Out-of-Band Authenticated Group Key Exchange: From Strong Authentication to Immediate Key Delivery

Moni Naor[*] Lior Rotem[†] Gil Segev[†]

Abstract

Given the inherent ad-hoc nature of popular communication platforms, *out-of-band authenticated key-exchange protocols* are becoming widely deployed: Key exchange protocols that enable users to detect man-in-the-middle attacks by manually authenticating one short value. In this work we put forward the notion of *immediate key delivery* for such protocols, requiring that even if some users participate in the protocol but do not complete it (e.g., due to losing data connectivity or to other common synchronicity issues), then the remaining users should still agree on a shared secret. A property of a similar flavor was introduced by Alwen, Correti and Dodis (EUROCRYPT '19) asking for immediate decryption of messages in user-to-user messaging *while assuming that a shared secret has already been established* – but the underlying issue is crucial already during the initial key exchange and goes far beyond the context of messaging.

Equipped with our immediate key delivery property, we formalize strong notions of security for out-of-band authenticated group key exchange, and demonstrate that the existing protocols either do not satisfy our notions of security or are impractical (these include, in particular, the protocols deployed by Telegram, Signal and WhatsApp). Then, based on the existence of any passively-secure key-exchange protocol (e.g., the Diffie-Hellman protocol), we construct an out-of-band authenticated group key-exchange protocol satisfying our notions of security. Our protocol is inspired by techniques that have been developed in the context of fair string sampling in order to minimize the effect of adversarial aborts, and offers the optimal tradeoff between the length of its out-of-band value and its security.

[*] Department of Computer Science and Applied Mathematics, Weizmann Institute of Science, Rehovot 76100, Israel. Email: `moni.naor@weizmann.ac.il`.

[†] School of Computer Science and Engineering, Hebrew University of Jerusalem, Jerusalem 91904, Israel. Email: `{lior.rotem,segev}@cs.huji.ac.il`.

Contents

1 Introduction

A fundamental challenge in cryptography is that of generating shared secrets in communication networks that are susceptible to man-in-the-middle attacks. When a public-key infrastructure is available, this task has been thoroughly studied, and many protocols have been suggested (see Section 1.2). The question remains, however, of how to agree on an initial secret when connections are formed ad-hoc, and a public-key infrastructure is impractical to maintain. Such scenarios include, for example, communication platforms offering end-to-end encrypted messaging services, audio calls or video calls [PM16, Telb, Wha, Vib, BMO$^+$19, FMB$^+$16, BSJ$^+$17, CCD$^+$17, KBB17, PR18a, JS18, CGCG$^+$18, DV19, SH19, ACD19, JMM19], as well as secure pairing of IoT devices (e.g., [Duq18, LSA19, Blu19]).

Out-of-band authenticated key exchange. Given that man-in-the-middle attacks are impossible to detect without any additional setup, one approach often taken is to provide users/devices with the ability to communicate "out-of-band", assuming that they have access to an external channel through which they can information-theoretically authenticate short values. Equipped with such an external channel, one can then rely on out-of-band authenticated key-exchange protocols: Protocols that are tailored to using both a standard insecure channel and a low-bandwidth out-of-band channel, and enable users to bootstrap the limited resource of information-theoretical authentication provided by the out-of-band channel in order to establish shared secrets while detecting man-in-the-middle attacks. Such an approach is taken by most communication platforms providing end-to-end encryption and by protocols for pairing of IoT devices (see the references above).

The out-of-band channel typically corresponds to having the users compare with each other a short value displayed by their devices (or having a single user compare a string displayed by all paired devices in the context of pairing of IoT devices), but can in fact be based on a variety of real-world assumptions (e.g., [MPR05, GSS$^+$06, SEK$^+$06, MG07, KFR09, Duq18, LSA19, Tela, Wha]). In most implementations the "manual" flavor of the out-of-band channel introduces a tradeoff between the effort invested by the users and the security guarantees: A longer out-of-band value may enable better security in principal, but also incurs a more intensive user effort, thus hurting usability and ultimately security.

Non-interactive vs. interactive protocols. As in standard key exchange, there are two main flavors of out-of-band authenticated key-exchange protocols: *Non-interactive* protocols in which each user sends at most one message and this message is sent independently of the other users' messages, and *interactive* protocols in which users may send more than one message and these messages may depend on other users' previously-sent messages.[1]

Non-interactive protocols are widely used by messaging platforms (e.g., WhatsApp and Signal [Wha]), since they do not require any two users to be online at any particular point in time. However, such protocols are inherently limited in the security they can provide – as we discuss in Section 1.3.

[1] These two flavors of protocols are sometimes referred to as "asynchronous" protocols vs. "synchronous" protocols (e.g., in the specific context of messaging protocols [PR18b, CGCG$^+$18, ACD19, JMM19] – to which we do not at all limit ourselves in this work). As discussed below we follow the more standard terminology of non-interactive protocols vs. interactive protocols since the standard model of synchronous computation in distributed computing is much more restrictive than the standard model required for interactive cryptographic protocols in general, and for the protocols considered in this paper in particular (e.g., a global clock synchronizing the entire execution of the protocol among the various parties is not required [Lyn96, AW04]).

Our focus: Immediate key delivery in interactive protocols. In various popular scenarios, such as voice and video calls or pairing of IoT devices, the users or devices participating in the protocol are typically expected to remain online throughout its execution. In these scenarios, unlike in messaging applications, interactive protocols may be used in order to ensure stronger security guarantees (e.g., security which is independent of the adversary's concrete running time). This is the case, for example, in the out-of-band key-exchange protocol Telegram uses for its voice calls [Tela] (see also Section 4.2).

An additional approach to constructing interactive out-of-band authenticated key exchange protocols is to first run any passively-secure key-exchange protocol, and then use an *out-of-band message authentication protocol* in order to authenticate its transcript [Vau05, PV06b, RS18a]. An out-of-band message authentication protocol allows for the authentication of long messages while using the out-of-band channel only to information-theoretically authenticate one short value. Although any solution to the general task of establishing shared keys must inherently rely on computational assumptions, out-of-band *message authentication* protocols may provide unconditional information-theoretical security. By now there is a sound theoretical understanding (i.e., protocols and matching lower bounds) of out-of-band message authentication protocols, in both the user-to-user and the group settings, as well as practically-relevant protocols in both settings [Vau05, PV06a, NSS06, RS18a, NRS18] – these works were indeed motivated by the task out-of-band authenticated key exchange.

In contrast, out-of-band authenticated key exchange has been studied in the user-to-user setting (e.g., [PV06b, Lin09]) but has been left without any rigorous treatment in the group setting. In particular, when considering the security of out-of-band authenticated key exchange in the group setting, a crucial requirement is that even if some users participate the protocol but do not complete it, then the remaining users should still agree on a shared key that will enable them to start interacting in an end-to-end encrypted manner. We refer to this property as *immediate key delivery*. Alwen, Correti and Dodis [ACD19] have recently suggested a property of such flavor to which they referred to as "immediate decryption". Their work was in the context of messaging protocols *assuming that a shared secret key has already been established* – but the underlying issue is crucial already during the initial key exchange.

Providing immediate key delivery is a challenge that arises only in the interactive setting, as it is trivially guaranteed by any non-interactive protocol (but, as discussed above, such protocols provide somewhat weaker security guarantees). Although interactive protocols are suitable for scenarios in which users are typically expected to remain online, protocols still have to address cases where some of the users do not complete the protocol. Otherwise, for example, any user who loses connectivity prevents the successful completion of the protocol by the remaining users. Moreover, if a protocol does not offer immediate key delivery, then it becomes very easy for an attacker to prevent the users from agreeing on a shared secret, by simply blocking all outgoing communication from a single user in the group.

The significant and practical importance of immediate key delivery, together with various other security considerations for out-of-band protocols, motivate an in-depth examination of out-of-band authenticated key exchange, including formal definitions and protocols that satisfy them.

1.1 Our Contributions

Motivated by the above-described state of affairs, we present the following contributions:

- We suggest a framework for analyzing out-of-band authenticated group key-exchange protocols, capturing crucial security and functionality properties that arise in the group setting for out-of-band protocols.

- We observe that the existing approaches for constructing out-of-band authenticated key-exchange protocols either do not satisfy our (standard) notions of security or are impractical (already for rather small groups). This situation highlights the fact that it is highly non-trivial to satisfy our notions of security while keeping the out-of-band value short.
- Based on the existence of any passively-secure user-to-user key-exchange protocol (e.g., the Diffie-Hellman protocol), we construct an out-of-band authenticated group key-exchange protocol satisfying our notions of security, and offering the optimal tradeoff between the length of its out-of-band value and its security. Moreover, for some possible use-cases, instantiating our protocol in the random-oracle models leads to a concrete and efficient protocol.

In what follows we briefly discuss each of these contributions, and the reader is referred to Section 1.3 for a more elaborate and technical overview. First, however, we would like to emphasize the following aspects regarding our work:

- Our work does not propose a communication protocol but rather a mechanism for establishing shared secrets – which can then be used as the initial step of any such protocol, while avoiding the assumption that the users have already established shared secrets, or that all public keys have already been authenticated.
- As a first step in strengthening out-of-band authenticated group key exchange, we focus on providing fundamental security properties for stand-alone executions. We leave to future work the important task of extending such properties to accommodate concurrent executions, in line with long line of research on authenticated key exchange (see Section 1.2).
- As mentioned above, our work focuses on the interactive setting, as non-interactive protocols trivially guarantee immediate key delivery on the one hand, but provide weaker security guarantees on the other hand.

Modeling out-of-band authenticated group key exchange. We consider a group of users communicating over a completely-insecure channel that is susceptible to man-in-the-middle attacks, and in addition assume that *some* user of the group can information-theoretically authenticate one short value to all other users who have not yet aborted, over the out-of-band channel (note that we do not make any assumptions as to the particular identity of that user).[2]

Within this communication model (which we formally define in Section 3), we put forth a realistic framework and notions of security for out-of-band authenticated key-exchange in the group setting, considering the following three requirements:

- **Pseudorandomness:** If a man-in-the-middle adversary does not interfere with the communication, the resulting shared key should be computationally indistinguishable from an independent and uniformly-distributed key given the transcript of the protocol *which includes the out-of-band value.*
- **Man-in-the-middle detection:** If a man-in-the-middle adversary does interfere with the communication, this should be detected except with probability $\epsilon(\lambda) + \mathsf{negl}(\lambda)$, where ϵ is a pre-determined function of the security parameter $\lambda \in \mathbb{N}$, and negl is a negligible function which may depend on the adversary.

  Most importantly, ϵ must be fixed for all adversaries, and in particular it is not allowed to depend on the adversary's on-line or off-line running time or space usage – as the *effective*

[2] The way that the out-of-band value is propagated through the group might be different; e.g., if some users recognize the voice of one user in a voice group call, and the other users recognize the voice of another user, then informing all users of the out-of-band value requires the two recognized users to read it out loud. Our model, in which there is a single out-of-band value and a single user who sends it, can always be easily translated to such situations.

length of the out-of-band value might not always be sufficiently long (e.g., when executed by "lazy users" who may not consider the out-of-band value in its entirety [NRS18]).

- **Immediate key delivery:** Even if a subset of the parties aborts the execution of the protocol before its completion, the remaining parties should still agree on a shared key (the abort decisions may be determined adversarially throughout the execution of the protocol). This requirement significantly strengthens the standard correctness requirement of key-exchange protocols, and achieving this requirement is the core technical contribution of our work.

Note that the pseudorandomness and man-in-the-middle detection requirements are relevant already in the user-to-user setting (and we consider natural extensions of these requirements from passively-secure protocols to out-of-band protocols), and that the immediate key delivery is a new requirement that we introduce in the group setting.

Existing protocols do not meet our requirements. We show that even though the three requirements listed above seem fairly standard as far as cryptographic definitions go, they are not met by existing protocols. Namely, we observe that each of the out-of-band authenticated key-exchange protocols deployed by Signal, WhatsApp and Telegram, and that the protocol suggested by Rotem and Segev [RS18a] does not satisfy at least one of the aforementioned requirements.

Already in the user-to-user setting, we show that the protocol deployed by Telegram does not satisfy our pseudorandomness requirement, and that the protocols deployed by Signal and WhatsApp do not satisfy our man-in-the-middle detection requirement. In the group setting, even though these protocols provide immediate key delivery, they are non-scalable in terms of the length of the out-of-band value, since they require running a user-to-user protocol with each member of the group separately, resulting in an out-of-band value whose length depends linearly on the size of the group. For example, in a group of size 32, in order to get 60 bits of security, the out-of-band value in these protocols has to be of length at least $31 \times 60 = 1860$ bits (i.e., the initiator of the key exchange has to compare at least 560 decimal digits with other users). In the group setting, the protocol of Rotem and Segev, which relies on the above-mentioned transcript-authentication approach [PV06b] is more practical, and satisfies our pseudorandomness and man-in-the-middle detection properties, but does not provide immediate key delivery.

We stress that as mentioned above, some of these protocols have their advantages in particular use cases. However, the fact that none of them provide both optimal security guarantees per our security notion and also immediate key delivery in the group setting, exemplifies in our view the difficulty that lies in satisfying all of these requirements simultaneously and highlights the challenges that need to be overcome. Looking ahead, the main reason that immediate key delivery is challenging to obtain without substantially increasing the length of the out-of-band value, is that an adversary may choose a subset of aborting users out of an *exponential* number of such subsets – and this allows the adversary significant control over the execution of the protocol.

From strong(er) message authentication to out-of-band authenticated key exchange. We construct an out-of-band authenticated group key-exchange protocol which satisfies our notions of security, based on any passively-secure user-to-user key-exchange protocol. Moreover, we prove that our protocol enjoys the optimal tradeoff (within lower-order terms) between the length of its out-of-band value and the probability of an active attack going undetected.[3]

[3] Our protocol provides such an optimal tradeoff even when executed by "lazy users", who may not consider the out-of-band value in its entirety, as recently formalized by Naor et al. [NRS18].

Theorem 1.1 (informal)**.** *Assuming the existence of any passively-secure user-to-user key-exchange protocol, then for any functions $n = n(\lambda)$ and $\ell = \ell(\lambda)$ there exists an out-of-band authenticated key-exchange protocol for groups of $n(\lambda)$ users, with an out-of-band value of length $\ell(\lambda)$ bits such that any active man-in-the-middle attack is detected except with probability $\epsilon(\lambda) \leq 2(n(\lambda)-1)\cdot(1/2 + o(1))^{\ell(\lambda)}$, where $\lambda \in \mathbb{N}$ in the security parameter.*

Our protocol is based on a general transformation that takes any passively-secure key-exchange protocol and produces an out-of-band authenticated key-exchange protocol. Concretely, we observe that although the above-mentioned transcript-authentication approach (i.e., using a group out-of-band message authentication protocol in order to authenticate the transcript of a passively-secure group key-exchange protocol) fails to guarantee immediate key delivery, this can be overcome if the underlying message authentication protocol provides a property we refer to as *immediate message delivery* (the precise transformation requires overcoming various additional challenges). We construct such a strengthened out-of-band message authentication protocol by starting from the basic structure of the group protocol of Rotem and Segev, and incorporating within it techniques from the realm of fair multi-party string-sampling protocols (i.e., protocols in which even if some parties abort then the remaining parties sample a "relatively unbiased" string [ABC$^+$85, Cle86] – see Section 1.3 for more details). We view this as our main technical contribution.

A benefit of the fact that we present our protocol as a general transformation while relying on generic building blocks, is that this enables for a much greater modularity in its instantiation. In particular, this allows for the reliance on post-quantum secure assumptions as opposed to the currently deployed protocols by Telegram, Signal and WhatsApp that are based on the Decisional Diffie-Hellman assumption.

1.2 Related Work

The problem of detecting man-in-the-middle attacks in key exchange protocols has been studied extensively in various models (see, for example, [BR93a, BCK98, Sho99, BPR00, CK01, LLM07] for user-to-user protocols, and [BR95, BCP$^+$01b, KY03] for group protocols). Our setting and definitions bare some resemblance in particular to that of password-authenticated key exchange (PAKE; see [Jab96, BMP00, KOY01, GL03, AFP05] and the references therein), in that in both cases the security is inherently a function of the unpredictability of some short value (the out-of-band value in our case, and the shared password in the case of PAKE).

In particular, in the PAKE setting, Fiore, Vasco and Soriente [FVS17] considered the problem of "partitioned group key exchange" which is conceptually somewhat similar to the problem we consider in this paper: Designing a PAKE protocol with the guarantee that even if some users provide a wrong password then all users who provided the correct password should still agree on a shared key. The main difference, however, between this problem and our work is the correctness requirement: Fiore et al. assume that all users are on-line and follow the instructions of the protocol, and require that all users who provide the same password output the same key, whereas we assume that some users may adversarially abort the protocol at any stage and require that all other users output the same key. This difference, together with the substantial differences of the two authentication models, lead to completely different technical challenges (and solutions).

More generally, although there are natural similarities between the various authentication models, there are several key differences between our work and the lines of works mentioned above. Namely, to provide immediate key delivery, our model and definitions accommodate users who abort prematurely, whereas most of the works on authenticated key-exchange are either in the user-to-user setting, or consider groups that remain static (i.e., no users are added or removed) *throughout the* *execution of the protocol.* Some works (e.g., [BCP01a, BCP02]) do consider dynamic groups that may change over time and their shared secret needs to be updated, but not the scenario that we are studying of users who abort *during the execution of the protocol itself.* In that respect, our work is focused on initial key exchange (and its authentication), and we do not explicitly consider the task of adding or removing users in later stages. In any case, adding a user to the group while communicating with only a single existing member of the group, as is the case with the deployed protocols, can and must be authenticated using a user-to-user out-of-band protocol. This approach can also be used to add users who went offline during the initial setup, which again must require an additional out-of-band verification.

In the out-of-band model, most previous works concentrated on message authentication [RS84, Vau05, NSS06, PV06a, RS18a, NRS18], with the exception of Pasini and Vaudenay [PV06b] and Lindell [Lin09], who studied key exchange explicitly, but only in the user-to-user setting. Pasini and Vaudenay followed the transcript-authentication paradigm described above, while Lindell focused on analyzing the specific Bluetooth v2.1 comparison-based key-exchange protocol.

1.3 Overview of Our Security Notions and Construction

In this section we first discuss the motivation underlying our three security requirements (which were briefly mentioned in Section 1.1 and are formally defined in Section 4). Next, we overview the "transcript authentication" approach for constructing an out-of-band authenticated group key-exchange protocol (which serves as our starting point), and point out its current limitations. Then, we provide a high-level overview of our construction and of its proof of security.

Our notions of security. Our work puts forward extensions of the standard notions of pseudorandomness and man-in-the-middle detection that are tailored to out-of-band protocols, as well as introduces the notion of immediate key delivery, as discussed in Section 1.1.

Requirement 1: Pseudorandomness given the out-of-band value. The out-of-band channel is assumed to provide authenticity for one short value, but it is not assumed to provide any form of secrecy, and thus all communication over this channel may be completely visible to an adversary. Thus, the natural extension of the standard pseudorandomness requirement for key-exchange protocol must consider an adversary observing both the communication over the insecure channel and over the out-of-band channel. For such an adversary, the resulting shared key should be computationally indistinguishable from an independent and uniformly-distributed key.

Requirement 2: Adversary-independent man-in-the-middle detection. The probability of detecting an active man-in-the-middle attack depends (at least) on the bit-length ℓ of the out-of-band authenticated value (in Section 4 we provide a simple proof showing that any protocol can be undetectably attacked with probability essentially $\epsilon = n \cdot 2^{-\ell}$). We require that active attacks are detected with probability that depends on the protocol itself (e.g., $\epsilon = n \cdot 2^{-\ell}$), and do not scale in a meaningful manner with the adversary's on-line or off-line running time or space usage. For example, our requirement rules out protocols that out-of-band authenticate an 80-bit value, and an adversary that can execute 2^{40} computations of a certain hash function can break its security with probability $2^{40} \cdot 2^{-80}$.

This property is even more crucial when considering the likely scenario of "lazy users", as formalized by Naor et al. [NRS18], where users may consider only a short sub-string of the out-of-band authenticated value. This renders the "effective length" of the out-of-band value much shorter than its actual length ℓ. For example, if the security that a protocol provides is $T \cdot 2^{-\ell}$, where T is roughly the running time of the adversary and ℓ is the length of the "de-facto out-of-band value", then if the users consider, say 20 bits from the out-of-band value, an adversary running in reasonable time can break the security of the protocol quite easily (instead of having the protocol still guarantee the best-possible security of $\epsilon = 2^{-20}$).

Requirement 3: Immediate key delivery. We require that even if a subset of the parties aborts the execution of the protocol before its completion, the remaining parties should still agree on a shared key. This is a crucial requirement not only due to the above-described nature of mobile-based messaging, but even more in order to protect against devastating adversarial denial-of-service attacks that are undetected by other users. For example, in the recently-suggested protocol of Rotem and Segev [RS18a], an adversary that can simply block the communication going out of just one user, can make sure that the other users will never agree on a shared key, leaving the group either completely vulnerable or utterly useless.

Although this property is a functionality-focused one, our main technical challenge in this work is to obtain it while retaining a good (and preferably optimal) level of security. As we discuss in length in Section 4.2 and in the continuation of this section, simple attempts to add immediate key delivery to the protocol of Rotem and Segev make it completely insecure.

Interaction is essential. Satisfying all three requirements simultaneously requires an interactive protocol. The pseudorandomness requirement may be satisfied both by interactive and by non-interactive protocols (under suitable assumptions). The third requirement, immediate key delivery, is trivially satisfied by any non-interactive protocol, but as mentioned above, the second requirement – adversary-independent MitM detection – cannot be satisfied by such protocols. Concretely, in Section 4 we show that for any non-interactive protocol and for any running time T, there exists a successful man-in-the-middle attacker that runs in time essentially T and is undetected with probability $\min\{1/3, \Omega(T \cdot 2^{-\ell})\}$, where ℓ is the bit-length of the out-of-band value. In this light our goal is to come up with interactive protocols that simultaneously guarantee all three requirements, while retaining a short out-of-band value.

Our starting point: The "transcript authentication" approach. As mentioned in Section 1.1, the out-of-band group key-exchange protocols deployed by WhatsApp, Signal and Telegram provide immediate key delivery, but impose a heavy burden on the users: These protocols require running a user-to-user protocol with each member of the group separately, resulting in an out-of-band value whose length depends linearly on the size of the group. In addition, recall that these protocols do not satisfy our two additional security requirements, and thus they do not seem to be promising starting points for designing protocols satisfying our goals.

Our starting point is the transcript-authentication approach described above [PV06b], while using the out-of-band group message authentication protocol of Rotem and Segev [RS18a]. Roughly speaking, this approach suggests running any passively-secure group key-exchange protocol,[4] and afterwards to authenticate its transcript via the following out-of-band message authentication protocol:

1. P_1 chooses $r_S \leftarrow \{0,1\}^\ell$ and commits to $\mathsf{trans} \| r_S$ to all other users, where trans is the transcript of the key-exchange protocol from P_1's point of view.

[4] Most naively, the initiator P_1 can execute a user-to-user protocol (such as the Diffie-Hellman protocol) with each other user P_i for obtaining a shared key k_i. Then, P_1 will sample a random key k and encrypt it to each other user P_i using the key k_i.

