The Security of Lazy Users in Out-of-Band Authentication

Moni Naor* Lior Rotem† Gil Segev†

Abstract

Faced with the threats posed by man-in-the-middle attacks, messaging platforms rely on "out-of-band" authentication, assuming that users have access to an external channel for authenticating one short value. For example, assuming that users recognizing each other's voice can authenticate a short value, Telegram and WhatApp ask their users to compare 288-bit and 200-bit values, respectively. The existing protocols, however, do not take into account the plausible behavior of users who may be "lazy" and only compare parts of these values (rather than their entirety).

Motivated by such a security-critical user behavior, we study the security of lazy users in out-of-band authentication. We start by showing that both the protocol implemented by WhatsApp and the statistically-optimal protocol of Naor, Segev and Smith (CRYPTO '06) are completely vulnerable to man-in-the-middle attacks when the users consider only a half of the out-of-band authenticated value. In this light, we put forward a framework that captures the behavior and security of lazy users. Our notions of security consider both statistical security and computational security, and for each flavor we derive a lower bound on the tradeoff between the number of positions that are considered by the lazy users and the adversary's forgery probability.

Within our framework we then provide two authentication protocols. First, in the statistical setting, we present a transformation that converts any out-of-band authentication protocol into one that is secure even when executed by lazy users. Instantiating our transformation with a new refinement of the protocol of Naor et al. results in a protocol whose tradeoff essentially matches our lower bound in the statistical setting. Then, in the computational setting, we show that the computationally-optimal protocol of Vaudenay (CRYPTO '05) is secure even when executed by lazy users – and its tradeoff matches our lower bound in the computational setting.

*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. Supported in part by a grant from the Israel Science Foundation.

†School of Computer Science and Engineering, Hebrew University of Jerusalem, Jerusalem 91904, Israel. Email: {lior.rotem,segev}@cs.huji.ac.il. Supported by the European Union's Horizon 2020 Framework Program (H2020) via an ERC Grant (Grant No. 714253), by the Israel Science Foundation (Grant No. 483/13), by the Israeli Centers of Research Excellence (I-CORE) Program (Center No. 4/11), and by the US-Israel Binational Science Foundation (Grant No. 2014632).

Contents

1 Introduction

Instant messaging platforms are gaining increased popularity and hold an overall user base of more than 1.5 billion active users (e.g., WhatsApp, Signal, Telegram and many more [Wik]). These platforms recognize user authentication and end-to-end encryption as key ingredients for ensuring secure communication within them, and extensive efforts are currently put into the security of messaging, both commercially (e.g., [PM16, Telb, Wha, Vib]) and academically (e.g., [FMB+16, BSJ+17, CCD+17, KBB17]). A key challenge in securing messaging platforms is that of protecting against man-in-the-middle attacks when setting up secure end-to-end channels. This is exacerbated by the ad-hoc nature of these platforms.

Out-of-band authentication. Faced with the threats posed by man-in-the-middle attacks, existing messaging platforms enable "out-of-band" authentication, assuming that users have access to an *external* channel for authenticating short values. These values are typically derived from the public keys of the users, or more generally from the transcript of any key-exchange protocol that the users execute for setting up a secure end-to-end channel.

For example, some messaging platforms offer users the ability to compare with each other a value that is displayed by their devices (see Telegram [Tela], WhatsApp [Wha], Viber [Vib] and more [Mem17]). This relies on the assumption two users can establish a *low-bandwidth authenticated channel* (e.g., by recognizing each other's voice): A man-on-the-middle adversary can view, delay or even remove any message sent over this channel, but cannot undetectably modify its content.

Such an authentication model that assumes a low-bandwidth authenticated channel was considered back in 1984 by Rivest and Shamir [RS84].[1] More recently, this model was formalized by Vaudenay [Vau05] in the computational setting (i.e., considering computationally-bounded adversaries) and extended by Naor et al. [NSS06, NSS08] to the statistical setting (i.e., considering computationally-unbounded adversaries) and by Rotem and Segev [RS18] to the group setting. The out-of-band message authentication problem considers a sender that would like to authenticate a message m to a receiver.[2] The users communicate over two channels: An insecure channel over which a man-in-the-middle adversary has complete control, and a low-bandwidth authenticated channel, enabling the sender to "out-of-band" authenticate one short value. The security requirement asks for an upper bound on any man-in-the-middle adversary's probability of fooling the receiver into accepting a fraudulent message.

An effort vs. security tradeoff. Given that the out-of-band channel has only low bandwidth, research on out-of-band authentication has so far focused on constructing protocols that offer the best-possible tradeoff between the length of their out-of-band authenticated values (corresponding to the amount of effort required from the users) and their security (corresponding to the adversary's forgery probability). Vaudenay [Vau05], Naor et al. [NSS06] and Rotem and Segev [RS18] provided complete characterizations of this tradeoff in their above-mentioned respective settings, providing both lower bounds and protocols that match them. However, these protocols rely on the assumption that the *human users* indeed follow the protocol in its entirety. In particular, they rely on the assumption that the users out-of-band authenticate the *entire* value that the protocols instruct them to authenticate.

[1] Rivest and Shamir proposed the "Interlock" protocol which enables two users, who recognize each other's voice, to mutually authenticate their public keys in the absence of a trusted infrastructure. Potential attacks on the Interlock protocol were identified later on [BM94, Ell96].

[2] As mentioned above, for messaging platforms the message m typically corresponds to the public keys of the users or to the transcript of any key-exchange protocol that they execute.

This assumption, however, may not always be realistic: The lengths of the out-of-band authenticated values offered by the existing messaging platforms may not align with the potential effort of different users. Specifically, existing messaging platforms ask their users to out-of-band authenticate values whose lengths range from roughly 200 bits (e.g., WhatsApp and Signal) to 288 bits (e.g., Telegram) – see Figure 1. Given that the out-of-band channel in implemented in these platforms via a manual comparison operation, the security of such protocols must take into account users that may compare only a subset of the positions of these values. We refer to such users, who out-of-band authenticate only a substring of the protocol's out-of-band authenticated value, as "lazy users".

As repeatedly demonstrated by research on usable security and human-computer interaction, it is rather likely that a substantial part of the messaging platforms' user base may in fact be considered lazy (see, for example, [LS03, PLF03, BA04, Her09, HZF+14, AFJ15, DDB+16] and the references therein). This state of affairs, where a security-critical user behavior is not taken into account, is extremely bothering.

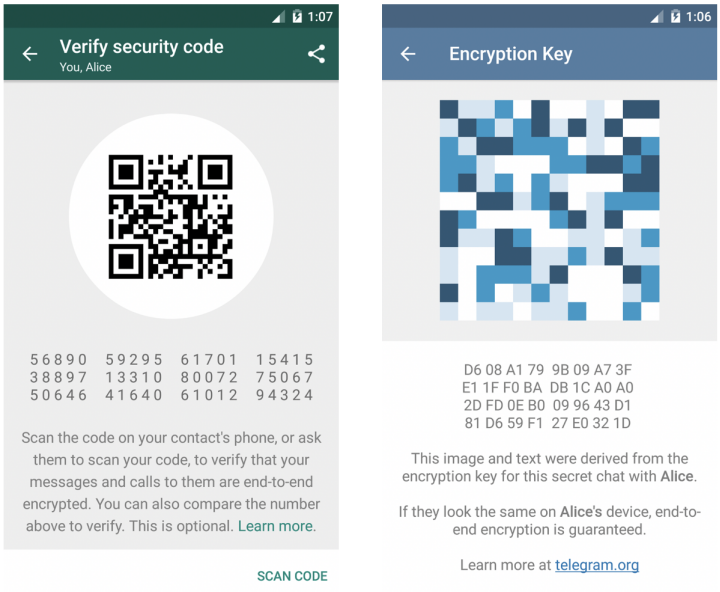

Figure 1: Out-of-band authentication in WhatsApp and Telegram. WhatApp and Telegram (as well as many other messaging platforms) implement the out-of-band channel by asking their users to manually compare two strings. WhatApp (on the right) asks its users to manually compare 60 decimal digits corresponding to an out-of-band authenticated value [Wha] of about 200 bits. Telegram (on the left) asks its users to manually compare 64 characters corresponding to a 288-bit out-of-band authenticated value [Telc]. The images are taken from [Mem17].

1.1 Our Contributions

Motivated by the above-described plausible and security-critical behavior of "lazy" users, we put forward a framework that captures the behavior and security of such users in out-of-band authentication. Within our framework we characterize the possible security guarantees for lazy users by presenting protocols together with essentially matching lower bounds both in the computational setting and in the statistical setting. Our main contributions are as follows.

The insecurity of existing protocols. We strengthen our motivation by showing that the protocol implemented by WhatsApp [Wha] and the protocol of Naor et al. [NSS06] are completely vulnerable to man-in-the-middle attacks when the parties consider only a half (or fewer) of the

characters of the out-of-band authenticated value. This demonstrates that it is not only the case that the existing protocols do not take security-critical user behavior into account, they may in fact become completely insecure when executed by lazy users. In the following section, we discuss the main underlying reason for these protocols' vulnerability, and how our constructions overcome it.

Modeling the behavior and security of lazy users. We put forward a framework that captures the behavior and security of lazy users. Our notions of security consider both computational security and statistical security, and for each flavor we derive a lower bound on the tradeoff between the number of positions that are considered by the lazy users out of the out-of-band authenticated value and the adversary's forgery probability. These lower bounds are summarized in Table 1, and we refer the reader to Section 1.3 for a more detailed overview.

	Our Protocols		**Our Lower Bounds**
	Forgery Probability	**Alphabet Size**	
Computational Security	$\mathbf{2^{-\lvert\mathcal{I}\rvert}}$	$\mathbf{2}$	$\mathbf{2^{-\lvert\mathcal{I}\rvert\cdot\log\lvert\Sigma\rvert} - 2^{-n}}$
Statistical Security	$\mathbf{2^{-\lvert\mathcal{I}\rvert}}$	$\mathbf{2^8}$	$\mathbf{2^{-\lvert\mathcal{I}\rvert\cdot\log\lvert\Sigma\rvert/2} - 2^{-n}}$

Table 1: Summary of our results – protocols vs. lower bounds. We denote by $\mathcal{I}$ the subset of positions of the out-of-band authenticated value that the users consider, by Σ the alphabet over which the out-of-band authenticated value is defined, and by n the length of the sender's input message. Our computationally-secure protocol relies on the existence of any one-way function (see Theorem 6.1), whereas our statistically-secure protocol and our two lower bounds do not rely on any computational assumptions (see Corollary 5.2, Theorem 7.1 and Corollary 7.3).
Note that our upper bound and lower bound in the computational setting match within an additive 2^{-n} term (which is a significantly lower-order term for not-too-short input messages). In the statistical setting our bounds match within a constant factor (in addition to the additive 2^{-n} term).

Immunizing statistically-secure protocols against lazy users. Recall that the statistically-secure protocol of Naor et al. [NSS06] becomes completely insecure when executed by lazy users. Intuitively, this is the case because the influence of each bit of the sender's input message (i.e., the message to be authenticated) is not "well-spread" across the out-of-band authenticated value (see Section 4 for an in-depth discussion).

Addressing this property, we provide a transformation that converts any statistically-secure protocol (that does not necessarily provide any security for lazy users) into a protocol that is statistically-secure for lazy users. Instantiating our transformation with the protocol of Naor et al. results in a concrete statistically-secure protocol for lazy users. Moreover, we then show that by refining the protocol of Naor et al. the resulting instantiation uses an alphabet whose size is as small as 2^8 – which nearly matches our above-mentioned lower bound in the statistical setting.[3] We stress that our transformation and the protocol resulted from applying it to the protocol of Naor et al. are oblivious to the subset $\mathcal{I}$ of positions that users eventually read or even to the number of positions they read. Meaning, we provide a *single* protocol that guarantees security for every possible subset $\mathcal{I}$. An interesting open question is whether a protocol which is statistically-secure for lazy users can be constructed over a binary alphabet.

[3]As we discuss in more detail in Section 1.3, when moving to the setting of lazy users, the size of the alphabet over which the out-of-band authenticated value is defined becomes of great importance. This is in contrast to the traditional (non-lazy) setting, in which this has no impact on security.

In fact, our transformation can also be applied to any computationally-secure protocol that satisfies a natural parallel composability guarantee. However, as shown by our next result, this is somewhat unnecessary.

Matching the optimal tradeoff for computationally-secure protocols. Whereas the statistically-optimal protocol of Naor et al. is completely insecure for lazy users, we show that the computationally-optimal protocol of Vaudenay [Vau05] is optimally secure for lazy users as well. Intuitively, this is due to the following observation: Even though the out-of-band authenticated value in this protocol is determined independently of the sender's input message (which is reminiscent of the protocol of Naor et al. in the statistical setting), the protocol "ties together" the message and the out-of-band authenticated value *in their entirety* using a non-malleable commitment scheme (which, in practice, can be replaced by a hash function modeled as a random oracle). Note that as in the statistical setting, the protocol is oblivious to the particular subset of positions that the users eventually consider.

Extensions. We also discuss possible extensions of our framework. First, we consider the notion of *adaptive laziness*, which gives the adversary the ability to choose the subset of positions to be considered by the users even *after* the out-of-band authenticated value is determined. Although we find this notion somewhat less motivated in the context of lazy users, we nevertheless extend our definitions and proofs of security to this stronger notion.

Second, we note that our notions of security, lower bounds and protocols naturally extend to the group setting considered by Rotem and Segev [RS18]. Specifically, in the computational setting the protocol of Rotem and Segev can be shown to be optimally-secure for lazy users; and in the statistical setting, our general transformation can be easily adapted to support group protocols (and can then be instantiated with the statistically-secure protocol of Rotem and Segev).

1.2 Related Work

Bounds for out-of-band authentication. In the standard setting of out-of-band authentication (i.e., with non-lazy users), Vaudenay [Vau05] and Vaudenay and Pasini [PV06] established tight bounds for the tradeoff between the length of the (entire) out-of-band authenticated value and the adversary's forgery probability in the computational setting. They provided a protocol [Vau05] in which the forgery probability is bounded by $2^{-\ell}$, where ℓ is the bit-length of the out-of-band authenticated value, and a matching lower bound [PV06]. Naor et al. [NSS06] observed a gap between the computational and the statistical settings: They proved that the forgery probability in the statistical setting of any protocol is always at least $2^{-\ell/2}$, and provided a protocol that matches this lower bound within a constant factor. We refer the reader to Table 2 for a summary of these bounds, and note that our results provide a similar characterization for lazy users in both the computational and the statistical settings (recall Table 1).

