How to Subvert Backdoored Encryption: Security Against Adversaries that Decrypt All Ciphertexts

Thibaut Horel[1]
Harvard University, Cambridge, MA, USA

Sunoo Park[2]
MIT, Cambridge, MA, USA

Silas Richelson
University of California, Riverside, CA, USA

Vinod Vaikuntanathan[3]
MIT, Cambridge, MA, USA

Abstract

In this work, we examine the feasibility of secure and undetectable point-to-point communication when an adversary (e.g., a government) can read all encrypted communications of surveillance targets. We consider a model where the only permitted method of communication is via a government-mandated encryption scheme, instantiated with government-mandated keys. Parties cannot simply encrypt ciphertexts of some other encryption scheme, because citizens caught trying to communicate outside the government's knowledge (e.g., by encrypting strings which do not appear to be natural language plaintexts) will be arrested. The one guarantee we suppose is that the government mandates an encryption scheme which *is* semantically secure against outsiders: a perhaps reasonable supposition when a government might consider it advantageous to secure its people's communication against foreign entities. But then, what good is semantic security against an adversary that holds all the keys and has the power to decrypt?

We show that even in the pessimistic scenario described, citizens *can* communicate securely and undetectably. In our terminology, this translates to a positive statement: all semantically secure encryption schemes support *subliminal communication*. Informally, this means that there is a two-party protocol between Alice and Bob where the parties exchange ciphertexts of what appears to be a normal conversation even to someone who knows the secret keys and thus can read the corresponding plaintexts. And yet, at the end of the protocol, Alice will have transmitted her secret message to Bob. Our security definition requires that the adversary not be able to tell whether Alice and Bob are just having a normal conversation using the mandated encryption scheme, or they are using the mandated encryption scheme for subliminal communication.

Our topics may be thought to fall broadly within the realm of *steganography*. However, we deal with the non-standard setting of an adversarially chosen distribution of cover objects (i.e., a stronger-than-usual adversary), and we take advantage of the fact that our cover objects are ciphertexts of a semantically secure encryption scheme to bypass impossibility results which we show for broader classes of steganographic schemes. We give several constructions of subliminal communication schemes under the assumption that key exchange protocols with pseudorandom messages exist (such as Diffie-Hellman, which in fact has truly random messages).

[1] Supported, in part, by the National Science Foundation under grants CAREER IIS-1149662, and CNS-1237235, by the Office of Naval Research under grants YIP N00014-14-1-0485 and N00014-17-1-2131, and by a Google Research Award.

[2] Supported by the Center for Science of Information STC (CSoI), an NSF Science and Technology Center (grant agreement CCF-0939370), MACS project NSF grant CNS-1413920, and a Simons Investigator Award Agreement dated 2012-06-05.

[3] Supported in part by NSF Grants CNS-1350619, CNS-1414119 and CNS-1718161, Alfred P. Sloan Research Fellowship, Microsoft Faculty Fellowship and a Steven and Renee Finn Career Development Chair from MIT.

10th Innovations in Theoretical Computer Science (ITCS 2019).
Editor: Avrim Blum; Article No. 42; pp. 42:1–42:20
LIPICS Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

2012 ACM Subject Classification Theory of computation → Cryptographic protocols

Keywords and phrases Backdoored Encryption, Steganography

Digital Object Identifier 10.4230/LIPIcs.ITCS.2019.42

Related Version A full version of the paper is available at [17], `https://eprint.iacr.org/2018/212`.

Acknowledgements We are grateful to Omer Paneth and Adam Sealfon for insightful remarks on an early draft.

1 Introduction

Suppose that we lived in a world where the government wished to read all the communications of its citizens, and thus decreed that citizens must not communicate in any way other than by using a specific, government-mandated encryption scheme with government-mandated keys. Even face-to-face communication is not allowed: in this Orwellian world, anyone who is caught speaking to another person will be arrested for treason. Similarly, anyone whose communications appear to be hiding information will be arrested: e.g., if the plaintexts encrypted using the government-mandated scheme are themselves ciphertexts of a different encryption scheme. However, the one assumption that we entertain in this paper, is that the government-mandated encryption scheme is, in fact, semantically secure: this is a tenable supposition with respect to a government that considers secure encryption to be in its interest, in order to prevent foreign powers from spying on its citizens' communications.

A natural question then arises: is there any way that the citizens would be able to communicate in a fashion undetectable to the government, based only on the semantic security of the government-mandated encryption scheme, and *despite the fact that the government knows the keys and has the ability to decrypt all ciphertexts*?[4] What can semantic security possibly guarantee in a setting where the adversary has the private keys?

This question may appear to fall broadly within the realm of *steganography*: the science of hiding secret communications within other innocent-looking communications (called "cover objects"), in an undetectable way. Indeed, it can be shown that if two parties have a shared secret, then based on slight variants of existing techniques for *secret-key steganography*, they can conduct communications hidden from the government.[5]

However, the question of whether two parties who have never met before can conduct hidden communications is more interesting. This is related to the questions of *public-key steganography* and *steganographic key exchange* which were both first formalized by von Ahn and Hopper [23]. Public-key steganography is inadequate in our setting since exchanging or publishing public keys is potentially conspicuous and thus is not an option in our setting. All prior constructions of steganographic key exchange require the initial sampling of a public random string that serves as a public parameter of the steganographic scheme. Intuitively, in these constructions, the public random string can be thought to serve the purpose of

[4] We note that one could, alternatively, consider an adversary with decryption capabilities arising from possession of some sort of "backdoor." For the purposes of this paper, we opted for the simpler and still sufficiently expressive model where the adversary's decryption power comes from knowledge of all the decryption keys.

[5] We refer the reader to Section 1.4 for more details.

selecting a specific steganographic scheme from a family of schemes *after* the adversary has chosen a strategy. That is, the schemes crucially assume that the adversary (the dystopian government, in our story above) cannot choose its covertext distribution as a function of the public parameter.

It is conservative and realistic to expect a malicious adversary to choose the covertext distribution *after* the honest parties have decided on their communication protocol (including the public parameters). After all, malice never sleeps [18]. Alas, we show that if the covertext distribution is allowed to depend on the communication protocol, steganographic communication is impossible. In other words, for every purported steganographic communication protocol, there is a covertext distribution (even one with high min-entropy) relative to which the communication protocol fails to embed subliminal messages. The relatively simple counterexample we construct is inspired by the impossibility of deterministic extraction.

Semantic Security to the Rescue? However, this impossibility result does not directly apply to our setting, as our covertext distribution is restricted to be a sequence of ciphertexts (that may encrypt arbitrary messages). Moreover, the ciphertexts are semantically secure against entities that are not privy to the private keys. We define the notion of a *subliminal communication scheme* (Definition 7) as a steganographic communication scheme where security holds relative to covertext distributions that are guaranteed to be ciphertexts of some semantically secure encryption scheme. Is there a way to use semantic security to enable subliminal communication?

Our first answer to this question is negative. In particular, consider the following natural construction: first, design an extractor function f; then, to subliminally transmit a message bit b, sample encryptions c of a (even adversarially prescribed) plaintext m using independent randomness every time, until $f(c) = b$. There are two reasons this idea does not work. First, if the plaintext bit b is not random, the adversary can detect this by applying the extractor function f to the transmitted covertext. Second, the government can pick an adversarial (semantically secure) encryption scheme where the extractor function f is constant on all ciphertexts; this is again similar to the impossibility of deterministic extraction.

Nevertheless, we show how to circumvent these difficulties and use the semantic security of the underlying (adversarial) encryption scheme to construct subliminal communication.

▶ **Theorem 1** (Informal version of Theorem 11). *Under the decisional Diffie-Hellman (DDH) assumption – or any other assumption that gives rise to a key exchange protocol with messages indistinguishable from random – there is a subliminal communication scheme which allows the transmission of $O(\log \kappa)$ many bits per ciphertext after a setup phase of $\widetilde{O}(\log \kappa)$ ciphertexts (κ is the security parameter).*

We then show how to improve our first construction to reduce the length of the setup phase under additional assumptions.

1.1 Overview of Our Construction

The first idea in our construction is implicit in essentially all the works in steganography starting from [20]: namely, to achieve subliminal communication of arbitrary messages, it is sufficient to be able to undetectably communicate *uniformly randomly distributed strings of one's choice.* In other words, Alice samples a string r which is randomly distributed, produces some ciphertext(s) to be sent to Bob, such that Bob is able to learn r from them, and yet a PPT eavesdropper Eve who sees the entire communication transcript cannot distinguish between the following two cases:

1. Alice is indeed sending (hereafter, "embedding") random strings to Bob, or
2. Alice is producing ciphertexts using the unmodified government-mandated encryption algorithm, without embedding such random strings.

To be more precise, the indistinguishability requirement holds for any given (adversarially specified) distribution $\mathcal{M}$ of message sequences that Alice may choose to encrypt using the government-mandated encryption scheme. Notice that this does not preclude that Eve may be able to learn r and indeed, our constructions do allow an eavesdropper to learn the embedded strings. Given the ability to undetectably communicated randomly distributed strings, Alice and Bob can then embed to each other the messages of a key-exchange protocol with randomly distributed messages (such as Diffie-Hellman) to establish a shared secret, and then embed to each other ciphertexts of a secret-key encryption scheme with pseudorandom ciphertexts, using the established secret as the key.

