# Practical & Provably Secure Distance-Bounding

Ioana Boureanu[1], Aikaterini Mitrokotsa[2], and Serge Vaudenay[1]

[1] Ecole Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland
ioana.boureanu@epfl.ch, serge.vaudenay@epfl.ch
[2] Chalmers University of Technology
Gothennburg, Sweden
mitrokatkm@gmail.com

**Abstract.** Distance-bounding is a practical solution to be used in security-sensitive contexts, to prevent relay attacks. Its applied cryptographic role is definitely spreading fast and it is clearly far reaching, extending from contactless payments to remote car unlocking. However, security models for distance-bounding are not well-established and, as far as we know, no existing protocol is *proven* to resist all classical attacks: distance-fraud, mafia-fraud, and terrorist-fraud. We herein amend the latter, whilst maintaining the lightweight nature that makes these protocols appropriate for concrete applications. Firstly, we develop a general formalism for distance-bounding protocols and their security requirements. In fact, we also propose specifications of generalised frauds, stemming from the (attack-prone) multi-party scenarios. This entails our incorporation of newly advanced threats, e.g., distance-hijacking. Recently, Boureanu *et al.* proposed the **SKI** protocol. We herein extend it and prove its security. To attain resistance to terrorist-fraud, we put forward the use of a *leakage scheme* and of secret sharing, which we specialise and reinforce with additional requirements. In view of resistance to generalised mafia-frauds (and terrorist-frauds), we further introduce the notion of *circular-keying* for pseudorandom functions (PRFs); this notion models the employment of a PRF, with possible *linear reuse* of the key. We also identify the need of *PRF masking* to fix common mistakes in existing security proofs/claims of distance-fraud security. We then enhance our design such that we guarantee resistance to terrorist-fraud in the presence of noise. To our knowledge, all this gives rises the first practical and *provably secure* class of distance-bounding protocols, even when our protocols are run in noisy communications, which is indeed the real-life setting of deployed, time-critical cryptographic protocols.

## 1 Introduction

Cryptography sees many applications in the world of smart-cards, from the more and more sophisticated NFC bankcards to the simpler RFID access cards. But the security protocols implied (e.g., protocols for ATM systems) are vulnerable to relay attacks or more general forms of man-in-the-middle attacks. Relay attacks on bankcards have been already mounted [17]. For access control, it is not guaranteed that the card computing the responses to the reader's challenges is indeed the one requiring access. Similarly, car manufacturers use RFID protocols to unlock and even start their vehicles (see, e.g., [20]), but protocols may unfortunately be compromised by relaying [21]. The most interesting cryptographic solution to these threats seems to be based on distance-bounding [17].

Distance-bounding (DB) protocols were introduced by Brands and Chaum [9]. Their purpose is for a prover to demonstrate his proximity to a verifier and to authenticate this honest prover to this verifier.[3] In the literature covering such protocols, three main types of possible attacks have been distinguished. The first is *distance-fraud*, in which a prover tries to convince the verifier that he is closer than he really is. The second type of attack is the *mafia-fraud attack*, which involves three entities: an honest prover, an honest verifier, and an adversary. The adversary communicates with both the prover and the verifier and tries to demonstrate to the verifier that the prover is in the verifier's proximity although the prover is in reality far away from the verifier. Finally, the third type of attack is denoted as *terrorist-fraud*.[4] Here, the adversary has the same goal as in the mafia-fraud attack, but in this case the prover is dishonest and colludes with the adversary up to the non-disclosure of essential information, e.g, (parts of) secret keys, that may facilitate later impersonations of this prover.

Ad-hoc countermeasures protecting against one or several such attacks have sometimes been provided [1]. It has also been claimed [27] that DB protocols in their commonly known form cannot protect against all

[3] In this paper, we consider *authenticated* distance-bounding. Namely: protocols where both participants use a pre-established secret.

[4] The terms "mafia-fraud" and "terrorist-fraud" were introduced in 1988 by Desmedt [15]. Although confusion-prone, these are the ones still used in the literature.

three frauds at a time. Unfortunately, these frauds have become even more dangerous through recent generalisations [14,18]. Nonetheless, DB protocols will most probably soon be implemented by car manufacturers or bank payment companies in their products, as platforms for such deployments arise [32]. In these contexts, security proofs and clear, solid security models become of paramount importance. However, unitary security models and respective compelling security proofs have not yet been formulated with respect to this class of protocols. In the following, we endeavour in overcoming this shortcoming, providing a comprehensive security model for distance-bounding protocols and constructing *practical* and provably secure protocols in the model herein.

### 1.1 Distance Bounding: Informal and Semi-formal Approaches

The academic literature and industrial interest in DB is certainly growing. Thus, we can only afford to include details on what we consider essential for a good understanding of the material herein.[5]

***Tolerance to Noise.*** Since distance-bounding protocols operate under time-critical constraints and with rapid-bit exchanges, they are likely to be subject to noise, i.e., to noisy communication channels. So, these protocols often tolerate a few faulty iterations, in such a way that honest executions would succeed with high probability. Of course, noisy, rapid-bit exchanges are a reality of applied cryptographic protocols. However, many results on DB assume noiseless conditions. In this paper, noise will be taken into consideration in our security assessments.

***Protocols, Attacks and Amendments.*** Many DB protocols [26,28,33,39] consist of a data agreement phase or *initialisation phase* and a *distance-bounding* phase. The distance-bounding phase is time-critical and it normally imposes very fast computation, typically of less than a single clock cycle per round. (Light travels one meter within about 3 nanoseconds. So, every bit must be treated on the fly, upon arrival, with no delay, and there is no time to run any complicated computation.) Nevertheless, even if the time-of-flight is critical, some DB protocols are not secure against terrorist-fraud (an attacker can find ways to collude defeating DB); such examples are Bussard and Bagga [10,11], Hancke and Kuhn [24], Munilla and Peinado [29], and Kim and Avoine [27]. Reid *et al.* [33] proposed generally follow-ups of each others' schemes, addressing either a better protection against terrorist-fraud or mafia-fraud, or a better suitability to practice, or a more formal description, etc. In general, attempts [28,37,39] to construct secure distance-bounding protocols have been proven flawed [31,30]. In fact, Kim *et al.* state [27] that there is no DB protocol, which has one-bit challenges/responses per iteration in the distance-bounding phase, resisting all three attacks (i.e., distance-, mafia-, and terrorist-frauds) with a significant probability. In [8,7, Table 1], the best-known attacks against the popular distance-bounding protocols are reported. Also in [8,7], Boureanu *et al.* proposed the **SKI** protocol. So far, no attack was reported, but no security proof was provided either.

Thus, the question of *provable security* against all frauds mounted onto DB stands prominently.

Moreover, more general attacks have been recently described. In [14], Cremers *et al.* described distance-hijacking as an extension of distance-fraud, yet as an attack that is close to terrorist-fraud in the same time; the fraud involves one dishonest, far-away prover and several honest provers, without the latter colluding with the former. Impersonation (a type of man-in-the-middle) is presented in [18]. In the current work, our threat model also incorporates these latter, powerful attacks.

In [2], a targeted protocol-analysis is carried on the TDB protocol by Avoine *et al.* They especially address the protection against terrorist-fraud for the Hancke and Kuhn protocol, using secret sharing schemes. However, [2] does state the sound, (necessary and) sufficient assumptions for combating terrorist-fraud. This will be amended and taken further in this paper; we generalise the underlying idea of using secret sharing scheme [2] and introduce a taxonomy of security-enforcing conditions (some of which are linked to secret sharing).

Recently, Hancke [23] observed that terrorist-frauds could also be mounted, by simply abusing the aforementioned, noise-tolerance property required from DB. Basically, a malicious prover could help an adversary to answer most challenges and not leak to this adversary the secret key but only a noisy version of the secret key. Also, this leaked information is such that it does not give the adversary any significant advantage in later attacks onto the scheme, i.e., the coerced prover mounts a valid terrorist-fraud. As a matter of fact, all but one protocols

[5] In Appendix C, we briefly describe the relation between DB and position-based cryptography, which is in an intersecting area of interest.

allegedly resisting the classical terrorist-frauds as they were known before Hancke's observation would now collapse under terrorist-frauds executed in this new scenario of Hancke's (at least, cnf. to [8,7, Table 1]). The one protocol left standing is the **SKI** protocol.

### 1.2 Distance Bounding: Towards Provable Security

***DB Formalisations.*** In [1], Avoine *et al.* give a complete but rather informal model for distance-bounding. Herein, we will refer to this line as to the *ABKLM model*. They define distance-bounding as the combination of authentication and distance-checking. They further carry on a tentative analysis of the Munilla-Peinado protocol [29]. As we will further discuss below, [1] does not clearly state the exact assumptions needed on the underlying primitives in order to achieve the alleged security.

So far, the most promising model for distance-bounding was presented recently by Dürholz *et al.* in [18]. We refer to it as the *DFKO model*. This model does not provide a clear communication model and its notions of time or distance are only implicit. The DFKO model formalises the three classical types of frauds and an extra notion of *impersonation fraud*. The attackers are very specific, presented in terms of protocol session interleaving. Maybe due to this specificity or to their requirements which may be too strong, Fischlin and Onete [19] prove/claim the insecurity of many protocols in the DFKO model. However, insecurity against impersonation or terrorist-fraud is a hard-to-defend claim as it leads to no convincing attack. This comes from their formalisation, as we will discuss herein (see page 7). Actually, in [19], Fischlin and Onete admit that their model is probably too strong and that finding a better model for resistance to terrorist-fraud is still open.

***Security shortcomings in DB.*** Practical DB should be also attack-proof. But, from the above, one can conclude that provably secure DB is still in the making. Indeed, some of the literature on distance-bounding uses either unsupported claims of the form "if $f$ is a PRF, then this protocol is secure against...". In fact, in the recent line of Boureanu *et al.* [6], it was proven, by the technique of *PRF programming*, that if PRFs exist, then these results are incorrect. When employed with some specific PRFs, the TDB [2] protocol, an enhancement of the Kim-Avoine protocol [18], Hancke and Kuhn's [24] protocol, Avoine and Tchamkerten's [3], Reid's *et al.* [33] protocol, and the Swiss-Knife [28] protocol, they were all shown to be indeed vulnerable to distance-fraud and/or man-in-the-middle attacks. The DB security claims recently disproven by Boureanu *et al.* [6] seem to come from a mis-use of PRF techniques: replacing a PRF (in security arguments) by a random function at a place where the adversary has access to the PRF key or at a place where the PRF key is simultaneously used at other places in the protocol. In a parallel line, [28] prove that many existing distance-bounding protocols are also subject to mafia-fraud. And, in [4], it is revealed that public-key techniques do not necessarily protect against terrorist-fraud. Also therein, a family of protocols is exposed to generalised mafia-fraud attacks. Finally, Hancke [23] shows that noisy communications and tolerance to them must also be addressed in the security analysis.

### 1.3 Contribution

In the context of the shortcomings above, our main contribution is three-fold:

1. We present a formalism for distance-bounding, which includes a sound communication and adversarial model. In these latter models, we incorporate the notion of time-of-flight for distance-based communication.[6] We further formalise security against distance-fraud, man-in-the-middle (MiM) generalising mafia-frauds, and an enhanced version of terrorist-fraud that we call *collusion-fraud*. As practice dictates, our formalisations take noisy communications into account.
2. Mainly in the context of security against generalised mafia-frauds (when TF-resistance is also enforced), we introduce the concept of *circular-keying security* to extend the security of a pseudorandom function (PRF) $f$ to its possible uses in maps of the form $y \mapsto L(x) + f_x(y)$, for a secret key $x$ and a transformation $L$. We also introduce a *leakage scheme*, to resist to collusion frauds, and a *PRF masking* technique to address distance-fraud issues. These formal mechanisms come to counteract mistakes like those in proofs based on PRF-constructions, errors of the kind exposed by Boureanu *et al.* [6] and Hancke [23].

[6] Since every send/receive action in our model is subject to a maximal transmission speed, there is no distinction between a lazy phase and a time-critical one as in the DFKO model [18,19].

3. We analyse and propose variants of **SKI** [8,7], leading us to the first provably secure, practical class of distance-bounding protocols. On the way to this, we formalise the DB-driven requirements of the **SKI** protocols' components. In addition to enjoying provable security, our protocols offer competitive performance and practical security. **Especially in terms of suitability to practice, we offer the only DB protocols that resist terrorist-frauds in the presence of noise**.

## 2 Model for Distance-Bounding Protocols

We consider a multiparty setting where each participant $U$ is modelled by a polynomially bounded interactive Turing machine (ITM), has a location $loc_U$, and where communication messages from a location to another take some time, depending on the distance to travel. Some participants may be corrupted. Some are set up with a pre-shared key.

