Tight Bounds for Unconditional Authentication Protocols in the Manual Channel and Shared Key Models

Moni Naor* Gil Segev† Adam Smith‡

Abstract

We address the message authentication problem in two seemingly different communication models. In the first model, the sender and receiver are connected by an insecure channel and by a low-bandwidth auxiliary channel, that enables the sender to "manually" authenticate one short message to the receiver (for example, by typing a short string or comparing two short strings). We consider this model in a setting where no computational assumptions are made, and prove that for any $0 < \epsilon < 1$ there exists a $\log^* n$-round protocol for authenticating n-bit messages, in which only $2\log(1/\epsilon) + O(1)$ bits are manually authenticated, and any adversary (even computationally unbounded) has probability of at most ϵ to cheat the receiver into accepting a fraudulent message. Moreover, we develop a proof technique showing that our protocol is essentially optimal by providing a lower bound of $2\log(1/\epsilon) - O(1)$ on the required length of the manually authenticated string.

The second model we consider is the traditional message authentication model. In this model the sender and the receiver share a short secret key; however, they are connected only by an insecure channel. We apply the proof technique above to obtain a lower bound of $2\log(1/\epsilon) - O(1)$ on the required Shannon entropy of the shared key. This settles an open question posed by Gemmell and Naor (CRYPTO '93).

Finally, we prove that one-way functions are *necessary* (and sufficient) for the existence of protocols breaking the above lower bounds in the computational setting.

Keywords: Authentication, Cryptographic protocols, Lower bounds, Unconditional security.

A preliminary version of this work appeared in *Advances in Cryptology - CRYPTO '06*, pages 214–231, 2006.

*Incumbent of the Judith Kleeman Professorial Chair, Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot 76100, Israel. Email: `moni.naor@weizmann.ac.il`. Research supported in part by a grant from the Israel Science Foundation.

†Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot 76100, Israel. Email: `gil.segev@weizmann.ac.il`.

‡Department of Computer Science and Engineering, Pennsylvania State University, University Park, PA 16803. Email: `asmith@cse.psu.edu`. Research was done at the Weizmann Institute of Science and supported by the Louis L. and Anita M. Perlman postdoctoral fellowship.

1 Introduction

Message authentication is one of the major issues in cryptography. Protocols for message authentication provide assurance to the receiver of a message that it was sent by a specified legitimate sender, even in the presence of an adversary who controls the communication channel. For more than three decades, numerous authentication models have been investigated, and many authentication protocols have been suggested. The security of these protocols can be classified according to the assumed computational resources of the adversary. Security that holds when one assumes a suitable restriction on the adversary's computing capabilities is called *computational security*, while security that holds even when the adversary is computationally unbounded is called *unconditional security* or *information-theoretic security*. This paper is concerned mostly with unconditional security of a single instance of message authentication protocols. We remark that there are three main advantages to unconditional security over computational security. The first is the obvious fact that no assumptions are made about the adversary's computing capabilities or about the computational hardness of specific problems. The second, less apparent advantage, is that unconditionally secure protocols are often more efficient than computationally secure protocols. The third advantage is that unconditional security allows exact evaluation of the error probabilities.

Shared key authentication. The first construction of an authentication protocol in the literature was suggested by Gilbert, MacWilliams and Sloane [12] in the information-theoretic adversarial setting. They considered a communication model in which the sender and the receiver share a key, which is not known to the adversary. Gilbert et al. presented a non-interactive protocol in which the length of the shared key is $2\max\{n, \log(1/\epsilon)\}$; henceforth, n is the length of the input message and ϵ is the adversary's probability of cheating the receiver into accepting a fraudulent message. They also proved a lower bound of $2\log(1/\epsilon)$ on the required entropy of the shared key in non-interactive deterministic protocols. Clearly, a lower bound on this entropy is $\log(1/\epsilon)$, since an adversary can merely guess the shared key. This model, to which we refer as the *shared key model*, became the standard model for message authentication protocols. Protocols in this model should provide authenticity of messages while minimizing the length of the shared key.

Wegman and Carter [34] suggested using ϵ-*almost strongly universal*$_2$ hash functions for authentication. This enabled them to construct a non-interactive protocol in which the length of the shared key is $O(\log n \log(1/\epsilon))$ bits. In 1984, Simmons [29] initiated a line of work on unconditionally secure authentication protocols (see, for example, [9, 17, 19, 26, 27, 30, 31]). Gemmell and Naor [11] proposed a non-interactive protocol with a shared key of length only $\log n + 5\log(1/\epsilon)$ bits. They also demonstrated that interaction can reduce the length of the shared key and make it independent of the length of the input message. More specifically, they described a $\log^* n$-round protocol that enables the sender to authenticate n-bit messages, where the length of the shared key is only $2\log(1/\epsilon) + O(1)$ bits. However, it was not known whether this upper bound is optimal, that is, if by introducing interaction the entropy of the shared key can be made smaller than $2\log(1/\epsilon)$.

Manual authentication. In 1984, Rivest and Shamir [24] were the first to incorporate human abilities into an authentication protocol. They constructed the "Interlock" protocol which enables two parties, who can recognize each other's voice, to mutually authenticate their public encryption keys in absence of trusted infrastructure[1]. Although such a communication model seems very

[1]However, the security of the protocol relies on very special non-malleability properties of the underlying encryption scheme. To the best of our knowledge, there is no construction for such an encryption scheme in the plain model.

realistic, until recently it never received a formal treatment in the literature.

In 2005, Vaudenay [33] formalized such a realistic communication model for message authentication, in which the sender and the receiver are connected by a bidirectional insecure channel, and by a unidirectional low-bandwidth auxiliary channel, but do not share any secret information. It is assumed that the adversary has full control over the insecure channel. In particular, the adversary can read any message sent over this channel, prevent it from being delivered, and insert a new message at any point in time. The low-bandwidth auxiliary channel enables the sender to "manually" authenticate one short string to the receiver (for example, by typing a short string or comparing two short strings). The adversary cannot modify this short string. However, the adversary can still read it, delay it, and remove it. We refer to the auxiliary channel as the *manual channel*, and to this communication model as the *manual channel model*. Protocols in this model should provide authenticity of long messages[2] while minimizing the length of the manually authenticated string. We remark that $\log(1/\epsilon)$ is an obvious lower bound in this model as well (see [22]).

The manual channel model is becoming very popular in real-world scenarios, whenever there are ad hoc networks with no trusted infrastructure[3]. In particular, this model was found suitable for initial pairing of devices in wireless networks (see, for example, [14, 15]), such as Wireless USB [4] and Bluetooth [2]. While in wired connections when a device is plugged in (i.e., when the wire is connected), the user can see that the connection is made, wireless connections may establish connection paths that are not straightforward. In fact, it may not be obvious when a device is connected or who its host is. Therefore, initial authentication in device and host connections is required so that the user will be able to validate both the device and its host.

Consider, for example, a user who wishes to connect a new DVD player to her home wireless network. Then, having the user read a short message from the display of the DVD player and type it on a PC's keyboard constitutes a manual authentication channel from the DVD player to the PC. An equivalent channel consists of the user comparing two short strings displayed by the two devices, as suggested by Gehrmann, Mitchell and Nyberg [10], and by Cagalj, Capkun and Hubaux [3]. Other possible implementations of the manual channel may include a visual channel [20] and an audio channel [13].

Constants do matter. The most significant constraint in the manual channel model is the length of the manually authenticated string. This quantity is determined by the environment in which the protocol is executed, and in particular by the capabilities of the user. While it is reasonable to expect a user to manually authenticate 20 or 40 bits, it is not reasonable to expect a user to manually authenticate 160 bits. Therefore, there is a considerable difference between manually authenticating $\log(1/\epsilon)$ or $2\log(1/\epsilon)$ bits, and manually authenticating a significantly longer string. This motivates the study of determining the exact lower bound on the required length of the manually authenticated string.

Our contribution. We present an unconditionally secure authentication protocol in the manual channel model, in which the sender manually authenticates only $2\log(1/\epsilon) + O(1)$ bits. Moreover, we prove that our protocol essentially minimizes the length of the manually authenticated string. The proof of optimality involves a careful accounting of how randomness is introduced into the protocol. In particular, our proof technique identifies the exact amount of randomness that each party is

[2]Short messages can be manually authenticated without the use of any authentication protocol.

[3]For example, the recently suggested ZRTP protocol [35] is an RTP (Real-time Transport Protocol) [25] header extension for a Diffie-Hellman exchange to agree on a session key.

required to contribute to the manually authenticated string. In the shared key setting, we use it to settle an open question posed by Gemmell and Naor [11] by deriving a similar lower bound of $2\log(1/\epsilon) - 2$ on the required entropy of the shared key, which matches their upper bound. Finally, we consider these two communication models in the computational setting, and prove that one-way functions are necessary for the existence of protocols breaking the above lower bounds.

Thus, we have now gained a complete understanding of unconditionally secure message authentication in the manual channel and shared key models, for each set of parameters ℓ and ϵ, where ℓ is the length of the manually authenticated string or the length of the shared key, and ϵ is the adversary's probability of cheating the receiver into accepting a fraudulent message. In addition, we have indicated an exact relation between the computational setting and the information-theoretic setting for authentication in both models. Figure 1 illustrates the achievable security according to our results.

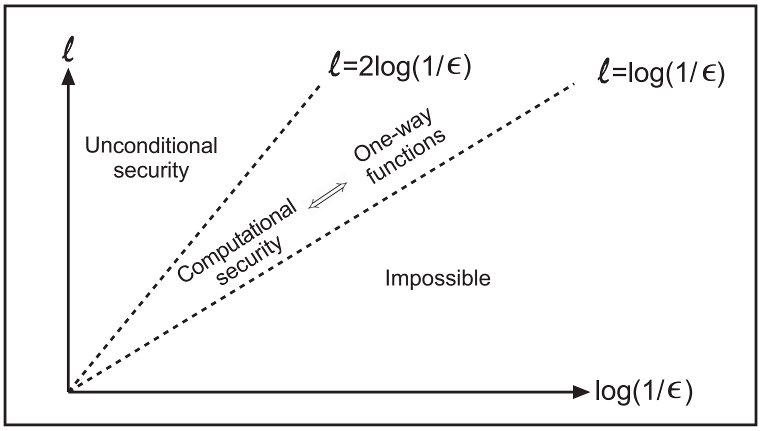

Figure 1: The achievable security for any parameters ℓ and ϵ.

Paper organization. The rest of the paper is organized as follows. We first briefly present some known definitions in Section 2. In Section 3 we describe the communication and adversarial models we deal with. Then, in Section 4 we present an overview of our results, and compare them to previous work. In Section 5 we propose an unconditionally secure message authentication protocol in the manual channel model. In Section 6 we describe the proof technique, that is then used to establish the optimality of our protocol. In Section 7 we apply the same proof technique to the shared key model, and prove a lower bound on the required entropy of the shared key. Finally, in Section 8 we prove that in the computational setting, one-way functions are necessary for the existence of protocols breaking the above lower bounds.

2 Preliminaries

We first present some notations used in this paper and several fundamental definitions from Information Theory. Then, we briefly present the definitions of *one-way* functions, *statistical distance* and *distributionally one-way* functions. All logarithms in this paper are to the base of 2. For a finite set S, we denote by $x \in_R S$ the experiment of choosing an element of S according to the uniform distribution. Given two strings u and v, we denote by $u \circ v$ the concatenation of u and v. For random variables X, Y and Z we use the following definitions:

- The *(Shannon) entropy* of X is defined as $\mathrm{H}(X) = -\sum_x \Pr[X = x] \log \Pr[X = x]$.
- The *conditional entropy* of X given Y is defined as $\mathrm{H}(X|Y) = \sum_y \Pr[Y = y] \mathrm{H}(X|Y = y)$.
- The *mutual information* of X and Y is defined as $\mathrm{I}(X; Y) = \mathrm{H}(X) - \mathrm{H}(X|Y)$.
- The *mutual information* of X and Y given Z is defined as $\mathrm{I}(X; Y|Z) = \mathrm{H}(X|Z) - \mathrm{H}(X|Z, Y)$.

Definition 2.1. *A function $\nu : \mathbb{N} \to \mathbb{R}$ is called* negligible *if for any $c \in \mathbb{N}$ there exists an integer n_c such that $|\nu(n)| < n^{-c}$ for every $n \geq n_c$.*

Definition 2.2. *A function $f : \{0,1\}^* \to \{0,1\}^*$ is called* one-way *if it is computable in polynomial-time, and for every probabilistic polynomial-time Turing machine*[4] *$\mathcal{M}$ it holds that*

$$\Pr\left[\mathcal{M}(f(x), 1^n) \in f^{-1}(f(x))\right] < \nu(n) \enspace ,$$

for some negligible function $\nu(n)$ and for all sufficiently large n, where the probability is taken uniformly over all the possible choices of $x \in \{0,1\}^n$ and all the possible outcomes of the internal coin tosses of $\mathcal{M}$.

Definition 2.3. *The* statistical distance *between two distributions $\mathcal{D}$ and $\mathcal{F}$, which we denote by $\Delta(\mathcal{D}, \mathcal{F})$, is defined as:*

$$\Delta(\mathcal{D}, \mathcal{F}) = \frac{1}{2} \sum_{\alpha} \left|\Pr_{x \leftarrow \mathcal{D}}[x = \alpha] - \Pr_{x \leftarrow \mathcal{F}}[x = \alpha]\right| \enspace .$$

*The distributions $\mathcal{D}$ and $\mathcal{F}$ are said to be ϵ-*statistically far *if $\Delta(\mathcal{D}, \mathcal{F}) \geq \epsilon$. Otherwise, $\mathcal{D}$ and $\mathcal{F}$ are* ϵ-statistically close*.*

Definition 2.4. *A function $f : \{0,1\}^* \to \{0,1\}^*$ is called* distributionally one-way *if it is computable in polynomial-time, and there exists a constant $c > 0$ such that for every probabilistic polynomial-time Turing machine $\mathcal{M}$, the distribution defined by $x \circ f(x)$ and the distribution defined by $\mathcal{M}(f(x)) \circ f(x)$ are n^{-c}-statistically far when $x \in_{\mathrm{R}} \{0,1\}^n$.*

Informally, it is hard to find a random inverse of a distributionally one-way function, although finding some inverse may be easy. Clearly, any one-way function is also a distributionally one-way function, but the converse is not always true. Nevertheless, Impagliazzo and Luby [16] proved that the *existence* of both primitives is equivalent.

3 Communication and Adversarial Models

We consider the message authentication problem in a setting where the sender and the receiver are connected by a bidirectional insecure communication channel, over which an adversary has full control. In particular, the adversary can read any message sent over this channel, delay it, prevent it from being delivered, and insert a new message at any point in time.

[4] For simplicity we focus on uniform adversaries. However, uniformity is not essential to our results.

3.1 The Manual Channel Communication Model

In addition to the insecure channel, we assume that there is a unidirectional low-bandwidth auxiliary channel, that enables the sender to "manually" authenticate one short string to the receiver (for example, by typing a short string or by comparing two short strings). The adversary cannot modify this short string. However, the adversary can still read it, delay it and remove it.