All known constructions of such *undetectable random string embedding* rely on the sampling of a public random seed after the adversarial strategy is fixed. In this paper, however, we are interested in bootstrapping hidden communications from the very ground up: we are not willing to assume that the parties start from a state where a seed is already present.

We observe that the ability to embed randomly distributed strings *of one's choice* – rather than, e.g., to apply a deterministic function to ciphertexts of the government-mandated encryption scheme, and thereby obtain randomly distributed strings which the creator of the ciphertexts did not choose – is crucial to the above-outlined scheme. The notion of undetectably embedding *exogenous* random strings – i.e., strings that are randomly distributed outside of Alice's control, but both Alice and Bob can read them – is seemingly much weaker, and certainly cannot be used to embed key exchange messages or secret-key ciphertexts. However, we observe that this weaker primitive turns out to be achievable, for our specific setting, without the troublesome starting assumption of a public random seed. We identify a method for embedding *exogenous* random strings into ciphertexts of an adversarially chosen encryption scheme (interestingly, our method does not generalize to embedding into arbitrary min-entropy distributions). We then exploit this method to allow the communicating parties to establish a random seed – from which point they can proceed to embed random strings *of their choice*, as described above.

In building this weaker primitive, in order to bypass our earlier-described impossibility result, we extract from two ciphertexts at a time, instead of one. We begin with the following simple idea: for each consecutive pair of ciphertexts c and c', a single hidden (random) bit b is defined by $b = f(c, c')$ where f is some two-source extractor. It is initially unclear why this should work because (1) c and c' are encryptions of messages m and m' which are potentially dependent, and two-source extractors are not guaranteed to work without independence; and (2) even if this difficulty could be overcome, ciphertexts of semantically secure encryption scheme can have min-entropy as small as $\omega(\log \kappa)$ (where κ is the security parameter) and no known two-source extractor known can extract from such a small min-entropy.

We overcome difficulty (1) by relying on the semantic security of the ciphertexts of the adversarially chosen encryption scheme. Paradoxically, even though the adversary knows the decryption key, we exploit the fact that semantic security still holds against the *extractor*, which does not have the decryption key. The inputs in our case are ciphertexts which are not necessarily independent, but semantic security implies that they are computationally indistinguishable from being independent. Thus, the output of $f(c, c')$ is pseudorandom. Indeed, when f outputs a single bit (as in our construction), the output is also statistically close to random. The crucial point here is that the semantic security of the encryption scheme is used not against the government, but rather against the extraction function f.

Our next observation, to address difficulty (2), is that the ciphertexts are not only computationally independent, but they are also computationally indistinguishable from i.i.d. In particular, each pair of ciphertexts is indistinguishable from a pair of encryptions of 0, by semantic security. Based on this observation, we can use a very simple "extractor", namely, the greater-than function GT. In fact, GT is an extractor with two input sources, whose output bit has negligible bias when the sources have $\omega(\log \kappa)$ min-entropy and are *independently and identically distributed* (this appears to be a folklore observation; see, e.g., [3]). Because of the last condition, GT is not a true two-source extractor according to standard definitions, but is still suitable for our setting.

By repeatedly extracting random bits from pairs of consecutive ciphertexts using GT, Alice and Bob can construct a shared random string s. Note that in this process, Alice and Bob generate ciphertexts using the unmodified government-mandated encryption scheme, so the indistinguishability requirement clearly holds. We stress again that s is also known to a passive eavesdropper of the communication. This part of our construction, up to the construction of the string s, is presented in details in Section 5.1. From there, constructing a subliminal communication scheme is not hard: Alice and Bob use s as the seed of a strong seeded extractor to subliminally communicate random strings *of their choice* as explained in Section 5.2. The complete description of our protocol is given in Section 5.3.

1.2 Improved Constructions for Specific Cases

While our first construction has the advantage of simplicity, the initial phase to agree on shared random string (using the GT function) transmits only one hidden bit per ciphertext of the government-mandated encryption scheme. A natural question is whether this rate of transmission can be improved. We show that if the government-mandated encryption scheme is *succinct* in the sense that the ciphertext expansion factor is at most 2, then it is possible to improve the rate of transmission in this phase to $O(\log \kappa)$ hidden bits per ciphertext using an alternative construction based on the extractor from [12]. In other words, our first result showed that if the government-mandated encryption scheme is semantically secure, we can use it to communicate subliminally; the second result shows that if the government-mandated encryption scheme is efficient, that is even better for us, in the sense that it can be used for more efficient subliminal communication.

▶ **Theorem 2** (Informal). *If there is a secure key exchange protocol whose message distribution is pseudorandom, then there is a subliminal communication scheme in which a shared seed is established in two exchanges of ciphertexts of a succinct encryption scheme.*

Theorem 1 exploited the specific nature of the cover object distribution in our setting (specifically, that a sequence of encryptions of arbitrary messages is indistinguishable from an i.i.d. sequence of encryptions of zero). Theorem 2 exploits an additional consequence of the semantic security of the government-mandated encryption scheme: if it is succinct, then ciphertexts are computationally indistinguishable from sources of high min-entropy (i.e., they have large HILL-entropy).

It may be possible to use more advanced two-source extractors to work with a larger class of encryption schemes (with larger expansion factors); however, the best known such extractors have an inverse polynomial error rate [8] (whereas our construction's extractor has negligible error). Consequently, designing a subliminal communication protocol using these extractors seems to require additional ideas, and we leave this as an open problem.

Finally, we show yet another approach in cases where the distribution of "innocent" messages to be encrypted under the government-mandated encryption scheme has a certain amount of conditional min-entropy. For such cases, we construct an alternative scheme that leverages the semantic security of the encryption scheme in a rather different way: namely, the key fact for this alternative construction is that (in the absence of a decryption key) a ciphertext appears independent of the message it encrypts. In this case, running a two-source extractor on the message and the ciphertext works. The resulting improvement in the efficiency of the scheme is comparable to that of Theorem 2.

▶ **Theorem 3** (Informal). *If there is a secure key-exchange protocol whose message distribution is pseudorandom, then there is a subliminal communication scheme for any cover distribution that* either

- *consists of ciphertexts of a semantically secure encryption scheme, if the* innocent message distribution $\mathcal{M}$ *has conditional min-entropy rate* $1/2$*,* or
- *consists of ciphertexts of a semantically secure* and succinct *encryption scheme, if the* innocent message distribution $\mathcal{M}$ *has conditional min-entropy* $\omega(\log \kappa)$*.*

In both cases, the shared seed is established during the setup phase in only two exchanges of ciphertexts.

Due to space constraints, the results described in this subsection (1.2) are not discussed further herein. They are presented in detail in the full version of this paper [17].

1.3 Final Introductory Remarks

On Our Modeling Assumptions. Our model considers a relatively powerful adversary that, for example, has the ability to choose the encryption scheme using which all parties must communicate, and to decrypt all such communications. We believe that this can be very realistic in certain scenarios, but it is also important to note the limitations that our model places on the adversary.

The most obvious limitation is that the encryption scheme chosen by the adversary must be semantically secure (against third parties that do not have the ability to decrypt). Another assumption is that citizens are able to run algorithms of their choice on their own computers without, for instance, having every computational step monitored by the government. Moreover, citizens may use encryption randomness of their choice when producing ciphertexts of the government-mandated encryption scheme: in fact, this is a key fact that our construction exploits. Interestingly, secrecy of the encryption randomness from the adversary is irrelevant: after all, the adversary can always choose an encryption scheme where the encryption randomness is recoverable given the decryption key. Despite this, the ability of the encryptor to choose the randomness to input to the encryption algorithm can be exploited – as by our construction – to allow for subliminal communication.

The Meaning of Semantic Security when the Adversary Can Decrypt. In an alternate light, our work may be viewed as asking the question: *what guarantee, if any, does semantic security provide against adversary in possession of the decryption key?* Our results find, perhaps surprisingly, that some meaningful guarantee is still provided by semantic security even against an adversary is able to decrypt: more specifically, that *any* communication channel allowing transmission of ciphertexts can be leveraged to allow for undetectable communications between two parties that have never met. From this perspective, our work may be viewed as the latest in a scattered series of recent works that consider what guarantees can be provided by cryptographic primitives that are somehow "compromised" – examples of recent works in this general flavor are cited in Section 1.4 below.

Concrete Security Parameters. From a more practical perspective, it may be relevant to consider that the government in our hypothetical Orwellian scenario would be incentivized to opt for an encryption scheme with the least possible security level so as to ensure security against foreign powers. In cases where the government considers itself to have more computational power than foreign adversaries (perhaps by a constant factor), this could create an interesting situation where the security parameter with which the government-mandated scheme must be instantiated is *below* what is necessary to ensure security against the government's own computational power.

Such a situation could be risky for citizens' hidden communications: intuitively, our constructions guarantee indistinguishability *against the citizens' own government* between an "innocent" encrypted conversation and one which is carrying hidden subliminal messages. However, the distinguishing advantage in this indistinguishability game *depends on the security parameter* of the government-mandated encryption scheme. Thus, it could be that the two distributions are far enough apart for the citizens' own government to distinguish (though not for foreign governments to distinguish). We observe that citizens cognizant of this situation can further reduce the distinguishing advantage beyond that provided by our basic construction, using the standard technique of amplifying the proximity of a distribution (which is far from random) to uniformly random, by taking the XOR of several samples from the far-from-random distribution.