As aforementioned, we model a generic two-party communication protocol by the interactive system run by ITMs [22]; we now fix the notations.[7] Consider two honest participants $P$ and $V$, each running a predefined *algorithm* denoting its side of the interaction to take place. Along standard lines, a general communication is formalised via an *experiment*, generically denoted $exp = (P(x;r_P) \longleftrightarrow V(y;r_V))$, where $r_{\langle\cdot\rangle}$ are the random coins of the participants and $x$ is an input of $P$ and $y$ is the input of $V$. In some cases, $x = y$ denoting a long-term shared secret. The experiment above can be "enlarged" with an adversary $\mathcal{A}_0$ which interferes in the communication, up to his abilities (which will be described below). This "enlargement" can be denoted as $(P(x;r_P) \longleftrightarrow \mathcal{A}_0(r_\mathcal{A}) \longleftrightarrow V(y;r_V))$. At the end of each experiment, participant $V$ has an *output*, denoted, $Out_V$. The *view* of a participant on an experiment is the collection of all its initial inputs (including coins) and his incoming messages, i.e., the view of $\mathcal{A}_0$ subsumes his "communication" with $P$ and his "communication" with $V$. In the notation $(P(\ldots) \longleftrightarrow \mathcal{A}(\ldots) \longleftrightarrow V(\ldots))$, we may group several participants under the same symbolic name; e.g., several (colluding) malicious participants encapsulated under a single $\mathcal{A}$ denomination.

***Bound on the Distance.*** To our modelling, we add a fixed integer constant $\mathbb{B}$ denoting the *distance-bound*. It defines what it means to be "close-enough" to a verifier $V$. Hence, the output of a verifier is 1 if the responses authenticate the prover and his estimated[8] location is not further than $\mathbb{B}$ in the metric space.

***The crux of the DB model.*** The crux of proving security of DB protocols lies in Lemma 1. This says the following: if $V$ sends a challenge $c$, the answer $r$ by a close participant $\mathcal{A}$ is locally computed by $\mathcal{A}$ from its own view and incoming messages from far-away participants $\mathcal{B}$ which are independent from $c$ and all forthcoming messages. I.e., $\mathcal{A}$ cannot get online help from far-away parties in order to respond to $V$'s challenges. On the one hand, we could just introduce a full model in which such a lemma holds. We do so in Appendix A. On the other hand, we could also just state the text of the lemma and take it axiomatically.

**Lemma 1.** *Assume an experiment $\mathcal{B}(z;r_\mathcal{B}) \leftrightarrow \mathcal{A}(u;r_\mathcal{A}) \leftrightarrow V(y;r_V)$ in which the verifier $V$ plays a two-round protocol where he broadcasts a message $c$, then $V$ receives a response $r$ from $\mathcal{A}$ , and $V$ accepts if $r$ took at most time $2\mathbb{B}$ to arrive. In the experiment, $\mathcal{A}$ is the set of all participants which are within a distance up to $\mathbb{B}$ to $V$, and $\mathcal{B}$ is the set of all other participants. For each user $U$, we consider his view $View_U$ just before the time when $U$ can see the broadcast message $c$. We say that a message by $U$ is independent from $c$ if it is the result of applying $U$ on $View_U$, or a prefix of it. If $V$ accepts, it must be the case that the response $c$ was either sent by some $B \in \mathcal{B}$ as a message independent from $c$, or by some $A \in \mathcal{A}$. In any case, it can be expressed as a function $r = \mathcal{A}(View_\mathcal{A}, c, w)$ where $View_\mathcal{A}$ is the list of all $View_A$, $A \in \mathcal{A}$, and $w$ is a list of messages independent from $c$.*

*Proof (w.r.t. the model in Appendix A).* We first assume a single participant in $\mathcal{A}$. Fig. 1 illustrates the communication flow. Let $(p;r_\mathcal{A})$ be the partial view such that $r = \mathcal{A}(p;r_\mathcal{A})$. Clearly, $p$ can be written $p = (v,c,w)$ with $(v;r_\mathcal{A}) = View_\mathcal{A}$ and a list $w$ of messages from $\mathcal{B}$ participants. If $w$ includes a message $m$ not independent from $c$, there is time for $c$ to arrive to $\mathcal{B}$, to compute $m$, sent it to $\mathcal{A}$, compute $r$ and sent it to $V$. Due to the distance

[7] We use standard notations for ITMs. Namely, random coins are separated from other inputs by a semicolumn or omitted for simplicity. Inputs consist of the initial input and the variable number of incoming messages.

[8] This estimation is based on round-trip time, i.e., each response ought to be received before $V$ has $2\mathbb{B}$ standby actions.

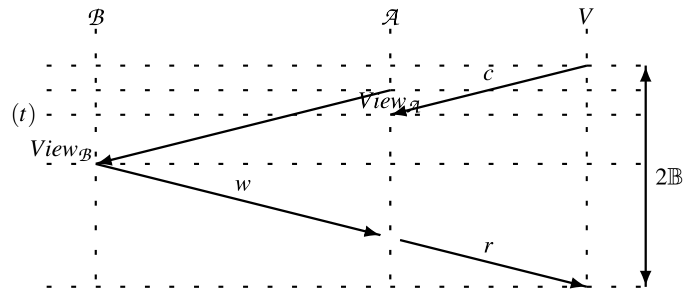


**Fig. 1.** Adversarial Communication Flow Over Time

between $\mathcal{B}$ and $V$, this is not the case. So, all messages in $w$ are independent from $c$. This means that, in due time, $\mathcal{A}$ cannot get any help from $\mathcal{B}$ to answer to $c$.

With several participants in $\mathcal{A}$, there is one $A \in \mathcal{A}$ for which $r = A(v_A, c, w_A; r_A)$ and messages in $w_A$ are either $\mathcal{A}$ messages, and can be written the same (recursively), or $\mathcal{B}$ messages which are independent from $c$. $\square$

In this paper, we will use this lemma every time when a too-long-distance has an implication on the data-flow. We believe such a clear-cut formalisation eases the proofs. I.e., in the DFKO model [18], the implicitness of timed communications requires an effective distinction between a lazy and a time-critical phase in the runs of the protocols, which may in turn hinder the construction of clear security proofs.

### 2.1 Formal Distance-Bounding

When modelling distance-bounding protocols, we consider provers, denoted by $P$ and verifiers, denoted by $V$. We let $\mathcal{A}$ denote the adversary and $P^*$ generally denote dishonest provers. We assume that provers have no output and verifiers output one bit $Out_V$ denoting acceptance, i.e. $Out_V = 1$, or rejection, i.e., $Out_V = 0$ (e.g., privileges are granted or not). We proceed with the definition of a DB protocol.

**Definition 2 (Distance-Bounding Protocols).** *A* distance-bounding (DB) protocol *is defined by a tuple* $(Gen, P, V, \mathbb{B})$, *where: 1. Gen is a randomised, key-generation algorithm such that* $(x, y)$ *is the output*[9] *of* $Gen(1^s; r_k)$, *where* $r_k$ *are the random coins of Gen and s is a security parameter; 2.* $P(x; r_P)$ *is a ppt. ITM running the algorithm of the prover with input x and random input* $r_P$*; 3.* $V(y; r_V)$ *is a ppt. ITM running the algorithm of the verifier with input y, and random input* $r_V$*; 4.* $\mathbb{B}$ *is a* distance-bound. *They must be such that the following two facts hold:*

- ***Termination****:* $(\forall s)(\forall R)(\forall r_k, r_V)(\forall loc_V)$ *if* $(\cdot, y) \leftarrow Gen(1^s; r_k)$ *and* $(R \longleftrightarrow V(y; r_V))$ *model the execution, it is the case that V halts in* $Poly(s)$ *computational steps, where R is any set of (unbounded) algorithms;*[10]
- *p-**Completeness**:* $(\forall s)$ $(\forall loc_V, loc_P$ *such that* $d(loc_V, loc_P) \leq \mathbb{B})$ *we have*

$$\Pr_{r_k, r_P, r_V}\left[Out_V = 1 : \begin{array}{l}(x, y) \leftarrow Gen(1^s; r_k)\\ P(x; r_P) \longleftrightarrow V(y; r_V)\end{array}\right] \geq p.$$

*Throughout, "*$\Pr_r[\text{event} : \text{experiment}]$*" denotes the probability that an event takes place after the experiment has happened, taken on the set of random coins r underlying the experiment. The random variable associated to the event is defined via the experiment. Hence, we are not referring here to two events conditioning one another, but just to an experiment leading to the description of a random variable.*

Our model implicitly assumes *concurrency* involving participants not sharing the secret inputs amongst them. In security definitions, these extra participants are implicitly universally quantified. When several provers using the same input $x$ appear in experiments, they will be explicitly mentioned. I.e., several instances of the same participant at different location and/or time.

The security requirements of DB protocols are described below, where $\alpha, \beta, \gamma, \gamma' \in [0, 1]$.

[9] We denote this output as $(x, y) \leftarrow Gen(1^s; r_k)$. For all protocols in this paper, there is just one common input, i.e., we assume $x = y$.
[10] In the above, only the termination of $V$ is of interest, since it is only the verifier who has a meaningful output.

**Definition 3** **($\alpha$-resistance to distance-fraud).** *$(\forall s)$ $(\forall P^*)$ $(\forall loc_V$ such that $d(loc_V, loc_{P^*}) > \mathbb{B})$ $(\forall r_k)$, we have*

$$\Pr_{r_V}\left[Out_V = 1 : \begin{array}{c}(x,y) \leftarrow Gen(1^s; r_k)\\ P^*(x) \longleftrightarrow V(y; r_V)\end{array}\right] \leq \alpha$$

*where $P^*$ is any (unbounded) dishonest prover. In a concurrent setting, we implicitly allow a polynomially bounded number of honest $P(x')$ and $V(y')$ close to $V(y)$ with independent $(x', y')$.*

In a 2-party setting, the above definition corresponds to the one of the ABKLM model [1]. When $\alpha$ is negligible, our security notion becomes equivalent to the one in the DFKO model [18].

***Relation with Distance Hijacking [14].*** Due to our concurrent setting, Def. 3 captures the notion of distance hijacking in [14], i.e., an experiment in which a dishonest far-away prover $P^*$ may use several provers to get authenticated as one, honest $P$ that is close to the verifier.

We now formalise resistance to mafia-frauds, and –in fact– to their generalisation, i.e., to MiM attacks. In MiM attacks, we consider that during a learning phase, the attacker $\mathcal{A}$ interacts, in parallel, with $m \geq 0$ provers and $z \geq 0$ verifiers and then —in the attack phase— $\mathcal{A}$ tries to win in an experiment in front of a verifier which is far-away from $\ell - m \geq 0$ provers.

**Definition 4** **($\beta$-resistance to MiM).** *$(\forall s)(\forall m, \ell, z)$ polynomially bounded, $(\forall \mathcal{A}_1, \mathcal{A}_2)$ polynomially bounded, for all locations such that $d(loc_{P_j}, loc_V) > \mathbb{B}$, where $j \in \{m+1, \ldots, \ell\}$, we have*

$$\Pr\left[Out_V = 1 : \begin{array}{l}(x,y) \leftarrow Gen(1^s)\\ P_1(x), \ldots, P_m(x) \longleftrightarrow \mathcal{A}_1 \longleftrightarrow V_1(y), \ldots, V_z(y)\\ P_{m+1}(x), \ldots, P_\ell(x) \longleftrightarrow \mathcal{A}_2(View_{\mathcal{A}_1}) \longleftrightarrow V(y)\end{array}\right] \leq \beta$$

*over all random coins, where $View_{\mathcal{A}_1}$ is the final view of $\mathcal{A}_1$. In a concurrent setting, we implicitly allow a polynomially bounded number of $P(x')$, $P^*(x')$, and $V(y')$ with independent $(x', y')$, anywhere.*

Definition 4 separates a *learning phase* (with the adversarial behaviour $\mathcal{A}_1$) from an *attack phase* (with the adversarial behaviour $\mathcal{A}_2$). Def. 4 model a practical setting where an attacker would have cloned several tags and would make them interact with several readers with which they are registered. From such a multi-party communication, the attacker can get potentially more benefits, in a shorter period of time.

Of course, the attacker can set up this learning phase as he pleases, to increase his gains. So, we can even imagine that he places prover-tags close to verifier-readers, even if being an active adversary between two neighbouring $P$ and $V$ is technically more challenging than interfering between two far-away parties. E.g., in this scenario, the adversary could interfere with the initial frequency synchronisation phase so that the $P \leftrightarrow \mathcal{A}$ and $\mathcal{A} \leftrightarrow V$ channels would become different (e.g., using different frequency bands) and $P$ and $V$ would not even be aware of the existence of the other channel.

In any case, note that the learning phase is not obligatory in our setting ($m$ and $z$ can be 0). Indeed, we further consider mafia-frauds as a specialisation of the above, where no learning phase is present. But, if and when a non-trivial learning phase is present, it renders a stronger threat model and proven resistance to such attacks entails better security.