The input of the sender $\mathcal{S}$ in this model is a message m, which she wishes to authenticate to the receiver $\mathcal{R}$. The input message m can be determined by the adversary $\mathcal{A}$. In the first round, $\mathcal{S}$ sends the message m and an authentication tag x_1 over the insecure channel. In the following rounds only a tag x_i is sent over the insecure channel[5]. The adversary receives each of these tags x_i and can replace them with $\widehat{x}_i$ of her choice, as well as replace the input message m with a different message $\widehat{m}$. In the last round, $\mathcal{S}$ may manually authenticate a short string s.

Notice that in the presence of a computationally unbounded adversary, additional insecure rounds (after the manually authenticated string has been sent) do not contribute to the security of the protocol. This is due to the fact that after reading the manually authenticated string, the unbounded adversary can always simulate the sender successfully (since the sender and the receiver do not share any secret information, and since the adversary has full control over the communication channel from this point on). Therefore, there is no loss of generality in assuming that the manually authenticated string is sent in the last round. Under the assumption that distributionally one-way functions do not exist, this is true also in the computational setting. In this case, as mentioned in Section 8, simulating the sender can be viewed as randomly inverting functions given the image of a random input. A generic protocol in this model is described in Figure 2.

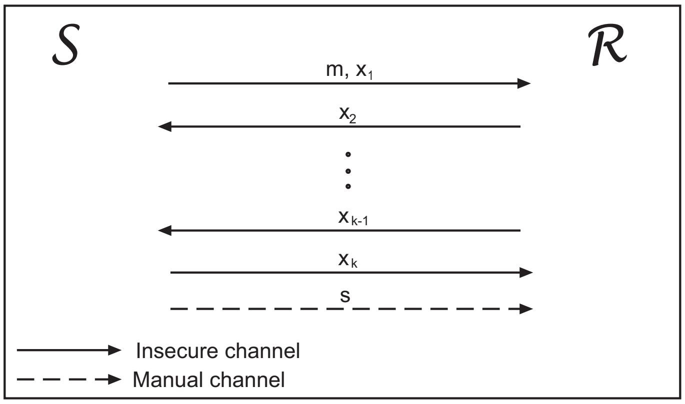

Figure 2: A generic protocol in the manual channel model.

We also allow the adversary to control the synchronization of the protocol's execution. More specifically, the adversary can carry on two separate, possibly asynchronous conversations, one with the sender and one with the receiver. However, the party that is supposed to send a message waits until it receives the adversary's message from the previous round.

Definition 3.1. *An* unconditionally secure (n, ℓ, k, ϵ)-authentication protocol *in the manual channel model is a k-round protocol in the communication model described above, in which the sender wishes to authenticate an n-bit input message to the receiver, while manually authenticating at most ℓ bits. The following requirements must hold:*

[5]In every communication round, *one* party delivers a message to the other party.

1. **Completeness:** *For all input messages m, when there is no interference by the adversary in the execution, the receiver accepts m with probability at least* $1/2$.

2. **Unforgeability:** *For any computationally unbounded adversary, and for all input messages m, if the adversary replaces m with a different message $\widehat{m}$, then the receiver accepts $\widehat{m}$ with probability at most ϵ.*

We now define the notion of a *computationally secure* protocol by actually considering a *sequence* of protocols. The sequence is parameterized by a security parameter t, that defines the power of the adversaries against which each protocol in the sequence is secure. The completeness requirement is as in Definition 3.1. However, the unforgeability requirement should now hold only against adversaries running in time poly(t), and we allow forgery probability of $\epsilon + \nu(t)$ for sufficiently large t and a negligible function $\nu(\cdot)$.

Definition 3.2. *A* computationally secure $(n, \ell, k, \epsilon, t)$-authentication protocol (sequence) *in the manual channel model is a $k(t)$-round protocol in the communication model described above, in which the sender wishes to authenticate an $n(t)$-bit input message to the receiver, while manually authenticating at most $\ell(t)$ bits. The following requirements must hold:*

1. **Completeness:** *For all input messages m, when there is no interference by the adversary in the execution, the receiver accepts m with probability at least* $1/2$.

2. **Unforgeability:** *For any adversary running in time poly(t) and for every input message m chosen by the adversary, if the adversary replaces m with a different message $\widehat{m}$, then the receiver accepts $\widehat{m}$ with probability at most $\epsilon + \nu(t)$ for some negligible function $\nu(t)$ and for all sufficiently large t.*

An authentication protocol in the manual channel model is said to be *perfectly complete* if for all input messages m, whenever there is no interference by the adversary in the execution, the receiver accepts m with probability 1.

3.2 The Shared Key Communication Model

In this model we assume that the sender and the receiver share a secret key s; however, they are connected only by an insecure channel. This key is not known to the adversary, but it is chosen from a probability distribution which is known to the adversary (usually the uniform distribution).

The input of the sender $\mathcal{S}$ in this model is a message m, which she wishes to authenticate to the receiver $\mathcal{R}$. The input message m can be determined by the adversary $\mathcal{A}$. In the first round, $\mathcal{S}$ sends the message m and an authentication tag x_1 over the insecure channel. In the following rounds only a tag x_i is sent over the insecure channel. The adversary receives each of these tags x_i and can replace them with $\widehat{x}_i$ of her choice, as well as replace the input message m with a different message $\widehat{m}$. As in the manual channel model, we allow the adversary to control the synchronization of the protocol's execution. A generic protocol in this model is described in Figure 3.

Definition 3.3. *An* unconditionally secure (n, ℓ, k, ϵ)-authentication protocol *in the shared key model is a k-round protocol in the communication model described above, in which the sender and the receiver share an ℓ-bit secret key, and the sender wishes to authenticate an n-bit input message to the receiver. The following requirements must hold:*

1. **Completeness:** *For all input messages m, when there is no interference by the adversary in the execution, the receiver accepts m with probability at least* $1/2$.

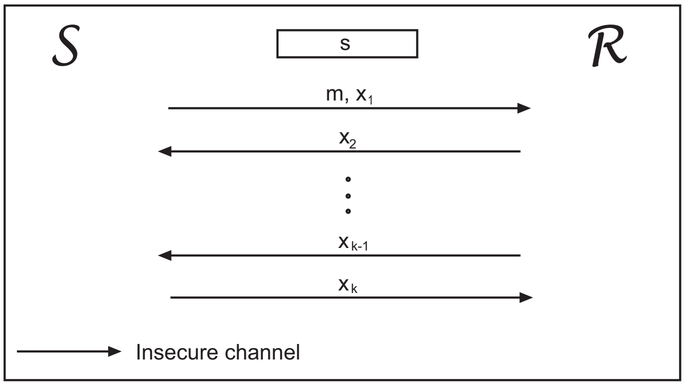

Figure 3: A generic protocol in the shared key model.

2. **Unforgeability:** *For any computationally unbounded adversary, and for all input messages m, if the adversary replaces m with a different message $\widehat{m}$, then the receiver accepts $\widehat{m}$ with probability at most ϵ.*

Similarly to Definition 3.2 we define a computationally secure sequence of authentication protocols in this model, which is parameterized by a security parameter t.

Definition 3.4. *A* computationally secure $(n, \ell, k, \epsilon, t)$-authentication protocol (sequence) *in the shared key model is a $k(t)$-round protocol in the communication model described above, in which the sender and the receiver share an $\ell(t)$-bit secret key, and the sender wishes to authenticate an $n(t)$-bit input message to the receiver. The following requirements must hold:*

1. **Completeness:** *For all input messages m, when there is no interference by the adversary in the execution, the receiver accepts m with probability at least* $1/2$.

2. **Unforgeability:** *For any adversary running in time poly(t) and for every input message m chosen by the adversary, if the adversary replaces m with a different message $\widehat{m}$, then the receiver accepts $\widehat{m}$ with probability at most $\epsilon + \nu(t)$ for some negligible function $\nu(t)$ and for all sufficiently large t.*

An authentication protocol in the shared key model is said to be *perfectly complete* if for all input messages m, whenever there is no interference by the adversary in the execution, the receiver accepts m with probability 1.

4 Overview of Our Results and Comparison with Previous Work

Vaudenay [33] formalized the manual channel model, and suggested an authentication protocol in this model. Given $0 < \epsilon < 1$, Vaudenay's protocol enables the sender to authenticate an arbitrary long message to the receiver in three rounds, by manually authenticating $\log(1/\epsilon)$ bits. This protocol guarantees that, under the assumption that a certain type of non-interactive commitment scheme exists, the forgery probability of any polynomial-time adversary is at most $\epsilon + \nu(t)$, where $\nu(\cdot)$ is a negligible function and t is a security parameter[6]. Laur and Nyberg [18] proved that the assumption required by Vaudaney's protocol is the existence of a non-interactive non-malleable commitment

[6]Manually authenticating $\log(1/\epsilon)$ bits was shown to be optimal in the computational setting by Pasini and Vaudenay [22] for any number of rounds.

scheme. Their proof shows that Vaudanay's protocol can rely also on an interactive non-malleable commitment scheme, and in this case the number of rounds in his protocol is essentially dominated by the number of rounds in the commitment scheme. Dolev, Dwork and Naor [7] showed how to construct an interactive non-malleable commitment scheme from any one-way function, and therefore we obtain the following corollary:

Corollary 4.1 ([7, 18, 33])**.** *If one-way functions exist, then there exists a computationally secure $(n, \ell, k, \epsilon, t)$-authentication protocol in the manual channel model, with $t = poly(n, \ell, k)$ and $\ell = \log(1/\epsilon)$.*

However, the non-malleable commitment scheme suggested by Dolev, Dwork and Naor is inefficient, as it utilizes generic zero-knowledge proofs and its number of rounds is logarithmic in its security parameter. Therefore, the protocol implied by Corollary 4.1 is currently not practical (this is also true if the protocols in [1, 23] are used). Currently, the only known constructions of efficient non-malleable commitment schemes are in the random oracle model, or in the common random string model (see, for example, [5, 6]). These are problematic for the manual channel model, since they require a trusted infrastructure. This state of affairs motivates the study of a protocol that can be proved secure under more relaxed computational assumptions or even without any computational assumptions.

In Section 5, we present an unconditionally secure perfectly complete authentication protocol in the manual channel model. For any odd integer $k \geq 3$, and any integer n and $0 < \epsilon < 1$, our k-round protocol enables the sender to authenticate an n-bit input message to the receiver, while manually authenticating at most $2\log(1/\epsilon) + 2\log^{(k-1)} n + O(1)$ bits. We prove that any adversary (even computationally unbounded) has probability of at most ϵ to cheat the receiver into accepting a fraudulent message. We note that our protocol only uses evaluations of polynomials over finite fields, for which very efficient implementations exist, and therefore it is very efficient and can be implemented on low-power devices. We prove the following theorem and corollary:

Theorem 4.2. *For any odd integer $k \geq 3$, and any integer n and $0 < \epsilon < 1$, there exists an unconditionally secure perfectly complete $(n, \ell = 2\log(1/\epsilon) + 2\log^{(k-1)} n + O(1), k, \epsilon)$-authentication protocol in the manual channel model.*

Corollary 4.3. *For any integer n and $0 < \epsilon < 1$, the following unconditionally secure perfectly complete protocols exist in the manual channel model:*

1. *A $\log^* n$-round protocol in which at most $2\log(1/\epsilon) + O(1)$ bits are manually authenticated.*
2. *A 3-round protocol in which at most $2\log(1/\epsilon) + \log\log n + O(1)$ bits are manually authenticated.*

Our protocol exhibits a trade-off between its number of communication rounds over the insecure channel and the length of the manually authenticated string. Informally speaking, the protocol is based on the hashing technique presented in [11], in which the two parties reduce in each round the problem of authenticating the original message to that of authenticating a shorter message. In each round of our protocol the parties reduce the length of the message roughly by a logarithmic scale (this explains the additive $\log^{(k-1)} n$ factor in the length of the manually authenticated string), and when the message is sufficiently short (but never shorter than $2\log(1/\epsilon)$ bits), it is manually authenticated.

In Section 6, we develop a proof technique for deriving lower bounds on unconditionally secure authentication protocols, which allows us to show that our $\log^* n$-round protocol is optimal with respect to the length of the manually authenticated string. Specifically, we prove the following theorem:

Theorem 4.4. *For any unconditionally secure (n, ℓ, k, ϵ)-authentication protocol in the manual channel model, it holds that if $n \geq 2\log(1/\epsilon) + 4$, then $\ell > 2\log(1/\epsilon) - 6$.*

We remark that if $\log(1/\epsilon) \leq n \leq 2\log(1/\epsilon)$, then the lower bound of $\ell \geq \log(1/\epsilon)$ (see [22]) implies that at least $n/2$ bits have to be manually authenticated. Also, in the case $n < \log(1/\epsilon)$ the lower bound is n, which is also an upper bound since the sender can manually authenticate the input message.

In Section 7 we consider the shared key communication model. Intensive research has been devoted to proving lower bounds on the required entropy of the shared key in unconditionally secure protocols. It was proved in several papers (see, for example, [19]), that in any perfectly complete non-interactive protocol, the required entropy of the shared key is at least $2\log(1/\epsilon)$. In addition, for such protocols, Gemmell and Naor [11] proved a lower bound of $\log n + \log(1/\epsilon) - \log\log(n/\epsilon) - 2$. Thus, there does not exist a perfectly complete non-interactive protocol that achieves the $2\log(1/\epsilon)$ bound. However, they also presented an *interactive* protocol that achieves the $2\log(1/\epsilon)$ bound. We remark that it was not previously known whether this bound is optimal, that is, if by introducing interaction the entropy of the shared key can be made smaller than $2\log(1/\epsilon)$. By applying the proof technique described in Section 6, we settle this long-standing open question by proving the following theorem:

Theorem 4.5. *For any unconditionally secure (n, ℓ, k, ϵ)-authentication protocol in the shared key model, it holds that $\mathrm{H}(S) \geq 2\log(1/\epsilon) - 2$, where S is the ℓ-bit shared key.*

Theorems 4.4 and 4.5 indicate that the two corresponding communication models are not equivalent: While in the manual channel model a lower bound can hold only when $n \geq \log(1/\epsilon)$, in the shared key model the lower bound holds even when authenticating only one bit. Nevertheless, the *technique* we develop applies to both models.

The idea underlying the lower bound proofs for the communication models under consideration can be briefly summarized as follows. The lower bound proofs involve a careful accounting of how randomness is introduced into the protocol. First, we represent the entropies of the manually authenticated string and of the shared key by splitting them in a way that captures their reduction during the execution of the protocol. This representation allows us to prove that both the sender and the receiver must each independently reduce the entropies by at least $\log(1/\epsilon)$ bits. This is proved by considering two possible natural attacks on the given protocol. In these attacks we use the fact that the adversary is computationally unbounded in that she can sample distributions induced by the protocol. This usage of the adversary's capabilities, can alternatively be seen as randomly inverting functions given the image of a random input.