Having outlined this potential concern and solution, the rest of the paper will disregard these issues in the interest of clarity of exposition, and present a purely asymptotic analysis.

Open Problems. Our work suggests a number of open problems. A natural one is the extent to which the modeling assumptions that this work makes – such as the ability of honest encryptors to use true randomness for encryption – can be relaxed or removed, while preserving the ability to communicate subliminally. For example, one could imagine yet another alternate universe, in which the hypothetical Orwellian government not only mandates that citizens use the prescribed encryption scheme, but also that their encryption randomness must be derived from a specific government-mandated pseudorandom generator.

The other open problems raised by our work are of a more technical nature and better understood in the context of the specific details of our constructions; for this reason we defer their discussion to Section 6.

1.4 Other Related Work

The scientific study of steganography was initiated by Simmons more than thirty years ago [20], and is the earliest mention of the term "subliminal channel" referring to the conveyance of information in a cryptosystem's output in a way that is different from the intended output,[6] of which we are aware. Subsequent works such as [7, 19, 27] initially explored information-theoretic treatments of steganography, and then Hopper, Langford, and von Ahn [16] gave the first complexity-theoretic (secret-key) treatment almost two decades later. Public-key variants of steganographic notions – namely, public-key steganography and steganographic key exchange – were first defined by [23]. There is very little subsequent literature on public-key steganographic primitives; one notable example is by Backes and Cachin [2], which considers public-key steganography against active attacks (their attack model, which is stronger than that of [23], was also considered in [16] but had never been applied to the public-key setting).

[6] This phrasing is loosely borrowed from [26].

The alternative perspective of our work as addressing the question of whether any sort of secret communication can be achieved via transmission of ciphertexts of an adversarially designed cryptosystem alone fits into a scattered series of recent works that consider what guarantees can or cannot be provided by compromised cryptographic primitives. For example, Goldreich [14], and later, Cohen and Klein [10], consider what unpredictability guarantee is achieved by the classic GGM construction [15] when the traditionally secret seed is known; Austrin et al. [1] study whether certain cryptographic primitives can be secure even in the presence of an adversary that has limited ability to tamper with honest parties' randomness; Dodis et al. [13] consider what cryptographic primitives can be built based on backdoored pseudorandom generators; and Bellare, Jaeger, and Kane [4] present attacks that work against any symmetric-key encryption scheme, that completely compromise security by undetectably corrupting the algorithms of the encryption scheme (such attacks might, for example, be feasible if an adversary could generate a bad version of a widely used cryptographic library and install it on his target's computer).

The last work mentioned above, [4], is actually part of the broader field of kleptography, originally introduced by Young and Yung [26, 25, 24]. Broadly speaking, a *kleptographic attack* "uses cryptography against cryptography" [26] – i.e., changes the behavior of a cryptographic system in a fashion undetectable to an honest user with black-box access to the cryptosystem, such that the use of the modified system leaks some secret information (e.g., plaintexts or key material) to the attacker who performed the modification. An example of such an attack might be to modify the key generation algorithm of an encryption scheme such that an adversary in possession of a "back door" can derive the private key from the public key, yet an honest user finds the generated key pairs to be indistinguishable from correctly produced ones. Kleptography has enjoyed renewed research activity since [5] introduced a formal model of a specific type of kleptographic attack called *algorithm substitution attacks* (ASAs), motivated by recent revelations suggesting that intelligence agencies have successfully implemented attacks of this nature at scale. Recently, [6] formalized an equivalence between certain variants of ASA and steganography.

Our setting differs significantly from kleptography in that the encryption algorithms are public and not tampered with (i.e., adhere to a purported specification), and in fact may be *known* to be designed by an adversarial party.

2 Preliminaries

Proofs of all propositions, lemmata, and theorems are in the full version of this paper [17] due to space constraints.

Notation. κ is the security parameter throughout. PPT means "probabilistic polynomial time." $[n]$ denotes the set $\{1, \ldots, n\}$. U_n is a uniform variable over $\{0,1\}^n$, independent of every other variable in this paper. We write $X \sim Y$ to express that X and Y are identically distributed. Given two variables X and Y over $\{0,1\}^k$, we denote by $\|X - Y\|_{\mathsf{s}}$ the statistical distance defined by:

$$\|X - Y\|_{\mathsf{s}} = \frac{1}{2} \sum_{x \in \{0,1\}^k} \big| \Pr[X = x] - \Pr[Y = x] \big| \,.$$

For a random variable X, we define the min-entropy of X by $H_\infty(X) = -\log \max_x \Pr[X = x]$. The collision probability is $\mathsf{CP}(X) = \sum_x \Pr[X = x]^2$.

2.1 Encryption and Key Exchange

We assume familiarity with the standard notions of semantically secure public-key and private-key encryption, and key exchange. This subsection defines notation and terminology.

Public-Key Encryption. We use the notation $\mathsf{E} = (\mathsf{E.Gen}, \mathsf{E.Enc}, \mathsf{E.Dec})$ for the public-key encryption scheme mandated by the adversary.

Secret-key Encryption. We write $\mathsf{SKE} = (\mathsf{SKE.Gen}, \mathsf{SKE.Enc}, \mathsf{SKE.Dec})$ to denote a secret-key encryption scheme. We define a *pseudorandom secret-key encryption scheme* to be a secret-key encryption scheme whose ciphertexts are indistinguishable from random. It is a standard result that pseudorandom secret-key encryption schemes can be built from one-way functions.

Key Exchange. A *key-exchange protocol* Λ is a two-party protocol executed between two parties P_0 and P_1, where each party outputs a key at the end of the protocol. The *correctness* guarantee for key-exchange Protocols requires that the two outputted keys be equal with overwhelming probability. The security guarantee for key-exchange protocols requires that $(T, K) \overset{c}{\approx} (T, K_\$)$, where T is a key-exchange protocol transcript, K is the shared key established in T, and $K_\$$ is a random unrelated key.

We define a *pseudorandom key-exchange protocol* to be a key-exchange protocol whose transcripts are distributed indistinguishably from random. That is, a pseudorandom key-exchange protocol has the stronger guarantee that $(T, K) \overset{c}{\approx} (U, K_\$)$ where U is the uniform distribution over message sequences of the appropriate length, where messages are drawn randomly from the message space of Λ.

The classical protocol of Diffie and Hellman [11] is pseudorandom; in fact, its messages are uniformly random over a cyclic group G. However, the constructions in this paper assume a key-exchange protocol whose messages are pseudorandom *over bit strings*. In fact, it is possible to transform a key-exchange protocol whose messages are pseudorandom over an arbitrary domain $G \subseteq \{0,1\}^\ell$ into a key-exchange protocol whose messages are pseudorandom over bit strings. Proposition 4, below, gives an encoding and decoding algorithm to transform uniformly random messages in G into a sequence of uniformly random messages in $\{0,1\}^\ell$. The encoding and decoding algorithms run in polynomial time as long as the density of messages $\frac{|G|}{2^\ell}$ is noticeable (i.e., at least $\frac{1}{\kappa^c}$ for some $c \geq 1$). This is the case, for example, when the Diffie-Hellman protocol is instantiated with the group of quadratic residues modulo a safe prime (in which case the density of message is constant close to $\frac{1}{2}$).

▶ **Proposition 4.** *Let G be a subset of $\{0,1\}^\ell$ and define $p = \frac{\kappa 2^\ell}{|G|}$. There is an encoding algorithm $E : G \to (\{0,1\}^\ell)^p$ and a decoding algorithm $D : (\{0,1\}^\ell)^p \to G \cup \{\perp\}$ that satisfy the following properties:*
1. Correctness: *for all $g \in G$,* $\Pr\left[D(E(g)) \neq g\right]$ *is negligible in κ.*
2. Randomness: *for uniformly random $g \leftarrow G$, $E(g)$ is uniformly random over $(\{0,1\}^\ell)^p$.*

Explicit descriptions of algorithms E, D are given in the full version of this paper.

2.2 Extractors

Next, we give standard definitions of two-source and seeded extractors.

▶ **Definition 5.** The family $\mathsf{2Ext} : \{0,1\}^n \times \{0,1\}^{n'} \to \{0,1\}^\ell$ is a (k_1, k_2, ε) *two-source extractor* if for all $\kappa \in \mathbb{N}$ and for all pairs (X, Y) of independent random variables over $\{0,1\}^{n(\kappa)} \times \{0,1\}^{n'(\kappa)}$ such that $H_\infty(X) \geq k_1(\kappa)$ and $H_\infty(Y) \geq k_1(\kappa)$, it holds that

$\left\|\mathsf{2Ext}_\kappa(X,Y) - U_{\ell(\kappa)}\right\|_{\mathsf{s}} \leq \varepsilon(\kappa)$. We say that 2Ext is *strong w.r.t. the first input* if it satisfies the stronger property that $\left\|(X, \mathsf{2Ext}_\kappa(X,Y)) - (X, U_{\ell(\kappa)})\right\|_{\mathsf{s}} \leq \varepsilon(\kappa)$. A strong two-source extractor w.r.t. the second input is defined analogously. Finally, we say that 2Ext is a (k,ε) *same-source* extractor if $n = n'$ and the extractor output is only required to be statistically close to uniform when (X,Y) is a pair of i.i.d. random variables with $H_\infty(X) = H_\infty(Y) \geq k(\kappa)$.

