Security of Distance-Bounding: A Survey

GILDAS AVOINE, INSA Rennes, IRISA, IUF, France
MUHAMMED ALI BINGÖL, TÜBİTAK BİLGEM, and Sabanci University, Turkey
IOANA BOUREANU, University of Surrey, United Kingdom
SRDJAN ČAPKUN, ETH Zurich, Switzerland
GERHARD HANCKE, City University of Hong Kong, Hong Kong SAR
SÜLEYMAN KARDAŞ, Batman University, Batman, Turkey
CHONG HEE KIM, Brightsight, The Netherlands
CÉDRIC LAURADOUX, INRIA, France
BENJAMIN MARTIN, Université catholique de Louvain, Louvain-la-Neuve, Belgium
JORGE MUNILLA and ALBERTO PEINADO, University of Málaga, Spain
KASPER BONNE RASMUSSEN, University of Oxford, United Kingdom
DAVE SINGELÉE, Katholieke Universiteit Leuven and IMEC, Belgium
ASLAN TCHAMKERTEN, Telecom ParisTech, Paris, France
ROLANDO TRUJILLO-RASUA, University of Luxembourg, Esch-sur-Alzette, Luxembourg
and Deakin University, Geelong, Australia
SERGE VAUDENAY, EPFL, Lausanne, Switzerland

Authors' addresses: G. Avoine, IRISA Rennes, Campus universitaire de Beaulieu, 263 Avenue du Général Leclerc, CS 74205, Building F, Office 401 (red floor), 35042 Rennes Cedex, France; email: gildas.avoine@irisa.fr; M. A. Bingöl, TUBİTAK BİLGEM Gebze Kocaeli, Turkey and Sabana University, Faculty of Engineering and Natural Sciences, Istanbul, Turkey; email: mabingol@sabanciuniv.edu; I. Boureanu, University of Surrey, Guildford, Surrey GU2 7XH, United Kingdom; email: i.boureanu@surrey.ac.uk; S. Čapkun, ETH Zurich, CNB F 102.2, Universitätstrasse 6, 8092 Zürich, Switzerland; email: srdjan.capkun@inf.ethz.ch; G. Hancke, Department of Computer Science, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon, Hong Kong; email: gp.hancke@cityu.edu.hk; S. KardaŞ, Batman University, Batman Üniversitesi Merkez Kampüsü, 72000 Merkez/Batman Merkez/Batman, Turkey; email: suleyman.kardas@batman.edu.tr; C. H. Kim, Brassersplein 2, 2612 CT Delft, The Netherlands; email: kim@brightsight.com; C. Lauradoux, INRIA Rhône-Alpes, 655 avenue de l'Europe, Montbonnot, 38334 ST ISMIER CEDEX, France; email: cedric.lauradoux@inria.fr; B. Martin, Université catholique de Louvain, Department of Computing Science and Engineering, Place Sainte Barbe, 2, B-1348 Louvain-la-Neuve, Belgium; J. Munilla and A. Peinado, E.T.S.I. Telecomunicación, Universidad de Málaga, Campus de Teatinos, 29071 Málaga, Spain; emails: {munilla, apeinado}@ic.uma.es; K. B. Rasmussen, Department of Computer Science, Wolfson Building, Parks Road, Oxford, OX1 3QD, United Kingdom; email: kasper.rasmussen@cs.ox.ac.uk; D. Singelée, Kasteelpark Arenberg 10, 3001 Heverlee, Belgium; email: Dave.Singelee@esat.kuleuven.be; A. Tchamkerten, Telecom ParisTech, 46 Rue Barrault, 75013 Paris, France; email: aslan.tchamkerten@telecom-paristech.fr; R. Trujillo-Rasua, School of Information Technology, Deakin University, Geelong, Australia; email: rolando.trujillo@deakin.edu.au; S. Vaudenay, EPFL IC LASEC, Station 14, CH-1015 Lausanne, Switzerland; email: serge.vaudenay@epfl.ch.

94

Distance-bounding protocols allow a verifier to both authenticate a prover and evaluate whether the latter is located in his vicinity. These protocols are of particular interest in contactless systems, e.g., electronic payment or access control systems, which are vulnerable to distance-based frauds. This survey analyzes and compares in a unified manner many existing distance-bounding protocols with respect to several key security and complexity features.

1 INTRODUCTION AND STATE OF THE ART

1.1 From Relay Attacks to Evolved Distance-Based Frauds

The basic concept of a relay attack was first described by Conway [25] in 1976, in a scenario referred to as the *Chess Grandmaster Problem*. In this scenario, any player could play against two Grandmasters by challenging both of them to a game of chess by post. The player would then simply forward the move received from one Grandmaster to the other, effectively making them playing against each other. This results in the player either winning one match, or earning a draw in both matches. Desmedt, Goutier, and Bengio [28] extended this concept to security protocols in 1987, with an attack on the Fiat-Shamir protocol [31, 32] they named *mafia fraud*. In general, a protocol is seen to be executed between a party making a claim, the prover, and a party verifying this claim, the verifier. Mafia fraud involves a malicious third party who aims to convince the verifier that he is the legitimate prover. To start, the third-party simply takes all the messages sent by the verifier and forwards these to the prover. As the messages are legitimate, the prover believes he is communicating with the legitimate verifier. The prover then generates a valid response that the third party forwards to the verifier. On receiving this response, the verifier is convinced that he is communicating with the legitimate prover and the attack succeeds. A variant of mafia fraud, denoted by *terrorist fraud*, is an attack in which the prover colludes with the adversary to deceive the verifier, and was subsequently proposed by Bengio et al. [10]. In practice, this involves a prover sharing protocol information, other than key material, with a third party in such a way that he allows this third party to convince the verifier that he is the legitimate prover without having to relay all the verifier's messages.

Even though mafia fraud could be classified as a special type of man-in-the-middle attack, there are fundamental differences between these attacks. In man-in-the-middle attacks, the third party actively modifies messages between the verifier and the prover, and in general the attack is made possible through a security vulnerability in the protocol. In other words, man-in-the-middle attacks can be mitigated with conventional security mechanisms. In mafia fraud, the third party is passive and simply relays messages. He does not need to perform any further logical attack on the messages or the protocol sequence and in fact the third party does not even need to know what he is relaying. The protocol and security mechanisms are irrelevant as the attacker just relays the entire message generated by the legitimate parties, regardless of their content, thereby

ensuring that both the verifier and the prover always receive a valid message. Conventional security mechanisms are therefore not an effective countermeasure.

Brands and Chaum early proposed the idea of using so-called distance-bounding protocols [17] to mitigate mafia fraud. In addition to mafia fraud, Brands and Chaum also considered the possibility of *distance fraud*. Distance fraud involves a fraudulent prover that wants to convince the verifier that he is closer than he really is. Most recently, a new fraud termed *distance hijacking* was proposed [26]. In this case, a fraudulent prover takes advantage of a protocol executed between an honest prover and the verifier. The fraudulent prover selectively uses parts of this protocol instance to convince the verifier that he is at a distance, at which some other honest prover resides, which differs from the actual distance of the dishonest prover to the verifier.

1.2 Practical Attacks

The frauds discussed above are of practical significance when considering real-world system security. For example, mafia fraud is especially relevant in access control and payment systems. An RFID door access reader might authenticate an access token by transmitting a challenge, e.g., a nonce, and then checking whether the cryptographic response, constructed with the token's key, is valid. In such a case, an attacker can present a proxy-token, a device under the attacker's control that emulates a token, to the door reader. At the same time, his accomplice has a proxy-reader, a reader under the attacker's control, which is used to communicate with a legitimate token. This can be done in a covert manner, e.g., holding the reader against the token holder's pocket while he is outside the premises. The attacker's proxy-token gets the challenge from the door reader and transmits it to the accomplice's proxy-reader. The latter sends the challenge to the legitimate token. The proxy-reader thus obtains the valid response, which is transmitted to the proxy-token and then sent to the door reader. The door reader is now convinced that the token it is communicating with is the legitimate token and opens the door. A practical mafia fraud of this nature was first demonstrated by Hancke [40], using a built-for-purpose proxy-token and relaying radio channel with an effective range of 50m, alongside a modified off-the-shelf reader for the purpose of proxy-reader. Francillon, Danev, and Čapkun have also practically demonstrated the feasibility of mafia fraud against remote keyless entry systems in vehicles [35].

Similarly, payment systems are also vulnerable to mafia fraud. An attacker could convince a customer to insert his payment card into a proxy-reader, perhaps to pay for a low-value item sold to the customer by the attacker. The attacker's accomplice, in the meantime, purchases a high-value item and inserts his proxy-card into the merchant's reader. The high-value transaction is then conducted, via the proxy devices, with the legitimate payment card. The proxy-reader only displays the low value amount for customer approval, who thinks that he is authorizing the transaction by entering his PIN on the proxy-reader. This PIN is transmitted to the accomplice and it is entered into the merchant's reader, which then verifies the PIN through the relay setup with the legitimate card. As a causality the customer ends up paying for the attacker's item. This attack scenario was implemented against the "Chip and Pin" card payment system in the United Kingdom by Drimer and Murdoch [29] and illustrates that mafia fraud can be a serious threat even when systems use strong cryptography and two-factor authentication. The implementation of near-field communication (NFC) in mobile phones has potentially decreased the complexity of implementing mafia fraud. An NFC-enabled mobile phone can act as a token and a reader, so it can either act as a proxy-token or proxy-reader, while offering multiple options with regards to communication channels for relaying messages. The potential use of NFC devices in mafia fraud is documented by Kfir and Wool [47], and a practical mafia fraud using NFC-enabled mobile phones has already been demonstrated by Francis et al. [37]. Some additional attack scenarios and a discussion on the practical implementation of mafia fraud can be found in References [36, 44].

Real-time location systems (RTLS) are increasingly used to track high-value assets and people. A RTLS relies on the fact that the physical relation between reference nodes, with fixed and known locations, and the target can be estimated. If these estimates are somehow modified by an attacker, then the overall localization process will be adversely affected and the location of the target could be misrepresented. Čapkun and Hubaux [21] have shown that in the case of trilateration, and the principle extends to multilateration, a target located outside a triangle of reference nodes cannot prove that it is inside the triangle without shortening the distance measured to at least one of the reference nodes. Similarly, a node located inside the triangle cannot prove it is at a different location without decreasing the measured distance to at least one of the nodes. This means that a fraudulent prover, wishing to misrepresent his own location, must perpetrate distance fraud against at least one reference node. In practice, distance fraud is relatively simple in certain RTLS systems. For example, if the distances are estimated using received signal strength, then an attacker could selectively attenuate or amplify his communication with a specific reference node. Some practical distance fraud strategies enabling a fraudulent prover to decrease the round-trip time of his responses are discussed in References [43] and [24].

Finally, relay attacks are particularly relevant in the field of digital rights management (DRM), although this issue is rarely discussed in the literature. For example, a provider may refuse to deliver a content to the customer if the latter is not in a clearly defined location, as stated for example in References [1, 19, 58].

1.3 Countermeasures to Relay Attacks

To counteract relay attacks, we need to look beyond the data exchanged and incorporate the physical context of the interaction between verifier and prover into the protocol. Desmedt was the first to introduce solutions of this type [27]; he proposed to sign the prover GPS coordinates. In a second proposal [11], the notion of timed message exchanges was introduced. Using precise timing, Beth and Desmedt managed to detect the little girl fraud via the delay introduced by the relay. Desmedt, alone at first and then later with Beth, was the first to remark that countering mafia fraud implies relying on physical properties (localization or timing) rather than only depending on the cryptographic parts. This observation yields to several propositions to measure this physical property. Among them, distance-bounding protocols were the most promising counter-measure. Distance-bounding protocols can be built on the Received Signal Strength (RSS) [8] by measuring the Angle-of-Arrival (AoA) [38], the noise level [23], the physical property of the communication channel [67], the ambient environment [39, 72], or the measure of the Time of Flight (ToF).

