-h$ **do**
    $\overline{c}_j \coloneqq H((A^{\mathsf{com}}_{1,\mathsf{HID}_j} A_{2,\mathsf{HID}_j})^s, (B^{\mathsf{com}}_{1,\mathsf{HID}_j} B_{2,\mathsf{HID}_j})^s)$
    $c_j \leftarrow \mathsf{Enc}(\mathsf{MPK}', \mathsf{HID}_j, \mathsf{com}\|m^*\|\mathsf{decom})$
$\overline{c}_{l-h+1} \coloneqq H((A^{\mathsf{com}}_{1,\mathsf{HID}_{l-h+1}} A_{2,\mathsf{HID}_{l-h+1}})^s, (B^{\mathsf{com}}_{1,\mathsf{HID}_{l-h+1}} B_{2,\mathsf{HID}_{l-h+1}})^s)$
$c_{l-h+1} \coloneqq c'$
for $j \coloneqq l-h+2$ **to** L **do**
    $\overline{c}_j \leftarrow\$ \{0,1\}^\lambda$
    $c_j \leftarrow\$ \{0,1\}^{\ell(3\lambda+1+|m^*|)}$
$\hat{c} \coloneqq \overline{c}_0\|\overline{c}_1\|c_1\|\ldots\|\overline{c}_L\|c_L$
$\sigma \coloneqq F(\hat{k}, \hat{c})$, $c^* \coloneqq \sigma\|\hat{c}\|\mathsf{com}$

Phase 2: Secret-key queries are handled similarly to *Phase 1*, with the usual restriction that $\mathcal{A}$ does *not* invoke a secret-key query i such that $i \in S^*$.

As for decryption queries, $\mathcal{B}$ replies to $(i, c = \sigma\|\hat{c}\|\mathsf{com})$, according to one of the following cases:

- If $c = c^*$ and $i \notin S^*$, then $\mathcal{B}$ proceeds as in *Phase 1*. (Note that in this case $\mathcal{B}$'s output will be $\perp$, as it should be.)
- If $c = c^*$, and $i \in S^*$, $\mathcal{B}$ just rejects since $\mathcal{A}$ is submitting an invalid query.
- If $c \neq c^*$ and $i \notin S^*$, then $\mathcal{B}$ proceeds as in *Phase 1*.
- If $c \neq c^*$ and $i \in S^*$, then $\mathcal{B}$ computes HID_i and proceeds as follows:

- ◇ If for all $z = 1$ to $n+1$, it is the case that $\mathsf{HID}_{i|z} \neq \mathsf{HID}_{l-h+1}$, then $\mathcal{B}$ proceeds as in *Phase 1*. Observe that the condition $\forall z \in [1, n+1] : \mathsf{HID}_{i|z} \neq \mathsf{HID}_{l-h+1}$ ensures that the decryption query that $\mathcal{B}$ will make to its challenger $\mathcal{C}'$ in the process of responding to $\mathcal{A}$'s query is allowed.
- ◇ If $\exists\ z \in [1, n+1]$ such that $\mathsf{HID}_{i|z} = \mathsf{HID}_{l-h+1}$, and c' does not appear among the ciphertext components of c, then again $\mathcal{B}$ proceeds as in *Phase 1*. Observe that the condition that c does not contain c' ensures that also in this case the decryption query that $\mathcal{B}$ will make to its challenger $\mathcal{C}'$ in the process of responding to $\mathcal{A}$'s query is allowed.
- ◇ If $\exists\ z \in [1, n+1]$ such that $\mathsf{HID}_{i|z} = \mathsf{HID}_{l-h+1}$, but c' appears among the ciphertext components of c, then $\mathcal{B}$ outputs $\perp$. Arguing that this (*i.e.*, $\perp$) is the real reply that $\mathcal{A}$ would get in either $\overline{\text{Game}}_h$ or Game_h requires some care, but can be done along the lines of the proofs of [8] and [19]. In a nutshell, the issue is the circularity in the PRF usage: in generating the σ component of the ciphertext, $F(\hat{k}, \cdot)$ is computed over $\hat{c}$, which includes ciphertext components that contain com and decom, which in turn correlate with $\hat{k}$. The reason this circularity does not break the argument is that the appearance of $\hat{k}$ into the ciphertext is mediated by the relaxed commitment scheme. In particular, since com is included both in the clear and inside each ciphertext component (which are individually AIBE-CCA-secure as part of c^*), and since the decryption algorithm checks that they be consistent, the adversary is forced to keep in the outer layer of her query ciphertext c the same value of com that was in the challenge c^*, or decryption would fail. Now for that value of com, by the relaxed binding property, the only valid PRF key that can be decommitted is $\hat{k}$. At this point the argument would seem to get stuck again, as it is not apparent how to guarantee that the adversary does not learn enough about $\hat{k}$ from the several ciphertext components in c^* so as to be able to compute F-values under that key. As it turns out, this point can also be tamed through a separate sequence-of-games analysis [8]. It then follows that the adversary will not be able to compute the proper σ for the ciphertext she was trying to craft, which finally fully justifies the $\perp$ reply by the simulator.