▶ **Definition 6.** The family $\mathsf{Ext} : \{0,1\}^n \times \{0,1\}^{n'} \to \{0,1\}^\ell$ is a (k,ε) *seeded extractor* if for all $\kappa \in \mathbb{N}$ and any random variable X over $\{0,1\}^{n'(\kappa)}$ such that $H_\infty(X) \geq k(\kappa)$, it holds that $\left\|\mathsf{Ext}_\kappa(U_{n(\kappa)}, X) - U_{\ell(\kappa)}\right\|_{\mathsf{s}} \leq \varepsilon(\kappa)$. We say moreover that Ext is *strong* if it satisfies the stronger property that $\left\|(U_{n(\kappa)}, \mathsf{Ext}_\kappa(U_{n(\kappa)}, X)) - (U_{n(\kappa)}, U_{\ell(\kappa)})\right\|_{\mathsf{s}} \leq \varepsilon(\kappa)$.

3 Subliminal Communication

Conversation Model. The protocols we will construct take place over a communication between two parties P_0 and P_1 alternatingly sending each other ciphertexts of a public-key encryption scheme. *W.l.o.g.*, we assume that P_0 initiates the communication, and that communication occurs over a sequence of *exchange-rounds* each of which comprises two sequential messages: in each exchange-round, one party P_b sends a message to P_{1-b} and then P_{1-b} sends a message to P_b. Let $m_{b,i}$ denote the plaintext message sent by P_b to P_{1-b} in exchange-round i, and let $\boldsymbol{m}_i = (m_{0,i}, m_{1,i})$ denote the pair of messages exchanged. For $i \geq 1$, let us denote by $\boldsymbol{\tau}_{0,i} = (\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{i-1})$ and $\boldsymbol{\tau}_{1,i} = (\boldsymbol{m}_1, \ldots, \boldsymbol{m}_{i-1}, m_{0,i})$ the plaintext transcripts available to P_0 and P_1 respectively during exchange-round i, in the case when P_0 sends the first message in exchange-round i.[7] We define $\boldsymbol{\tau}_{0,0}$ and $\boldsymbol{\tau}_{1,0}$ to be empty lists (i.e., empty starting transcripts). (Note that when a notation contains both types of subscripts, we write the subscripts denoting the party and round in blue and red respectively, to improve readability.)

Recall that our adversary has the power to decrypt all ciphertexts under its chosen public-key encryption scheme E. Intuitively, it is therefore important that the plaintext conversation between P_0 and P_1 appears innocuous (and does not, for example, consist of ciphertexts of another encryption scheme). To model this, we assume the existence of a next-message distribution $\mathcal{M}$, which outputs a next innocuous message given the transcript of the plaintext conversation so far. This is denoted by $m_{b,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{b,i})$.

In all the protocols we consider, the symbol $\mathfrak{s}$ is used to denote internal state kept locally by P_0 and P_1. It is implicitly assumed that each party's state contains an up-to-date transcript of all messages received during the protocol. Parties may additionally keep other information in their internal state, as a function of the local computations they perform. For $i \geq 1$, $\mathfrak{s}_{b,i}$ denotes the state of P_b at the conclusion of exchange-round i. Initial states $\mathfrak{s}_{b,0} = \varnothing$ are empty.

We begin with a simpler definition that only syntactically allows for the transmission of a single message (Definition 7). This both serves as a warm-up to the multi-message definition presented next (Definition 8), and will be used in its own right to prove impossibility results.

▶ **Definition 7.** A *subliminal communication scheme* is a two-party protocol:

$$\Pi^{\mathsf{E}} = \left(\Pi^{\mathsf{E}}_{0,1}, \Pi^{\mathsf{E}}_{1,1}, \Pi^{\mathsf{E}}_{0,2}, \Pi^{\mathsf{E}}_{1,2}, \ldots, \Pi^{\mathsf{E}}_{0,r}, \Pi^{\mathsf{E}}_{1,r}; \Pi^{\mathsf{E}}_{1,\mathsf{out}}\right)$$

[7] If instead P_1 spoke first in round i, then $\boldsymbol{\tau}_{0,i}$ would contain $m_{1,i}$, and $\boldsymbol{\tau}_{1,i}$ would not contain $m_{0,i}$.

where $r \in \mathsf{poly}$ is the number of exchange-rounds and each $\Pi^{\mathsf{E}}_{b,i}$ is a PPT algorithm with oracle access to the algorithms of a public-key encryption scheme E. Party P_0 is assumed to receive as input a message msg (of at least one bit) that is to be conveyed to P_1 in an undetectable fashion. The algorithms $\Pi^{\mathsf{E}}_{b,i}$ are used by P_b in round i, respectively, and $\Pi^{\mathsf{E}}_{1,\mathsf{out}}$ denotes the algorithm run by P_1 to produce an output msg' at the end of the protocol.

A subliminal communication scheme must satisfy the following syntax, correctness and security guarantees.

- **Syntax.** In each exchange-round $i = 1, \ldots, r$:
  1. P_0 performs the following steps:
     a. Sample "innocuous message" $m_{0,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{0,i-1})$.
     b. Generate ciphertext and state $(c_{0,i}, \mathfrak{s}_{0,i}) \leftarrow \Pi^{\mathsf{E}}_{0,i}(\mathsf{msg}, m_{0,i}, \mathsf{pk}_1, \mathfrak{s}_{0,i-1})$.
     c. Locally store $\mathfrak{s}_{0,i}$ and send $c_{0,i}$ to P_1.
  2. Then, P_1 performs the following steps:[8]
     a. Sample "innocuous message" $m_{1,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{1,i-1})$.
     b. Generate ciphertext and state $(c_{1,i}, \mathfrak{s}_{1,i}) \leftarrow \Pi^{\mathsf{E}}_{1,i}(m_{1,i}, \mathsf{pk}_0, \mathfrak{s}_{1,i-1})$.
     c. Locally store $\mathfrak{s}_{1,i}$ and send $c_{1,i}$ to P_0.

  After r rounds, P_1 computes $\mathsf{msg}' = \Pi^{\mathsf{E}}_{1,\mathsf{out}}(\mathsf{sk}_1, \mathfrak{s}_{1,r})$ and halts.
- **Correctness.** For any $\mathsf{msg} \in \{0,1\}^\kappa$, if P_0 and P_1 play Π^{E} honestly, then $\mathsf{msg}' = \mathsf{msg}$ with probability $1 - \mathsf{negl}(\kappa)$. The probability is taken over the key generation $(\mathsf{pk}_0, \mathsf{sk}_0), (\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow \mathsf{E.Gen}$ and the randomness of the protocol algorithms, as well as the message distribution $\mathcal{M}$.
- **Subliminal Indistinguishability.** For any semantically secure public-key encryption scheme E, any $\mathsf{msg} \in \{0,1\}^\kappa$ and any next-message distribution $\mathcal{M}$, for $(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow \mathsf{E.Gen}$, $i \in \{0,1\}$, the following distributions are computationally indistinguishable:

$\mathsf{Ideal}(\mathsf{pk}_0, \mathsf{sk}_0, \mathsf{pk}_1, \mathsf{sk}_1, \mathcal{M})$:	$\mathsf{Subliminal}_\Pi(\mathsf{msg}, \mathsf{pk}_0, \mathsf{sk}_0, \mathsf{pk}_1, \mathsf{sk}_1, \mathcal{M})$:
for $i = 1, \ldots, r$:	for $i = 1, \ldots, r$:
$m_{0,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{0,i})$	$m_{0,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{0,i})$
$m_{1,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{1,i})$	$m_{1,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{1,i})$
$c_{0,i} \leftarrow \mathsf{E.Enc}(\mathsf{pk}_1, m_{0,i})$	$(c_{0,i}, \mathfrak{s}_{0,i}) \leftarrow \Pi^{\mathsf{E}}_{0,i}(\mathsf{msg}, m_{0,i}, \mathsf{pk}_1, \mathfrak{s}_{0,i-1})$
$c_{1,i} \leftarrow \mathsf{E.Enc}(\mathsf{pk}_0, m_{1,i})$	$(c_{1,i}, \mathfrak{s}_{1,i}) \leftarrow \Pi^{\mathsf{E}}_{1,i}(m_{1,i}, \mathsf{pk}_0, \mathfrak{s}_{1,i-1})$
output $\left(\mathsf{pk}_0, \mathsf{sk}_0, \mathsf{pk}_1, \mathsf{sk}_1; (c_{b,i})_{b\in\{0,1\},i\in[r]}\right)$	output $\left(\mathsf{pk}_0, \mathsf{sk}_0, \mathsf{pk}_1, \mathsf{sk}_1; (c_{b,i})_{b\in\{0,1\},i\in[r]}\right)$

  If the *subliminal indistinguishability* requirement is satisfied only for next-message distributions $\mathcal{M}$ in a restricted set $\mathbb{M}$, rather than for any $\mathcal{M}$, then Π is said to be a *subliminal communication scheme for* $\mathbb{M}$.