2. $P_2, \ldots, P_n$ cooperatively choose a string r_R: Each P_i chooses $r_i \leftarrow \{0,1\}^\ell$ and commits to it to all other users. After all users have committed, each P_i decommits to reveal r_i, and sets $r_R = \bigoplus_{i \in \{2,\ldots,n\}} r_i$.

3. P_1 decommits to reveal r_S, and then out-of-band authenticates to $\sigma = r_S \oplus r_R$. Each of the other users accepts (and outputs the key agreed upon in the key exchange step) if and only if σ and trans are both consistent with her view.

This protocol falls short of satisfying our definition for out-of-band authenticated group key exchange in two respects. First, our definition requires that an active attack will be detected except with some pre-determined probability, but the only guarantee provided by the protocol of Rotem and Segev is that if trans is inconsistent with the view of some P_i, then with high probability this P_i will reject. It might still be the case though, that an active adversary modifies messages sent during the out-of-band message authentication phase described above.

This problem may be addressed in a simple manner (and in this specific protocol it is not that devastating to begin with): Instead of using the out-of-band message authentication protocol in order to authenticate the transcript trans of the group key exchange, P_1 samples a pair $(\mathsf{sk}, \mathsf{vk})$ of signing and verification keys for a *one-time strongly unforgeable signature scheme* (See Section 2); then uses the out-of-band message authentication protocol to authenticate vk to the other users; and finally uses sk to sign the transcripts of *both* the key-exchange protocol and the out-of-band message authentication protocol.

The second, more fundamental, problem is that the protocol of Rotem and Segev does not provide "immediate key delivery", even if the underlying passively-secure key-exchange protocol does provide it[5]. This is true since a user who identifies a deviation from the protocol (including a premature abort) terminates and rejects. In order for the out-of-band authenticated group key-exchange protocol to provide immediate key delivery, the out-of-band message authentication protocol needs to satisfy a similar property, to which we refer as *immediate message delivery.* This property essentially requires that even if a subset of the receivers in the protocol abort, but the execution is otherwise honest, the rest of the receivers should still accept the message.

Alas, the lacuna in the out-of-band message authentication protocol of Rotem and Segev, due to which it does not provide immediate message delivery, is far from being a mere technicality. To see why, consider what happens if we simply ignore aborting users, and take r_R to be the exclusive-or of only the r_i's of the users who opened their commitments. This might provide immediate key delivery, but gravely hurts the security of the protocol, by giving the man-in-the-middle adversary the ability to choose which commitments to open to each P_i *after observing* r_i. Concretely, in Section 4.2, we present an attack showing that this change exponentially increases the forgery probability from roughly $n \cdot 2^{-\ell}$ to roughly $2^n \cdot 2^{-\ell}$, where n is the number of users in the group and ℓ is the length of the out-of-band value.

The underlying issue with the protocol of Rotem and Segev (explaining the exponential increase), is that a man-in-the-middle adversary interacting with, say P_i, can choose to abort any subset of $\{P_2, \ldots, P_n\} \setminus \{P_i\}$ towards P_i, before forwarding the decommitments of the users in this subset to P_i. Even if the interaction with P_i is otherwise honest, each possible aborting subset might induce a different value for r_R in the view of P_i. This enables a man-in-the-middle adversary to substantially "steer" the r_R that P_i computes, such that the attack will go undetected.

[5] The naive protocol described in Footnote 4 is a passively-secure protocol with immediate key delivery: Even if some user aborts then the remaining users still output the key k chosen by P_1.

Providing immediate message delivery: Attempt I. As a first attempt to limit the additional power provided to the adversary by allowing aborts, consider a "restart-after-abort" variant of the Rotem-Segev protocol, in which after an abort by any of the users, the remaining users start a fresh execution of the protocol. Intuitively, now the adversary has no incentive to abort more then a single user in each execution of the original Rotem-Segev protocol, and the identity of the particular user who aborts (if such a user exists) is of no consequence due to the symmetry of the protocol. Hence, instead of exponentially many choices of aborting subsets, in each execution of the original Rotem-Segev protocol the adversary effectively has only two (abort or not).

The problem with this approach however, is that now the adversary has up to $n-1$ attempts to break the security of the Rotem-Segev protocol, yielding a forgery probability of roughly $n^2 \cdot 2^{-\ell}$. This is much better than the $2^n \cdot 2^{-\ell}$ forgery probability of the "vanilla" Rotem-Segev protocol, but still quite far from optimal: The forgery probability grows quadratically with the number of users in the group, which may be significant in large groups, and as we show below, this can be avoided. Moreover, when the protocol is executed by lazy users as discussed above (who may not consider the out-of-band value in its entirety [NRS18]), the effective value of ℓ might be relatively small, resulting in a substantial forgery probability. Instead, we are interested in a solution that provides security which is optimal with respect to the size of the group and to the length of the out-of-band value, so that it provides reasonable security even for lazy users (looking ahead, our protocol provides the *optimal* tradeoff within lower-order terms between the length of its out-of-band value and its security even when executed by lazy users).

Providing immediate message delivery: Attempt II. In light of the above, and inspired by techniques from protocols for fair string sampling, we construct a group out-of-band message authentication protocol that provides immediate message delivery – while retaining an optimal level of security (within lower order terms). The main idea behind our protocol is to replace the manner r_R is chosen in the protocol of Rotem and Segev, with a way which is more resilient to aborts. By that, intuitively speaking, we mean that even a man-in-the-middle adversary interacting with some P_i, and can simulate control over all users but P_i in that interaction, cannot force the r_R computed by P_i to hit the particular value that it needs in order for the attack to go unnoticed by P_i.

Concretely, instead of selecting r_R in "one shot" as done in the protocol of Rotem and Segev, in our protocol it is chosen in more gradual manner, which considerably limits the effect of adversarial aborts. Concretely, the users iteratively choose T ℓ-bit values $r_{R,1}, \ldots, r_{R,T}$ (where T is a parameter of the protocol) one after the other, in T consecutive iterations. In the tth iteration $r_{R,t}$ is chosen by the remaining users among $P_2, \ldots, P_n$ (i.e., the users who have not yet aborted) in the same manner as r_R is chosen in the protocol of Rotem and Segev. Finally, the value of r_R in our protocol is then taken to be the bit-wise majority of $r_{R,1}, \ldots, r_{R,T}$: The kth bit of r_R is the majority bit over the kth bits of $r_{R,1}, \ldots, r_{R,T}$. We refer the reader to Sections 5 and 6 for a complete and formal description of our protocol.

With this change, analyzing our new protocol proves to be technically involved, as a man-in-the-middle adversary has numerous more possible synchronizations to impose on an execution of the protocol. Nevertheless, we manage to prove that when the commitment scheme used in our protocol is statistically-binding and concurrent non-malleable (see Section 2), then the forgery probability is bounded roughly by $n \cdot (1/2 + n/\sqrt{T})^{\ell}$. Setting the parameter T to be $n^2 \cdot \omega(1)$ (e.g., $n^2 \cdot \log^* \lambda$), we get that the forgery probability is $n \cdot (1/2 + o(1))^{\ell}$, matching our lower bound of $\min\{1/3, \Omega(n \cdot 2^{-\ell})\}$ (see Section 4) within lower order terms.

Overview of our proof of security. We provide a brief and high level overview of the proof of unforgeability of our out-of-band message authentication protocol, ignoring various technical difficulties and focusing on the main ideas. We prove that for every $i \in \{2, \ldots, n\}$, if the man-in-the-middle changes the verification key sent to P_i in the beginning of our out-of-band authenticated key-exchange protocol,[6] then the probability that P_i will not detect this interference (i.e., will not output $\perp$) is upper bounded by roughly $(1/2 + n/\sqrt{T})^{\ell}$. We do so by considering all possible synchronizations that a man-in-the-middle might impose on an execution of the protocol relative to P_i, and bound the probability of forgery in each of them relying on the statistical binding and on the concurrent non-malleability of the underlying commitment scheme. We manage to partition all possible such synchronizations into two families, and handle each one separately. For simplicity of presentation in this overview, we focus on the case where $\ell = 1$ (i.e., the initiator P_1 out-of-band authenticates a single bit), and the reader is referred to Section 6 for our formal proof of security.

Proof of security: Case I. In the first family of synchronizations, P_1 decommits to reveal r_S before P_i receives the first round of commitments from $P_2, \ldots, P_{i-1}, P_{i+1}, \ldots, P_n$. In this case, by the statistical binding, the values of r_S and r_R according to the view of P_1 and the value of r_S according to the view of P_i, have all been determined by the time P_i receives the first round of commitments. Hence, in order for P_i to not reject, the man-in-the-middle must make sure that r_R according to the view of P_i hits the unique value $r_R^* \in \{0, 1\}$ which is the exclusive-or of the three aforesaid determined values. We bound the probability that $r_R = r_R^*$ using the concurrent non-malleability of the commitment scheme, where the heart of the proof lies in two lemmata, a computational lemma and a statistical lemma.

The computational lemma states that no strategy of the man-in-the-middle can result in a noticeably-greater probability that $r_R = r_R^*$ then the following strategy, denoted M_{opt}: (1) In each iteration $t \in [T]$ and for every P_j that has not yet aborted according to the view of P_i, send P_i a commitment to the value 1 from P_j; (2) If the sampled value in this round $r_{R,t}$ is equal to r_R^* (when no user aborts), then open all commitments; (3) Otherwise, open all commitments except for that of the minimal-index user P_j that has not yet aborted (since P_j committed to the value 1, this is guaranteed to flip the bit $r_{R,t}$, so that it is equal to r_R^*).

We prove that this strategy is optimal in forcing $r_R = r_R^*$ (within a negligible additive factor) via a hybrid argument: We start with any other man-in-the-middle adversary M and gradually change its strategy to M_{opt}, iteration by iteration, proving that the probability that $r_R = r_R^*$ cannot decrease by too much in each change, or the concurrent non-malleability of the commitment scheme is violated. Concretely, we consider $T + 1$ hybrids, where the adversary in the tth hybrid, denoted by M_t, plays as M in the first $T - t$ iterations and as M_{opt} in the remaining t iterations. Observe, that in order for M_t to succeed with noticeably-greater probability than M_{t+1} (in forcing $r_R = r_R^*$), it must be the case that in the tth hybrid, $\Pr[r_{R,t} = r_R^*]$ is noticeably greater than $1/2$. This contradicts the concurrent non-malleability of the underlying commitment scheme: Intuitively, this is because in an ideal experiment in which the bit contributed by P_i in the tth iteration is sampled anew just before P_i decommits, it holds that $\Pr[r_{R,t} = r_R^*] = 1/2$.

The statistical lemma states that the optimal adversary described above, M_{opt}, succeeds in forcing $r_R = r_R^*$ with probability no greater than roughly $1/2 + n/\sqrt{T}$. To prove this, it is convenient to think of an equivalent experiment, in which $r_{R,1}, \ldots, r_{R,T}$ are first sampled uniformly from $\{0, 1\}$, and then the adversary is given the option to flip $n - 2$ of them.[7] In this experiment, the adversary can force

[6] Our protocol in Section 6 is a general-purpose out-of-band message authentication protocol. For concreteness in this overview, we focus on the case where the message to be authenticated is the verification key sampled by P_1, as is the case in our out-of-band authenticated key-exchange protocol.

[7] For the case $\ell = 1$, this is indeed equivalent. For the general case of $\ell \geq 1$, this may only add power to the

$r_R = r_R^*$ if and only if $|\{t \in [T] : r_{R,t} = r_R^*\}| \geq T/2 - n + 2$. We observe that $|\{t \in [T] : r_{R,t} = r_R^*\}|$ is a random variable distributed according to the binomial distribution with parameters $1/2$ and T. We then use the symmetry of this distribution and the fact that every value in its support is obtained with probability no greater than roughly $1/\sqrt{T}$, in order to bound the probability that $|\{t \in [T] : r_{R,t} = r_R^*\}| \geq T/2 - n + 2$ by roughly $1/2 + n/\sqrt{T}$.

Proof of security: Case II. In the second family of possible synchronizations, P_1 decommits to reveal r_S after P_i has received at least one round (and possibly many) of commitments from the other users. Denote the last round of commitments received by P_i before P_1 decommits by $t^* \in [T]$. In this case the man-in-the-middle adversary's situation is worse than in Case 1: The hiding and the concurrent non-malleability of the commitment scheme, and in particular of the commitment by P_1 to the value r_S, imply that the adversary cannot hope to force any of $r_{R,1}, \ldots, r_{R,t^*}$ to be equal to r_R^* with probability noticeably greater than $1/2$. This is because in an ideal experiment in which r_S is sampled anew just before P_1 decommits, it holds that $\Pr[r_{R,t} = r_R^*] = 1/2$ for every $t \in [t^*]$ and independently of the other rounds, and irrespective of the identity of the aborted users in rounds $1, \ldots, t^*$. Intuitively speaking, it follows that the adversary is only more limited than in the previous case, as she can use her "abort quota" effectively only in rounds $t^* + 1, \ldots, T$, and hence the forgery probability in this case cannot be noticeably greater than that of the previous case.

1.4 Paper Organization

The remainder of this paper is organized as follows. In Section 2 we review the basic notation and definitions of standard cryptographic primitives used in this paper. In Section 3 we review the out-of-band communication model, and in Section 4 we present our notions of security for out-of-band authenticated group key-exchange protocols. In Section 5 we show that any passively-secure user-to-user key-exchange protocol can be transformed into an out-of-band authenticated group key-exchange protocol that satisfies our notions of security, and in Section 6 we construct an out-of-band message authentication protocol with immediate message delivery, which is the main building block underlying our transformation. Finally, in Appendix A we revisit the user-to-user setting, and present a concrete and practically-relevant protocol that satisfies our notions of security in the random-oracle model (with an out-of-band of optimal length).

2 Preliminaries

In this section we present the basic notions and standard cryptographic tools 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 $s \in \{0,1\}^*$ and an index $i \in [|s|]$, we let s_i (sometimes we may write $(s)_i$) denote the ith bit of s. 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 $\nu(n) \leq 1/p(n)$.

Passively-secure key-exchange protocols. A key-exchange protocol enables two or more parties to jointly generate a key that is computationally indistinguishable from a uniformly-distributed key given the transcript of the protocol. For such a protocol $\Pi = \langle P_1, \ldots, P_n \rangle$ for $n \geq 2$ parties we

adversary, and hence the probability that $r_R = r_R^*$ can only increase. Hence, in the general case as well, bounding the probability that $r_R = r_R^*$ in this experiment bounds the probability that the man-in-the-middle adversary can force $r_R = r_R^*$.

let $\langle P_1, \ldots, P_n \rangle (1^\lambda)$ denote be the distribution over $(n+1)$-tuples $(\mathsf{trans}, \mathsf{k}_1, \ldots, \mathsf{k}_n)$ induced by an honest execution of the protocol, where each party receives the security parameter 1^λ as input, trans denotes the transcript of the execution, and $\mathsf{k}_1, \ldots, \mathsf{k}_n$ are the outputs of $P_1, \ldots, P_n$, respectively. The following definition captures the standard correctness and security (with respect to passive adversaries) requirements of key-exchange protocols:

Definition 2.1. A key-exchange protocol over key space $\mathcal{K} = \{\mathcal{K}_\lambda\}_{\lambda \in \mathbb{N}}$ is a probabilistic polynomial-time protocol $\Pi = \langle P_1, \ldots, P_n \rangle$ satisfying the following requirements:

- **Correctness:** For every $\lambda \in \mathbb{N}$ it holds that

$$\Pr_{(\mathsf{trans}, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \langle P_1, \ldots, P_n \rangle (1^\lambda)} [\mathsf{k}_1 = \cdots = \mathsf{k}_n \in \mathcal{K}_\lambda] = 1.$$

- **Security against passive adversaries:** For every probabilistic polynomial-time algorithm D there exists a negligible function $\nu(\cdot)$ such that

$$\mathsf{Adv}^{\mathsf{KE}}_{\Pi,D}(\lambda) \overset{\mathsf{def}}{=} \left| \Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}_1) = 1 \right] - \Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}) = 1 \right] \right| \leq \nu(\lambda)$$

  for all sufficiently large $\lambda \in \mathbb{N}$, where the probabilities are taken over the choice of $(\mathsf{trans}, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \langle P_1, \ldots, P_n \rangle (1^\lambda)$ and $\mathsf{k} \leftarrow \mathcal{K}_\lambda$.

One-time strongly-unforgeable signature schemes. A signature scheme is a triplet $(\mathsf{KG}, \mathsf{Sign}, \mathsf{Vrfy})$ of probabilistic polynomial-time algorithms, where KG is the key-generation algorithm, Sign is the signing algorithm, and Vrfy is the verification algorithm. The key-generation algorithm KG receives as input the security parameter 1^λ, and outputs a verification key vk and a signing key sk. The signing algorithm Sign receives as input a signing key sk and a message m, and outputs a signature σ. The verification algorithm Vrfy receives as input a verification key vk, a message m, and a signature σ, and outputs a bit b. In terms of functionality, for any $\lambda \in \mathbb{N}$ and message m it should hold that $\mathsf{Vrfy}(\mathsf{vk}, m, \mathsf{Sign}(\mathsf{sk}, m)) = 1$, where $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KG}(1^\lambda)$. In terms of security, we consider the following standard notion of one-time strong unforgeability.

Definition 2.2. A signature scheme $\Pi = (\mathsf{KG}, \mathsf{Sign}, \mathsf{Vrfy})$ is *one-time strongly unforgeable* if for any probabilistic polynomial-time adversary $A = (A_1, A_2)$ there exists a negligible function $\nu(\cdot)$ such that $\Pr\left[\mathsf{Expt}^{\mathsf{1Sig}}_{\Pi,A}(\lambda) = 1 \right] \leq \nu(\lambda)$ for all sufficiently large $\lambda \in \mathbb{N}$, where the experiment $\mathsf{Expt}^{\mathsf{1Sig}}_{\Pi,A}(\lambda)$ is defined as follows for any $\lambda \in \mathbb{N}$:

1. $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KG}(1^\lambda)$.
2. $(m, \mathsf{state}) \leftarrow A_1(1^\lambda, \mathsf{vk})$.
3. $(m^*, \sigma^*) \leftarrow A_2(\mathsf{state}, \sigma)$, where $\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, m)$.
4. If $\mathsf{Vrfy}(\mathsf{vk}, m^*, \sigma^*) = 1$ and $(m^*, \sigma^*) \neq (m, \sigma)$ then output 1, and otherwise output 0.

Non-malleable commitment schemes. In this paper we rely on the notion of statistically-binding concurrent 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. Lin and Pass [LP11] and Goyal [Goy11] have shown that constant-round concurrent non-malleable commitment schemes can be constructed from any one-way function (the round complexity was further improved by Ciampi et al. [COS+17] to just

4 rounds). From a more practical perspective, such schemes can be constructed efficiently in the random-oracle model [BR93b]. For further information regarding non-malleable and concurrent non-malleable commitment schemes see, for example, [DDN00, CIO98, FF00, CF01, PR05, PR08, LPV08] and the references therein.

Intuitively speaking, a concurrent non-malleable commitment scheme has the following guarantee: Any efficient adversary cannot use commitments to m values $v_1, \ldots, v_k$ in order to produce commitments to values $\widehat{v_1}, \ldots, \widehat{v_k}$ that are "non-trivially" related to $v_1, \ldots, v_k$. More formally, Let $\mathsf{Com} = (C, R)$ be a statistically-binding commitment scheme, and let $k = k(\cdot)$ be a function of the security parameter $\lambda \in \mathbb{N}$, bounded by some polynomial. 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 k "left" interactions and in k "right" interactions: In the left interactions, A interacts with the committer C, and receives commitments to values $v_1, \ldots, v_k$. Denote the resulting commitments (transcripts of the interaction) by $c_1, \ldots, c_k$. In the right interactions, A interacts with the receiver R, resulting in k commitments $\widehat{c_1}, \ldots, \widehat{c_k}$. We define k related values $\widehat{v_1}, \ldots, \widehat{v_k}$ in the following manner. For every $i \in [k]$, if $\widehat{c_i} = c_j$ for some $j \in [k]$, if $\widehat{c_i}$ is not a valid commitment, or if $\widehat{c_i}$ can be opened to more than one value, we let $\widehat{v_i} = \perp$ (note that by the statistical binding property of Com, the latter case only happens with negligible probability). Otherwise, $\widehat{v_i}$ is the unique value to which $\widehat{c_i}$ may be opened. Denoting by $\mathbf{v} = (v_1, \ldots, v_k)$ the vector of values to which A receives commitments in the left interactions, we let $\mathsf{mim}^{A}_{\mathsf{Com}}(\mathbf{v}, z)$ denote the random variable that includes the values $\widehat{v_1}, \ldots, \widehat{v_k}$ and A's view at the end of the afore-described experiment.

Definition 2.3. Let A and D be a pair of algorithms. We define the advantage of (A, D) with respect to security parameter $\lambda \in \mathbb{N}$ as

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

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

3 The Out-of-Band Communication Model

In this section we review the out-of-band communication model as well as the notion of an out-of-band message authentication protocol [Vau05, PV06a, RS18a].

The out-of-band channel and man-in-the-middle attacks. As formalized by Vaudenay and by Naor et al. in the user-to-user setting [Vau05, NSS08] and extended by Rotem and Segev to the group setting [RS18a], interaction among users in the out-of-band communication model occurs over two types of channels: Insecure channels and a low-bandwidth authenticated channel (referred to as the "out-of-band channel"). It is assumed that a man-in-the-middle adversary has complete control over the insecure channels: The adversary can read, delay and remove messages sent by the parties over the insecure channels, as well as insert new messages at any point in time. One may consider various topologies for the network of insecure channels. For our protocols we assume the minimal such topology: An insecure channel between some user (e.g., the initiator of the protocol) and any other user in the group (i.e., a star network).

As for the out-of-band channel, it is assumed that there exists some user that can out-of-band authenticate one short value to all other users in the group. This value is assumed to be authenticated but not secret: The adversary may read or remove this message for some or all users, and may delay it for different periods of time for different users, but cannot modify it in an undetectable manner. We stress that our requirement of the out-of-band channel is a rather weak one: We only require that there exists *some* user that can out-of-band authenticate a short value to the rest of the group, and we do not apply any restrictions as to who that user is.

In addition, we do not make any synchronization assumption regarding the out-of-band channel: We do not assume that all users have to be on-line when the out-of-band value is transmitted. Specifically, any subset of the users may be off-line at that time, and any user that comes back on-line will be able to make her own decision regarding the authenticity of the execution if and when the out-of-band value reaches her (recall that the attacker can block the out-of-band value to all or to some of the users).

Out-of-band message authentication. An out-of-band message authentication protocol enables a sender S to authenticate a message m, which may be chosen by the adversary, to all other users $R_1, \dots, R_n$ in the group ($n = 1$ is the user-to-user setting, whereas $n \geq 2$ is the group setting). Once the execution is completed, each receiver R_i outputs either some message $\widehat{m}_i$ or the unique symbol $\perp$ implying rejection. The following definition was introduced by Rotem and Segev [RS18a], naturally extending those of Vaudenay and Naor et al. [Vau05, NSS08]:

Definition 3.1. Let $\ell = \ell(\lambda), \epsilon = \epsilon(\lambda)$ and $n = n(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. A group (ℓ, ϵ)-out-of-band message authentication protocol for $n(\lambda)$ receivers and message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda \in \mathbb{N}}$ is an $(n(\lambda) + 1)$-party protocol, in which S sends at most $\ell(\lambda)$ bits over the out-of-band channel, and the following requirements hold:

1. **Correctness:** For every $\lambda \in \mathbb{N}$, for every $m \in \mathcal{M}_\lambda$ and every $i \in [n(\lambda)]$ it holds that $\Pr[\widehat{m}_i = m] = 1$, where the probability is over the randomness of the parties in an honest execution of the protocol.