The security of messaging platforms. Many recent works addressed the goals of formalizing the security guarantees of messaging platforms, as well as analyzing the security of the protocols used by these platforms and identifying potential weaknesses within them – see, for example, [FMB$^+$16, HL16, BSJ$^+$17, CCD$^+$17, CGCG$^+$17, CGC17, KBB17, SKH17, RMS18, Gre18a, Gre18b] and the references therein. Throughout this extensive line of research, the security of messaging protocols assumes an initial authentication phase for avoiding man-in-the-middle attacks. As mentioned in most of the afore-listed references, such an initial authentication phase is based on out-of-band authentication.

	Protocols	Lower Bounds
Computational Security [Vau05, PV06]	$\mathbf{2^{-\ell}}$	$\mathbf{2^{-\ell} - 2^{-n}}$
Statistical Security [NSS06]	$\boldsymbol{O\left(2^{-\ell/2}\right)}$	$\mathbf{2^{-\ell/2} - 2^{-n}}$

Table 2: Previous work – protocols vs. lower bounds. We denote by ℓ the length of the out-of-band authenticated value and by n the length of the sender's input message. The computationally-secure protocol of Vaudenay [Vau05] relies on the existence of any one-way function, whereas the statistically-secure protocol of Naor et al. [NSS06] and the two lower bounds [NSS06, PV06] do not rely on any computational assumptions.

1.3 Overview of Our Contributions

We extend the existing framework for out-of-band authentication protocols [Vau05, PV06, NSS06, RS18] to accommodate the security-critical behavior of "lazy users", that may consider only a certain part of the out-of-band authenticated value (e.g., its left-most half, its right-most 10 characters, or a few randomly-chosen positions). We model this behavior by having the sender send only a substring of the out-of-band authenticated value, and requiring that for any such substring the man-in-the-middle attacker's forgery probability is bounded by some pre-defined parameter associated with it. That is, whereas a standard (i.e., "non-lazy") out-of-band authentication protocol is parameterized by an upper bound $\epsilon \in (0,1)$ on the adversary's forgery probability, a protocol in our framework is parameterized by a function $\epsilon(\cdot)$ which maps every subset $\mathcal{I}$ of positions of the out-of-band authenticated value to an associated upper bound $\epsilon(\mathcal{I})$.[4]

In addition, our definitions also extend those of Vaudenay and Naor et al. by accounting for out-of-band authentication values over *non-binary* alphabets (indeed, in the existing real-world implementations of out-of-band authentication protocols, the out-of-band authenticated value is displayed to the users as a string over some non-binary alphabet – recall Figure 1). When the users are assumed to consider the entire out-of-band authenticated value, the particular choice of alphabet (and alphabet size) is mainly a matter of providing a convenient user interface. In the presence of lazy users, however, the size of the alphabet of the out-of-band authenticated value plays an important role in what may be referred to as the "granularity" of the users' laziness.

Let us consider for concreteness a pair of users that read some 32 bits out of a 64-bit out-of-band authenticated value. If the out-of-band authenticated value is simply a 64-bit string (i.e., over a binary alphabet), then the users may possibly read any of the $\binom{64}{32} > 1.83 \times 10^{18}$ many 32-bit substrings of it. On the other hand, if the alphabet is of larger size, say 8 characters, the users' ability to partially access the out-of-band authenticated value is more coarse-grained. In particular, they can still read only a substring of the authenticated value, but are restricted to reading specific blocks of consecutive 8 bits in their entirety. In other words, users that read 32 bits in this setting may read only one of $\binom{8}{4} = 70$ many 32-bit substrings of the out-of-band authenticated value.

Identifying the weakness in existing protocols. It is quite simple to construct a contrived example of a secure protocol that is completely insecure when executed by lazy users. Thus, we chose to focus on the protocols of WhatsApp [Wha] and Naor et al. [NSS06] for the following reasons: (1) the protocol implemented by WhatsApp is among the most widely-used out-of-band authentication

[4]Note that protocols in our framework must explicitly address (in terms of both completeness and soundness) the case where only part of the out-of-band authenticated value is considered. This is the case, in particular, in our motivating example where verification is done by comparing the out-of-band authenticated string to a value that is computed by the receiver.

protocols, and (2) the protocol of Naor et al. offers the optimal tradeoff between the length of the out-of-band authenticated value and the adversary's forgery probability in the statistical setting (thus showing that both computationally-secure protocols and statistically-secure ones may become completely insecure when executed by lazy users).

Analyzing our rather simple attacks on these protocols (see Section 4), we identify a key property that they have in common which makes them completely insecure when executed by lazy users: Intuitively, different sections of the sender input message (i.e, the message m to be authenticated) influence different sections of the out-of-band authenticated value. Hence, if the users only consider a subset of positions of the out-of-band authenticated value that is independent in some sense from a particular part of the message to be authenticated, the adversary can replace this part of the message in an undetected manner (we refer to this property as "over locality"). In what follows, we discuss why the protocol of Vaudenay in the computational setting does not suffer from over locality; and how our general transformation in the statistical setting addresses it.

Naive approaches that fail. A potential approach to immunizing any comparison-based out-of-band authentication protocol against lazy users, is to have the parties run the protocol and then hash the out-of-band authenticated value with a random oracle (in addition to transmitting it over the insecure channel). On the face of it, this resolves any over dependency on locality the initial protocol might have exhibited. However, this approach may generally suffer from the major shortcoming of introducing a tradeoff between the adversary's running time and its success probability (aside, of course, from relying on a random oracle which may be undesirable if the security of the underlying protocol does not require it). More concretely, an adversary that runs in time $T(\lambda)$ has forgery probability that is roughly (at least) $T(\lambda)/2^{-|\mathcal{I}|}$, where $\mathcal{I}$ is the subset of positions that the parties consider. When $\mathcal{I}$ is small (which is exactly the case with lazy users), then the asymptotics "do not kick in", and the latter forgery probability is significant. This is precisely the reason why we are interested in protocols in which for every such subset $\mathcal{I}$, the forgery probability is bounded by $\epsilon(\mathcal{I})+\nu(\lambda)$ (where $\nu(\cdot)$ is a negligible function of the security parameter λ) *for every* polynomial-time adversary.

An additional potential approach is to have the parties apply some fixed error-correcting code to the out-of-band authenticated value. Though this may have the effect of increasing the fraction of inconsistent positions in the out-of-band authenticated value at the end of any forgery attempt, it does not provide the security guarantees we seek: If before applying the error-correcting code there was some subset of t positions for some fixed t, for which there was an attack causing the receiver to output a fraudulent message with probability ϵ, this may still be the case after applying the code. Moreover, this approach has the consequence of worsening the tradeoff between the length of the out-of-band authenticated value and the adversary's forgery probability. Similarly, adding redundancy to the input message itself (e.g., by applying an error-correcting code to it) is not necessarily helpful in immunizing protocols against lazy users.

Another possibility is to reduce the number of characters in the out-of-band authenticated value by mapping it to a larger alphabet. As discussed above, this has the effect of restricting the lazy behavior of the users; in particular, assuming that the users read at least one character of the out-of-band value, after increasing the alphabet size, this single character constitutes a larger fraction of the out-of-band value. Alas, even if the new alphabet is sufficiently large so that the out-of-band value consists just of two characters, the resulting protocol may still be insecure for lazy users who read only one of them (this is the case, for example, with the protocols of WhatsApp [Wha] and Naor et al. [NSS06]). On the other hand, our lower bounds on the bit-length of the out-of-band value (see Section 7) imply that in order for the out-of-band value to consists only of a single character,

its alphabet size has to be at least $1/\epsilon$, where ϵ is the forgery probability. For any reasonable level of security, this means an impractical-sized alphabet has to be used.

Security for lazy users via "influence spreading". Our transformation in the statistical setting takes as input a parameter $t \in \mathbb{N}$ and any statistically-secure out-of-band authentication π with out-of-band authenticated value of length ℓ and forgery probability at most ϵ. It proceeds by having the sender S and the receiver R run t parallel executions of π with the same input message m to S. Afterwards, S parses each of the resulting t out-of-band authentication values as a single character from an alphabet of the appropriate size, concatenates them into a single string of length t (over the larger alphabet) and sends it over the out-of-band channel. When considering some subset $\mathcal{I} \subseteq [t]$ of the characters in the new out-of-band authenticated value, the receiver R accepts the message m if and only if it accepts m in each of the executions corresponding to the subset $\mathcal{I}$. We show that for every subset $\mathcal{I} \subseteq [t]$, the forgery probability in this new protocol is bounded by $\epsilon'(\mathcal{I}) \leq \epsilon^{|\mathcal{I}|}$.

In light of our observations regarding protocols that are insecure for lazy users, this transformation can be thought of in the following manner: We start with a protocol that might be insecure for lazy users and suffer from over locality, and we "spread" the influence of each bit of the input message across all characters of the new out-of-band authenticated value via the parallel invocations of the basic protocol.

When instantiated with the protocol of Naor et al. [NSS06] (while setting its security to $\epsilon = 1/2$), our transformation yields a protocol with a constant-size alphabet which is statistically-secure for lazy users: For every subset $\mathcal{I} \subseteq [t]$, the forgery probability corresponding to $\mathcal{I}$ is bounded by $2^{-|\mathcal{I}|}$. However, using the protocol of Naor et al. and their analysis "off the shelf" results in an alphabet which is, though constant-size, large and impractical (concretely, it is of size $2^{16} = 65536$). Hence, in Section 8.2, we show by a refined analysis of the protocol of Naor et al. that this constant can be reduced to $2^8 = 256$ (which fits nicely, for example, in the set of 333 emoji Telegram uses as the alphabet in the verification of their voice calls).

Leveraging the "local sensitivity" of non-malleable commitments. Informally speaking, the protocol of Vaudenay [Vau05] consists of the following steps: (1) On input m, S sends m to R, chooses a random r_S and commits to the message (m, r_S); (2) R sends a random r_R to S; (3) S reveals r_S; and (4) S sends $r_S \oplus r_R$ over the out-of-band authenticated channel. In the lazy user setting, where the users only read the subset $\mathcal{I}$ of positions in the out-of-band authenticated value, R accepts m if and only if the value $(r_S \oplus r_R)_{\mathcal{I}}$ sent over the out-of-band channel is consistent with her view of the protocol.

In Section 6 we prove that when the commitment scheme used in Step (1) is a non-malleable commitment scheme, then this protocol is optimal for lazy users (considering the matching lower bound from Section 7). Our proof goes about by considering all potential synchronizations that a man-in-the middle attacker might impose while attacking an execution of the protocol, and showing that in each of them, an attack on the protocol that succeeds with probability noticeably larger than $2^{-|\mathcal{I}|}$ can be translated into an attack on a different property of the underlying commitment scheme.

From a more conceptual point of view, our proof leverages the fact that the non-malleability of commitment schemes is a property which is "locally sensitive" in the following sense. Informally, in a non-malleable commitment scheme, it should be impossible, given a commitment c to some value v, to produce a related commitment $\widehat{c}$ for some value $\widehat{v}$ such that v and $\widehat{v}$ satisfy *any* efficiently recognizable relation. This includes, in particular, relations that are defined with respect to a subset of the positions in v and $\widehat{v}$; and namely, the relation induced by a successful forgery in Vaudenay's protocol when the users only consider the subset $\mathcal{I}$ of positions of the out-of-band authenticated

value.

1.4 Paper Organization

The remainder of this paper is organized as follows. In Section 2 we present the notation and basic definitions that are used in this work. In Section 3 we introduce our framework for modeling the behavior and security of lazy users in out-of-band message authentication protocols. In Section 4 we show that existing out-of-band authentication protocols may become completely insecure when executed by lazy users. In Sections 5 and 6 we present statistically-secure and computationally-secure out-of-band authentication protocols, respectively. In Section 7 we derive lower bounds on the tradeoff between the adversary's forgery probability and the length of the out-of-band authenticated value in out-of-band authentication protocols that are executed by lazy users. Finally, in Section 8 we discuss several extensions of our framework and results.

2 Preliminaries

In this section we present the notation and basic definitions that are used in this work. For a distribution X we denote by $x \leftarrow X$ the process of sampling a value x from the distribution X. Similarly, for a set $\mathcal{X}$ we denote by $x \leftarrow \mathcal{X}$ the process of sampling a value x from the uniform distribution over $\mathcal{X}$. For an integer $n \in \mathbb{N}$ we denote by $[n]$ the set $\{1, \ldots, n\}$. For a string s and a subset $\mathcal{I} \subseteq [|s|]$ of positions, we let $s_{\mathcal{I}}$ (sometimes we may write $(s)_{\mathcal{I}}$) denote the substring of s obtained by concatenating the characters of s in the positions specified by the set $\mathcal{I}$ in increasing order. A function $\nu : \mathbb{N} \to \mathbb{R}^+$ is *negligible* if for any polynomial $p(\cdot)$ there exists an integer N such that for all $n > N$ it holds that $\nu(n) \leq 1/p(n)$.

Shannon entropy. For a random variable X defined over a finite domain Ω, we rely the standard notion of Shannon entropy defined as $\mathrm{H}(X) = -\sum_{x \in \Omega} \Pr[X = x] \cdot \log_2 \Pr[X = x]$. Note that for any such X it holds that $\mathrm{H}(X) \leq \log_2 |\Omega|$.

Non-malleable commitment schemes [DDN00]. We rely on the notion of statistically-binding non-malleable commitments (for basic definitions and background on commitment schemes, we refer the reader to [Gol01]). We follow the indistinguishability-based definition of Lin and Pass [LP11], though we find it convenient to consider non-malleability with respect to content, other than with respect to identities. Intuitively speaking, a non-malleable commitment scheme has the following guarantee: Any efficient adversary cannot use a commitment to some value v in order to produce a commitment to a value $\widehat{v}$ which is "non-trivially" related to v. For formal definitions regarding commitment schemes and non-malleable commitment schemes in particular, see Appendix A.

Dolev et al. [DDN00] constructed non-malleable commitment schemes from any one-way function. Subsequently, Lin and Pass [LP11] and Goyal [Goy11] have shown that constant-round non-malleable commitments can be constructed from the same assumption. The round complexity was further improved by Goyal et al. [GRR+14] to 4 rounds, and by Goyal et al. [GPR16] to 3 rounds assuming the existence of an injective one-way function. Such schemes can also be constructed efficiently in a simple manner in the random-oracle model [BR93]. For further information regarding non-malleable commitment schemes in the standard model see the references above as well as, for example, [Bar02, PR08, LP09, PPV08, PW10, Wee10, GLO+12] and the references therein.

3 Modeling the Security of Lazy Users

In this section we introduce our framework for modeling the behavior and security of lazy users in out-of-band message authentication protocols. We start by reviewing the communication model and existing notions of security for out-of-band message authentication [Vau05, NSS06], and then present our notions of security for the case of lazy users.