The RSS and the AoA methods are usually discarded due to implicit security flaws. Indeed, an adversary can by increasing its signal strength or building special antenna deceive these measurements [22]. Methods based on the noise level or on channel properties work in theory. However, they are not practical to implement.

ToF methods are more reliable and often used to evaluate the distance d between two parties by calculating $d = s_{pr} \cdot t_p$, where s_{pr} is the propagation speed of signals on the medium of the communication channel and t_p is the one-way propagation time between the transmitter and the receiver [42]. An attacker committing mafia fraud will unavoidably increase the time that the message takes to travel between the prover and verifier, even simply forwarding and transmitting messages increases the ToF. Measuring this time and checking for unexpected delay in a response is therefore recognized as a feasible method for detecting mafia fraud [11]. ToF distance estimation comprises both Time-of-arrival (ToA) or round-trip-time (RTT) approaches. ToA requires both a verifier and prover to share a synchronized, high-precision clock, and only the propagation time of a single message is measured. For example, the verifier sends a challenge *chall* to the prover and records the time t_0 it was sent. The prover records the time $t_0 + t_p$ the challenge was received

and responds with the authenticated message $\{t_0 + t_p, chall\}$. If both the prover and the verifier are trusted, then this protocol is effective in detecting mafia frauds. However, it is vulnerable to distance fraud as the prover can simply decrease the value of $t_0 + t_p$ in the response to appear closer. From a practical perspective, both the prover and the verifier might not have a synchronized precise clock, e.g., an RFID reader could have such a precise clock but an RFID tag not.

1.4 Distance-bounding Based on RTT

Both these issues can be addressed using an RTT distance-estimation approach. In RTT, the verifier measures the time t_m from the moment he has sent a challenge to the moment the response is received. The verifier is therefore completely in control of the measurement, and he is also the only entity that requires a precise clock. In this case the verifier can estimate the distance $d = c \cdot (t_m - t_d)/2$, where t_m is the round-trip-time, equal to $2 \cdot t_p + t_d$, and t_d is the time the prover takes to calculate the response. For example, the verifier sends a challenge *chall* at t_0, which the prover receives at $t_0 + t_p$. The prover sends a response back at $t_0 + t_p + t_d$, and this is received at $t_v = t_0 + 2 \cdot t_p + t_d$, allowing for the RTT to be calculated as $t_m = t_v - t_0$. The fraudulent prover can no longer directly influence the measurement, as is the case with ToA, but he could try to send his response earlier than he receives the challenge. To prevent this, a protocol must be designed in such a way that the response depends on the challenge, i.e., $r = f(chall)$, so that the prover has to wait for the challenge before responding. This response function f also determines the length of the processing time t_d, which must be minimal and deterministic, to achieve an accurate estimate. The response function must therefore be of minimal complexity and should be processable in a short and predictable time.

Distance-bounding protocols are closely linked to aspects of the physical communication channel, a side effect of requiring accurate timing measurements. The channel on which the challenges and responses are to be transmitted must therefore be chosen in such a way that it does not adversely affects the security of the protocol or the accuracy of the distance estimated. Conventional communication channels have been shown to be unsuitable for secure distance-bounding protocols, due to the possibility that an attacker could exploit the latency introduced in these channels by error-resistant measures, such as framing/integrity data and filters in transceivers [24, 43]. In practice, building a distance-bounding channel is a hard problem. Even if we only consider the distance estimation requirements, a timing measurement error of 1ns could result in a distance estimation error of approximately 30cm, and measuring the RTT to this level of accuracy is not feasible in systems often suggested to benefit from distance-bounding. Implementing suitable channels is still an open research question, although there are several proposals already described and practically demonstrated in the literature [29, 41, 62–64].

In 2006, Clulow et al. [24] proposed four principles for implementing a secure channel for timed challenge–response exchanges:

(1) Use a communication medium with a propagation speed as close as possible to the physical limit, i.e., speed of light.
(2) Use a communication format in which only a single symbol is transmitted as challenge of response.
(3) Minimize the length of this symbol or the time taken to decide the value of the symbol.
(4) Design the protocol such that it copes with errors during the challenge–response exchange.

These principles have historical significance, as this work was the first to look at the security implications of the underlying implementation of the exchange channel. However, there is a growing opinion that these principles, aiming for theoretical security, are not fully achievable in practice. As such it is perhaps better to consider the intentions behind the principles' definition, which

helps us understand potential security threats and evaluate the effectiveness of a channel used for distance-bounding, rather than considering these as hard conditions for secure distance-bounding. The first principle advises against the use of channels with a relatively low propagation speed as this would allow an attacker to use a faster channel to relay the communication and not be detected. For example, if distance-bounding is conducted across a sound channel the attacker can execute an undetectable relay attack using wired or radio communication. The second principle advises against sending multiple challenges and responses during a single timed exchange, and against the transmission of any additional information even for purposes of error detection or formatting, e.g., any parity bits, cyclic redundancy checks (CRC), headers/trailers, or start/stop bits. In both cases, it is shown that a dishonest prover could exploit such exchanges to correctly send a reply earlier than what is expected from a honest prover adhering to the channel rules. The nature of the attack depends on the format of the message but the general idea is that the dishonest prover can calculate and prepare the response before the entire challenge message is received, thus shortening the response time compared to a honest prover waiting for the entire message. The third principle advises that the decision should be made as quickly as possible. If the symbol modulation/encoding is such that the entire symbol must be received, then the symbol period must be minimized or the receiver should determine the value early on in the symbol period. This is meant to protect against early detect/late send relay attacks, where the attacker can take advantage of the duration between the start of the symbol and when the receiver actually determines its value. For example, when using non-return to zero coding the receiver usually samples the symbol after $t_s/2$, where t_s is the symbol period, which allows for the maximum tolerance to data clock differences between the sender and receiver. If the attacker can sample the symbol at $t_s/10$, then he has $4 \cdot t_s/10$ to relay its value and transmit it to the receiver. In such a case, there will be no detectable delay in the communication and distance-bounding would be ineffective. To minimize the amount of time available to the attacker, the receiver must make its decision as early as possible during the symbol period. The fourth principle, taking into account that principles two and three would not allow for conventional error detection/correction measures and reduces the receiver's tolerances for reliably decoding of data, advises that the protocol cannot expect that the exchange channel will have no communication errors and that this has to be taken into consideration elsewhere in the system.

1.5 Protocol Evolution

Distance-bounding protocols are based on the RTT of challenge–response messages and are essentially meant to detect any unexpected delay in the provers' responses inherently caused by the messages being relayed over a larger distance by a third party. To effectively achieve this goal, protocols must meet some simple requirements to obtain an accurate propagation time measurement, as explained in the previous section: The response and challenge must be single bits, the response must be dependent on the challenge and the time taken to calculate the response must be minimal and predictable. There are a number of protocols that aim to implement distance-bounding but do not adhere to these requirements, e.g., References [11, 58, 75]. However, they are not capable of providing accurate distance estimates because of the variation in the time taken to calculate the response, which makes them unsuitable for many use cases. For example, the time taken by a smart token to encrypt a message or perform a digital signature differs each time. If such a token usually takes 100ms to calculate a response and if there is even a 0.1% variation, then this results in a RTT variation of 0.1ms and hence a 30,000km distance estimate error. This article only considers protocols proposals adhering to the prescribed requirements for distance-bounding.

In a distance-bounding protocol, not all exchanged messages are subject to round-trip-time measurements. The protocol can be divided into three distinct phases: setup, exchange, and verification. During the *setup* phase, the verifier and the prover exchange some initial information and

Brands and Chaum
1993

SECTOR
2003
(mutual authentication)

Walter and Felten
2003
(one pass fast phase)

Rasmussen and Capkun
2011
(No demodulation)

Hancke and Kuhn
2005
(Registers)

Peris-Lopez, Hernandez-
Castro, Palomar,
Tapiador, and
Van der Lubbe
2010
(cryptographic puzzles)

Munilla and Peinado
2006, 2008
(third state and
tag detection)

Kardas, Kiraz, Bin-
gol, and Demirci
2011
(PUF)

Bussard and Bagga
2005
(terrorist fraud
resistance)

Avoine and
Tchamkerten
2009
(tree)

MAD
2007
(noise resilience)

Nikov and Vauclair
2008
(exchanges lightening)

Kim and Avoine
2009, 2011
(tag detection)

MUSE
2009
(multistate)

Reid, Neito, Tang,
and Senadji
2007
(computation
lightening)

Poulidor
2010
(graph)

Yum, Kim,
Hong, and Lee
2011
(mutual authentication)

Tu and Piramuthu
2007
(protocol structure)

Swiss-knife
2008
(third phase)

Avoine, Lau-
radoux, and Martin
2011
(secret sharing)

Hitomi
2009
(register computation)

Boureanu, Mitrokotsa,
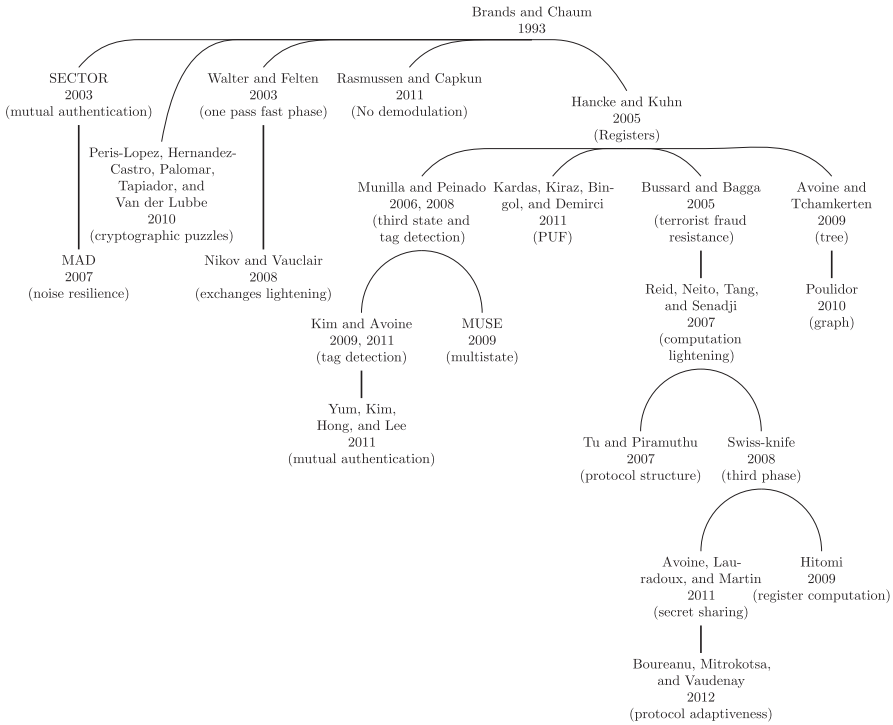and Vaudenay
2012
(protocol adaptiveness)

Fig. 1. Distance-bounding evolution.

determine the cryptographic material used during the rest of the protocol. During the *exchange* stage, the verifier measures the round-trip-time of the challenge–response pairs. The validity of the responses and the distance-bound is checked during the *verification* stage. The setup and verification phases are commonly referred to as the "slow" phases, while the exchange phase is referred to as the "fast" phase, due to the nature of the communication during these phases. The slow phase uses a conventional channel while the fast phase requires a special channel.