2. **Unforgeability:** For every probabilistic polynomial-time adversary M there exists a negligible function $\nu(\cdot)$ such that for every input message $m \in \mathcal{M}_\lambda$ chosen by the adversary for the sender S it holds that
$$\Pr[\exists i \in [n(\lambda)] : \widehat{m}_i \notin \{m, \perp\}] \leq \epsilon(\lambda) + \nu(\lambda)$$
for all sufficiently large $\lambda \in \mathbb{N}$, where the probability is taken over the randomness of the parties and the randomness of M in an execution of the protocol with M as the man-in-the-middle adversary.

Existing out-of-band message authentication protocols. In the user-to-user setting, Vaudenay [Vau05] constructed a protocol in which the forgery probability ϵ is upper bounded by $2^{-\ell}$, where ℓ is the bit-length of the out-of-band authenticated value, and Vaudenay and Pasini [PV06a] proved a matching lower bound. In the group setting, considering a strengthened version of Definition 3.1, Rotem and Segev [RS18a] constructed a protocol for groups of size n in which the forgery probability is bounded by $(n-1) \cdot 2^{-\ell}$, and proved a matching lower bound. Both protocols can be based on the existence of any one-way function [LN06, RS18a] via non-malleable commitments.

4 Out-of-Band Authenticated Group Key Exchange

In this section we first present our strengthened notion of security for out-of-band authenticated key-exchange protocols (Section 4.1). Then, we show that the protocols deployed by Signal, WhatsApp and Telegram do not satisfy it already in the user-to-user setting[8], and that the protocol obtained via the "out-of-band transcript authentication" approach does not satisfy it in the group setting (Section 4.2).

4.1 Pseudorandomness, MitM Detection and Immediate Key Delivery

Our strengthened notion of security for out-of-band key-exchange protocols consists of three requirements: Pseudorandomness and man-in-the-middle detection that are relevant already in the user-to-user setting, and immediate key delivery that we introduce in the group setting (as discussed in Section 1.3). Our pseudorandomness and man-in-the-middle detection requirements are natural extensions of these requirements to the out-of-band model:

- If a man-in-the-middle adversary does not interfere with the communication, the resulting shared key should be computationally indistinguishable from an independent and uniformly-distributed key given the transcript of the protocol *which includes the out-of-band value.*
- If a man-in-the-middle adversary does interfere with the communication, this should be detected except with probability $\epsilon(\lambda) + \mathsf{negl}(\lambda)$, where ϵ is a pre-determined function of the security parameter $\lambda \in \mathbb{N}$, and negl is a negligible function which may depend on the adversary. Most importantly, ϵ must be fixed for all adversaries (e.g., it is not allowed to depend on the adversary's on-line or off-line running time or space usage).

  Our security definition requires that an active attack is detected by all users on the receiving end of the out-of-band channel, for whom communication to or from them has been actively modified by the attacker. The task of notifying all other users (who are still online at the end of the execution) of an active attack can be achieved, for example, by assuming that all users can send an "out-of-band feedback" signal to all other members, indicating an attack. Observe that such an assumption (or an assumption of the same nature) is essential in order for all users to detect an active attack, as without it (i.e., with only a single user that can send a message out-of-band and all other communication being subject to man-in-the-middle manipulation) an active attack in which some of the users do not identify the attack is always possible.

Our immediate key delivery requirement significantly strengthens the standard correctness requirement of key-exchange protocols stated in Definition 2.1: Even if a subset of the parties aborts the execution of the protocol before its completion, the remaining parties should still agree on a shared key. To capture this requirement, for an algorithm A and an n-party protocol π, we let $\mathsf{FailStopExec}(\pi, A, \lambda)$ denote the output of the following experiment:

1. Start an execution of π with joint input 1^λ.
2. For every $i \in [n] \setminus \{1\}$, before P_i sends a message v according to π, invoke $\mathsf{decision} \leftarrow A(1^\lambda, \mathsf{PartialTrans})$, where $\mathsf{decision} \in \{\mathsf{abort}, \mathsf{continue}\}$ and $\mathsf{PartialTrans}$ is the partial transcript of the execution up to this point. If $\mathsf{decision} = \mathsf{continue}$, P_i sends v, and the execution continues. If $\mathsf{decision} = \mathsf{abort}$, P_i aborts and the execution continues without P_i.
3. The output of $\mathsf{FailStopExec}(\pi, A, \lambda)$ is a $(n + 1)$-tuples $(\mathsf{AbortSet}, \mathsf{v}_1, \ldots, \mathsf{v}_n)$, where $\mathsf{AbortSet}$ denotes the set of indices of aborted parties at the end of the execution and v_i is the output of P_i if $P_i \notin \mathsf{AbortSet}$ and $\mathsf{v}_i = \perp$ otherwise.

Note that in order for this experiment to be well defined, the protocol π has to be well defined for any possible pattern of aborts. In that case, this experiment is well defined both for group key-exchange protocols (including passively-secure ones) and for group authentication protocols (looking

[8] In Appendix A we show that there is a simple and practically-relevant user-to-user protocol that does satisfy our notion of security (and offers the optimal trade-off between the length of its out-of-band authenticated value and its man-in-the-middle detection probability).

ahead, this experiment will enable us to formalize a notion of "immediate message delivery" for authentication protocols). When π is a key-exchange protocol, we use $\mathsf{k}_1, \dots, \mathsf{k}_n$ instead of $\mathsf{v}_1, \dots, \mathsf{v}_n$ to denote the output keys of the users.

Also note that we assume that P_1 does not prematurely abort throughout the execution. This is essential, as we will assume without loss of generality that P_1 is the user that can send a short message over the out-of-band channel. Hence, if P_1 aborts prior to sending the out-of-band value, no meaningful security can be guaranteed. Practically speaking, in the context of messaging platforms, P_1 who initiates the key-exchange protocol is typically the first party to send an encrypted message to the group. Hence, if P_1 aborts, the need for a shared key is postponed until another message is sent (at which point, the users will execute the out-of-band group key-exchange protocol when initiated a potentially different user).

Our definition, provided below, relies on the following notation. We denote by $\mathsf{MitMExec}(\pi, M, \lambda)$ the distribution over $(n+1)$-tuples $(\mathsf{view}_M, \mathsf{k}_1, \dots, \mathsf{k}_n)$ induced by an execution of the protocol with a man-in-the-middle M, where the adversary and all parties run on input 1^λ, and view_M is the view of M at the end of the protocol ($\mathsf{k}_1, \dots, \mathsf{k}_n$ are defined as before). For every $i \in \{2, \dots, n\}$, let Active_i be the event in which the adversary actively changes the communication from or to P_i; i.e., by either modifying or removing messages sent from or to P_i or by inserting new message to or from P_i.

We also define the event Active: Informally, Active is the event in which the man-in-the-middle adversary M changes the communication among the parties in any manner that goes beyond simulating an abort by a subset of the parties (by simulating an abort by a party, we mean blocking all messages sent by that party from some point onward). More formally, let $q = q(\lambda)$ be a bound on the number of rounds in an execution of π on joint input 1^λ. For an execution according to $\mathsf{MitMExec}(\pi, M, \lambda)$, we denote by $\mathsf{Msgs}_i = (m_{i,1}, \dots, m_{i,q})$ the vector of messages sent (in order) by P_i, where if P_i has sent t messages for $t < q$, we denote $m_j = \perp$ for every $j \in \{t+1, \dots, q\}$. Similarly, we denote by $\widehat{\mathsf{Msgs}_i} = (\widehat{m_{i,1}}, \dots, \widehat{m_{i,q}})$ the vector of messages received (in order) by parties other than P_i, as messages from P_i. We denote by Active the event in which for some $i \in [n]$, there exits $t \in [p]$ such that $m_{i,t} \neq \widehat{m_{i,t}}$ and at least one of the following conditions hold: (1) $\widehat{m_{i,t}} \neq \perp$; or (2) There exists $t' > t$ such that $m_{i,t'} \neq \perp$.

Definition 4.1. Let $n = n(\lambda), \ell = \ell(\lambda)$ and $\epsilon = \epsilon(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. A group out-of-band (ℓ, ϵ)-key-exchange protocol over key space $\mathcal{K} = \{\mathcal{K}_\lambda\}_{\lambda \in \mathbb{N}}$ for a group of size $n = n(\lambda)$ is an n-party protocol $\pi = \langle P_1, \dots, P_n \rangle$, in which P_1 sends at most $\ell(\lambda)$ bits over the out-of-band channel and the following requirements hold:

- **Immediate key delivery:** For every $\lambda \in \mathbb{N}$ and every probabilistic polynomial-time algorithm A, it holds that
$$\Pr\left[\forall i \in [n(\lambda)] \setminus \mathsf{AbortSet} : \mathsf{k}_1 = \mathsf{k}_i \in \mathcal{K}_\lambda\right] = 1$$
where $(\mathsf{AbortSet}, \mathsf{k}_1, \dots, \mathsf{k}_n) \leftarrow \mathsf{FailStopExec}(\pi, A, \lambda)$.

- **Man-in-the-middle detection:** For any probabilistic polynomial-time algorithm M there exists a negligible function $\nu(\cdot)$ such that
$$\Pr\left[\exists i \in \{2, \dots, n(\lambda)\} : \mathsf{Active}_i \wedge \mathsf{k}_i \neq \perp\right] \leq \epsilon(\lambda) + \nu(\lambda)$$
for all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_1, \dots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$.

- **Pseudorandomness:** For any probabilistic polynomial-time algorithms M and D there exists a negligible function $\nu(\cdot)$ such that
$$\left|\Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}_1) = 1\right] - \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}) = 1\right]\right| \leq \nu(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_1, \dots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$ and $\mathsf{k} \leftarrow \mathcal{K}_\lambda$.

We note that when $n = 2$, Definition 4.1 captures the user-to-user setting. In this case, the immediate key delivery property simply reverts back to the standard correctness property of key-exchange protocols as it appears in Definition 2.1. In addition, note that the immediate key delivery property is defined with respect to an efficient algorithm A, but our construction provides immediate key delivery even in the case where A is unbounded and receives access to the random coins of the users.

Interaction is essential. As mentioned in Section 1.3, no non-interactive protocol can satisfy our man-in-the-middle detection requirement. To see why that is, let π be such a non-interactive protocol and let P_i be any user participating in the protocol (other than the one in charge of sending the out-of-band value). Consider the following man-in-the-middle attacker, that can compute the secret key outputted by P_i:

1. The attacker forwards all messages sent by the users to all users participating in the protocol, other than to P_i. Let σ be the true out-of-band value sent as a result.

2. Let m_i be the message sent by P_i. The attacker samples T independent tuples of messages $M_{-i}^{(1)}, \dots, M_{-i}^{(T)}$ for the other users participating in the protocol, and computes the T resulting out-of-band values $\sigma^{(1)}, \dots, \sigma^{(T)}$ (i.e., $\sigma^{(j)}$ is the out-of-band value in the execution in which the messages sent are m_i and the messages in $M_{-i}^{(j)}$).

3. If for any $j^* \in [T]$ it holds that $\sigma^{(j^*)} = \sigma$, then the attacker sends the messages in the tuple $M_{-i}^{(j^*)}$ to P_i (as the messages sent by the other users in the protocol). Otherwise, the attacker has failed and she terminates the attack.

Observe that if the attacker completes the attack, then: (1) She knows the randomness used to sample the messages in $M_{-i}^{(j^*)}$, so she can compute the key outputted by P_i; and (2) The view of P_i is the same as in an honest execution in which the messages are m_i and the messages in $M_{-i}^{(j^*)}$, so the attack is undetected by P_i.

Hence, in order to analyze the probability that this attack is successful, we need to look at the probability that there exists such an index j^*. It turns out that we can bound this probability for any choice of m_i, so let us fix m_i and look at σ and $\sigma^{(1)}, \dots, \sigma^{(T)}$ when P_i sends m_i. These are $T+1$ independent samples from the distribution over the out-of-band value in a random execution of π, conditioned on P_i sending the message m_i. One can verify that the probability that there exists an index $j^* \in [T]$ such that $\sigma^{(j^*)} = \sigma$ is minimized when this conditional distribution is the uniform distribution over $\{0,1\}^\ell$. For this distribution, the probability that there exists such an index j^* – and that the attack is successful – is at least $\min\{1/3, \Omega(T \cdot 2^{-\ell})\}$. The complete analysis is identical to the one which appears in the proof of Lemma 4.3 below.

The required length of the out-of-band value. Theorem 4.2 states that any out-of-band group key-exchange protocol for n users with an out-of-band value of length ℓ bits can be undetectably attacked by an efficient man-in-the-middle adversary with probability roughly $n \cdot 2^{-\ell}$. As discussed in Section 1.3, a key goal in the out-of-band model is to construct protocols offering the optimal trade-off between their security and the length of their out-of-band authenticated value, and our protocols in this paper offer this optimal trade-off both in the user-to-user setting and in the group setting (within lower order terms).

Theorem 4.2. *Let $\ell = \ell(\lambda), n = n(\lambda)$ and $\epsilon = \epsilon(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. For any out-of-band (ℓ, ϵ)-key-exchange protocol (over any key space $\mathcal{K}$) for a group of size $n(\lambda)$, there exists a negligible function $\nu(\cdot)$ such that*

$$\epsilon(\lambda) \geq \min\left\{\frac{1}{3}, \frac{n(\lambda)-1}{4} \cdot 2^{-\ell}\right\} - \nu(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$.

Proof. Let $\ell = \ell(\lambda), n = n(\lambda)$ and $\epsilon = \epsilon(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$, and let $\pi = \langle P_1, \ldots, P_n \rangle$ be an (ℓ, ϵ)-out-of-band authenticated key-exchange protocol. Consider the following man-in-the-middle adversary, denoted by M.

The Man-in-the-Middle Adversary M

Input: The security parameter 1^λ.
The attack:

1. Run an execution of the protocol with P_1, while simulating $P_2, \ldots, P_n$ honestly. Let σ denote the out-of-band value in this execution.
2. For each $i \in \{2, \ldots, n\}$, run an execution of the protocol with P_i, while simulating all other users honestly. Let $\widehat{\sigma_i}$ denote the out-of-band value computed by the simulated P_1 in this execution.
3. Forward σ to P_i for each $i \in \{2, \ldots, n\}$.

It holds that:

$$\begin{aligned}
&\Pr\left[\exists i \in \{2, \ldots, n\} : \mathsf{Active}_i \wedge \mathsf{k}_i \neq \bot\right] \\
&\quad \geq \Pr\left[\forall j \in \{2, \ldots, n\} : \mathsf{Active}_j \wedge \exists i \in \{2, \ldots, n\} : \mathsf{k}_i \neq \bot\right] \\
&\quad = 1 - \Pr\left[\exists j \in \{2, \ldots, n\} : \overline{\mathsf{Active}_j} \vee \forall i \in \{2, \ldots, n\} : \mathsf{k}_i = \bot\right] \\
&\quad \geq \Pr\left[\exists i \in \{2, \ldots, n\} : k_i \neq \bot\right] - \Pr\left[\exists i \in \{2, \ldots, n\} : \overline{\mathsf{Active}_i}\right] \\
&\quad \geq \Pr\left[\exists i \in \{2, \ldots, n\} : \widehat{\sigma_i} = \sigma\right] - \Pr\left[\exists i \in \{2, \ldots, n\} : \overline{\mathsf{Active}_i}\right] \qquad (4.1)
\end{aligned}$$

where all probabilities are over $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$ and (4.1) follows from the correctness of the protocol. The theorem then follows from the following two lemmata.

Lemma 4.3. $\Pr\left[\exists i \in \{2, \ldots, n\} : \widehat{\sigma_i} = \sigma\right] \geq \min\left\{1/3, ((n-1)/4) \cdot 2^{-\ell}\right\}$*, where the probability is taken over* $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$*.*

Proof of Lemma 4.3. The proof is taken mutatis mutandis from [RS18b]. Let $\Sigma = \{\Sigma_\lambda\}_{\lambda \in \mathbb{N}}$ denote the ensemble of random variables corresponding to the out-of-band value in an honest execution of $\langle P_1, \ldots, P_n \rangle$ on joint input 1^λ. Observe that σ as well as $\widehat{\sigma_2}, \ldots, \widehat{\sigma_n}$ are all sampled according to Σ. Hence, for every $\lambda \in \mathbb{N}$ and for every $\mathcal{I} \subseteq [n(\lambda)]$, it holds that

$$\Pr_{(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)}\left[\forall i \in \mathcal{I} : \widehat{\sigma_i} = \sigma\right] = \sum_{\sigma \in \{0,1\}^\ell} \left(\Pr_{\sigma \leftarrow \Sigma_\lambda}[\sigma]\right)^{|\mathcal{I}|+1}.$$

Let $k = n - 1$. The inclusion-exclusion principle now yields that

$$\Pr\left[\exists i \in \{2, \ldots, n\} : \widehat{\sigma_i} = \sigma\right] = \sum_{i=1}^{k} (-1)^{i+1} \cdot \binom{k}{i} \cdot \left(\sum_{\sigma \in \{0,1\}^\ell} \left(\Pr_{\sigma \leftarrow \Sigma_\lambda}[\sigma]\right)^{i+1}\right).$$

The above probability is minimized when the distribution of Σ_λ over a random execution of the protocol as described above is uniform. Hence, it holds that

$$\Pr\left[\exists i \in [k] \text{ s.t. } R_i \text{ outputs } \widehat{m_i^i}\right] \geq \sum_{i=1}^{k} (-1)^{i+1} \cdot \binom{k}{i} \cdot 2^{-i\cdot\ell}.$$

In order to bound this expression, we differentiate between two cases. First, consider the case where $k \geq 3 \cdot 2^{\ell-1}$. The above expression can be thought of in the following manner: k balls are independently thrown into 2^ℓ bins uniformly at random. Let B be the random variable denoting the number of balls in the first bin at the end to the experiment. Then, the expression we wish to bound is exactly $\Pr[B > 0]$. Let N be a geometric random variable denoting the number of balls thrown until a ball hits the first bin.[9] Then, by Markov's bound it holds that

$$\begin{aligned}
\Pr\left[\exists i \in [k] \text{ s.t. } R_i \text{ outputs } \widehat{m_i^i}\right] &\geq \Pr[B > 0] \\
&= \Pr[N \leq k] \\
&= 1 - \Pr[N > k] \\
&\geq 1 - \frac{\mathbb{E}[N]}{k} \\
&\geq 1 - \frac{2^\ell}{3 \cdot 2^{\ell-1}} = \frac{1}{3}.
\end{aligned}$$

Now consider the case where $k < 3 \cdot 2^{\ell-1}$. In this case, it holds that

$$\begin{aligned}
&\Pr\left[\exists i \in [k] \text{ s.t. } R_i \text{ outputs } \widehat{m_i^i}\right] \\
&\qquad \geq \sum_{i=1}^{k} (-1)^{i+1} \cdot \binom{k}{i} \cdot 2^{-i\cdot\ell} \\
&\qquad \geq \sum_{i=1}^{\lfloor k/2 \rfloor} \left(\binom{k}{2i-1} \cdot 2^{-(2i-1)\cdot\ell} - \binom{k}{2i} \cdot 2^{-2i\cdot\ell} \right) \\
&\qquad = \sum_{i=1}^{\lfloor k/2 \rfloor} \left(1 - \frac{\binom{k}{2i} \cdot 2^{-2i\cdot\ell}}{\binom{k}{2i-1} \cdot 2^{-(2i-1)\cdot\ell}} \right) \cdot \binom{k}{2i-1} \cdot 2^{-(2i-1)\cdot\ell} \\
&\qquad = \sum_{i=1}^{\lfloor k/2 \rfloor} \left(1 - \frac{k-2i+1}{2i} \cdot 2^{-\ell} \right) \cdot \binom{k}{2i-1} \cdot 2^{-(2i-1)\cdot\ell} \\
&\qquad > \sum_{i=1}^{\lfloor k/2 \rfloor} \left(1 - \frac{k}{2} \cdot 2^{-\ell} \right) \cdot \binom{k}{2i-1} \cdot 2^{-(2i-1)\cdot\ell} \\
&\qquad > \frac{1}{4} \cdot \sum_{i=1}^{\lfloor k/2 \rfloor} \binom{k}{2i-1} \cdot 2^{-(2i-1)\cdot\ell} > \frac{k}{4} \cdot 2^{-\ell}.
\end{aligned}$$

Putting the two cases together, and substituting $k = n - 1$, we get

$$\Pr[\exists i \in \{2, \ldots, n\} : \widehat{\sigma_i} = \sigma] \geq \min\left\{ \frac{1}{3}, \frac{n(\lambda) - 1}{4} \cdot 2^{-\ell} \right\}$$

concluding the proof of Lemma 4.3. ■

[9] For the sake of defining N, balls are thrown until a ball is thrown into the first bin.

Lemma 4.4. *There exists a negligible function $\nu(\cdot)$ such that*

$$\Pr\left[\exists i \in \{2, \ldots, n\} : \overline{\mathsf{Active}_i}\right] \leq \nu(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the probability is taken over $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$.

Proof of Lemma 4.4. We prove that for every $i \in \{2, \ldots, n\}$ there exists a negligible function $\nu'(\cdot)$ such that

$$\Pr\left[\overline{\mathsf{Active}_i}\right] \leq \nu'(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, and the lemma follows by a taking a union bound over all $i \in \{2, \ldots, n\}$.

Let $i \in \{2, \ldots, n\}$. Note that $\overline{\mathsf{Active}_i}$ occurs if and only if the transcript of the execution with P_i according to P_i's view is consistent with the communication between P_j and the simulated P_i in the execution M runs with P_j, for every $j \in [n] \setminus \{i\}$. Denote by $\mathsf{trans}_i(i)$ the transcript according to P_i's view, and by $\mathsf{trans}_{i\leftrightarrow j}(j)$ the communication between P_j and P_i according to P_j's view. Finally, for a transcript $t \in \{0,1\}^*$ of an execution of the protocol according to P_i's view, and for $j \in [n] \setminus \{i\}$, we denote by $t_{i\leftrightarrow j}$, the segment of the transcript that corresponds to the communication between P_i and P_j.

According to the definition of the man-in-the-middle adversary, $\mathsf{trans}_i(i)$ and $\{\mathsf{trans}_{i\leftrightarrow j}(j)\}_{j\in[n]\setminus\{i\}}$ are all sampled independently from distributions induced by an honest execution of the protocol. Hence, for every $\lambda \in \mathbb{N}$, it holds that:

$$\begin{aligned}
\Pr\left[\overline{\mathsf{Active}_i}\right] &= \Pr\left[\forall j \in [n] \setminus \{i\} : (\mathsf{trans}_i(i))_{i\leftrightarrow j} = \mathsf{trans}_{i\leftrightarrow j}(j)\right] && (4.2)\\
&= \sum_{t\in\{0,1\}^*} \left(\Pr\left[\mathsf{trans}_i(i) = t\right] \cdot \prod_{j\in[n]\setminus\{i\}} \Pr\left[\mathsf{trans}_{i\leftrightarrow j}(j) = t_{i\leftrightarrow j}\right] \right) && (4.3)\\
&\leq \left(\max_{t^*\in\{0,1\}^*} \Pr\left[\mathsf{trans}_i(i) = t^*\right] \right) \cdot \sum_{t\in\{0,1\}^*} \prod_{j\in[n]\setminus\{i\}} \Pr\left[\mathsf{trans}_{i\leftrightarrow j}(j) = t_{i\leftrightarrow j}\right] \\
&\leq \max_{t^*\in\{0,1\}^*} \Pr\left[\mathsf{trans} = t^*\right] && (4.4)
\end{aligned}$$

where the probabilities in (4.2) are over $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$, and from (4.3) onward, all probabilities are over an honest execution of π.