3.1 Out-of-Band Authentication

Following the framework of Vaudenay [Vau05] and Naor et al. [NSS06], we model the interaction between the sender and the receiver as occurring over two types of channels: A bidirectional insecure channel that is completely vulnerable to man-in-the middle attacks, and an authenticated unidirectional low-bandwidth channel from the sender to the receiver. The adversary is assumed to have complete control over the insecure channel: She can read, delay and remove any messages sent by the two parties, as well as insert new messages of her choice at any point in time. In particular, this provides the adversary with considerable control over the synchronization of the protocol's execution. Nonetheless, the execution is still guaranteed to be "marginally synchronized": Each party sends her message in the ith round of the protocol only upon receiving the due message of round $i-1$. As for the out-of-band channel, we assume that the sender is equipped with a low-bandwidth channel, through which the sender may send a short message to the receiver in an authenticated manner (but without any secrecy guarantee). The adversary may read or remove this message, and may delay it for different periods of time, but cannot modify it in an undetectable manner.

We follow the definitions of Vaudenay [Vau05] and Naor et al. [NSS06], generalizing naturally to consider out-of-band authenticated values over general alphabets and not only over the binary alphabet. As we discuss later on, this is of little importance in the standard setting (where the parties are assumed to read the entire out-of-band authenticated value), but will play a significant role when considering lazy users. Following Naor et al. we differentiate between protocols that are computationally secure and ones that are statistically secure. We formalize the notion of *statistically-secure* out-of-band authentication protocols as:

Definition 3.1. Let $n, \ell, r \in \mathbb{N}$, let $\epsilon \in (0,1)$ and let Σ be an alphabet. A statistically-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol over Σ is an r-round protocol in which the sender S is invoked on an n-bit message and sends at most ℓ characters of Σ over the out-of-band authenticated channel. The following requirements must hold:

1. **Correctness:** In an honest execution of the protocol, for any input message $m \in \{0,1\}^n$ on which S is invoked, R outputs m with probability 1.
2. **Unforgeability:** For any man-in-the-middle adversary A and for any adversarially-chosen input message $m \in \{0,1\}^n$ on which S is invoked, the probability that R outputs some message $\widehat{m} \notin \{m, \perp\}$ in an execution with S that is attacked by A is at most ϵ.

A *computationally-secure* out-of-band authentication protocol is defined similarly, except that security need only hold against efficient adversaries, and the probability of forgery is also allowed to additively grow (with respect to the statistical setting) by a negligible function of the security parameter.

Definition 3.2. Let $n = n(\lambda), \ell = \ell(\lambda), r = r(\lambda), \epsilon = \epsilon(\lambda)$, and $\Sigma = \Sigma(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. A computationally-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol over alphabet Σ is an r-round protocol in which the sender S is invoked on an n-bit message and sends at most ℓ characters of Σ over the out-of-band authenticated channel. The following requirements must hold:

1. **Correctness:** In an honest execution of the protocol, for any input message $m \in \{0,1\}^n$ on which S is invoked, R outputs m with probability 1.
2. **Unforgeability:** For any probabilistic polynomial-time man-in-the-middle adversary A there exists a negligible function $\nu(\cdot)$ such that: For any input message $m \in \{0,1\}^n$ chosen by the adversary and on which S is invoked, the probability that R outputs some message $\widehat{m} \notin \{m, \perp\}$ in an execution with S that is attacked by A is at most $\epsilon + \nu(\lambda)$.

3.2 The Security of Lazy Users

In order to formally capture the lazy-users setting, given an out-of-band authentication protocol we define a collection of "lazy protocols", one per each possible subset of positions of the out-of-band authenticated value. Informally speaking, given a protocol π in which the out-of-band authenticated value consists of ℓ characters, for a subset $\mathcal{I} \subseteq [\ell]$ of indexes, we consider the "lazy protocol" $\pi_{\mathcal{I}}$ in which the parties execute π, with the exception that S only sends over the out-of-band channel the substring of the out-of-band authenticated value that corresponds to the positions in the set $\mathcal{I}$.

Specifically, let π be a (statistically-secure or computationally-secure) out-of-band (n, ℓ, r, ϵ)-authentication protocol over an alphabet Σ (recall Definitions 3.1 and 3.2). For every subset $\mathcal{I} \subseteq [\ell]$ of the positions of its out-of-band authenticated value, the "lazy protocol" $\pi_{\mathcal{I}}$ is defined as follows:

1. On input $m \in \{0,1\}^n$ to S, the sender S and receiver R run the first $r-1$ rounds of π. Let $v \in \Sigma^\ell$ be the out-of-band authenticated value that S is due to send in round r.
2. S receives $\mathcal{I}$ and sends only $v_{\mathcal{I}}$ over the out-of-band authenticated channel.
3. R receives $\mathcal{I}$ and $v_{\mathcal{I}}$, and decides on her output according to π.[5]

Using this notion, Definitions 3.3 and 3.4 below formalize the extensions discussed above in the statistical setting and computational setting, respectively. Intuitively, we define the security of out-of-band authentication protocols for lazy users by letting the bound on the forgery probability be a function of the subset $\mathcal{I}$ considered by the users. Concretely, an out-of-band authentication protocol π is parameterized by some function ϵ, which maps each possible set of positions $\mathcal{I}$ of the out-of-band authenticated value to be read by the users to a matching upper bound on the forgery probability. That is, in case the users only read the out-of-band authentication value in positions $\mathcal{I}$, an adversary should be able to make the receiver output a fraudulent message with probability at most $\epsilon(\mathcal{I})$. This approach has the benefit of being very general on the one hand, while coinciding with the standard definitions (see Definitions 3.1 and 3.2) when $\mathcal{I} = [\ell]$. We note, however, that one may still consider a more restrictive notion where the forgery probability should only depend on the size of $\mathcal{I}$ (observe that this is a strict restriction of our notion).

Definition 3.3. Let $n, \ell, r \in \mathbb{N}$ and let $\epsilon : 2^{[\ell]} \to [0,1]$. A protocol π is *a statistically-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol for lazy users* over alphabet Σ if for every $\mathcal{I} \subseteq [\ell]$ the protocol $\pi_{\mathcal{I}}$ is a statistically-secure out-of-band $(n, |\mathcal{I}|, r, \epsilon(\mathcal{I}))$-authentication protocol.

Definition 3.4. Let $n = n(\lambda), \ell = \ell(\lambda), r = r(\lambda)$ and $\Sigma = \Sigma(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$, and let $\epsilon = \epsilon(\lambda, \cdot) : 2^{[\ell]} \to [0,1]$. A protocol π is *a computationally-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol for lazy users* over alphabet Σ if for every $\mathcal{I} = \mathcal{I}(\lambda) \subseteq [\ell]$ the protocol $\pi_{\mathcal{I}}$ is a computationally-secure out-of-band $(n, |\mathcal{I}|, r, \epsilon(\cdot, \mathcal{I}))$-authentication protocol.

[5] As noted before, the protocols we consider in this paper must be defined for every substring of the out-of-band authenticated value.

4 The Insecurity of Existing Protocols

In this section we show that existing out-of-band authentication protocols may become completely insecure when executed by lazy users. We focus on the computationally-secure protocol implemented by WhatsApp [Wha] and on the statistically-secure protocol of Naor et al. [NSS06], and show that these protocols are completely vulnerable to man-in-the-middle attacks when the parties consider only a half (or less) of the out-of-band authenticated value.

Concretely, for each of these two protocols we present an efficient man-in-the-middle attacker that fools the receiver into accepting a fraudulent message with probability 1. Then, we discuss the basic underlying structure that these two protocols share, which makes them completely insecure when executed by lazy users.

WhatsApp's protocol [Wha]. Consider any protocol where in order to authenticate a message m, the sender S partitions m into two halves $m = m_1 \| m_2$, and authenticates each half using some out-of-band authentication protocol separately and independently. The out-of-band authenticated value is then $\sigma = \sigma_1 \| \sigma_2$, where σ_1 and σ_2 are the out-of-band authenticated values of the two executions. If the underlying out-of-band authentication protocol is secure and the users read the entire string σ, then this newly-defined protocol is secure as well (though, possibly, with a sub-optimal tradeoff between the adversary's forgery probability and the length of the out-of-band authenticated value). However, consider for example the case where the parties only read σ_1 (or a substring of it). In this case, no security is guaranteed and a man-in-the-middle adversary can trivially make R output a fraudulent message of the form $\widehat{m} = m_1 \| \widehat{m_2}$ for some $\widehat{m_2} \neq m_2$. A similar problem arises when the parties read only σ_2 (or a substring of it).

The above protocol might seem like a pathological example, specifically contrived for our needs, but this is in fact exactly the approach used by WhatsApp. Concretely, a pair of WhatsApp users wishing to verify that each of them has the correct key of the other user compare a 60-digit sequence displayed on each of their screens. This sequence is derived by hashing each user's key into a 30-digit string, and concatenating the two strings.[6] It is not hard to see that if the users only compare the first half of the out-of-band authenticated value, it might very well be the case that one of them holds a fraudulent key, completely compromising the secrecy of their chat.

The protocol of Naor et al. [NSS06]. Naor et al. [NSS06] presented a construction of a statistically-secure out-of-band authentication protocol that relies on the following idea. Loosely speaking, the two parties iteratively hash the message into shorter intermediate values until reaching a short enough value that can be transmitted out-of-band. More concretely, in each round of the protocol the parties cooperatively choose an algebraic hash function: They treat the input message and the intermediate values as polynomials over finite fields of appropriate sizes, and in each round, one party chooses a random element in the field on which the polynomial is evaluated, and the other party chooses a random shift to apply to the result. When choosing the last hash function, the sender S is the one to choose the element on which the polynomial is evaluated. The out-of-band authenticated value then consists of two parts: (1) The result of the last hash function (according to the view of S); (2) and the last element S chose.

[6] From WhatsApp's security white paper [Wha, p. 10]: "WhatsApp users additionally have the option to verify the keys of the other users with whom they are communicating so that they are able to confirm that an unauthorized third party (or WhatsApp) has not initiated a man-in-the-middle attack. This can be done by scanning a QR code, or by comparing a 60-digit number. [...] The 60-digit number is computed by concatenating the two 30-digit numeric fingerprints for each user's Identity Key".

Yet again, if the parties read and compare the entire out-of-band authenticated value, then Naor et al. proved that this protocol is secure (and provides the optimal tradeoff between the adversary's forgery probability and the length of the out-of-band authenticated value). Alas, if the users are lazy, and read only one of the two parts of the out-of-band authenticated value, then the protocol becomes completely insecure. Concretely, if the parties only read the part that corresponds to the last field element chosen by S, then a trivial attack exists: The man-in-the-middle adversary simply runs two independent executions, one with the sender S and one with the receiver R, on two different input messages, with the exception of choosing the same field element as S does in the last hash function of her interaction with R.

Summary: The underlying weakness. The property that both of the above examples share and which makes them completely insecure in the face of rather trivial attacks can be articulated in the following manner: In both cases, different sections of the input message to be authenticated affect different sections of the out-of-band authenticated value. In the case of WhatsApp, each user's key affects only half of the out-of-band authentication value (but both keys should be verified). In the case of Naor et al. [NSS06], the input message to be authenticated goes into the computation of only half of the out-of-band authenticated value, while the other half is simply a random value generated during the execution of the protocol.

It is instructive to view our positive results also in this light, as this may provide the reader with additional intuition regarding the security of our constructions:

1. In the statistical setting, our transformation (and its resulting protocol when instantiated with that of Naor et al. [NSS06]) can be interpreted as follows. We start with an out-of-band authentication protocol that guarantees no security for lazy users to begin with (but does guarantee security for users who fully comply with the protocol), and in particular may suffer from the same problematic property described above. We transform this protocol into a protocol that provides security for lazy users by "spreading" the influence of each bit of the input message m across all characters of the out-of-band authenticated value of the resulting protocol.

2. In the computational setting we consider Vaudenay's protocol [Vau05] whose out-of-band authenticated value is simply a uniformly-distributed string that is generated during the execution of the protocol. Intuitively speaking, even though this value is determined independently of the input message, we "tie together" the message *in its entirety* and the out-of-band authenticated value using cryptographic tools (namely, a non-malleable commitment scheme).

5 Immunizing Statistically-Secure Protocols Against Lazy Users

In this section we present a generic transformation that uses any out-of-band authentication protocol that is secure under a certain form of parallel repetition for constructing an out-of-band authentication protocol for lazy users. In particular, our transformation can be applied to any statistically-secure protocol, and can thus be instantiated with the protocol of Naor et al. [NSS06]. As our transformation itself is statistically secure, this yields a statistically-secure protocol (that comes very close to matching our lower bound on the tradeoff between adversary's forgery probability and the length of the partial out-of-band authenticated value considered by the lazy users – see Corollary 7.3).

We first present and analyze our transformation for statistically-secure protocols, as well as discuss the properties of its instantiation with the protocol of Naor et al. [NSS06]. Then, we discuss

the specific composability property required of computationally-secure protocols in order for them to be compatible with our transformation (this, however, is somewhat less motivated given that our computationally-secure protocol in Section 6 already matches our lower bound in the computational setting).

The transformation. The building block underlying our transformation is an out-of-band authentication protocol that does not necessarily guarantee any form of security for lazy users. Loosely speaking, our transformation proceeds as follows: On input message m, the parties run ℓ parallel and independent executions of the underlying protocol with the same message m, and parse each of the resulting ℓ out-of-band authentication values as a single character from an alphabet of the appropriate size. The sender S then concatenates these ℓ characters into a single string of length ℓ (over the larger alphabet) and sends it over the out-of-band authenticated channel. In a lazy execution of the protocol, where the receiver considers only some number $t \leq \ell$ out of the ℓ out-of-band authenticated characters, the receiver accepts m if and only if it m is accepted in each of the corresponding t executions.

Intuitively, if the forgery probability of the underlying protocol is bounded by ϵ', then fooling a receiver that reads only a predetermined t-character subset of the out-of-band authenticated value requires the adversary to break the unforgeability (in the standard sense, not considering lazy users) of t copies of the underlying protocol, and hence the adversary's forgery probability is bounded by $(\epsilon')^t$ in the statistical setting.

More formally, let $n', \ell', r' \in \mathbb{N}$, let $\epsilon' \in (0,1)$, and let π' is a statistically-secure out-of-band $(n', \ell', r', \epsilon')$-authentication protocol; that is, π' is an r'-round protocol for out-of-band authentication of messages of length n', where the sender out-of-band authenticates at most ℓ' bits, and the probability of forgery is bounded by ϵ'. We use π' to construct a statistically-secure out-of-band $(n = n', \ell, r = r', \epsilon)$-authentication protocol for lazy users, denoted π_{Lazy}, for any $\ell \in \mathbb{N}$, such that $\epsilon(\mathcal{I}) = (\epsilon')^{|\mathcal{I}|}$ for every $\mathcal{I} \subseteq [\ell]$.