In Section 8, we take advantage of this point of view and prove that one-way functions are necessary for the existence of protocols breaking the above lower bounds in the computational setting.[7] Specifically, we show that if distributionally one-way functions do not exist, then a polynomial-time adversary can run the above mentioned attacks with almost the same success probability:

Theorem 4.6. *If there exists a computationally secure $(n, \ell, k, \epsilon, t)$-authentication protocol in the manual channel model, such that $n \geq 2\log(1/\epsilon) + 4$, $t = \Omega(poly(n, k, 1/\epsilon))$ and $\ell < 2\log(1/\epsilon) - 8$, then one-way functions exist.*

Theorem 4.7. *If there exists a computationally secure $(n, \ell, k, \epsilon, t)$-authentication protocol in the shared key model, such that $n \geq 2\log(1/\epsilon) + 4$, $t = \Omega(poly(n, k, 1/\epsilon))$ and $\mathrm{H}(S) < 2\log(1/\epsilon) - 6$, where S is the ℓ-bit shared key, then one-way functions exist.*

[7] We note that the existence of an information-theoretic lower bound does not directly imply that breaking this lower bound in the computational setting implies the existence of one-way functions.

A similar flavor of statement has recently been proved by Naor and Rothblum [21] in the context of memory checking, showing that one-way functions are necessary for efficient on-line memory checking. Both results are based on combinatorial constructions (in our case these are the two attacks carried by an unbounded adversary), which are shown to be polynomial-time computable if one-way functions do not exist. However, we note that whereas Naor and Rothblum obtained asymptotic results (there is a multiplicative constant between the upper bound and the lower bound), we detect a sharp threshold.

5 The Message Authentication Protocol

In this section we prove Theorem 4.2 and Corollary 4.3 by constructing an authentication protocol, $\mathrm{P}_{n,k,\epsilon}$. The protocol is based on the hashing technique of Gemmell and Naor [11], in which the two parties reduce in each round the problem of authenticating the original message to that of authenticating a shorter message. In the first round the input message is sent, and then in each round the two parties *cooperatively* choose a hash function that defines a small, random "fingerprint" of the input message that the receiver should have received. If the adversary has changed the input message, then with high probability the fingerprint for the message received by the receiver will not match the fingerprint for the message that was sent by the sender. In the preliminary version of [11] (Crypto 1993), this hashing technique was susceptible to synchronization attacks, as noted by Gehrmann [8]. However, in the later (full) version of their paper, this was corrected by making *both* parties choose the random hash function used for fingerprinting the message.

We improve the hashing technique presented in [11] as follows. First, we apply a different hash function, which enables us to manually authenticate a shorter string. Any adaption of the original hash function to the manual channel model would require the sender to manually authenticate at least $3\log(1/\epsilon)$ bits, while our construction manages to reduce this amount to only $2\log(1/\epsilon)+O(1)$ bits. In addition, our protocol is asymmetric in the following sense: The roles of the sender and the receiver in cooperatively choosing the hash function are switched in every round. This enables us to optimize the number of communication rounds in the protocol.

In addition, we wish to emphasize that although the hash functions we apply in each round in order to reduce the length of the authenticated message can be viewed as variants of Reed-Solomon codes, the security of our protocol does not rely only on the properties of such codes. The security relies in a crucial manner on the particular synchronization that the protocol imposes on the cooperative choice of the hash function used in each round.

Preliminaries. Denote by $\mathrm{GF}[Q]$ the Galois field with Q elements. For a message $m = m_1 \ldots m_k \in \mathrm{GF}[Q]^k$ and $x \in \mathrm{GF}[Q]$ let $C_x(m) = \sum_{i=1}^{k} m_i x^i$. In other words, m is parsed as a polynomial of degree k over $\mathrm{GF}[Q]$ (without a constant term), and evaluated at the point x. Then, for any two different messages $m, \widehat{m} \in \mathrm{GF}[Q]^k$ and for any $c, \widehat{c} \in \mathrm{GF}[Q]$ the polynomials $C_x(m)+c$ and $C_x(\widehat{m})+\widehat{c}$ are different as well, and therefore $\Pr_{x \in_{\mathrm{R}} \mathrm{GF}[Q]}\left[C_x(m) + c = C_x(\widehat{m}) + \widehat{c}\right] \leq \frac{k}{Q}$. The functions we apply in order to reduce the length of the message m in each round are of the form $C_x(\cdot) + c$, where one party chooses a random $x \in \mathrm{GF}[Q]$ and the other party chooses a random $c \in \mathrm{GF}[Q]$.

The construction. Protocol $\mathrm{P}_{n,k,\epsilon}$ applies a sequence of hash functions $C^1, \ldots, C^{k-1}$ in order to obtain a shorter and shorter message. Specifically, given the protocol's parameters: n - the length of the input message, k - the number of rounds, and ϵ - the upper bound on the adversary's forgery probability, each C^j parses n_j-bit strings to polynomials over $\mathrm{GF}[Q_j]$, where $n_1 = n$, $\frac{2^{k-j} n_j}{\epsilon} \leq Q_j <$

$\frac{2^{k-j+1} n_j}{\epsilon}$, and $n_{j+1} = \lceil 2 \log Q_j \rceil$. The protocol's parameters n, k and ϵ are fixed and are known to both parties. The protocol is described in Figure 4. Since the adversary can replace any authentication tag sent by any one of the parties over the insecure channel, then for such a tag x we denote by $\widehat{x}$ the tag that was actually received by the other party. Note that addition and multiplication are defined by the $\mathrm{GF}[Q_j]$ structures, and that $\langle u, v \rangle$ denotes the concatenation of the strings u and v.

Protocol $\mathbf{P}_{n,k,\epsilon}$:

1. $\mathcal{S}$ sends $m_{\mathcal{S}}^1 = m$ to $\mathcal{R}$.
2. $\mathcal{R}$ receives $m_{\mathcal{R}}^1$.
3. For $j = 1$ to $k - 1$:
   (a) If j is odd, then
      i. $\mathcal{S}$ chooses $i_{\mathcal{S}}^j \in_{\mathrm{R}} \mathrm{GF}[Q_j]$ and sends it to $\mathcal{R}$.
      ii. $\mathcal{R}$ receives $\widehat{i}_{\mathcal{S}}^j$, chooses $i_{\mathcal{R}}^j \in_{\mathrm{R}} \mathrm{GF}[Q_j]$, and sends it to $\mathcal{S}$.
      iii. $\mathcal{S}$ receives $\widehat{i}_{\mathcal{R}}^j$, and computes $m_{\mathcal{S}}^{j+1} = \langle \widehat{i}_{\mathcal{R}}^j, C_{\widehat{i}_{\mathcal{R}}^j}^j(m_{\mathcal{S}}^j) + i_{\mathcal{S}}^j \rangle$.
      iv. $\mathcal{R}$ computes $m_{\mathcal{R}}^{j+1} = \langle i_{\mathcal{R}}^j, C_{i_{\mathcal{R}}^j}^j(m_{\mathcal{R}}^j) + \widehat{i}_{\mathcal{S}}^j \rangle$.
   (b) If j is even, then
      i. $\mathcal{R}$ chooses $i_{\mathcal{R}}^j \in_{\mathrm{R}} \mathrm{GF}[Q_j]$ and sends it to $\mathcal{S}$.
      ii. $\mathcal{S}$ receives $\widehat{i}_{\mathcal{R}}^j$, chooses $i_{\mathcal{S}}^j \in_{\mathrm{R}} \mathrm{GF}[Q_j]$, and sends it to $\mathcal{R}$.
      iii. $\mathcal{R}$ receives $\widehat{i}_{\mathcal{S}}^j$, and computes $m_{\mathcal{R}}^{j+1} = \langle \widehat{i}_{\mathcal{S}}^j, C_{\widehat{i}_{\mathcal{S}}^j}^j(m_{\mathcal{R}}^j) + i_{\mathcal{R}}^j \rangle$.
      iv. $\mathcal{S}$ computes $m_{\mathcal{S}}^{j+1} = \langle i_{\mathcal{S}}^j, C_{i_{\mathcal{S}}^j}^j(m_{\mathcal{S}}^j) + \widehat{i}_{\mathcal{R}}^j \rangle$.
4. $\mathcal{S}$ manually authenticates $m_{\mathcal{S}}^k$ to $\mathcal{R}$.
5. $\mathcal{R}$ accepts if and only if $m_{\mathcal{S}}^k = m_{\mathcal{R}}^k$.

Figure 4: The k-round authentication protocol.

Note that the two parties can combine some of their messages, and therefore the protocol requires only $k-1$ rounds of communication over the insecure channel, and then a single manually authenticated string. An alternative way to describe the protocol is in a recursive fashion. The k-round protocol consists of $\mathcal{S}$ sending the message $m_1 = m$, as well as $\mathcal{S}$ and $\mathcal{R}$ exchanging $i_{\mathcal{S}}^1$, and $i_{\mathcal{R}}^1$. Then the two parties use the $(k-1)$-round protocol to authenticate the message m_2, which is a computed hash value of m_1 using $i_{\mathcal{S}}^1$, and $i_{\mathcal{R}}^1$. Clearly, this protocol is perfectly complete.

An important property in this setting is that the parties will each be able to choose the Q_j's and the representations of $\mathrm{GF}[Q_j]$ in a deterministic way[8]. Otherwise, they will have to agree on these parameters with some security consequences (see, for example, Vaudenay's attack on the domain parameter generation algorithm of ECDSA [32]). This is required, since the sender will manually authenticate a *string of bits* and not an actual *field element*, and the same string represents different field elements under different representations of the field. One possible solution is for the two parties to choose Q_j as the smallest prime number in the above mentioned interval. Another solution would be to choose Q_j as the unique power of 2 in this interval, then apply a deterministic algorithm to compute an irreducible polynomial of degree $\log Q_j$ in $\mathrm{GF}_2[x]$ (e.g., Shoup's algorithm [28]), and

[8]Alternatively, these parameters can be explicitly specified in the description of the protocol, together with the parameters n, k and ϵ which are known to the parties.

represent the field in the standard way.

In the remainder of this section we prove the security of our protocol. For any odd integer $k \geq 3$, and any integer n and $0 < \epsilon < 1$, we argue that protocol $\mathrm{P}_{n,k,\epsilon}$ satisfies Definition 3.1 of an unconditionally secure (n, ℓ, k, ϵ)-authentication protocol in the manual channel model for $\ell = 2\log(1/\epsilon) + 2\log^{(k-1)} n + O(1)$. We fix the parameters k, n and ϵ for the rest of the section.

Lemma 5.1. *Any computationally unbounded adversary has probability of at most ϵ to cheat the receiver into accepting a fraudulent message in protocol $\mathrm{P}_{n,k,\epsilon}$.*

Proof. Given an execution of the protocol in which an adversary cheats the receiver into accepting a fraudulent message, it holds that $m^1_{\mathcal{S}} \neq m^1_{\mathcal{R}}$ and $m^k_{\mathcal{S}} = m^k_{\mathcal{R}}$. Therefore, there exists an integer $1 \leq j \leq k-1$ such that $m^j_{\mathcal{S}} \neq m^j_{\mathcal{R}}$ and $m^{j+1}_{\mathcal{S}} = m^{j+1}_{\mathcal{R}}$. Denote this event by D_j. In what follows, we bound the probability of this event, showing $\Pr[D_j] \leq \frac{\epsilon}{2^{k-j}}$. Therefore, the adversary's cheating probability is at most $\sum_{j=1}^{k-1} \Pr[D_j] \leq \sum_{j=1}^{k-1} \frac{\epsilon}{2^{k-j}} < \epsilon$.

For any variable y in the protocol and for a given execution, let $T(y)$ be the time at which the variable y is fixed, i.e., $T(i^j_{\mathcal{R}})$ denotes the time in which $\mathcal{R}$ sent the tag $i^j_{\mathcal{R}}$, and $T(\widehat{i^j_{\mathcal{R}}})$ denotes the time in which $\mathcal{S}$ received from the adversary the tag $\widehat{i^j_{\mathcal{R}}}$ corresponding to $i^j_{\mathcal{R}}$.

We first assume that j is odd, then we have to consider three possible cases:

1. $\boldsymbol{T(\widehat{i^j_{\mathcal{R}}}) < T(i^j_{\mathcal{R}})}$. In this case, the receiver chooses $i^j_{\mathcal{R}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]$ only after the adversary chooses $\widehat{i^j_{\mathcal{R}}}$. Therefore,
$$\Pr[D_j] \leq \Pr_{i^j_{\mathcal{R}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]}\left[i^j_{\mathcal{R}} = \widehat{i^j_{\mathcal{R}}}\right] = \frac{1}{Q_j} \leq \frac{\epsilon}{2^{k-j}} \enspace .$$

2. $\boldsymbol{T(\widehat{i^j_{\mathcal{R}}}) \geq T(i^j_{\mathcal{R}})}$ **and** $\boldsymbol{T(\widehat{i^j_{\mathcal{S}}}) \geq T(i^j_{\mathcal{S}})}$. In this case, the adversary chooses $\widehat{i^j_{\mathcal{R}}}$ not before the receiver chooses $i^j_{\mathcal{R}}$. If the adversary chooses $\widehat{i^j_{\mathcal{R}}} \neq i^j_{\mathcal{R}}$, then $m^{j+1}_{\mathcal{S}} \neq m^{j+1}_{\mathcal{R}}$, i.e., $\Pr[D_j] = 0$. Now suppose that the adversary chooses $\widehat{i^j_{\mathcal{R}}} = i^j_{\mathcal{R}}$. Since j is odd, the receiver chooses $i^j_{\mathcal{R}}$ only after he receives $\widehat{i^j_{\mathcal{S}}}$, therefore $T(i^j_{\mathcal{R}}) > T(\widehat{i^j_{\mathcal{S}}}) \geq T(i^j_{\mathcal{S}}) > T(m^j_{\mathcal{S}})$, and also $T(i^j_{\mathcal{R}}) > T(m^j_{\mathcal{R}})$. This means that $i^j_{\mathcal{R}}$ is chosen when $m^j_{\mathcal{R}}, \widehat{i^j_{\mathcal{S}}}, m^j_{\mathcal{S}}$ and $i^j_{\mathcal{S}}$ are fixed. Since $m^j_{\mathcal{S}} \neq m^j_{\mathcal{R}}$ and by the fact that for any choice of $i^j_{\mathcal{S}}$ and $\widehat{i^j_{\mathcal{S}}}$ the polynomials $C^j_{i^j_{\mathcal{R}}}(m^j_{\mathcal{R}}) + \widehat{i^j_{\mathcal{S}}}$ and $C^j_{i^j_{\mathcal{R}}}(m^j_{\mathcal{S}}) + i^j_{\mathcal{S}}$ are different as functions of $i^j_{\mathcal{R}}$, it follows that
$$\Pr[D_j] \leq \Pr_{i^j_{\mathcal{R}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]}\left[C^j_{i^j_{\mathcal{R}}}(m^j_{\mathcal{S}}) + i^j_{\mathcal{S}} = C^j_{i^j_{\mathcal{R}}}(m^j_{\mathcal{R}}) + \widehat{i^j_{\mathcal{S}}}\right] \leq \frac{1}{Q_j}\left\lceil \frac{n_j}{\log Q_j} \right\rceil \leq \frac{\epsilon}{2^{k-j}} \enspace .$$