***Relation with Mafia-fraud.*** The classical notion of mafia-fraud (the one from the ABKLM model [1]) corresponds to $m = z = 0$ and $\ell = 1$. The classical notion of impersonation for identification schemes corresponds to $\ell = m$ (i.e., there is no prover in the attack phase). The DFKO model [18] of mafia-fraud already includes the above general extension since concurrent settings are implicit in the DFKO model.

We now describe a special type of MiM attackers. We say that a (MiM) attacker is *non-narrow* [40] if he can learn the bit the verifier outputs. A way in which this can be trivially formalised is by adding a return channel to the communication, here denoting that the verifier $V$ sends $Out_V$ as a final message, just before $V$ halts. In real life this is the case, e.g., there is a LED on a door turning green denoting "access-granted" and turning red otherwise. Moreover, in the generalised MF presented in [4], it is this sort of return channel that facilitates the attacks (i.e., logically, intruders learn more information by looking also at whether the run was successful or not.). To avoid defining a new class of attacks (as done in the literature [40]), we define this as a property of the protocol.

**Definition 5 (Non-narrow MiM).** *A distance-bounding protocol is called* non-narrow *if it terminates by $V$ sending $Out_V$ to $P$ as his final message.*

We now formalise the resistance to an attack-pattern that extends the terrorist-fraud.

**Definition 6 ($(\gamma,\gamma')$-resistance to collusion-fraud).** $(\forall s)(\forall P^*)$ $(\forall loc_{V_0}$ *such that* $d(loc_{V_0}, loc_{P^*}) > \mathbb{B})$ $(\forall \mathcal{A}^{\mathsf{CF}}$ *ppt.) such that*

$$\Pr\left[Out_{V_0} = 1 : \begin{array}{l}(x,y) \leftarrow Gen(1^s)\\ P^*(x) \longleftrightarrow \mathcal{A}^{\mathsf{CF}} \longleftrightarrow V_0(y)\end{array}\right] \geq \gamma$$

*over all random coins, there exists a (kind of)*[11] *MiM attack $m, \ell, z, \mathcal{A}_1, \mathcal{A}_2, P_i, P_j, V_{i'}$ using $P$ and $P^*$ in the learning phase, such that*

$$\Pr\left[Out_V = 1 : \begin{array}{l}(x,y) \leftarrow Gen(1^s)\\ P_1^{(*)}(x),\ldots,P_m^{(*)}(x) \longleftrightarrow \mathcal{A}_1 \longleftrightarrow V_1(y),\ldots,V_z(y)\\ P_{m+1}(x),\ldots,P_\ell(x) \longleftrightarrow \mathcal{A}_2(View_{\mathcal{A}_1}) \longleftrightarrow V(y)\end{array}\right] \geq \gamma'$$

*where $P^*$ is any (unbounded) dishonest prover and $P^{(*)}$ runs either $P$ or $P^*$. Following the MiM requirements, $d(loc_{P_j}, loc_V) > \mathbb{B}$, for all $j \in \{m+1, \ell\}$. In a concurrent setting, we implicitly allow a polynomially bounded number of $P(x')$, $P^*(x')$, and $V(y')$ with independent $(x', y')$, but no honest participant close to $V_0$.*

Definition 6 expresses the following. If a prover $P^*$, situated far-away from $V_0$, can help an adversary $\mathcal{A}^{\mathsf{CF}}$ that is closer to $V_0$ pass a distance-bounding protocol, then a malicious $(\mathcal{A}_1, \mathcal{A}_2)$ could run a rather successful MiM attack[12] playing with possibly multiple instances of $P^*(x)$ in the learning phase. In other words, a dishonest prover $P^*$ cannot successfully collude with $\mathcal{A}^{\mathsf{CF}}$ without leaking some private information.

One problem with collusion frauds is that they are non-falsifiable. But this is inherent to terrorist frauds.

***Relation with Terrorist-fraud.*** The notion of terrorist-fraud (in the ABKLM or DFKO models) corresponds here to the specialised case where $m = z = \ell = 1$ and $\mathcal{A}_1$ just runs $\mathcal{A}^{\mathsf{CF}}$ in the learning phase. I.e., $\mathcal{A}^{\mathsf{CF}}$ gets information to directly impersonate the prover. (So, collusion-frauds are more general.)

In the DFKO model [18], the formalisation of terrorist-fraud further considers $p_A = \Pr[Out_{V_0} = 1]$, and $p_S = \Pr[Out_V = 1 | Out_{V_0} = 1]$. Following some results from [19], a protocol resists to terrorist-fraud if for every $\mathcal{A}^{\mathsf{CF}}$ there is a $\mathcal{A}_2$ such that $p_A \leq p_S$. However, we think that illustrating some $\mathcal{A}^{\mathsf{CF}}$ such that $p_A$ is negligible but for no $\mathcal{A}_2$ we would have $p_A \leq p_S$ [19] is not argument enough for insecurity. It rather shows that the definition from [18] is too strong. In our approach, we decided to characterise resistance herein through a pair of probabilities $(\gamma, \gamma')$.

## 3 Practical and Secure Distance-Bounding Protocols

### 3.1 SKI: DESCRIPTION AND COMPLETENESS

At a high level, the protocol schema **SKI** is presented in Fig. 2. We use the parameters $(s, q, n, k, t, t')$, where $s$ is the security parameter. The **SKI** protocols are built using a PRF (pseudorandom function), denoted $(f_x)_{x \in GF(q)^s}$, with $q$ being a small power of prime. In the concrete examples in the main body of the paper, we employ $q = 2$, i.e., $x$, $a$ are simply bitstrings as it is most practical. In the DB phase, $n$ rounds are used, with $n \in \Omega(s)$. Then, **SKI** uses the value $f_x(N_P, N_V, L) \in GF(q)^{t'n}$, with nonces $N_P, N_V \in \{0,1\}^k$ and a mask $M \in GF(q)^{t'n}$, where $k \in \Omega(s)$. In our main proposal, we use $t' = 2$, i.e., we keep the lightweight character. The element $a = (a_1, \ldots, a_n)$ is

[11] Def. 4 defines MiM attacks as using a honest $P(x)$. Here, we deviate a bit by introducing $P^*(x)$ as well.

[12] In practice, $\mathcal{A}^{\mathsf{MiM}}$ and $\mathcal{A}^{\mathsf{CF}}$ represent the same adversarial party; we simply differentiate to show that different algorithms/attack-strategies may be involved.

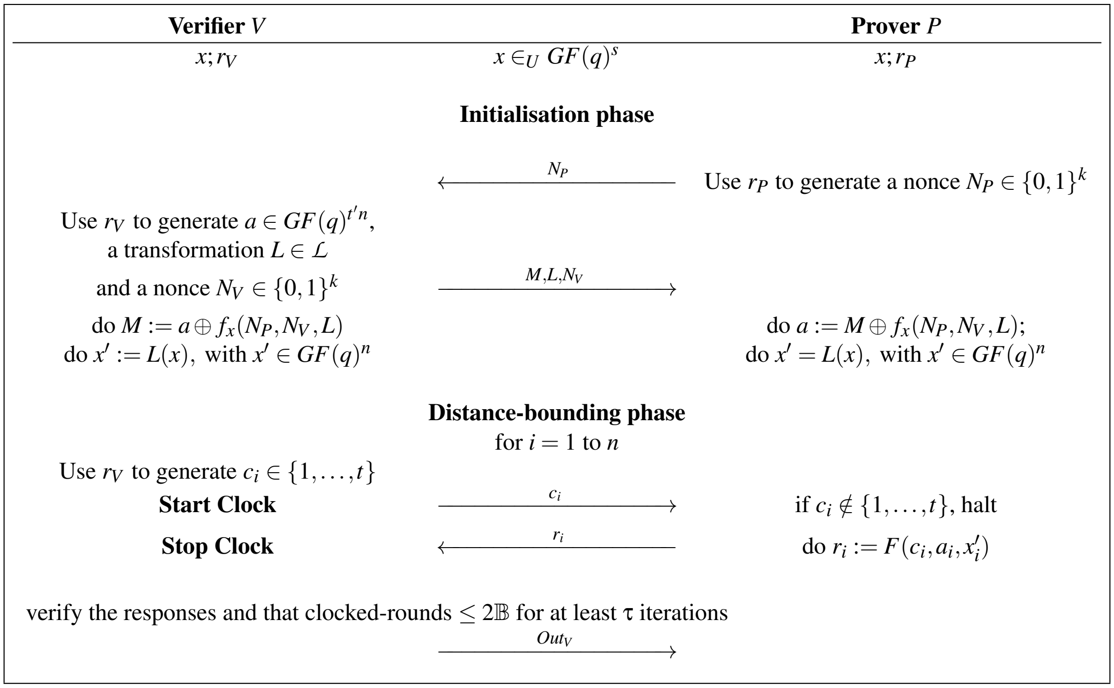


**Fig. 2.** The **SKI** schema of Distance-Bounding Protocols

established by $V$ in the initialisation phase, and it is sent encrypted as $M := a \oplus f_x(N_P, N_V, L)$, with $M \in GF(q)^{t'n}$. Similarly, $V$ selects a random linear transformation $L$ from a set[13] $\mathcal{L}$ which is specified by the **SKI** protocol instance and the parties compute $x' = L(x)$. Further, $c = (c_1, \ldots, c_n)$ is the challenge-vector with $c_i \in \{1, \ldots, t\}$, $r_i := F(c_i, a_i, x'_i)$ is the $i$-th response to the $i$-th challenge $c_i$, with $i \in \{1, \ldots, n\}$, $r_i \in GF(q)$ and $F$ as specified below.[14] In our concrete proposals, we use $t = 3$, or $t = 2$ for the lighter version. The protocol ends with a message $Out_V$ denoting the output of the verifier (i.e., the success/failure of the protocol), to capture the notion of MiM attackers on a non-narrow protocol.

***SKI Instances.*** We first depict $\mathbf{SKI_{pro}}$ through Fig. 3.

In fact, in Boureanu *et al.* [8,7], several variants of **SKI** were proposed. We further concentrate on two of them:

- $\mathbf{SKI_{pro}}$ with $q = 2$, $t' = 2$, $t = 3$, with the response-function

$$F(1, a_i, x'_i) = (a_i)_1 \qquad F(2, a_i, x'_i) = (a_i)_2 \qquad F(3, a_i, x'_i) = x'_i + (a_i)_1 + (a_i)_2,$$

  where $(a_i)_j$ denotes the $j$th bit of $a_i$, with the transforms $L_\mu$ defined each from a vector $\mu \in GF(q)^s$ by

$$L_\mu(x) = (\mu \cdot x, \ldots, \mu \cdot x)$$

  i.e., $n$ repetitions of the same bit $\mu \cdot x$, the dot product of $\mu$ and $x$.
- $\mathbf{SKI_{lite}}$ with $q = 2$, $t' = 2$, $t = 2$, with the response-function

$$F(1, a_i, x'_i) = (a_i)_1 \qquad F(2, a_i, x'_i) = (a_i)_2,$$

  with the transform-set $\mathcal{L} = \{\mathbf{0}\}$.

Namely, note that $\mathbf{SKI_{lite}}$ never uses the $c_i = 3$ challenge, i.e., it never uses the part $x'$ having to do directly with the secret key $x$ in the DB responses. Each $\mathbf{SKI_{pro}}$ session uses a transform $L_\mu$ on $x$ such that on $x'$ all coordinates are set to the scalar product between $\mu$ and $x$. Since $\mathbf{SKI_{lite}}$ never uses $x'$, $\mathcal{L}$ can be left empty.

[13] The $\mathcal{L}$ set will be later introduced as a *leakage scheme*; its purpose is to leak $L(x)$ in the case of a collusion-fraud/terrorist-fraud.

[14] This will be called the *F-scheme* and it will incorporate requirements towards (generalised) DF-, TF- and MF-resistance.