The protocol for lazy users, denoted π_{Lazy}, is defined as follows for every $\mathcal{I} \subseteq [\ell]$ (i.e., this is the "lazy protocol" $\pi_{\mathsf{Lazy},\mathcal{I}}$ – see Section 3):

1. On input message m to S, S and R run ℓ parallel executions of π' up to (and including) round $r' - 1$ with the same input message m to S in all executions. Denote the out-of-band authenticated values that S computes in these executions by $\sigma_1 \cdots \sigma_\ell \in \{0,1\}^{\ell'}$.

2. For each $i \in [\ell]$, S parses σ_i as a single character over an alphabet of size $k = 2^{\ell'}$; denote the ith character by β_i. S then receives $\mathcal{I} = \{i_1, \ldots, i_{|\mathcal{I}|}\} \subseteq [\ell]$ and sends $\sigma = \beta_{i_1} \| \ldots \| \beta_{i_{|\mathcal{I}|}}$ over the out-of-band authenticated channel.

3. R receives $\mathcal{I}$, parses $\sigma = \sigma_{i_1} \cdots \sigma_{i_{|\mathcal{I}|}}$ as $|\mathcal{I}|$ binary strings of length ℓ' each. For every $i \in \mathcal{I}$, denote by $\widehat{m_i}$ the output of R in the ith execution given R's view of that execution (including σ_i). If for every $i, j \in \mathcal{I}$ it holds that $\widehat{m_i} = \widehat{m_j}$, then R outputs $\widehat{m_{i_1}}$. Otherwise, R outputs $\perp$.

The correctness and security of the protocol π_{Lazy} are stated in the following theorem.

Theorem 5.1. *Let π' be a statistically-secure out-of-band (n, ℓ', r, ϵ')-authentication protocol, let $k = 2^{\ell'}$ and let $\ell \in \mathbb{N}$. Then, π_{Lazy} is a statistically-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol for lazy users over an alphabet of size k, where $\epsilon(\mathcal{I}) = (\epsilon')^{|\mathcal{I}|}$ for every $\mathcal{I} \subseteq [\ell]$.*

The correctness and round complexity of π_{Lazy} follow immediately from the correctness and round complexity of π', respectively. The unforgeability of π_{Lazy} for lazy users (vis-à-vis Definition 3.3) is proven in Lemma 5.3 below, yielding the above theorem.

A concrete instantiation. Naor et al. [NSS06] constructed a statistically-secure out-of-bound (n, ℓ', r, ϵ')-authentication protocol for any $n, r \in \mathbb{N}$ and any $\epsilon' \in (0,1)$, where $\ell' \leq \log(1/\epsilon') + \log^{(r-1)} + O(1)$. Instantiating our protocol π_{Lazy} with the protocol of Naor et al. as π', while setting $r = \Omega(\log^* n)$ and $\epsilon' = 1/2$, yields a statistically-secure out-of-band authentication protocol for lazy users with the same round complexity and a constant-size alphabet. This is formalized by the following corollary.

Corollary 5.2. *For any $n, \ell \in \mathbb{N}$, there exists a statistically-secure out-of-band $(n, \ell, \log^* n, \epsilon)$-authentication protocol for lazy users over a constant size alphabet, where $\epsilon(\mathcal{I}) = 2^{-|\mathcal{I}|}$ for every $\mathcal{I} \subseteq [\ell]$.*

Specifically, a more refined analysis of the protocol of Naor et al. for the case $\epsilon' = 1/2$ shows that a careful adjustment of the parameters of the protocol yields $\ell' \leq 8$, implying an out-of-band authentication protocol for lazy users over an alphabet of size 2^8 (see Section 8.2 for the technical details). This is already a reasonably-practical constant: Telegram, for example, uses an alphabet comprised of $333 > 2^8$ emoji symbols for its out-of-band key verification in voice calls [Tela].

We now turn to prove the security of our transformation. Lemma 5.3 yields the unforgeability of our protocol π_{Lazy}, proving Theorem 5.1. Let $t \in [\ell]$ denote the size of the subset $\mathcal{I}$ of indexes of the out-of-band authenticated value sent. For an adversary A, let Forge_A denote the event in which R outputs $\widehat{m} \neq \{m, \perp\}$ in an execution of π_{Lazy} with A.

Lemma 5.3. *For any computationally unbounded adversary A and for any $t \in [\ell]$, it holds that $\Pr[\mathsf{Forge}_A] \leq (\epsilon')^t$.*

Proof. For every $i \in \mathcal{I}$, $\widehat{m_i}$ is the output of the receiver in an execution of the basic protocol π' when S is invoked on the input m. Hence, by the unforgeability of basic protocol π',

$$\forall i \in \mathcal{I} : \Pr[\widehat{m_i} \notin \{m, \perp\}] \leq \epsilon'.$$

We wish to prove that $\Pr[\forall i \in \mathcal{I} : \widehat{m_i} \notin \{m, \perp\}] \leq (\epsilon')^t$. This follows from a standard argument (see, for example, [Gol98, p. 153]) that we briefly sketch here for completeness. Since the adversary A is computationally unbounded, we may assume without loss of generality that A is deterministic. Hence, the execution of π_{Lazy} with A is fully determined by the random tapes of S and R (the random tape of each party can be viewed as the concatenation of ℓ independently sampled random tapes, one per execution of the basic protocol π'). Let r_i denote the random coins of S and R in the ith execution of π', and let r_{-i} denote their random coins for all other executions. Then, it must hold that

$$\forall r_{-i} \in \{0,1\}^* : \Pr_{r_i}[\widehat{m_i} \notin \{m, \perp\} | r_{-i}] \leq \epsilon'.$$

If this is not the case – i.e., there exists some $r^*_{-i} \in \{0,1\}^*$ such that $\Pr_{r_i}\left[\widehat{m_i} \notin \{m, \perp\} | r^*_{-i}\right] > \epsilon'$ – then we can construct an adversary A' that breaks the unforgeability of π'. Concretely, A' simply invokes A, while simulating S and R in the remaining $\ell - 1$ executions of π' in π_{Lazy} with randomness r^*_{-i}. The above inequality implies in particular that

$$\forall i \in \mathcal{I} : \Pr[\widehat{m_i} \notin \{m, \perp\} | \forall i > j \in \mathcal{I} : \widehat{m_j} \notin \{m, \perp\}] \leq \epsilon'.$$

Indeed, it thus holds that

$$\Pr[\forall i \in \mathcal{I} : \widehat{m_i} \notin \{m, \perp\}] = \prod_{i \in \mathcal{I}} \Pr[\widehat{m_i} \notin \{m, \perp\} | \forall i > j \in \mathcal{I} : \widehat{m_j} \notin \{m, \perp\}] \leq (\epsilon')^t.$$

Finally, as R outputs some $\widehat{m} \neq \perp$ only if this is the output of all the executions corresponding to $\mathcal{I}$. Hence,

$$\begin{aligned}\Pr\left[\mathsf{Forge}_A\right] &\leq \Pr\left[\widehat{m} \notin \{m, \perp\}\right] \\ &\leq \Pr\left[\forall i \in \mathcal{I} : \widehat{m_i} \notin \{m, \perp\}\right] \\ &\leq \left(\epsilon'\right)^t.\end{aligned}$$

■

Applying our transformation to computationally-secure protocols. Should we consider only *efficient* man-in-the-middle adversaries, then the proof of Lemma 5.3 only uses the statistical-security of π' to deduce that it permits parallel composition while maintaining the security of each individual copy of the basic protocol exactly, independently of the other executions of the basic protocol. Therefore, it is indeed the case that if π' is a computationally-secure protocol that retains its security under parallel composition, then the same approach will yield a *computationally-secure* out-of-band authentication protocol for lazy users. More formally, the property we require of the basic protocol is the following: If the forgery probability in a stand-alone execution of the protocol is bounded by ϵ', then the probability that in t parallel executions of the protocol with the same input message m to S, the receiver R outputs some $\widehat{m} \notin \{m, \perp\}$ in *all* t executions is at most $(\epsilon')^t$.

6 Matching the Optimal Tradeoff for Computationally-Secure Protocols

In this section we show that Vaudenay's computationally-secure protocol [Vau05] can be extended to allow execution by lazy users, and that the resulting protocol matches our lower bound on the tradeoff between the adversary's forgery probability and the length of the out-of-band authenticated value for lazy users (see Theorem 7.1). That is, the protocol offers the optimal tradeoff between the adversary's forgery probability and the length of the partial out-of-band authenticated value considered by the lazy users.

The basic building block used by the protocol is any non-malleable statistically-binding commitment scheme Com. From a foundational point of view, such a scheme with a constant number of rounds can be constructed based on any one-way function in the standard model, and from a more practical point of view, such a scheme can be constructed by simply invoking a hash function modeled as a random oracle (see Section 2).

The protocol, which we denote by π_{Comp}, is parametrized by the security parameter $\lambda \in \mathbb{N}$, the message length $n = n(\lambda) \in \mathbb{N}$ and the length of the out-of-band authenticated value $\ell = \ell(\lambda) \in \mathbb{N}$, and is defined as follows:

1. On input the security parameter $\lambda \in \mathbb{N}$ and a message $m \in \{0,1\}^n$, the sender S chooses a random $r_S \leftarrow \{0,1\}^\ell$, sends m to the receiver R, and commits to the pair (m, r_S) to receiver R using Com. Denote the resulting commitment by c_S and its corresponding decommitment by d_S.[7] Denote the message and commitment as received by R by $\widehat{m}$ and $\widehat{c_S}$, respectively.

2. The receiver R chooses a random $r_R \leftarrow \{0,1\}^\ell$ and sends it to the sender S. Denote by $\widehat{r_R}$ the value that S receives.

[7] As a commitment scheme may be interactive, when referring to a commitment, we mean the transcript of the interaction between the committer and the receiver during an execution of the commit phase of the commitment scheme. When the scheme is non-interactive, a commitment is simply a single string sent from the committer to the receiver.

3. The sender S sends the decommitment d_S to R. Denote by $\widehat{d_S}$ the decommitment R receives. If $\widehat{d_S}$ is not a valid decommitment to $\widehat{c_S}$ or if the revealed value is not of the form $(\widehat{m}, *)$, then R outputs $\perp$. Otherwise, let $(\widehat{m}, \widehat{r_S})$ be the revealed value.

4. The sender S sends $\sigma = r_S \oplus \widehat{r_R}$ over the out-of-band channel. R checks if $\widehat{r_S} \oplus r_R = \sigma$. If so, R outputs $\widehat{m}$, and otherwise R outputs $\perp$.

The following theorem captures the security of the above protocol, stating that it provides the optimal tradeoff as discussed above.

Theorem 6.1. *Let $n = n(\cdot), r = r(\cdot)$ and $\ell = \ell(\cdot)$ be functions of the security parameter $\lambda \in \mathbb{N}$ and let Com be an r-round statistically-binding non-malleable commitment scheme. Then, protocol π_{Comp} is a computationally-secure out-of-band $(n, \ell, r+3, \epsilon)$-authentication protocol for lazy users (over the alphabet $\Sigma = \{0,1\}$), where $\epsilon(\lambda, \mathcal{I}) = 2^{-|\mathcal{I}|}$ for every $\lambda \in \mathbb{N}$ and for every $\mathcal{I} \subseteq [\ell(\lambda)]$.*

Our protocol incurs an almost minimal overhead in the number of rounds relative to the round complexity of the underlying commitment scheme: The number of rounds of insecure communication is $r + 2$ (this includes the $r + 1$ rounds necessary for commitment and decommitment), to which we add only a single message over the insecure channel, and a single message over the out-of-band authenticated channel. In the plain model, a non-malleable commitment is known to exist with $r = 3$, while in the random oracle model, there exist non-interactive non-malleable commitments (i.e., with $r = 1$).

The security proof of our protocol considers all possible synchronizations a man-in-the-middle adversary may impose on an execution of the protocol. For each such synchronization and for every possible subset $\mathcal{I} \subseteq [\ell]$ of positions of the out-of-band authenticated value, we bound the forgery probability by $2^{-|\mathcal{I}|} + \nu(\lambda)$, for a negligible function $\nu(\lambda)$, by converting an adversary achieving better forgery probability into an adversary that breaks a specific security property of the underlying commitment scheme (i.e., binding, hiding or non-malleability).

In what follows, for an adversary A we let Forge_A denote the event in which R outputs $\widehat{m} \notin \{m, \perp\}$ in an execution of π_{Comp} with A, where m is the input to S. Then, the following lemma establishes the unforgeability of π_{Comp}, yielding Theorem 6.1.

Lemma 6.2. *For any probabilistic polynomial time adversary A there exists a negligible function $\nu = \nu(\lambda)$, such that for all sufficiently large $\lambda \in \mathbb{N}$ and for every $\mathcal{I} = \mathcal{I}(\lambda) \subseteq [\ell(\lambda)]$ it holds that*

$$\Pr[\mathsf{Forge}_A] \leq 2^{-|\mathcal{I}|} + \nu(\lambda).$$

Proof. For simplicity of presentation, we first prove Lemma 6.2 assuming Com is a non-interactive commitment scheme (in addition to being statistically-binding and non-malleable), and then discuss how the proof extends to interactive schemes as well.

For a message v sent in the execution of the protocol, we denote by $T(v)$ the time in which v was sent. By the definition of the protocol π_{Comp}, it always holds that

$$T(c_S) < T(\widehat{r_R}) < T(d_S)$$

and

$$T(\widehat{c_S}) < T(r_R) < T(\widehat{d_S}).$$

We also assume without loss of generality that any adversary abides by the following timing restrictions:

1. Whenever a party (either S or R) is due to send a message according to the protocol, the adversary waits until this message is sent before deciding on its next action. This implies, for instance, that if $T(c_S) > T(\widehat{c_S})$ then it also holds that $T(c_S) > T(r_R)$.[8]

2. The adversary delays sending $\widehat{d_S}$ until the end of the execution; i.e., $T(\widehat{d_S}) > T(d_S)$.

Note that this assumption is indeed without loss of generality, as any adversary can be converted into an adversary that adheres to the above timing restrictions without damaging the probability of a successful forgery.

Given the above assumption, a man-in-the-middle adversary has three possible attack timings:

1. $T(c_S) < T(\widehat{r_R}) < T(d_S) < T(\widehat{c_S}) < T(r_R) < T(\widehat{d_S})$.

2. $T(\widehat{c_S}) < T(r_R) < T(c_S) < T(\widehat{r_R}) < T(d_S) < T(\widehat{d_S})$.

3. $T(c_S) < T(\widehat{c_S}) < T(r_R) < T(\widehat{r_R}) < T(d_S) < T(\widehat{d_S})$.