For simplicity, Definition 7 presents a communication scheme in which only a single hidden message msg is transmitted. More generally, it is desirable to transmit multiple messages, and bidirectionally, and perhaps in an adaptive manner.[9] In multi-message schemes, it may be beneficial for efficiency that the protocol have a two-phase structure where some initial

[8] Note that the steps executed by P_0 and P_1 are entirely symmetric except in the following two aspects: first, P_0's input msg is present in step 1b but not in step 2b; and secondly, the state $\mathfrak{s}_{1,i-1}$ used in step 2b contains the round-i message $c_{0,i}$, whereas the state $\mathfrak{s}_{0,i-1}$ used in step 1b depends only on the transcript until round $i-1$.

[9] That is, the messages to be transmitted may become known as the protocol progresses, rather than all being known at the outset. This is the case, for example, if future messages depend on responses to previous ones.

preprocessing is done in the first phase, and then the second phase can thereafter be invoked many times to transmit different hidden messages.[10] This is a useful notion later in the paper, for our constructions, so we give the definition of a multi-message scheme here.

▶ **Definition 8.** A *multi-message subliminal communication scheme* is a two-party protocol defined by a pair (Φ, Ξ) where Φ ("Setup Phase") and Ξ ("Communication Phase") each define a two-party protocol. Each party outputs a state at the end of Φ, which it uses as an input in each subsequent invocation of Ξ. An execution of a multi-message subliminal communication scheme consists of an execution of Φ followed by one or more executions of Ξ. More formally:

$$\Phi^{\mathsf{E}} = \left(\Phi^{\mathsf{E}}_{0,1}, \Phi^{\mathsf{E}}_{1,1}, \Phi^{\mathsf{E}}_{0,2}, \Phi^{\mathsf{E}}_{1,2}, \dots, \Phi^{\mathsf{E}}_{0,r}, \Phi^{\mathsf{E}}_{1,r}\right)$$
$$\Xi^{\mathsf{E}} = \left(\Xi^{\mathsf{E}}_{0,1}, \Xi^{\mathsf{E}}_{1,1}, \Xi^{\mathsf{E}}_{0,2}, \Xi^{\mathsf{E}}_{1,2}, \dots, \Xi^{\mathsf{E}}_{0,r'}, \Xi^{\mathsf{E}}_{1,r'}; \Xi^{\mathsf{E}}_{1,\mathsf{out}}\right)$$

where $r, r' \in \mathsf{poly}$ are the number of exchange-rounds in Φ and Ξ respectively. and where each $\Phi^{\mathsf{E}}_{b,i}, \Xi^{\mathsf{E}}_{b,i}$ is a PPT algorithm with oracle access to the algorithms of a public-key encryption scheme E. The protocol must satisfy the following syntax, correctness and security guarantees.

- **Syntax.** In each exchange-round $i = 1, \dots, r$ of Φ: P_0 executes the following steps for $b = 0$, and then P_1 executes the same steps for $b = 1$.
  1. Sample "innocuous message" $m_{b,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{b,i-1})$.
  2. Generate ciphertext and state $(c_{b,i}, \mathfrak{s}_{b,i}) \leftarrow \Phi^{\mathsf{E}}_{b,i}(m_{b,i}, \mathsf{pk}_{1-b}, \mathfrak{s}_{b,i-1})$.
  3. Locally store $\mathfrak{s}_{b,i}$ and send $c_{b,i}$ to P_{1-b}.

  After the completion of Φ, either party may initiate Ξ by sending a first message of the Ξ protocol (with respect to a message msg to be steganographically hidden, known to the initiating party). Let P_S and P_R denote the initiating and non-initiating parties in an execution of Ξ, respectively.[11] Let $\mathsf{msg} \in \{0,1\}^\kappa$ be the hidden message that P_S is to transmit to P_R in an undetectable fashion during an execution of Ξ.
  The execution of Ξ proceeds as follows over exchange-rounds $i' = 1, \dots, r'$:
  - P_S acts as follows:
    1. Sample $m_{S,r+i'} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{S,r+i'-1})$.
    2. Generate $(c_{S,r+i'}, \mathfrak{s}_{S,r+i'}) \leftarrow \Xi^{\mathsf{E}}_{0,i'}(\mathsf{msg}, m_{S,r+i'}, \mathsf{pk}_R, \mathfrak{s}_{S,r+i'-1})$.
    3. Locally store $\mathfrak{s}_{S,r+i'}$ and send $c_{S,r+i'}$ to P_R.
  - P_R acts as follows:
    1. Sample $m_{R,r+i'} \leftarrow \mathcal{M}(\boldsymbol{\tau}'_{R,r+i'-1})$.
    2. Generate $(c_{R,r+i'}, \mathfrak{s}_{R,r+i'}) \leftarrow \Xi^{\mathsf{E}}_{1,i'}(m_{R,r+i'}, \mathsf{pk}_S, \mathfrak{s}_{R,r+i'-1})$.
    3. Locally store $\mathfrak{s}_{R,r+i'}$ and send $c_{R,r+i'}$ to P_S.

  At the end of an execution of Ξ, P_R computes $\mathsf{msg}' = \Xi^{\mathsf{E}}_{1,\mathsf{out}}(\mathsf{sk}_1, \mathfrak{s}_{1,r+r'})$.
- **Correctness.** For any $\mathsf{msg} \in \{0,1\}^\kappa$, if P_0 and P_1 execute (Φ, Ξ) honestly, then for every execution of Ξ, the transmitted and received messages msg and msg' are equal with overwhelming probability. The probability is taken over the key generation $(\mathsf{pk}_0, \mathsf{sk}_0), (\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow \mathsf{E.Gen}$ and the randomness of the protocol algorithms, as well as the message distribution $\mathcal{M}$.

---

[10] As a concrete example: consider a protocol for transmitting multiple encrypted messages with a one-time "phase 1" consisting of key exchange, and a "phase 2" encompassing the ciphertext transmission which can be invoked many times.

[11] Subscripts $S, R \in \{0,1\}$ stand for "sender" and "receiver," respectively.

- **Subliminal Indistinguishability.** For any semantically secure public-key encryption scheme E, any polynomial $p = p(\kappa)$, any sequence of hidden messages $\vec{\mathsf{msg}} = (\mathsf{msg}_i)_{i\in[p]} \in (\{0,1\}^\kappa)^p$, any sequence of bits $\vec{b} = (b_1,\dots,b_p) \in \{0,1\}^p$ and any next-message distribution $\mathcal{M}$, for $(\mathsf{pk}_b, \mathsf{sk}_b) \leftarrow \mathsf{E.Gen}$, $b \in \{0,1\}$ the following distributions are computationally indistinguishable:

$\underline{\mathsf{Ideal}(\mathsf{pk}_0, \mathsf{sk}_0, \mathsf{pk}_1, \mathsf{sk}_1, \mathcal{M})\text{:}}$

for $i = 1,\dots,r+pr'$:
  $m_{0,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{0,i})$
  $m_{1,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{1,i})$
  $c_{0,i} \leftarrow \mathsf{E.Enc}(\mathsf{pk}_1, m_{0,i})$
  $c_{1,i} \leftarrow \mathsf{E.Enc}(\mathsf{pk}_0, m_{1,i})$
output:
  $\big(\mathsf{pk}_0, \mathsf{sk}_0, \mathsf{pk}_1, \mathsf{sk}_1; (c_{b,i})_{b\in\{0,1\}, i\in[r+pr']}\big)$

$\underline{\mathsf{Subliminal}_{\Phi,\Xi}(\vec{\mathsf{msg}}, \vec{b}, \mathsf{pk}_0, \mathsf{sk}_0, \mathsf{pk}_1, \mathsf{sk}_1, \mathcal{M})\text{:}}$

for $i = 1,\dots,r$:
  $m_{0,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{0,i})$
  $m_{1,i} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{1,i})$
  $(c_{0,i}, \mathfrak{s}_{0,i}) \leftarrow \Phi^{\mathsf{E}}_{0,i}(\mathsf{msg}, m_{0,i}, \mathsf{pk}_1, \mathfrak{s}_{0,i-1})$
  $(c_{1,i}, \mathfrak{s}_{1,i}) \leftarrow \Phi^{\mathsf{E}}_{1,i}(m_{1,i}, \mathsf{pk}_0, \mathfrak{s}_{1,i-1})$
for $j = 1,\dots,p$:
  let $\beta = b_j$ and $\bar\beta = 1 - b_j$
  for $i' = 1,\dots,r'$:
    let $\iota = r + (j-1)r' + i'$
    $m_{\beta,\iota} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{\beta,\iota})$
    $m_{\bar\beta,\iota} \leftarrow \mathcal{M}(\boldsymbol{\tau}_{\bar\beta,\iota})$
    $(c_{\beta,\iota}, \mathfrak{s}_{\beta,\iota}) \leftarrow \Xi^{\mathsf{E}}_{\beta,i'}(\mathsf{msg}, m_{\beta,\iota}, \mathsf{pk}_{\bar\beta}, \mathfrak{s}_{\beta,\iota-1})$
    $(c_{\bar\beta,\iota}, \mathfrak{s}_{\bar\beta,\iota}) \leftarrow \Xi^{\mathsf{E}}_{\bar\beta,i'}(m_{\bar\beta,\iota}, \mathsf{pk}_{\beta}, \mathfrak{s}_{\bar\beta,\iota-1})$
output:
  $\big(\mathsf{pk}_0, \mathsf{sk}_0, \mathsf{pk}_1, \mathsf{sk}_1; (c_{b,i})_{b\in\{0,1\}, i\in[r+pr']}\big)$

If the *subliminal indistinguishability* requirement is satisfied only for M in a restricted set $\mathbb{M}$, rather than for any $\mathcal{M}$, then (Φ, Ξ) is said to be a *multi-message subliminal communication scheme for* $\mathbb{M}$.