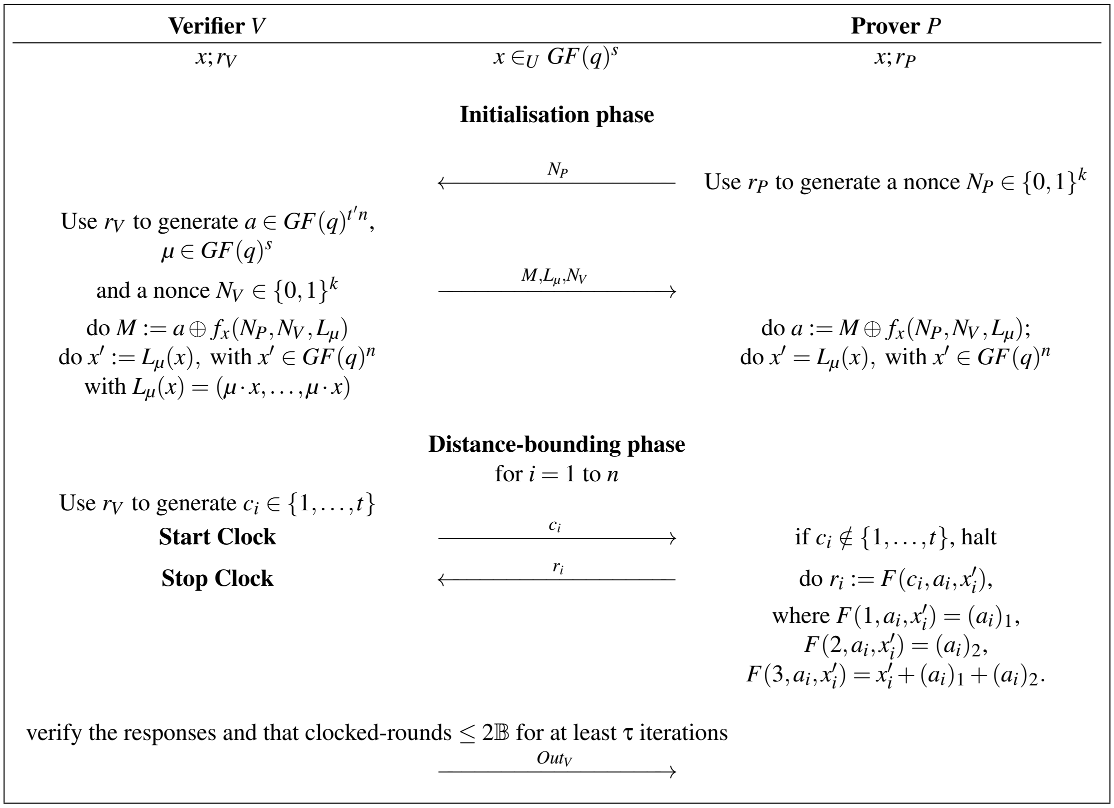


**Fig. 3.** The $\mathbf{SKI_{pro}}$ Distance-Bounding Protocol ($q = 2$, $t = 3$, $t' = 2$)

We note that both instances are efficient. Indeed, we could precompute the table of $F(\cdot, a_i, x'_i)$ and just do a table lookup to compute $r_i$ from $c_i$. For $\mathbf{SKI_{pro}}$, this can be done with a circuit of only 7 NAND gates and depth 4. For $\mathbf{SKI_{lite}}$, 3 NAND gates and a depth of 2 are enough.

However, in our design, we need the reuse of $x$ for protection against terrorist-fraud and/or collusion-fraud. Along these lines, the $\mathbf{SKI_{lite}}$ protocols do not assume circular-keying security (as defined below), but $\mathbf{SKI_{pro}}$ do.

In Appendix B, we consider other variants of **SKI** from [8,7] with different $F$-schemes (using, e.g., two-bit responses) which we still deem very practical.

***SKI Completeness (in Noisy Communications).*** We would like inquire on the suitability of the parameters; we verify for which parameters our proposal **SKI** is in line with Definition 2, i.e., it definitely terminates, but the completeness bound can be "tuned".

Each $(c_i, r_i)$ exchange is time-critical, so it is subject to errors. To address this, we introduce the probability $p_{noise}$ of one response being erroneous (à la Hancke-Kuhn [24]). Then, our protocol specifies that the verifier accepts only if the number of correct answers is at least $\tau$, where $\tau$ is an extra parameter. The probability that at least $\tau$ responses out of $n$ are correct is clearly given by:

$$B(n, \tau, 1 - p_{noise}) = \sum_{i=\tau}^{n} \binom{n}{i} (1 - p_{noise})^i p_{noise}^{n-i}$$

It is natural to choose $\tau$ (and other parameters) such that we operate with correct DB protocols, cnf. with Definition 2. I.e., the protocol is complete: honest communications succeed with high probability.

**Lemma 7.** *Let* $\varepsilon > 0$*. For* $\tau \leq (1 - p_{\mathsf{noise}} - \varepsilon)n$*, the* ***SKI*** *protocols are* $(1 - e^{-2\varepsilon^2 n})$*-complete.*

*Proof.* Due to the the Chernoff-Hoeffding bound [13,25], $\tau \leq (1 - p_{\mathsf{noise}} - \varepsilon)n$ implies $B(n, \tau, 1 - p_{noise}) \geq 1 - e^{-2\varepsilon^2 n}$. According to Definition 2, this makes the **SKI** protocols $(1 - e^{-2\varepsilon^2 n})$-complete. □

In practice, we may use a constant $p_{noise}$ (i.e., hard-coded in the protocol implementation). This also entails employing $\tau$ as some parameter which is linear in terms of $n$. A detailed analysis of the optimal selection of this threshold $\tau$ is provided in [16].

### 3.2 SKI: SECURITY-DRIVEN DESIGN & SECURITY ASSESSMENT

In this subsection, we discuss the design choices that we made in order to render the instances of **SKI** provably secure.

***PRF masking.*** Importantly, **SKI** applies a random mask $M$ on the output of $f_x$ to fix the problems raised in Boureanu *et al.* [6]. We call this *PRF masking*. As detailed in [6], when $M$ is not used (or equivalently, that $M$ is always set to 0), then we could construct a PRF such that, e.g., for all $x$ and $N_V$, $f_x(x, N_V, L)$ is a special value $a$ such that $F(c_i, a_i, x'_i)$ does not depend on $c_i$. This way, a malicious prover could set $N_P = x$ and predict the answer $F(c_i, a_i, x'_i)$ without having received the challenge $c_i$. Hence, he could mount a successful distance-fraud. By having the verifier decide $a$, **SKI** enforces that the distribution of $a$ cannot be influenced by a malicious prover.

***$F$-scheme.*** In our way to prove security, we need some notions related to the response-function $F$; these characterise the concept of $F$-*scheme*. At the same time, these concepts give the sufficient conditions to protect against *all* three frauds possible against the concrete **SKI** instances to follow. Such a characterisation is different from the approach in Avoine *et al.* [2], where a response-function based on secret sharing is proposed for the protection against terrorist-fraud *only*, but no formal justification was given to that end; also, the relation between the other frauds and the response-function was not addressed therein. Thus, we stress that using a secret sharing scheme in computing the responses may be too strong and/or insufficient to characterise the protection against frauds mounted onto DB protocols, and we amend this with Definition 8 and Definition 12.

**Definition 8** (**$F$-scheme**). *Let $t, t' \geq 2$. The response-function $F : \{1, \ldots, t\} \times GF(q)^{t'} \times GF(q) \to GF(q)$ gives an $F$-*scheme*, which is characterised as follows.*

- *We say that the $F$-scheme is* linear *if for all challenges $c_i$ in their domain, the $F(c_i, \cdot, \cdot)$ function is a linear form over the $GF(q)$-vector space $GF(q)^{t'} \times GF(q)$ which is non-degenerate in the $a_i$ component.*
- *We say the $F$-scheme is* pairwise uniform *if*

$$(\forall I \subsetneq \{1, \ldots, n\}, \#I \leq 2)(H(x'_i | F(c_i, a_i, x'_i)_{c_i \in I}) = H(x'_i)),$$

  *where $(a_i, x'_i) \in_U GF(q)^{t'} \times GF(q)$, $\#S$ denotes the cardinality of a set $S$, and $H$ denotes the Shannon entropy.*
- *We say the $F$-scheme is $t$-*leaking *if there exists a polynomial time algorithm $E$ such that for all $(a_i, x'_i) \in GF(q)^{t'} \times GF(q)$, we have $E\big(F(1, a_i, x'_i), \ldots, F(t, a_i, x'_i)\big) = x'_i$.*
- *Let $F_{a_i, x'_i}$ denote $F(\cdot, a_i, x'_i)$. We say that the $F$-scheme is* $\sigma$-bounded *if for any $x'_i \in GF(q)$, we have $\mathbb{E}_{a_i}\left(\max_y \left(\#(F^{-1}_{a_i, x'_i}(y))\right)\right) \leq \sigma$, where $x' \in GF(q)$ and the expected-value is $\mathbb{E}$ taken over $a_i \in GF(q)^{t'}$.*

We shortly discuss the definition above. The pairwise uniformity and the $t$-leaking property of the $F$-scheme say that knowing the complete table of the response-function $F$ for a given $c_i$ leaks $x'_i$, yet knowing only up to 2 entries challenge-response in this table discloses no information about $x'_i$. The $\sigma$-boundedness of the schemes says that the expected value (taken on the choice of the subsecrets $a_i$) of the largest preimage of the map $c_i \mapsto F(c_i, a_i, x'_i)$ is bounded by a constant $\sigma$. We have $\frac{t}{q} \leq \sigma \leq t$ due to the pigeonhole principle, since $\sum_y \#(F^{-1}_{a_i, x'_i}(y)) = t$.

In relation with the definitions of the $F$-schemes above, we now prove the following lemma.

**Lemma 9.** *The $F$-schemes used in $\mathbf{SKI_{pro}}$ are linear, pairwise uniform, $\frac{9}{4}$-bounded, and $t$-leaking. The $F$-scheme used in $\mathbf{SKI_{lite}}$ is linear, pairwise uniform, $\frac{3}{2}$-bounded, but not $t$-leaking.*

This lemma extends to Lem. 15 given and proven in Appendix B.

***Leakage scheme.*** We can consider several sets $\mathcal{L}$ of transformations to be used in the PRF-instance, of the **SKI** initialisation phase. The idea of the set $\mathcal{L}$ is that, when leaking some noisy versions of $L(x)$ for some random $L \in \mathcal{L}$, the adversary can reconstruct $x$ without noise.

We define $\mathcal{L}_{\mathsf{classic}} = \{L\}$, with only one transformation: the identity function $L$, i.e., $L(x) = x$. Unfortunately, this is not sufficient to add protection against collusion fraud due to Hancke [23]: given a constant $\theta$, a malicious prover could select a vector $e$ of Hamming weight $n - \tau + \theta$ and provide the full table of all $c_i \mapsto F(c_i, a_i, x_i)$ functions, only that some entries in the table had been changed. Namely, for each $i \in \{1, \dots, n\}$ with $e_i = 1$, the dishonest prover flips $F(c_i, a_i, x_i)$ in this leaked table. Then, we would have $\gamma = \left(1 - \frac{1}{t}\right)^{\theta}$, but this helped attacker can only reconstruct $x + e$. Using multiple coerced provers $P^*$ will not reveal anything more, if the function $g(x)$ giving $e$ is *deterministic* (i.e., then, several runs would have no randomised, adaptive choices of $g(x)$, coming from $P^*$'s). Depending on such functions $g$, and since $n - \tau$ is linear, recovering $x$ takes exponential time. So, the value of $x + g(x)$ is not enough to run a MiM attack since we need $x$ to evaluate $f_x$.

We consider the leakage scheme $\mathcal{L}_{\mathsf{bit}}$ of $\mathbf{SKI_{pro}}$, consisting of all $L_\mu$ transforms, where $L_\mu$ is defined from a vector $\mu \in GF(q)^s$ by

$$L_\mu(x) = (\mu \cdot x, \dots, \mu \cdot x)$$

More formally, we introduce the following notion.

**Definition 10 (Leakage scheme).** *Let $\mathcal{L}$ be a set of linear functions from $GF(q)^s$ to $GF(q)^n$. Given $x \in GF(q)^s$ and a ppt. algorithm $e(x, L; r)$, we define an oracle $O_{\mathcal{L},x,e}$ producing a random pair $(L, e(x, L))$ with $L \in_U \mathcal{L}$. $\mathcal{L}$ is a* $(T, r)$-leakage scheme *if there exists an oracle ppt. algorithm $\mathcal{A}^{\langle\cdot\rangle}$ such that for all $x \in GF(q)^s$, for all ppt. e,* $\Pr[\mathcal{A}^{O_{\mathcal{L},x,e}} = x] \geq \Pr_r[d_H(e(x, L), L(x)) < T]^r$*, where $d_H$ denotes the Hamming distance.*

Intuitively, this means that based on $r$ values of $L$ and a noisy $L(x)$, we can decode $x$.

**Lemma 11.** *$\mathcal{L}_{\mathsf{bit}}$ is a $(\frac{n}{2}, s)$-leakage scheme.*

*Proof.* $\mathcal{A}$ calls the oracle $s$ times, then —by computing the majority– $\mathcal{A}$ deduces $\mu \cdot x$ with probability $p$, for each of the obtained $\mu$. We run $O_{\mathcal{L},x,e}$ until we collect $s$ linearly independent $\mu$ values. All the $s$ obtained $\mu \cdot x$ are correct with probability $p^s$. Then, we deduce $x$ by solving a linear system. □

***Circular-Keying Security.*** On our way to prove the security of the **SKI** protocols, we need and hereby introduce the notion of *security against circular-keying*. This notion of security will help protect against MiM, in the context in which the key $x$ is used in the response-function to protect against TF. To attain provable security against MiM attackers, we take *secure* circular-keying as an extra assumption to the PRF $(f_x)_{x \in GF(q)^s}$ to handle the reuse of a fixed $x$ outside of a PRF instance $f_x$.