The first distance-bounding protocol was proposed by Brands and Chaum [17]. This protocol, based on Beth and Desmedt's [11] idea that RTT can detect mafia fraud, bounds the distance between the parties by measuring the RTT of single-bit challenges and responses. During the setup phase, the prover cryptographically commits to a random string that he will use to calculate the responses using an XOR operation. During the verification stage, the prover signs a message containing the challenges received and the responses sent. The protocol achieves an optimal $\left(\frac{1}{2}\right)^n$ resistance against both mafia fraud and distance fraud, where n is the number of challenge–response exchanges. This concept formed the basis for numerous protocols, whose evolution is represented in Figure 1.

There are four direct descendants of Brands and Chaum's protocol: References [20, 42, 59, 63], each of which improved Brands and Chaum in its own way. Peris-Lopez et al. [59] propose that cryptographic puzzles should be used to provide privacy in distance-bounding protocols. Rasmussen and Čapkun protocol [63] is based on XOR and a comparison function and has the benefit that the prover does not need to demodulate the signal to answer to the verifier's challenges. The MAD protocol proposed by Čapkun et al. [20] allows for mutual distance-bounding. This protocol was enhanced by the protocol of Singelée and Preneel [65], which added bit-error resilience to MAD by using error correcting codes. The Hancke and Kuhn's protocol [42], originally

designed to be used in the RFID environment and is thus optimized for execution time and minimal prover complexity, uses pre-computation, instead of a commitment step, during the setup phase in such a way that no additional messages need to be transmitted during the verification stage.

Hancke and Kuhn's protocol has two issues: It does not take terrorist attacks into account, and it achieves a sub-optimal performance-security tradeoff with respect to mafia and distance frauds of $(\frac{3}{4})^n$. Subsequently, numerous proposals based on the pre-computation method used by Hancke and Kuhn were proposed in an effort to improve its performance. Bussard and Bagga's protocol [18] and all its descendants introduce resistance to terrorist fraud. These protocols are based on Bussard and Bagga's idea that the prover long-term secret is incorporated into the pre-computed response options in such a way that if the prover reveals all the options his accomplice would also get the prover's key. This therefore discourages the prover to participate in a terrorist fraud but at the cost of a complex proof-of-knowledge operation. Its descendants aim to achieve the same functionality but with decreased computational complexity. Reid et al. [64] so improves the computational efficiency but the fraud resistance is $(\frac{3}{4})^n$ in comparison to Bussard and Bagga's $(\frac{1}{2})^n$. Tu and Piramuthu's protocol [71] proposes a protocol compounded by a succession of fast and slow phases. However, this protocol suffers from several vulnerabilities, discussed in References [50, 56], that reveal the secret to an eavesdropper during a legitimate protocol run. The *Swiss-knife* protocol [50] fixes the poor mafia fraud resistance problem by adding a third phase to Reid et al.'s protocol, and it also provides mutual authentication. In Reference [60], the authors claim that they found an attack on this protocol based on nonce repetitions and thus propose the Hitomi variation. However, if the assumption is made that nonces repeat, then Hitomi suffers, to a lesser extent, of a similar flaw. A further variation of the Swiss-knife protocol [5] explicitly introduces secret-sharing to counter terrorist fraud and studies the best settings in which to use it. Avoine and Tchamkerten's protocol [7] introduces binary trees to compute the prover answers during the exchange phase and succeeds in improving mafia fraud resilience to almost $(\frac{1}{2})^n$. Indeed, various graph structures can be used instead of a tree structure. The interest of cyclic and q-partite graphs has been demonstrated in Reference [69] and References [51, 52], respectively. Finally, Trujillo et al. [70] show that precomputation-based protocols can also deal with noise without sacrificing security.

Munilla and Peinado [54, 55] initiated a new branch of the Hancke-Kuhn pre-computation family. Their protocol communicates during the exchange phase using binary symbols, 0 and 1, and also an additional "nothing" state. MUSE [3] is a generalization of this idea relaxing the number of possible states. Kim and Avoine's protocols [48, 49] enhance the attack detection mechanism. Its descendant [76] uses the detection mechanism to also provide mutual authentication. Finally, Kardaş et al. [46] introduce PUFs in Hancke and Kuhn's protocol and claim the protocol now resists to terrorist fraud.

1.6 Provable Security

Most distance-bounding protocols have been analyzed without a formal approach. Instead, generic best-known attacks are usually adapted to the specific features of the protocol at hand, which has led to unsound analyzes and unfair comparisons. Examples are the protocols proposed in References [54], [71], and [76], whose flaws are explored in References [2], [56], and [4], respectively. The first comprehensive formalization for analyzing distance-bounding protocols was proposed by Avoine et al. [2]; this is not a provable security formalism, but it is a framework that can describe attack-scenarios in a unitary fashion and thus offer a systematic manner of computing upper-bounds on the probabilities of typical attacks in distance-bounding and its variants. This unified framework [2] defines the following important objects: the prover model (depending on the prover tampering-resistance, it can be either *black-box* or *white-box*), the prover's computing capabilities

(e.g., whether the prover can exploit latencies between the slow and fast phases), and the attacker's strategies (e.g., *pre-ask*, *post-ask*, and *early-reply*).

Recent efforts have been made on proving security for distance-bounding [14–16, 30, 34, 73]. However, this is still a very young field that needs to overcome three main, inter-dependent challenges: (i) the introduction of sound communication, network, and adversarial models that capture the notion of time-of-flight, (ii) the definition of clear and rigorous specifications of the classical frauds (i.e., formal definitions of these frauds that can be *proven* to hold or to be refuted within the model); and (iii) formal security proofs based on cryptographic assumptions. To illustrate, for instance, the difficulty of the third challenge, Reference [13] proved that many protocols fall short in having their security based on the pseudorandom function (PRF) assumption of some underlying primitive.

The first formalism in this direction was put forward by Dürholz et al. [30]. The authors formalize the impossibility of illegitimate yet sufficiently fast round-trip communications using the notion of *tainted sessions*; to encode timing-restrictions, tainted sessions only allow certain flows of communication. Then, a protocol is said to be secure if no adversary executing it with tainted sessions can violate its security properties. The model comprises a formalization of all the classical frauds and provides several (partial) security proofs for some protocols [30, 33]. This formal model is a step in the right direction towards provably secure distance-bounding.

Another line on provably secure distance-bounding, which builds on the model by Dürholz et al., is in Reference [34]. One addition therein is proposing a distance-bounding protocol that uses not one but two different secret keys for the slow and fast phases. This bypasses the aforementioned problem of using just the PRF-assumption to argue the security of (one-key) distance-bounding.

In References [14, 16], the authors provide a rather general model that captures the notion of concurrency (i.e., allowing adversaries to interact with many provers and verifiers, sometimes with the same keys). Their notions of distance and mafia frauds additionally capture the one of distance hijacking [26] and impersonation [30], respectively. Furthermore, their definition for terrorist-fraud is more general than the notion of terrorist fraud adopted in this manuscript: After the initial collusion, the possible threats to protect against may be stronger in References [14, 16] (e.g., MiM in concurrent settings). The authors also propose a set of distance-bounding schemes offering provable security against all forms of attacks within their model. A simplified version of Reference [14], where the elements of provable security are played down to best-attack scenarios, is available in Reference [15].

As depicted above, attention to provably secure distance-bounding is increasing. However, we underline that there is little consensus on which formalizations are appropriate by different metrics. This is evident, for instance, in the formalization of terrorist-fraud (TF) resistance, arguably due in part to its non-falsifiable nature that, in turn, renders it hard to (provably) attain in distance-bounding designs.

First, we underline a distinction between a commonplace view on TF resistance and the formal expressions of this. There is a widespread acceptance in the distance-bounding literature that TF resistance ought to repose on the reduction to the impossibility to protect against the "trivial vulnerability," whereby a prover gives away his secret-key to the adversary. That is, a protocol is often popularly understood to be TF resistant if the dishonest prover who helps the adversary authenticate fraudulently in one run leaks his secret key to the adversary. While this is a valid acceptation, the formal models above generally do not formalize precisely this commonplace view on terrorist-fraud resistance. Some approaches, e.g., References [12, 34], formalize the following statement: The protocol is resistant to TF or some generalizations thereof if, whenever the dishonest prover helps the adversary authenticate fraudulently, the adversary gains advantages in future authentication attempts in the absence of the illicit help. Other approaches [73] encode formally that the protocol

is sound (or terrorist-fraud resistant in a generalized sense) if the following holds: For all protocol-runs with a verifier, there exists an extractor who reconstructs the secret when he is given the knowledge of all participants that were close to the verifier in several successful executions.

Second, authors have changed and abridged their own formal definitions of these expressions of TF resistance. Dürholz et al. define *SimTF*, *StrongSimTF*, and *GameTF* terrorist-fraud resistance [30, 34]. Boureanu et al. put forward formalizations of terrorist-fraud resistance in Reference [15], as well as formalizations of generalizations of TF resistance in notions of collusion-fraud resistance [12, 14]. To this end, Vaudenay took collusion-fraud resistance further into a notion of soundness [73] akin to similar expressions in interactive proofs.

Third, one can argue that some of these formal definitions for TF resistance might yield a too-strong requirement, disproving security all-throughout (like the *SimTF* formulation of terrorist-fraud resistance in Reference [30]) or might be too general (like aforementioned collusion-fraud resistance in Reference [14] that suits the provable security of the SKI schemes [12]) or less realistic (like the *SimTF* formulation for terrorist-fraud resistance in Reference [30] in which the dishonest prover and the adversary are not allowed to communicate during the fast rounds). On the one hand, when we fix the model, we can nonetheless see that some of these definitions imply one another (*StrongSimTF* in Reference [34] implies *SimTF* in Reference [30], and soundness in Reference [73] implies collusion-fraud resistance in Reference [12], for certain parameters). On the other hand, even in one such fixed model, other definitions remain however incomparable (e.g., *GameTF* and *StrongSimTF* in Reference [34]), underlying further the unsettlement of formalizing terrorist-fraud resistance even within one and the same formalism.

Last but not least, formal comparisons between the session-based model in References [30, 34] and the model inspired by interactive proofs in References [12, 73] do not exist. In the absence of a formal proof aligning the two models and their security definitions, it appears that *SimTF* resistance in Reference [30] is equivalent to the notion of terrorist-fraud resistance in Reference [15] and that *GameTF* resistance in Reference [34] is equivalent to collusion-fraud resistance in Reference [14] (for some parameters).

Similar discussions apply—of course—to the formalizations of threats other than terrorist-fraud in the aforementioned formalism. As such, formal relations between the existing formal models for distance-bounding and their formal definitions of security is an avenue of future research.

Due to such differences between the formal models, we decided to carry out our analyses in the general framework by Avoine et al. [2]. This framework does not repose on such fine-grained formalizations of the distance-bounding threats,[1] but instead it formalizes classes of interactions between the provers and the attackers to better classify attack strategies towards an unitary approach to assessing the security/insecurity of distance-bounding.

1.7 Contributions

This article provides an in-depth security comparison of existing distance-bounding protocols. After the introduction of the notation and the methodology in Section 2, the next 12 sections present important published protocols. Each section presents in a unified way the considered protocol and its security analysis. Those who are not familiar with the presented protocols will be able to consult Appendix A that describes the protocols. Section 15 presents the comparison methodology and results. The article also includes Appendix B, which discusses about variants and extensions that can be applied to most of the considered protocols.

---

[1]For instance, the popular take on TF resistance by reduction to impossible protection against the "trivial vulnerability" is not attainable in the "white box model for TF" from Reference [2], while some of the formal expressions for TF resistance summarized above would be.