Guess: $\mathcal{A}$ outputs a guess b and $\mathcal{B}$ passes this bit as his guess for b' to $\mathcal{C}'$.

Observe that, by construction, it holds that if $\mathcal{C}'$ chooses $b' = 0$, then $\mathcal{B}$ is playing Game_{h-1}, whereas if $b' = 1$, then $\mathcal{B}$ is playing $\overline{\text{Game}}_h$. Therefore, the PRF and the AIBE\$-IND-CCA advantage of B is essentially $\mathcal{A}$'s advantage in distinguishing Game_{h-1} from $\overline{\text{Game}}_h$: $\left|\mathsf{Adv}^{h-1}_{\mathcal{A},\Pi} - \overline{\mathsf{Adv}}^{h}_{\mathcal{A},\Pi}\right| \leq \epsilon_1 + \epsilon_2$. ■

Lemma B.2: *For $1 \leq h \leq l$, if $\mathcal{H}_{es}$ in an (t_3, ϵ_3)-entropy smoothing hash function family and DDH is (t_4, ϵ_4)-hard in $\mathbb{G}$, then $\mathcal{A}$'s advantage of distinguishing $\overline{Game}_h$ from $Game_h$ is at most $\epsilon_3 + (\epsilon_4 + \frac{Q_d}{q})$.* □

Proof Sketch. The proof of this lemma follow with the help of two intermediate games $\widetilde{\text{Game}}_{1,h}$ and $\widetilde{\text{Game}}_{2,h}$. During the transition from $\overline{\text{Game}}_h$ to $\widetilde{\text{Game}}_{1,h}$, we replace $(B^{\mathsf{com}}_{1,\mathsf{HID}_{l-h+1}} B_{2,\mathsf{HID}_{l-h+1}})^s$ with a random group element $r_2 \in \mathbb{G}$. Next, during the transition from $\widetilde{\text{Game}}_{1,h}$ to $\widetilde{\text{Game}}_{2,h}$, we replace

$(A^{\mathsf{com}}_{1,\widetilde{\mathsf{HID}_{l-h+1}}} A_{2,\mathsf{HID}_{l-h+1}})^s$ with another random group element $r_1 \in \mathbb{G}$. Finally, during the transition from $\mathrm{Game}_{2,h}$ to Game_h, we replace $H(r_1, r_2)$ with a truly random bit-string of length λ.