Denote the events in which each of the timings occurs by $\mathsf{T_1}, \mathsf{T_2}, \mathsf{T_3}$, respectively.

We can now turn to bound the probability of a successful forgery. Consider the event in which one of the (at most two different) commitments sent during the execution of π_{Comp} can be opened (information-theoretically speaking) to more than one value; denote this event by Coll. By the statistical binding property of Com, there exists a negligible function $\nu_1(\cdot)$ such that $\Pr[\mathsf{Coll}] < \nu_1(\lambda)$ for all sufficiently large $\lambda \in \mathbb{N}$. So for the remainder of the proof we will condition our analysis on the event $\overline{\mathsf{Coll}}$.

We bound the probability of forgery for each of the three possible attack timings separately, in three different claims below, and the lemma will then follow. The first claim address the first timing listed above. Intuitively speaking, conditioned on $\mathsf{T_1}$, by the time R randomly chooses r_R, the values $r_S, \widehat{r_S}$ and $\widehat{r_R}$ have already been determined and fixed. Hence, for any $\mathcal{I} \subset [\ell]$, the probability that $\sigma_{\mathcal{I}}$ is consistent with the view of R is $2^{-|\mathcal{I}|}$. Recall that for a string s and a set $\mathcal{I} \subseteq [|s|]$, the notation $s_{\mathcal{I}}$ denoted the substring of s obtained by concatenating the characters in positions $\mathcal{I}$ (we will use this notation throughout the remainder of the proof).

Claim 6.3. *Let $\lambda \in \mathbb{N}$ and let $\ell = \ell(\lambda)$. Then, for any adversary A and for every subset $\mathcal{I} \subseteq [\ell]$, it holds that*

$$\Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}} \wedge \mathsf{T_1}\right] = 2^{-|\mathcal{I}|}.$$

Proof of Claim 6.3. Conditioned on $\mathsf{T_1}$, it holds in particular that $T(r_R) > T(c_S), T(\widehat{c_S}), T(\widehat{r_R})$. Moreover, conditioned on $\overline{\mathsf{Coll}}$, the commitments c_S and $\widehat{c_S}$ can each be opened to (at most) a single value. This means that the values $r_S, \widehat{r_S}$ and $\widehat{r_R}$ have already been fixed when R chooses r_R. Hence, it holds that

$$\begin{aligned}
\Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}} \wedge \mathsf{T_1}\right] &\leq \Pr_{r_R}\left[(r_S \oplus \widehat{r_R})_{\mathcal{I}} = (\widehat{r_S} \oplus r_R)_{\mathcal{I}}\right] \\
&= \Pr_{r_R}\left[(r_R)_{\mathcal{I}} = (r_S \oplus \widehat{r_R} \oplus \widehat{r_S})_{\mathcal{I}}\right] \\
&= 2^{-|\mathcal{I}|}.
\end{aligned}$$

■

[8] Note that c_S is sent only after A decides on the input message m to S.

Claim 6.4. *Let $\lambda \in \mathbb{N}$ and let $\ell = \ell(\lambda)$. Then, for every probabilistic polynomial-time adversary A, there exists a probabilistic polynomial-time adversary B such that for every $\mathcal{I} \subseteq [\ell]$, it holds that*

$$\mathsf{Adv}^{B}_{\mathsf{Com-Hiding}}(\lambda) \geq \Pr\left[\overline{\mathsf{Coll}}\right] \cdot \Pr\left[\mathsf{T}_2\right] \cdot \left(\Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}} \wedge \mathsf{T}_2\right] - 2^{-|\mathcal{I}|}\right).$$

Proof of Claim 6.4. Given an adversary A against protocol π_{Comp}, we construct an adversary B that breaks the hiding property of Com whenever A succeeds in breaking the unforgeability of π_{Comp} and $\overline{\mathsf{Coll}}$ and T_2 occur.

More concretely, conditioned on T_2, the timing of A's attack is the following:

1. A sends $\widehat{m}$ and a commitment $\widehat{c_S}$ to R, and R sends r_R in response.
2. S sends m and a commitment c_S; A replies S with $\widehat{r_R}$.
3. S opens the commitment c_S followed by A opens the commitment $\widehat{c_S}$.

Let $r^* = r^*(\lambda)$ and $r^*_R = r^*_R(\lambda)$ be the random coins of A and the choice of r_R, respectively, that maximize A's success probability on input 1^λ conditioned on $\overline{\mathsf{Coll}} \wedge \mathsf{T}_2$. Note that in order for the attack of A to succeed, the commitment in Step 1 above has to be a valid commitment to some pair $(\widehat{m}, \widehat{r_S})$ (at least one such pair). So, if A has non-zero probability of succeeding, it means that conditioned on $\overline{\mathsf{Coll}}$, r^* uniquely determines some $\widehat{m}^*, \widehat{r_S}^*$ to which A commits in Step 1. r^* and r^*_R also uniquely define the message m^* which is set as the input to S.

Consider the following (non-uniform) adversary B, that on input 1^λ and a commitment c, distinguishes between the case that c is a commitment to (m, r_S) and the case that it is a commitment to $(m, 0)$ for strings m and r_S of its choice. B gets $r^* = r^*(\lambda)$ and $r^*_R = r^*_R(\lambda)$ as non-uniform advice, as well as $\widehat{m}^* = \widehat{m}^*(\lambda), \widehat{r_S}^* = \widehat{r_S}^*(\lambda)$ and $m^* = m^*(\lambda)$,[9] and is defined as follows:

1. B invokes $A(1^\lambda; r^*)$. A first outputs $\widehat{m}^*$ and a commitment to $(\widehat{m^*}, \widehat{r_S}^*)$ as her first message to R. B replies to A with r^*_R.
2. B chooses a random $r_S \leftarrow \{0,1\}^\ell$ and outputs $v_0 = (m^*, 0^\ell)$ and $v_1 = (m^*, r_S)$ as the challenge values in the hiding experiment of Com (see Definition A.2). In response, B gets a commitment c to v_b for some $b \in \{0,1\}$.
3. B forwards c to A as the commitment c_S of S in Step 2 above. A replies with $\widehat{r_R}$.
4. B receives $\widehat{r_R}$ and decides on her output as follows: If $(\widehat{r_R})_{\mathcal{I}} = (\widehat{r_S}^* \oplus r^*_R \oplus r_S)_{\mathcal{I}}$, output 1; otherwise, output 0.

For our analysis of B's success probability, we consider the following two cases:

- If $b = 1$: In this case, conditioned on $\overline{\mathsf{Coll}} \wedge \mathsf{T}_2$ and on r^*_R, B perfectly simulates an execution of π_{Comp} to A (for Steps 1 and 2 above). Conditioned on $\overline{\mathsf{Coll}}$, each of the commitments sent in this execution uniquely defines (at most) a single committed value. Hence, a necessary condition for A to enforce a successful forgery is that indeed $(\widehat{r_R})_{\mathcal{I}} = (\widehat{r_S}^* \oplus r^*_R \oplus r_S)_{\mathcal{I}}$. It follows that in case $b = 1$,

$$\Pr\left[\mathsf{Hiding}^{B}_{\mathsf{Com}}(\lambda, 1) = 1 | \overline{\mathsf{Coll}} \wedge \mathsf{T}_2\right] \geq \Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}} \wedge \mathsf{T}_2\right].$$

[9] m^* can be obtained from A given r^* and r^*_R. For a concise proof however, we give m^* to B as non-uniform advice as well.

- If $b = 0$: In this case, the view of A is independent of r_S, and in particular of $(r_S)_{\mathcal{I}}$ which is uniformly distributed over $\{0,1\}^{|\mathcal{I}|}$. Hence,

$$\Pr\left[\mathsf{Hiding}^{B}_{\mathsf{Com}}(\lambda,0)=1|\overline{\mathsf{Coll}}\wedge\mathsf{T}_2\right]\leq\Pr\left[(\widehat{r_R})_{\mathcal{I}}=(\widehat{r_S}^{*}\oplus r_R^{*}\oplus r_S)_{\mathcal{I}}\right]=2^{-|\mathcal{I}|}.$$

Putting things together,

$$\mathsf{Adv}^{B}_{\mathsf{Com}-\mathsf{Hiding}}(\lambda)\geq\Pr\left[\overline{\mathsf{Coll}}\right]\cdot\Pr\left[\mathsf{T}_2\right]\cdot\left(\Pr\left[\mathsf{Forge}_A|\overline{\mathsf{Coll}}\wedge\mathsf{T}_2\right]-2^{-|\mathcal{I}|}\right).$$

■

Claim 6.5. *For every probabilistic polynomial-time adversary A, there exists a pair of probabilistic polynomial-time algorithms (C,D) such that for every $\mathcal{I}\subseteq[\ell]$, it holds that*

$$\mathsf{Adv}^{(C,D)}_{\mathsf{Com}-\mathsf{NM}}(\lambda)\geq\Pr\left[\overline{\mathsf{Coll}}\right]\cdot\Pr\left[\mathsf{T}_3\right]\cdot\left(\Pr\left[\mathsf{Forge}_A|\overline{\mathsf{Coll}}\wedge\mathsf{T}_3\right]-2^{-|\mathcal{I}|}\right).$$

Proof of Claim 6.5. Let $r^*=r^*(\lambda)$ be the random coins of A that maximize the forgery probability (over the randomness of S and R). Conditioned on T_3, r^* uniquely determines the input message to S. Denote this message by $m^*=m^*(\lambda)$. The adversary C gets as input the security parameter 1^λ and a uniformly chosen ℓ-bit string the $z\leftarrow\{0,1\}^\ell$ as an auxiliary input. C also gets r^* as non-uniform advice, and is defined in the following manner:

1. In the left interaction, C gets a commitment c to a value v.
2. C chooses $r_R\leftarrow\{0,1\}^\ell$ at random. It then invokes $A(1^\lambda;r^*)$ and simulates a partial execution of π_{Comp} as follows:
   (a) C forwards the commitment c to A as the commitment of S to (m^*,r_S). A replies with $\widehat{c_S}$ as the commitment that is to be delivered to R. C then outputs $\widehat{c_S}$ in its right interaction.
   (b) C sends r_R to A, who outputs $\widehat{r_R}$ in reply. After receiving $\widehat{r_R}$, C terminates.

The distinguisher D then gets as input the random variable $\mathsf{mim}^{C}_{\mathsf{Com}}(v,z)$, which includes the following information (among other things):

- The auxiliary input z.
- The random strings r_R and $\widehat{r_R}$.
- The value v which either the value to which $\widehat{c_S}$ may be opened or $\perp$. Note that conditioned on $\overline{\mathsf{Coll}}$, it holds that $v=\perp$ either in the case that the $\widehat{c_S}$ is not a valid commitment (and cannot be opened to any value) or that it is identical to c (the commitment of v received in the left interaction). In the latter case, again by conditioning of $\overline{\mathsf{Coll}}$, this means that the only value this commitment might be opened to is v.

We now turn to define the distinguisher D. On input 1^λ and $\mathsf{mim}^{C}_{\mathsf{Com}}(v,z)$, the distinguisher D acts as follows:

1. If $v=\perp$, return 0.
2. Otherwise, parse v as (m^*,r_S) and check if $(z\oplus\widehat{r_R})_{\mathcal{I}}=(\widehat{r_S}\oplus r_R)_{\mathcal{I}}$. If so, output 1; otherwise, output 0.

For the analysis of the attack, we consider the following two cases.

Case 1: If $v = (m^*, z)$ (in particular, the second part of v is a uniformly distributed ℓ-bit string), then C perfectly simulates the honest parties of π_{Comp} to A until the latter terminates. Conditioned in $\overline{\mathsf{Coll}}$, the commitment $\widehat{c_S}$ can be opened to at most one value. If such a value exists, denote it by $(\widehat{m}, \widehat{r_S})$. A necessary condition for a successful forgery is that $\widehat{c_S}$ may be opened to some $(\widehat{m}, \widehat{r_S})$ and that $(r_S \oplus \widehat{r_R})_{\mathcal{I}} = (\widehat{r_S} \oplus r_R)_{\mathcal{I}}$. Therefore, a successful forgery implies that D outputs 1. Hence,

$$\Pr\left[D(1^\lambda, \mathsf{mim}^C_{\mathsf{Com}}((m^*, z), z)) = 1 | \overline{\mathsf{Coll}} \wedge \mathsf{T}_3\right] \geq \Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}} \wedge \mathsf{T}_3\right].$$

Case 2: If $v = (m^*, 0^\ell)$, then the view of A (as ran by C) is independent of z. In particular, the values $\widehat{r_S}$ and $\widehat{r_R}$ as produced by A are independent of z. The value r_R is drawn by C uniformly at random, and is therfore also independent of z. Hence, since z is distributed uniformly over $\{0,1\}^\ell$, it holds that

$$\begin{aligned}\Pr\left[D(1^\lambda, \mathsf{mim}^C_{\mathsf{Com}}((m^*, 0^\ell), z)) = 1 | \overline{\mathsf{Coll}} \wedge \mathsf{T}_3\right] &\leq \Pr\left[(z \oplus \widehat{r_R})_{\mathcal{I}} = (\widehat{r_S} \oplus r_R)_{\mathcal{I}}\right] \\ &= \Pr\left[z_{\mathcal{I}} = (\widehat{r_S} \oplus r_R \oplus \widehat{r_R})_{\mathcal{I}}\right] \\ &= 2^{-|\mathcal{I}|}.\end{aligned}$$

By an averaging argument, there exists some $z^* \in \{0,1\}^\ell$ such that

$$\begin{aligned}\mathsf{Adv}^{(\mathsf{C},\mathsf{D})}_{\mathsf{Com}-\mathsf{NM}}(\lambda) &\geq \Pr\left[\overline{\mathsf{Coll}}\right] \cdot \Pr\left[\mathsf{T}_3\right] \cdot \Big(\Pr\left[D(\mathsf{mim}^C_{\mathsf{Com}}((m^*, z^*), z^*)) = 1 | \overline{\mathsf{Coll}} \wedge \mathsf{T}_3\right] \\ &\quad - \Pr\left[D(\mathsf{mim}^C_{\mathsf{Com}}((m^*, 0^\ell), z^*)) = 1 | \overline{\mathsf{Coll}} \wedge \mathsf{T}_3\right]\Big) \\ &\geq \Pr\left[\overline{\mathsf{Coll}}\right] \cdot \Pr\left[\mathsf{T}_3\right] \cdot \left(\Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}} \wedge \mathsf{T}_3\right] - 2^{-|\mathcal{I}|}\right).\end{aligned}$$

and the claim follows. ■

We are now ready to conclude the proof of Lemma 6.2. By the above claims, it holds that

$$\begin{aligned}\Pr\left[\mathsf{Forge}_A\right] &\leq \Pr\left[\mathsf{Coll}\right] + \Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}}\right] \cdot \Pr\left[\overline{\mathsf{Coll}}\right] \\ &\leq \Pr\left[\mathsf{Coll}\right] + \Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}} \wedge \mathsf{T}_1\right] + \Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}} \wedge \mathsf{T}_2\right] \cdot \Pr\left[\overline{\mathsf{Coll}}\right] \cdot \Pr\left[\mathsf{T}_2\right] \\ &\quad + \Pr\left[\mathsf{Forge}_A | \overline{\mathsf{Coll}} \wedge \mathsf{T}_3\right] \cdot \Pr\left[\overline{\mathsf{Coll}}\right] \cdot \Pr\left[\mathsf{T}_3\right] \\ &\leq 2^{-|\mathcal{I}|} + \Pr\left[\mathsf{Coll}\right] + \mathsf{Adv}^B_{\mathsf{Com}-\mathsf{Hiding}}(\lambda) + \mathsf{Adv}^{(C,D)}_{\mathsf{Com}-\mathsf{NM}}(\lambda).\end{aligned}$$