2 ANALYSIS METHODOLOGY AND NOTATIONS

This article analyzes 12 distance-bounding protocols using a unique methodology, based on the distance-bounding framework published in Reference [2]. Table 1 contains the unified notations used throughout the article. Each protocol description is divided into three steps, namely initialization, protocol, and final phase. Each protocol is depicted in a figure that is available in the supplementary online material, along with a table that summarizes the protocol parameters. Protocols consist of slow phases that are not time-constrained, and fast phases where the verifier measures the round-trip times of exchanged messages. The fast phases are identified with a left square bracket. Anything in the bracket is repeated n times, except if stated otherwise. Each protocol description is followed by a security analysis according to the template provided in Section 2.2. The properties and performance are analyzed according to Section 2.3 and 2.4.

2.1 Fraud Definitions

A distance-bounding protocol is a process whereby a party (known as *verifier*) is assured (i) of the identity of a second party (known as *prover*) and (ii) that the prover is located in his close vicinity (known as *neighborhood*). Four frauds against distance-bounding are usually considered, *impersonation*, *distance*, *mafia*, and *terrorist* frauds [2], which are introduced below.

Impersonation. An *impersonation fraud* is an attack where an adversary acting alone purports to be a legitimate prover.

Distance Fraud. A *distance fraud* is an attack where a dishonest prover purports to be in the neighborhood of the verifier. He cheats without help of other entities located in the neighborhood.

Mafia Fraud. A *mafia fraud* is an attack where an adversary defeats a distance-bounding protocol using a man-in-the-middle between the verifier and an honest prover located outside the neighborhood.

Terrorist Fraud. A *terrorist fraud* is an attack where an adversary defeats a distance-bounding protocol using a man-in-the-middle between the verifier and a dishonest prover located outside of the neighborhood under the following circumstances. The dishonest prover actively helps the adversary to maximize her current attack success probability but without giving her any advantage for future man-in-the-middle attacks. (In such attacks, the man-in-the-middle (MiM) would attempt to pass the distance-bounding protocol as a valid prover/tag that the MiM does not represent/possess.)

Note that protocols that are known to suffer from a key-recovery attack are not analyzed in this article. This includes Tu and Piramuthu's protocol [71] whose flaws are discussed in References [50, 56], Reid et al.'s protocol [64] broken in References [5, 53], and Hitomi whose vulnerabilities are described in Reference [66]. While Reference [9] points out a key recovery attack on Bussard and Bagga's protocol [18], this protocol is kept in this analysis, because the attacks presented in Reference [9] could be applied to other protocols and designers must be aware of their existence to avoid them. Note also that the length of the long-term secret keys of the parties, the length of the signatures (when appropriate), and the length of the nonces are assumed to be large enough, such that exhaustive search and replay attack are not relevant. Finally, the pseudo random functions used in the protocols are assumed to be without design flaws, i.e., no trapdoor pseudo random functions, like those discussed in Reference [13].

Another type of fraud, known as *distance hijacking*, has recently been introduced in Reference [26]. The fraud considers a dishonest prover who aims to convince a verifier that he is located within the verifier's neighborhood, abusing for that some other provers who are indeed in the verifier's neighborhood. For example, a dishonest prover can reach his goal by hijacking the fast phase of a distance-bounding protocol executed between an honest (closer) prover and the

Table 1. Notations

Prover and Verifier	
P, ID_P	Prover, Prover identity
V, ID_V	Verifier, Verifier identity
N_V, N_P	Nonces sent by verifier and prover, respectively.
Rounds	
n	Number of rounds in the fast phase
i	Index of the current round
Secrets	
K	Long-term secret key shared by prover and verifier
K_e, K_d	Public/Private keys for Encryption/Decryption
K_s, K_v	Private/Public keys for Signature/Verification
Time	
Δt_i	RTT measured during round i
t_{max}	Threshold on the round-trip time (typically, there is a round failure if $\Delta t_i > t_{max}$)
Challenges and Responses	
c_i	Challenge sent by the verifier in round i
c_i'	Challenge received by the prover in round i
r_i	Response sent by the prover in round i
r_i'	Response received by the verifier in round i
Registers	
R^0, R^1	Main registers
Z^0, Z^1,...	Additional registers, when needed.
H	Crypto function output, usually viewed as a register, e.g., $H = h(N_V, N_P)$
Sizes	
σ	Size of the signature, commitment, or MAC (in the slow phase)
ι_P, ι_V, ι	Size of ID_P and ID_V. If $|\mathrm{ID}_P|=|\mathrm{ID}_V|$, then the value is denoted ι (bits)
κ	Size of K (bits)
δ_P, δ_V, δ	Size of the nonces N_V and N_P. If $|N_V| = |N_P|$, then the value is denoted δ (bits)
Errors	
e_X	Number of errors of type X, e.g., e_C, e_R, e_T
e_{max}	Threshold on the number of errors
Functions	
$d_{\mathcal{H}}(.,.)$	Hamming distance
$\mathcal{H}(.)$	Hamming weight
$\mathrm{Sign}_{K_s}(.)$	Signature function with private key K_s
$\mathrm{Verif}_{K_v}(.)$	Public-key signature verification function with public key K_v
$\mathrm{Commit}(.)$	Commitment function
$\mathrm{Open}(.)$	Open commitment function
$h(.)$	Cryptographic hash function
$h_K(.)$	Cryptographic hash function keyed with the secret key K
$\mathrm{MAC}_K(.)$	Message authentication code keyed with the secret key K
$f_K(.)$	Pseudorandom function keyed with the secret key K
$E_K(.)$	Encryption function keyed with the secret key K
$D_K(.)$	Decryption function keyed with the secret key K

(Continued)

Table 1. Continued

Misc	
$\mathbb{E}(.)$	Mathematical expectation
$\in_R$	Randomly and uniformly picked in...
$\in_R \{0, 1\}^x$	Randomly and uniformly picked in the set $\{0, 1\}^x$, typically $x = \delta$
$||$	Concatenation of words (possibly 1-bit words)
p	Number of runs of the cryptographic function, in the analyzes
$\mathfrak{p}, \mathfrak{q}$	Prime numbers.
p_X	Probability of event X
w	Hamming weight, e.g., $w = \mathcal{H}(x)$

Table 2. Attack Scenarios

Fraud	Prover Model	P's Computing Capability	Adversary Strategy	Success Probability
Impers.	(1)	(1)	(4)	$\Pr_{\text{Imp}}$
Mafia	(1)	(1)	*pre-ask*	$\Pr_{\text{MF|pre}}$
			post-ask	$\Pr_{\text{MF|post}}$
Distance	*black-box*	(2)	*pre-ask & early-reply*	$\Pr_{\text{DF|BB|pre&early}}$
			post-ask & early reply	$\Pr_{\text{DF|BB|post&early}}$
	white-box	*single run*	*early-reply*	$\Pr_{\text{DF|WB(1)|early}}$
		multiple run		$\Pr_{\text{DF|WB(p)|early}}$
Terrorist	*black-box*	(2)	(3)	(3)
	white-box	*single run*	*early-provide*	$\Pr_{\text{TF|WB(1)}}$
		multiple run		$\Pr_{\text{TF|WB(p)}}$

(1) The computing capability is not relevant with an honest prover.
(2) The computing capability is not relevant in the black box model.
(3) This case is equivalent to the mafia fraud case.
(4) No strategy is defined in Reference [2] for impersonation.

verifier. Conceptually, distance hijacking can be placed between distance fraud and terrorist fraud. Unlike terrorist fraud, where a dishonest prover colludes with another attacker, distance hijacking considers a dishonest prover who interacts with (abuses) other honest provers. Unlike distance fraud that only involves a dishonest prover and a verifier, distance hijacking also involves other honest provers. These seemingly subtle differences have significant consequences, e.g., the countermeasures proposed against terrorist fraud strictly depend on the fact that the dishonest prover needs to share data with another attacker. In fact, the protocols BC [17], MAD [20], and RC [63] are not resistant against hijacking fraud according to Reference [26]. The version of RC presented in Section 10 comes from Reference [61]. This is a version that has been modified to be resilient to distance hijacking. Cremers et al. provide in Reference [26] a clear analysis of existing protocols that resist to the hijacking fraud. Vaudenay analyzes additional protocols in Reference [74]. We consequently refer the reader to these articles to get more information about distance hijacking.

2.2 Security

The analyses usually performed in distance-bounding do not provide a security proof but state the resistance of a protocol given a clearly defined scenario, which includes the type of fraud but also the adversary's capabilities and strategies, described below and summarized in Table 2.

Prover Model. Depending on the tamper-resistance of the prover, two models are defined: *black-box* and *white-box*. In the black-box model, the prover can neither observe nor tamper with the execution of the algorithm. In the white-box model, the prover has full access to the implementation of the algorithm and a complete control over the execution environment, as detailed in Reference [2].

Prover Computing Capabilities. The prover computing capabilities may affect the security of the protocol when considering distance and terrorist fraud in the white box model, given that the prover is also the attacker in such frauds. For example, in HK protocol, the prover may exploit a latency between the slow and fast phases to generate registers with a low Hamming distance [2].

Adversary Strategies. The framework [2] points out that three relevant adversary's strategies should be considered when analyzing a distance-bounding protocol: *pre-ask*, *post-ask*, and *early-reply* strategies. In the pre-ask strategy, the adversary relays the first slow phase between the verifier and the prover and then executes the fast phase with the prover before the verifier starts it. In the post-ask strategy, the adversary relays the first slow phase and then executes the fast phase with the verifier without involving the prover. The adversary then queries the prover with the correct challenges received during the fast phase. This strategy is meaningful when the protocol is completed with a second slow phase used to check that the challenges received by the prover are correct. In the early-reply strategy, the adversary anticipates the replies to make them arrive on time, which is particularly relevant with distance fraud. No strategy for the terrorist fraud is defined in Reference [2]. We introduce here the *early-provide* strategy: In this strategy, the adversary located inside the neighborhood, first relays the slow phase to the prover. The latter then provides to the adversary some information to help him to improve her success probability during the fast phase with the verifier. Finally, the adversary relays the final slow phase, if any.

Remark 1 (Circle Analysis). A prover located outside the neighborhood of the verifier but not too far may receive some challenges while the protocol is still running. When the rounds of the fast phase are independent, this late information is useless. However, the adversary may use this information to increase her success probability when the rounds are not independent. Consequently, when analyzing the resistance of a protocol against distance and terrorist frauds the area the prover is located should be considered. However, in all the analyzed protocols, either this scenario is not relevant due to the round independency or the calculation of the success probability is still an open problem.

Remark 2 (Multiple-Execution). The framework also points out that some information could leak when the protocol is executed several times. Typically, this case occurs when the prover and the verifier generate two registers without involving randomness from the prover. None of the protocols analyzed in the article are known to suffer from this weakness. Consequently, it is not explicitly addressed in the analysis.

2.3 Properties

The protocol properties considered in the article are described below and summarized in Table 3. Note that the type of data exchanged during the fast phase is usually binary. This is the case for all protocols considered in this analysis, except the one discussed in Section 14.

Adaptiveness. Indicates whether the protocol provides an adjustable tradeoff between resistance to mafia and distance frauds.

Mutual Authentication. Indicates whether the protocol provides *mutual* authentication. Note that, mutual authentication does not imply mutual distance-bounding: While the identity proof is bilateral in that case, the distance proof is unilateral in all the analyzed protocols.

Table 3. Properties

Property	Value
Adaptiveness	Yes/No
Mutual auth.	Yes/No/Opt.
Second slow phase	Yes/No
Round independence	Yes/No

Table 4. Performance

Performance	Value
Cryptographic blocks	Type
Exchanged bits (slow)	bits
Exchanged bits (fast)	bits
Memory consumption	bits

Second Slow Phase. Indicates whether there is a *second slow phase* in the protocol after the fast phase.

Independence of the Rounds. Indicates whether each expected response during the fast phase depends on the *current challenge only*.