3. $\boldsymbol{T(\widehat{i^j_{\mathcal{R}}}) \geq T(i^j_{\mathcal{R}})}$ **and** $\boldsymbol{T(\widehat{i^j_{\mathcal{S}}}) < T(i^j_{\mathcal{S}})}$. As in the previous case, we can assume that the adversary chooses $\widehat{i^j_{\mathcal{R}}} = i^j_{\mathcal{R}}$. It always holds that $T(i^j_{\mathcal{S}}) > T(m^j_{\mathcal{S}})$ and $T(i^j_{\mathcal{R}}) > T(m^j_{\mathcal{R}})$. Since j is odd, the receiver sends $i^j_{\mathcal{R}}$ only after he receives $\widehat{i^j_{\mathcal{S}}}$, and therefore we can assume $T(\widehat{i^j_{\mathcal{S}}}) < T(i^j_{\mathcal{R}}) < T(i^j_{\mathcal{S}})$. This implies that the sender chooses $i^j_{\mathcal{S}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]$ when $m^j_{\mathcal{S}}$, $\widehat{i^j_{\mathcal{S}}}$, $m^j_{\mathcal{R}}$ and $i^j_{\mathcal{R}}$ are fixed. Hence,
$$\Pr[D_j] \leq \Pr_{i^j_{\mathcal{S}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]}\left[i^j_{\mathcal{S}} = C^j_{i^j_{\mathcal{R}}}(m^j_{\mathcal{R}}) + \widehat{i^j_{\mathcal{S}}} - C^j_{i^j_{\mathcal{R}}}(m^j_{\mathcal{S}})\right] = \frac{1}{Q_j} \leq \frac{\epsilon}{2^{k-j}} \enspace .$$

We now assume that j is even, then again we have to consider three possible cases:

1. $\boldsymbol{T(\widehat{i}^j_{\mathcal{S}}) < T(i^j_{\mathcal{S}})}$. In this case, the sender chooses $i^j_{\mathcal{S}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]$ only after the adversary chooses $\widehat{i}^j_{\mathcal{S}}$. Therefore,
$$\Pr\left[D_j\right] \leq \Pr_{i^j_{\mathcal{S}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]}\left[i^j_{\mathcal{S}} = \widehat{i}^j_{\mathcal{S}}\right] = \frac{1}{Q_j} \leq \frac{\epsilon}{2^{k-j}} \enspace .$$

2. $\boldsymbol{T(\widehat{i}^j_{\mathcal{S}}) \geq T(i^j_{\mathcal{S}})}$ **and** $\boldsymbol{T(\widehat{i}^j_{\mathcal{R}}) \geq T(i^j_{\mathcal{R}})}$. In this case, the adversary chooses $\widehat{i}^j_{\mathcal{S}}$ not before the sender chooses $i^j_{\mathcal{S}}$. If the adversary chooses $\widehat{i}^j_{\mathcal{S}} \neq i^j_{\mathcal{S}}$, then $m^{j+1}_{\mathcal{S}} \neq m^{j+1}_{\mathcal{R}}$, i.e., $\Pr\left[D_j\right] = 0$. Now suppose that the adversary chooses $\widehat{i}^j_{\mathcal{S}} = i^j_{\mathcal{S}}$. Since j is even, it holds that $T(i^j_{\mathcal{S}}) > T(\widehat{i}^j_{\mathcal{R}})$, and therefore also $T(i^j_{\mathcal{S}}) > T(i^j_{\mathcal{R}}) > T(m^j_{\mathcal{R}})$. This means that $i^j_{\mathcal{S}}$ is chosen when $m^j_{\mathcal{S}}, \widehat{i}^j_{\mathcal{R}}, m^j_{\mathcal{R}}$ and $i^j_{\mathcal{R}}$ are fixed. Since $m^j_{\mathcal{S}} \neq m^j_{\mathcal{R}}$ and by the fact that for any choice of $i^j_{\mathcal{R}}$ and $\widehat{i}^j_{\mathcal{R}}$ the polynomials $C^j_{i^j_{\mathcal{S}}}(m^j_{\mathcal{S}}) + \widehat{i}^j_{\mathcal{R}}$ and $C^j_{i^j_{\mathcal{S}}}(m^j_{\mathcal{R}}) + i^j_{\mathcal{R}}$ are different as functions of $i^j_{\mathcal{S}}$, it follows that
$$\Pr\left[D_j\right] \leq \Pr_{i^j_{\mathcal{S}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]}\left[C^j_{i^j_{\mathcal{S}}}(m^j_{\mathcal{R}}) + i^j_{\mathcal{R}} = C^j_{i^j_{\mathcal{S}}}(m^j_{\mathcal{S}}) + \widehat{i}^j_{\mathcal{R}}\right] \leq \frac{\epsilon}{2^{k-j}} \enspace .$$

3. $\boldsymbol{T(\widehat{i}^j_{\mathcal{S}}) \geq T(i^j_{\mathcal{S}})}$ **and** $\boldsymbol{T(\widehat{i}^j_{\mathcal{R}}) < T(i^j_{\mathcal{R}})}$. As in the previous case, we can assume that the adversary chooses $\widehat{i}^j_{\mathcal{S}} = i^j_{\mathcal{S}}$. Since j is even, the sender chooses $i^j_{\mathcal{S}}$ after she receives $\widehat{i}^j_{\mathcal{R}}$, and therefore we can assume that $T(i^j_{\mathcal{R}}) > T(i^j_{\mathcal{S}})$. It always holds that $T(i^j_{\mathcal{R}}) > T(m^j_{\mathcal{R}})$, and $T(i^j_{\mathcal{S}}) > T(m^j_{\mathcal{S}})$. This implies that the receiver chooses $i^j_{\mathcal{R}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]$ when $m^j_{\mathcal{R}}$, $\widehat{i}^j_{\mathcal{R}}$, $m^j_{\mathcal{S}}$ and $i^j_{\mathcal{S}}$ are fixed. Hence,
$$\Pr\left[D_j\right] \leq \Pr_{i^j_{\mathcal{R}} \in_{\mathrm{R}} \mathrm{GF}[Q_j]}\left[i^j_{\mathcal{R}} = C^j_{i^j_{\mathcal{S}}}(m^j_{\mathcal{S}}) + \widehat{i}^j_{\mathcal{R}} - C^j_{i^j_{\mathcal{S}}}(m^j_{\mathcal{R}})\right] = \frac{1}{Q_j} \leq \frac{\epsilon}{2^{k-j}} \enspace .$$

■

The following claims conclude this section by showing that our choice of parameters guarantees that in protocol $\mathrm{P}_{n,k,\epsilon}$ the sender manually authenticates at most $2\log(1/\epsilon) + 2\log^{(k-1)} n + O(1)$ bits. We first show that the length n_{j+1} of the fingerprint computed in round j is roughly logarithmic in the length n_j of the fingerprint computed in round $j-1$, and then we use this fact to upper bound the length n_k of the manually authenticated fingerprint.

Claim 5.2. *If $n_j > \frac{2^{k-j}}{\epsilon}$ for every $1 \leq j \leq k-2$, then $n_{k-1} \leq \max\{4\log^{(k-2)} n_1 + 4\log 5 + 3,\ 27\}$.*

Proof. Fix k, n and ϵ and assume that $n_r > \frac{2^{k-r}}{\epsilon}$ for every $1 \leq r \leq k-2$. We prove by induction on j that $n_{j+1} \leq \max\{4\log^{(j)} n_1 + 4\log 5 + 3,\ 27\}$ for every $1 \leq j \leq k-2$. The claim then follows by setting $j = k-2$.

First note that by the choice of Q_j, if $n_j > \frac{2^{k-j}}{\epsilon}$ then
$$\begin{aligned} Q_j &< \frac{2^{k-j+1} n_j}{\epsilon} \\ &= 2 \cdot \frac{2^{k-j}}{\epsilon} \cdot n_j \\ &< 2 \cdot n_j \cdot n_j \\ &= 2n_j^2 \enspace , \end{aligned}$$

and therefore

$$n_{j+1} = \lceil 2\log Q_j \rceil \leq 4\log n_j + 3 \ .$$

Now, for $j = 1$ we have $n_2 \leq 4\log n_1 + 3$, and the claim holds. Suppose now that the claim holds for $j-1$ and we show that it holds for j as well. That is, we assume that $n_j \leq \max\{4\log^{(j-1)} n_1 + 4\log 5 + 3,\ 27\}$, and we show that $n_{j+1} \leq \max\{4\log^{(j)} n_1 + 4\log 5 + 3,\ 27\}$. If $n_j \leq 27$, then the sender can manually authenticate $m_{\mathcal{S}}^j$ and stop the protocol. Else, if $n_j \leq 4\log^{(j-1)} n_1 + 4\log 5 + 3$, then

$$n_{j+1} \leq 4\log n_j + 3 \leq 4\log\left(4\log^{(j-1)} n_1 + 4\log 5 + 3\right) + 3 \ ,$$

and there are two cases to consider:

1. If $\log^{(j-1)} n_1 \leq 4\log 5 + 3$, then $n_{j+1} \leq 4\log(20\log 5 + 15) + 3 < 27$.

2. If $\log^{(j-1)} n_1 > 4\log 5 + 3$, then $n_{j+1} < 4\log(5\log^{(j-1)} n_1) + 3 = 4\log^{(j)} n_1 + 4\log 5 + 3$.

Therefore $n_{j+1} \leq \max\{4\log^{(j)} n_1 + 4\log 5 + 3,\ 27\}$. ■

Claim 5.3. *In protocol* $\mathrm{P}_{n,k,\epsilon}$ *the sender manually authenticates at most* $2\log(1/\epsilon) + 2\log^{(k-1)} n + O(1)$ *bits.*

Proof. We consider two possible cases:

1. If $n_j > \frac{2^{k-j}}{\epsilon}$ for every $1 \leq j \leq k-2$, then Claim 5.2 implies that $n_{k-1} \leq \max\{4\log^{(k-2)} n_1 + 4\log 5 + 3,\ 27\}$. Therefore $n_k = \lceil 2\log Q_{k-1} \rceil \leq \lceil 2\log \frac{4n_{k-1}}{\epsilon} \rceil \leq 2\log(1/\epsilon) + 2\log^{(k-1)} n + O(1)$.

2. If there exists some $1 \leq j \leq k-2$ such that $n_j \leq \frac{2^{k-j}}{\epsilon}$, then $n_{k-2} \leq \frac{4}{\epsilon}$, and therefore

$$\begin{aligned} Q_{k-2} &\leq \frac{2^{k-(k-2)+1} n_{k-2}}{\epsilon} \leq \frac{32}{\epsilon^2} \ , \\ n_{k-1} &= \lceil 2\log Q_{k-2} \rceil \leq 4\log\frac{1}{\epsilon} + 11 \ . \end{aligned}$$

   In this case it is sufficient to choose $Q_{k-1} = \Theta(1/\epsilon)$ (instead of $Q_{k-1} = \Theta\left(\frac{1}{\epsilon}\log\frac{1}{\epsilon}\right)$), which implies that $n_k = 2\log(1/\epsilon) + O(1)$.

■

6 Lower Bound in the Manual Channel Model

In this section we prove a lower bound on the length of the manually authenticated string. The same proof technique will be used again in Section 7 to prove a similar lower bound in the shared key model. For simplicity, we first present in Section 6.1 a proof for the simplified case of a perfectly complete 3-round protocol where $n \geq 3\log(1/\epsilon)$. This simplified proof already captures the main ideas and difficulties of the general proof. The general proof is based on the same analysis, and is described in Section 6.2.

In the following proofs, given any authentication protocol in the manual channel model, we identify the messages sent by the sender and the receiver with corresponding random variables. More specifically, when the input message m is chosen uniformly at random, the honest execution of the protocol defines a probability distribution on the message m, the tags x_i and the manually authenticated string s. We denote by M, X_i and S the random variables corresponding to m, x_i and s, respectively. We note that our lower bound holds even for protocols which are secure only when authenticating a random message m. This should not be confused with the security properties of our protocol in Section 5 which is secure even if the adversary is allowed to pick any message m (which corresponds to the security model described in Section 3.1).

6.1 A Simplified Proof

We prove the following theorem:

Theorem 6.1. *For any perfectly complete $(n, \ell, 3, \epsilon)$-authentication protocol in the manual channel model, where no authentication tag is sent in the last round, if $n \geq 3\log(1/\epsilon)$, then $\ell \geq 2\log(1/\epsilon) - 2$.*

As mentioned above, when the input message m is chosen uniformly at random, the honest execution of the protocol defines a probability distribution on the message m, the authentication tag x_1 (sent by the sender in the first round together with m), the authentication tag x_2 (sent by the receiver in the second round), and the manually authenticated string s (sent by the sender in the third round). We denote by M, X_1, X_2 and S the corresponding random variables.

The main idea of this proof is representing the entropy of the manually authenticated string S by splitting it as follows:

$$\begin{aligned} \mathrm{H}(S) &= (\mathrm{H}(S) - \mathrm{H}(S|M, X_1)) + (\mathrm{H}(S|M, X_1) - \mathrm{H}(S|M, X_1, X_2)) + \mathrm{H}(S|M, X_1, X_2) \\ &= \mathrm{I}(S; M, X_1) + \mathrm{I}(S; X_2|M, X_1) + \mathrm{H}(S|M, X_1, X_2) \ . \end{aligned}$$

This representation captures the reduction of $\mathrm{H}(S)$ during the execution of the protocol, and allows us to prove that both the sender and the receiver must each independently reduce this entropy by at least $\log(1/\epsilon) - 1$ bits. We prove this by considering two possible man-in-the-middle attacks on the given protocol. In these attacks we use the fact that the adversary is computationally unbounded in that she can sample distributions induced by the protocol. For example, in the first attack, the adversary samples the distribution of X_2 given M, X_1 and S. While the distribution of X_2 given only M and X_1 can be sampled by merely following the protocol, this is not the case when sampling the distribution of X_2 given M, X_1 and S.

We first state and prove two lemmata on the reduction in $\mathrm{H}(S)$ during the execution of the protocol, and then show that these lemmata and the above representation of $\mathrm{H}(S)$ imply Theorem 6.1.

Lemma 6.2. *If $n \geq 2\log\frac{1}{\epsilon}$, then $\mathrm{I}(S; M, X_1) + \mathrm{H}(S|M, X_1, X_2) > \log\frac{1}{\epsilon} - 1$.*

Proof. Consider the following attack:

1. The adversary $\mathcal{A}$ chooses $\widehat{m} \in_{\mathrm{R}} \{0,1\}^n$ and runs an honest execution with the receiver. Denote by s the manually authenticated string fixed by this execution. Now, $\mathcal{A}$'s goal is to cause the sender to manually authenticate this string.

2. $\mathcal{A}$ chooses $m \in_{\mathrm{R}} \{0,1\}^n$ as the sender's input, and receives x_1 from the sender.

3. If $\Pr[m, x_1, s] = 0$ in an honest execution, then $\mathcal{A}$ quits (in this case $\mathcal{A}$ has zero probability in convincing the sender to manually authenticate s). Otherwise, $\mathcal{A}$ samples $\widehat{x}_2$ from the distribution of X_2 given (m, x_1, s), and sends it to the sender. The sender manually authenticates some string.