Claim 4.5. *There exists a negligible function $\nu'(\cdot)$ such that*

$$\max_{t^*\in\{0,1\}^*} \Pr\left[\mathsf{trans}_i(i) = t^*\right] \leq \nu'(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$.

Proof of Claim 4.5. By the correctness of π, there exists a (non-efficiently computable) function $K = \{K_\lambda : \{0,1\}^* \to \mathcal{K}_\lambda\}_{\lambda\in\mathbb{N}}$ mapping the transcript of an execution according to P_i's view to the corresponding key; i.e., for every $\lambda \in \mathbb{N}$ it holds that

$$\Pr_{(\mathsf{trans}, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \langle P_1, \ldots, P_n \rangle (1^\lambda)} \left[\mathsf{k}_1 = K(\mathsf{trans}_i(i))\right] = 1.$$

Now consider the following distinguisher D attacking the key pseudorandomness of π:

The Distinguisher D

Input: The security parameter 1^λ, $\mathsf{trans} \in \{0,1\}^*$ and k^*, where $(\mathsf{trans}, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \langle P_1, \ldots, P_n \rangle (1^\lambda)$, $\mathsf{k} \leftarrow \mathcal{K}_\lambda$ and $\mathsf{k}^* \in \{\mathsf{k}_A, \mathsf{k}\}$.
Non-uniform advice: $t_{\mathsf{max}} = \arg\max_{t \in \{0,1\}^*} \{\Pr[\mathsf{trans}_i(i) = t]\}$ and $k_{\max} = K(t_{\mathsf{max}})$, where the probability is taken over $(\mathsf{trans}, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \langle P_1, \ldots, P_n \rangle (1^\lambda)$.
The attack:

1. If $\mathsf{trans}_i(i) = t_{\mathsf{max}}$ and $\mathsf{k}^* = k_{\mathsf{max}}$ then output 1.
2. Otherwise, sample $b \leftarrow \{0,1\}$ and output b.

For simplicity, we described D in a non-uniform manner, but this is clearly not essential: If there exists a certain transcript $t_{\max}$ that is obtained with a non-negligible probability $1/p(\lambda)$ for some polynomial $p(\cdot)$, then D can sample $\lambda \cdot p(\lambda)$ transcripts to obtain a list of transcripts and their corresponding keys. With all but an exponentially small probability $\exp(-\lambda)$, the highly-probable transcript $t_{\max}$ will be in that list (together with its corresponding key).

We now continue to analyze the advantage of the above non-uniform distinguisher. Denote the event in which $\mathsf{trans}_i(i) = t_{\mathsf{max}}$ by E. We first consider the case where $\mathsf{k}^* = \mathsf{k}_1$:

$$\begin{aligned}
\Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}_1) = 1\right] &= \Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}_1) = 1 \middle| \mathsf{E}\right] \cdot \Pr[\mathsf{E}] \\
&\quad + \Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}_1) = 1 \middle| \overline{\mathsf{E}}\right] \cdot \Pr\left[\overline{\mathsf{E}}\right] \\
&= \Pr[\mathsf{E}] + \frac{\Pr\left[\overline{\mathsf{E}}\right]}{2}.
\end{aligned}$$

In the case where $\mathsf{k}^* = \mathsf{k} \leftarrow \mathcal{K}_\lambda$ it holds that:

$$\begin{aligned}
\Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}) = 1\right] &= \Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}) = 1 \middle| \mathsf{E}\right] \cdot \Pr[\mathsf{E}] \\
&\quad + \Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}) = 1 \middle| \overline{\mathsf{E}}\right] \cdot \Pr\left[\overline{\mathsf{E}}\right] \\
&= \Pr[\mathsf{E}] \cdot \left(\Pr[\mathsf{k} = k_{\mathsf{max}}] + \frac{\Pr(\mathsf{k} \neq k_{\mathsf{max}})}{2}\right) + \frac{\Pr\left[\overline{\mathsf{E}}\right]}{2} \\
&\leq \frac{3}{4} \cdot \Pr[\mathsf{E}] + \frac{\Pr\left[\overline{\mathsf{E}}\right]}{2} \qquad (4.5)
\end{aligned}$$

where (4.5) holds by the non-triviality of the key space; i.e., the fact that $|\mathcal{K}_\lambda| \geq 2$ for every $\lambda \in \mathbb{N}$. Hence, there exists a negligible function $\nu(\cdot)$ such that:

$$\begin{aligned}
\max_{t^* \in \{0,1\}^*} \Pr[\mathsf{trans} = t^*] &= \Pr[\mathsf{E}] \\
&\leq 4 \cdot \left(\Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}_1) = 1\right] - \Pr\left[D(1^\lambda, \mathsf{trans}, \mathsf{k}) = 1\right]\right) \\
&\leq \nu'(\lambda) \qquad (4.6)
\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$, where (4.6) follows from the pseudorandomness of π. ■

Lemma 4.4 follows immediately from (4.4) and from Claim 4.5. ■

This concludes the proof of Theorem 4.2. ■

Lazy users. Motivated by the recent work of Naor et al. [NRS18], we consider in addition the security of out-of-band key-exchange protocols when executed by lazy users who may not consider the out-of-band value in its entirety (e.g., users who compare with each other only a subset of its positions). Given an out-of-band key-exchange protocol $\pi = \langle P_1, \dots, P_n \rangle$ we define a collection of "lazy protocols", one per each possible subset of positions of the out-of-band authenticated value. Specifically, 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 the party who sends the out-of-band value does not send the entire value, but rather sends only its substring that corresponds to the positions in the set $\mathcal{I}$ (we refer the reader to the work of Naor et al. [NRS18] for an in-depth discussion of lazy protocols and of the motivation underlying them).

Definition 4.6. Let $n = n(\lambda), \ell = \ell(\lambda)$ and $\epsilon = \epsilon(\lambda, \cdot) : 2^{[\ell]} \to [0, 1]$ be functions of the security parameter $\lambda \in \mathbb{N}$. A group out-of-band (ℓ, ϵ)-key-exchange protocol $\pi = \langle P_1, \dots, P_n \rangle$ for a group of size $n(\lambda)$ is *secure for lazy users* if for every $\mathcal{I} = \mathcal{I}(\lambda) \subseteq [\ell]$ the lazy protocol $\pi_{\mathcal{I}}$ is a group out-of-band $(|\mathcal{I}|, \epsilon(\cdot, \mathcal{I}))$-key-exchange protocol for a group of size $n(\lambda)$.

4.2 Existing Protocols Do Not Meet Our Definition

In this section we show that the out-of-band key-exchange protocols deployed by Signal, WhatsApp, and Telegram and the protocol obtained by applying the "transcript authentication" approach using the group authentication protocol of Rotem and Segev [RS18a], do not satisfy our notion of security formalized in Definition 4.1. Some of the above approaches have advantages in certain scenarios and use cases, but the fact that they do not satisfy our security definitions while guaranteeing immediate key delivery exemplifies the difficulty in doing so, and exposes the barriers which we overcome in our construction (Section 6). In more detail, we show that:

- The protocol deployed by Telegram does not satisfy our pseudorandomness requirement (its resulting key is easily distinguishable from a random key given the out-of-band value).
- The protocols deployed by Signal and WhatsApp do not satisfy our man-in-the-middle detection requirement (there is no fixed $\epsilon = \epsilon(\lambda)$ such that any active attack is detected except with probability $\epsilon(\lambda) + \mathsf{negl}(\lambda)$). This issue is not unique to Signal and WhatsApp, and is in fact common to all non-interactive protocols. Roughly speaking, this is due to the fact that such protocols provide the attacker with the ability to sample many possible completions for a given execution, until one of them "hits" a particular out-of-band value.
- The protocol obtained by applying the "transcript authentication" approach using the group authentication protocol of Rotem and Segev does not provide immediate key delivery.

Telegram. Telegram's out-of-band key-exchange protocol (see Figure 4.1) [Tela, Telc] is a "delayed" variant of the Diffie-Hellman protocol: Alice commits to her message g^a by sending $\mathsf{Hash}(g^a)$ to Bob, who then replies with his message g^b, and then Alice reveals g^a. The shared key is then g^{ab} and the out-of-band value is $\mathsf{Hash}(g^{ab})$ (where SHA-256 is used as the hash function Hash).[10]

In this protocol, the out-of-band value is an efficiently-computable deterministic function of the shared key g^{ab}, and thus reveals a significant amount of information: Given the out-of-band value, it is trivial to distinguish between the actual shared key and an independent uniformly-distributed key with an overwhelming probability (for the actual key k it always holds that $\mathsf{Hash}(\mathsf{k})$ is equal to

[10] In fact, Telegram describes two different protocols. The protocol we describe here is used in its voice calls [Tela], and a simplified protocol where Alice does not commit to g^a is used in secret chats [Telc]. Our observation regarding the pseudorandomness requirement equally applies to both protocols.

the out-of-band value, whereas for a random key this holds only with a small probability). Thus, the protocol clearly does not satisfy our notion of pseudorandomness (even when the two hash values in the protocol, $\mathsf{Hash}(g^a)$ and $\mathsf{Hash}(g^{ab})$, are computed by two independent random oracles).

This is not merely a definitional issue, but also a practically-relevant one: The shared key g^{ab} will be later on used for deriving other cryptographic keys via a key-derivation function, and this function must produce "sufficiently strong" keys even when the out-of-band value $\mathsf{Hash}(g^{ab})$ is already known to attackers.[11] Concretely, even if the initial shared key itself is derived from g^{ab} using some key-derivation function (KDF), one still has to make sure that the resulting key is indeed pseudorandom even given $\mathsf{Hash}(g^{ab})$. Arguing this typically involved salting the KDF using a random salt, but in our scenario it is unclear how this salt is authenticated. Arguing pseudorandomness without salting the KDF requires making very strong assumption on two deterministic and unsalted functions (Hash and the KDF).

This situation is not inherent to the high-level idea used by Telegram's protocol and is actually relatively easy to fix – for example by replacing the out-of-band value with $\mathsf{Hash}(g^a \| g^b)$ (see Appendix A). However, it does show that the pseudorandomness property, though standard in the passive security regime, is quite easy to overlook in the context of out-of-band key exchange, where an attacker is exposed to the additional information sent over the out-of-band channel.

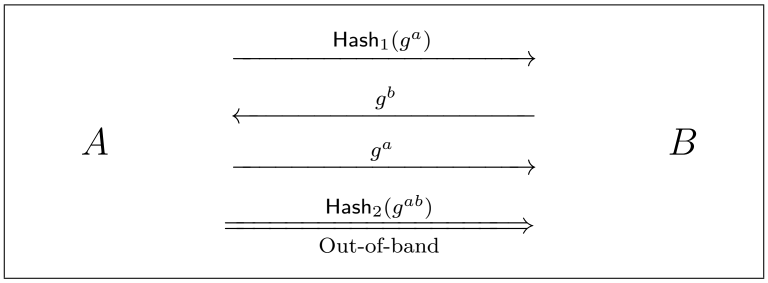

Figure 4.1: The out-of-band key-exchange protocol used by Telegram in its voice calls. The shared secret key is g^{ab} and the out-of-band value is $\mathsf{Hash}_2(g^{ab})$.

Signal and WhatsApp. Signal and WhatsApp equip each user with an "identity key" pair that is associated with a specific conversation, as well as a corresponding "fingerprint" [Wha, Mar16]. Each identity key consists of a private identity key and a public identity key. Roughly speaking, the initial secret key that is shared between two users is the result of executing a Diffie-Hellman key-exchange protocol, using the public identity keys of both users as their messages in the protocol. Each user's fingerprint is obtained by hashing her public identity key via a cryptographic hash function[12], and the out-of-band value in a conversation between two users, Alice and Bob, is then the concatenation of both fingerprints (see Figure 4.2).

Although this is a rather simplified description of the protocol used by Signal and by WhatsApp, the following simple observation regarding its misalliance with our security definition applies to the full-fledged protocol as well. Specifically, all of the in-band authentication during the full-fledged protocol is done using the same public identity keys that are used to compute the out-of-band value in our simplified description. Thus, any man-in-the-middle attack in which an adversary replaces, for example, Bob's public identity key (g^b) that is sent to Alice with a different public identity key

[11] As an extreme example, note that using the same hash function Hash both for computing the out-of-band value and for key derivation will fully expose the derived key.

[12] The result is truncated and converted into decimal digits.

($g^{\widehat{b}}$ for some $\widehat{b} \neq b$) without modifying the out-of-band value, will go undetected with the full-fledged protocol.

Recall that Definition 4.1 requires the parties to detect any man-in-the-middle attack except with probability $\epsilon(\lambda) + \mathsf{negl}(\lambda)$, where ϵ is a pre-determined function of the security parameter $\lambda \in \mathbb{N}$, for some negligible function negl which may depend on the adversary. Most importantly, ϵ must be fixed for all adversaries. However, in this protocol an adversary running in time roughly T can find, for example, a pair of public identity keys for Bob (i.e., $g^b \neq g^{\widehat{b}}$ with the same out-of-band value (i.e., with the same hash value) with probability at least $T/2^{\ell}$, where ℓ is the length of the out-of-band value, and this probability may or may not be negligible. This means that an adversary can run an execution of the key-exchange protocol with each of the users, agreeing on independent keys both of which are known to the adversary.

In addition to the above, the protocol of Signal and WhatsApp may become exposed to man-in-the-middle attacks when executed by lazy users – as already observed by Naor et al. [NRS18]. Concretely, if the users only compare the half of the out-of-band value that corresponds to, say, Alice's public identity key, then an attack that modifies the identity key that is sent to Alice as Bob's will go undetected.

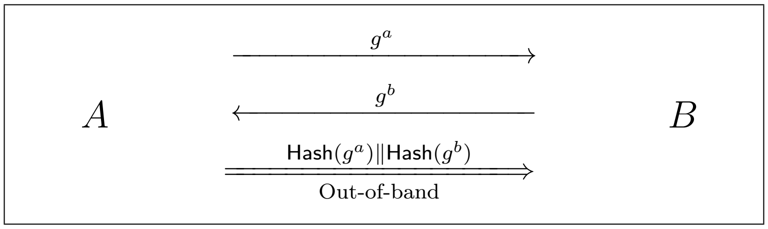

Figure 4.2: A simplified version of the out-of-band authenticated key-exchange protocol used by Signal and by WhatsApp. The shared secret key is g^{ab} and the out-of-band value is $\mathsf{Hash}(g^a)\|\mathsf{Hash}(g^b)$.

Transcript authentication [RS18a]. Rotem and Segev have defined group out-of-band message authentication and constructed a protocol meeting their definition (see definition 3.1). Following Pasini and Vaudenay [PV06b], they suggested that running any passively-secure group key-exchange protocol and then using their protocol in order to authenticate the transcript of the key exchange, yields a secure out-of-band authenticated group key-exchange protocol. This might be the case if none of the users aborts prematurely, but otherwise this approach fails to satisfy our immediate key delivery property. This lacuna is not a mere technicality, as we show that natural approaches to mend their protocol to accommodate aborting users severely degrade the level of security it provides (in particular, the resulting protocols will be very far from providing the optimal tradeoff between the length of their out-of-band value and their security).

Informally, when trans is the key-exchange transcript to be authenticated, the protocol of Rotem and Segev proceeds as follows:

1. P_1 samples $r_1 \leftarrow \{0,1\}^{\ell}$, and commits to all other users to the value $\mathsf{trans}\|r_1$.
2. For every $i \in \{2, \ldots, n\}$, P_i samples $r_i \leftarrow \{0,1\}^{\ell}$, and commits to it to all other users.
3. P_1 opens her commitment.
4. Each P_i opens her commitment.

5. P_1 out-of-band authenticates $\bigoplus_{i \in [n]} r_i$, and each P_i accepts if and only if this value is consistent with her view of the protocol (note that $r_2, \ldots, r_n$ according to the view of P_1 may be different than the actual values the other users committed to).

According to the protocol as presented by Rotem and Segev, any user who identifies a deviation from the protocol, including a premature abort, should terminate and reject. This is perfectly aligned with the standard notion of message authentication, but of course results in a failure to provide immediate key delivery when used as part of a key-exchange protocol. One possible approach to remedy this situation is have P_1 out-of-band authenticate the exclusive-or of only the r_i's of the users who completed Step 4. This, however, gravely hurts the security of the protocol, by giving the man-in-the-middle adversary the ability to choose which commitments to open to each P_i *after observing* r_i. Concretely, assume for simplicity of presentation that $\ell \geq n-2$,[13] and consider the following attack aimed at convincing P_n to accept a fraudulent transcript $\widehat{\mathsf{trans}}$ (a symmetric attack exists against P_i for every $i \in \{2, \ldots, n\}$):

1. Run an execution with all users but P_n, while honestly simulating P_n to all other users. Let σ be the out-of-band value in this execution.

2. Run an execution with P_n, simulating all other users:

   (a) Commit to $\widehat{\mathsf{trans}} \| 0^\ell$ to P_n as the commitment from P_1.

   (b) For every $i \in \{2, \ldots, n-1\}$, commit to $r_i = e_{i-1}$ as the commitment from P_i, where for every $j \in [\ell]$, e_j is an ℓ-bit string with all zeros but in the jth location.

   (c) Open the commitment of (the simulated) P_1, at which point P_n opens her commitment and reveals r_n.

   (d) For every $i \in \{2, \ldots, n-1\}$: If $\left(\bigoplus_{j \in [n] \setminus \{1\}} r_j\right)_i = \sigma_i$, open the commitment of P_i. Otherwise, abort P_i.

Observe, that the above attack guarantees that the first $n-2$ bits of the out-of-band value are *always* consistent with P_n's view. Moreover, each of the remaining bits of the out-of-band value is consistent with P_n's view with probability $1/2$ and independently of the other bits. Hence, the probability that P_n will accept the fraudulent transcript is $\min\left\{2^{n-\ell-2}, 1\right\}$, and in particular, if $n-2 \geq \ell$ the attack *always* succeeds. This is a much worse guarantee than the optimal $2^{-\ell}$ forgery probability, which we achieve using a different protocol in Section 6. An additional approach is to have P_1 authenticate not only the exclusive-or of the r_i's of the users who completed Step 4, but also the identities of the users who completed Step 4 (or who did not complete Step 4). This would prevent the above-described attack, but increases the length of the out-of-band value by nearly n bits, leading to a similar far-from-optimal and impractical tradeoff.

5 From Strong Authentication to Key Exchange

We show that any passively-secure key-exchange protocol can be transformed into an out-of-band authenticated key-exchange protocol that satisfies our strong notion of security (see Definition 4.1). Moreover, the resulting protocol offers the optimal trade-off between the length of its out-of-band value and its security within lower order terms (see Theorem 4.2). We prove the following theorem:

[13] If $n-2 > \ell$, a very similar attack still succeeds with probability 1.

Theorem 5.1. *Assuming the existence of any passively-secure key-exchange protocol, then for any functions $\ell = \ell(\lambda)$ and $n = n(\lambda)$ of the security parameter $\lambda \in \mathbb{N}$ there exists an (ℓ, ϵ)-out-of-band authenticated key-exchange protocol for a group of size $n(\lambda)$ over the same key space, where $\epsilon(\lambda) \leq 2 \cdot (n-1) \cdot (1/2 + o(1))^{\ell(\lambda)}$ for every $\lambda \in \mathbb{N}$.*

5.1 Immediate Message Delivery and Passively-Secure Immediate Key Delivery

Our construction relies on two main building blocks that satisfy a property similar to that of immediate key delivery, as defined in Section 4 via our experiment $\mathsf{FailStopExec}(\pi, A, \lambda)$ for modeling a fail-stop execution of a protocol (note that this experiment is well defined not only for key-exchange protocols, and can in fact be used to model aborting parties in a wide range of protocols).

Out-of-band message authentication with immediate message delivery. Our first building block is a strengthened form of an out-of-band group message authentication protocol, extending the notion introduced by Rotem and Segev [RS18a] for such protocols (see Definition 3.1) by asking for *immediate message delivery*: Even if a subset of the parties aborts the execution of the authentication protocol before its completion, the remaining parties should still output the sender's input message m. Relying on the notion we introduced in Section 4, this property is formalized by strengthening Definition 3.1 as follows:

Definition 5.2. Let $\ell = \ell(\lambda), \epsilon = \epsilon(\lambda)$ and $n = n(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. We say that an (ℓ, ϵ)-out-of-band group message authentication protocol $\pi = \langle S, R_1, \ldots, R_{n-1} \rangle$ for groups of size n and message space $\mathcal{M} = \{\mathcal{M}_\lambda\}_{\lambda \in \mathbb{N}}$ provides *immediate message delivery*, if for every $\lambda \in \mathbb{N}$, for every algorithm A, and for every input message $m \in \mathcal{M}_\lambda$ to S, it holds that

$$\Pr_{(\mathsf{AbortSet}, m_1, \ldots, m_n) \leftarrow \mathsf{FailStopExec}(\pi, A, \lambda)} [\forall i \in [n] \setminus \mathsf{AbortSet} : m_i = m] = 1.$$

In Section 6 we show that an out-of-band message authentication protocol with immediate message delivery can be constructed based on the existence of any a statistically-binding concurrent non-malleable commitment scheme (and thus based on any one-way function – see Section 2). Moreover, the protocol we construct offers the optimal tradeoff between the length of its out-of-band value and its security (i.e., the adversary's forgery probability).

Passively-secure key exchange with immediate key delivery. Our second building block is a passively-secure key-exchange protocol with immediate key delivery. This is naturally defined by replacing the standard correctness requirement of passively-secure key-exchange protocols (see Definition 2.1) with our immediate key delivery requirement stated in Definition 4.1. A passively-secure key exchange protocol $\langle P_1, \ldots, P_n \rangle$ with immediate key delivery can be easily obtained, for example, from any user-to-user passively-secure key-exchange protocol via the following simple transformation:

1. P_1 samples a random key $\mathsf{k} \leftarrow \mathcal{K}_\lambda$.
2. For every $i \in \{2, \ldots, n\}$, P_1 and P_i invoke the user-to-user key-exchange protocol and establish a shared key k_i.
3. For every $i \in \{2, \ldots, n\}$, P_1 uses a CPA-secure symmetric encryption scheme (whose existence is implied by that of any one-way function) to encrypt k using key k_i, and sends the resulting ciphertext to P_i.
4. Each P_i uses k_i from Step 2 to decrypt the received ciphertext, and then outputs the result of the decryption. P_1 outputs k.

It is straightforward to verify that this transformation indeed yields a passively-secure group key-exchange protocol with immediate key delivery: Even if a subset of the parties aborts the execution of the protocol before its completion, the remaining parties all output the key k chosen by P_1.