4 Impossibility Results

4.1 Locally Decodable Subliminal Communication Schemes

A first attempt at achieving subliminal communication might consider schemes with the following natural property: the receiving party P_1 extracts hidden bits *one ciphertext at a time*, by the application of a single decoding function. We refer to such schemes as *locally decodable* and our next impossibility theorem shows that non-trivial locally decodable schemes do not exist if the encryption scheme E is chosen adversarially.

▶ **Theorem 9.** *For any locally decodable protocol* Π *satisfying the syntax of a single-message subliminal communication scheme (Definition 7), there exists a semantically secure public-key encryption scheme* E *such that* E *violates the correctness condition of Definition 7. Therefore, no locally decodable protocol* Π *is a subliminal communication scheme.*

▶ Remark. The essence of the above theorem is the impossibility of deterministic extraction: no single deterministic function can deterministically extract from ciphertexts of arbitrary encryption schemes. The way to bypass this impossibility is to have the extractor depend on the encryption scheme. Note that multiple-source extraction, which is used in our constructions in the subsequent sections, implicitly do depend on the underlying encryption scheme, since the additional sources of input depend on the encryption scheme and thus can be thought of as "auxiliary input" that is specific to the encryption scheme at hand.

4.2 Steganography for Adversarial Cover Distributions

Our second impossibility result concerns a much more general class of communication schemes, which we call *steganographic communication schemes*. Subliminal communication schemes, as well as the existing notions of public-key steganography and steganographic key exchange from the steganography literature, are instantiations of the more general definition of a (multi-message) steganographic communication scheme. To our knowledge, the general notion of a steganographic communication scheme has not been formalized in this way in prior work. In the context of this work, the general definition is helpful for proving broad impossibilities across multiple types of steganographic schemes.

As mentioned in the introduction, a limitation of all existing results in the steganographic literature, to our knowledge, is that they assume that the *cover distribution* – i.e., the distribution of innocuous objects in which steganographic communication is to be embedded – is fixed *a priori*. In particular, the cover distribution is assumed not to depend on the description of the steganographic communication scheme. The impossibility result given in Section 4.1 is an example illustrative of the power of adversarially choosing the cover distribution: Theorem 9 says that by choosing the encryption scheme E to depend on a given subliminal communication scheme, an adversary can rule out the possibility of any hidden communication at all.

Our next impossibility result (Theorem 10) shows that if the cover distribution is chosen adversarially, then non-trivial steganographic communication is impossible.

▶ **Theorem 10.** *Let* Π *be a protocol with the syntax of a steganographic communication scheme. Then for any* $k \in \mathbb{N}$*, there exists a cover distribution* $\mathcal{C}$ *of conditional min-entropy* k *such that steganographic indistinguishability of* Π *does not hold.*

We have elected to present the definition of a *steganographic communication scheme* as well as the proof of Theorem 10 in the full version of this paper [17] since the definition introduces a set of new notation only used for the corresponding impossibility result, and both the definition and the impossibility result are somewhat tangential to the main results of this work, whose focus is on subliminal communication schemes.

5 Construction of the Subliminal Scheme

The goal of this section is to establish the following theorem, which states that our construction $(\Phi^\star, \Xi^\star)$ is a subliminal communication scheme when instantiated with a pseudorandom key-exchange protocol (such as Diffie-Hellman).

▶ **Theorem 11.** *The protocol* $(\Phi^\star, \Xi^\star)$ *given in Definition 17, when instantiated with a pseudorandom key-exchange protocol* Λ*, is a multi-message subliminal communication scheme.*

The description of our scheme can be found in the following subsections. Our construction makes no assumption on the message distribution $\mathcal{M}$ and in particular holds when the exchanged plaintexts (of the adversarial encryption scheme E) are a fixed, adversarially chosen sequence of messages. An informal outline of the construction is given next.

▶ **Definition 12.** Outline of the construction.

1. **Setup Phase** $\Phi^\star$
   - **a.** A $\widetilde{O}(\log \kappa)$-bit string S is established between P_0 and P_1 by extracting randomness from pairs of consecutive ciphertexts. (*Protocol overview in Section 5.1.*)

b. Let Ext be a strong seeded extractor, and let S serve as its seed. By rejection-sampling ciphertexts c until $\mathsf{Ext}_S(c) = \mathsf{str}$, either party can embed a random string str *of their choice* in the conversation. (*Protocol overview in Section 5.2.*) By embedding in this manner the messages of a pseudorandom key-exchange protocol, both parties establish a shared secret sk^*.[12]

2. Communication Phase $\Xi^\star$

Both parties can now communicate arbitrary messages of their choice by *(1)* encrypting them using a pseudorandom secret-key encryption scheme SKE using sk^* as the secret key, and *(2)* embedding the ciphertexts of SKE using the rejection-sampling technique described in Step 1b.[13] (*Detailed protocol in Section 5.3.*)

The full protocol is given in Section 5.3.

5.1 Establishing a Shared Seed

In this section, we give a protocol which allows P_0 and P_1 to establish a random public parameter which will be used in subsequent phases of our subliminal scheme. This can be thought of as drawing a subliminal scheme at random from a family of subliminal schemes. The parameter is public in the sense that any eavesdropper gains knowledge of it. A crucial point is that the random draw occurs *after* the adversarial encryption scheme E is fixed, thus bypassing the impossibility results of Section 4.

Our strategy is simple: extract randomness from pairs of ciphertexts. Since the extractor does not receive the key, semantic security holds with respect to the extractor: a pair of ciphertexts for two arbitrary messages is indistinguishable from two encryptions of a fixed message; thus, a same-source extractor suffices for our purposes. Even though semantic security guarantees only $\omega(\log \kappa)$ min-entropy of ciphertexts, we can use the "greater-than" extractor (Definition 13) applied to pairs of ciphertexts, and obtain Theorem 14.

▶ **Definition 13.** The *greater-than extractor* GT is defined by $\mathsf{GT}(x, y) = \mathbf{1}[x \geq y]$.

▶ **Theorem 14.** *Let E be a semantically secure public-key encryption scheme and let $\rho \in \mathsf{poly}$. Define random variables as follows.*

- *For $b \in \{0, 1\}$, let $K_b = (PK_b, SK_b) = \mathsf{E.Gen}(1^\kappa)$.*
- *For $b \in \{0, 1\}$ and $i \in [2\rho]$, let $C_{b,i} = \mathsf{E.Enc}(PK_{1-b}, m_{b,i})$ representing the ciphertexts exchanged between P_0 and P_1 during 2ρ exchange-rounds.*
- *Let $S = \big(\mathsf{GT}(C_{0,1}, C_{0,2}), \mathsf{GT}(C_{1,1}, C_{1,2}), \ldots, \mathsf{GT}(C_{1,2\rho-1}, C_{1,2\rho})\big)$.*

There exists a negligible function ε such that $\|(K_0, K_1, S) - (K_0, K_1, U_{2\rho})\|_{\mathsf{s}} \leq \varepsilon$.

5.2 Embedding Random Strings

In this section, we assume that both parties have access to a public parameter S and construct a protocol which allows for embedding of uniformly random strings into ciphertexts of an adversarially chosen encryption scheme E, as required by Steps 1b and 2 of the construction outline (Definition 12). The security guarantee is that for a uniformly random parameter S and uniformly random strings to be embedded, the ciphertexts of E with embedded random

[12] An eavesdropper who knows S can compute str. Nonetheless, (1) sk^* is unknown to the eavesdropper by the security of the key-exchange protocol and (2) the transcript is indistinguishable from one in which no key exchange occurred at all, due to the pseudorandomness of the key-exchange messages.

[13] Again, an eavesdropper could know the SKE ciphertexts exchanged, if he knew the seed S, but could not distinguish the SKE ciphertexts from truly random strings, and thus could not tell whether any subliminal communication was occuring at all. See also footnote 12.

Algorithm 1 Rejection sampler $\Sigma^{\mathsf{E},S}$.

PUBLIC PARAMETER: S (a d-bit seed).
INPUT: $(\mathsf{str}, m, \mathsf{pk})$ where str is the string to be embedded.

1. Generate encryption $c \leftarrow \mathsf{E.Enc}(\mathsf{pk}, m)$.
2. If $\mathsf{Ext}(S, c) = \mathsf{str}$, then output c. Else, go back to step 1.

strings are indistinguishable from ciphertexts of E produced by direct application of $\mathsf{E.Enc}$, even to an adversary who knows the decryption keys of E. This can be thought of as a relaxation of subliminal indistinguishability (Definition 7) where the two main differences are that *(1)* the parties have shared knowledge of a random seed, and *(2)* indistinguishability only holds when embedding a *random* string, rather than for arbitrary strings. Our construction (Theorem 15) relies on a strong seeded extractor that can extract logarithmically many bits from sources of super-logarithmic min-entropy, we note that almost universal hashing is a simple such extractor.