4. If the sender did not authenticate s, $\mathcal{A}$ quits. Otherwise, $\mathcal{A}$ forwards s to the receiver.

By the unforgeability requirement of the protocol we obtain:

$$\epsilon \geq \Pr[\mathcal{R} \text{ accepts and } m \neq \widehat{m}] \geq \Pr[\mathcal{R} \text{ accepts}] - 2^{-n} \ .$$

Therefore, the assumption $n \geq 2\log\frac{1}{\epsilon}$ implies that $\Pr[\mathcal{R} \text{ accepts}] \leq \epsilon + 2^{-n} \leq \epsilon + \epsilon^2 < 2\epsilon$. Now we analyze the probability that the receiver accepts. Notice that:

- m and x_1 are chosen independently of s.
- $\widehat{x}_2$ is chosen conditioned on m, x_1 and s.
- The manually authenticated string sent by the sender is chosen conditioned on m, x_1 and x_2.

Therefore[9],

$$\begin{aligned}
\Pr[\mathcal{R} \text{ accepts}] &= \sum_{s} \Pr[s] \Pr[\mathcal{R} \text{ accepts}|s] \\
&= \sum_{s} \Pr[s] \sum_{\substack{m,x_1: \\ \Pr[m,x_1,s]>0}} \Pr[m, x_1] \Pr[\mathcal{R} \text{ accepts}|m, x_1, s] \\
&= \sum_{\substack{m,x_1,s: \\ \Pr[m,x_1,s]>0}} \Pr[s] \Pr[m, x_1] \sum_{\widehat{x}_2} \Pr[\widehat{x}_2|m, x_1, s] \Pr[\mathcal{R} \text{ accepts}|m, x_1, \widehat{x}_2, s] \\
&= \sum_{\substack{m,x_1,\widehat{x}_2,s: \\ \Pr[m,x_1,\widehat{x}_2,s]>0}} \Pr[s] \Pr[m, x_1] \Pr[\widehat{x}_2|m, x_1, s] \Pr[s|m, x_1, \widehat{x}_2] \\
&= \sum_{\substack{m,x_1,\widehat{x}_2,s: \\ \Pr[m,x_1,\widehat{x}_2,s]>0}} \Pr[m, x_1, s] \frac{\Pr[s]}{\Pr[s|m, x_1]} \Pr[\widehat{x}_2|m, x_1, s] \Pr[s|m, x_1, \widehat{x}_2] \qquad (6.1) \\
&= \sum_{\substack{m,x_1,\widehat{x}_2,s: \\ \Pr[m,x_1,\widehat{x}_2,s]>0}} \Pr[m, x_1, \widehat{x}_2, s]\, 2^{-\left\{\log \frac{\Pr[s|m,x_1]}{\Pr[s]} + \log \frac{1}{\Pr[s|m,x_1,\widehat{x}_2]}\right\}} \ ,
\end{aligned}$$

where Equation (6.1) follows from Bayes' rule. By Jensen's inequality,

$$\begin{aligned}
\Pr[\mathcal{R} \text{ accepts}] &\geq 2^{-\sum_{\substack{m,x_1,\widehat{x}_2,s: \\ \Pr[m,x_1,\widehat{x}_2,s]>0}} \Pr[m,x_1,\widehat{x}_2,s]\left\{\log \frac{\Pr[s|m,x_1]}{\Pr[s]} + \log \frac{1}{\Pr[s|m,x_1,\widehat{x}_2]}\right\}} \\
&= 2^{-\{\mathrm{I}(S;M,X_1) + \mathrm{H}(S|M,X_1,X_2)\}} \ ,
\end{aligned}$$

and therefore $\mathrm{I}(S; M, X_1) + \mathrm{H}(S|M, X_1, X_2) > \log\frac{1}{\epsilon} - 1$. ■

Lemma 6.3. *If $n \geq 3\log\frac{1}{\epsilon}$ and $\ell < 2\log(1/\epsilon) - 2$, then $\mathrm{I}(S; X_2|M, X_1) \geq \log\frac{1}{\epsilon} - 1$.*

Proof. Consider the following attack:

1. $\mathcal{A}$ chooses $m \in_{\mathrm{R}} \{0, 1\}^n$, as the sender's input, and runs an honest execution with the sender. At the end of this execution, the sender manually authenticates a string s. $\mathcal{A}$ reads s, and delays it. Now, $\mathcal{A}$'s goal is to cause the receiver to accept this string together with a different input message $\widehat{m}$.

[9] For any random variable Z we write $\Pr[z]$ instead of $\Pr[Z = z]$.

2. $\mathcal{A}$ samples $(\widehat{m}, \widehat{x}_1)$ from the joint distribution of (M, X_1) given s, and sends them to the receiver, who answers x_2.

3. If $\Pr[\widehat{m}, \widehat{x}_1, x_2, s] = 0$, then $\mathcal{A}$ quits. Otherwise, $\mathcal{A}$ forwards s to the receiver.

As in Lemma 6.2, $\epsilon \geq \Pr[\mathcal{R} \text{ accepts}] - \Pr[\widehat{m} = m]$. However, in this attack, unlike the previous attack, the messages m and $\widehat{m}$ are not chosen uniformly at random and independently. First m is chosen uniformly at random, then s is picked from the distribution of S given m, and then $\widehat{m}$ is chosen from the distribution of M given s. Therefore,

$$\begin{aligned}\Pr[\widehat{m} = m] &= \sum_s \Pr[s] \sum_m (\Pr[m|s])^2 \leq \sum_s \Pr[s] \max_m \Pr[m|s] \sum_m \Pr[m|s] \\ &= \sum_s \Pr[s] \max_m \Pr[m|s] = \sum_s \max_m \Pr[m, s] \leq \sum_s \max_m \Pr[m] \quad .\end{aligned}$$

Since the distribution of messages is uniform, and the authenticated string takes at most 2^ℓ values, we obtain $\Pr[\widehat{m} = m] \leq 2^{-n+\ell}$. From the assumptions that $\ell < 2\log(1/\epsilon) - 2$ and $n \geq 3\log(1/\epsilon)$ we get that $\Pr[\widehat{m} = m] < \epsilon$, and therefore $\Pr[\mathcal{R} \text{ accepts}] < 2\epsilon$. Now we analyze the probability that the receiver accepts. Notice that,

- $\widehat{m}$ and $\widehat{x}_1$ are chosen conditioned on s.
- x_2 is chosen conditioned only on $\widehat{m}$ and $\widehat{x}_1$.

Therefore,

$$\begin{aligned}\Pr[\mathcal{R} \text{ accepts}] &= \sum_{\widehat{m}, \widehat{x}_1, s} \Pr[\widehat{m}, \widehat{x}_1, s] \Pr[\mathcal{R} \text{ accepts} | \widehat{m}, \widehat{x}_1, s] \\ &= \sum_{\widehat{m}, \widehat{x}_1, s} \Pr[\widehat{m}, \widehat{x}_1, s] \sum_{\substack{x_2: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, s] > 0}} \Pr[x_2 | \widehat{m}, \widehat{x}_1] \\ &= \sum_{\substack{\widehat{m}, \widehat{x}_1, x_2, s: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1, x_2, s] \frac{\Pr[x_2 | \widehat{m}, \widehat{x}_1]}{\Pr[x_2 | \widehat{m}, \widehat{x}_1, s]} \qquad (6.2) \\ &= \sum_{\substack{\widehat{m}, \widehat{x}_1, x_2, s: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1, x_2, s] \, 2^{-\log \frac{\Pr[x_2 | \widehat{m}, \widehat{x}_1, s]}{\Pr[x_2 | \widehat{m}, \widehat{x}_1]}} \quad ,\end{aligned}$$

where Equation (6.2) follows from Bayes' rule. By Jensen's inequality,

$$\Pr[\mathcal{R} \text{ accepts}] \geq 2^{-\sum_{\substack{\widehat{m}, \widehat{x}_1, x_2, s: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1, x_2, s] \log \frac{\Pr[x_2 | \widehat{m}, \widehat{x}_1, s]}{\Pr[x_2 | \widehat{m}, \widehat{x}_1]}} = 2^{-\mathrm{I}(S; X_2 | M, X_1)} \quad ,$$

and therefore $\mathrm{I}(S; X_2 | M, X_1) > \log \frac{1}{\epsilon} - 1$. ∎

Now, Theorem 6.1 can be derived as follows. Suppose for contradiction that there exists a perfectly complete $(n, \ell, 3, \epsilon)$-authentication protocol, where no authentication tag is sent in the last round, and $n \geq 3\log(1/\epsilon)$ but $\ell < 2\log(1/\epsilon) - 2$. By using the fact that $\ell \geq \mathrm{H}(S)$, we can easily derive a contradiction: The above mentioned representation of $\mathrm{H}(S)$ and Lemmata 6.2 and 6.3 imply that $\mathrm{H}(S) > 2\log(1/\epsilon) - 2$. Therefore $\ell \geq 2\log(1/\epsilon) - 2$ in any such protocol. This concludes the proof of Theorem 6.1.

6.2 Proof of Theorem 4.4

In this section we present the proof Theorem 4.4. This proof is a natural generalization of the proof presented in Section 6.1. Note that, we can assume that in the last round the sender does not send an authentication tag x_i over the insecure channel. We represent the entropy of the manually authenticated string S by splitting it as follows:

$$\begin{aligned}\mathrm{H}(S) &= \mathrm{H}(S) - \mathrm{H}(S|M, X_1) + \sum_{i=1}^{k-2} \left(\mathrm{H}(S|M, X_1, \ldots, X_i) - \mathrm{H}(S|M, X_1, \ldots, X_{i+1})\right) \\ &\quad + \mathrm{H}(S|M, X_1, \ldots, X_{k-1}) \\ &= \mathrm{I}(S; M, X_1) + \sum_{i=1}^{k-2} \mathrm{I}(S; X_{i+1}|M, X_1, \ldots, X_i) + \mathrm{H}(S|M, X_1, \ldots, X_{k-1}) \ .\end{aligned}$$

Lemma 6.4. *If $n \geq 2\log\frac{1}{\epsilon}$, then*

$$\mathrm{I}(S; M, X_1) + \sum_{\substack{\text{even} \\ i}} \mathrm{I}(S; X_{i+1}|M, X_1, \ldots, X_i) + \mathrm{H}(S|M, X_1, \ldots, X_{k-1}) > \log\frac{1}{\epsilon} - 2 \ .$$

Proof. Consider the following attack:

1. The adversary $\mathcal{A}$ chooses $\widehat{m} \in_{\mathrm{R}} \{0,1\}^n$, and runs an honest execution with the receiver. Denote by s the manually authenticated string fixed by this execution. Now, $\mathcal{A}$'s goal is to cause the sender to manually authenticate this string.
2. $\mathcal{A}$ chooses $m \in_{\mathrm{R}} \{0,1\}^n$ as the sender's input, and receives x_1 from the sender.
3. It may be that x_1 is illegal given (m, s) (i.e., $\Pr[m, x_1, s] = 0$ in an honest execution), in which case $\mathcal{A}$ quits. Otherwise, $\mathcal{A}$ samples $\widehat{x}_2$ from the distribution of X_2 given (m, x_1, s) and sends it to the sender, who replies with x_3.
4. The rest of the attack is described as follows. Suppose that the sender replied with x_{i-1}. It may be that $\Pr[m, x_1, \widehat{x}_2, \ldots, x_{i-1}, s] = 0$ in an honest execution, in which case $\mathcal{A}$ quits. Otherwise, $\mathcal{A}$ samples $\widehat{x}_i$ from the distribution of X_i given $(m, x_1, \widehat{x}_2, \ldots, x_{i-1}, s)$ and sends it to the sender, who replies with x_{i+1}.
5. In the last round, the sender manually authenticates some string. If the sender did not authenticate s, $\mathcal{A}$ quits. Otherwise, $\mathcal{A}$ forwards s to the receiver.

By the unforgeability requirement of the protocol we obtain:

$$\epsilon \geq \Pr[\mathcal{A} \text{ cheats and } m \neq \widehat{m}] \geq \Pr[\mathcal{R} \text{ accepts}] - 2^{-n} \ .$$

Therefore, the assumption $n \geq 2\log\frac{1}{\epsilon}$ implies that $\Pr[\mathcal{R} \text{ accepts}] < 2\epsilon$. Now we analyze the probability that the receiver accepts. For ease of notation, we denote by t_i the transcript of the protocol before round $i+1$, i.e., $t_i = (m, x_1, \ldots, \widehat{x}_i)$ for even values of i, and $t_i = (m, x_1, \ldots, x_i)$ for odd values of i. Notice that,

- m and x_1 are chosen independently of s.
- The $\widehat{x}_i$'s are chosen conditioned on t_{i-1} and on s.

- The x_i's and s are chosen conditioned only on t_{i-1}.

Therefore, similarly to the analysis in Lemma 6.2 and by using Bayes' rule (note that in this general case the protocol is not perfectly complete, and therefore we are only guaranteed that the receiver accepts "good" transcripts with probability at least 1/2),

$$\begin{aligned}
\Pr[\mathcal{R}\text{ accepts}] &\geq \frac{1}{2} \sum_{\substack{t_{k-1},s:\\ \Pr[t_{k-1},s]>0}} \Bigg(\Pr[s]\Pr[m,x_1] \\
&\qquad \times \Bigg\{ \prod_{\substack{\text{even}\\ i<k-1}} \Pr[\widehat{x}_i|t_{i-1},s]\Pr[x_{i+1}|t_i] \Bigg\} \Pr[\widehat{x}_{k-1}|t_{k-2},s]\Pr[s|t_{k-1}] \Bigg) \\
&= \frac{1}{2} \sum_{\substack{t_{k-1},s:\\ \Pr[t_{k-1},s]>0}} \Bigg(\Pr[m,x_1,s] \frac{\Pr[s]}{\Pr[s|m,x_1]} \\
&\qquad \times \Bigg\{ \prod_{\substack{\text{even}\\ i<k-1}} \Pr[\widehat{x}_i,x_{i+1}|t_{i-1},s] \frac{\Pr[x_{i+1}|t_i]}{\Pr[x_{i+1}|t_i,s]} \Bigg\} \Pr[\widehat{x}_{k-1}|t_{k-2},s]\Pr[s|t_{k-1}] \Bigg) \\
&= \frac{1}{2} \sum_{\substack{t_{k-1},s:\\ \Pr[t_{k-1},s]>0}} \Pr[t_{k-1},s] \frac{\Pr[s]}{\Pr[s|m,x_1]} \Bigg\{ \prod_{\substack{\text{even}\\ i}} \frac{\Pr[x_{i+1}|t_i]}{\Pr[x_{i+1}|t_i,s]} \Bigg\} \Pr[s|t_{k-1}] \\
&= \frac{1}{2} \sum_{\substack{t_{k-1},s:\\ \Pr[t_{k-1},s]>0}} \Pr[t_{k-1},s]\, 2^{-\left\{ \log\frac{\Pr[s|m,x_1]}{\Pr[s]} + \sum_{\text{even } i} \log\frac{\Pr[x_{i+1}|t_i,s]}{\Pr[x_{i+1}|t_i]} + \log\frac{1}{\Pr[s|t_{k-1}]} \right\}} .
\end{aligned}$$

By Jensen's inequality,

$$\begin{aligned}
\Pr[\mathcal{R}\text{ accepts}] &\geq 2^{-\sum_{\substack{t_{k-1},s:\\ \Pr[t_{k-1},s]>0}} \Pr[t_{k-1},s] \left\{ \log\frac{\Pr[s|m,x_1]}{\Pr[s]} + \sum_{\text{even } i} \log\frac{\Pr[x_{i+1}|t_i,s]}{\Pr[x_{i+1}|t_i]} + \log\frac{1}{\Pr[s|t_{k-1}]} \right\} - 1} \\
&= 2^{-\left\{ \mathrm{I}(S;M,X_1) + \sum_{\text{even } i} \mathrm{I}(S;X_{i+1}|M,X_1,\dots,X_i) + \mathrm{H}(S|M,X_1,\dots,X_{k-1}) + 1 \right\}} .
\end{aligned}$$

and therefore $\mathrm{I}(S;M,X_1) + \sum_{\text{even } i} \mathrm{I}(S;X_{i+1}|M,X_1,\dots,X_i) + \mathrm{H}(S|M,X_1,\dots,X_{k-1}) > \log\frac{1}{\epsilon} - 2$. ∎

Lemma 6.5. *If $n \geq 3\log\frac{1}{\epsilon}$ and $\ell < 2\log(1/\epsilon) - 4$, then*

$$\sum_{\substack{\text{odd}\\ i}} \mathrm{I}(S;X_{i+1}|M,X_1,\dots,X_i) > \log\frac{1}{\epsilon} - 2 .$$

Proof. Consider the following attack:

1. $\mathcal{A}$ chooses $m \in_{\mathrm{R}} \{0,1\}^n$, as the sender's input, and runs an honest execution with the sender. At the end of this execution, the sender manually authenticates a string s. $\mathcal{A}$ reads s, and delays it. Now, $\mathcal{A}$'s goal is to cause the receiver to accept this string together with a different input message $\widehat{m}$.