**Definition 12 (Circular-Keying).** *Let $s$ be some security parameter, let $b$ be a bit, let $q \geq 2$, let $m \in Poly(s)$, and let $x, \bar{x} \in GF(q)^s$ be two row-vectors. Let $(f_x)_{x \in GF(q)^s}$ be a family of (keyed) functions, e.g., $f_x : \{0,1\}^* \to GF(q)^m$. For an input $y$, the output $f_x(y)$ can be represented as a row-vector in $GF(q)^m$.*

*We define an oracle $O_{f_x,\bar{x}}$ such that upon a query of form $(y_i, A_i, B_i)$, with $A_i \in GF(q)^s$, $B_i \in GF(q)^m$, it answers $(A_i \cdot \bar{x}) + (B_i \cdot f_x(y_i))$. The game $Circ_{f_x,\bar{x}}$ of* circular-keying *with an adversary $\mathcal{A}$ is described as follows: we set $b_{f_x,\bar{x}} := \mathcal{A}^{O_{f_x,\bar{x}}}$, where the queries $(y_i, A_i, B_i)$ from $\mathcal{A}$ must follow the restriction that*

$$(\forall c_1, \dots, c_k \in GF(q))\left(\#\{y_i; c_i \neq 0\} = 1, \sum_{j=1}^{k} c_j B_j = 0 \Longrightarrow \sum_{j=1}^{k} c_j A_j = 0\right).$$

*We say that the family of functions $(f_x)_{x \in GF(q)^s}$ is an* $(\varepsilon, C, Q)$-circular-PRF *if for any ppt. adversary $\mathcal{A}$ making $Q$ queries and having complexity $C$, it is the case that* $\Pr[b_{f_x,x} = b_{f^*,\bar{x}}] \leq \frac{1}{2} + \varepsilon$*, where the probability is taken over the random coins of $\mathcal{A}$ and over the random selection of $x, \bar{x} \in GF(q)^s$ and the random function $f^*$.*

The condition on the queries means that for any set of queries with the same value $y_i$, any linear combination making $B_j$ vanish makes $A_j$ vanish at the same time. (Otherwise, we would trivially extract some information about $\bar{x}$ by linear combinations.)

We note that it is possible to create secure circular-keying in the random oracle model (ROM) [5]. This is a "sanity check" for our circular-keying notion.

**Lemma 13.** *Let $f_x(y) = H(x,y)$, where $H$ is a random oracle, $x \in \{0,1\}^s$, and $y \in \{0,1\}^*$. Then, $f$ is a $(T2^{-s}, T, Q)$-circular PRF for any $T$ and $Q$.*

*Proof.* Let $(y, A_i, B_i)$, $i \in 1,\ldots,k$, be some queries to $O_{f_x,\bar{x}}$ that share the same $y$, made by some $\mathcal{A}$, making no query to $H$. We define the matrices $A = (A_1 \cdots A_k)^T$ and $B = (B_1 \cdots A_k)^T$. Thus, $\mathcal{A}$ learns $A\bar{x} + BH(x,y)$. Now, w.l.o.g., assume that $\mathcal{A}$ multiplies $A\bar{x} + BH(x,y)$ to the left by a conveniently chosen, invertible matrix $P$, i.e., such that $PB = (I_p\ 0)^T$ where $I_p$ is the identity matrix of rank $p$ of $B$ and 0 is a zero matrix block.

By taking $c = c'P$ with $c' = (0,\ldots,0,1,0\ldots,0)$, where 1 appears at some position $j$ for any $j > p$, we have that $cB = 0$. Then, by circular keying, we have that $cA = 0$. Thus, all rows from positions beyond $p$, i.e., $p+1, p+2, \ldots$ "downwards" inside the matrix $PA$, are filled with zeroes. Thus, $\mathcal{A}$ learns $A'\bar{x} + H(x,y)$, where $A'$ is the "upper-part" of PA, i.e., above the $p$th row. We have shown that $\mathcal{A}$ is equivalent to an adversary learning $A'\bar{x} + H(x,y)$ for some random matrix $A'$. So, we can replace $H(x,y)$ by something random and the advantage of the adversary $\mathcal{A}$ in this game would not change.

Now, in the random oracle model, $\mathcal{A}$ also queries $H$. We consider the hybrids of $\mathcal{A}$ in which the first queries to $H$ are simulated and the hybrid stops before making the next query to $H$ (there are up to $T$ hybrids). We apply the previous argument to the hybrids to show that they cannot query $H$ with $x$, except by guessing it with probability $2^{-s}$. □

We proceed with inspecting the rest of the security requirements on these protocols.

**Theorem 14.** *The **SKI** protocols are secure distance-bounding protocols, i.e.,:*

- *A. If the $F$-scheme is linear and $\sigma$-bounded, if $(f_x)_{x \in GF(q)^n}$ is a $(\varepsilon, nN, C)$-circular PRF, then the **SKI** protocols offer $\alpha$-resistance to distance-fraud, with $\alpha = B(n, \tau, \max(\frac{\sigma}{t}, \frac{1}{t})) + \varepsilon$, for attacks limited to complexity $C$ and $N$ participants. So, we need $\frac{\tau}{n} > \frac{\sigma}{t}$ for security.*
- *B. If the $F$-scheme is linear and pairwise uniform, if $(f_x)_{x \in GF(q)^n}$ is a $(\varepsilon, n(\ell + z + 1), C)$-circular PRF, if $\mathcal{L}$ is a set of linear mappings, the **SKI** protocols are $\beta$-resilient against (non-narrow) MiM attackers with parameters $\ell$ and $z$ and a complexity bounded by $C$, $\beta = B(n, \tau, \frac{1}{t} + \frac{t-1}{t} \times \frac{1}{q}) + 2^{-k}\left(\frac{\ell(\ell-1)}{2} + \frac{z(z+1)}{2}\right) + \varepsilon$. So, we need $\frac{\tau}{n} > \frac{1}{t} + \frac{t-1}{t} \times \frac{1}{q}$ for security.*
- *$B'$. If the $F$-scheme is linear and pairwise uniform, if $(f_x)_{x \in GF(q)^n}$ is a $(\varepsilon, n(\ell + z + 1), C)$-PRF, if the function $F(c_i, a_i, \cdot)$ is constant for each $c_i, a_i$, the **SKI** protocols are $\beta$-resilient against (non-narrow) MiM attackers with parameters $\ell$ and $z$ and a complexity bounded by $C$, $\beta = B(n, \tau, \frac{1}{t} + \frac{t-1}{t} \times \frac{1}{q}) + 2^{-k}\left(\frac{\ell(\ell-1)}{2} + \frac{z(z+1)}{2}\right) + \varepsilon$. So, we need $\frac{\tau}{n} > \frac{1}{t} + \frac{t-1}{t} \times \frac{1}{q}$ for security.*
- *C. If the $F$-scheme is $t$-leaking, if $\mathcal{L}$ is a $(T, r)$-leakage scheme, for all $\theta \in ]0,1[$, the **SKI** protocols offer $\left(\gamma, \gamma'\right)$-resistance to collusion-fraud, for $\gamma \geq B(T, T + \tau - n, \frac{t-1}{t})^{1-\theta}$, $\gamma^{-1}$ is polynomially bounded, and $\gamma' = \left(1 - B(T, T + \tau - n, \frac{t-1}{t})^{\theta}\right)^r$. So, we need $\frac{\tau}{n} > 1 - \frac{T}{tn}$ for security.*

The proof of Th. 14.B$'$ is similar (and simplified) as the one of Th. 14.B. So, we prove the A, B, and C parts only.

*Proof (Th. 14.A).* For each key $x'$ which is different from $x$ and for which there is a $P(x')$ close to $V$ (so, there is no $P^*(x')$ anywhere, due to the distance-fraud model), we apply the circular-PRF reduction. (Details as for why we can apply this reduction will appear in the proof of Th. 14.B.) We are losing a probability up to $\varepsilon$ in this reduction.

We recall that if the $F$-scheme is linear, then $F(c_i, a_i, x'_i)$ must be non-degenerate in $a_i$. So, answers $r_i$ coming from $P(x')$ instead of $P^*(x)$ are correct with probability $\frac{1}{t}$, since $a_i$ is random, after the circular-PRF reduction.

If $r_i$ now comes from $P^*$, due to Lem. 1, $r_i$ must be a function independent from $c_i$. I.e., $P^*$ must have $F(c_i, a_i, x'_i)$ ready, before $c_i$ arrives from $V$. So, for any secret $x$ and $a$, the probability to get one response right is given by $p_i = \Pr_{c_i \in \{1,\ldots,t\}}[r_i = F(c_i, a_i, x'_i)]$.

Thanks to PRF masking, the distribution of the $a_i$'s is uniform. Namely, $P^*$ cannot influence their distribution by selecting $N_P$ maliciously.

To establish the probabilities $p_i$, consider the partitions $I_j$, $j \in \{1,\dots,t\}$ as follows: for $i \in I_j$, the largest preimage of $F_{a_i,x'_i}: c_i \mapsto F(c_i,a_i,x'_i)$ has size $j$, i.e., $\max_y \left(\#(F^{-1}_{a_i,x'_i}(y)\right) = j$. Then, we are looking at the probability

$$P_j(x'_i) := \Pr_{a_i}\left[\max_y \left(\#(F^{-1}_{a_i,x'_i}(y))\right) = j\right],$$

where $\#(S)$ denotes the cardinality of a set $S$. Given $x'$ fixed, each iteration has a probability to succeed equal to

$$\frac{P_1}{t} + \frac{2P_2}{t} + \cdots + \frac{tP_t}{t} = \frac{\sigma}{t}$$

So, the probability to win the experiment is bounded by $p = B(n,\tau,\frac{\sigma}{t})$. □

*Proof (Th. 14.B).* In the next, $P(\dots)$ and $V(\dots)$ respectively denote the algorithm/(part of the) protocol of a generic prover $P$ and that of a generic verifier $V$, out of the $\ell$ provers and $z+1$ verifiers in this attack-game, run on specific parameters to be specified in-line. We herein denote $V$ in the MiM-resistance definition as $V_{z+1}$.

We use the game-reduction methodology [36] to prove this lemma. Let $Game_0$ be the non-narrow MiM attack-game described in Definition 4 played by $\mathcal{A}$ against the honest parties in a **SKI** protocol.

Below we consider a prover $P_j$ and a verifier $V_k$ in an experiment, $j \in \{1,\dots,\ell\}, k \in \{1,\dots,z+1\}$. Let $(N_{P,j},\overline{M}_j,\overline{L}_j,\overline{N}_{V,j})$ be the values of the nonces $(N_P,N_V)$, of the mask $M$, and of the transformation $L$ that the prover $P_j$ generates or sees respectively, and $(\overline{N}_{P,k},M_k,L_k,N_{V,k})$ be the values of the nonces $(N_P,N_V)$, mask $M$, and transformation $L$ that a verifier $V_k$ generates or sees at his turn, $j \in \{1,\dots,\ell\}, k \in \{1,\dots,z+1\}$.

We apply a reduction by failure-event to prove that the game $Game_0$ is indistinguishable to the adversary $\mathcal{A}$ from a game $Game_1$ where no repetitions on $N_{P,j}$ or on $N_{V,k}$ happen for $j \in \{1,\dots,\ell\}$, $k \in \{1,\dots,z+1\}$, i.e., there is no collision on the nonces generated by the provers and there is no collision on the nonces of the verifiers.

Assume that $F$ is the event that at least a collision as above happens, i.e.,

$$F \equiv \Big(\bigvee_{0<i<j\leq\ell} (N_{P,i} = N_{P,j})\Big) \bigvee \Big(\bigvee_{0<i'<j'\leq z+1} (N_{V,i'} = N_{V,j'})\Big).$$

We want to have that, from the point of view of the adversary $\mathcal{A}$, $Game_0 \wedge \neg F \Leftrightarrow Game_1 \wedge \neg F \Leftrightarrow Game_1$. But,

$$\|Pr[A \text{ wins in } Game_0] - Pr[A \text{ wins in } Game_1]\| \leq \Pr[F].$$

Then, $\Pr[F] \leq 2^{-k}\left(\frac{\ell(\ell-1)}{2} + \frac{z(z+1)}{2}\right)$.