5.2 Our Construction

Our protocol relies on the following building blocks:

- A group (ℓ, ϵ)-out-of-band message authentication protocol $\langle S, R_1, \ldots, R_{n-1} \rangle$ with immediate message delivery, where $\ell = \ell(\lambda)$ and $\epsilon = \epsilon(\lambda)$ are functions of the security parameter $\lambda \in \mathbb{N}$.
- A passively-secure group key-exchange protocol $\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle$ with key space $\mathcal{K} = \{\mathcal{K}_\lambda\}_{\lambda \in \mathbb{N}}$ and immediate key delivery. We assume without loss of generality that each party in $\{P_2, \ldots, P_n\}$ sends messages to and receives messages from P_1 only.[14]
- A one-time strongly-unforgeable signature scheme $(\mathsf{KG}, \mathsf{Sign}, \mathsf{Vrfy})$.

Our protocol, which is denoted by $\langle P_1, \ldots, P_n \rangle$ and formally described below, starts by using the underlying out-of-band message authentication protocol for authenticating a verification key for the one-time signature scheme. This verification key is generated by the initiating party (denoted P_1), and its corresponding signing key is then used to sign the transcript of the out-of-band message authentication protocol, as well as the transcript of an execution of the passively-secure key-exchange protocol. The shared key resulting from executing the passively-secure key-exchange protocol is the output of each party, assuming that from this party's point of view the signature verifies correctly and the out-of-band message authentication protocol terminates successfully (i.e., no forgery was detected).

For describing the protocol, we assume for simplicity of presentation that all messages in the protocols $\langle S, R_1, \ldots, R_{n-1} \rangle$ and $\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle$ are sent to all participating users (and hence, the transcript of an honest execution of each of the protocols is the same according to the view of all users).

Out-of-Band Authenticated Group Key-Exchange Protocol $\langle \boldsymbol{P_1, \ldots, P_n} \rangle$

Joint input: The security parameter 1^λ.

1. P_1 samples $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KG}(1^\lambda)$ and sends vk to all other users.
2. $P_1, \ldots, P_n$ execute the out-of-band message authentication protocol $\langle S, R_1, \ldots, R_{n-1} \rangle$, where P_1 runs S on input $(1^\lambda, \mathsf{vk})$, and P_i runs R_{i-1} on input 1^λ for every $i \in \{2, \ldots, n\}$. Denote by $\widehat{\mathsf{vk}}_i$ the output of R_{i-1} in this execution.
3. $P_1, \ldots, P_n$ execute the passively-secure key-exchange protocol $\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle$, where P_i runs $P_{\mathsf{KE},i}$ on input 1^λ for every $i \in [n]$. Denote by k_i the output of $P_{\mathsf{KE},i}$ in this execution.
4. Denote by trans_i the transcript of Steps 2 and 3 according to the view of P_i. P_1 computes $\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, \mathsf{trans}_1)$ and sends σ to $P_2, \ldots, P_n$.
5. Denote by $\widehat{\sigma}_i$ the signature received by P_i for $i \in \{2, \ldots, n\}$. If $\widehat{\mathsf{vk}}_i \neq \bot$ and $\mathsf{Vrfy}(\widehat{\mathsf{vk}}_i, \mathsf{trans}_i, \widehat{\sigma}_i) = 1$ then P_i outputs k_i, and otherwise P_i outputs $\bot$.

The following theorem establishes the correctness and security of our protocol according to Definition 4.1:

[14] Note that this is the case in the construction from any passively-secure (user-to-user) key-exchange protocol sketched in Section 5.1. Moreover, any passively-secure group key-exchange protocol can be easily compiled into one in which all parties communicate directly solely with P_1, by re-routing all messages through P_1 (i.e., if P_i wishes to send some message to P_j, it sends it to P_1 who then forwards it to P_j).

Theorem 5.3. *Let $\ell = \ell(\lambda), \epsilon = \epsilon(\lambda)$ and $n = n(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. Then $\langle P_1, \ldots, P_n \rangle$ is an (ℓ, ϵ)-out-of-band authenticated group key-exchange protocol with key space $\mathcal{K}$ for groups of size n, assuming that:*

1. *$\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle$ is a passively-secure group key-exchange protocol with key space $\mathcal{K}$ and immediate key delivery.*
2. *$(\mathsf{KG}, \mathsf{Sign}, \mathsf{Vrfy})$ is a one-time strongly-unforgeable signature scheme.*
3. *$\langle S, R_1, \ldots, R_{n-1} \rangle$ is a group (ℓ, ϵ)-out-of-band message authentication protocol with immediate message delivery for $n-1$ receivers.*

If, in addition, $\langle S, R_1, \ldots, R_{n-1} \rangle$ is secure when executed by lazy users, then $\langle P_1, \ldots, P_n \rangle$ is secure when executed by lazy users.

Proof. The fact that the immediate key delivery property is satisfied is straightforward: A user P_i who completes the execution completes in particular the execution of $\langle S, R_1, \ldots, R_{n-1} \rangle$. Hence, P_i receives the verification key vk since $\langle S, R_1, \ldots, R_{n-1} \rangle$ satisfies the immediate message delivery. P_i also completes the execution of $\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle$ and hence receives $\mathsf{k}_i = \mathsf{k}_1 \in \mathcal{K}_\lambda$, since the passively-secure $\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle$ provides immediate key delivery. Finally, by the correctness of $(\mathsf{KG}, \mathsf{Sign}, \mathsf{Vrfy})$, the verification done by P_i in Step 5 of the protocol succeeds, and P_i indeed outputs k_1, as required.

In the rest of the proof, we focus on proving the two security properties of the protocol. This is done in two lemmata: First, Lemma 5.4 proves that the protocol provides man-in-the-middle detection; and second, Lemma 5.5 shows that it provides pseudorandom keys. We first state and prove Lemma 5.4 without addressing the possibility of lazy users, and afterwards discuss how to extend the proof to accommodate lazy users. In the notation below, we sometimes denote $\langle P_1, \ldots, P_n \rangle$ by π.

Lemma 5.4. *For any probabilistic polynomial-time algorithm M there exists a negligible function $\nu(\cdot)$ such that*

$$\Pr\left[\exists i \in \{2, \ldots, n\} : \mathsf{Active}_i \wedge \mathsf{k}_i \neq \bot\right] \leq (n(\lambda) - 1) \cdot \epsilon(\lambda) + \nu(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$.

Proof of Lemma 5.4. Let M be any probabilistic polynomial-time algorithm, and consider an execution of $\langle P_1, \ldots, P_n \rangle$ with M as the man-in-the-middle adversary. Let $i \in \{2, \ldots, n\}$. We prove that

$$\Pr\left[\mathsf{Active}_i \wedge \mathsf{k}_i \neq \bot\right] \leq \epsilon(\lambda) + \nu(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$, and the lemma follows by talking a union bound over the users $P_2, \ldots, P_n$. We consider two different cases.

Case 1: $\widehat{\mathsf{vk}}_i \neq \mathsf{vk}$. In this case, the security guarantee of the out-of-band message authentication protocol $\langle S, R_1, \ldots, R_{n-1} \rangle$ implies that there exists a negligible function $\nu_1(\cdot)$ such that

$$\Pr\left[\widehat{\mathsf{vk}}_i \neq \bot \,\middle|\, \widehat{\mathsf{vk}}_i \neq \mathsf{vk}\right] \leq \epsilon(\lambda) + \nu_1(\lambda)$$

For all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$. In addition, by the specification of the protocol $\langle P_1, \ldots, P_n \rangle$ it holds that $\mathsf{k}_i \neq \perp$ implies $\widehat{\mathsf{vk}}_i \neq \perp$. Therefore

$$\begin{aligned}
\Pr\left[\mathsf{Active}_i \wedge (\mathsf{k}_i \neq \perp) \wedge \left(\widehat{\mathsf{vk}}_i \neq \mathsf{vk}\right)\right] & \\
&\leq \Pr\left[\mathsf{Active}_i \wedge (\mathsf{k}_i \neq \perp) \middle| \widehat{\mathsf{vk}}_i \neq \mathsf{vk}\right] \\
&\leq \Pr\left[\mathsf{k}_i \neq \perp \middle| \widehat{\mathsf{vk}}_i \neq \mathsf{vk}\right] \\
&\leq \Pr\left[\mathsf{vk}_i \neq \perp \middle| \widehat{\mathsf{vk}}_i \neq \mathsf{vk}\right] \\
&\leq \epsilon(\lambda) + \nu_1(\lambda)
\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$.

Case 2: $\widehat{\mathsf{vk}_i} = \mathsf{vk}$. If $\mathsf{k}_i \neq \perp$, it implies in particular that $\mathsf{Vrfy}(\widehat{\mathsf{vk}}_i, \mathsf{trans}_i, \widehat{\sigma}) = 1$, and since $\widehat{\mathsf{vk}}_i = \mathsf{vk}$ it holds that $\mathsf{Vrfy}(\mathsf{vk}, \mathsf{trans}_i, \widehat{\sigma}) = 1$. Moreover, the fact that Active_i occurs but $\widehat{\mathsf{vk}}_i = \mathsf{vk}$ implies that $(\mathsf{trans}_1, \sigma) \neq (\mathsf{trans}_i, \widehat{\sigma}_i)$. This means that M receives a signature σ for a message trans_1, and produces either a *different* valid signature $\widehat{\sigma}$ for the same message, or a valid signature to a different message, in contradiction to the strong-unforgeability of the signature scheme.

More formally, we construct an adversary F that breaks the one-time strong-unforgeability of the signature scheme with probability at least

$$\Pr\left[\mathsf{Active}_i \wedge (\mathsf{k}_i \neq \perp) \wedge \left(\widehat{\mathsf{vk}}_i = \mathsf{vk}\right)\right].$$

On input 1^λ, the adversary F simulates to $M(1^\lambda)$ an execution of $\langle P_1, \ldots, P_n \rangle$ on joint input 1^λ with $M(1^\lambda)$ as the man-in-the-middle in the following manner:

1. F gets a verification key vk and forwards vk to M as the first message of A.[15]
2. F continues in the simulation:
   - When Step 3 of the protocol is completed according to the view of P_1, denote by trans_1 the transcript of the protocol according to her view at this point. F requests a signature for trans_1, receives a signature σ, and forwards it to M as the message of A in Step 4 of the protocol.
   - When P_i completes Step 2 of the protocol (i.e., completes the out-of-band message authentication protocol $\langle S, R_1, \ldots, R_{n-1} \rangle$), F verifies that $\widehat{\mathsf{vk}}_i = \mathsf{vk}$, and otherwise aborts.
   - If at any point P_i outputs $\mathsf{k}_i = \perp$ or if $\overline{\mathsf{Active}_i}$ occurs (this can be decided only once P_1 and P_i have completed Step 4 of the protocol), F aborts.
3. If not aborted, F outputs the pair $(\mathsf{trans}_i, \widehat{\sigma}_i)$, where $\widehat{\sigma}_i$ is the signature that M sends to P_i in Step 4 of the protocol, and trans_i is the transcript of Steps $2 - 3$ of the protocol according to the view of P_i.

By the definition of F and of the protocol $\langle P_1, \ldots, P_n \rangle$, if F outputs a pair $(\mathsf{trans}_i, \widehat{\sigma}_i)$ (and does not abort prior to that), it is necessarily the case that $\mathsf{Vrfy}(\mathsf{vk}, \mathsf{trans}_i, \widehat{\sigma}_i) = 1$, and $(\mathsf{trans}_1, \sigma) \neq (\mathsf{trans}_i, \widehat{\sigma}_i)$.

[15] We assume without loss of generality that vk is always the first message sent in an execution of $\langle P_1, \ldots, P_n \rangle$ with M.

Note that F only aborts if: (1) $\widehat{\mathsf{vk}}_i \neq \mathsf{vk}$; (2) $\mathsf{k}_i = \perp$; or (3) $\overline{\mathsf{Active}_i}$ occurs. Therefore, there exists some negligible function $\nu_2(\cdot)$ such that

$$\begin{aligned}
&\Pr\left[\mathsf{Active}_i \wedge (\mathsf{k}_i \neq \perp) \wedge \left(\widehat{\mathsf{vk}}_i = \mathsf{vk}\right)\right] && (5.1)\\
&\qquad \leq \Pr\left[\mathsf{Vrfy}(\mathsf{vk}, \mathsf{trans}_i, \widehat{\sigma}_i) = 1 \wedge (\mathsf{trans}_1, \sigma) \neq (\mathsf{trans}_i, \widehat{\sigma}_i)\right]\\
&\qquad \leq \nu_2(\lambda) && (5.2)
\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{trans}_i, \widehat{\sigma}_i)$ is the output of F if not aborted and $(\perp, \perp)$ otherwise, and (5.2) follows from the one-time strong-unforgeability of the signature scheme. Since F simulates $\langle P_1, \ldots, P_n\rangle$ to M perfectly (as long as the simulation lasts), (5.1) is indeed over $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$.

Putting both cases together and denoting $\nu(\cdot) = \nu_1(\cdot) + \nu_2(\cdot)$, it holds that

$$\begin{aligned}
\Pr\left[\mathsf{Active}_i \wedge (\mathsf{k}_i \neq \perp)\right] &\leq \Pr\left[\mathsf{Active}_i \wedge (\mathsf{k}_i \neq \perp) \wedge \left(\widehat{\mathsf{vk}}_i \neq \mathsf{vk}\right)\right]\\
&\quad + \Pr\left[\mathsf{Active}_i \wedge (\mathsf{k}_i \neq \perp) \wedge \left(\widehat{\mathsf{vk}}_i = \mathsf{vk}\right)\right]\\
&\leq \epsilon(\lambda) + \nu(\lambda)
\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$. ■

Accommodating lazy users in the proof of Lemma 5.4. In the case where $\langle S, R_1, \ldots, R_{n-1}\rangle$ is secure for lazy users, then for any subset $\mathcal{I} = \mathcal{I}(\lambda) \subseteq [\ell(\lambda)]$ it holds in particular that there exists some negligible function $\nu_1(\cdot)$ such that

$$\Pr\left[\widehat{\mathsf{vk}}_i \neq \perp \middle| \widehat{\mathsf{vk}}_i \neq \mathsf{vk}\right] \leq \epsilon(\lambda) + \nu_1(\lambda) \leq \epsilon(\lambda, \mathcal{I}) + \nu_1(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the users consider the positions in $\mathcal{I}$ of the out-of-band value. This implies that in Case 1 in the proof of Lemma 5.4, it holds

$$\Pr\left[\mathsf{Active}_i \wedge (\mathsf{k}_i \neq \perp) \wedge \left(\widehat{\mathsf{vk}}_i \neq \mathsf{vk}\right)\right] \leq \epsilon(\lambda, \mathcal{I}) + \nu_1(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$. Case 2 remains unchanged, and putting the two cases together, there exists a neglgible function $\nu(\cdot)$ such that

$$\Pr\left[\mathsf{Active}_i \wedge (\mathsf{k}_i \neq \perp)\right] \leq \epsilon(\lambda, \mathcal{I}) + \nu(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, implying that $\langle P_1, \ldots, P_n\rangle$ is secure for lazy users.

Lemma 5.5. *For any probabilistic polynomial-time algorithms M and D there exists a negligible function $\nu(\cdot)$ such that*

$$\left|\Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}_1) = 1\right] - \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}) = 1\right]\right| \leq \nu(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$ *and* $\mathsf{k} \leftarrow \mathcal{K}_\lambda$.

Proof of Lemma 5.5. The proof is by reduction to the security of the underlying key-exchange protocol $\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n}\rangle$. Let M and D be any pair of probabilistic polynomial-time algorithms attacking the pseudorandomness of $\langle P_1, \ldots, P_n\rangle$. We construct an algorithm D_{KE} that distinguishes between a key produced by a random execution of $\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle$ and a uniformly random key with distinguishing advantage

$$\left| \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}_A) = 1 \right] - \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}) = 1 \right] \right| .$$

On input $(1^\lambda, \mathsf{trans}, \mathsf{k}^*)$, D_{KE} is defined as follows:

1. Simulate to $M(1^\lambda)$ an execution of $\langle P_1, \ldots, P_n \rangle$ on joint input 1^λ with $M(1^\lambda)$ as the man-in-the-middle. At any point during the simulation, if Active occurs, sample a random bit $b \leftarrow \{0,1\}$, output b and terminate. Otherwise, the simulation is done by:
   (a) Sampling $(\mathsf{sk}, \mathsf{vk}) \leftarrow \mathsf{KG}(1^\lambda)$, and sending vk to M as the first message of A in the execution.
   (b) Simulating $\langle S, R_1, \ldots, R_{n-1} \rangle$ to M, where S runs on input $(1^\lambda, \mathsf{vk})$ and $R_1, \ldots, R_{n-1}$ run on input 1^λ.
   (c) Simulating $\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle$ on joint input 1^λ to M in accordance with trans; i.e., whenever M is expecting a message from any of $P_1, \ldots, P_n$ in Step 3 of the protocol, D_{KE} forwards it the corresponding message from trans.
   (d) Computing $\sigma \leftarrow \mathsf{Sign}(\mathsf{sk}, \mathsf{trans}_1)$ and sending σ to M, where trans_1 is the transcript of the simulation up to this point.
2. Invoke $D(1^\lambda, \mathsf{view}_M, \mathsf{k}^*)$, where view_M consists of the full transcript of the simulation and the randomness of M (which D_{KE} sampled). Output as $D(1^\lambda, \mathsf{view}_M, \mathsf{k}^*)$ does.

For $(\mathsf{trans}, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle (1^\lambda)$, $\mathsf{k} \leftarrow \mathcal{K}_\lambda$ and $\mathsf{k}^* \in \{\mathsf{k}_1, \mathsf{k}\}$, it holds that

$$\begin{aligned}
&\Pr\left[D_{\mathsf{KE}}(1^\lambda, \mathsf{trans}, \mathsf{k}^*) = 1 \right] \\
&\qquad = \Pr\left[\overline{\mathsf{Active}} \wedge D_{\mathsf{KE}}(1^\lambda, \mathsf{trans}, \mathsf{k}^*) = 1 \right] \\
&\qquad\quad + \Pr\left[\mathsf{Active} \wedge D_{\mathsf{KE}}(1^\lambda, \mathsf{trans}, \mathsf{k}^*) = 1 \right] \\
&\qquad = \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}^*) = 1 \right] + \frac{1}{2} \qquad (5.3)
\end{aligned}$$

for every $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_A, \mathsf{k}_B) \leftarrow \langle A, M, B \rangle (1^\lambda)$ and (5.3) follows from the definition of D_{KE}: Since conditioned on $\overline{\mathsf{Active}}$, D_{KE} simulates the execution of $\langle P_1, \ldots, P_n \rangle$ perfectly, and conditioned on Active, D_{KE} outputs a uniformly random bit. This implies that there exists a negligible function $\nu(\cdot)$ such that

$$\begin{aligned}
&\left| \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}_1) = 1 \right] - \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}) = 1 \right] \right| \\
&\qquad = \left| \Pr\left[D_{\mathsf{KE}}(1^\lambda, \mathsf{trans}, \mathsf{k}'_1) = 1 \right] - \Pr\left[D_{\mathsf{KE}}(1^\lambda, \mathsf{trans}, \mathsf{k}) = 1 \right] \right| \\
&\qquad \le \nu(\lambda) \qquad (5.4)
\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $(\mathsf{view}_M, \mathsf{k}_1, \ldots, \mathsf{k}_n) \leftarrow \mathsf{MitMExec}(\pi, M, \lambda)$, $(\mathsf{trans}, \mathsf{k}'_1, \ldots, \mathsf{k}'_n) \leftarrow \langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle (1^\lambda)$ and $\mathsf{k} \leftarrow \mathcal{K}_\lambda$, and (5.4) follows from the security of $\langle P_{\mathsf{KE},1}, \ldots, P_{\mathsf{KE},n} \rangle$. ■

This concludes the proof of Theorem 5.3. ■

Note that the existence of any user-to-user passively-secure key-exchange protocol implies the existence of a one-way function, which in turn implies the existence of a strongly-unforgeable signature scheme, and (as we show in Section 6) of a group (ℓ, ϵ)-out-of-band message authentication protocol with immediate message delivery and $\epsilon(\lambda) \leq 2 \cdot n(\lambda) \cdot (1/2 + o(1))^{\ell(\lambda)}$. In addition, as discussed in Section 5.1, any user-to-user passively-secure key-exchange protocol implies the existence of such a protocol with immediate key delivery. Theorem 5.1 thus immediately follows as a corollary of Theorem 5.3.

6 Out-of-Band Message Authentication with Immediate Message Delivery

In this section we construct a group out-of-band message authentication protocol with immediate message delivery based on the existence of any one-way function (instantiating the required building blocks in the random-oracle model leads to a concrete and efficient protocol). We prove the following theorem:

Theorem 6.1. *Let $\ell = \ell(\lambda)$ and $n = n(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$ and assume the existence of one-way functions. Then, there exists a group (ℓ, ϵ)-out-of-band message authentication protocol for $n(\lambda)$ receivers with immediate message delivery, where $\epsilon(\lambda) = 2 \cdot n(\lambda) \cdot (1/2 + o(1))^{\ell(\lambda)}$.*

For a string $s \in \{0,1\}^*$ and an index $i \in [|s|]$, we let s_i (or $(s)_i$) denote the ith bit of s. Let $\mathsf{BitWiseMajority}$ be the operation that on input a set of strings $s^1, \ldots, s^q$ of length ℓ, returns a string s^* whose kth coordinate is the majority among the kth coordinates of $s^1, \ldots, s^q$; i.e., if $\mathsf{BitWiseMajority}(s^1, \ldots, s^q) = s^*$, then for every $k \in [\ell]$, $(s^*)_k = \mathsf{Majority}((s^1)_k, \ldots, (s^q)_k)$. Our protocol, denoted by π, is parameterized by the number of receivers $n = n(\lambda)$ and by a function $T = T(\lambda)$ of the security parameter $\lambda \in \mathbb{N}$. The protocol uses as a building block a statistically-binding concurrent non-malleable commitment scheme Com (see Definition 2.3). As a commitment scheme may be interactive (unless one assumes the random-oracle model), when describing our protocol and referring to a commitment to a certain value, 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).

Group Out-of-Band Message Authentication Protocol $\pi = \langle S, R_1 \ldots, R_n \rangle$

Joint input: The security parameter 1^λ.

Phase 0: Initialization

1. Each party initializes a set of aborted receivers, based on her view of the protocol. We denote by $\mathcal{A}_S$ the set initialized by S, and by $\mathcal{A}_i$ the set initialized by each R_i. At the beginning of the execution $\mathcal{A}_S = \mathcal{A}_1 = \cdots = \mathcal{A}_n = \emptyset$.

Phase 1: Commitments for string selection

2. The sender S, on input m, chooses a random string $r_s \leftarrow \{0,1\}^\ell$, and executes n (possibly parallel) executions of Com to commit to the message (m, r_s) to each receiver R_i. Denote the resulting commitments according to the view of S by $c_s^1, \ldots, c_s^n$, and denote the commitment received by each R_i by $\widehat{c_s^i}$. S also appends to the first message it sends each R_i the message m. Denote by $\widehat{m}_i$ the message received by each R_i.
3. Each receiver R_i chooses random ℓ-bit strings $r_{i,1}, \ldots, r_{i,T} \leftarrow \{0,1\}^\ell$, and commits to them to the sender S using T (parallel) executions of Com. For every $i \in [n]$ denote the resulting commitments according to the view of R_i by $c_{i,1}, \ldots, c_{i,T}$, and denote the commitments received by

S by $\widehat{c_{i,1}}, \dots, \widehat{c_{i,T}}$. If some receiver R_i aborts during the commitment protocol, then S updates $\mathcal{A}_S = \mathcal{A}_S \cup \{i\}$.