2. $\mathcal{A}$ samples $(\widehat{m}, \widehat{x}_1)$ from the joint distribution of (M, X_1) given s, and sends them to the receiver, who answers x_2.

3. The rest of the attack is described as follows. Suppose that the receiver answers with x_{i-1}. It may be that $\Pr[\widehat{m}, \widehat{x}_1, \ldots, x_{i-1}, s] = 0$ in an honest execution, in which case $\mathcal{A}$ quits. Otherwise, $\mathcal{A}$ samples $\widehat{x}_i$ from the distribution of X_i given $(\widehat{m}, \widehat{x}_1, \ldots, x_{i-1}, s)$ and sends it to the sender, who replies with x_{i+1}.

4. In the last round, $\mathcal{A}$ forwards s to the receiver.

As in Lemma 6.3, $\Pr[\mathcal{R} \text{ accepts}] < 2\epsilon$. Now we analyze the probability that the receiver accepts. For ease of notation, we denote again by t_i the transcript of the protocol before round $i+1$, i.e., $t_i = (\widehat{m}, \widehat{x}_1, \ldots, \widehat{x}_i)$ for odd values of i, and $t_i = (\widehat{m}, \widehat{x}_1, \ldots, x_i)$ for even values of i. Notice that,

- $\widehat{m}$ and the $\widehat{x}_i$'s are chosen conditioned on t_{i-1} and on s.
- The x_i's are chosen conditioned only on t_{i-1}.

Therefore, similarly to the analysis in Lemma 6.3 and by using Bayes' rule,

$$\begin{aligned}
\Pr[\mathcal{R} \text{ accepts}] &\geq \frac{1}{2} \sum_{\substack{t_{k-1}, s: \\ \Pr[t_{k-1}, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1, s] \Pr[x_2 | t_1] \prod_{\substack{\text{odd} \\ 3 \leq i < k}} \Pr[\widehat{x}_i | t_{i-1}, s] \Pr[x_{i+1} | t_i] \\
&= \frac{1}{2} \sum_{\substack{t_{k-1}, s: \\ \Pr[t_{k-1}, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1, x_2, s] \frac{\Pr[x_2 | t_1]}{\Pr[x_2 | t_1, s]} \prod_{\substack{\text{odd} \\ 3 \leq i < k}} \Pr[\widehat{x}_i, x_{i+1} | t_{i-1}, s] \frac{\Pr[x_{i+1} | t_i]}{\Pr[x_{i+1} | t_i, s]} \\
&= \frac{1}{2} \sum_{\substack{t_{k-1}, s: \\ \Pr[t_{k-1}, s] > 0}} \Pr[t_{k-1}, s] \prod_{\substack{\text{odd} \\ i}} \frac{\Pr[x_{i+1} | t_i]}{\Pr[x_{i+1} | t_i, s]} \\
&= \frac{1}{2} \sum_{\substack{t_{k-1}, s: \\ \Pr[t_{k-1}, s] > 0}} \Pr[t_{k-1}, s] \, 2^{-\sum_{\text{odd } i} \log \frac{\Pr[x_{i+1} | t_i, s]}{\Pr[x_{i+1} | t_i]}} .
\end{aligned}$$

By Jensen's inequality,

$$\begin{aligned}
\Pr[\mathcal{R} \text{ accepts}] &\geq 2^{-\left\{ \sum_{\substack{t_{k-1}, s: \\ \Pr[t_{k-1}, s] > 0}} \Pr[t_{k-1}, s] \sum_{\text{odd } i} \log \frac{\Pr[x_{i+1} | t_i, s]}{\Pr[x_{i+1} | t_i]} + 1 \right\}} \\
&= 2^{-\left\{ \sum_{\text{odd } i} \mathrm{I}(S; X_{i+1} | M, X_1, \ldots, X_i) + 1 \right\}} ,
\end{aligned}$$

and therefore $\sum_{\text{odd } i} \mathrm{I}(S; X_{i+1} | M, X_1, \ldots, X_i) > \log \frac{1}{\epsilon} - 2$. ■

Now, Theorem 4.4 can be derived as follows. Let Π be an unconditionally secure (n, ℓ, k, ϵ)-authentication protocol, and suppose for contradiction that $n \geq 2\log(1/\epsilon) + 4$ and $\ell \leq 2\log(1/\epsilon) - 6$. Then, a similar argument to that in Lemma 8.1 implies that there exists an unconditionally secure $(n' = 3\log(1/\epsilon), \ell, k', 2\epsilon)$ authentication protocol Π', for some $k' > k$, where no authentication tag is sent in the last round. Now, the above mentioned representation of $\mathrm{H}(S)$ (where S is the manually authenticated string in Π') and Lemmata 6.4 and 6.5 imply that $\mathrm{H}(S) > 2\log(1/2\epsilon) - 4 = 2\log(1/\epsilon) - 6$, and in particular $\ell > 2\log(1/\epsilon) - 6$. This contradicts the existence of Π, and concludes the proof of Theorem 4.4.

7 Lower Bound in the Shared Key Model

In this section we prove a lower bound on the entropy of the shared key. For simplicity, we first present in Section 7.1 a proof for the simplified case of a perfectly complete 3-round protocol. This simplified proof already captures the main ideas and difficulties of the general proof. The general proof is based on the same analysis, and is described in Section 7.2.

As in Section 6, in the following proofs given any authentication protocol in the shared key model we identify the messages sent by the sender and the receiver and the shared key with corresponding random variables. More specifically, when the shared key is chosen from its specified distribution (which is defined by the given protocol), and the input message m is chosen uniformly at random, the honest execution of the protocol defines a probability distribution on the shared key s, the message m, and the tags x_i. We denote by S, M and X_i the random variables corresponding to s, m and x_i, respectively. We note that our lower bound holds even for protocols which are secure only when authenticating a random message m.

7.1 A Simplified Proof

We prove the following theorem:

Theorem 7.1. *For any perfectly complete $(n, \ell, 3, \epsilon)$-authentication protocol in the shared key model, it holds that $\mathrm{H}(S) \geq 2\log(1/\epsilon)$, where S is the ℓ-bit shared key.*

As mentioned above, when the shared key s is chosen from its specified distribution, and the input message m is chosen uniformly at random, the honest execution of the protocol defines a probability distribution on the shared key s, the message m, the authentication tag x_1 (sent by the sender in the first round together with m), the authentication tag x_2 (sent by the receiver in the second round), and the authentication tag x_3 (sent by the sender in the third round). We denote by S, M, X_1, X_2 and X_3 the corresponding random variables.

We apply again the proof technique described in Section 6, and represent the entropy of the shared key S by splitting it as follows:

$$\begin{aligned}
\mathrm{H}(S) &= (\mathrm{H}(S) - \mathrm{H}(S|M, X_1)) + (\mathrm{H}(S|M, X_1) - \mathrm{H}(S|M, X_1, X_2)) \\
&\quad + (\mathrm{H}(S|M, X_1, X_2) - \mathrm{H}(S|M, X_1, X_2, X_3)) + \mathrm{H}(S|M, X_1, X_2, X_3) \\
&= \mathrm{I}(S; M, X_1) + \mathrm{I}(S; X_2|M, X_1) + \mathrm{I}(S; X_3|M, X_1, X_2) + \mathrm{H}(S|M, X_1, X_2, X_3) \ .
\end{aligned}$$

As in Section 6.1, we first state and prove two lemmata on the reduction in $\mathrm{H}(S)$ during the execution of the protocol, and then show that correctness of Theorem 7.1 follows. Notice that in this model we manage to prove the lower bound for the entropy of the shared key and not only for its length, as in Section 6.

Lemma 7.2. $\mathrm{I}(S; M, X_1) + \mathrm{I}(S; X_3|M, X_1, X_2) \geq \log \frac{1}{\epsilon}$.

Proof. Consider the following attack, in which $\mathcal{A}$ impersonates the sender, and tries to cheat the receiver into accepting a fraudulent message $\widehat{m}$:

1. Denote by s the actual shared key. $\mathcal{A}$ chooses $(\widehat{m}, \widehat{x}_1)$ according to the joint distribution of (M, X_1). It may be that the chosen $\widehat{x}_1$ is illegal given $(\widehat{m}, s)$ (i.e., $\Pr[\widehat{m}, \widehat{x}_1, s] = 0$ in an honest execution), in which case the receiver will quit the protocol. Otherwise, the receiver answers x_2.

2. $\mathcal{A}$ chooses $\widehat{x}_3$ according to the distribution of X_3 given $(\widehat{m}, \widehat{x}_1, x_2)$. By the perfect completeness assumption, if the chosen $\widehat{x}_3$ is legal given $(\widehat{m}, \widehat{x}_1, x_2, s)$, then the receiver must accept $\widehat{m}$.

By the unforgeability requirement, we have that $\epsilon \geq \Pr[\mathcal{R} \text{ accepts}]$. Notice that,

- $\widehat{m}$ and $\widehat{x}_1$ are chosen independently of s.
- x_2 is chosen conditioned on $\widehat{m}, \widehat{x}_1$ and s.
- $\widehat{x}_3$ is chosen conditioned only on $\widehat{m}, \widehat{x}_1$ and x_2.

Therefore,

$$\begin{aligned}
\Pr[\mathcal{R} \text{ accepts}] &= \sum_s \Pr[s] \Pr[\mathcal{R} \text{ accepts} | s] \\
&= \sum_s \Pr[s] \sum_{\substack{\widehat{m}, \widehat{x}_1: \\ \Pr[\widehat{m}, \widehat{x}_1, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1] \Pr[\mathcal{R} \text{ accepts} | \widehat{m}, \widehat{x}_1, s] \\
&= \sum_{\substack{\widehat{m}, \widehat{x}_1, s: \\ \Pr[\widehat{m}, \widehat{x}_1, s] > 0}} \Pr[s] \Pr[\widehat{m}, \widehat{x}_1] \sum_{x_2} \Pr[x_2 | \widehat{m}, \widehat{x}_1, s] \Pr[\mathcal{R} \text{ accepts} | \widehat{m}, \widehat{x}_1, x_2, s] \\
&= \sum_{\substack{\widehat{m}, \widehat{x}_1, x_2, s: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, s] > 0}} \Pr[s] \Pr[\widehat{m}, \widehat{x}_1] \Pr[x_2 | \widehat{m}, \widehat{x}_1, s] \sum_{\substack{\widehat{x}_3: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s] > 0}} \Pr[\widehat{x}_3 | \widehat{m}, \widehat{x}_1, x_2] \\
&= \sum_{\substack{\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s] > 0}} \Pr[s] \Pr[\widehat{m}, \widehat{x}_1] \Pr[x_2 | \widehat{m}, \widehat{x}_1, s] \Pr[\widehat{x}_3 | \widehat{m}, \widehat{x}_1, x_2] \\
&= \sum_{\substack{\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1, s] \frac{\Pr[s]}{\Pr[s | \widehat{m}, \widehat{x}_1]} \Pr[x_2, \widehat{x}_3 | \widehat{m}, \widehat{x}_1, s] \frac{\Pr[\widehat{x}_3 | \widehat{m}, \widehat{x}_1, x_2]}{\Pr[\widehat{x}_3 | \widehat{m}, \widehat{x}_1, x_2, s]} \\
&= \sum_{\substack{\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s]\, 2^{-\left\{\log \frac{\Pr[s | \widehat{m}, \widehat{x}_1]}{\Pr[s]} + \log \frac{\Pr[\widehat{x}_3 | \widehat{m}, \widehat{x}_1, x_2, s]}{\Pr[\widehat{x}_3 | \widehat{m}, \widehat{x}_1, x_2]}\right\}} .
\end{aligned}$$

By Jensen's inequality,

$$\begin{aligned}
\Pr[\mathcal{R} \text{ accepts}] &\geq 2^{-\sum_{\substack{\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s: \\ \Pr[\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1, x_2, \widehat{x}_3, s] \left\{\log \frac{\Pr[s | \widehat{m}, \widehat{x}_1]}{\Pr[s]} + \log \frac{\Pr[\widehat{x}_3 | \widehat{m}, \widehat{x}_1, x_2, s]}{\Pr[\widehat{x}_3 | \widehat{m}, \widehat{x}_1, x_2]}\right\}} \\
&= 2^{-\{\mathrm{I}(S; M, X_1) + \mathrm{I}(S; X_3 | M, X_1, X_2)\}} ,
\end{aligned}$$

and therefore $\mathrm{I}(S; M, X_1) + \mathrm{I}(S; X_3 | M, X_1, X_2) \geq \log \frac{1}{\epsilon}$. ∎

Lemma 7.3. $\mathrm{I}(S; X_2 | M, X_1) + \mathrm{H}(S | M, X_1, X_2, X_3) \geq \log \frac{1}{\epsilon}$.

Proof. Consider the following attack, in which $\mathcal{A}$ impersonates the receiver, runs the protocol with the sender, and then tries to guess the shared key. If $\mathcal{A}$ guessed correctly, she can then impersonate the sender, and run the protocol with the receiver on any message she wishes.

1. Denote by s the actual shared key. $\mathcal{A}$ chooses $m \in_{\mathrm{R}} \{0,1\}^n$ as the sender's input, and receives x_1 from the sender.

2. $\mathcal{A}$ chooses $\widehat{x}_2$ according to the distribution of X_2 given (m, x_1). It may be the chosen $\widehat{x}_2$ is illegal given (m, x_1, s) (i.e., $\Pr[m, x_1, \widehat{x}_2, s] = 0$ in an honest execution), in which case the sender will quit the protocol. Otherwise, the sender answers x_3.

3. $\mathcal{A}$ tries to guess the shared key by sampling the distribution of S given $(m, x_1, \widehat{x}_2, x_3)$. Therefore, $\mathcal{A}$ guesses the correct value s with probability $\Pr[s|m, x_1, \widehat{x}_2, x_3]$.