Since the $F$-scheme is linear, we can write $F(c_i,a_i,x'_i) = u_i(c_i)x'_i + (v_i(c_i)\cdot a_i)$ where $u_i(c_i) \in GF(q), v_i(c_i) \in GF(q)^{t'}$. Note that, in terms of $i$, the $(v_i(1),\dots,v_i(t))$'s span independent linear spaces. In $Game_1$, each $(N_P,N_V,L,i)$ tuple can be invoked only twice (with a prover and a verifier) by the adversary. The pairwise uniformity of the $F$-scheme implies that $yv_i(c_i) + y'v_i(c'_i) = 0$ implies $yu_i(c_i) + y'u_i(c'_i) = 0$ for all $c_i, c'_i \in \{1,\dots,t\}$ and all $y, y' \in GF(q)$. So, we deduce that the condition to apply the circular-keying reduction is fulfilled. We can thus apply the circular-PRF reduction and reduce to $Game_2$, where $F(c_i, f_x(N_P,N_V,L)_i, x'_i)$ is replaced by $u_i(c_i)\tilde{x}_i + (v_i(c_i)\cdot f^*(N_P,N_V,L)_i)$, where $f^*$ is a random function. This reduction has a probability loss of up to $\varepsilon$.

From here, we use a simple bridging step to say that the adversary $\mathcal{A}$ has virtually no advantage over $Game_2$ and a game $Game_3$, where the vector $a = f^*(N_P,N_V,L)$ is selected at random; we recall that this is the case since there is no repetition on $N_P$ and $f^*$ is a random function. The $(N_P,N_V,L)$ triplet used by $V$ in the attack phase can be used by only one $P_j$, in the attack phase as well, where $j \in \{m+1,\dots,\ell\}$. We can simulate all other $P$'s and $V$'s based on a (simulated) random $a$. This reduces to an adversary making no use of the learning phase and using only $P_j$ and $V$ in the attack phase.

So, the probability $p$ of $\mathcal{A}$ of succeeding in $Game_3$ is the probability that at least $\tau$ rounds have a correct $r_i$. Due to Lem. 1, $r_i$ must be computed by $\mathcal{A}$ (and not $P_j$). Getting $r_i$ correct for $c_i$ can thus be attained in two distinct ways: 1. in the event $e1$ of guessing $c'_i = c_i$ and sending it beforehand to $P_j$ and getting the correct response $r_i$, or 2. in the event $e2$ of simply guessing the correct answer $r_i$ (for a challenge $c'_i \neq c_i$). So, $p = B(n,\tau,\Pr[e1]+\Pr[e2]) = B(n,\tau,\frac{1}{t} + \frac{t-1}{t}\times\frac{1}{q})$. □

*Proof (Th. 14.C).* Assume as per the requirement for resistance to collusion-fraud that there is an experiment $exp^{\mathsf{CF}} = (P^*(x) \longleftrightarrow \mathcal{A}^{\mathsf{CF}}(r_{\mathsf{CF}}) \longleftrightarrow V_0(y; r_{V_0}))$, with $P^*$ a coerced prover who is far away from $V_0$ and that $\Pr_{r_{V_0}, r_{\mathsf{CF}}}[Out_{V_0} = 1] = \gamma$. Given some random $c_1, \ldots, c_n$ from the verifier, we define the random variable $View_i$ as being the view of $\mathcal{A}^{\mathsf{CF}}$ before receiving $c_i$ from $V$, and the random variable $w_i$ being all the information that $\mathcal{A}^{\mathsf{CF}}$ has received from $P^*$ before the time when sending out $r_i$ would become critical (i.e., before it would be too late to send $r_i$ on to $V_0$). This answer $r_i$ done by $\mathcal{A}^{\mathsf{CF}}$ is formalised in Lem. 1. So, $r_i := \mathcal{A}^{\mathsf{CF}}(View_i \| c_i \| w_i)$.

Let $C_i$ be the set of all possible $c_i$'s on which the functions $\mathcal{A}^{\mathsf{CF}}(View_i \| \cdot \| w_i)$ and $F(., a_i, x'_i)$ match (i.e., $\mathcal{A}^{\mathsf{CF}}$ answers correctly to the challenge $c_i$ at round $i$). Let $S$ be the set of $i$'s such that $c_i \in C_i$ (i.e., $\mathcal{A}^{\mathsf{CF}}$ answers correctly at round $i$). Finally, let $R$ be the set of $i$'s such that $\#C_i = t$ (i.e., $\mathcal{A}^{\mathsf{CF}}$ answers correctly at round $i$ whatever the challenge). I.e., $C_i = \{c \in \{1, \ldots, t\} \,|\, \mathcal{A}^{\mathsf{CF}}(View_i \| c \| w_i) = F(c, a_i, x'_i)\}$, $S = \{i \in \{1, \ldots, n\} \,|\, c_i \in C_i\}$, and $R = \{i \in \{1, \ldots, n\} \,|\, \#C_i = t\}$. The adversary $\mathcal{A}$ succeeds in $exp^{\mathsf{CF}}$ if $\#S \geq \tau$, i.e., if he can pass at least $\tau$ rounds, for the challenges that $V_0$ will fix in those rounds.

For terrorist-fraud resistance, we would also like that—in the second, MiM experiment—the adversary $\mathcal{A}_2$ can answer $\tau$ rounds (or more), no matter what the challenge, i.e., in this way, $\mathcal{A}$ could extract $x$ and the TF would be invalid. In other words, we would like that $\#R$ is large, i.e. $\#R > n - T$ so that we can decode.

So, if we were to pick a set of challenges such that $\#S \geq \tau$ and $\#R \leq n - T$, we should select a good challenge (from no more than $t-1$ existing out of $t$), for at least $T + \tau - n$ rounds out of $T$. In other words, $\Pr[\#S \geq \tau, \#R \leq n - T] \leq B(T, T + \tau - n, \frac{t-1}{t})$. But, by the hypothesis, $\Pr[\#S \geq \tau] \geq \gamma$. So, we deduce immediately that $\Pr[\#R \leq n - T | \#S \geq \tau] \leq \gamma^{-1} B(T, T + \tau - n, \frac{t-1}{t})$. Therefore, $\Pr[\#R > n - T | \#S \geq \tau] \geq 1 - \gamma^{-1} B(T, T + \tau - n, \frac{t-1}{t})$.

We use $m = \ell = z = O(\gamma^{-1} r)$ (i.e., $\mathcal{A}_2$ will directly impersonate $P$ to $V$ after $\mathcal{A}_1$ ran $m$ times the collusion fraud, with $P^*$ and $V$). We define $\mathcal{A}_2$ such that, for each execution of the collusion fraud with $P^*$ and $V$, it gets $View_i$, $w_i$. For each $i$, $\mathcal{A}_2$ computes the table $c \mapsto \mathcal{A}^{\mathsf{CF}}(View_i \| c \| w_i)$ and apply the $t$-leaking function $E$ of the $F$-scheme on this table to obtain $y_i = E(c \mapsto \mathcal{A}^{\mathsf{CF}}(View_i \| c \| w_i))$. For each $i \in R$, the table matches the one of $c \mapsto F(c, a_i, x'_i)$ with $x' = L(x)$, and we have $y_i = x'_i$. So, $\mathcal{A}_2$ computes a vector $y$. If $V$ accepts the proof, then $y$ coincides with $L(x)$ on at least $n - T + 1$ positions, with a probability of at least $p := 1 - \gamma^{-1} B(T, T + \tau - n, \frac{t-1}{t})$. That is, after $O(\gamma^{-1})$ runs, $\mathcal{A}_2$ implements an oracle which produces a random $L \in \mathcal{L}$ and a $y$ which has a Hamming distance to $L(x)$ up to $T - 1$.

By applying the leakage scheme decoder $e$ on this oracle, with $r$ samples, it can fully recover $x$, with probability at least $p^r$. Then, by taking $\gamma = B(T, T + \tau - n, \frac{t-1}{t})^{1-\theta}$ and $\gamma' = \left(1 - B(T, T + \tau - n, \frac{t-1}{t})^{\theta}\right)^s$, we obtain our result. $\square$

Thus, under the circumstances where protection against terrorist-fraud and/or collusion-fraud[15] is not of primary importance, one can use our proposed $\mathbf{SKI_{lite}}$ protocols, the security of which does not rely on the assumption of circular-keying security.

Following Lem. 9 and Th. 14, it is clear that the probabilities $\alpha$ and $\beta$ to succeed respectively in a distance-fraud and MiM, against the **SKI** protocols are based on:

| | $\mathbf{SKI_{pro}}$ | $\mathbf{SKI_{lite}}$ |
|---|---|---|
| $\alpha$: | $B(n, \tau, \frac{3}{4})$ | $B(n, \tau, \frac{3}{4})$ |
| $\beta$: | $B(n, \tau, \frac{2}{3})$ | $B(n, \tau, \frac{3}{4})$ |
| $\gamma$: | $B(\frac{n}{2}, \tau - \frac{n}{2}, \frac{2}{3})$ | 1 |

***SKI's parameters:*** Let $\varepsilon > 0$. Remember (from page 9, Lem. 7) that the **SKI** protocols are $(1 - e^{-2\varepsilon^2 n})$-complete if $\tau$ is at most $(1 - p_{\mathsf{noise}} - \varepsilon)n$.

According to the data in the table above, we must take $1 - p_{\mathsf{noise}} - \varepsilon \geq \frac{\tau}{n} \geq \frac{3}{4} + \varepsilon$ to make the above instances of **SKI** secure, with a failure probability bounded $\beta$ by $e^{-2\varepsilon^2 n}$ (by the Chernoff-Hoeffding bound [13,25]).

By changing the $F$-scheme, we can decrease the value $\frac{3}{4}$ in $\alpha$. For instance, using the Shamir secret sharing [35], we reduce it to $\frac{5}{8}$, as shown in Appendix B.

If we require TF-resistance (as per Th. 14.C), we also get a constraint of $\frac{\tau}{n} > \frac{5}{6} + \frac{\varepsilon}{2}$, similarly.

We observe that Th. 14 is tight for $\mathbf{SKI_{pro}}$ and $\mathbf{SKI_{lite}}$, due to the attacks shown in [8,7].

[15] It is clear that these $\mathbf{SKI_{lite}}$ protocols do not protect against terrorist-fraud (given the $F$-scheme used inside them).

## 4 Conclusion

In this paper, we have specified distance-bounding protocols and their security requirements, i.e., resistance to (generalised) distance-fraud, man-in-the-middle, terrorist-fraud attacks, in a general formalism for modelling location-driven security protocols developed herein. We also proposed the formal proofs for what is to our knowledge the first provably secure class of practical protocols for distance-bounding, by identifying the requirements on the building blocks (i.e., the $F$-scheme, the leakage scheme, PRF masking, and the circular-keying security). Thus, these protocols are practical, efficient and provably secure. As a by-product, we introduced (at least) a new security notion, i.e., circular-keying for pseudorandom functions (PRFs); this models the employment of a PRF, with possible linear reuse of the key.

Our protocols are secure when the level of noise is below $\frac{1}{6}$, using $t = 3$ possible challenges per round, and a leakage scheme requiring $s$ executions. We leave the problem of improving these parameters as an open problem.