2.4 Performance

The protocol performance is described below and summarized in Table 4.

Cryptographic Primitives. Type of cryptographic primitives needed to be implemented on the prover side: cryptographically secure pseudo-random number generator, hash, encryption, commitment, and signature. Hash functions and ciphers are actually aggregated into a single category that is denoted *symmetric primitive*.

Exchanged Bits (Slow Phase). Number of exchanged bits during the slow phase(s).

Exchanged Bits (Fast Phase). Number of exchanged bits during the fast phase.

Memory Consumption. Amount of memory that is needed during the entire fast phase by the prover.

3 BRANDS AND CHAUM'S PROTOCOL (1993)

In 1993, Brands and Chaum designed several distance-bounding protocols [17]. This analysis focuses on their protocol (Algorithm 1 in Appendix A) that mitigates both mafia and distance fraud.

Assuming that the signature scheme is secure, impersonating the prover can only be done by sending a randomly selected correct signature. Such a naive attack has success probability $\Pr_{\text{Imp}} = (1/2)^\ell$. However, while nonce-based replay attacks are not addressed in this article (Section 2.1), a challenge-based replay attack should be considered. Indeed, if the challenges sent by the verifier are used twice, then the adversary can reuse the same m_i's and thus obtains the correct Open(Commit) and $\text{Sign}_{K_s}(c_1||r_1||\ldots||c_n||r_n)$. After eavesdropping one execution before the attack, the success probability becomes $\Pr_{\text{Imp}} = (1/2)^n$.

Pre-ask strategy (mafia fraud). The adversary gets the commitment and queries the prover with random bits (c_i) during the fast phase. The adversary then receives the responses (r_i) and the final signature. With this information, the adversary computes $m_i = c_i \oplus r_i$, then sends the valid responses to the verifier during the fast phase, and, finally, the commitment and the signature during the second slow phase. However, the signature received from the prover is not valid for this protocol run, except if the challenges sent by the adversary to the prover and the challenges sent by the verifier to the adversary are the same. The success probability of this strategy is the probability of guessing the challenges correctly: $\Pr_{\text{MF}|\text{pre}} = (\frac{1}{2})^n$ [2].

Post-ask strategy (mafia fraud). The adversary must predict the correct responses to be sent to the verifier during the fast phase without any assistance. We thus have $\Pr_{\text{MF}|\text{post}} = (\frac{1}{2})^n$.

Early-reply strategy with one run (distance fraud). Given that the adversary must predict the current challenge correctly beforehand, her success probability is provided by the formula $\mathrm{Pr}_{\mathrm{DF|WB(1)|early}} = (\frac{1}{2})^n$ [2].

Early-reply strategy with p runs (distance fraud). No cryptographic function is used to compute registers, contrary to Hancke and Kuhn's approach. This fact trivially yields $\mathrm{Pr}_{\mathrm{DF|WB(p)|early}} = \mathrm{Pr}_{\mathrm{DF|WB(1)|early}}$.

Pre-ask combined with early-reply strategy (distance fraud). With the pre-ask strategy, the adversary learns all the possible answers. However, she does not know the challenges, so when she sends her answers in advance, two cases occur: (a) the verifier uses the same challenge as she did with the verifier. Therefore she always succeeds, (b) the verifier picks another challenge and she has sent an incorrect answer to the verifier. Hence, the success probability of this strategy is $\mathrm{Pr}_{\mathrm{DF|BB|pre\&early}} = (\frac{1}{2})^n$.

Post-ask combined with early-reply strategy (distance fraud). Given that the adversary must commit during the first slow phase, she cannot just answer randomly during the fast phase and she will therefore need to predict the responses expected by the verifier. Hence we have $\mathrm{Pr}_{\mathrm{DF|BB|post\&early}} = (\frac{1}{2})^n$.

This protocol is not designed to resist to terrorist fraud in the white box model. Indeed, the prover without revealing his secret K_s is able to provide to his accomplice the commitment and the signature, which are required to succeed. Consequently, $\mathrm{Pr}_{\mathrm{TF|WB}} = 1$ [50].

4 ČAPKUN, BUTTYÁN, AND HUBAUX'S PROTOCOL (2003)

In 2003, Čapkun, Buttyán, and Hubaux introduced MAD [20], a protocol that works quite similarly to the BC protocol [17] but provides mutual authentication. Although denoted by P and V, the two parties act as both prover and verifier during the execution of the protocol (Algorithm 2 in Appendix A). The notations used in Reference [20] are kept in the description below.

The basic way to impersonate the prover is to generate the random numbers r and r' and to complete the first slow phase and the fast phase. The adversary must then guess the output of the MAC function in the second slow phase. The probability of a correct guess is $\mathrm{Pr}_{\mathrm{Imp}} = (\frac{1}{2})^\sigma$.

Without loss of generality, we assume that the adversary seeks to impersonate P against V.

Pre-ask strategy (mafia fraud). To succeed in the mafia fraud, the output of the MAC function in the second slow phase needs to be valid. Since the adversary cannot compute this value, she needs to ensure that P sends the correct output of the MAC function to V. This will only be the case if the adversary has guessed the values s_i correctly during the pre-ask stage. Hence, $\mathrm{Pr}_{\mathrm{MF|pre}} = (\frac{1}{2})^n$ [20].

Post-ask strategy (mafia fraud). Similarly, the adversary needs to ensure that P sends the correct output of MAC_K. This will only be the case if she guessed all correct r_i values in advance: $\mathrm{Pr}_{\mathrm{MF|post}} = (\frac{1}{2})^n$ [20].

Without loss of generality, we assume that P wants to perform a distance fraud (the distance fraud success probability of V is equal to the one of P).

Early-reply strategy with one run (distance fraud). The responses α_i are computed by XORing the values of the responses r_i, which are completely controlled by the adversary, and the challenges β_i. The latter are uniformly distributed, and the values α_i inherit the same statistical distribution. So even if the adversary fully controls her hardware, the best strategy is to guess the challenges β_i in advance. We have thus: $\mathrm{Pr}_{\mathrm{DF|WB(1)|early}} = (\frac{1}{2})^n$.

Early-reply strategy with p runs (distance fraud). Similarly to Algorithm 1, no cryptographic function is used to compute registers, and so $\mathrm{Pr}_{\mathrm{DF|WB(p)|early}} = \mathrm{Pr}_{\mathrm{DF|WB(1)|early}}$.

We assume that the fraudulent party that performs the distance fraud is P.

Pre-ask combined with early-reply strategy (distance fraud). The best strategy consists in guessing the n challenges β_i. By querying itself in advance, the prover learns the values r_i and computes the responses α_i. The adversary uses these responses in the early-reply strategy. They are correct when the values β_i were guessed correctly, and, consequently, the MAC computed by the prover in the second slow phase is correct as well. If one of the challenges is guessed incorrectly, then the prover will compute incorrect values s_i, and $\text{MAC}_K(.)$ will be wrong. The distance fraud success probability is $\text{Pr}_{\text{DF|BB|pre\&early}} = \left(\frac{1}{2}\right)^n$.

Post-ask combined with early-reply strategy (distance fraud). The adversary has no information on the bits r_i. The best strategy is to send n random responses α_i. In each round, the adversary has a 1/2 probability of being successful. This occurs when both r_i and β_i are guessed correctly or when both guesses were wrong. When one of these values is correct and the other one is incorrect, the response of the adversary will be wrong. As a result, the distance fraud success probability is $\text{Pr}_{\text{DF|BB|post\&early}} = \left(\frac{1}{2}\right)^n$.

This protocol is not designed to resist to terrorist fraud in the white box model. Indeed, the prover without revealing her secret K, gives her accomplice the output of the commitment, and the values α_i and r' or β_i and s'. After the fast phase, the accomplice gives the observed values s_i or r_i to the prover, who can then computes the MAC. This output is then sent back to the accomplice, who finally forwards it to the verifier. Hence, $\text{Pr}_{\text{TF|WB}} = 1$ [50].

5 HANCKE AND KUHN'S PROTOCOL (2005)

In 2005, Hancke and Kuhn published the first distance-bounding protocol [42] (Algorithm 3 in Appendix A) clearly dedicated to RFID. The protocol relies on the original ideas of Desmedt et al. [10, 28] but is different from Brands and Chaum's work [17] in the sense that Hancke and Kuhn's protocol does not have any final signature after the fast phase.

The common attack consists in guessing all the answers during the fast phase: $\text{Pr}_{\text{Imp}} = \left(\frac{1}{2}\right)^n$.

Pre-ask strategy (mafia fraud). We have $\text{Pr}_{\text{MF|pre}} = \left(\frac{3}{4}\right)^n$ [42].

Post-ask strategy (mafia fraud). This protocol does not contain any second slow phase, and the first slow phase consists of nonce exchanges only. As per Section 2, we have $\text{Pr}_{\text{MF|post}} = \text{Pr}_{\text{Imp}}$.

Early-reply strategy with one run (distance fraud). We have: $\text{Pr}_{\text{DF|WB(1)|early}} = \left(\frac{3}{4}\right)^n$ [69].

Early-reply strategy with p runs (distance fraud). The formula expressing the attack success probability for this strategy was originally presented in Reference [2], but it contained a typing error. The correct formula is

$$\text{Pr}_{\text{DF|WB(p)|early}} = \frac{1}{2^{pn}} \cdot \left(\sum_{i=0}^{i=n-1} \left(\frac{1}{2}\right)^i \cdot \left[\left(\sum_{j=i}^{j=n} \binom{n}{j} \right)^p - \left(\sum_{j=i+1}^{j=n} \binom{n}{j} \right)^p \right] + \left(\frac{1}{2}\right)^n \right).$$

Pre-ask combined with early-reply strategy (distance fraud). With the pre-ask strategy, the adversary learns half of the possible responses. However, she does not know the challenge, so when she sends her responses in advance, two situations can occur: (1) The verifier asks her the same challenge that she asked the prover and therefore her response is correct and (2) the verifier sends a different challenge in which case she succeeds if the two possible responses were the same, i.e., her response is the same as the alternative response and fails if the possible responses are different. Hence, the distance fraud success probability is $\text{Pr}_{\text{DF|BB|pre\&early}} = \left(\frac{3}{4}\right)^n$.

Post-ask combined with early-reply strategy (distance fraud). This protocol does not contain any second slow phase, and the first slow phase consists of nonce exchanges only. As per Section 2, we have $\text{Pr}_{\text{DF|BB|post\&early}} = \text{Pr}_{\text{Imp}}$.

This protocol is not designed to resist to terrorist fraud in the white box model. Indeed, the prover is able to provide to his accomplice the two registers required to successfully execute the protocol, without revealing his secret K: $\text{Pr}_{\text{TF|WB}} = 1$ [50].

6 BUSSARD AND BAGGA'S PROTOCOL (2005)

Bussard and Bagga published the DBPK-Log protocol (Algorithm 4 in Appendix A), which is a distance-bounding protocol based on a proof of knowledge and a commitment scheme [18].

In References [18], the authors described a statistical key recovery attack. They established the success probability of this attack: $\text{Pr}_{\text{Imp}} = (1/2)^{-4m'}$, where m' is a security parameter.

Pre-ask strategy (mafia fraud). The adversary must pass the final slow phase to defeat the protocol. Forging the Open function is definitely not the best option. Instead, the adversary should try to send the correct challenges to the prover during the pre-ask attack and then relay the final slow phases. Her success probability with such a strategy is $\text{Pr}_{\text{MF|pre}} = (1/2)^n$.

Post-ask strategy (mafia fraud). Due to the presence of a complex second slow phase, the post-ask strategy is as good as the pre-ask strategy against DBPK-Log, which yields the success probability: $\text{Pr}_{\text{MF|post}} = (1/2)^n$.