The idea of the proof of the first two transitions is to reduce from the DDH problem and build a PPT adversary $\mathcal{B}$ that internally executes the oABE\$-IND-CCA game with the adversary $\mathcal{A}$ in order to gain advantage in breaking the DDH assumption. This reduction argument proceeds along the same lines as Lemma 1 of [36]. As for the second transition, we employ the fact that $\mathcal{H}_{es}$ is an entropy smoothing hash function. ■

C Proof of Theorem 5.1

Proof. We organize the proof as a sequence of games (Game_0, Game_1, Game_2) between a BS-IND-CHA adversary $\mathcal{A}$ and a challenger $\mathcal{C}$. During the *Challenge* phase in Game_0, $\mathcal{A}$ is given a stegotext for m^* under S^*, whereas in Game_2, $\mathcal{A}$ is given a covertext consisting of some samples from the channel oracle.

Game_0: is the actual BS-IND-CHA game when the challenge bit b^* is fixed to 0. The interaction between $\mathcal{A}$ and $\mathcal{C}$ during *Setup*, *Phase 1*, *Phase 2*, and *Guess* follows as specified in Definition 3.3. During the *Challenge* phase, $\mathcal{A}$ sends $\mathcal{C}$ a message $m^* \in \mathcal{MSP}$, a legal history $h \in \Sigma^*$, and a set of user identities $S^* \subseteq U$ with the restriction that $S^* \cap R = \emptyset$. Next, $\mathcal{C}$ generates the challenge stegotext s^*, which is later sent to $\mathcal{A}$, as follows:

1 $c \leftarrow \mathsf{Encrypt}'(\mathsf{MPK}', S^*, m^*)$
2 $s^* \leftarrow \mathsf{Sample}(\lambda, h, H, c)$

Game_1: is similar to Game_0, but $\mathcal{C}$ computes the challenge s^* as follows:

1 $c \leftarrow\$ \{0,1\}^{\ell(|m^*|)}$
2 $s^* \leftarrow \mathsf{Sample}(\lambda, h, H, c)$

Game_2: is similar to Game_1, but $\mathcal{C}$ now computes the challenge s^* as a covertext consisting of samples from the channel oracle:

1 $s^* \leftarrow \mathfrak{C}_h^{\ell(|m^*|)}$

For $0 \leq i \leq 2$, let $\mathsf{Adv}^i_{\mathcal{A},\Pi}$ denote $\mathcal{A}$'s advantage of winning Game_i. Since Π' is $(t_2, Q_{sk}, \epsilon_2)$-oABE\$-CPA-secure, it follows from a straightforward reduction argument that $\mathcal{A}$'s advantage in distinguishing Game_0 from Game_1 is at most ϵ_2 (*i.e.*, $\left|\mathsf{Adv}^0_{\mathcal{A},\Pi} - \mathsf{Adv}^1_{\mathcal{A},\Pi}\right| \leq \epsilon_2$). Once we bound the total message length by the polynomial μ, it follows from another simple reduction argument that $\mathcal{A}$'s advantage in distinguishing Game_1 from Game_2 is at most $\mu\epsilon_1$ (*i.e.*, $\left|\mathsf{Adv}^1_{\mathcal{A},\Pi} - \mathsf{Adv}^2_{\mathcal{A},\Pi}\right| \leq \mu\epsilon_1$). Therefore, we have

$$\left|\mathsf{Adv}^0_{\mathcal{A},\Pi} - \mathsf{Adv}^2_{\mathcal{A},\Pi}\right| \leq \mu\epsilon_1 + \epsilon_2.$$

The theorem then follows from the observation that Game_2 amounts to the actual BS-IND-CHA game when the challenge bit b^* is fixed to 1. ■

D Proof of Theorem 5.2

Proof. We organize this proof as a sequence of games (Game_0, Game_1, Game_2, Game_3) between a BS-IND-CCA adversary $\mathcal{A}$ and a challenger $\mathcal{C}$. During the *Challenge* phase of Game_0, $\mathcal{A}$ is given

a stegotext for m^* under S^*. The stegotext given to $\mathcal{A}$ during the *Challenge* phase of Game_3, on the other hand, consists just of documents sampled from the channel function under uniform randomness.