▶ **Theorem 15.** *Let* $\mathsf{Ext} : \{0,1\}^d \times \{0,1\}^n \to \{0,1\}^v$ *be a strong seeded extractor for super-logarithmic min-entropy with* $v = O(\log \kappa)$*, and let* E *be a semantically secure encryption scheme with ciphertext space* $\mathcal{C} = \{0,1\}^n$*. Let* $\Sigma^{\mathsf{E},S}$ *be defined as in Algorithm 1. Then the following guarantees hold:*

1. Correctness: *for any* $S \in \{0,1\}^d$ *and* $\mathsf{str} \in \{0,1\}^v$*, if* $c = \Sigma^{\mathsf{E},S}(\mathsf{str}, m, \mathsf{pk})$*, and* $\mathsf{str}' = \mathsf{Ext}(S, c)$*, then* $\mathsf{str}' = \mathsf{str}$*.*
2. Security: *let* $(PK, SK) = \mathsf{E.Gen}(1^\kappa)$*,* $C = \mathsf{E.Enc}(PK, m)$ *and* $C' = \Sigma^{\mathsf{E},U_d}(U_v, m, PK)$*; then* $\|(PK, SK, U_d, C) - (PK, SK, U_d, C')\|_{\mathsf{s}} \leq \varepsilon(\kappa)$ *for some negligible function* ε*.*

▶ Remark. Rejection sampling is a simple and natural approach that has been used by prior work in the steganographic literature, such as [2]. Despite the shared use of this common technique, our construction is more different from prior art than it might seem at first glance. The novelty of our construction arises from the challenges of working in a model with a stronger adversary who can choose the distribution of ciphertexts (i.e., the adversary gets to choose the public-key encryption scheme E). We manage to bypass the impossibilities outlined in Section 4 notwithstanding this stronger adversarial model, and in contrast to prior work, construct a protocol to established a shared seed from scratch, rather than simply assuming that one has been established in advance.

5.3 Full Protocol $(\Phi^\star, \Xi^\star)$

▶ **Definition 16** (Key-exchange protocol syntax)**.** A key-exchange protocol is a two-party protocol defined by $\Lambda = (\Lambda_{0,1}, \Lambda_{1,1}, \Lambda_{0,2}, \Lambda_{1,2}, \ldots, \Lambda_{0,k}, \Lambda_{1,k}, \Lambda_{0,\mathsf{out}}, \Lambda_{1,\mathsf{out}})$. We assume k simultaneous communication rounds, where $\Lambda_{b,i}$ represents the computation performed by P_b in the ith round. The parties are stateful and their state is implicitly updated at each round to contain the transcript so far and any local randomness generated so far. Each $\Lambda_{b,i}$ takes as input the transcript up to round $i-1$ and the state of P_b, and outputs a message $\lambda_{b,i}$ to be sent in the ith round. For notational simplicity, we write explicitly only the first input to $\Lambda_{b,i}$, and leave the second input (i.e., the state) implicit. $\Lambda_{0,\mathsf{out}}, \Lambda_{1,\mathsf{out}}$ are run by P_0, P_1 respectively to compute the shared secret at the conclusion of the protocol.

Next, we give the full construction of $(\Phi^\star, \Xi^\star)$ following the outline in Definition 12.

▶ **Definition 17.** $(\Phi^\star, \Xi^\star)$ is parametrized by the following.

- $\mathsf{Ext} : \{0,1\}^d \times \{0,1\}^n \to \{0,1\}^v$, a strong seeded extractor.
- Λ, a pseudorandom key-exchange protocol with ℓ-bit messages.[14]
- SKE, a pseudorandom secret key encryption scheme with ξ-bit ciphertexts.

We define each phase of our construction in turn.

1. **Setup Phase $\Phi^\star$**
   a. **Establishing a d-bit shared seed**
      - For $b \in \{0,1\}$ and $i \in \{1,\ldots,d\}$, $\Phi^\star_{b,i}(m_{b,i}, \mathsf{pk}_{1-b}, \mathfrak{s}_{b,i-1})$ outputs a ciphertext $c_{b,i} = \mathsf{E.Enc}(\mathsf{pk}_{1-b}, m_{b,i})$ and sets the updated state $\mathfrak{s}_{b,i}$ to be the transcript of all protocol messages sent and received so far.
      - At the conclusion of the d exchange-rounds, each party updates his state to contain the seed S which is defined by

$$S = \big(\mathsf{GT}(c_{0,1}, c_{0,2}), \mathsf{GT}(c_{1,1}, c_{1,2}), \ldots, \mathsf{GT}(c_{1,d-1}, c_{1,d})\big) .$$

        This seed S is assumed to be accessible in all future states throughout both phases during the remainder of the protocol.

   b. **Subliminal key exchange**
      Let $\nu = \frac{\ell}{v}$. Subliminal key exchange occurs over $k \cdot \nu$ exchange-rounds.
      - For $j \in \{1,\ldots,k\}$ and $b \in \{0,1\}$:
        - P_b retrieves from his state the key-exchange transcript so far $(\lambda_{b,j'})_{b\in\{0,1\},j'<j}$.
        - P_b computes the next key-exchange message $\lambda_{b,j} \leftarrow \Lambda_{b,j}\big((\lambda_{b,j'})_{b\in\{0,1\},j'<j}\big)$.
        - P_b breaks $\lambda_{b,j}$ into v-bit blocks $\lambda_{b,j} = \lambda^1_{b,j}||\ldots||\lambda^\nu_{b,j}$.
        - The ν blocks are transmitted sequentially as follows. For $\iota \in \{1,\ldots,\nu\}$:
          Let $i = d + (j-1)\nu + \iota$.
          $\Phi^\star_{b,i}(m_{b,i}, \mathsf{pk}_{1-b}, \mathfrak{s}_{b,i-1})$ outputs $c_{b,i} \leftarrow \Sigma^{\mathsf{E},S}(\lambda^\iota_{b,j}, m_{b,i}, \mathsf{pk}_{1-b})$ and sets the updated state $\mathfrak{s}_{b,i}$ to contain the transcript of all protocol messages sent and received so far.
        - At the conclusion of the ι exchange-rounds, each party $b \in \{0,1\}$ updates his state to contain the jth key-exchange message $\lambda_{1-b,j}$ computed as follows:

$$\lambda_{1-b,j} = \mathsf{Ext}(S, c_{b,d+(j-1)\nu+1})||\ldots||\mathsf{Ext}(S, c_{b,d+j\nu})$$

      - At the conclusion of the $k\cdot\nu$ exchange rounds, each party updates his state to contain the secret key sk^* computed as: $\mathsf{sk}^* = \mathsf{SKE.Gen}\left(1^\kappa; \Lambda_{\mathsf{out}}\big((\lambda_{b,j})_{b\in\{0,1\},j\in[k]}\big)\right)$.

2. **Communication Phase $\Xi^\star$**
   Each communication phase occurs over $r' = \xi/v$ exchange-rounds.
   Let $\beta \in \{0,1\}$ be the initiating party and let $\bar\beta = 1 - \beta$.
   P_β performs the following steps.
   - Generate $c^* \leftarrow \mathsf{SKE.Enc}(\mathsf{sk}^*, \mathsf{msg})$.
   - Break c^* into v-bit blocks $c^* = c^*_1||\ldots||c^*_{r'}$.

   For $i' \in \{1,\ldots,r'\}$:

   - Let $i'' = r + i'$.
   - $\Xi^\star_{0,i'}(\mathsf{msg}, m_{0,i''}, \mathsf{pk}_{\bar\beta}, \mathfrak{s}_{\beta,i''-1})$ outputs $c_{\beta,i''} \leftarrow \Sigma^{\mathsf{E},S}(c^*_{i'}, m_{\beta,i''}, \mathsf{pk}_{\bar\beta})$.
   - $\Xi^\star_{1,i'}(m_{\bar\beta,i''}, \mathsf{pk}_\beta, \mathfrak{s}_{\bar\beta,i''-1})$ outputs $c_{\bar\beta,i''} \leftarrow \mathsf{E.Enc}(\mathsf{pk}_\beta, m_{\bar\beta,i''})$.

[14] In presenting our construction $(\Phi^\star, \Xi^\star)$, we do not denote the state of parties w.r.t. the key-exchange protocol Λ by a separate variable, but assume that it is part of the state $\mathfrak{s}_{b,i}$ of the overall protocol.

- Both parties update their state to contain the transcript of all protocol messages exchanged so far.

After the r' exchange-rounds, $P_{\bar{\beta}}$ computes $c^{**} = \mathsf{Ext}(S, c_{\beta,r+1})||\ldots||\mathsf{Ext}(S, c_{\beta,r+r'})$. Then, $P_{\bar{\beta}}$ outputs $\mathsf{msg}' \leftarrow \mathsf{SKE.Dec}(\mathsf{sk}^*, c^{**})$. (That is, $\Xi^{\star}_{1,\mathsf{out}}(\mathfrak{s}_{\bar{\beta},r'}) = \mathsf{msg}'$.)