4. For every $i \in [n]$, S forwards to R_i the commitments $\{(\widehat{c_{j,1}}, \dots, \widehat{c_{j,T}})\}_{j \in [n] \setminus \{i\}}$ received by her in Step 3 of the protocol, as well as $\mathcal{A}_S$. We denote by $\left\{(\widehat{c_{j,1 \to i}}, \dots, \widehat{c_{j,T \to i}})\right\}_{j \in [n] \setminus \{i\}}$ and $\widehat{\mathcal{A}_{Si}}$ the forwarded commitments and the aborted set, respectively, as received by R_i. In addition, R_i updates $\mathcal{A}_i = \widehat{\mathcal{A}_{Si}}$.

Phase 2: Gradual decommitments for string selection

5. For $t = 1, \dots, T$:

   (a) For every $i \in [n]$, R_i sends to S a decommitment $d_{i,t}$ of her commitment $c_{i,t}$ from Step 3. Let $\widehat{d_{i,t}}$ denote the decommitment received by S. For every $i \in [n]$ the sender S then checks whether $\widehat{d_{i,t}}$ is a valid decommitment to $\widehat{c_{i,t}}$. If so, let $\widehat{r_{i,t}}$ denote the committed value. If some receiver R_i either sends an invalid decommitment or aborts before sending $d_{i,t}$, then S updates $\mathcal{A}_S = \mathcal{A}_S \cup \{i\}$. For every $i \in \mathcal{A}_S$, S lets $\widehat{r_{i,t}} = 0^\ell$.

   (b) For every $i \in [n]$, S forwards R_i the decommitments $(\widehat{d_{j,t}})_{j \in [n] \setminus \{i\}}$, as well as $\mathcal{A}_S$. We let $(\widehat{d_{j,t \to i}})_{j \in [n] \setminus \{i\}}$ and $\widehat{\mathcal{A}_{Si}}$ denote the decommitments and the set received by R_i, respectively. R_i updates $\mathcal{A}_i = \mathcal{A}_i \cup \widehat{\mathcal{A}_{Si}}$. If for some $j \in [n] \setminus (\mathcal{A}_i \cup \{i\})$ it holds that $\widehat{d_{j,t \to i}}$ is not a valid decommitment to $\widehat{c_{j,t \to i}}$ received by R_i is Step 4, then R_i updates $\mathcal{A}_i = \mathcal{A}_i \cup \{j\}$. Otherwise, denote by $(\widehat{r_{j,t \to i}})_{j \in [n] \setminus \{i\}}$ the values obtained by opening the commitments. For every $j \in \mathcal{A}_i$, R_i lets $\widehat{r_{j,t \to i}} = 0^\ell$.

   (c) S computes $\sigma_t = \bigoplus_{i \in [n]} \widehat{r_{i,t}}$, and for every $i \in [n]$, R_i computes $\widehat{\sigma}_{i,t} = r_{i,t} \bigoplus_{j \in [n] \setminus \{i\}} \widehat{r_{j,t \to i}}$.

6. For every $i \in [n]$, the sender S sends receiver R_i a decommitment d_s^i to the corresponding commitment from Step 2. Denote by $\widehat{d_s^i}$ the decommitment received by R_i. For every $i \in [n]$ the receiver R_i checks if $\widehat{d_s^i}$ is a valid decommitment to $\widehat{c_s^i}$. If it is, denote the committed value by $(\widehat{m}_i, \widehat{r_s^i})$. If it is not a valid decommitment, then R_i outputs $\perp$ and terminates.

Phase 3: Out-of-band verification

7. S computes $\sigma_R = \mathsf{BitWiseMajority}(\sigma_1, \dots, \sigma_T)$ and sends $\sigma = r_S \oplus \sigma_R$ over the out-of-band channel. For every $i \in [n]$, R_i computes $\widehat{\sigma_{Ri}} = \mathsf{BitWiseMajority}(\widehat{\sigma}_{i,1}, \dots, \widehat{\sigma}_{i,T})$, and outputs $\widehat{m}_i$ if $\sigma = \widehat{r_s^i} \oplus \widehat{\sigma_{Ri}}$. Otherwise, R_i outputs $\perp$.

The following theorem captures the security of our protocol (recall that in Section 1.3 we provided a high-level overview of the proof). Setting $T(\lambda) = (n(\lambda))^2 \cdot \omega(1)$ yields Theorem 6.1 as an immediate corollary.

Theorem 6.2. *Let $\ell = \ell(\lambda), T = T(\lambda)$ and $n = n(\lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$, and let Com be a statistically-binding concurrent non-malleable commitment scheme. Then, the protocol π is a group (ℓ, ϵ)-out-of-band message authentication protocol for n receivers with immediate message delivery, where $\epsilon(\lambda) = 2 \cdot n(\lambda) \cdot \left(1/2 + O\left(n(\lambda)/\sqrt{T(\lambda)}\right)\right)^{\ell(\lambda)}$.*

Proof. The correctness of π is straightforward, so we now turn to prove its security. For an adversary M and an integer $i \in [n]$, let $\mathsf{Forge}_{M,i}$ denote the event in which the receiver R_i outputs a message $\widehat{m}_i \notin \{m, \perp\}$ in an execution of π with M as the man-in-the-middle adversary, and let $\mathsf{Forge}_M = \bigcup_{i \in [n]} \mathsf{Forge}_{M,i}$.

Lemma 6.3. *For every probabilistic polynomial-time algorithm M there exists a negligible function $\nu(\cdot)$ such that*

$$\Pr\left[\mathsf{Forge}_M\right] \leq 2 \cdot n(\lambda) \cdot \left(1/2 + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n(\lambda)}{\sqrt{T}}\right)^{\ell} + \nu(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$.

The proof relies on the following precise version of Sterling's Approximation, which follows for example from Robbins [Rob55], and on the subsequent corollary.

Fact 6.4. *For any* $a \in \mathbb{N} \setminus \{0\}$*, it holds that*

$$\sqrt{2\pi} \cdot a^{a+1/2} \cdot e^{-a} \leq a! \leq \sqrt{2\pi} \cdot a^{a+1/2} \cdot e^{-a+1}.$$

Corollary 6.5. *For any even* $a \in \mathbb{N} \setminus \{0\}$*, it holds that*

$$\binom{a}{a/2} \leq \frac{e}{\sqrt{2\pi}} \cdot \frac{1}{\sqrt{a}} \cdot 2^{a+1}.$$

Proof.

$$\begin{aligned}
\binom{a}{a/2} &= \frac{a!}{(a/2!)^2} \\
&\leq \frac{\sqrt{2\pi} \cdot a^{a+1/2} \cdot e^{-a+1}}{\left(\sqrt{2\pi} \cdot (a/2)^{a/2+1/2} \cdot e^{-a/2}\right)^2} \\
&= \frac{\sqrt{2\pi}}{2\pi} \cdot \frac{a^{a+1/2}}{a^{a+1}} \cdot \frac{e^{-a+1}}{e^{-a}} \cdot 2^{a+1} \\
&= \frac{e}{\sqrt{2\pi}} \cdot \frac{1}{\sqrt{a}} \cdot 2^{a+1} \qquad (6.1)
\end{aligned}$$

where (6.1) follows from Fact 6.4. ■

We now turn to prove Lemma 6.3.

Proof of Lemma 6.3. Let M be a probabilistic polynomial-time algorithm attacking the protocol π. Recall that for any $i \in [n]$, $\mathsf{Forge}_{M,i}$ denotes the event in which the receiver R_i outputs a message $\widehat{m}_i \notin \{m, \perp\}$ in an execution of π with M as the man-in-the-middle adversary. For every $i \in [n]$ we prove that

$$\Pr\left[\mathsf{Forge}_{M,i}\right] \leq 2 \cdot \left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^{\ell} + \nu'(\lambda)$$

for some negligible function $\nu'(\cdot)$, and the lemma follows be taking a union bound over all receivers.

We denote by $\mathsf{collision}_M$ the event in which one or more of the commitments in an execution of π with M as the man-in-the-middle adversary can be opened (information-theoretically speaking) to more than a single value. By the statistical binding of Com, there exists a negligible function $\nu_1(\cdot)$ such that $\Pr\left[\mathsf{collision}_M\right] \leq \nu_1(\lambda)$ for all sufficiently large $\lambda \in \mathbb{N}$. Thus, for the rest of the proof, we focus on bounding $\Pr\left[\mathsf{Forge}_{M,i} \middle| \overline{\mathsf{collision}_M}\right]$.

For any message v sent during the execution of the protocol with M as the man-in-the-middle adversary, we denote by $T(v)$ the time in which v was sent. We assume without loss of generality that whenever a user is due to send a message, the adversary waits until this user sends the message before deciding on its next action. Denote by Sync_M the event in which

$$T\left(d_S^i\right) > T\left(\left\{(\widehat{c_{j,1\to i}}, \ldots, \widehat{c_{j,T\to i}})\right\}_{j\in[n]\setminus\{i\}} \| \widehat{\mathcal{A}_{Si}}\right),$$

and by $\overline{\mathsf{Sync}_M}$ its complement. Meaning, Sync_M is the event in which S completes Step 6 of the protocol after R_i completes Step 4. In what follows we first bound the forgery probability in case Sync_M does not occur (see Lemma 6.6), and then bound the forgery probability in case that Sync_M does occur (see Lemma 6.9).

Lemma 6.6. *There exists a negligible function $\nu_2(\cdot)$ such that*

$$\Pr\left[\mathsf{Forge}_{M,i} \wedge \overline{\mathsf{Sync}_M} \middle| \overline{\mathsf{Collision}_M}\right] \le \left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^{\ell} + \nu_2(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$.

Proof of Lemma 6.6. In particular, when $\overline{\mathsf{Sync}_M}$ occurs, S decommits to reveal r_S before M forwards to R_i the commitments $\left\{(\widehat{c_{j,1\to i}}, \ldots, \widehat{c_{j,T\to i}})\right\}_{j\in[n]\setminus\{i\}}$ and the commitment $\widehat{c^i_S}$. Moreover, conditioned on $\overline{\mathsf{Collision}_M}$, the commitment $\widehat{c^i_S}$ uniquely defines the value $\widehat{r^i_S}$ to which it may be opened to. This means that by the time R_i sends its first decommitment, $d_{i,1}$, both $\widehat{r^i_S}$ and σ (the out-of-band value that is sent by S) are determined. So in order for M to successfully fool R_i, it must be the case that $\widehat{\sigma_{R_i}} = \sigma \oplus \widehat{r^i_S}$. We prove that this event happens with probability at most $\left(1/2 + O\left(n(\lambda)/\sqrt{T(\lambda)}\right)\right)^{\ell} + \nu_2(\lambda)$ for some negligible function $\nu_2(\cdot)$.

The proof is by reduction to the concurrent non-malleability of the underlying commitment scheme. Consider the following adversary A attacking the concurrent non-malleability of Com:

1. On input 1^λ and auxiliary input $z \in \{0,1\}^*$, A takes part in $T(\lambda)$ left interactions, and receives T commitments $c_1, \ldots, c_T$ for values $v_1, \ldots, v_T$.
2. A invokes M on input 1^λ and simulates and execution of π to M until the point where R_i is due to send her first decommitment $d_{i,1}$:
   (a) A simulates S and all of the receivers other than R_i honestly according to π.
   (b) To simulate R_i: A forwards the commitments $c_1, \ldots, c_T$ to M as the commitments $c_{i,1}, \ldots, c_{i,T}$ of R_i.[16]
   (c) During the simulation, M outputs a commitment $\widehat{c^i_S}$ and at most $(n-1)\cdot T$ commitments denoted by $\left\{(\widehat{c_{j,1\to i}}, \ldots, \widehat{c_{j,T\to i}})\right\}_{j\in[n]\setminus\{i\}}$ and the simulates S sends an out-of-band value σ.
3. A forwards $\widehat{c^i_S}$ and $\left\{(\widehat{c_{j,1\to i}}, \ldots, \widehat{c_{j,T\to i}})\right\}_{j\in[n]\setminus\{i\}}$ as the commitments on the right interactions.

Now, let $z = (r_{i,1}, \ldots, r_{i,T})$ such that $r_{i,t} \leftarrow \{0,1\}^\ell$ for every $t \in [T]$, and consider a distinguisher D getting as input $\mathsf{mim}^A_{\mathsf{Com}}(v_1, \ldots, v_T, z)$. In particular, D gets as input the event from $\{\mathsf{Sync}_M, \overline{\mathsf{Sync}_M}\}$ that has occurred in the simulation, as well as the values the values $(\widehat{m_i}, \widehat{r^i_S})$ and $\left\{(\widehat{r_{j,1\to i}}, \ldots, \widehat{r_{j,T\to i}})\right\}_{j\in[n]\setminus\{i\}}$ to which the commitments $\widehat{c^i_S}$ and $\left\{(\widehat{c_{j,1\to i}}, \ldots, \widehat{c_{j,T\to i}})\right\}_{j\in[n]\setminus\{i\}}$ can be (information-theoretically speaking) opened. Note that conditioned on $\mathsf{Forge}_{M,i} \wedge \overline{\mathsf{Collision}_M}$, it holds that $\widehat{r^i_S} \neq \perp$. D then decides on its output as follows:

1. If Sync_M has occurred or $\widehat{r^i_S} = \perp$, output 0 and terminate. Otherwise, let $v^* = \sigma \oplus \widehat{r^i_S}$.
2. For each $k = 1, \ldots, \ell$:
   (a) Initialize a counter $\mathsf{cnt} = 0$ and a set $\mathcal{A} = \emptyset$.

[16] If the commitment scheme is interactive, "forwarding" the commitments means relaying messages between the committers in the left interactions and A.

(b) For $t = 1, \ldots, T$:

- If $\left(r_{i,t} \bigoplus_{j\in[n]\setminus(\{i\}\cup\mathcal{A})} \widehat{r_{j,t\to i}}\right)_k = (v^*)_k$ (i.e., the kth bit of both strings is the same): Update $\mathsf{cnt} = \mathsf{cnt} + 1$.
- If $\left(r_{i,t} \bigoplus_{j\in[n]\setminus(\{i\}\cup\mathcal{A})} \widehat{r_{j,t\to i}}\right)_k \neq (v^*)_k$ and $|\mathcal{A}| < n - 1$: Update $\mathsf{cnt} = \mathsf{cnt} + 1$ and $\mathcal{A} = \mathcal{A} \cup \{h\}$ for the minimal $h \in [n] \setminus (\mathcal{A} \cup \{i\})$.

(c) If $\mathsf{cnt} < T/2$, output 0 and terminate. Otherwise, continue to the next iteration.

3. Output 1 and terminate.

We consider two cases. In the first case, $(v_1, \ldots, v_T) = z$, and in the second case $(v_1, \ldots, v_T) = (0^\ell, \ldots, , 0^\ell)$. The following two claims, proven after the proof of Lemma 6.6, establish the advantage of D in distinguishing between the two cases.

Claim 6.7. *There exists a negligible function $\nu_1'(\cdot)$ such that*

$$\Pr\left[D(\mathsf{mim}^{A}_{\mathsf{Com}}(z, z))\right] \geq \Pr\left[\mathsf{Forge}_{M,i} \wedge \overline{\mathsf{Sync}_M} \middle| \overline{\mathsf{Collision}_M}\right] - \nu_1'(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the probability on the left hand side is also over the choice of $z \leftarrow \{0,1\}^{\ell\times T}$.

Claim 6.8. *There exists a negligible function $\nu_2'(\cdot)$ such that*

$$\Pr\left[D(\mathsf{mim}^{A}_{\mathsf{Com}}(0^\ell, \ldots, 0^\ell, z))\right] \leq \left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^\ell + \nu_2'(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the probability on the left hand side is also over the choice of $z \leftarrow \{0,1\}^{\ell\times T}$.

Putting these two claims together, and by an averaging argument, there exist $z^* \in \{0,1\}^{\ell\times T}$ and a negligible function $\nu_3'(\cdot)$ such that

$$\begin{aligned}
\Pr&\left[\mathsf{Forge}_{M,i} \wedge \overline{\mathsf{Sync}_M} \middle| \overline{\mathsf{Collision}_M}\right] \\
&\leq \Pr\left[D(\mathsf{mim}^{M}_{\mathsf{Com}}(z^*, z^*))\right] - \Pr\left[D(\mathsf{mim}^{M}_{\mathsf{Com}}(0^\ell, \ldots, 0^\ell, z^*))\right] \\
&\qquad + \left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^\ell + \nu_1'(\lambda) + \nu_2'(\lambda) \\
&\leq \left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^\ell + \nu_1'(\lambda) + \nu_2'(\lambda) + \nu_3'(\lambda) \qquad (6.2)
\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$, where (6.2) follows from the concurrent non-malleability of Com. Denoting $\nu_2(\cdot) = \nu_1'(\cdot) + \nu_2'(\cdot) + \nu_3'(\cdot)$, this yields that

$$\Pr\left[\mathsf{Forge}_{M,i} \wedge \overline{\mathsf{Sync}_M} \middle| \overline{\mathsf{Collision}_M}\right] \leq \left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^\ell + \nu_2(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $\nu(\cdot)$ is a negligible function, concluding the proof of Lemma 6.6.

■

Lemma 6.9. *There exists a negligible function $\nu_3(\cdot)$ such that*

$$\Pr\left[\mathsf{Forge}_{M,i} \wedge \mathsf{Sync}_M \middle| \overline{\mathsf{Collision}_M}\right] \leq \left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^{\ell} + \nu_3(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$.

Proof of Lemma 6.9. The proof is by reduction to the concurrent non malleability of Com. Let $\rho_M^* = \rho_M^*(\lambda)$ and $\left\{\rho_j^* = \rho_j^*(\lambda)\right\}_{j\in[n]}$ be the random coins of M and of $\{R_j\}_{j\in[n]}$, respectively, that maximize $\Pr\left[\mathsf{Forge}_{M,i} \wedge \mathsf{Sync}_M \middle| \overline{\mathsf{Collision}_M}\right]$. In particular, M's random coins determine the message $m^* = m^*(\lambda)$ that M chooses as the input message to S.

Consider the following man-in-the-middle adversary A attacking the concurrent non-malleability of Com:

1. On input 1^λ and auxiliary input z, A takes part in a single left interaction to receive a commitment c for a value v.

2. A invokes M on input 1^λ and simulates an execution of π to M, up until the point where S is due to send d_S^i, in the following manner:

   (a) To simulate S: A waits until M outputs the input message m^* to S, and then forwards to M the commitment c along with m^* as the commitment c_S^i by S. Otherwise, A simulates S honestly.

   (b) A simulates $R_1, \dots, R_n$ honestly according to π.

   (c) When S is due to send d_S^i (i.e., once the last decommitment among $\left\{\widehat{d_{j,T}}\right\}_{j\in[n]}$ is sent to S by M), A quits the simulation.

3. A takes part in at most $n \cdot T + 1$ right interactions, in which it forwards the commitments among $\widehat{c_S^i}$, $\{c_{i,t}\}_{t\in[T]}$ and $\left\{\left(\widehat{c_{j,1\to i}}, \dots, \widehat{c_{j,T\to i}}\right)\right\}_{j\in[n]\setminus\{i\}}$, that have been completed until the simulation terminated.

Consider a distinguisher D getting as input $\mathsf{mim}_{\mathsf{Com}}^A(v, z)$. In particular, this includes the auxiliary input z, and the event from $\left\{\mathsf{Sync}_M, \overline{\mathsf{Sync}_M}\right\}$ that has occurred in the simulated execution. Conditioned on $\overline{\mathsf{Collision}_M}$ and on Sync_M, the input to D also includes:

1. The values $\widehat{r_S^i}$, $\{r_{i,t}\}_{t\in[T]}$ and $\left\{\left(\widehat{r_{j,1\to i}}, \dots, \widehat{r_{j,T\to i}}\right)\right\}_{j\in[n]\setminus\{i\}}$ to which the commitments $\widehat{c_S^i}$, $\{c_{i,t}\}_{t\in[T]}$ and $\left\{\left(\widehat{c_{j,1\to i}}, \dots, \widehat{c_{j,T\to i}}\right)\right\}_{j\in[n]\setminus\{i\}}$, respectively, may be opened to (information-theoretically speaking).

2. The value $\widehat{\sigma}$ computed by (the simulated) S.

D decides on its output as follows:

1. If $\overline{\mathsf{Sync}_M}$ has occurred in the simulated execution, output 0 and terminate. Otherwise, set $v^* = z \oplus \widehat{\sigma_R}$.

2. For each $k = 1, \dots, \ell$, D checks the following:

   (a) Initialize a counter $\mathsf{cnt} = 0$ and a set $\mathcal{A} = \emptyset$.

(b) For $t = 1, \ldots, T$:

- If $\left(r_{i,t} \bigoplus_{j \in [n] \setminus (\{i\} \cup \mathcal{A})} \widehat{r_{j,t \to i}}\right)_k = (v^*)_k$ (i.e., the kth bit of both strings is the same): Update $\mathsf{cnt} = \mathsf{cnt} + 1$.
- If $\left(r_{i,t} \bigoplus_{j \in [n] \setminus (\{i\} \cup \mathcal{A})} \widehat{r_{j,t \to i}}\right)_k \neq (v^*)_k$ and $|\mathcal{A}| < n - 1$: Update $\mathsf{cnt} = \mathsf{cnt} + 1$ and $\mathcal{A} = \mathcal{A} \cup \{h\}$ for the minimal $h \in [n] \setminus (\mathcal{A} \cup \{i\})$.

(c) If $\mathsf{cnt} < T/2$, output 0 and terminate. Otherwise, continue to the next iteration.

3. Output 1 and terminate.

Now, consider the case where the auxiliary input z is sampled uniformly $z \leftarrow \{0,1\}^\ell$. Looking ahead, we will first analyze the probability of D outputting 1 also over the choice of z, and then use an averaging argument in order to fix z to a specific value. Consider two cases:

- **Case 1:** $v = m^* \| z$. In this case A perfectly simulates π to M until the simulation terminates. This is the same case as in the one considered in Claim 6.7, with the following difference: In the case considered in Claim 6.7, the simulation terminates before Step 5 of the protocol, whereas in this case it may terminate after several iterations of Step 5 have passed, and in particular, after M has already decided weather or not to send R_i aborts by some of the receivers in these iterations. In both cases, the distinguisher D outputs 1 if the optimal strategy of M from Step 5 onward is such that results in R_i outputting $\widehat{m}_i$, given the commitments from Steps $2-4$. Meaning, in the case in hand, even if several iterations of Step 5 have passed in the simulation, D assumes that M has made the optimal choices in them. Hence, there exists a negligible function $\nu'(\cdot)$ such that

$$\Pr\left[D(\mathsf{mim}^A_{\mathsf{Com}}(m^* \| z, z)) = 1\right] \geq \Pr\left[\mathsf{Forge}_{M,i} \wedge \mathsf{Sync}_M \middle| \overline{\mathsf{Collision}_M}\right] - \nu'(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the probability on the left hand side is also over the choice of $z \leftarrow \{0,1\}^\ell$.