Game_0: is the actual BS-IND-CCA game when the challenge bit b^* is fixed to 0. The interaction between $\mathcal{A}$ and $\mathcal{C}$ during *Setup*, *Phase 1*, *Phase 2*, and *Guess* follows as specified in Definition 3.3. After $\mathcal{A}$ submitted a message $m^* \in \mathcal{MSP}$ and a set of user identities $S^* \subseteq U$ (with the restriction that $S^* \cap R = \emptyset$) during the *Challenge* phase, $\mathcal{C}$ generates the challenge stegotext s^*, which is later given to $\mathcal{A}$, as follows:

$\hat{r} \leftarrow\$ \{0,1\}^\lambda$
$c \leftarrow \mathsf{Encrypt}'(\mathsf{MPK}', S^*, \hat{r} \| m^*)$
$r := G(\hat{r}, |c| \cdot \lambda^2)$
$s^* := \mathsf{DSample}(\lambda, H, c, r)$

Game_1: is similar to Game_0, but $\mathcal{C}$ computes the challenge stegotext s^* as follows:

$\hat{r} \leftarrow\$ \{0,1\}^\lambda$
$c \leftarrow\$ \{0,1\}^{\ell(\lambda + |m^*|)}$
$r := G(\hat{r}, |c| \cdot \lambda^2)$
$s^* := \mathsf{DSample}(\lambda, H, c, r)$

Game_2: is similar to Game_1, but $\mathcal{C}$ now computes the challenge stegotext s^* as:

$c \leftarrow\$ \{0,1\}^{\ell(\lambda + |m^*|)}$
$r \leftarrow\$ \{0,1\}^{|c| \cdot \lambda^2}$
$s^* := \mathsf{DSample}(\lambda, H, c, r)$

Game_3: is similar to Game_2, but $\mathcal{C}$ generates the challenge stegotext s^* as follows:

$l := \ell(\lambda + |m^*|)$
for $j := 1$ **to** l **do**
    $r \leftarrow\$ \{0,1\}^\lambda$
    $s_j^* := \mathsf{Channel}(r)$
$s^* := s_1^* \| \ldots \| s_l^*$

For $0 \leq i \leq 3$, let $\mathsf{Adv}^i_{\mathcal{A},\Pi}$ denote $\mathcal{A}$'s advantage of winning Game_i. Because Π' is $(t_3, Q_{sk}, Q_d, \epsilon_3)$-oABE\$-CCA-secure, it follows from a simple reduction argument that $\mathcal{A}$'s advantage in distinguishing Game_0 from Game_1 is at most ϵ_3 (*i.e.*, $|\mathsf{Adv}^0_{\mathcal{A},\Pi} - \mathsf{Adv}^1_{\mathcal{A},\Pi}| \leq \epsilon_3$). Since G is (t_2, ϵ_2)-hard, it follows from another straightforward reduction argument that A's advantage in distinguishing Game_1 from Game_2 is at most ϵ_2 (*i.e.*, $|\mathsf{Adv}^1_{\mathcal{A},\Pi} - \mathsf{Adv}^2_{\mathcal{A},\Pi}| \leq \epsilon_2$). Once we bound the total message length by the polynomial μ, it follows from yet another simple reduction argument that $\mathcal{A}$'s advantage in distinguishing Game_2 from Game_3 is at most $\mu\epsilon_1$ (*i.e.*, $|\mathsf{Adv}^2_{\mathcal{A},\Pi} - \mathsf{Adv}^3_{\mathcal{A},\Pi}| \leq \mu\epsilon_1$). Thus, $\left|\mathsf{Adv}^0_{\mathcal{A},\Pi} - \mathsf{Adv}^3_{\mathcal{A},\Pi}\right| \leq \mu\epsilon_1 + \epsilon_2 + \epsilon_3$.

The theorem then follows from the observation that Game_3 amounts to the actual BS-IND-CCA game when the challenge bit b^* is fixed to 1. ■