6 Open problems

Deterministic Extraction. Our impossibility result in Theorem 9 holds because the adversary can choose the encryption scheme E as a function of a given candidate subliminal scheme. However, note that under the additional assumption that E is restricted to a predefined class E of encryption schemes, we could bypass this impossibility as long as a deterministic extractor that can extract randomness from ciphertexts of any encryption scheme in E exists. We are only aware of two deterministic extractors leading to a positive result for restricted classes of encryption schemes:

- if an upper bound on the circuit size of E is known, then we can use the deterministic extractor from [21]. This extractor relies on strong complexity-theoretic assumptions and requires the sources to have min-entropy $(1-\gamma)n$ for some unspecified constant γ.
- if E is computed by a circuit of constant depth ($\mathbf{AC^0}$), then the deterministic extractor of [22] can be used and requires $\sqrt{n}$ min-entropy.

Both these extractors have a min-entropy requirement which is not satisfied by ciphertexts of arbitrary encryption schemes. However, it would be interesting to consider improved constructions for the case of specific encryption schemes, or to consider extractors specifically for encryption circuits as opposed to arbitrary circuits satisfying a min-entropy requirement. This would also have direct implications for the efficiency of the subliminal scheme of Section 5.2: indeed, one could then skip Step 1a and use a deterministic extractor directly in Steps 1b and 2, thus saving $\widetilde{O}(\log \kappa)$ exchange-rounds in the setup phase.

Multi-Source Extraction. Another interesting question is whether multi-source extractors for the specific case when the sources are independent and identically distributed can achieve better parameters than extractors for general independent sources. We already saw that a very simple extractor (namely, the "greater-than" function) works for i.i.d. sources and extracts one bit with negligible bias, even when the sources only have $\omega(\log \kappa)$ min-entropy. The non-constructive result of [9] guarantees the existence of a two-source extractor of negligible bias and output length $\omega(\log \kappa)$ for sources of min-entropy $\omega(\log \kappa)$. However, known *explicit* constructions are far from achieving the same parameters, and improving them in the specific case of identically distributed sources is an interesting open problem which was also mentioned in [3].

References

1. Per Austrin, Kai-Min Chung, Mohammad Mahmoody, Rafael Pass, and Karn Seth. On the Impossibility of Cryptography with Tamperable Randomness. In *CRYPTO 2014, Proceedings, Part I*, volume 8616 of *Lecture Notes in Computer Science*, pages 462–479. Springer, 2014. doi:10.1007/978-3-662-44371-2_26.
2. Michael Backes and Christian Cachin. Public-Key Steganography with Active Attacks. In *TCC 2005, Proceedings*, volume 3378 of *Lecture Notes in Computer Science*, pages 210–226. Springer, 2005. doi:10.1007/978-3-540-30576-7_12.

3 Boaz Barak, Russell Impagliazzo, and Avi Wigderson. Extracting Randomness Using Few Independent Sources. In *45th Symposium on Foundations of Computer Science (FOCS 2004), Proceedings*, pages 384–393, 2004. doi:10.1109/FOCS.2004.29.

4 Mihir Bellare, Joseph Jaeger, and Daniel Kane. Mass-surveillance without the State: Strongly Undetectable Algorithm-Substitution Attacks. In Indrajit Ray, Ninghui Li, and Christopher Kruegel, editors, *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS) 2015*, pages 1431–1440. ACM, 2015. doi:10.1145/2810103.2813681.

5 Mihir Bellare, Kenneth G. Paterson, and Phillip Rogaway. Security of Symmetric Encryption against Mass Surveillance. In *CRYPTO 2014, Proceedings, Part I*, volume 8616 of *Lecture Notes in Computer Science*, pages 1–19. Springer, 2014. doi:10.1007/978-3-662-44371-2_1.

6 Sebastian Berndt and Maciej Liśkiewicz. Algorithm Substitution Attacks from a Steganographic Perspective. In Bhavani M. Thuraisingham, David Evans, Tal Malkin, and Dongyan Xu, editors, *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, CCS 2017, Dallas, TX, USA, October 30 - November 03, 2017*, pages 1649–1660. ACM, 2017. doi:10.1145/3133956.3133981.

7 Christian Cachin. An Information-Theoretic Model for Steganography. In David Aucsmith, editor, *Information Hiding, Second International Workshop, Portland, Oregon, USA, April 14-17, 1998, Proceedings*, volume 1525 of *Lecture Notes in Computer Science*, pages 306–318. Springer, 1998. doi:10.1007/3-540-49380-8_21.

8 Eshan Chattopadhyay and David Zuckerman. Explicit two-source extractors and resilient functions. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016*, pages 670–683, 2016.

9 Benny Chor and Oded Goldreich. Unbiased Bits from Sources of Weak Randomness and Probabilistic Communication Complexity. *SIAM J. Comput.*, 17(2):230–261, 1988. doi:10.1137/0217015.

10 Aloni Cohen and Saleet Klein. The GGM Function Family Is a Weakly One-Way Family of Functions. In *Theory of Cryptography - 14th International Conference, TCC 2016-B, Part I*, volume 9985 of *Lecture Notes in Computer Science*, pages 84–107, 2016. doi:10.1007/978-3-662-53641-4_4.

11 Whitfield Diffie and Martin E. Hellman. New directions in cryptography. *IEEE Trans. Information Theory*, 22(6):644–654, 1976. doi:10.1109/TIT.1976.1055638.

12 Yevgeniy Dodis, Ariel Elbaz, Roberto Oliveira, and Ran Raz. Improved Randomness Extraction from Two Independent Sources. In *7th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2004, and 8th International Workshop on Randomization and Computation, RANDOM 2004, Proceedings*, pages 334–344, 2004.

13 Yevgeniy Dodis, Chaya Ganesh, Alexander Golovnev, Ari Juels, and Thomas Ristenpart. A Formal Treatment of Backdoored Pseudorandom Generators. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Proceedings, Part I*, volume 9056 of *Lecture Notes in Computer Science*, pages 101–126. Springer, 2015. doi:10.1007/978-3-662-46800-5_5.

14 Oded Goldreich. The ggm construction does not yield correlation intractable function ensembles. In Oded Goldreich, editor, *Studies in Complexity and Cryptography*, pages 98–108. Springer-Verlag, Berlin, Heidelberg, 2011. URL: http://dl.acm.org/citation.cfm?id=2028116.2028129.

15 Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986. doi:10.1145/6490.6503.

16 Nicholas J. Hopper, John Langford, and Luis von Ahn. Provably Secure Steganography. In *CRYPTO 2002, Santa Barbara, California, USA, Proceedings*, volume 2442 of *Lecture Notes in Computer Science*, pages 77–92. Springer, 2002. doi:10.1007/3-540-45708-9_6.

17 Thibaut Horel, Sunoo Park, Silas Richelson, and Vinod Vaikuntanathan. How to Subvert Backdoored Encryption: Security Against Adversaries that Decrypt All Ciphertexts. Cryptology ePrint Archive, Report 2018/212, 2018. URL: https://eprint.iacr.org/2018/212.

18 Silvio Micali. ALGORAND: the efficient and democratic ledger. *CoRR*, abs/1607.01341, 2016. arXiv:1607.01341.

19 Thomas Mittelholzer. An Information-Theoretic Approach to Steganography and Watermarking. In *Information Hiding, Third International Workshop, IH'99, Proceedings*, volume 1768 of *Lecture Notes in Computer Science*, pages 1–16. Springer, 1999. doi:10.1007/10719724_1.

20 Gustavus J. Simmons. The Prisoners' Problem and the Subliminal Channel. In *CRYPTO 1983.*, pages 51–67. Plenum Press, New York, 1983.

21 Luca Trevisan and Salil P. Vadhan. Extracting Randomness from Samplable Distributions. In *FOCS 2000, Redondo Beach, California, USA*, pages 32–42, 2000.

22 Emanuele Viola. Extractors for Circuit Sources. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011*, pages 220–229. IEEE Computer Society, 2011. doi:10.1109/FOCS.2011.20.

23 Luis von Ahn and Nicholas J. Hopper. Public-Key Steganography. In *EUROCRYPT 2004, Proceedings*, volume 3027 of *Lecture Notes in Computer Science*, pages 323–341. Springer, 2004. doi:10.1007/978-3-540-24676-3_20.

24 Adam L. Young and Moti Yung. Cryptovirology: Extortion-Based Security Threats and Countermeasures. In *1996 IEEE Symposium on Security and Privacy, May 6-8, 1996, Oakland, CA, USA*, pages 129–140. IEEE Computer Society, 1996. doi:10.1109/SECPRI.1996.502676.

25 Adam L. Young and Moti Yung. The Dark Side of "Black-Box" Cryptography, or: Should We Trust Capstone? In *CRYPTO 1996, Proceedings*, volume 1109 of *Lecture Notes in Computer Science*, pages 89–103. Springer, 1996. doi:10.1007/3-540-68697-5_8.

26 Adam L. Young and Moti Yung. Kleptography: Using Cryptography Against Cryptography. In *EUROCRYPT '97, Proceedings*, volume 1233 of *Lecture Notes in Computer Science*, pages 62–74. Springer, 1997. doi:10.1007/3-540-69053-0_6.

27 Jan Zöllner, Hannes Federrath, Herbert Klimant, Andreas Pfitzmann, Rudi Piotraschke, Andreas Westfeld, Guntram Wicke, and Gritta Wolf. Modeling the Security of Steganographic Systems. In David Aucsmith, editor, *Information Hiding, Second International Workshop, Portland, Oregon, USA, April 14-17, 1998, Proceedings*, volume 1525 of *Lecture Notes in Computer Science*, pages 344–354. Springer, 1998. doi:10.1007/3-540-49380-8_24.