- **Case 2:** $v = m^* \| 0^\ell$. In this case, by the same analysis as in the proof of Claim 6.8, it holds that

$$\Pr\left[D(\mathsf{mim}^A_{\mathsf{Com}}(m^* \| 0^\ell, z)) = 1\right] \leq \left(1/2 + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^\ell$$

where the probability on the left hand side is also over the choice of $z \leftarrow \{0,1\}^\ell$.

By an averaging argument, there exists a $z^* \in \{0,1\}^\ell$ as well as a negligible function $\nu_3(\cdot)$ such that

$$\begin{aligned}
&\Pr\left[\mathsf{Forge}_{M,i} \wedge \mathsf{Sync}_M \middle| \overline{\mathsf{Collision}_M}\right] \\
&\qquad \leq \Pr\left[D(\mathsf{mim}^A_{\mathsf{Com}}(m^* \| z^*, z^*)) = 1\right] - \Pr\left[D(\mathsf{mim}^A_{\mathsf{Com}}(m^* \| 0^\ell, z^*)) = 1\right] \\
&\qquad\quad + \left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^\ell + \nu'(\lambda) \\
&\qquad \leq \left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}} \cdot \frac{n}{\sqrt{T}}\right)^\ell + \nu_3(\lambda) \qquad (6.3)
\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$, where (6.3) follows from the concurrent non-malleability of Com. ■

We now conclude the proof of Lemma 6.3 by observing that

$$\begin{aligned}\Pr\left[\mathsf{Forge}_{M,i}\right] &\leq \Pr\left[\mathsf{Forge}_{M,i}\middle|\overline{\mathsf{Collision}_M}\right] + \Pr\left[\mathsf{Collision}\right]\\ &= \Pr\left[\mathsf{Forge}_{M,i} \wedge \overline{\mathsf{Sync}_M}\middle|\overline{\mathsf{Collision}_M}\right]\\ &\quad + \Pr\left[\mathsf{Forge}_{M,i} \wedge \mathsf{Sync}_M\middle|\overline{\mathsf{Collision}_M}\right] + \nu_1(\lambda)\\ &\leq 2\cdot\left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}}\cdot\frac{n}{\sqrt{T}}\right)^{\ell} + \nu_1(\lambda) + \nu_2(\lambda) + \nu_3(\lambda)\\ &\leq 2\cdot\left(\frac{1}{2} + \frac{2e}{\sqrt{2\pi}}\cdot\frac{n}{\sqrt{T}}\right)^{\ell} + \nu(\lambda)\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$, where $\nu(\lambda) = \nu_1(\lambda) + \nu_2(\lambda) + \nu_3(\lambda)$ is a negligible function by Lemma 6.6, Lemma 6.9 and the statistical binding of Com. ■

Proof of Claim 6.7. We prove that once the commitments sent to R_i as the commitments from the other receivers, $\left\{(\widehat{c_{j,1\to i}}, \ldots, \widehat{c_{j,T\to i}})\right\}_{j\in[n]\setminus\{i\}}$, are all fixed, the best attack that M can launch has a success probability that is upper bounded by the success probability of the attack simulated by D.

In fact, we prove a stronger claim than the one stated by providing M with additional power: We assume that M can abort up to $n-1$ receivers *per each bit of the out-of-band value*; i.e., instead of running Step 5 of the protocol just once, it is executed ℓ times (with the same $c_{i,1}, \ldots, c_{i,T}$ and $\left\{(\widehat{c_{j,1\to i}}, \ldots, \widehat{c_{j,T\to i}})\right\}_{j\in[n]\setminus\{i\}}$ fixed) and in each execution M can refuse to open commitments for all other $n-1$ receivers (implying abort) regardless of the aborts in the other executions of Step 5. For every $k \in [\ell]$, at the end of the kth execution, R_i sets the kth bit of $\widehat{\sigma_{Ri}} = \mathrm{Majority}((\widehat{\sigma}_{i,1})_k, \ldots, (\widehat{\sigma}_{i,T})_k)$ (where $\widehat{\sigma}_{i,1}, \ldots, \widehat{\sigma}_{i,T}$ are the values computed in the kth execution of Step 5).

Informally, we prove that assuming the concurrent non-malleability of Com, for any $k \in [\ell]$, the optimal strategy of M in order for the event $(\widehat{\sigma_{Ri}})_k = \left(\sigma \oplus \widehat{r_S^i}\right)_k$ to occur is as follows:

1. For every $j \in [n]\setminus\{i\}$ and $t \in [T]$, let $r_{j,t\to i} = 1^{\ell}$, and commit to $r_{j,t\to i}$ to R_i as the commitment $c_{j,t\to i}$.
2. Initialize a set $\mathcal{A} = \emptyset$.
3. For every round $t \in [T]$, upon receiving the decommitment $d_{i,t}$:
   - If $\left(r_{i,t} \bigoplus_{j\in[n]\setminus(\{i\}\cup\mathcal{A})} \widehat{r_{j,t\to i}}\right)_k \neq (v^*)_k$ and there exists an index $h \in [n] \setminus (\{i\} \cup \mathcal{A})$ such that $\left(r_{i,t} \bigoplus_{j\in[n]\setminus(\{i,j\}\cup\mathcal{A})} \widehat{r_{j,t\to i}}\right)_k = (v^*)_k$, then choose the minimal such index $h \in [n]$, add h to $\mathcal{A}$ and send to R_i the decommitments $(\widehat{d_{j,t\to i}})_{j\in[n]\setminus(\mathcal{A}\cup\{i\})}$ along with $\mathcal{A}$.
   - Otherwise, send to R_i the decommitments $(\widehat{d_{j,t\to i}})_{j\in[n]\setminus(\mathcal{A}\cup\{i\})}$ along with $\mathcal{A}$.

Denote by M_{OPT} the above-described algorithm, and for every $t \in [T]$ let Hyb_t denote the algorithm that is defined by M in the first t iterations and by M_{OPT} in the remaining $T-t$ iterations. Moreover, for $k \in [\ell]$ and an algorithm B, denote by $\mathsf{Hit}_{B,i,k}$ the event in which $(\widehat{\sigma_{Ri}})_k = \left(\sigma \oplus \widehat{r_S^i}\right)_k$ in an execution with B as the man-in-the-middle adversary. We prove that for every $k \in [\ell]$ and $t \in [T]$, there exists a negligible function $\nu'(\cdot)$ such that

$$\Pr\left[\mathsf{Hit}_{\mathsf{Hyb}_t,i,k}\right] \leq \left[\mathsf{Hit}_{\mathsf{Hyb}_{t-1},i,k}\right] + \nu'(\cdot)$$

For all sufficiently large $\lambda \in \mathbb{N}$.

Let $k \in [\ell]$ and let $t \in [T]$. We make three observation:

1. Until (and not including) the tth iteration of Step 5, the distribution over the aborting receivers and the indices $t' \in [t-1]$ for which $\left(\widehat{\sigma}_{i,t'}\right)_k = \left(\sigma \oplus \widehat{r^i_S}\right)_k$ is the same both in an execution with Hyb_t and with Hyb_{t-1}.
2. In both executions, for every $q \in \{t+1, \ldots, T\}$, the probability that $(\widehat{\sigma}_{i,q})_k = \left(\sigma \oplus \widehat{r^i_S}\right)_k$ (over the randomness of M_{OPT} and of R_i) is a function only of the number of receivers aborted thus far, and not of their identities. This is by the definition of M_{OPT}.
3. In the tth iteration of the execution with Hyb_{t-1}: If less then $n-1$ receivers have aborted, then $\Pr\left[(\widehat{\sigma}_{i,t})_k = \left(\sigma \oplus \widehat{r^i_S}\right)_k\right] = 1$, and the probability that another receiver aborts in the tth iteration is $1/2$. If $n-1$ receivers have already aborted, then $\Pr\left[(\widehat{\sigma}_{i,t})_k = \left(\sigma \oplus \widehat{r^i_S}\right)_k\right] = 1/2$.

For $t \in [T], k \in [\ell]$ and an algorithm B, denote by $\mathsf{E}_{B,i,k,t}$ the event in which $\left(r_{i,t} \bigoplus_{j \in [n] \setminus (\{i\} \cup \mathcal{A})} \widehat{r_{j,t \to i}}\right)_k = \left(\sigma \oplus \widehat{r^i_S}\right)_k$ in an execution with B as the man-in-the-middle (i.e., $(\widehat{\sigma}_{i,t})_k = \left(\sigma \oplus \widehat{r^i_S}\right)_k$ without B having to abort any receivers in the tth iteration). By the concurrent non-malleability of Com, for every $t \in [T], k \in [\ell]$ and for every probabilistic polynomial-time algorithm B, there exists a negligible function $\nu'(\cdot)$ such that

$$\Pr\left[\mathsf{E}_{B,i,k,t}\right] \leq \frac{1}{2} + \nu'(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$. It follows that for every $t \in [T]$ and $k \in [\ell]$, there exists a negligible function $\nu'(\cdot)$ such that

$$\begin{aligned}\Pr\left[\mathsf{Hit}_{\mathsf{Hyb}_t,i,k}\right] - \Pr\left[\mathsf{Hit}_{\mathsf{Hyb}_{t-1},i,k}\right] &\leq \Pr\left[\mathsf{E}_{\mathsf{Hyb_t,i,k,t}}\right] - \Pr\left[\mathsf{E}_{\mathsf{Hyb_{t-1},i,k,t}}\right] \\ &\leq \frac{1}{2} + \nu'_t(\lambda) - \frac{1}{2} \\ &\leq \nu'(\lambda)\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$.

Since $\mathsf{Hyb}_0 = M_{\mathsf{OPT}}$ and $\mathsf{Hyb}_T = M$, it follows that there exists a negligible function $\nu''(\cdot)$ such that

$$\Pr\left[\mathsf{Hit}_{M,i,k}\right] \leq \Pr\left[\mathsf{Hit}_{M_{\mathsf{OPT}},i,k}\right] + \nu''(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$.

Since this holds for every $k \in [\ell]$, since $(\widehat{\sigma_{Ri}})_k = \left(\sigma \oplus \widehat{r^i_S}\right)_k$ for every $k \in [\ell]$ is a necessary condition for $\mathsf{Forge}_{M,i}$, and by the definition of the distinguisher D, it holds that there exists a negligible function $\nu'_1(\cdot)$ such that

$$\begin{aligned}\Pr\left[D(\mathsf{mim}^{\mathcal{A}}_{\mathsf{Com}}(z,z))\right] &\geq \Pr\left[\forall k \in [\ell] : \mathsf{Hit}_{M_{\mathsf{OPT}},i,k}\right] \\ &\geq \Pr\left[\forall k \in [\ell] : \mathsf{Hit}_{M,i,k}\right] - \nu'_1(\lambda) \\ &\geq \Pr\left[\mathsf{Forge}_{M,i} \wedge \overline{\mathsf{Sync}_M} \middle| \overline{\mathsf{Collision}_M}\right] - \nu'_1(\lambda)\end{aligned}$$

for all sufficiently large $\lambda \in \mathbb{N}$. ■

We now turn to prove Claim 6.8.

Proof of Claim 6.8. Throughout the proof we assume without loss of generality that T is even (if that's not the case, we can simply add an iteration to Step 5 of the protocol). Note that in the case considered in the claim, the view of M is independent of the values $r_{i,1},\dots,r_{i,T}$, so we can think of an equivalent experiment in which they are sampled *after* the commitments $\left\{(\widehat{c_{j,1\to i}},\dots,\widehat{c_{j,T\to i}})\right\}_{j\in[n]\setminus\{i\}}$ are sent by M. In this case, for every $k\in[\ell]$ and for every $t\in[T]$ it holds that $\Pr\left[\left(r_{i,t}\bigoplus_{j\in[n]\setminus\{i\}}\widehat{r_{j,t\to i}}\right)_k=(v^*)_k\right]=1/2$. Hence, the probability that D outputs 1 is equal to the probability that in k independent samples from the binomial distribution $X_1,\dots,X_k\leftarrow B(T,1/2)$ (i.e., with T samples of Bernoulli variables with parameter 1/2), it holds that $X_i<T/2+n-1$ for every $k\in[\ell]$ for which $(v^*)_i=0$ and $X_i>T/2-n+1$ for every $k\in[\ell]$ for which $(v^*)_i=1$. By the symmetry of the binomial distribution, this is exactly $\Pr\left[\forall k\in[\ell]:X_k<T/2+n-1\right]$. Since $X_1,\dots,X_k$ are independent and identically distributed, it holds that

$$\begin{aligned}
\Pr\left[\forall k\in[\ell]:X_k<\frac{T}{2}+n-1\right] &= \left(\Pr\left[X_1<\frac{T}{2}+n-1\right]\right)^{\ell}\\
&= \left(\sum_{a=0}^{T/2+n-2}\binom{T}{a}\cdot 2^{-T}\right)^{\ell}\\
&= \left(\sum_{a=0}^{T/2-1}\binom{T}{a}2^{-T}+\sum_{a=T/2}^{T/2+n-2}\binom{T}{a}\cdot 2^{-T}\right)^{\ell}\\
&= \left(1/2+\sum_{a=T/2}^{T/2+n-2}\binom{T}{a}\cdot 2^{-T}\right)^{\ell} \qquad (6.4)\\
&\le \left(1/2+(n-1)\cdot\binom{T}{T/2}\cdot 2^{-T}\right)^{\ell}\\
&\le \left(1/2+\cdot\frac{2e}{\sqrt{2\pi}}\cdot\frac{n}{\sqrt{T}}\right)^{\ell} \qquad (6.5)
\end{aligned}$$

where (6.4) follows from the symmetry of the binomial distribution and the (6.5) follows from Corollary 6.5. ■

■

References

[ABC+85] B. Awerbuch, M. Blum, B. Chor, S. Goldwasser, and S. Micali. How to implement Bracha's $O(\log n)$ byzantine agreement algorithm. Unpublished manuscript, 1985.

[ACD19] J. Alwen, S. Coretti, and Y. Dodis. The double ratchet: Security notions, proofs, and modularization for the signal protocol. In *Advances in Cryptology – EUROCRYPT '19*, pages 129–158, 2019.

[AFP05] M. Abdalla, P.-A. Fouque, and D. Pointcheval. Password-based authenticated key exchange in the three-party setting. In *Proceedings of the 8th International Conference on Practice and Theory in Public-Key Cryptography*, pages 65–84, 2005.

[AW04] H. Attiya and J. Welch. Distributed computing: fundamentals, simulations, and advanced topics. John Wiley & Sons, 2004.

[BCK98] M. Bellare, R. Canetti, and H. Krawczyk. A modular approach to the design and analysis of authentication and key exchange protocols. In *Proceedings of the 30th annual ACM Symposium on Theory of Computing*, pages 419–428, 1998.

[BCP01a] E. Bresson, O. Chevassut, and D. Pointcheval. Provably authenticated group Diffie-Hellman key exchange - the dynamic case. In *Advances in Cryptology – ASIACRYPT '01*, pages 290–309, 2001.

[BCP$^+$01b] E. Bresson, O. Chevassut, D. Pointcheval, and J. J. Quisquater. Provably authenticated group Diffie-Hellman key exchange. In *Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 255–264, 2001.

[BCP02] E. Bresson, O. Chevassut, and D. Pointcheval. Dynamic group Diffie-Hellman key exchange under standard assumptions. In *Advances in Cryptology – EUROCRYPT '02*, pages 321–336, 2002.

[Blu19] Bluetooth Special Interest Group. Bluetooth core specification v. 5.1, 2019. Available at `https://www.bluetooth.com/specifications/bluetooth-core-specification/` (accessed 11-Dec-2019).

[BMO$^+$19] R. Barnes, J. Millican, E. Omara, K. Cohn-Gordon, and R. Robert. The messaging layer security protocol, 2019. Available at `https://datatracker.ietf.org/doc/draft-ietf-mls-protocol/` (accessed 11-Dec-2019).

[BMP00] V. Boyko, P. MacKenzie, and S. Patel. Provably secure password-authenticated key exchange using Diffie-Hellman. In *Advances in Cryptology – EUROCRYPT '00*, pages 156–171, 2000.

[BPR00] M. Bellare, D. Pointcheval, and P. Rogaway. Authenticated key exchange secure against dictionary attacks. In *Advances in Cryptology – EUROCRYPT '00*, pages 139–155, 2000.

[BR93a] M. Bellare and P. Rogaway. Entity authentication and key distribution. In *Advances in Cryptology – CRYPTO '93*, pages 232–249, 1993.

[BR93b] 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.

[BR95] M. Bellare and P. Rogaway. Provably secure session key distribution: the three party case. In *Proceedings of the 27th annual ACM Symposium on Theory of Computing*, pages 57–66, 1995.

[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.

[CF01] R. Canetti and M. Fischlin. Universally composable commitments. In *Advances in Cryptology – CRPYTO '01*, pages 19–40, 2001.

[CGCG$^+$18] K. Cohn-Gordon, C. Cremers, L. Garratt, J. Millican, and K. Milner. On ends-to-ends encryption: Asynchronous group messaging with strong security guarantees. In *Proceedings of the 25th ACM conference on Computer and Communications Security*, pages 1802–1819, 2018.

[CIO98] G. D. Crescenzo, Y. Ishai, and R. Ostrovsky. Non-interactive and non-malleable commitment. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing*, pages 141–150, 1998.

[CK01] R. Canetti and H. Krawczyk. Analysis of key-exchange protocols and their use for building secure channels. In *Advances in Cryptology – EUROCRYPT '01*, pages 453–474, 2001.

[Cle86] R. Cleve. Limits on the security of coin flips when half the processors are faulty. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing*, pages 364–369, 1986.

[COS$^+$17] M. Ciampi, R. Ostrovsky, L. Siniscalchi, and I. Visconti. Four-round concurrent non-malleable commitments from one-way functions. In *Advances in Cryptology – CRYPTO '17*, pages 127–157, 2017.

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

[Duq18] A. Duque. Deep dive into Bluetooth LE security. *Medium.* Available at `https://medium.com/rtone-iot-security/deep-dive-into-bluetooth-le-security-d2301d640bfc` (accessed 11-Dec-2019), 2018.

[DV19] F. B. Durak and S. Vaudenay. Bidirectional asynchronous ratcheted key agreement without key-update primitives. In *Advances in Information and Computer Security – IWSEC '19*, pages 343–362, 2019.

[FF00] M. Fischlin and R. Fischlin. Efficient non-malleable commitment schemes. In *Advances in Cryptology – CRYPTO '00*, pages 413–431, 2000.

[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.

[FVS17] D. Fiore, M. I. G. Vasco, and C. Soriente. Partitioned group password-based authenticated key exchange. *The Computer Journal*, 60(12):1912–1922, 2017.

[GL03] R. Gennaro and Y. Lindell. A framework for password-based authenticated key exchange. In *Advances in Cryptology – EUROCRYPT '03*, pages 524–543, 2003.

[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.

[GSS$^+$06] M. T. Goodrich, M. Sirivianos, J. Solis, G. Tsudik, and E. Uzun. Loud and clear: Human-verifiable authentication based on audio. In *26th IEEE International Conference on Distributed Computing Systems*, page 10, 2006.

[Jab96] D. Jablon. Strong password-only authenticated key exchange. *ACM SIGCOMM Computer Communication Review*, 26(5):5–26, 1996.

[JMM19] D. Jost, U. Maurer, and M. Mularczyk. Efficient ratcheting: Almost-optimal guarantees for secure messaging. In *Advances in Cryptology – EUROCRYPT '19*, pages 159–188, 2019.

[JS18] J. Jaeger and I. Stepanovs. Optimal channel security against fine-grained state compromise: The safety of messaging. In *Advances in Cryptology – CRYPTO '18*, pages 33–62, 2018.

[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.

[KFR09] R. Kainda, I. Flechais, and A. Roscoe. Usability and security of out-of-band channels in secure device pairing protocols. In *Symposium on usable privacy and security (SOUPS)*, pages 11:1–11:12, 2009.

[KOY01] J. Katz, R. Ostrovsky, and M. Yung. Efficient password-authenticated key exchange using human-memorable passwords. In *Advances in Cryptology – EUROCRYPT '01*, pages 475–494, 2001.

[KY03] J. Katz and M. Yung. Scalable protocols for authenticated group key exchange. In *Advances in Cryptology – CRYPTO '03*, pages 110–125, 2003.

[Lin09] Y. Lindell. Comparison-based key exchange and the security of the numeric comparison mode in bluetooth v2.1. In *CT-RSA '09*, pages 66–83, 2009.

[LLM07] B. LaMacchia, K. Lauter, and A. Mityagin. Stronger security of authenticated key exchange. In *International Conference on Provable Security '07*, pages 1–16, 2007.

[LN06] S. Laur and K. Nyberg. Efficient mutual data authentication using manually authenticated strings. In *International Conference on Cryptology and Network Security*, pages 90–107, 2006.

[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.

[LPV08] H. Lin, R. Pass, and M. Venkitasubramaniam. Concurrent non-malleable commitments from any one-way function. In *Proceedings of the 5th Theory of Cryptography Conference*, pages 571–588, 2008.

[LSA19] S. Latvala, M. Sethi, and T. Aura. Evaluation of out-of-band channels for IoT security. *SN Computer Science*, 1(1):1–18, 2019.

[Lyn96] N. A. Lynch. Distributed algorithms. Elsevier, 1996.

[Mar16] M. Marlinspike. Safety number updates, 2016. Available at `https://signal.org/blog/safety-number-updates` (accessed 11-Dec-2019).

[MG07] R. Mayrhofer and H. Gellersen. Shake well before use: Authentication based on accelerometer data. In *International Conference on Pervasive Computing*, pages 144–161, 2007.

[MPR05] J. M. McCune, A. Perrig, and M. K. Reiter. Seeing-is-believing: using camera phones for human-verifiable authentication. In *IEEE Symposium on Security and Privacy*, pages 110–124, 2005.

[NRS18] M. Naor, L. Rotem, and G. Segev. The security of lazy users in out-of-band authentication. In *Proceedings of the 16th Theory of Cryptography Conference*, pages 575–599, 2018.

[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.

[PM16] T. Perrin and M. Marlinspike. The double ratchet algorithm, 2016. Available at `https://signal.org/docs/specifications/doubleratchet/doubleratchet.pdf` (accessed 11-Dec-2019).

[PR05] R. Pass and A. Rosen. Concurrent non-malleable commitments. In *Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science*, pages 563–572, 2005.

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

[PR18a] B. Poettering and P. Rösler. Towards bidirectional ratcheted key exchange. In *Advances in Cryptology – CRYPTO '18*, pages 3–32, 2018.

[PR18b] B. Poettering and P. RÃűsler. Asynchronous ratcheted key exchange. Cryptology ePrint Archive, Report 2018/296, 2018.

[PV06a] S. Pasini and S. Vaudenay. An optimal non-interactive message authentication protocol. In *CT-RSA '06*, pages 280–294, 2006.

[PV06b] S. Pasini and S. Vaudenay. SAS-based authenticated key agreement. In *Proceedings on the 9th International Conference on Theory and Practice of Public-Key Cryptography*, pages 395–409, 2006.

[Rob55] H. Robbins. A remark on Stirling's formula. *The American Mathematical Monthly*, 61(1):26–29, 1955.

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

[RS18a] 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.