By the unforgeability requirement, we have that $\epsilon \geq \Pr[\mathcal{A} \text{ guesses } S]$. Notice that,

- x_1 is chosen conditioned on m and s.
- $\widehat{x}_2$ is chosen conditioned only on m and x_1.
- x_3 is chosen conditioned on $m, x_1, \widehat{x}_2$ and s.

Therefore, similarly to the analysis in Lemma 6.2 and by using Bayes' rule,

$$\begin{aligned}
\Pr[\mathcal{A} \text{ guesses } S] &= \sum_{m,x_1,s} \Pr[m, x_1, s] \Pr[\mathcal{A} \text{ guesses } S|m, x_1, s] \\
&= \sum_{m,x_1,s} \Pr[m, x_1, s] \sum_{\substack{\widehat{x}_2: \\ \Pr[m,x_1,\widehat{x}_2,s]>0}} \Pr[\widehat{x}_2|m, x_1] \Pr[\mathcal{A} \text{ guesses } S|m, x_1, \widehat{x}_2, s] \\
&= \sum_{\substack{m,x_1,\widehat{x}_2,s: \\ \Pr[m,x_1,\widehat{x}_2,s]>0}} \Pr[m, x_1, s] \Pr[\widehat{x}_2|m, x_1] \sum_{x_3} \Pr[x_3|m, x_1, \widehat{x}_2, s] \Pr[\mathcal{A} \text{ guesses } S|m, x_1, \widehat{x}_2, x_3, s] \\
&= \sum_{\substack{m,x_1,\widehat{x}_2,x_3,s: \\ \Pr[m,x_1,\widehat{x}_2,x_3,s]>0}} \Pr[m, x_1, s] \Pr[\widehat{x}_2|m, x_1] \Pr[x_3|m, x_1, \widehat{x}_2, s] \Pr[s|m, x_1, \widehat{x}_2, x_3] \\
&= \sum_{\substack{m,x_1,\widehat{x}_2,x_3,s: \\ \Pr[m,x_1,\widehat{x}_2,x_3,s]>0}} \Pr[m, x_1, \widehat{x}_2, s] \frac{\Pr[\widehat{x}_2|m, x_1]}{\Pr[\widehat{x}_2|m, x_1, s]} \Pr[x_3|m, x_1, \widehat{x}_2, s] \Pr[s|m, x_1, \widehat{x}_2, x_3] \\
&= \sum_{\substack{m,x_1,\widehat{x}_2,x_3,s: \\ \Pr[m,x_1,\widehat{x}_2,x_3,s]>0}} \Pr[m, x_1, \widehat{x}_2, x_3, s]\, 2^{-\left\{\log \frac{\Pr[\widehat{x}_2|m,x_1,s]}{\Pr[\widehat{x}_2|m,x_1]} + \log \frac{1}{\Pr[s|m,x_1,\widehat{x}_2,x_3]}\right\}}
\end{aligned}$$

By Jensen's inequality,

$$\begin{aligned}
\Pr[\mathcal{A} \text{ guesses } S] &\geq 2^{-\sum_{\substack{m,x_1,\widehat{x}_2,x_3,s: \\ \Pr[m,x_1,\widehat{x}_2,x_3,s]>0}} \Pr[m,x_1,\widehat{x}_2,x_3,s]\left\{\log \frac{\Pr[\widehat{x}_2|m,x_1,s]}{\Pr[\widehat{x}_2|m,x_1]} + \log \frac{1}{\Pr[s|m,x_1,\widehat{x}_2,x_3]}\right\}} \\
&= 2^{-\{\mathrm{I}(S;X_2|M,X_1) + \mathrm{H}(S|M,X_1,X_2,X_3)\}} ,
\end{aligned}$$

and therefore $\mathrm{I}(S; X_2|M, X_1) + \mathrm{H}(S|M, X_1, X_2, X_3) \geq \log \frac{1}{\epsilon}$. ■

Now, Theorem 7.1 can be derived as follows. Let Π be any perfectly complete $(n, \ell, 3, \epsilon)$-authentication protocol in the shared key model. Then, the above representation of the entropy of the shared key used in Π and Lemmata 7.2 and 7.3 imply that $\mathrm{H}(S) \geq 2\log(1/\epsilon)$. This concludes the proof of Theorem 7.1.

7.2 Proof of Theorem 4.5

In this section we present the proof of Theorem 4.5. This proof is a natural generalization of the proof presented in Section 7.1. Again, we represent the entropy of the shared key S by splitting it as follows:

$$\begin{aligned}
\mathrm{H}(S) &= \mathrm{H}(S) - \mathrm{H}(S|M, X_1) + \sum_{i=1}^{k-2} (\mathrm{H}(S|M, X_1, \ldots, X_i) - \mathrm{H}(S|M, X_1, \ldots, X_{i+1})) \\
&\quad + \mathrm{H}(S|M, X_1, \ldots, X_{k-1}) \\
&= \mathrm{I}(S; M, X_1) + \sum_{i=1}^{k-2} \mathrm{I}(S; X_{i+1}|M, X_1, \ldots, X_i) + \mathrm{H}(S|M, X_1, \ldots, X_{k-1}) \ .
\end{aligned}$$

We first state and prove two lemmata on the reduction of $\mathrm{H}(S)$ during the execution of the protocol, and then show that these lemmata, and the above representation of $\mathrm{H}(S)$ imply Theorem 4.5. Notice that in this model we manage to prove the lower bound for the entropy of the shared key and not only for its length, as in Section 6.2.

Lemma 7.4. $\mathrm{I}(S; M, X_1) + \sum_{\substack{\text{even} \\ i}} \mathrm{I}(S; X_{i+1}|M, X_1, \ldots, X_i) > \log \frac{1}{\epsilon} - 1$.

Proof. Consider the following attack, in which $\mathcal{A}$ impersonates the sender, and tries to cheat the receiver into accepting a fraudulent message $\widehat{m}$:

1. Denote by s the actual shared key. $\mathcal{A}$ chooses $(\widehat{m}, \widehat{x}_1)$ according to the joint distribution of (M, X_1). It may be that the chosen $\widehat{x}_1$ is illegal given $(\widehat{m}, s)$ (i.e., $\Pr[\widehat{m}, \widehat{x}_1, s] = 0$ in an honest execution), in which case the receiver will quit the protocol. Otherwise, the receiver answers x_2.

2. The rest of the attack is described as follows. Suppose that the receiver replied x_i. $\mathcal{A}$ chooses $\widehat{x}_{i+1}$ according to the distribution of X_{i+1} given $(\widehat{m}, \widehat{x}_1, \ldots, x_i)$. If the chosen $\widehat{x}_{i+1}$ is legal given $(\widehat{m}, \widehat{x}_1, \ldots, x_i, s)$, then the receiver continues the protocols, and answers x_{i+2}.

3. In the last round, if $\mathcal{A}$ chooses $\widehat{x}_k$ that is legal given $(\widehat{m}, \widehat{x}_1, \ldots, x_{k-1}, s)$, then the receiver must accept with probability at least $1/2$.

Then, $\epsilon \geq \Pr[\mathcal{R} \text{ accepts}]$. For ease of notation, we denote again by t_i the transcript of the protocol before round $i+1$, i.e., $t_i = (\widehat{m}, \widehat{x}_1, \ldots, \widehat{x}_i)$ for odd values of i, and $t_i = (\widehat{m}, \widehat{x}_1, \ldots, x_i)$ for even values of i. Notice that,

- $\widehat{m}$ and the $\widehat{x}_i$'s are chosen conditioned only on t_{i-1}.
- The x_i's are chosen conditioned on t_{i-1} and on s.

Therefore, similarly to the analysis in Lemma 7.2 and by using Bayes' rule,

$$\begin{aligned}
\Pr[\mathcal{R} \text{ accepts}] &= \frac{1}{2} \sum_{\substack{t_k, s: \\ \Pr[t_k, s] > 0}} \Pr[s] \Pr[\widehat{m}, \widehat{x}_1] \prod_{\substack{\text{even} \\ i}} \Pr[x_i|t_{i-1}, s] \Pr[\widehat{x}_{i+1}|t_i] \\
&= \frac{1}{2} \sum_{\substack{t_k, s: \\ \Pr[t_k, s] > 0}} \Pr[\widehat{m}, \widehat{x}_1, s] \frac{\Pr[s]}{\Pr[s|\widehat{m}, \widehat{x}_1]} \prod_{\substack{\text{even} \\ i}} \Pr[x_i, \widehat{x}_{i+1}|t_{i-1}, s] \frac{\Pr[\widehat{x}_{i+1}|t_i]}{\Pr[\widehat{x}_{i+1}|t_i, s]} \\
&= \frac{1}{2} \sum_{\substack{t_k, s: \\ \Pr[t_k, s] > 0}} \Pr[t_k, s] \, 2^{-\left\{\log \frac{\Pr[s|\widehat{m}, \widehat{x}_1]}{\Pr[s]} + \sum_{\substack{\text{even} \\ i}} \log \frac{\Pr[\widehat{x}_{i+1}|t_i, s]}{\Pr[\widehat{x}_{i+1}|t_i]}\right\}} \ .
\end{aligned}$$

By Jensen's inequality,

$$\Pr\left[\mathcal{R} \text{ accepts}\right] \geq 2^{-\sum_{\substack{t_k,s:\\ \Pr[t_k,s]>0}} \Pr[t_k,s]\left\{\log\frac{\Pr[s|\widehat{m},\widehat{x}_1]}{\Pr[s]}+\sum_{\substack{\text{even}\\ i}}\log\frac{\Pr[\widehat{x}_{i+1}|t_i,s]}{\Pr[\widehat{x}_{i+1}|t_i]}\right\}-1}$$
$$= 2^{-\left\{\mathrm{I}(S;M,X_1)+\sum_{\substack{\text{even}\\ i}}\mathrm{I}(S;X_{i+1}|M,X_1,\ldots,X_i)+1\right\}} ,$$

and therefore $\mathrm{I}(S;M,X_1)+\sum_{\substack{\text{even}\\ i}}\mathrm{I}(S;X_{i+1}|M,X_1,\ldots,X_i) \geq \log\frac{1}{\epsilon}-1$. ■

Lemma 7.5. $\sum_{\substack{\text{odd}\\ i}}\mathrm{I}(S;X_{i+1}|M,X_1,\ldots,X_i)+\mathrm{H}(S|M,X_1,\ldots,X_{k-1}) > \log\frac{1}{\epsilon}-1$.

Proof. Consider the following attack, in which $\mathcal{A}$ impersonates the receiver, runs the protocol with the sender, and then tries to guess the shared key. If $\mathcal{A}$ guessed correctly, she can then impersonate the sender, and run the protocol with the receiver on any message she wishes.

1. Denote by s the actual shared key. $\mathcal{A}$ chooses $m \in_{\mathrm{R}} \{0,1\}^n$ as the sender's input, and receives x_1 from the sender.

2. The rest of the attack is described as follows. Suppose that the sender replied x_i. $\mathcal{A}$ chooses $\widehat{x}_{i+1}$ according to the distribution of X_{i+1} given $(m,x_1,\ldots,x_i)$. It may be the chosen $\widehat{x}_{i+1}$ is illegal given $(m,x_1,\ldots,x_i,s)$, in which case the sender will quit the protocol. Otherwise, the sender replies x_{i+2}.

3. $\mathcal{A}$ tries to guess the shared key by sampling the distribution of S given $(m,x_1,\ldots,x_k)$. Therefore, $\mathcal{A}$ guesses the correct value s with probability $\Pr[s|m,x_1,\ldots,x_k]$.

Then, $\epsilon \geq \frac{1}{2}\Pr[\mathcal{A} \text{ guesses } S]$. We denote again by t_i the transcript of the protocol before round $i+1$, i.e., $t_i=(m,x_1,\ldots,x_i)$ for odd values of i, and $t_i=(m,x_1,\ldots,\widehat{x}_i)$ for even values of i. Notice that

- The $\widehat{x}_i$'s are chosen conditioned only on t_{i-1}.
- The x_i's are chosen conditioned on t_{i-1} and on s.

Therefore, similarly to the analysis in Lemma 7.3 and by using Bayes' rule,

$$\Pr[\mathcal{A} \text{ guesses } S] = \sum_{\substack{t_k,s:\\ \Pr[t_k,s]>0}} \Pr[m,x_1,s]\Pr[\widehat{x}_2|t_1]\left\{\prod_{\substack{\text{odd}\\ 1<i}}\Pr[x_i|t_{i-1},s]\Pr[\widehat{x}_{i+1}|t_i]\right\}\Pr[s|t_k]$$
$$= \sum_{\substack{t_k,s:\\ \Pr[t_k,s]>0}} \Pr[m,x_1,\widehat{x}_2,s]\frac{\Pr[\widehat{x}_2|t_1]}{\Pr[\widehat{x}_2|t_1,s]}\left\{\prod_{\substack{\text{odd}\\ 1<i}}\Pr[x_i,\widehat{x}_{i+1}|t_{i-1},s]\frac{\Pr[\widehat{x}_{i+1}|t_i]}{\Pr[\widehat{x}_{i+1}|t_i,s]}\right\}\Pr[s|t_k]$$
$$= \sum_{\substack{t_k,s:\\ \Pr[t_k,s]>0}} \Pr[t_k,s]\,2^{-\left\{\sum_{\substack{\text{odd}\\ i}}\log\frac{\Pr[\widehat{x}_{i+1}|t_i,s]}{\Pr[\widehat{x}_{i+1}|t_i]}+\log\frac{1}{\Pr[s|t_k]}\right\}} .$$

By Jensen's inequality,

$$\Pr[\mathcal{A} \text{ guesses } S] \geq 2^{-\sum_{\substack{t_k,s:\\ \Pr[t_k,s]>0}} \Pr[t_k,s]\left\{\sum_{\substack{\text{odd}\\ i}}\log\frac{\Pr[\widehat{x}_{i+1}|t_i,s]}{\Pr[\widehat{x}_{i+1}|t_i]}+\log\frac{1}{\Pr[s|t_k]}\right\}}$$
$$= 2^{-\left\{\sum_{\substack{\text{odd}\\ i}}\mathrm{I}(S;X_{i+1}|M,X_1,\ldots,X_i)+\mathrm{H}(S|M,X_1,\ldots,X_{k-1})\right\}} ,$$

and therefore $\sum_{\substack{\text{odd}\\ i}}\mathrm{I}(S;X_{i+1}|M,X_1,\ldots,X_i)+\mathrm{H}(S|M,X_1,\ldots,X_{k-1}) \geq \log\frac{1}{\epsilon}-1$. ■

Now, Theorem 4.5 can be derived as follows. Let Π be any (n, ℓ, k, ϵ)-authentication protocol in the shared key model. Then, the above representation of the entropy of the shared key used in Π and Lemmata 7.4 and 7.5 imply that $\mathrm{H}(S) \geq 2\log(1/\epsilon) - 2$. This concludes the proof of Theorem 4.5.

8 Breaking the Lower Bounds Implies One-Way Functions

In this section we prove Theorems 4.6 and 4.7, namely, we show that in the computational setting one-way functions are necessary for the existence of protocols breaking the lower bounds stated in Theorems 4.4 and 4.5. We prove here only the result for the manual channel model, since the proof for the shared key model is very similar and can be easily derived from the following proof.