By the security of Com (namely, the binding, hiding and non-malleability of Com), there exists a negligible function $\nu = \nu(\cdot)$ such that

$$\Pr\left[\mathsf{Forge}_A\right] \leq 2^{-|\mathcal{I}|} + \nu(\lambda).$$

■

Uniform vs. non-uniform reductions. The proof of Theorem 6.1 relies on non-uniform security reductions (namely, this is the case in the proofs of Claim 6.4 and of Claim 6.5). Hence, for security to be guaranteed by our proof the protocol has to be instantiated with a commitment scheme that is secure against non-uniform adversaries. From a theoretical standpoint, such constant-round schemes are known to exists (see Section 2) assuming one-way functions (that are secure against non-uniform adversaries), and from a more practical perspective, such schemes can be easily built in the random oracle model (for further discussion on the possible instantiations, see [RS18]). It is nevertheless possible to prove the security of the underlying approach of the protocol via a uniform reduction, if one or more of the following adjustments is applied to the protocol and/or to the security proof:

- One possibility is to have R commit to r_R at the beginning of the protocol, before S commits to (m, r_S) (and then open the commitment after S finishes her commitment, but before she opens it). In this case, π_{Comp} is a special case of the group out-of-band authentication protocol of Rotem and Segev [RS18], and their (uniform) proof readily extends to apply to lazy users.

- Another possibility is to assume that the adversary chooses the input message m to S *non-adaptively*. That is, she must choose m at the beginning of the protocol and cannot, for example, send $\widehat{c_S}$ to R and then choose m only upon receiving r_R. If the adversary chooses m non-adaptively, then we can assume without loss of generality that the first message sent in the execution is by S (since any adversary that does not obey this assumption can be transformed into one that does without damaging the probability of forgery), essentially reducing timing T_2 to timing T_3. We still need to address the non-uniformity in the proof of Claim 6.5. This proof can be made uniform if we consider a "more uniform" definition for non-malleability of commitments, in which the adversary first chooses the two values v_0 and v_1 from Definition A.1 (or a distribution from which they are sampled), and then receives a commitment c to one of them (chosen randomly) in its left interaction. The distinguisher D then needs to distinguish between the case where c is a commitment to v_0 and the case where it is a commitment to v_1.

Supporting interactive commitments. The proof of Lemma 6.2 can be easily extended to support the use of non-malleable commitment schemes which are interactive. First, let's observe that any such commitment scheme can be converted into a scheme in which the sender's first message is in and of itself statistically biding; i.e., with overwhelming probability, once this message is sent, even a computationally unrestricted adversary cannot complete the commitment phase in two different ways, generating two different commitments c_1 and c_2, such that it is possible to open them into two distinct messages. This can be done by first having the sender committing to the message using Naor's statistically-binding two-message commitment scheme [Nao91], and then committing to the same message using the original non-malleable scheme. It is not hard to see that this transformation achieves the afore-described property, while retaining non-malleability that is sufficient for our needs; i.e., an efficient adversary cannot produce a commitment for a related yet different value.[10]

Now, for a commitment c, we denote by $T(c)$ the time of the first message sent by the committer as part of the commitment. Under this notation, Claim 6.3 still holds as conditioned on $\mathsf{T}_1 \wedge \overline{\mathsf{Coll}}$, it is still true that r_R is sampled after all other variables contributing to the computation of σ and of $\widehat{\sigma}$ are uniquely determined. Claim 6.4 holds since it is still the case, conditioned on $\mathsf{T}_2 \wedge \overline{\mathsf{Coll}}$, that by the time d_S is due to be sent in the simulation of π_{Comp} that B conducts, B has sufficient information to determine whether or not $(\widehat{r_R})_{\mathcal{I}} = (\widehat{r_S}^* \oplus r_R^* \oplus r_S)_{\mathcal{I}}$. In addition, the "binding part" (which consists of Naor's commitment scheme) of the commitments c (which B forwards as c_S) is completed, and hence B can be seen as an adversary against Naor's commitment scheme. Similarly, Claim 6.5 holds since conditioned on $\mathsf{T}_3 \wedge \overline{\mathsf{Coll}}$, by the time d_S is due in the simulation of π_{Comp} that C conducts, sufficient information was generated for the distinguisher D to determine if indeed $(z \oplus \widehat{r_R})_{\mathcal{I}} = (\widehat{r_S} \oplus r_R)_{\mathcal{I}}$.

A final, rather technical modification to the proof, is that in the proof of Claims 6.4 and 6.5, the adversaries we construct have to synchronize between the commitments they receive/produce as part of the security experiments in which they participate (i.e., the hiding and non-malleability experiment defined in Appendix A) and the commitments they relay to and receive from the man-in-the middle adversary A, in the simulations of π_{Comp} they conduct.

[10] The adversary might be able to produce a different commitment for the *same* value, but it is easy to see that this is benign in our protocol, and our proof of security still holds.

7 Lower Bounds on the Security of Lazy Users

Vaudenay [Vau05] and Naor et al. [NSS06] established tight bounds on the tradeoff between the adversary's forgery probability and the length of the out-of-band authenticated value in out-of-band authentication. In this section we show that their lower bounds, in both the computational and statistical setting, directly translate into corresponding lower bounds for protocols that are executed by lazy users.

7.1 Computationally-Secure Protocols

In any computationally-secure out-of-band authentication protocol where the probability of forgery is bounded by $\epsilon > 0$, the sender must out-of-band authenticate at least $\log(1/\epsilon)$ bits. This can be seen, for example, by analyzing the collision probability of the random variable corresponding to the out-of-band authenticated value (see for example, [PV06]). Below, we show that this reasoning generalizes to the case of lazy users: Namely, for each number $k \in [\ell]$ of bits read from the out-of-band authenticated value, we provide a corresponding lower bound.

Theorem 7.1. *For any computationally-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol for lazy users over alphabet Σ, it holds that*

$$\epsilon(\mathcal{I}) \geq 2^{-|\mathcal{I}| \cdot \log |\Sigma|} - 2^{-n}$$

for every $\mathcal{I} \subseteq [\ell]$.

Proof. Let π be any computationally-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol for lazy users over alphabet Σ. Let $\lambda \in \mathbb{N}$ and $\ell = \ell(\lambda)$ and fix any $\mathcal{I} \subseteq [\ell]$. Consider the following attack:

1. Choose a random $m \leftarrow \{0,1\}^n$ and run an honest execution with S on input m (with the adversary playing the role of R). Denote by v the out-of-band authenticated value S sends at the end of the execution. Delay the relaying of v to (the real) R until the end of the attack.

2. Choose a random $\widehat{m} \leftarrow \{0,1\}^n$ and run an honest execution with R, where the adversary plays the role S on input $\widehat{m}$. Denote by $\widehat{v}$ the out-of-band authenticated value that the simulated sender sends at the end of the execution. If $\widehat{v}_{\mathcal{I}} = v_{\mathcal{I}}$, forward v to R; otherwise, terminate.

Denote by $V_{\mathcal{I}}$ the random variable corresponding to the substring of the out-of-band authenticated value defined by the positions in $\mathcal{I}$, where the distribution of $V_{\mathcal{I}}$ is induced by an honest execution of π on a randomly chosen input message to S. Then, the following holds:

$$\begin{aligned}\Pr_{(\widehat{v}_{\mathcal{I}}, v_{\mathcal{I}}) \leftarrow V_{\mathcal{I}} \times V_{\mathcal{I}}} [\widehat{v}_{\mathcal{I}} = v_{\mathcal{I}}] &= \sum_{v_{\mathcal{I}}} \left(\Pr[V_{\mathcal{I}} = v_{\mathcal{I}}]\right)^2 = 2^{\log \sum_{v_{\mathcal{I}}} (\Pr[V_{\mathcal{I}} = v_{\mathcal{I}}])^2} \\ &\geq 2^{\sum_{v_{\mathcal{I}}} \Pr[V_{\mathcal{I}} = v_{\mathcal{I}}] \log(\Pr[V_{\mathcal{I}} = v_{\mathcal{I}}])} = 2^{-\mathrm{H}(V_{\mathcal{I}})}.\end{aligned}$$

The inequality above follows from Jensen's inequality.

Let $\mathsf{Forge}_{\mathcal{I}}$ denote the event in which the above attack goes through; i.e., R outputs a fraudulent message. By the correctness of π, it holds that

$$\begin{aligned}\Pr[\mathsf{Forge}_{\mathcal{I}}] &\geq \Pr[\widehat{v}_{\mathcal{I}} = v_{\mathcal{I}} \wedge \widehat{m} \neq m] \\ &\geq \Pr[\widehat{v}_{\mathcal{I}} = v_{\mathcal{I}}] - \Pr[\widehat{m} = m] \\ &\geq 2^{-\mathrm{H}(V_{\mathcal{I}})} - 2^n.\end{aligned}$$

On the one hand, by the unforgeability of π, it must hold that $\epsilon(\mathcal{I}) \geq 2^{-\mathrm{H}(V_{\mathcal{I}})} - 2^n$. On the other hand, it is always the case that $\mathrm{H}(V_{\mathcal{I}}) \leq |\mathcal{I}| \cdot \log|\Sigma|$. Taken together, these inequalities yield the theorem. ■

The lower bound of Theorem 7.1 should be thought of in the following terms. On the one hand, if the message to be authenticated is short (relative to the bandwidth of the out-of-band authenticated channel), then the sender can just go ahead and send it over the out-of-band channel. On the other hand, if it is long, then the term 2^{-n} is small and of little significance, and the attack from our proof succeeds with probability close to $2^{-|\mathcal{I}|\cdot\log|\Sigma|}$. Specifically, for any protocol in which the length of the out-of-band authenticated value is independent of the length of the input message to be authenticated, the success probability of our attack can be made arbitrarily close to $2^{-|\mathcal{I}|\cdot\log|\Sigma|}$ (while considering arbitrarily long input messages).

7.2 Statistically-Secure Protocols

Naor et al. [NSS06] proved a lower bound on the length of the out-of-band authenticated value in any statistically-secure out-of-band authentication protocol. More precisely, they provided a lower bound on the Shannon entropy of the random variable corresponding to the out-of-band authenticated value. If we denote this random value by V, the lower bound of Naor et al. can be articulated as follow:

Theorem 7.2 ([NSS06])**.** *For any statistically-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol it holds that*

$$\epsilon \geq 2^{-\mathrm{H}(V)/2} - 2^{-n}$$

Theorem 7.2 implies the following, more general, lower bound for out-of-band authentication protocols for lazy users over possibly non-binary alphabets.

Corollary 7.3. *For any statistically-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol for lazy users over alphabet Σ, it holds that for every $\mathcal{I} \subseteq [\ell]$*

$$\epsilon(|\mathcal{I}|) \geq 2^{-|\mathcal{I}|\cdot\log(|\Sigma|)/2} - 2^{-n}.$$

Proof. Let π be any (n, ℓ, r, ϵ)-authentication protocol for lazy users over alphabet Σ. By definition, this means that for any $\mathcal{I} \subseteq [\ell]$, the induced protocol $\pi_{\mathcal{I}}$ is an $(n, |\mathcal{I}|, r, \epsilon(\mathcal{I}))$-authentication protocol. For every $\mathcal{I} \subseteq [\ell]$, denote by $V_{\mathcal{I}}$ the random variable corresponding to the substring of the out-of-band authenticated value that is induced by the subset $\mathcal{I}$. Hence, by Theorem 7.2, for every $\mathcal{I} \subseteq [\ell]$ it holds that

$$\epsilon(|\mathcal{I}|) \geq 2^{-\mathrm{H}(V_{\mathcal{I}})/2} - 2^{-n}.$$

For every $\mathcal{I} \subseteq [\ell]$ it holds that $\mathrm{H}(V_{\mathcal{I}}) \leq |\mathcal{I}| \cdot \log|\Sigma|$, and combining this fact with the above inequality completes the proof. ■

8 Extensions

In this section we consider two possible extensions of our work. First, in Section 8.1, we discuss a stronger notion of security which we call "adaptive laziness" and how our definitions and results extend to it. Then, in Section 8.2, we provide a refined version and analysis of the Naor et al. [NSS06] protocol for the case where the forgery probability is bounded by a constant, yielding an out-of-band authenticated value of a small constant size, which translates to a small alphabet in our transformation of Section 5.

8.1 Adaptive Laziness

In what follows we discuss a stronger notion of security to which we refer as "adaptive laziness". Informally, in the definitions of Section 3, the unforgeability property requires that an adversary cannot make the receiver accept a fraudulent message, even if the parties consider only a substring of the out-of-band authenticated value, where the positions to be considered may be arbitrarily chosen – but are fixed before the execution. In some scenarios, it might make sense to consider an adversary that can adaptively choose these positions *after* the execution of the protocol is completed (and the out-of-band authenticated value is known).

8.1.1 Defining Adaptive Laziness

To formally define the notion of adaptive laziness, we introduce the following notation. For any out-of-band authentication protocol π and an adversary A, we consider a related protocol $\pi_{\mathsf{AL}}(A,t)$ (where AL stands for "adaptive laziness"), in which A chooses the subset $\mathcal{I}$ of t locations after the out-of-band authenticated value is sent. We start by considering the statistical setting. Let π be a statistically-secure out-of-band (n,ℓ,r,ϵ)-authentication protocol over alphabet Σ. The protocol $\pi_{\mathsf{AL}}(A,t)$ is comprised of three steps:

1. On input $m \in \{0,1\}^n$ to S, the sender S and receiver R run the first $r-1$ rounds of π. Let $v \in \Sigma^\ell$ be the out-of-band authenticated value that S is due to send in round r.
2. A receives v, and chooses a subset $\mathcal{I} \subseteq [\ell]$ of t locations (after observing the transcript of the protocol and in particular v).
3. S receives $\mathcal{I}$ and sends only $v_{\mathcal{I}}$ over the out-of-band authenticated channel.
4. R receives $v_{\mathcal{I}}$ and $\mathcal{I}$ and decides on her output according to π.