Early-reply strategy with one run (distance fraud). The adversary can search for a random R^0 that minimizes the Hamming distance between R^0 and R^1. Denoting a as the Hamming distance between R^0 and R^1 ($a = d_{\mathcal{H}}(R^0, R^1)$), we have $\text{Pr}_{\text{DF|WB(1)|early}} = (1/2)^a$. This points out that the analysis provided in Reference [18] under-evaluates the success probability of the adversary, because the white box model is not considered.

Example 6.1. Let us consider the safe prime $p = 59$ ($q = 29$ and $n = 6$) and $x = 27$. If $R^0 = 32$, then $d_{\mathcal{H}}(R^0, R^1) = 1$ and so the success probability is $(1/2)$.

Early-reply strategy with p runs (distance fraud). Running the pseudo-random generator for choosing R^0 once, or several times, has no impact in the protocol security: The malicious prover can choose an appropriate value for R^0 to maximize its success probability in distance fraud. Hence: $\text{Pr}_{\text{DF|WB(p)|early}} = \text{Pr}_{\text{DF|WB(1)|early}}$.

Pre-ask combined with early-reply strategy (distance fraud). The adversary must succeed in the second slow phase and has the same success probability as mafia fraud. We then have $\text{Pr}_{\text{DF|BB|pre\&early}} = (1/2)^n$.

Post-ask combined with early-reply strategy (distance fraud). The adversary must succeed in the fast phase without the knowledge of the challenge. Then, the prover is queried by the adversary to gain information for the final slow phase. Similarly to mafia fraud, the success probability is $\text{Pr}_{\text{DF|BB|post\&early}} = (1/2)^n$.

Early-provide strategy with one run (terrorist fraud). This protocol is designed to resist to terrorist fraud in the white box model. Indeed, the prover cannot reveal R^0 and R^1 without exposing the key, making him able to provide only R^0 or R^1 to the external adversary. Note that the prover cannot try to optimize the Hamming distance between R^0 and R^1 as in distance fraud.

Remark 3. The probability of terrorist fraud calculated in Reference [18] is lower than the one provided here. Indeed, the authors considers that the final slow phase cannot be relayed by the adversary.

Early-provide strategy with p runs (terrorist fraud). Similarly to distance fraud strategy with early provide one run, we have $\text{Pr}_{\text{TF|WB(p)}} = \text{Pr}_{\text{TF|WB(1)}}$.

A Dedicated Distance Fraud and Terrorist Fraud. We describe here a distance fraud attack from Reference [9]. The key idea is that a malicious prover could select $R^0 \approx \frac{x}{2} \bmod (p-1)$. That is,

if x is even, he takes $R^0 = \frac{x}{2}$ and gets $R^1 = R^0$. Otherwise, he takes $R^0 = \frac{x \pm 1}{2}$ and get $R^1 = R^0 \pm 1$ so that R^0 and R^1 differ in their least significant bit only. He can then run the protocol normally. We note that $R_i^0 = R_i^1$ except for one single round. So the answers to the received challenges do not depend on it, except in one round. By sending the answer before the challenge arrives, the malicious prover can succeed in an early-reply strategy to run a distance fraud with a success probability larger than $\frac{1}{2}$.

Reference [9] also describes a terrorist fraud attack. The idea of the attack is that the malicious prover starts the protocol but does not give the commit values. Instead, he computes z and discloses it to the adversary through an early-provide strategy. The adversary will commit to random bits for R_i^0 and R_i^1 except for round $i = 1$. Then, he guesses the value c_1 and commit to a random bit for $R_1^{c_1}$. Finally, the commit value for $R_1^{1-c_1}$ is adjusted so that the equation $z = \prod_{i=1}^{n}(C_i(R^0)C_i(R^1))^{2^{i-1}}$ mod p holds. Clearly, the adversary can answer all challenges (if his guess for c_1 is correct), since he knows the bits he committed to. Next, he can get the help of the malicious prover to run the PK protocol through the slow phase. Due to the zero-knowledge property of the PK protocol, this leaks no information about x. This attack works with probability $\frac{1}{2}$ (due to the guess of c_1).

Finally, Reference [9] proposes some man-in-the-middle attacks against variants of this protocol that are not using public-key cryptography, i.e., where PK is not used and x is shared.

7 MUNILLA AND PEINADO'S PROTOCOL (2006)

Munilla and Peinado introduced in References [54, 57] the concept of void challenges as a tool to improve distance-bounding protocols. These void challenges can also be used to decrease the mafia fraud success probability when applied to Hancke and Kuhn's protocol [55], which is the case analyzed in this section. Thus, for this protocol (Algorithm 5 in Appendix A), the challenges can be 0, 1, or void, where a void challenge means that no challenge is sent. Void challenges are used to detect a mafia fraud attack using the pre-ask strategy.

The adversary must guess the responses to the non-void challenges and the signature. Hence:

$$\text{Pr}_{\text{Imp}} = \left(1 - \frac{p_f}{2}\right)^n \cdot \left(\frac{1}{2}\right)^{3n}.$$

Pre-ask strategy (mafia fraud). The calculation of the success probability of the pre-ask strategy is:

$$\text{Pr}_{\text{MF|pre}} = \begin{cases} \left(1 - p_f\right)^n & \text{if } p_f < 4/7 \\ \left(p_f \cdot \frac{3}{4}\right)^n & \text{if } p_f \geq 4/7 \end{cases}. \qquad [2]$$

Note that $\text{Pr}_{\text{MF|pre}}$ is calculated in Reference [2] and provided above is an approximation of the real value. Indeed, once the adversary is detected by the device, she does not receive any useful information any more. However, she can still guess the correct answers to be sent to the verifier. Given that being detected by the device forces the adversary to guess the final signature, this case is nevertheless negligible (Section 2.1).

Post-ask strategy (mafia fraud). The adversary must predict the correct responses to the non-void challenges. We so have

$$\text{Pr}_{\text{MF|post}} = \left(1 - \frac{p_f}{2}\right)^n. \qquad [2]$$

Remark 4 (Best strategy). The best strategy is post-ask when $p_f < 4/5$ and pre-ask when $p_f > 4/5$.

Early-reply strategy with one run (distance fraud). When the challenge is not void, the adversary can correctly respond to the verifier with probability 1 if $R_i^0 = R_i^1$ and with probability 1/2 if

$R_i^0 \neq R_i^1$. Consequently,

$$\mathrm{Pr}_{\mathrm{DF|WB(1)|early}} = \left(1 - \frac{p_f}{4}\right)^n . \qquad [2]$$

Early-reply strategy with p runs (distance fraud). This strategy is efficient against the protocol if the verifier sends his nonce first. This weakness can be easily fixed, though. The success probability is provided in Reference [2]:

$$\mathrm{Pr}_{\mathrm{DF|WB(p)|early}} = \left((1 - p_f) + p_f \cdot \left(1 - \frac{1}{2} \cdot \frac{\mathbb{E}(d_H(v^0, v^1))}{n}\right)\right)^n ,$$

where $\mathbb{E}(d_H(R^0, R^1))$ is the expected minimum Hamming distance between R^0 and R^1 for the non-void challenges after the hash function is run p times with a different N_P. We have:

$$\lim_{p \to \infty}[\mathrm{Pr}_{\mathrm{DF|WB(p)|early}}] = 1.$$

Pre-ask combined with early-reply strategy (distance fraud). In the black box model, the adversary carries out an attack similar to mafia fraud but on its own device: $\mathrm{Pr}_{\mathrm{DF|BB|pre\&early}} \approx \mathrm{Pr}_{\mathrm{MF|pre}}$. The approximation is due to a small difference in the two frauds as explained hereafter. As long as the adversary is not detected by the device, she has the same strategy (and same probability of success) in both mafia fraud and distance fraud. In particular, she no longer receives useful information from the device once she is detected. However, in mafia fraud, she can still determine whether or not a round contains a void challenge when communicating with the verifier, as she does not have time to get this information in distance fraud attacks. The difference is, however, negligible, because she has to guess the final signature in both cases.

Post-ask combined with early-reply strategy (distance fraud). In this strategy, the adversary definitely obtains the correct final signature. However, she does not know when a void or non-void challenge is expected. Therefore, if the probability of a non-void challenge is lower (respectively, higher) than 2/3 then her best strategy is to keep quiet (respectively, try to guess every response, with probability 1/2).

$$\mathrm{Pr}_{\mathrm{DF|BB|post\&early}} = \begin{cases} \left(1 - p_f\right)^n & \text{if } p_f < 2/3 \\ \left(\frac{p_f}{2}\right)^n & \text{if } p_f \geq 2/3 \end{cases} . \qquad [2]$$

This protocol is not designed to resist to terrorist fraud in the white box model. Indeed, the prover, without revealing his secret K, is able to provide to his accomplice the two registers required to successfully complete the protocol. We so have $\mathrm{Pr}_{\mathrm{TF|WB}} = 1$ [2].

7.1 Published Attacks

A technique to reduce the required memory [54] consists in using only one $(n + 1)$-bit register, where the responses are selected from the two edges. However, Reference [2] demonstrated that this technique opens the door to an attack where the adversary queries in advance the two values of the edges.

8 KIM, AVOINE, KOEUNE, STANDAERT, AND PEREIRA'S PROTOCOL (2008)

Kim, Avoine, Koeune, Standaert, and Pereira introduced a protocol in Reference [50] known as the *Swiss-knife distance-bounding protocol*[2] (Algorithm 6 in Appendix A). We only consider in this analysis the case where $T = 1$, that is, when the protocol is not noise resilient.

---

[2]Like Swiss-army knives used during WWII, the Swiss-knife protocol is a multi-purpose tool. The authors claim their protocol *"resists against both mafia fraud and terrorist attacks, reaches the best known false acceptance rate, preserves pri-*

The attacker could impersonate the prover by guessing all the answers during the fast phase and T_B in the second slow phase. To succeed, the adversary would need to guess $\sigma + n$ bits. Therefore, it is better for the adversary to guess K which size is σ. Consequently, we have $\mathrm{Pr_{Imp}} = (\frac{1}{2})^{\sigma}$.

Pre-ask strategy (mafia fraud). The success probability of the pre-ask strategy is $\mathrm{Pr_{MF|pre}} = (\frac{1}{2})^n$ [50].

Post-ask strategy (mafia fraud). The adversary must guess the responses expected in the fast phase: $\mathrm{Pr_{MF|post}} = (\frac{1}{2})^n$.

Early-reply strategy with one run (distance fraud). As the prover can access to the internal state of the registers, she knows the content of the two registers. If $v_i^0 = v_i^1$, then she always responds correctly; otherwise, she has to guess the correct answer with probability $\frac{1}{2}$. Hence, $\mathrm{Pr_{DF|WB(1)|early}} = (\frac{3}{4})^n$.

Early-reply strategy with p runs (distance fraud). The Swiss-knife Protocol generates only one register. Hence, multiple-run of PRF does not increase the adversary success probability: $\mathrm{Pr_{DF|WB(p)|early}} = \mathrm{Pr_{DF|WB(1)|early}}$.

Pre-ask combined with early-reply strategy (distance fraud). The adversary sends her own challenges to the prover in advance. To obtain the correct signature in the second slow phase, the challenges sent by the adversary must be the same as the challenges sent by the verifier. We thus have $\mathrm{Pr_{DF|BB|pre\&early}} = (\frac{1}{2})^n$.

Post-ask combined with early-reply strategy (distance fraud). The adversary has to correctly guess the response in each round. Hence: $\mathrm{Pr_{DF|BB|post\&early}} = (\frac{1}{2})^n$.

Early-provide strategy with one run (terrorist fraud). $\mathrm{Pr_{TF|WB(1)}} = (\frac{3}{4})^n$ [50].