The idea underlying the proof is showing that if distributionally one-way functions do not exist, then the attacks described in Section 6 can be carried out by a polynomial-time adversary with almost the same success probability. In Section 6 we used the fact that the adversary is computationally unbounded in that she can sample distributions induced by the protocol. This usage of the adversary's capabilities, can alternatively be seen as randomly inverting functions given the image of a random input. The proof proceeds in two stages:

1. In Lemma 8.1 we prove that if there exists an authentication protocol that enables the user to authenticate relatively short messages, then this protocol can be used to *efficiently* authenticate long messages.
2. In Lemma 8.2 we prove that if there exists a computationally secure $(n, \ell, k, \epsilon, t)$-authentication protocol, such that $n \geq 1/\epsilon$, $\ell < 2\log(1/\epsilon) - 6$ and $t = \Omega(\mathrm{poly}(n, k, 1/\epsilon))$, then distributionally one-way functions exist.

We note that the lengthening of the input messages in the first stage is necessary for the second stage, since the polynomial-time adversary succeeds with *almost* the same probability as in Section 6. Therefore, in order to overcome this inaccuracy we need input messages which are much longer than the manually authenticated strings.

Lemma 8.1. *If there exists a computationally secure $(n = 2\log(1/\epsilon) + 4, \ell, k, \epsilon, t)$-authentication protocol Π in the manual channel model, then for any $n' = poly(t)$ there exists a computationally secure $(n', \ell, k' = k + \log^* n + 2, \epsilon' = 2\epsilon, t)$-authentication protocol Π'.*

Proof. We construct the protocol Π' by composing the protocol $\mathrm{P}_{\log^* n'+2}$, which was described in Section 5, with the given protocol Π as follows. Given an n'-bit input message m, the sender and the receiver execute protocol $\mathrm{P}_{\log^* n'+2}$ for authenticating m with ϵ as the bound on the adversary's forgery probability. However, they replace step 4 of the execution with an execution of Π to authenticate $m_{\mathcal{S}}^{\log^* n'+2}$ (rather than manually authenticating it).

This composition is possible, since a proof similar to that of Claim 5.3 shows that the length of $m_{\mathcal{S}}^{\log^* n'+2}$ is at most $2\log(1/\epsilon) + 4$ bits. Clearly, Π' is a $(k + \log^* n' + 2)$-round protocol, in which at most ℓ bits are manually authenticated. Moreover, for any adversary running in time $\mathrm{poly}(t)$ and for any constant $c > 0$, the forgery probability of the adversary is at most $2\epsilon + t^{-c}$ for sufficiently large t (otherwise, such an adversary can be used to forge either in $\mathrm{P}_{\log^* n'+2}$ or in Π). ■

Lemma 8.2. *If there exists a computationally secure $(n, \ell, k, \epsilon, t)$-authentication protocol in the manual channel model, such that $n = \lceil 1/\epsilon \rceil$, $\ell < 2\log(1/\epsilon) - 6$ and $t = \Omega(poly(n, k, 1/\epsilon))$, then distributionally one-way functions exist.*

Proof. As mentioned above, we prove that if distributionally one-way functions do not exist, then the attacks described in Section 6 can be carried out by a polynomial-time adversary with almost the same success probability. As in section 6, we present here the proof for the simplified case of a perfectly complete 3-round protocol, and then explain how it can be generalized. We first focus on the attack described in Lemma 6.2.

Let f be a function defined as follows: f takes as input three strings $r_{\mathcal{S}}$, $r_{\mathcal{R}}$ and m, and outputs (m, x_1, x_2, s) – the transcript of the protocol, where $r_{\mathcal{S}}$, $r_{\mathcal{R}}$ and m are the random coins of the sender, the random coins of the receiver, and the input message, respectively. Let f' denote the function that is obtained from f by eliminating its third output, i.e., $f'(r_{\mathcal{S}}, r_{\mathcal{R}}, m) = (m, x_1, s)$.

If distributionally one-way functions do not exist, then for any constant $c > 0$ there exists a probabilistic polynomial-time Turing machine $\mathcal{M}$ that on input (m, x_1, s) produces a distribution that is n^{-c}-statistically close to the uniform distribution on all the pre-images of (m, x_1, s) under f'. The polynomial-time adversary will use this $\mathcal{M}$ in the attack.

Let $\mathcal{A}$ denote the unbounded adversary that carried the attack described in Lemma 6.2, and let $\mathcal{A}^{\mathrm{PPT}}$ denote a polynomial-time adversary that carries the following attack (our goal is that the receiver will not be able to distinguish between $\mathcal{A}$ and $\mathcal{A}^{\mathrm{PPT}}$):

1. $\mathcal{A}^{\mathrm{PPT}}$ chooses $\widehat{m} \in_{\mathrm{R}} \{0,1\}^n$ and runs an honest execution with the receiver. Denote by s the manually authenticated string fixed by this execution.

2. $\mathcal{A}^{\mathrm{PPT}}$ chooses $m \in_{\mathrm{R}} \{0,1\}^n$ as the sender's input, and receives x_1 from the sender.

3. $\mathcal{A}^{\mathrm{PPT}}$ executes $\mathcal{M}$ on input (m, x_1, s), and then applies f to $\mathcal{M}$'s answer to compute $\widehat{x}_2$ and send it to the sender. The sender manually authenticates some string s^*.

4. $\mathcal{A}^{\mathrm{PPT}}$ forwards s^* to the receiver (who must accept $\widehat{m}$ if $s^* = s$ by the perfect completeness).

Let $\mathrm{Prob}^{\mathcal{R}}$ and $\mathrm{Prob}^{\mathrm{PPT},\mathcal{R}}$ denote the probabilities that the receiver $\mathcal{R}$ accepts $\widehat{m}$ when interacting with $\mathcal{A}$ and when interacting with $\mathcal{A}^{\mathrm{PPT}}$, respectively. Then, from the point of view of the receiver, the only difference in the these two executions is in the distribution of s^*. Therefore, $|\mathrm{Prob}^{\mathcal{R}} - \mathrm{Prob}^{\mathrm{PPT},\mathcal{R}}|$ is at most twice the statistical distance between s^* in the interaction with $\mathcal{A}$ and s^* in the interaction with $\mathcal{A}^{\mathrm{PPT}}$. By the above mentioned property of $\mathcal{M}$, this statistical difference is at most n^{-c}. Therefore, for sufficiently large t, we obtain as in Lemma 6.2

$$2\epsilon > \mathrm{Prob}^{\mathrm{PPT},\mathcal{R}} \geq \mathrm{Prob}^{\mathcal{R}} - 2n^{-c} \geq 2^{-\{\mathrm{I}(S;M,X_1)+\mathrm{H}(S|M,X_1,X_2)\}} - 2n^{-c} \ .$$

In particular, and since $n \geq 1/\epsilon$, we can choose the constant c such that $2n^{-c} < \epsilon$, and obtain

$$3\epsilon > 2^{-\{\mathrm{I}(S;M,X_1)+\mathrm{H}(S|M,X_1,X_2)\}} \ ,$$

which implies that $\mathrm{I}(S;M,X_1) + \mathrm{H}(S|M,X_1,X_2) > \log\frac{1}{\epsilon} - 2$. In the general case, where the number of rounds is some integer k, $\mathcal{A}^{\mathrm{PPT}}$ has to carry out $\frac{k-1}{2}$ inversions, therefore

$$2\epsilon > \mathrm{Prob}^{\mathrm{PPT},\mathcal{R}} \geq \mathrm{Prob}^{\mathcal{R}} - (k-1)n^{-c} \geq 2^{-\{\mathrm{I}(S;M,X_1)+\mathrm{H}(S|M,X_1,X_2)\}} - (k-1)n^{-c} \ .$$

However, since the number of rounds k is at most polynomial in n, and since $n \geq 1/\epsilon$, we can again choose the constant c such that $(k-1)n^{-c} < \epsilon$, and obtain the same result.

A similar argument applied to the attack described in Lemma 6.3 yields $\mathrm{I}(S;X_2|M,X_1) > \log\frac{1}{\epsilon} - 2$, and therefore $\mathrm{H}(S) > 2\log\frac{1}{\epsilon} - 4$. We note that when the protocol is not perfectly complete, a more careful analysis yields $\mathrm{H}(S) > 2\log\frac{1}{\epsilon} - 6$. ∎

Finally, we derive Theorem 4.6 from the above lemmata. Suppose that there exists a computationally secure $(n, \ell, k, \epsilon, t)$-authentication protocol Π, such that $n \geq \log(1/\epsilon) + 4$, $t = \Omega(\mathrm{poly}(n, k, 1/\epsilon))$ and $\ell < 2\log(1/\epsilon) - 8$. Then, Lemma 8.1 implies that there exists a $(n', \ell, k', \epsilon', t)$-authentication protocol Π', such that $n' = \lceil 1/\epsilon' \rceil$, $t = \Omega(\mathrm{poly}(n', k', 1/\epsilon'))$ and $\ell < 2\log(1/\epsilon') - 6$ (here, $\epsilon' = 2\epsilon$). Therefore, by Lemma 8.2, distributionally one-way functions exist, and as mentioned in Section 2, the existence of one-way functions is equivalent to the existence of distributionally one-way functions. This concludes the proof of Theorem 4.6.

Acknowledgments

We thank Benny Pinkas, Danny Segev, Serge Vaudenay and the anonymous referees for their remarks and suggestions.

References

[1] B. Barak. Constant-round coin-tossing with a man in the middle or realizing the shared random string model. In *Proceedings of the 43rd Symposium on Foundations of Computer Science*, pages 345–355, 2002.

[2] Bluetooth. http://www.bluetooth.com/bluetooth/.

[3] M. Cagalj, S. Capkun, and J. P. Hubaux. Key agreement in peer-to-peer wireless networks. *Proceedings of the IEEE*, 94(2):467–478, 2006.

[4] Certified Wireless USB. http://www.usb.org/developers/wusb/.

[5] G. Di Crescenzo, Y. Ishai, and R. Ostrovsky. Non-interactive and non-malleable commitment. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 141–150, 1998.

[6] G. Di Crescenzo, J. Katz, R. Ostrovsky, and A. Smith. Efficient and non-interactive non-malleable commitment. In *Advances in Cryptology - EUROCRYPT '01*, pages 40–59, 2001.

[7] D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. *SIAM Journal on Computing*, 30(2):391–437, 2000. A preliminary version appeared in *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, pages 542–552, 1991.

[8] C. Gehrmann. Cryptanalysis of the Gemmell and Naor multiround authentication protocol. In *Advances in Cryptology - CRYPTO '94*, pages 121–128, 1994.

[9] C. Gehrmann. Multiround unconditionally secure authentication. *Journal of Designs, Codes and Cryptography*, 15(1):67–86, 1998.

[10] C. Gehrmann, C. J. Mitchell, and K. Nyberg. Manual authentication for wireless devices. *RSA Cryptobytes*, 7:29–37, 2004.

[11] P. Gemmell and M. Naor. Codes for interactive authentication. In *Advances in Cryptology - CRYPTO '93*, pages 355–367, 1993.

[12] E. Gilbert, F. MacWilliams, and N. Sloane. Codes which detect deception. *Bell System Technical Journal*, 53(3):405–424, 1974.

[13] M. T. Goodrich, M. Sirivianos, J. Solis, G. Tsudik, and E. Uzun. Loud and Clear: Human-verifiable authentication based on audio. In *Proceedings of the 26th IEEE International Conference on Distributed Computing Systems*, page 10, 2006.

[14] J. H. Hoepman. The ephemeral pairing problem. In *Proceedings of the 8th International Financial Cryptography Conference*, pages 212–226, 2004.

[15] J. H. Hoepman. Ephemeral pairing on anonymous networks. In *Proceedings of the 2nd International Conference on Security in Pervasive Computing*, pages 101–116, 2005.

[16] R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. In *Proceedings of the 30th Annual Symposium on Foundations of Computer Science*, pages 230–235, 1989.

[17] R. Johannesson and A. Sgarro. Strengthening Simmons' bound on impersonation. *IEEE Transactions on Information Theory*, 37(4):1182–1185, 1991.

[18] S. Laur and K. Nyberg. Efficient mutual data authentication using manually authenticated strings. In *Proceeding of the 5th International Conference on Cryptology and Network Security*, pages 90–107, 2006. A preliminary version appeared as Report 2005/424 in the Cryptology ePrint Archive (with additional co-author N. Asokan).

[19] U. M. Maurer. Authentication theory and hypothesis testing. *IEEE Transactions on Information Theory*, 46(4):1350–1356, 2000.

[20] J. M. McCune, A. Perrig, and M. K. Reiter. Seeing-Is-Believing: Using camera phones for human-verifiable authentication. In *Proceedings of the 2005 IEEE Symposium on Security and Privacy*, pages 110–124, 2005.

[21] M. Naor and G. N. Rothblum. The complexity of online memory checking. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 573–584, 2005.

[22] S. Pasini and S. Vaudenay. An optimal non-interactive message authentication protocol. In *Topics in Cryptology - The Cryptographers' Track at the RSA Conference*, pages 280–294, 2006.

[23] R. Pass and A. Rosen. New and improved constructions of non-malleable cryptographic protocols. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 533–542, 2005.

[24] R. L. Rivest and A. Shamir. How to expose an eavesdropper. *Communications of the ACM*, 27(4):393–395, 1984.

[25] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A Transport Protocol for Real-Time Applications. RFC 3550 (Standard), 2003. http://www.ietf.org/rfc/rfc3550.txt.

[26] A. Sgarro. Informational divergence bounds for authentication codes. In *Advances in Cryptology - EUROCRYPT '89*, pages 93–101, 1989.

[27] A. Sgarro. Information-theoretic bounds for authentication frauds. *Journal of Computer Security*, 2:53–64, 1993.

[28] V. Shoup. New algorithms for finding irreducible polynomials over finite fields. *Mathematics of Computations*, 54:435–447, 1990.

[29] G. J. Simmons. Authentication theory/coding theory. In *Advances in Cryptology - CRYPTO '84*, pages 411–431, 1984.

[30] G. J. Simmons. The practice of authentication. In *Advances in Cryptology - EUROCRYPT '85*, pages 261–272, 1985.

[31] D. R. Stinson. Some constructions and bounds for authentication codes. *Journal of Cryptology*, 1(1):37–52, 1988.

[32] S. Vaudenay. Digital signature schemes with domain parameters: Yet another parameter issue in ECDSA. In *Proceedings of the 9th Australasian Conference on Information Security and Privacy*, pages 188–199, 2004.

[33] S. Vaudenay. Secure communications over insecure channels based on short authenticated strings. In *Advances in Cryptology - CRYPTO '05*, pages 309–326, 2005.

[34] M. N. Wegman and L. Carter. New hash functions and their use in authentication and set equality. *Journal of Computer and System Sciences*, 22(3):265–279, 1981.

[35] P. Zimmermann, A. Johnston, and J. Callas. ZRTP: Extensions to RTP for Diffie-Hellman key agreement for SRTP. Internet Draft, 2006. http://www.ietf.org/internet-drafts/draft-zimmermann-avt-zrtp-01.txt.