The modified protocol is defined similarly for the computational setting, with the exception that the parameters involved may be functions of the security parameter $\lambda \in \mathbb{N}$. Using the above notation, we move on to define statistically-secure out-of-band authentication with *adaptive laziness*. For ease of presentation, Definitions 8.1 and 8.2 below requires that the probability of forgery be bounded as a function of the *size* of $\mathcal{I}$ (and not $\mathcal{I}$ itself, as in Definitions 3.3 and 3.4), but one can easily generalize it to accommodate more refined bounds on the forgery probability as well.

Definition 8.1. Let $n,\ell,r \in \mathbb{N}$ and let $\epsilon : [\ell] \to [0,1]$. A protocol π is *a statistically-secure out-of-band (n,ℓ,r,ϵ)-authentication protocol with adaptive laziness* over alphabet Σ if for every adversary A and for every $t \in [\ell]$, the protocol $\pi_{\mathsf{AL}}(A,t)$ is a statistically-secure out-of-band $(n,t,r,\epsilon(t))$-authentication protocol.

Definition 8.2. Let $n = n(\lambda), \ell = \ell(\lambda), r = r(\lambda)$ and $\Sigma = \Sigma(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$, and let $\epsilon = \epsilon(\lambda,\cdot) : [\ell] \to [0,1]$. A protocol π is *a computationally-secure out-of-band (n,ℓ,r,ϵ)-authentication protocol with adaptive laziness* over alphabet Σ if for every probabilistic polynomial-time adversary A and for every $t \in [\ell]$, the protocol $\pi_{\mathsf{AL}}(A,t)$ is a computationally-secure out-of-band $(n,t,r,\epsilon(\cdot,t))$-authentication protocol.

8.1.2 Extending Our Security Proofs to Adaptive Laziness

In this section we discuss how to extend our security proofs presented in Sections 5 and 6 to the setting of adaptive laziness. We first introduce the notion of *substring collision probability*, denoted $\mathsf{SS\text{-}CP}$.

Definition 8.3. Let $\ell, t \in \mathbb{N}$, let Σ be an alphabet and let D be a distribution over strings of length ℓ over Σ. Then, the t-substring collision probability of D, denoted $\mathsf{SS\text{-}CP}(D,t)$ is defined as follows:

$$\mathsf{SS\text{-}CP}(D,t) \stackrel{\text{def}}{=} \Pr_{(x,x') \leftarrow D \times D}\left[\exists \mathcal{I} \in \binom{[\ell]}{t} \text{ s.t. } x_{\mathcal{I}} = x'_{\mathcal{I}}\right]$$

where for a set $\mathcal{S}$ and an integer c, the notation $\binom{\mathcal{S}}{c}$ denotes the family of all subsets of $\mathcal{S}$ consisting of exactly c elements.
As a particular case, we define the (k,ℓ,t)-substring collision probability as follows:

$$\mathsf{SS\text{-}CP}(k,\ell,t) \stackrel{\text{def}}{=} \mathsf{SS\text{-}CP}\left(U\left[[k]^{\ell}\right], t\right)$$

where $U\left[[k]^{\ell}\right]$ is the uniform distribution over $[k]^{\ell}$.

The computational setting. The proof of Lemma 6.2 can be adjusted to show that the protocol π_{Comp} remains secure even in the face of adaptive laziness. Informally, in each of the claims used in the proof of Lemma 6.2, the term $2^{-|\mathcal{I}|}$ may be easily replaced by the term $\mathsf{SS\text{-}CP}(2,\ell,|\mathcal{I}|)$ (recall Definition 8.3), yielding the following lemma.

Lemma 8.4. *For any probabilistic polynomial-time adversary A there exists a negligible function $\nu = \nu(\lambda)$, such that for all sufficiently large $\lambda \in \mathbb{N}$ and for every $t \in [\ell]$ it holds that*

$$\Pr\left[\exists \mathcal{I} \in \binom{[\ell]}{t} : (r_S \oplus \widehat{r_R})_{\mathcal{I}} = (\widehat{r_S} \oplus r_R)_{\mathcal{I}}\right] \leq \mathsf{SS\text{-}CP}(2,\ell,t) + \nu(\lambda).$$

Since the revised proof is very similar to the proof of Lemma 6.2, we skip the full proof and briefly review the modifications required in each of the claims:

- **Claim 6.3:** In the case considered in this claim, r_R is sampled once $r_S, \widehat{r_S}$ and $\widehat{r_R}$ are fixed, and hence
$$\Pr\left[\exists \mathcal{I} \in \binom{[\ell]}{t} : (r_S \oplus \widehat{r_R})_{\mathcal{I}} = (\widehat{r_S} \oplus r_R)_{\mathcal{I}} \,|\, \overline{\mathsf{Coll}} \wedge \mathsf{T_1}\right] \leq \mathsf{SS\text{-}CP}(2,\ell,t).$$
- **Claim 6.4:** The adversary B against the hiding property of the commitment scheme is modified as follows. Upon receiving $\widehat{r_S}$, B outputs 1 if and only if there exists a subset $\mathcal{I} \subset [\ell]$ of size t of positions such that $(r_S \oplus \widehat{r_R})_{\mathcal{I}} = (\widehat{r_S} \oplus r_R)_{\mathcal{I}}$. Note that indeed this check can be made efficiently, as the above condition holds if and only if the Hamming distance between $r_S \oplus \widehat{r_R}$ and $\widehat{r_S} \oplus r_R$ is at most $\ell - t$. The analysis of the reduction can then be modified as follows. The case $b = 1$ remains virtually unchanged, while in the case $b = 0$, r_S is uniformly distributed and independent of $\widehat{r_S}, \widehat{r_R}$ and r_R and hence the probability that the B outputs 1 in this case is $\mathsf{SS\text{-}CP}(2,\ell,t)$.
- **Claim 6.5:** The distinguisher D against the non-malleability of the commitment scheme is modified in a similar manner to the previous case. It checks weather there exists a subset $\mathcal{I} \subseteq [\ell]$ of positions such that $(z \oplus \widehat{r_R})_{\mathcal{I}} = (\widehat{r_S} \oplus r_R)_{\mathcal{I}}$, and output 1 if and only if this is indeed the case. The analysis then also changes similarly to the previous case.

By the way R decides on her output according to the definition of π_{Comp}, Lemma 8.4 immediately implies the following theorem.

Theorem 8.5. *Let $n = n(\cdot), r = r(\cdot)$ and $\ell = \ell(\cdot)$ be functions of the security parameter $\lambda \in \mathbb{N}$ and let Com be an r-round statistically-binding non-malleable commitment scheme. Then, protocol π_{Comp} is a computationally-secure out-of-band $(n,\ell,r+3,\epsilon)$-authentication protocol with adaptive laziness over an alphabet of size 2, where $\epsilon(\lambda,t) = 2^{-t}$ for every $\lambda \in \mathbb{N}$ and for every $t \in [\ell(\lambda)]$.*

The statistical setting. Consider a protocol π_{Lazy} which is the result of applying our general transformation from Section 5 to a statistically-secure out-of-band (n, ℓ', r, ϵ)-authentication protocol π, using ℓ parallel repetitions of π. For every $i \in [\ell]$, let $\widehat{M_i}$ be the random variable denoting the intended output of R in the ith parallel execution of π. Observe, that for every (computationally unbounded) adversary A and for every $t \in [\ell]$, the following holds: The probability that after the execution of π_{Lazy}, there exist a message $\widehat{m} \notin \{m, \perp\}$ (where m is the input message to S) and a subset $\mathcal{I} \in \binom{[\ell]}{t}$ such that $\widehat{M_i} = \widehat{m}$ for every $i \in \mathcal{I}$ is bounded by $\mathsf{SS\text{-}CP}(1/\epsilon, \ell, t)$. This follows from the observation we make in the proof of Lemma 5.3, that for every $i \in [\ell]$ the probability that $\widehat{M_i} = \widehat{m}$ is bounded by ϵ, even conditioned on any arbitrary outcome of all other executions.

Recall that by the definition of π_{Lazy}, in order for R to output some message $\widehat{m} \notin \{m, \perp\}$, it must be the case that $\widehat{M_i} = \widehat{m}$ for every $i \in \mathcal{I}$. Hence, when the adversary chooses a subset $\mathcal{I}$ of t positions for the parties to compare, it holds that the probability that R outputs a fraudulent message is bounded by $\mathsf{SS\text{-}CP}(1/\epsilon, \ell, t)$ as well. Namely, where π is the protocol of Naor et al. [NSS06] when instantiated with $\epsilon = 1/2$ and $\ell' = 2^8$ (see Section 8.2), we have that for every $t \in [\ell]$, the probability that an adversary can adaptively choose a subset of size t and make R outputs a fraudulent message is at most $\mathsf{SS\text{-}CP}(2, \ell, t)$. This yields the following theorem (for which we omit the formal proof, which is very similar to that of Theorem 5.1).

Theorem 8.6. *Let π' be a statistically-secure out-of-band (n, ℓ', r, ϵ')-authentication protocol, let $k = 2^{\ell'}$ and let $\ell \in \mathbb{N}$. Then, π_{Lazy} is a statistically-secure out-of-band (n, ℓ, r, ϵ)-authentication protocol with adaptive laziness over an alphabet of size k, where $\epsilon(t) = \mathsf{SS\text{-}CP}(1/\epsilon, \ell, t)$ for every $t \in [\ell]$.*

8.2 Statistical Security with Smaller Alphabets

In this section we present a more refined analysis of the length ℓ of the out-of-band authenticated value in the protocol Naor et al. [NSS06], when invoked with parameters that ensure a constant bound on the forgery probability ϵ (we consider $\epsilon = 1/2$ for concreteness). Concretely, we show that a careful adjustment of the protocol and its parameters ensure $\ell \leq 8$. Hence, when their protocol is plugged into our construction of Section 5, the resulted out-of-band authentication protocol for lazy users is over an alphabet of size at most 2^8 (which for example, as discussed in Section 5, fits in nicely in a sufficiently diverse set of emoji).

The protocol of Naor et al. relies on iterative hashing of the input message, where each party computes a sequence of intermediate messages as a function of the transcript. Informally speaking, the protocol advances in rounds, where in each round the two parties view the intermediate message as a polynomial over a finite field of some carefully chosen size, evaluate it on a random field element, and apply a random shift to the result. Denote the number of hashing rounds by r, and for $j \in [r]$ denote the length of the intermediate message of round j by n_j and the size of the field of round j by q_j. The exact choice of the values $\{q_j\}_{j \in [r]}$ should aim to strike a balance between two (supposedly contradicting) objectives:

1. On the one hand, in terms of efficiency, the field sizes dictate the length of the intermediate messages and of the out-of-band authenticated value. Namely, the out-of-band authenticated value is made up of two elements from the final field, which is of size q_r, and thus the length of this value is bounded by $\lceil 2 \log q_r \rceil$. Hence, the sharper the decline in the sizes of the fields, the faster the decrease in length of the intermediate messages. Intuitively, this means that if choose small q_j's, we can achieve a short out-of-band authenticated value in less rounds.

2. On the other hand, large fields translate into better security guarantees. As Naor et al. note, the first message to get hashed is the input message to the sender S, and the result of last hash function is sent as part of the out-of-band authenticated value. This means that a prerequisite for a successful forgery is the existence of a round $j \in [r]$ such that the intermediate messages the two parties see before the jth hash function is applied are different, but they both compute the same hash value in round j. Denote this event by Coll_j. For every $j \in [r]$, Naor et al. bound $\Pr[\mathsf{Coll}_j]$ by $\max\{1/q_j, 1/q_j \cdot \lceil n_j/\log q_j \rceil\}$, where n_j is the length of the jth intermediate message (i.e., just before the jth hash function is applied). For reasonable choices of q_j, the dominant term is $1/q_j \cdot \lceil n_j/\log q_j \rceil$, which is simply an upper bound on the probability that two distinct polynomials over a finite field of size q_j, each of them encoded using n_j bits, evaluate to the same value when applied to a random field element (this follows from the Schwartz-Zippel lemma, since an n_j-bit string can be seen as a polynomial of degree at most $\lceil n_j/\log q_j \rceil$ over a field of size q_j). Taking a union bound, the total probability of forgery can be bounded by $\sum_{j=1}^{r} 1/q_j \cdot \lceil n_j/\log q_j \rceil$.

Given these two considerations, Naor et al. set each q_j to be in the segment $\left[\frac{2^{r-j+1}\cdot n_j}{\epsilon}, \frac{2^{r-j+2}\cdot n_j}{\epsilon}\right)$, where $n_j = \lceil 2q_{j-1} \rceil$ for every $j \in \{2, \ldots, r\}$ (n_1 is the length of the input message), and the parameters r and ϵ (the number of hashing rounds and the desired bound on the forgery probability, respectively) parametrize the protocol. In terms of security, this indeed assures that $1/q_j \cdot \lceil n_j/\log q_j \rceil \leq \epsilon/2^{r-j+1}$ for every $j \in [r]$, and hence $\sum_{j=1}^{r} 1/q_j \cdot \lceil n_j/\log q_j \rceil < \epsilon$. In terms of efficiency however, if the protocol is invoked with $r = \Omega(\log^* n)$ and a constant ϵ, their analysis yields a constant length out-of-band authenticated value, but this constant is fairly large and impractical. This is to be expected since their analysis applies to more general choices of the parameters, but in the context of Corollary 5.2 setting $r = \Omega(\log^* n)$ and $\epsilon = 1/2$ is sufficient.

Consider the following choice of parameters. Set $r = r' + 2$ for $r' = \Omega(\log^* n)$. For $j \in [r']$ set q_j to be in the interval $\left[8 \cdot 2^{r-j-1}, 8 \cdot 2^{r-j}\right)$, and set $q_{r-1} = 33$ and $q_r = 13$. As before $n_1 = n$ and $n_j = \lceil 2 \log q_{j-1} \rceil$ for every $j \in \{2, \ldots, r\}$. Indeed, since the length ℓ of the out-of-band authenticated value is at most $\lceil 2 \log q_r \rceil$, our choice of parameters implies that $\ell \leq \lceil 2 \log 13 \rceil \leq 8$ as we wanted to show.

We are left with showing that the total probability of forgery is at most $1/2$. To that end, we note that the above choice of parameters can be seen as first running the protocol of Naor et al. (with their general purpose choices of $\{q_j\}_j$) for $r' = r - 2$ hashing rounds and with $\epsilon = 1/8$, and then run two more hashing rounds with specifically tailored choices of q_{r-1} and q_r. This view enables us to use the analysis of Naor et al. "out of the box" to deduce n_{r-1}, leaving us with the task of bounding $\Pr[\mathsf{Coll}_{r-1} \cup \mathsf{Coll}_r]$.