[RS18b] L. Rotem and G. Segev. Out-of-band authentication in group messaging: Computational, statistical, optimal. Cryptology ePrint Archive, Report 2018/493, 2018.

[SEK$^+$06] N. Saxena, J.-E. Ekberg, K. Kostiainen, and N. Asokan. Secure device pairing based on a visual channel. In *IEEE Symposium on Security and Privacy*, pages 306–313, 2006.

[SH19] M. Schliep and N. Hopper. End-to-end secure mobile group messaging with conversation integrity and deniability. In *Proceedings of the 18th ACM Workshop on Privacy in the Electronic Society*, pages 55–73, 2019.

[Sho99] V. Shoup. On formal models for secure key exchange. Theory of Cryptography Library (available at `www.shoup.net/papers/skey.pdf`), 1999.

[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 11-Dec-2019).

[Telb] Telegram. End-to-end encryption. Available at `https://core.telegram.org/api/end-to-end` (accessed 11-Dec-2019).

[Telc] Telegram. Perfect forward secrecy – key visualization. Available at `https://core.telegram.org/api/end-to-end/pfs` (accessed 11-Dec-2019).

[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 11-Dec-2019).

[Wha] WhatsApp encryption overview. Available at `https://www.whatsapp.com/security/WhatsApp-Security-Whitepaper.pdf` (accessed 11-Dec-2019).

[Zim96] P. R. Zimmermann. PGPfone owner's manual, 1996. Available at `http://philzimmermann.com/docs/pgpfone10b7.pdf` (accessed 11-Dec-2019).

A A Concrete and Practical User-to-User Protocol

In Section 4.2 we showed that the user-to-user protocols deployed by Signal, WhatsApp and Telegram do not satisfy our notion of security (see Definition 4.1). Here, we show that in the random-oracle model, there is a simple and practically-relevant protocol that does satisfy it (and offers the optimal trade-off between the length of its out-of-band authenticated value and its man-in-the-middle detection probability). Recall that, in the user-to-user setting, our immediate key delivery requirement reverts to a standard correctness requirement (i.e., both parties should output the same key in an honest execution).

The protocol, which we denote by $\langle A^*, B^* \rangle$ and formally describe below, was attributed by Pasini and Vaudenay [PV06b] to Zimmermann's PGPfone key-exchange protocol. Nevertheless, the protocol described in the owner's manual of PGPfone [Zim96] resembles the one used by Telegram in that the out-of-band value is obtained by hashing *the shared key* itself – and thus this protocol does not provide the pseudorandomness property of Definition 4.1 as demonstrated in Section 4.2 (this is essentially the protocol deployed by Telegram).

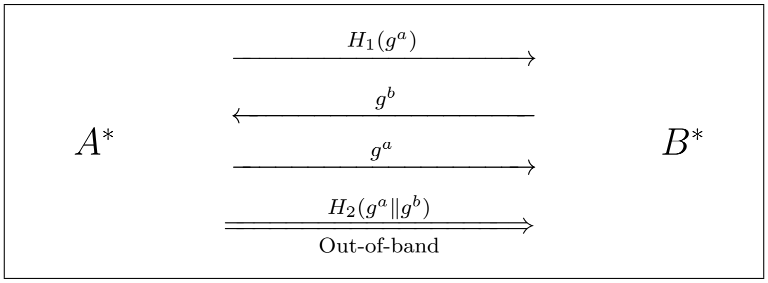

Figure A.1: A concrete out-of-band authenticated key-exchange protocol $\langle A^*, B^* \rangle$ in the random-oracle model.

Out-of-Band Authenticated Key-Exchange Protocol $\langle A^*, B^* \rangle$

Joint input:

- The security parameter 1^λ, and a description $(\mathbb{G}, g, p) \leftarrow \mathsf{GroupGen}(1^\lambda)$ of a group $\mathbb{G}$ of prime order p, where p is a λ-bit prime number and g is a generator of $\mathbb{G}$.
- The descriptions of two hash functions $H_1 : \mathbb{G} \to \{0,1\}^{n(\lambda)}$ and $H_2 : \mathbb{G} \times \mathbb{G} \to \{0,1\}^{\ell(\lambda)}$.

Protocol execution:

1. A^* samples $a \leftarrow \mathbb{Z}_p$, and sends $h_{A^*} = H_1(g^a)$ to B^* (denote by $\widehat{h_{A^*}}$ the value received by B^*).
2. B^* samples $b \leftarrow \mathbb{Z}_p$, and sends $y_{B^*} = g^b$ to B^* (denote by $\widehat{y_{B^*}}$ the value received by A^*).
3. A^* sends $y_{A^*} = g^a$ to B (denote by $\widehat{y_{A^*}}$ the value received by B^*), and out-of-band authenticates $\sigma = H_2(y_{A^*} \| \widehat{y_{B^*}})$. Then, A^* outputs $\mathsf{k}_{A^*} = (\widehat{y_{B^*}})^a$.
4. If $H_1(\widehat{y_{A^*}}) = \widehat{h_{A^*}}$ and $\sigma = H_2(\widehat{y_{A^*}} \| y_{B^*})$ then B^* outputs $\mathsf{k}_{B^*} = (\widehat{y_{A^*}})^b$ and otherwise B^* outputs $\perp$.

Theorem A.1 below establishes the correctness and security of the protocol according to Definition 4.1. Note that Definition 4.1 naturally extends to the random-oracle model by providing all algorithms with oracle access to the two functions H_1 and H_2 (which are sampled uniformly and independently from the sets of all functions with the appropriate domains and ranges).

Theorem A.1. *Let $\ell = \ell(\lambda)$ and $n = n(\lambda) = \lambda + \omega(\log \lambda)$ be functions of the security parameter $\lambda \in \mathbb{N}$. If the DDH assumption holds relative to* $\mathsf{GroupGen}$ *and the hash functions H_1 and H_2 are modeled as random oracles, then $\langle A^*, B^* \rangle$ is an out-of-band (ℓ, ϵ)-key-exchange protocol in the random-oracle model, where $\epsilon(\lambda) = 2^{-\ell(\lambda)}$.*

The correctness of the protocol $\langle A^*, B^* \rangle$ is straightforward, and thus it remains to prove the man-in-the-middle detection and the pseudorandomness requirements of Definition 4.1. We prove that the protocol satisfies these requirements in Claims A.2 and A.3.

Claim A.2. *If the DDH assumption holds relative to* $\mathsf{GroupGen}$*, then for any probabilistic polynomial-time oracle-aided algorithms M and D there exists a negligible function $\nu(\cdot)$ such that*

$$\left| \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}_{A^*}) = 1 \right] - \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}) = 1 \right] \right| \leq \nu(\lambda)$$

for all sufficiently large $\lambda \in \mathbb{N}$*, where* $(\mathsf{view}_M, \mathsf{k}_{A^*}, \mathsf{k}_{B^*}) \leftarrow \langle A^*, M, B^* \rangle (1^\lambda)$ *and* $\mathsf{k} \leftarrow \mathbb{G}$ *for* $(\mathbb{G}, g, p) \leftarrow \mathsf{GroupGen}(1^\lambda)$.

Proof of Claim A.2. Let M and D be any probabilistic polynomial-time oracle-aided algorithms attacking the pseudorandomness of $\langle A^*, B^* \rangle$. We construct a probabilistic polynomial-time algorithm D_{DDH} (in the standard model) for the decisional Diffie-Hellman problem for which for every $\lambda \in \mathbb{N}$ it holds that

$$\begin{aligned}&\left|\Pr\left[D_{\mathsf{DDH}}\left(\mathbb{G}, g, p, g^a, g^b, g^{ab}\right) = 1\right] - \Pr\left[D_{\mathsf{DDH}}\left(\mathbb{G}, g, p, g^a, g^b, g^c\right) = 1\right]\right| \\ &\quad = \left|\Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}_{A^*}) = 1\right] - \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}) = 1\right]\right|,\end{aligned}$$

where the probabilities are taken over $(\mathbb{G}, g, p) \leftarrow \mathsf{GroupGen}(1^\lambda)$, $a, b, c \leftarrow \mathbb{Z}_p$, and $(\mathsf{view}_M, \mathsf{k}_{A^*}, \mathsf{k}_{B^*}) \leftarrow \langle A^*, M, B^* \rangle (1^\lambda)$. The claim then follows from the assumed hardness of the DDH problem relative to $\mathsf{GroupGen}$.

On input $(\mathbb{G}, g, p, g_a, g_c, h)$, the algorithm D_{DDH} simulates to M and D two random functions $H_1 : \mathbb{G} \rightarrow \{0,1\}^{n(\lambda)}$ and $H_2 : \mathbb{G} \times \mathbb{G} \rightarrow \{0,1\}^{\ell(\lambda)}$, and is defined as follows:

1. Simulate to $M(1^\lambda)$ an execution of $\langle A^*, B^* \rangle$ using the values $H_1(g_a)$, g_a, and $H_2(g_a \| g_b)$ as the messages sent by A^* and using the value g_b as the message sent by B^*. At any point during the simulation, if Active occurs, sample a random bit $b \leftarrow \{0,1\}$, output b and terminate.

2. Return the output of $D(1^\lambda, \mathsf{view}_M, h)$, where view_M consists of the full transcript of the simulation and the randomness of M.

Then, for $(\mathbb{G}, g, p) \leftarrow \mathsf{GroupGen}(1^\lambda)$, $a, b \leftarrow \mathbb{Z}_p$, and for any value $h \in \mathbb{G}$ it holds that

$$\begin{aligned}\Pr&\left[D_{\mathsf{DDH}}\left(\mathbb{G}, g, p, g^a, g^b, h\right) = 1\right] \\ &= \Pr\left[\overline{\mathsf{Active}} \wedge D_{\mathsf{DDH}}\left(\mathbb{G}, g, p, g^a, g^b, h\right) = 1\right] \\ &\quad + \Pr\left[\mathsf{Active} \wedge D_{\mathsf{DDH}}\left(\mathbb{G}, g, p, g^a, g^b, h\right) = 1\right] \\ &= \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, h) = 1\right] + \frac{1}{2}\end{aligned}$$

where $(\mathsf{view}_M, \mathsf{k}_{A^*}, \mathsf{k}_{B^*}) \leftarrow \langle A^*, M, B^* \rangle (1^\lambda)$. Considering the two cases $h = g^{ab} = \mathsf{k}_{A^*}$ and $h = g^c = \mathsf{k} \leftarrow \mathbb{G}$, this implies that

$$\begin{aligned}&\left|\Pr\left[D_{\mathsf{DDH}}\left(\mathbb{G}, g, p, g^a, g^b, g^{ab}\right) = 1\right] - \Pr\left[D_{\mathsf{DDH}}\left(\mathbb{G}, g, p, g^a, g^b, g^c\right) = 1\right]\right| \\ &\quad = \left|\left(\Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}_{A^*}) = 1\right] + \frac{1}{2}\right)\right. \\ &\qquad \left. - \left(\Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}) = 1\right] + \frac{1}{2}\right)\right| \\ &\quad = \left|\Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}_{A^*}) = 1\right]\right. \\ &\qquad \left. - \Pr\left[\overline{\mathsf{Active}} \wedge D(1^\lambda, \mathsf{view}_M, \mathsf{k}) = 1\right]\right|,\end{aligned}$$

where the probabilities are taken over $(\mathbb{G}, g, p) \leftarrow \mathsf{GroupGen}(1^\lambda)$, $a, b, c \leftarrow \mathbb{Z}_p$, and $(\mathsf{view}_M, \mathsf{k}_{A^*}, \mathsf{k}_{B^*}) \leftarrow \langle A^*, M, B^* \rangle (1^\lambda)$. ■

Claim A.3. *For any oracle-aided algorithm M that makes $q_1 = q_1(\lambda)$ and $q_2 = q_2(\lambda)$ queries to H_1 and H_2, respectively, it holds that*

$$\Pr_{(\mathsf{view}_M, \mathsf{k}_{A^*}, \mathsf{k}_{B^*}) \leftarrow \langle A^*, M, B^* \rangle(1^\lambda)} [\mathsf{Active} \wedge \mathsf{k}_{B^*} \neq \perp] \leq \frac{1}{2^\ell} + \frac{q_1 + q_2 + 1}{2^\lambda} + \frac{(q_1)^2}{2^n} + \frac{2^\lambda}{2^n - q_1 - 1}$$

for all sufficiently large $\lambda \in \mathbb{N}$, where the probability is taken additionally over the choices of H_1 and H_2.

Setting $n(\lambda) = \lambda + \omega(\log \lambda)$ yields Theorem A.1.

Proof of Claim A.3. Let M be any oracle-aided algorithm attacking $\langle A^*, B^* \rangle$, and let $q_1 = q_1(\lambda)$ and $q_2 = q_2(\lambda)$ be bounds on the number of queries M makes to H_1 and to H_2, respectively, on input 1^λ. Denote by Q_1 and Q_2 the sets of queries made by M to H_1 and to H_2, respectively, throughout the execution. Let Q'_1 denote the set of queries made by M to H_1 by the time in which $\widehat{h_{A^*}}$ is sent, and let Hit denote the event in which $\widehat{h_{A^*}} \notin \{h_{A^*}\} \cup \{H_1(q) : q \in Q'_1\}$ and $\widehat{h_{A^*}} \in H_1(\mathbb{G})$, where $H_1(\mathbb{G}) = \{H_1(g^x) : x \in \mathbb{Z}_p\}$ (i.e., $\widehat{h_{A^*}}$ is in the image of H_1 when operating on $\mathbb{G}$, but it is not the result of query by M to H_1 nor is it h_{A^*}). Since conditioned on the view of M at the end of the execution, every element in $\{0,1\}^{n(\lambda)} \setminus (\{h_{A^*}\} \cup \{H_1(q) : q \in Q'_1\})$ has the same probability of being in the image of $H_1(\mathbb{G})$, it holds that

$$\begin{aligned} \Pr[\mathsf{Hit}] &\leq \frac{p - |Q'_1|}{2^n - |Q'_1| - 1} \\ &\leq \frac{2^\lambda}{2^n - q_1 - 1}. \end{aligned}$$

Let $\mathsf{Collision}$ denote the event in which there exist distinct $q, q' \in Q_1$ such that $H_1(q) = H_1(q')$ (i.e., M finds a collision in H_1). By a standard argument, it holds that

$$\Pr[\mathsf{Collision}] \leq \frac{(q_1)^2}{2^n}.$$

We now turn to consider the possible synchronizations that M might impose on an execution of $\langle A^*, B^* \rangle$. For a message v sent during the execution of the protocol, let $T(v)$ denote the time in which v was sent, and assume without loss of generality that:

1. Whenever A^* or B^* are due to send a message, M waits until this message is sent before deciding on its next action.
2. $T(\widehat{y_{A^*}}) > T(y_{B^*})$. This means that $\widehat{y_{A^*}}$ is the last message sent in the execution.

Observe that any adversary can be converted into one that abides the above assumptions while retaining the same success probability. This leaves two possible attack synchronizations to consider:

1. $T(h_{A^*}) < T(\widehat{h_{A^*}}) < T(y_{B^*}) < T(\widehat{y_{B^*}}) < T(y_{A^*}) < T(\sigma) < T(\widehat{y_{A^*}})$.
2. $T(h_{A^*}) < T(\widehat{y_{B^*}}) < T(y_{A^*}) < T(\sigma) < T(\widehat{h_{A^*}}) < T(y_{B^*}) < T(\widehat{y_{A^*}})$.

Let Sync denote the event in which the first synchronization occurs, and $\overline{\mathsf{Sync}}$ denote the event in which the second one occurs. Conditioned on $\overline{\mathsf{Sync}}$, it holds that when B^* chooses $b \leftarrow \mathbb{Z}_p$, the values y_{A^*}, $\widehat{y_{B^*}}$ and $\widehat{h_{A^*}}$ are all fixed. Conditioned on $\overline{\mathsf{Hit}} \wedge \overline{\mathsf{Collision}}$, it is also the case that when B^* chooses $b \leftarrow \mathbb{Z}_p$, there is at most one value $\widehat{y_{A^*}}$ that is consistent with $\widehat{h_{A^*}}$. Hence

$$\begin{aligned}
&\Pr_{(\mathsf{view}_M,\mathsf{k}_{A^*},\mathsf{k}_{B^*})\leftarrow\langle A^*,M,B^*\rangle(1^\lambda)}\left[\mathsf{Active} \wedge \mathsf{k}_{B^*} \neq \bot \,\middle|\, \overline{\mathsf{Sync}} \wedge \overline{\mathsf{Hit}} \wedge \overline{\mathsf{Collision}}\right] \\
&\quad\le \Pr_{(\mathsf{view}_M,\mathsf{k}_{A^*},\mathsf{k}_{B^*})\leftarrow\langle A^*,M,B^*\rangle(1^\lambda)}\left[\mathsf{k}_{B^*} \neq \bot \,\middle|\, \overline{\mathsf{Sync}} \wedge \overline{\mathsf{Hit}} \wedge \overline{\mathsf{Collision}}\right] \\
&\quad\le \Pr_{b\leftarrow\mathbb{Z}_p}\left[H_2(y_{A^*}\|\widehat{y_{B^*}}) = H_2(\widehat{y_{A^*}}\|g^b)\right] \\
&\quad\le \Pr_{b\leftarrow\mathbb{Z}_p}\left[H_2(y_{A^*}\|\widehat{y_{B^*}}) = H_2(\widehat{y_{A^*}}\|g^b) \;\middle|\; \left(\widehat{y_{A^*}}\|g^b\right) \notin Q_2 \wedge g^b \neq \widehat{y_{B^*}}\right] \\
&\qquad + \Pr_{b\leftarrow\mathbb{Z}_p}\left[\left(\widehat{y_{A^*}}\|g^b\right) \in Q_2\right] + \Pr_{b\leftarrow\mathbb{Z}_p}\left[g^b = \widehat{y_{B^*}}\right] \\
&\quad\le \frac{1}{2^\ell} + \frac{q_2+1}{p} \\
&\quad\le \frac{1}{2^\ell} + \frac{q_2+1}{2^\lambda},
\end{aligned}$$

where the probabilities are also over the choice of H_1 and H_2.

We now turn to consider the event Sync. Let Invert denote the event in which $y_{A^*} \in Q_1'$. By a standard argument

$$\Pr\left[\mathsf{Invert}\right] \le \frac{|Q_1'|}{p} \le \frac{q_1}{2^\lambda}.$$

Now, it is always the case that by the time A^* sends y_{A^*}, $\widehat{y_{B^*}}$ has already been sent. Moreover, conditioned on Sync, $\widehat{h_{A^*}}$ and y_{B^*} have also already been sent. Moreover, conditioned on $\overline{\mathsf{Invert}}$, we can think of an equivalent execution in which before A^* sends y_{A^*}, a is re-sampled from $\mathbb{Z}_p \setminus \{a' : g^{a'} \in Q_1'\}$ (and H_1 is modified accordingly). Conditioned also on $\overline{\mathsf{Collision}} \wedge \overline{\mathsf{Hit}}$, there is at most one value $\widehat{y_{A^*}}$ that M can send to B^* which is consistent with $\widehat{h_{A^*}}$. If $\widehat{h_{A^*}} \neq h_{A^*}$, then $\widehat{y_{A^*}}$ is fixed at the time of the re-sampling. Otherwise (i.e., $\widehat{h_{A^*}} = h_{A^*}$), assume without loss of generality that M queries H_1 with y_{A^*} after it is sent by A^*. Then, conditioned $\overline{\mathsf{Collision}}$, the re-sampling in this case also determines $\widehat{y_{A^*}} = y_{A^*}$. Hence, by Active, it must be that $\widehat{y_{B^*}} \neq y_{B^*}$. It follows that

$$\begin{aligned}
&\Pr_{(\mathsf{view}_M,\mathsf{k}_{A^*},\mathsf{k}_{B^*})\leftarrow\langle A^*,M,B^*\rangle(1^\lambda)}\left[\mathsf{Active} \wedge \mathsf{k}_{B^*} \neq \bot \,\middle|\, \mathsf{Sync} \wedge \overline{\mathsf{Hit}} \wedge \overline{\mathsf{Invert}} \wedge \overline{\mathsf{Collision}}\right] \\
&\quad\le \Pr\left[\widehat{h_{A^*}} \neq h_{A^*}\right] \cdot \Pr\left[H_2(g^a\|\widehat{y_{B^*}}) = H_2(\widehat{y_{A^*}}\|y_{B^*})\right] \\
&\qquad + \Pr\left[\widehat{h_{A^*}} = h_{A^*}\right] \cdot \Pr\left[H_2(g^a\|\widehat{y_{B^*}}) = H_2(g^a\|y_{B^*}) \middle| \widehat{y_{B^*}} \neq y_{B^*}\right] \\
&\quad\le \Pr\left[\widehat{h_{A^*}} \neq h_{A^*}\right] \cdot \Pr\left[H_2(g^a\|\widehat{y_{B^*}}) = H_2(\widehat{y_{A^*}}\|y_{B^*})\right] \\
&\qquad + \Pr\left[\widehat{h_{A^*}} = h_{A^*}\right] \cdot \frac{1}{2^\ell} \\
&\quad\le \Pr\left[\widehat{h_{A^*}} \neq h_{A^*}\right] \cdot \frac{1}{2^\ell} + \Pr\left[\widehat{h_{A^*}} = h_{A^*}\right] \cdot \frac{1}{2^\ell} \\
&\quad= \frac{1}{2^\ell}.
\end{aligned}$$

Putting everything together, it holds that

$$\begin{aligned}
&\Pr_{(\mathsf{view}_M,\mathsf{k}_{A^*},\mathsf{k}_{B^*})\leftarrow\langle A^*,M,B^*\rangle(1^\lambda)}\left[\mathsf{Active}\wedge\mathsf{k}_{B^*}\neq\bot\right]\\
&\quad\leq \Pr\left[\mathsf{Sync}\right]\cdot\Pr\left[\mathsf{Active}\wedge\mathsf{k}_{B^*}\neq\bot\,\middle|\,\mathsf{Sync}\wedge\overline{\mathsf{Hit}}\wedge\overline{\mathsf{Invert}}\wedge\overline{\mathsf{Collision}}\right]\\
&\qquad+\Pr\left[\overline{\mathsf{Sync}}\right]\cdot\Pr\left[\mathsf{Active}\wedge\mathsf{k}_{B^*}\neq\bot\,\middle|\,\overline{\mathsf{Sync}}\wedge\overline{\mathsf{Hit}}\wedge\overline{\mathsf{Collision}}\right]\\
&\qquad+\Pr\left[\mathsf{Hit}\right]+\Pr\left[\mathsf{Invert}\right]+\Pr\left[\mathsf{Collision}\right]\\
&\quad\leq\left[\mathsf{Sync}\right]\cdot\frac{1}{2^\ell}+\Pr\left[\overline{\mathsf{Sync}}\right]\cdot\left(\frac{1}{2^\ell}+\frac{q_2+1}{2^\lambda}\right)\\
&\qquad+\frac{2^\lambda}{2^n-q_1-1}+\frac{q_1}{2^\lambda}+\frac{(q_1)^2}{2^n}\\
&\quad\leq\frac{1}{2^\ell}+\frac{q_1+q_2+1}{2^\lambda}+\frac{(q_1)^2}{2^n}+\frac{2^\lambda}{2^n-q_1-1}.
\end{aligned}$$

concluding the proof of Claim A.3. ■