Early-provide strategy with p runs (terrorist fraud). For the same reason given at Section 8: $\mathrm{Pr_{TF|WB(p)}} = \mathrm{Pr_{TF|WB(1)}}$.

8.1 Published Attacks

Peris-Lopez et al. proposed a passive full disclosure attack on the Swiss-knife RFID distance-bounding protocol [60]. However, their assumption is not correct: They assume that the size of the secret key (K) and random nonces (N_V and N_P) are equal to n (number of iterations in the fast phase) and n is insecurely short; for example, 32 bits or less in the Swiss-knife protocol. Based on this assumption, they assert that the Swiss-knife protocol is insecure. The authors of the Swiss-knife RFID distance-bounding protocol never claimed that their protocol is secure when the size of the long-term key and random nonces are so short. Under this assumption, all the distance-bounding protocols can be broken.

9 AVOINE AND TCHAMKERTEN'S PROTOCOL (2009)

The protocol (Algorithm 7 in Appendix A) introduced by Avoine and Tchamkerten in Reference [7] is a generalization of Hancke and Kuhn's protocol that is more secure in terms of mafia and distance fraud.

To impersonate a legitimate prover one needs to guess the c authentication bits and the n replies of the fast phase. Hence: $\mathrm{Pr_{Imp}} = (\frac{1}{2})^{c+n}$.

Pre-ask strategy (mafia fraud). $\mathrm{Pr_{MF|pre}} = 2^{-n}(\frac{d}{2}+1)^{\frac{n}{d}} = 2^{-d\cdot\ell}(\frac{d}{2}+1)^{\ell}$ with $n = d\ell$ [7].

Post-ask strategy (mafia fraud). Without any final slow phase, a post-ask strategy is useless: $\text{Pr}_{\text{MF}|\text{post}} = (\frac{1}{2})^{c+n}$.

Early-reply strategy with one run (distance fraud). The analysis of the distance fraud probability in the case of the tree-based protocol is very similar to the analysis of the Poulidor protocol (Section 11) that is provided in Reference [69]. Unfortunately, this analysis only yields rough upper bounds. To find such an upper bound on the adversary success probability for distance fraud for the tree-based protocol, Theorem 3, available in Reference [69], is used. This theorem is related to Poulidor, but the only difference between Poulidor and the tree-based protocol is that the latter creates a full tree as graph. Therefore, the distance fraud success probability of the tree-based protocol is upper bounded by

$$\frac{1}{2}\left(\frac{1}{2^n} + \sqrt{\frac{1}{2^{2n}} - \frac{4}{2^n} + 4q} \right), \text{ where } q = \prod_{i=1}^{i=n}\left(\frac{1}{2} + \frac{1}{2^{2i+1}} \sum_{k=0}^{k=2n-1} (A^i[0,k])^2 \right).$$

The authors of Reference [69] define $A^i[0,k]$ for a tree, considering that the nodes in the tree are labeled between 0 and $2^n - 1$ using a breadth-first algorithm, then

$$A^i[0,k] = \begin{cases} 1 \text{ if } & 2^i - 1 \le k < 2^{i+1} - 1, \\ 0 \text{ otherwise}, \end{cases} \text{ and, finally, } q = \prod_{i=1}^{i=n}\left(\frac{1}{2} + \frac{1}{2^{i+1}} \right).$$

Early-reply strategy with p runs (distance fraud). Similarly to the Poulidor case (Section 11), this strategy makes sense for this protocol, but so far neither $\text{Pr}_{\text{DF}|\text{WB}(1)|\text{early}}$ nor $\text{Pr}_{\text{DF}|\text{WB}(p)|\text{early}}$ have been calculated.

Pre-ask combined with early-reply strategy (distance fraud). $\text{Pr}_{\text{DF}|\text{BB}|\text{pre\&early}} = 2^{-n}(\frac{d}{2} + 1)^{\frac{n}{d}}$ [7].

Post-ask combined with early-reply strategy (distance fraud). Note that the post-ask strategy will not allow the adversary to gain any information, i.e., $\text{Pr}_{\text{DF}|\text{BB}|\text{post\&early}} = (\frac{1}{2})^{c+n}$.

This protocol is not designed to resist to terrorist fraud in the white box model. An attacker can reveal the tree node labelization to an accomplice who so successfully passes the fast phase with $\text{Pr}_{\text{TF}|\text{WB}} = 1$.

10 RASMUSSEN AND ČAPKUN'S PROTOCOL (2010)

The protocol (Algorithm 8 in Appendix A) was introduced by Rasmussen and Čapkun and originally appeared in Reference [63]. In this article, we consider the updated version that appeared in Reference [61].

We assume here that key size and nonce size are large enough to ensure that the probability of a key-recovery attack and a replay attack are negligible. The easiest manner to impersonate a prover is by forging the final signature. The success probability of this attack is $\text{Pr}_{\text{Imp}} = (\frac{1}{2})^\sigma$.

Pre-ask strategy (mafia fraud). To implement a mafia fraud attack using a pre-ask strategy an attacker has to guess the nonce N_V of the verifier. Otherwise, the final signature will not be valid. So, $\text{Pr}_{\text{MF}|\text{pre}} = (\frac{1}{2})^{\delta_V}$.

Post-ask strategy (mafia fraud). An attacker wishing to execute a mafia fraud attack must guess all the bits of the prover's nonce to be able to reply correctly. Thus, $\text{Pr}_{\text{MF}|\text{post}} = (\frac{1}{2})^{\delta_P}$.

Early-reply strategy with one run (distance fraud). A malicious prover wishing to execute a distance fraud attack must guess all the bits of the verifier's nonce to reply correctly. Hence, $\text{Pr}_{\text{DF}|\text{WB}(1)|\text{early}} = (\frac{1}{2})^{\delta_V}$ [61].

Early-reply strategy with p runs (distance fraud). The concept of round does not exist in this protocol, therefore $\mathrm{Pr}_{\mathrm{DF|WB(p)|early}} = \mathrm{Pr}_{\mathrm{DF|WB(1)|early}}$.

The security of this protocol does not depend on a well-behaved prover. Consequently, black-box success probabilities are the same as in the white-box model.

This protocol is not designed to resist to terrorist fraud in the white box model. Indeed, the prover, without revealing his secret K, is able to provide his accomplice with N_P, which is sufficient to successfully execute the fast phase. Hence, $\mathrm{Pr}_{\mathrm{TF|WB}} = 1$.

11 TRUJILLO-RASUA, MARTIN, AND AVOINE'S PROTOCOL (2010)

Poulidor, the graph-based distance-bounding protocol (Algorithm 9 in Appendix A) designed by Trujillo-Rasua, Martin, and Avoine [69], uses specific node and edge dependencies in the tree of the AT protocol [7]—which then can alternatively be represented by an acyclic graph. Poulidor benefits from a lower memory requirement compared to the AT protocol. Security is also reduced.

The common manner to impersonate a prover is by guessing all the answers during the fast phase. Hence, we have $\mathrm{Pr}_{\mathrm{Imp}} = (\frac{1}{2})^n$.

Pre-ask strategy (mafia fraud). Let

$$g(i,j,k) = \frac{1}{2} + \frac{1}{2^{j+i-2k+2}} \times \sum_{t=0}^{t=2n-1} \left(A^{j-t}[1,t]A^{i-t}[2,t] + A^{j-k}[2,t]A^{i-k}[1,t] \right),$$

where A is the adjacency matrix of the graph that represents the graph-based protocol [69]. Also, let

$$f(i,j,k) = \begin{cases} 1 & \text{if } j < k \text{ and } i = j, \\ \frac{1}{2} & \text{if } j < k \text{ and } i \neq j, \\ \frac{1}{2} & \text{if } j \geq k \text{ and } i < k, \\ g(i,j,k) & \text{if } j \geq k \text{ and } i \geq k. \end{cases}$$

We then have $\mathrm{Pr}_{\mathrm{MF|pre}} = \sum_{k=1}^{k=n} \frac{1}{2^k} \left(\prod_{j=k}^{j=n} \max(f(1,j,k), \ldots, f(n,j,k)) \right) + \frac{1}{2^n}$ [69].

Post-ask strategy (mafia fraud). This protocol does not contain any second slow phase, and the first slow phase consists of nonce exchanges only. As per Section 2, we have $\mathrm{Pr}_{\mathrm{MF|post}} = \mathrm{Pr}_{\mathrm{Imp}}$.

Early-reply strategy with one run (distance fraud). $\mathrm{Pr}_{\mathrm{DF|WB(1)|early}}$ is upper bounded by [69]

$$\frac{1}{2} \left(\frac{1}{2^n} + \sqrt{\frac{1}{2^{2n}} - \frac{4}{2^n} + 4q} \right), \text{ where } q = \prod_{i=1}^{i=n} \left(\frac{1}{2} + \frac{1}{2^{2i+1}} \sum_{k=0}^{k=2n-1} (A^i[0,k])^2 \right).$$

Computing in a similar way than in EReference [69], we find the following relation for $A^i[0,k]$:

$$A^i[0,k] = \begin{cases} \binom{i}{k-i} & \text{if } i \leq k \leq 2i, \\ 0 & \text{otherwise,} \end{cases} \text{ and finally: } q = \prod_{i=1}^{i=n} \left(\frac{1}{2} + \frac{\binom{2i}{i}}{2^{2i+1}} \right).$$

Note that finding an exact value for $\mathrm{Pr}_{\mathrm{DF|WB(1)|early}}$ is an NP-hard problem [68].

Early-reply strategy with p runs (distance fraud). This strategy makes sense for this protocol, but so far, neither has been computed $\mathrm{Pr}_{\mathrm{DF|WB(1)|early}}$, nor can $\mathrm{Pr}_{\mathrm{DF|WB(p)|early}}$ be computed.

Pre-ask combined with early-reply strategy (distance fraud). With the pre-ask strategy, the adversary may learn the values of a walk in the graph. Note that this is exactly the same knowledge obtained for an adversary attempting to perform a mafia fraud attack by using the pre-ask

strategy. However, contrary to the mafia fraud attack, the adversary does not receive any challenge from the verifier when she is performing a distance fraud attack. We consequently have $\text{Pr}_{\text{DF|BB|pre\&early}} \leq \text{Pr}_{\text{MF|pre}}$. The equality of this equation holds when the adversary actually receives every challenge before sending its corresponding response, i.e., when the adversary is in the close vicinity of the verifier.

Post-ask combined with early-reply strategy (distance fraud). $\text{Pr}_{\text{DF|BB|post\&early}} = \text{Pr}_{\text{Imp}}$.

This protocol is not designed to resist to terrorist fraud in the white box model. Indeed, the prover is able, without revealing his secret K, to provide his accomplice with the graph required to successfully pass thorough the protocol. Hence, $\text{Pr}_{\text{TF|WB}} = 1$.

12 KIM AND AVOINE'S PROTOCOL (KA2) (2011)

Kim and Avoine introduced in 2009 a distance-bounding protocol with mixed challenges [48], namely challenges known and challenges unknown in advance by the prover. Challenges known in advance allow the prover to help the verifier to detect an attack, but these challenges also allow the prover to succeed in performing a distance fraud. Kim and Avoine improved their protocol in 2011, yielding a new variant known as KA2 [49], which is analyzed in this section (Algorithm 10 in Appendix A).

Guessing the n answers r_i is enough to impersonate the prover: $\text{Pr}_{\text{Imp}} = (\frac{1}{2})^n$.

Pre-ask strategy (mafia fraud). $\text{Pr}_{\text{MF|pre}} = (\frac{3}{4})^{n-\alpha}(\frac{1}{2})^{\alpha} + \alpha(\frac{1}{2})^{n+1}$ [49].