Adopting the above point of view, the analysis of Naor et al. yields that $n_{r-1} \leq 19$. By our choice of q_{r-1}, it follows that

$$\Pr[\mathsf{Coll}_{r-1}] \leq 1/33 \cdot \lceil 19/\log 33 \rceil < 1/8.$$

Moreover, $n_r = \lceil 2 \log q_{r-1} \rceil = 8$, implying

$$\Pr[\mathsf{Coll}_r] \leq 1/13 \cdot \lceil 8/\log 13 \rceil < 1/4.$$

Since the properties of the Naor et al. guarantee that $\Pr\left[\bigcup_{j\in[r-2]} \mathsf{Coll}_j\right] \leq 1/8$, then by a simple union bound, it holds that the probability of foregery is bounded by

$$\Pr\left[\bigcup_{j\in[r-2]} \mathsf{Coll}_j\right] + \Pr[\mathsf{Coll}_{r-1}] + \Pr[\mathsf{Coll}_r] \leq 1/2.$$

References

[AFJ15] D. Alghamdi, I. Flechais, and M. Jirotka. Security practices for households bank customers in the kingdom of Saudi Arabia. In *Symposium on usable privacy and security (SOUPS)*, pages 297–308, 2015.

[BA04] D. Besnard and B. Arief. Computer security impaired by legitimate users. *Computers & Security*, 23(3):253–264, 2004.

[Bar02] B. Barak. Constant-round coin-tossing with a man in the middle or realizing the shared random string model. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, pages 345–355, 2002.

[BM94] S. M. Bellovin and M. Merritt. An attack on the Interlock protocol when used for authentication. *IEEE Transactions on Information Theory*, 40(1):273–275, 1994.

[BR93] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM Conference on Computer and Communications Security*, pages 62–73, 1993.

[BSJ+17] M. Bellare, A. C. Singh, J. Jaeger, M. Nyayapati, and I. Stepanovs. Ratcheted encryption and key exchange: The security of messaging. In *Advances in Cryptology – CRYPTO '17*, pages 619–650, 2017.

[CCD+17] K. Cohn-Gordon, C. J. F. Cremers, B. Dowling, L. Garratt, and D. Stebila. A formal security analysis of the Signal messaging protocol. In *Proceedings of the 2nd IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 451–466, 2017.

[CGC17] K. Cohn-Gordon and C. Cremers. Mind the gap: Where provable security and real-world messaging don't quite meet. Cryptology ePrint Archive, Report 2017/982, 2017.

[CGCG+17] K. Cohn-Gordon, C. Cremers, L. Garratt, J. Millican, and K. Milner. On ends-to-ends encryption: Asynchronous group messaging with strong security guarantees. Cryptology ePrint Archive, Report 2017/666, 2017.

[DDB+16] J. L. Dupree, R. Devries, D. M. Berry, and E. Lank. Privacy personas: Clustering users via attitudes and behaviors toward security practices. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 5228–5239. ACM, 2016.

[DDN00] D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. *SIAM Journal on Computing*, 30(2):391–437, 2000.

[Ell96] C. M. Ellison. Establishing identity without certification authorities. In *Proceedings of the 6th USENIX Security Symposium*, pages 7–7, 1996.

[FMB+16] T. Frosch, C. Mainka, C. Bader, F. Bergsma, J. Schwenk, and T. Holz. How secure is TextSecure? In *Proceedings of the 1st IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 457–472, 2016.

[GLO+12] V. Goyal, C.-K. Lee, R. Ostrovsky, and I. Visconti. Constructing non-malleable commitments: A black-box approach. In *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science*, pages 51–60, 2012.

[Gol98] O. Goldreich. Modern cryptography, probabilistic proofs and pseudorandomness. Springer, 1998.

[Gol01] O. Goldreich. Foundations of Cryptography – Volume 1: Basic Techniques. Cambridge University Press, 2001.

[Goy11] V. Goyal. Constant round non-malleable protocols using one way functions. In *Proceedings of the 43rd Annual ACM Symposium on Theory of Computing*, pages 695–704, 2011.

[GPR16] V. Goyal, O. Pandey, and S. Richelson. Textbook non-malleable commitments. In *Proceedings of the 48th annual ACM Symposium on Theory of Computing*, pages 1128–1141, 2016.

[Gre18a] M. Green. Attack of the week: Group messaging in WhatsApp and Signal. *A Few Thoughts on Cryptographic Engineering*. Available at `https://blog.cryptographyengineering.com/2018/01/10/attack-of-the-week-group-messaging`, 2018.

[Gre18b] A. Greenberg. WhatsApp security flaws could allow snoops to slide into group chats. *Wired Magazine*. Available at `https://www.wired.com/story/whatsapp-security-flaws-encryption-group-chats`, 2018.

[GRR+14] V. Goyal, S. Richelson, A. Rosen, and M. Vald. An algebraic approach to non-malleability. In *Proceedings of the 55th Annual IEEE Symposium on Foundations of Computer Science*, pages 41–50, 2014.

[Her09] C. Herley. So long, and no thanks for the externalities: the rational rejection of security advice by users. In *Proceedings of the workshop on New security paradigms*, pages 133–144, 2009.

[HL16] A. Herzberg and H. Leibowitz. Can Johnny finally encrypt?: Evaluating E2E-encryption in popular IM applications. In *Proceedings of the 6th Workshop on Socio-Technical Aspects in Security and Trust*, pages 17–28, 2016.

[HZF+14] M. Harbach, E. V. Zezschwitz, A. Fichtner, A. D. Luca, and M. Smith. It's a hard lock life: A field study of smartphone (un)locking behavior and risk perception. In *Symposium on usable privacy and security (SOUPS)*, pages 213–230, 2014.

[KBB17] N. Kobeissi, K. Bhargavan, and B. Blanchet. Automated verification for secure messaging protocols and their implementations: A symbolic and computational approach. In *Proceedings of the 2nd IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 435–450, 2017.

[LP09] H. Lin and R. Pass. Non-malleability amplification. In *Proceedings of the 41st annual ACM Symposium on Theory of Computing*, pages 189–198, 2009.

[LP11] H. Lin and R. Pass. Constant-round non-malleable commitments from any one-way function. In *Proceedings of the 43rd annual ACM symposium on Theory of computing*, pages 705–714, 2011.

[LS03] S. Li and H.-Y. Shum. Secure human-computer identification against peeping attacks (SecHCI): A survey, 2003.

[Mem17] T. Membe. A look at how private messengers handle key changes. *Medium*. Available at `https://medium.com/@pepelephew/a-look-at-how-private-messengers-handle-key-changes-5fd4334b809a`, 2017.

[Nao91] M. Naor. Bit commitment using pseudorandomness. *Journal of Cryptology*, 4(2):151–158, 1991.

[NSS06] M. Naor, G. Segev, and A. Smith. Tight bounds for unconditional authentication protocols in the manual channel and shared key models. In *Advances in Cryptology – CRYPTO'06*, pages 214–231, 2006.

[NSS08] M. Naor, G. Segev, and A. D. Smith. Tight bounds for unconditional authentication protocols in the manual channel and shared key models. *IEEE Transactions on Information Theory*, 54(6):2408–2425, 2008.

[PLF03] A. S. Patrick, A. C. Long, and S. Flinn. Hci and security systems. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1056–1057, 2003.

[PM16] T. Perrin and M. Marlinspike. The double ratchet algorithm, 2016. Available at `https://signal.org/docs/specifications/doubleratchet/doubleratchet.pdf` (accessed 16-May-2018).

[PPV08] O. Pandey, R. Pass, and V. Vaikuntanathan. Adaptive one-way functions and applications. In *Advances in Cryptology – CRYPTO '08*, pages 57–74, 2008.

[PR08] R. Pass and A. Rosen. New and improved constructions of nonmalleable cryptographic protocols. *SIAM Journal on Computing*, 38(2):702–752, 2008.

[PV06] S. Pasini and S. Vaudenay. An optimal non-interactive message authentication protocol. In *Topics in Cryptology – CT-RSA '06*, pages 280–294, 2006.

[PW10] R. Pass and H. Wee. Constant-round non-malleable commitments from sub-exponential one-way functions. In *Advances in Cryptology – EUROCRYPT '10*, pages 638–655, 2010.

[RMS18] P. Rösler, C. Mainka, and J. Schwenk. More is less: On the end-to-end security of group chats in Signal, WhatsApp, and Threema. In *Proceedings of the 3nd IEEE European Symposium on Security and Privacy (EuroS&P)*, 2018.

[RS84] R. L. Rivest and A. Shamir. How to expose an eavesdropper. *Communications of the ACM*, 27(4):393–395, 1984.

[RS18] L. Rotem and G. Segev. Out-of-band authentication in group messaging: Computational, statistical, optimal. In *Advances in Cryptology – CRYPTO '18*, pages 63–89, 2018.

[SKH17] M. Schliep, I. Kariniemi, and N. Hopper. Is Bob sending mixed signals? In *Proceedings of the 2017 on Workshop on Privacy in the Electronic Society*, pages 31–40, 2017.

[Tela] Telegram. End-to-end encrypted voice calls – key verification. Available at `https://core.telegram.org/api/end-to-end/voice-calls#key-verification` (accessed 16-May-2018).

[Telb] Telegram. End-to-end encryption. Available at `https://core.telegram.org/api/end-to-end` (accessed 16-May-2018).

[Telc] Telegram. FAQ for the technically inclined – hash collisions for Diffie-Hellman keys. Available at `https://core.telegram.org/techfaq#hash-collisions-for-diffie-hellman-keys` (accessed 16-May-2018).

[Vau05] S. Vaudenay. Secure communications over insecure channels based on short authenticated strings. In *Advances in Cryptology – CRYPTO'05*, pages 309–326, 2005.

[Vib] Viber encryption overview. Available at `https://www.viber.com/app/uploads/Viber-Encryption-Overview.pdf` (accessed 16-May-2018).

[Wee10] H. Wee. Black-box, round-efficient secure computation via non-malleability amplification. In *Proceedings of the 51st Annual IEEE Symposium on Foundations of Computer Science*, pages 531–540, 2010.

[Wha] WhatsApp encryption overview. Available at `https://www.whatsapp.com/security/WhatsApp-Security-Whitepaper.pdf` (accessed 16-May-2018).

[Wik] Wikipedia. Instant messaging. Available at `https://en.wikipedia.org/wiki/Instant_messaging` (accessed 16-May-2018).

A Non-Malleable Commitment Schemes

As discussed in Section 2, we follow the indistinguishability-based definition of Lin and Pass [LP11], though we find it convenient to consider non-malleability with respect to content, other than with respect to identities. Let $\mathsf{Com} = (C, R)$ be a statistically-binding commitment scheme, and let $\lambda \in \mathbb{N}$ be the security parameter. Consider an efficient adversary A that gets an auxiliary input $z \in \{0,1\}^*$ (in addition to the security parameter) and participates in the following "man-in-the-middle" experiment. A takes part in a single "left" interaction and a single "right" interaction: In the left interaction, A interacts with the committer C, and receives a commitment to a value v. Denote the resulting commitment (transcript of the interaction) by c. In the right interactions, A interacts with the receiver R, resulting in some commitment $\widehat{c}$. We define the related value $\widehat{v}$ in the following manner. If $\widehat{c} = c$, if $\widehat{c}$ is not a valid commitment, or if $\widehat{c}$ can be opened (information-theoretically speaking) to more than one value, we let $\widehat{v} = \perp$ (note that by the statistical binding property of Com, the latter case only happens with negligible probability). Otherwise, $\widehat{v}$ is the unique value to which $\widehat{c}$ may be opened. Let $\mathsf{mim}^A_{\mathsf{Com}}(v, z)$ denote the random variable that includes the value $\widehat{v}$ and A's view at the end of the afore-described experiment.

Definition A.1. Let A and D be a pair of algorithms. We define the advantage of (A, D) in breaking the non-malleability of Com, with respect to security parameter $\lambda \in \mathbb{N}$ as

$$\mathsf{Adv}^{A,D}_{\mathsf{Com}-\mathsf{NM}}(\lambda) \stackrel{\mathsf{def}}{=} \max_{v,v' \in \{0,1\}^\lambda} \left\{ \Pr\left[D(1^\lambda, \mathsf{mim}^A_{\mathsf{Com}}(v, z)) = 1\right] - \Pr\left[D(1^\lambda, \mathsf{mim}^A_{\mathsf{Com}}(v', z)) = 1\right] \right\}.$$

We say that a statistically-binding commitment scheme is *non-malleable* if for any pair of probabilistic polynomial-time algorithms (A, D) there exists a negligible function $\nu = \nu(\cdot)$ such that $\mathsf{Adv}^{A,D}_{\mathsf{Com}-\mathsf{NM}}(\lambda) \leq \nu(\lambda)$ for all sufficiently large $\lambda \in \mathbb{N}$.

For completeness, we also provide a formal definition for the hiding property of commitment schemes, of which we make use in Section 6 (though in general, it is easy to see that a non-malleable commitment scheme vis-á-vis Definition A.1 is also hiding). We first define the following experiment, which we denote by $\mathsf{Hiding}^A_{\mathsf{Com}}(\lambda, b)$ for every $\lambda \in \mathbb{N}$ and for each $b \in \{0,1\}$:

1. $v_0, v_1, \mathsf{st} \leftarrow A(1^\lambda)$.
2. $A(\mathsf{st})$ interacts with C (the commitment algorithm of Com) to receive a commitment to v_b, and outputs some state st'.
3. $b' \leftarrow A(\mathsf{st}')$.
4. The output of the experiment is b'.

The hiding property is then captured by the following definition.

Definition A.2. For an algorithm A and a commitment scheme Com, we define the advantage of A in breaking the hiding of Com with respect to security parameter $\lambda \in \mathbb{N}$ as

$$\mathsf{Adv}^A_{\mathsf{Com-Hiding}}(\lambda) \stackrel{\mathsf{def}}{=} \left|\Pr\left[\mathsf{Hiding}^A_{\mathsf{Com}}(\lambda, 0) = 1\right] - \Pr\left[\mathsf{Hiding}^A_{\mathsf{Com}}(\lambda, 1) = 1\right]\right|.$$

We say that a commitment scheme is *hiding* if for any probabilistic polynomial-time algorithm A there exists a negligible function $\nu = \nu(\cdot)$ such that $\mathsf{Adv}^A_{\mathsf{Com-Hiding}}(\lambda) \leq \nu(\lambda)$ for all sufficiently large $\lambda \in \mathbb{N}$.