## References


1. G. Avoine, M. Bingöl, S. Kardas, C. Lauradoux, and B. Martin. A framework for analyzing RFID distance bounding protocols. *Journal of Computer Security*, 19(2):289–317, 2011.
2. G. Avoine, C. Lauradoux, and B. Martin. How Secret-sharing can Defeat Terrorist Fraud. In *Proceedings of the 4th ACM Conference on Wireless Network Security – WiSec'11*, Hamburg, Germany, June 2011. ACM Press.
3. G. Avoine and A. Tchamkerten. An efficient distance bounding RFID authentication protocol: Balancing false-acceptance rate and memory requirement. In *Proceedings of Information Security*, volume 5735, pages 250–261, 2009.
4. A. Bay, I. C. Boureanu, A. Mitrokotsa, I.-D. Spulber, and S. Vaudenay. The Bussard-Bagga and Other Distance-Bounding Protocols under Attacks. In *the 88th China International Conference on Information Security and Cryptology (Inscrypt 2012)*, 2012.
5. 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*, CCS '93, pages 62–73, New York, NY, USA, 1993. ACM.
6. I. Boureanu, A. Mitrokotsa, and S. Vaudenay. On the Pseudorandom Function Assumption in (Secure) Distance-Bounding Protocols. In A. Hevia and G. Neven, editors, *Progress in Cryptology – LATINCRYPT 2012*, Lecture Notes in Computer Science, pages 100–120. Springer, 2012.
7. I. Boureanu, A. Mitrokotsa, and S. Vaudenay. Secure & Lightweight Distance-Bounding. In *Proceedings of LIGHTSEC 2013*, 2013. to appear.
8. I. Boureanu, A. Mitrokotsa, and S. Vaudenay. Towards Secure Distance Bounding. In *the 20th anniversary annual Fast Software Encryption (FSE 2013)*, 2013. Invited Talk by Serge Vaudenay, to appear.
9. S. Brands and D. Chaum. Distance-Bounding Protocols (Extended Abstract). In *EUROCRYPT*, pages 344–359, 1993.
10. L. Bussard and W. Bagga. Distance-Bounding Proof of Knowledge Protocols to Avoid Terrorist Fraud Attacks. Technical Report RR-04-109, Institute EURECOM, May 2004.
11. L. Bussard and W. Bagga. Distance-bounding proof of knowledge to avoid real-time attacks. In *Security and Privacy in the Age of Ubiquitous Computing, IFIP TC11 20th International Conference on Information Security (SEC 2005), May 30 - June 1, 2005, Chiba, Japan*, pages 223–238. Springer, 2005.
12. N. Chandran, V. Goyal, R. Moriarty, and R. Ostrovsky. Position based cryptography. In S. Halevi, editor, *Advances in Cryptology - CRYPTO 2009, 29th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 16-20, 2009. Proceedings*, volume 5677 of *Lecture Notes in Computer Science*, pages 391–407. Springer, 2009.
13. H. Chernoff. A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. *The Annals of Mathematical Statistics*, 23(4):493–507, 1952.
14. C. Cremers, K. B. Rasmussen, and S. Čapkun. Distance hijacking attacks on distance bounding protocols. In *IEEE Symposium on Security and Privacy*, pages 113–127, 2012.
15. Y. Desmedt. Major Security Problems with the "Unforgeable" (Feige)-Fiat-Shamir Proofs of Identity and How to Overcome Them. In *Proceedings of the 6th Worldwide Congress on Computer and Communications Security and Protection - SecuriCom '88*, pages 147–159, Paris, France, 15-17 March 1988. SEDEP.
16. C. Dimitrakakis, A. Mitrokotsa, and S. Vaudenay. Expected Loss Bounds for Authentication in Constrained Channels. In *Proceedings of INFOCOM 2012*, pages 478–485, Orlando, FL, USA, March 2012. IEEE press.
17. S. Drimer and S. J. Murdoch. Keep your enemies close: distance bounding against smartcard relay attacks. In *Proceedings of 16th USENIX Security Symposium*, pages 7:1–7:16, Berkeley, CA, USA, 2007. USENIX Association.
18. U. Dürholz, M. Fischlin, M. Kasper, and C. Onete. A formal approach to distance bounding RFID protocols. In *Proceedings of the $14^{th}$ Information Security Conference ISC 2011*, LNCS, pages 47–62. SPRINGER, 2011.
19. M. Fischlin and C. Onete. Subtle kinks in distance-bounding: an analysis of prominent protocols. In *Proceedings of WISEC 2013*, pages 195–206. ACM, 2013.
20. Ford. Safe and Secure *SecuriCode*$^{TM}$ Keyless Entry. http://www.ford.com/technology/, 2011.
21. A. Francillon, B. Danev, and S. Čapkun. Relay attacks on passive keyless entry and start systems in modern cars. Cryptology ePrint Archive, Report 2010/332, 2010. `http://eprint.iacr.org/`.
22. O. Goldreich. *Foundations of Cryptography: Volume 1*. Cambridge University Press, New York, NY, USA, 2006.
23. G. P. Hancke. Distance bounding for RFID: Effectiveness of terrorist fraud. In *Proceedings of IEEE RFID-TA*. IEEE, 2012.

24. G. P. Hancke and M. G. Kuhn. An RFID distance bounding protocol. In *SECURECOMM*, pages 67–73, 2005.
25. W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):pp. 13–30, 1963.
26. G. Kapoor, W. Zhou, and S. Piramuthu. Distance bounding protocol for multiple RFID tag authentication. In C.-Z. Xu and M. Guo, editors, *Proceedings of the 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing - Volume 02 – EUC'08*, pages 115–120, Shanghai, China, December 2008. IEEE Computer Society.
27. C. H. Kim and G. Avoine. RFID distance bounding protocol with mixed challenges to prevent relay attacks. In *Proceedings of the 8th International Conference on Cryptology and Networks Security (CANS 2009)*, volume 5888, pages 119–131, 2009.
28. C. H. Kim, G. Avoine, F. Koeune, F. Standaert, and O. Pereira. The swiss-knife RFID distance bounding protocol. In *International Conference on Information Security and Cryptology – ICISC*, Lecture Notes in Computer Science. Springer-Verlag, December 2008.
29. J. Munilla and A. Peinado. Distance bounding protocols for RFID enhanced by using void-challenges and analysis in noisy channels. *Wireless Communications and Mobile Computing*, 8:1227–1232, November 2008.
30. J. Munilla and A. Peinado. Security Analysis of Tu and Piramuthu's Protocol. In *New Technologies, Mobility and Security – NTMS'08*, pages 1–5, Tangier, Morocco, November 2008. IEEE Computer Society.
31. J. Munilla and A. Peinado. Attacks on a Distance Bounding Protocol. *Computer Communications*, 33:884–889, 2010.
32. K. B. Rasmussen and S. Čapkun. Realization of RF distance bounding. In *Proceedings of the 19th USENIX conference on Security*, USENIX Security'10, pages 25–25, Berkeley, CA, USA, 2010. USENIX Association.
33. J. Reid, J. M. Gonzalez Nieto, T. Tang, and B. Senadji. Detecting Relay Attacks with Timing-based Protocols. In *ASIACCS '07: Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security*, pages 204–213. ACM, 2007.
34. A. Schuster and J. Nicholson. *An Introduction to the Theory of Optics*. Edward Arnold, London, 3rd edition, 1924.
35. A. Shamir. How to share a secret. *Communications of the ACM*, 22:612–613, November 1979.
36. V. Shoup. Sequences of Games: a Tool for Taming Complexity in Security Proofs. Manuscript, 2006.
37. D. Singelée and B. Preneel. Distance Bounding in Noisy Environments. In *Proceedings of the European Workshop on Security and Privacy in Ad-Hoc and Sensor Networks (ESAS)*, volume 4572 of *LNCS*, pages 101 –115. Springer-Verlag, 2007.
38. B. Toiruul, K. O. Lee, and J. M. Kim. SLAP - a secure but light authentication protocol for RFID based on modular exponentiation. In *International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies*, pages 29–34, November 2007.
39. Y.-J. Tu and S. Piramuthu. RFID distance bounding protocols. In *Proceedings of the First International EURASIP Workshop on RFID Technology*, 2007.
40. S. Vaudenay. On privacy models for RFID. In *Proceedings on Advances in cryptology*, ASIACRYPT '07, pages 68–87, New York, NY, USA, 2007. Springer-Verlag New York, Inc.

## A A Communication Model

We introduce a model for distance-bounding protocols. We first specify the main ideas at a high-level and then, in Section A.2, we formalise our communication and our threat model.

### A.1 General Communication Principles

We impose the following **gold principles**: 1. participants have a location; 2. messages travelling one unit of distance between two locations require one time-unit for delivery; 3. messages under transmission are broadcast and become readable at a location when they physically reach its proximity. We now explain the above in more depth and add some extra specifications.

A participant has a physical location, modelled as a centre of a sphere with the radius of one distance-unit. A sender $S$ who wants to send a message to a receiver $R$ just broadcasts the message, setting $R$ as the aimed "delivery address". Every time-unit, a message sent by $S$ moves from the sphere centred on $S$ to another sphere with a radius augmented by one unit (see Fig. 4). Participant $R$ can read the message as soon as the growing sphere on which the message is travelling includes $R$.

Honest participants are supposed to read only the messages for which they are the purported recipient.

There is no implicit authentication: received messages may have been previously sent by any participant.

The adversary can change the destinator to himself (so that the legitimate receiver does not read the corresponding message).

In the following, we give further, more formal explanations on these, as well as on time-increments and the communication model.

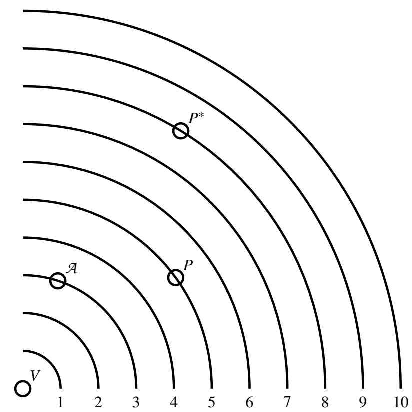


**Fig. 4.** Sketch of Message-Transmitting Model: A message send by $V$ is broadcast and travels at one unit of distance per clock cycle. Assume $P$ is the purported recipient. However, $\mathcal{A}$ can read the message two clock cycles before $P$, whereas $P^*$ must wait three clock cycles more than $P$ before the message reaches him.

### A.2 Computation and Communication Models

***Formalised Participants.*** Each participant in the protocol is formally described by an interactive Turing machine (ITM). The ITMs we use in this formalisation have the following tapes: an input tape, a random tape, an incoming communication tape, an outgoing communication tape, a read/write working tape, and an output tape. Each machine has an assigned algorithm, which describes the behaviour of that participant in the protocol to model. As suggested, each participant $U$ has a location denoted by $loc_U$ in a metric space, where $d$ is the distance-function of this metric space (i.e., there is a distance-unit and the classical requirements to measure distances). The distance is assumed to measure the time-of-flight of messages between two locations (i.e., as if messages were travelling uniformly at a speed of one distance-unit per time-unit). At this stage, the reader can refer to Fermat's principle [34] for the notion of time-of-flight.

The time-of-flight is further described by a global counter called *clock*. This *clock* is incremented at certain execution-points, as the communication model will explain below. We underline that the complexity of the machines is measured in the number of computational[16] steps and it is **not** linked to this notion of time-of-flight. Thus, we assume that all computation of (parts of) messages is instantaneous (in terms of the ticks of the *clock*). Only other actions, e.g., sending a message from one location to another, have a time-duration on the *clock*.

Also, there is a global system-recall called *history*. The tuples stored in this register are of the form

$$(\mathsf{message}, \mathsf{locationOfOrigin}, \mathsf{timeOfSending}, \mathsf{destination})$$

i.e., a message that has been sent, from some initiator-origin, departing at some time and being aimed at some participant.

***Communication.*** In the following, we assume that the network is asynchronous. We consider insecure and noisy channels. However, the adversary receives messages with no noise.[17] In addition to this, protocol messages which are not "time-critical" (as clearly explained later) can be assumed to be noiseless, or equivalently, that participants use a computations overhead for error correction. All channels employed in this model are timed, i.e., by the (units of) global counter *clock*. As aforementioned, we assume that all communication happens through a broadcast anonymous channel.

All machines have communication-related actions of three types: *send*, *standby* and *halt*. If a machine does a *halt* action, then its execution is terminated. Before halting, the machines write their output on the output tape. If a machine $M$ performs the action $send(m,P)$, this denotes that the message $m$ is aimed at a participant $P$. Namely, the message $m$ is written on the outgoing communication tape of the sending machine $M$ and the tuple

[16] We will still consider "time complexity", namely polynomial versus non-polynomial computational complexity, but it does not relate to the notion of time-of-flight that we refer to in this section.

[17] This is due to adversaries using a more elaborate equipment.

$(m, loc_M, clock\_value, P)$ is added to the *history*, where *clock_value* is the value of the *clock* register at the time of this sending by $M$. After some sendings or simply at some point, the machine will do a *standby* action: i.e., the machine waits for a reactivation. When all participants are in a "hanging" state (e.g., some in standby, some halted), the global counter *clock* is incremented by a unit and the participants standing by are reactivated.

Let *clock_value* be the current value of the global *clock* register. For each tuple in the global *history* of the form $(m, loc_M, time\_sent, dest)$, if $d(loc_P, loc_M) \leq (clock\_value - time\_sent)$ then the participant $P$ *can* read[18] the content $m$ of its incoming communication tape. However, an honest participant $P$ will not read $m$ if $dest \neq P$.

***Adversary.*** An adversary $\mathcal{A}$ is modelled by an ITM of the above kind, i.e., he is part of the system as described, he has a location, etc. Moreover, an adversary $\mathcal{A}$ has the following abilities: 1. reading messages for which he is not necessarily the intended recipient; 2. corrupting the channel between any two participants $P$ and $V$ (i.e., upon corruption, for an action $send(m, P)$ done by $V$, the system performs the action $send(m, \mathcal{A})$ instead, re-aiming the message $m$ to $\mathcal{A}$); 3. sending his own messages to different participants. An adversary is not able to: a) block already sent messages on their way to their intended destination; b) modify sent messages.

Instead of doing[19] *a*) and/or *b*) (i.e., blocking a message far along its course), $\mathcal{A}$ can change the destinator from $R$ to $\mathcal{A}$ and may send another message to $R$. We believe this does not decrease the capabilities compared to practice since adversaries can still carry out man-in-the-middle attacks.

Also, the adversary has no control over the global counter *clock*. This is normal, since the counter *clock* simply models time passing, as we know it. However, the adversary is the first to be activated after each increment of the *clock* (i.e., as he may, e.g., corrupt a channel before a new message is sent on it).

## B SKI Variants

Our $F$-scheme can be instantiated to produce different **SKI** protocols, some arguably more practical/secure than others. In the main body of this paper, we presented a version that is in-line with the existing literature in the field, i.e., one-bit responses and a set of values for challenges of small cardinality, e.g., 3. Irrespective of this alignment of ours with the state-of-the-art, we consider that the practicality of today's RFID/NFC cards goes beyond one-bit responses [38]. Moreover, pre-computation tables can be used.

As formalised above, to attain security, the idea behind such an $F$-scheme is that it should be a secret sharing scheme in which the response to the $t > 2$ challenges in round $i$ reveals the component $x_i$ of the secret, but the answers to only 2 of these challenges (e.g., one from the prover and another indirectly leaked by the verifier, e.g., within a non-narrow MiM attack) do not reveal $x_i$. Namely, we will consider two generic such response-functions in which the $i$th response ($1 \leq i \leq n$) is produced as follows:

$$\mathbf{F_{shamir}}(c_i, a_i, x_i) = x_i + (a_i)_1 \bar{c}_i + (a_i)_2 \bar{c}_i^{\,2} + \ldots + (a_i)_{t-1} \bar{c}_i^{\,t-1}$$

where $x_i \in GF(q)$, $q \geq 4$, $c_i \in \{1, \ldots, t\}$ is mapped to $\bar{c}_i \in GF(q)^*$ by an arbitrary injective mapping, $(a_i)_j \in GF(q), j \in \{1, \ldots, t-1\}$;

$$\mathbf{F_{xor}}(c_i, a_i, x_i) = x_i 1_{c_i = t} + (a_i)_1 1_{c_i \in \{t,1\}} + \ldots + (a_i)_{t-1} 1_{c_i \in \{t,t-1\}}$$

where $c_i \in \{1, \ldots, t\}$, $x_i \in GF(q)$, $q \geq 2$, $(a_i)_j \in GF(q)$, $j \in \{1, \ldots, t-1\}$, and $1_R$ is 1 if $R$ is true and 0 otherwise.

Note that the function $\mathbf{F_{xor}}$ has been invoked in the main body of this paper to define $\mathbf{SKI_{pro}}$ and $\mathbf{SKI_{lite}}$. We give two more variants of it, $\mathbf{SKI_{shamir}}$ and $\mathbf{SKI_4}$.

In our numerical studies, we actually look at three specific $F$-schemes dictated by the functions above, giving three specific **SKI** protocols as follows:

[18] This formalises the discussion in page 16 about broadcasting and reading messages when the intended recipients are on the correct spheres.

[19] If the adversary $\mathcal{A}$ could modify a flying message sent by $S$ to $R$ before $R$ could actually read it, this would implement a super-fast channel contradicting our gold principles. We could then design the following trivial (but unrealistic) distance-fraud. The adversary can send random responses before receiving the challenges, use this super-fast channel to get the verifier's challenges faster than communication allows, and modify his own responses accordingly "on-the-fly", when they have not reached the verifier yet. Clearly, any sent message could thuswise be used as a "carrier" to send messages faster than allowed by our gold principles.

- $\mathbf{SKI_{shamir}}$: defined by $\mathcal{L} = \mathcal{L}_{\mathsf{bit}}$, and the response-function $\mathbf{F_{shamir}}$ above, with $q = 4$, $t = 3$, $t' = 2$, i.e., $F(c_i, a_i, x_i) = x_i + (a_i)_1 \bar{c}_i + (a_i)_2 \bar{c}_i{}^2$, with $x_i, (a_i)_1, (a_i)_2 \in GF(4)$ and $\bar{c}_i \in GF(4)^*$;
- $\mathbf{SKI_{pro}}$: defined by $\mathcal{L} = \mathcal{L}_{\mathsf{bit}}$, and the response-function $\mathbf{F_{xor}}$ above, with $q = 2, t = 3, t' = 2$, i.e., $F(c_i, a_i, x_i) = (a_i)_{c_i}$ for $c_i \in \{1,2\}$ and $F(3, a_i, x_i) = x_i + (a_i)_1 + (a_i)_2$, with $(a_i)_1, (a_i)_2, x_i \in GF(2)$;
- $\mathbf{SKI_4}$: defined by $\mathcal{L} = \mathcal{L}_{\mathsf{bit}}$, and the response-function $\mathbf{F_{xor}}$ above, with $q = 2, t = 4, t' = 3$, i.e., $F(c_i, a_i, x_i) = (a_i)_{c_i}$ for $c_i \in \{1,2,3\}$ and $F(4, a_i, x_i) = x_i + (a_i)_1 + (a_i)_2 + (a_i)_3$, with $(a_i)_1, (a_i)_2, (a_i)_3, x_i \in GF(2)$;
- $\mathbf{SKI_{lite}}$: defined by a variant of response-function $\mathbf{F_{xor}}$ above (not depending on $x_i$), with $q = 2$, $t = t' = 2$, i.e., $F(c_i, a_i, x_i) = (a_i)_{c_i}$ for $c_i \in \{1,2\}$, with $(a_i)_1, (a_i)_2 \in GF(2)$. Since $x'$ is not used, $\mathcal{L}$ can be let empty.

In relation with the definitions of the $F$-schemes and protocols above, we prove the following lemma.

**Lemma 15.** *The $F$-schemes used in* $\mathbf{SKI_{shamir}}$*,* $\mathbf{SKI_{pro}}$ *and* $\mathbf{SKI_4}$ *are linear, pairwise uniform, $t$-leaking. The $F$-scheme used in* $\mathbf{SKI_{lite}}$ *is linear, pairwise uniform and not $t$-leaking.*

- ***Lemma 15.1:*** The $F$-scheme used in $\mathbf{SKI_{shamir}}$ is $\frac{15}{8}$-bounded.
- ***Lemma 15.2:*** The $F$-scheme used in $\mathbf{SKI_{pro}}$ is $\frac{9}{4}$-bounded.
- ***Lemma 15.3:*** The $F$-scheme used in $\mathbf{SKI_4}$ is 3-bounded.
- ***Lemma 15.4:*** The $F$-scheme used in $\mathbf{SKI_{lite}}$ is $\frac{3}{2}$-bounded.

Following Lem. 15 and Th. 14, it is clear that the probabilities $\alpha$ and $\beta$ to succeed respectively in distance-frauds and in MiMs, against the **SKI** protocols are:

| | $\mathbf{SKI_{shamir}}$ | $\mathbf{SKI_{pro}}$ | $\mathbf{SKI_4}$ | $\mathbf{SKI_{lite}}$ |
|---|---|---|---|---|
| $\alpha$: | $B(n,\tau,\frac{5}{8})$ | $B(n,\tau,\frac{3}{4})$ | $B(n,\tau,\frac{3}{4})$ | $B(n,\tau,\frac{3}{4})$ |
| $\beta$: | $B(n,\tau,\frac{1}{2})$ | $B(n,\tau,\frac{2}{3})$ | $B(n,\tau,\frac{5}{8})$ | $B(n,\tau,\frac{3}{4})$ |

*Proof.* The first three properties (i.e, linearity, pairwise uniformity, $t$-leaking property) follow easily from the respective definitions of the three functions.

For the property of $\sigma$-boundedness, we will carry the proof using the notation

$$P_j(x_i) := \Pr_{a_i}\left[\max_y \left(\#(F^{-1}_{a_i,x_i}(y))\right) = j\right]$$

for $F_{a_i,x_i} : c_i \mapsto F(c_i, a_i, x_i)$. We will compute the bound $\sigma$ as $\max_{x_i} \sum_{j=1}^{t} jP_j(x_i)$. We recall that $P_j(x_i) = 0$ for $j < \frac{t}{q}$.

We start by proving Lem. 15.1, i.e., the response-function $F$ that gives the $i$th response as $F(c_i, a_i, x_i) = x_i + (a_i)_1 \bar{c}_i + (a_i)_2 \bar{c}_i^2$, with $x_i, (a_i)_1, (a_i)_2 \in GF(4)$ and $\bar{c}_i \in GF(4)^*$ is the mapped of the challenge $c_i \in \{1, \ldots, t\}$.

We can show that:

$$\max_y \left(\#(F^{-1}_{a_i,x_i}(y))\right) = 1 \Leftrightarrow (a_i)_2 = 0 \text{ and } (a_i)_1 \neq 0$$

$$\max_y \left(\#(F^{-1}_{a_i,x_i}(y))\right) = 2 \Leftrightarrow (a_i)_2 \neq 0$$

$$\max_y \left(\#(F^{-1}_{a_i,x_i}(y))\right) = 3 \Leftrightarrow (a_i)_2 = (a_i)_1 = 0$$

So, for a component $x_i$ in the secret vector $x$ as per above, it holds that:

$$P_1(x_i) = \frac{3}{16}, \quad P_2(x_i) = \frac{3}{4}, \quad P_3(x_i) = \frac{1}{16}.$$

Thus, $\sigma = 1 \times \frac{3}{16} + 2 \times \frac{3}{4} + 3 \times \frac{1}{16} = \frac{15}{8}$. This ends the proof of Lem. 15.1.

We now proceed to proving Lem. 15.2, i.e., the response-function $F$ that gives the $i$th response as $F(c_i, a_i, x_i) = (a_i)_{c_i}$ for $c_i \in \{1,2\}$ and $F(3, a_i, x_i) = x_i + (a_i)_1 + (a_i)_2$, with $(a_i)_1, (a_i)_2, x_i \in GF(2)$.

Following a similar calculation as above, we have:

$$\max_y \left(\#(F^{-1}_{a_i,x_i}(y))\right) = 3 \Leftrightarrow (a_i)_1 = (a_i)_2 = x_i, \text{ thus } P_3(x_i) = \frac{1}{4}.$$

For $j < \frac{t}{q}, P_j(x_i) = 0$, so since $1 < \frac{3}{2}$ we have that $P_1(x_i) = 0$. So, $P_2(x_i) = 1 - P_3(x_i) = \frac{3}{4}$. Thus, $\sigma = (2 \times \frac{3}{4} + 3 \times \frac{1}{4}) = \frac{9}{4}$. This ends the proof of Lem. 15.2.

We now proceed to proving Lem. 15.3, i.e., the response-function $F$ that gives $F(c_i, a_i, x_i) = (a_i)_{c_i}$ for $c_i \in \{1,2,3\}$ and $F(4, a_i, x_i) = x_i + (a_i)_1 + (a_i)_2 + (a_i)_3$, with $(a_i)_1, (a_i)_2, (a_i)_3, x_i \in GF(2)$. For $j < \frac{t}{q}, P_j(x_i) = 0$, so since $1 < \frac{4}{2}$, $P_1(x_i) = 0$.

If $x_i = 0$ we have:

$$\max_y \left(\#(F^{-1}_{a_i,x_i}(y))\right) = 4 \Leftrightarrow (a_i)_1 = (a_i)_2 = (a_i)_3, \text{ thus } P_4(x_i) = \frac{1}{4}.$$

We have that $\max_y \left(\#(F^{-1}_{a_i,x_i}(y))\right) = 3$ is impossible, i.e., $P_3(x_i) = 0$. So, $P_2(x_i) = 1 - P_4(x_i) = \frac{3}{4}$. Finally, $\left(4 \times \frac{1}{4} + 2 \times \frac{3}{4}\right) = \frac{5}{2}$.

If $x_i = 1$, then $\max_y \left(\#(F^{-1}_{a_i,x_i}(y))\right) = 4$ or 2 are impossible, i.e, $P_4(x_i) = 0$. Thus, for $x_i = 1$ we have $\max_y \left(\#(F^{-1}_{a_i,x_i}(y))\right) = 3$. We conclude that $\sigma = \max\left\{\frac{5}{2}, 3\right\} = 3$. This ends the proof of Lem. 15.3.

The proof of Lem. 15.4 is along the same lines as in the above, especially as in Lem. 15.2. □

## C On Location-Based Cryptography

Position-based cryptography (PBC) [12] becomes possible through secure positioning (SP), which involves a set of verifiers ensuring that a given prover is indeed at some claimed position. In other words, in PBC a verifier within the network not only estimates the distance to another device but is also helped by, e.g., trusted base-stations that offer position-data for coordinate-triangulation in his final decisions. In SP, this assistance by, e.g., base-stations can happen repeatedly, to defend against malicious behaviour. This is not the case in DB, where the verifier is on his own, with his much simpler measurements at hand. However, distance-bounding protocols could potentially be used as building blocks for SP.

The model needed to achieve PBC bears similarities with the one to follow, yet distance-bounding is a weaker requirement than secure positioning. DB informally implies one prover proving to *one* verifier only that the former is close enough to the latter, using the time-of-flight of their exchanges. Thus, while the "geometry" needed for achieving distance-bounding is much simpler, the notion of time is of greater importance for distance-bounding.