Post-ask strategy (mafia fraud). This protocol does not contain any second slow phase, and the first slow phase consists of nonce exchanges only. As per Section 2, we have $\text{Pr}_{\text{MF|post}} = \text{Pr}_{\text{Imp}}$.

Early-reply strategy with one run (distance fraud). $\text{Pr}_{\text{DF|WB(1)|early}} = (\frac{3}{4})^{n-\alpha}$ [49].

Early-reply strategy with p runs (distance fraud). The success probability in case of early-reply strategy with p runs of the pseudo-random function is provided in Reference [2]:

$$\frac{1}{2^{p(n-\alpha)}}\left(\sum_{i=0}^{i=n-\alpha-1}\left(\frac{1}{2}\right)^i \cdot \left[\left(\sum_{j=i}^{j=n-\alpha}\binom{n-\alpha}{j}\right)^p - \left(\sum_{j=i+1}^{j=n-\alpha}\binom{n-\alpha}{j}\right)^p\right] + \left(\frac{1}{2}\right)^n\right).$$

Pre-ask combined with early-reply strategy (distance fraud). As with mafia fraud with pre-ask strategy, the success probability is $\text{Pr}_{\text{DF|BB|pre\&early}} = \text{Pr}_{\text{MF|pre}}$.

Post-ask combined with early-reply strategy (distance fraud). This protocol does not contain any second slow phase, and the first slow phase consists of nonce exchanges only. As per Section 2, we have $\text{Pr}_{\text{DF|BB|post\&early}} = \text{Pr}_{\text{Imp}}$.

This protocol is not designed to resist to terrorist fraud in the white box model. The prover can give the registers to his accomplice to successfully pass the protocol: $\text{Pr}_{\text{TF|WB}} = 1$.

13 YUM, KIM, HONG, AND LEE'S PROTOCOL (2010)

Yum, Kim, Hon, and Lee created a distance-bounding protocol with mutual authentication [76].

The only known way to succeed at the impersonation consists of guessing all the answers during the fast phase, which leads to the probability $\text{Pr}_{\text{Imp}} = (\frac{1}{2})^n$.

Pre-ask strategy (mafia fraud). Avoine and Kim proposed a new attack that yields a higher adversary success probability [4]. Their attack depends on the probability of finding D_i's, Pr_D, which varies according to the system parameters. Following this attack, the success probability of a mafia

fraud attack is at least:

$$\Pr_{\mathrm{MF|pre}} = \begin{cases} (\frac{5}{8})^n + \sum_{i=1}^{n} \frac{1}{4} \cdot (\frac{5}{8})^{i-1} \cdot (\frac{3}{4})^{n-i}, & \text{if } \Pr_D = 1 \\ (\frac{1}{2})^n + \sum_{i=1}^{n} \frac{3}{8} \cdot (\frac{1}{2})^{i-1} \cdot (\frac{5}{8})^{n-i}, & \text{if } \Pr_D = \frac{3}{4} \end{cases}. \qquad [4]$$

Post-ask strategy (mafia fraud). This protocol does not contain any second slow phase, and the first slow phase consists of nonce exchanges only. As per Section 2, we have $\Pr_{\mathrm{MF|post}} = \Pr_{\mathrm{Imp}}$.

Early-reply strategy with one run (distance fraud). $\Pr_{\mathrm{DF|WB(1)|early}} = (\frac{7}{8})^n$ [76].

Early-reply strategy with p runs (distance fraud). An attacker who impersonates B wins when $D_i = 1$ or $R_i^0 = R_i^1$. Indeed, when $D_i = 1$, the prover sends a challenge to the verifier (A) and so trivially wins the round. When $D_i = 0$, the roles are inverted. To win a round, the prover must send his response in advance. When $R_i^0 = R_i^1$, the potential answers are the same and the prover definitely wins. Running the cryptographic function p times allows the prover to find D with a higher Hamming weight than the average one, and R^0 and R^1 with a lower Hamming distance. In conclusion, the probability of success is higher with the YKHL Protocol than with the HK protocol, where the prover wins only when $R_i^0 = R_i^1$. The probability of success can be calculated by considering $\Pr(X = x) = \binom{n}{x}(\frac{1}{4})^x(\frac{3}{4})^{n-x}/2^{3n}$ instead of $\Pr(X = x) = \binom{n}{x}/2^n)$ in Reference [2].

Pre-ask combined with early-reply strategy (distance fraud). A distance fraud attack with the pre-ask strategy is similar to a mafia fraud in this case. Hence, $\Pr_{\mathrm{DF|BB|pre\&early}} = \Pr_{\mathrm{MF|pre}}$.

Post-ask combined with early-reply strategy (distance fraud). This protocol does not contain any second slow phase, and the first slow phase consists of nonce exchanges only. As per Section 2, we have $\Pr_{\mathrm{DF|BB|post\&early}} = \Pr_{\mathrm{Imp}}$.

This protocol is not designed to resist to terrorist fraud in the white box model. Indeed, the prover can give the registers to his accomplice to successfully pass the protocol: $\Pr_{\mathrm{TF|WB}} = 1$.

13.1 Published Attacks

Avoine and Kim demonstrated in Reference [4] that the security of YKHL protocol is far below what is claimed in Reference [76] and could be worse than the HK protocol w.r.t. mafia fraud resistance.

14 SKI PROTOCOLS (2013)

In References [14–16], the authors introduced a series of protocols called *SKI*. These protocols are presented in Algorithm 12.

The only known way to do this attack is to guess all the answers during the fast phase. So $\Pr_{\mathrm{Imp}} = (\frac{1}{q})^n$.

Pre-ask strategy (mafia fraud). With this strategy, the adversary can obtain one set of answers from the prover before executing the fast phase with the verifier. Without loss of generality, we assume that he obtains $\{F(1, a_1, K_1'), \ldots, F(1, a_n, K_n')\}$, i.e., the answers corresponding to the challenges c_i's equal to 1. Hence, at each round two cases occur: (a) the verifier's challenge is 1 and she knows the answer, this happens with probability $1/t$, or (b) the verifier's challenge is not 1, and thus she has to guess the answer and succeeds with probability $1/q$. Thus, the rounds independence yields to

$$\Pr_{\mathrm{MF|pre}} = \left(\frac{1}{t} \cdot 1 + \left(1 - \frac{1}{t}\right) \cdot \frac{1}{q}\right)^n = \left(\frac{q + t - 1}{qt}\right)^n. \quad \text{For SKI}_{\mathrm{pro}}, \text{ this is } \left(\frac{2}{3}\right)^n.$$

Post-ask strategy (mafia fraud). This protocol does not contain any second slow phase, and the first slow phase consists of nonce exchanges only. As per Section 2, we have $\Pr_{\mathrm{MF|post}} = \Pr_{\mathrm{Imp}}$.

Early-reply strategy with one run (distance fraud). Using this strategy, the adversary has to send her answers in advance. Due to the similarity between Hancke and Kuhn's protocol and SKI's protocol, the adversary applies a similar strategy to maximize her success probability. At each round she answers the most probable value among the possible registers. The most probable answer for a given round is the one that appears the most within the set $\{F(1, a_i, K_i'),$ $F(2, a_i, K_i'), \ldots, F(t, a_i, K_i')\}$ of possible answers for this round. To compute the adversary success probability, we define the events:

- $\mathcal{W}$: the adversary provides the correct answer at a given round.
- $\mathcal{B}_j$: $j = \max_{1 \leq l \leq q}\{X_l\}$,

where X_l is the number of appearance times of the lth element from $\mathbb{F}_q$ among the set $\{F(1, a_i,$ $K_i'), F(2, a_i, K_i'), \ldots, F(t, a_i, K_i')\}$. We then trivially have $\Pr(\mathcal{W}) = \sum_{j=1}^{j=t} \Pr(\mathcal{W}|\mathcal{B}_j)\Pr(\mathcal{B}_j)$, with $\Pr(\mathcal{W}|\mathcal{B}_i) = \frac{i}{t}$. Thus, using the above equation, we deduce $\Pr(\mathcal{W}) = \mathbb{E}(\max_{1 \leq l \leq q}\{X_l\}) \cdot \frac{1}{t}$. The tricky task consists in computing $\mathbb{E}(\max_{1 \leq l \leq q}\{X_l\})$. This is done below for the **SKI** protocol configurations suggested in Reference [14],

- $q = 2$, and $t = 2$: $\mathbb{E}(\max_{1 \leq l \leq q}\{X_l\}) = \frac{3}{2}$, and $\Pr(\mathcal{W}) = \frac{3}{4}$.
- $q = 2$, and $t = 3$: $\mathbb{E}(\max_{1 \leq l \leq q}\{X_l\}) = \frac{9}{4}$ and $\Pr(\mathcal{W}) = \frac{3}{4}$.
- $q = 2$, and $t = 4$: $\mathbb{E}(\max_{1 \leq l \leq q}\{X_l\}) = 3$ and $\Pr(\mathcal{W}) = \frac{3}{4}$.
- $q = 4$, and $t = 3$: $\mathbb{E}(\max_{1 \leq l \leq q}\{X_l\}) = \frac{15}{8}$ and $\Pr(\mathcal{W}) = \frac{5}{8}$.

Finally, the independence of the rounds provides $\Pr_{\text{DF|WB(1)|early}} = (\Pr(\mathcal{W}))^n$. For SKI$_{\text{pro}}$, this is $\left(\frac{3}{4}\right)^n$.

Early-reply strategy with p runs (distance fraud). This strategy does not make sense against these protocols. Indeed, since the prover does not have the verifier's nonce before he sends its, he cannot compute several outputs of the the pseudo-random function.

Pre-ask combined with early-reply strategy (distance fraud). Distance fraud with pre-ask strategy is here similar to mafia fraud: $\Pr_{\text{DF|BB|pre\&early}} = \Pr_{\text{MF|pre}}$.

Post-ask combined with early-reply strategy (distance fraud). This protocol does not contain any second slow phase, and the first slow phase consists of nonce exchanges only. As per Section 2, we have $\Pr_{\text{DF|BB|post\&early}} = \Pr_{\text{Imp}}$.

Early-provide strategy with one run (terrorist fraud). Using this strategy, the adversary obtains register(s) before the start of the fast phase. First, note that the setting in which $t' = t = q = 2$ (i.e., SKI$_{\text{lite}}$) does not resist against terrorist fraud. Second, to compute the success probability in the other cases, let k denote the number of registers given by the prover to the adversary. As stated in Reference [5], the insurance that no information could leak is provided by $k = t - 2$.

Once the adversary gets the $t - 2$ registers, she starts the fast phase with the verifier. Two cases occur as follows: (a) The verifier asks an answer coming from one of the $t - 2$ known registers. Thus, the adversary definitely knows the correct answer. Or (b) the verifier asks her an answer coming from one the two unknown registers, and she has to guess the correct answer. The adversary consequently succeeds with probability $\frac{1}{q}$. Given the rounds are independent, we finally have

$$\Pr_{\text{TF|WB(1)}} = \left(\frac{t-2}{t} \cdot 1 + \frac{2}{t} \cdot \frac{1}{q}\right)^n = \left(\frac{qt + 2(1-q)}{qt}\right)^n. \text{ For SKI}_{\text{pro}}, \text{ this is } \left(\frac{2}{3}\right)^n.$$

Table 5. Properties and Performance

Protocol	Adaptiveness	Mutual Authentication	Second Slow Phase	Independence of the Rounds	Exchanged Bits during Slow Phase	Exchanged Bits during Fast Phase	Memory
BC	No	No	Yes	Yes	$2\ell + n$	$2n$	$2n$
MAD	No	Yes	Yes	No	$2(\ell + \delta + \sigma)$	$2n$	$2n$
HK	No	No	No	Yes	2δ	$2n$	$2n$
MP	No	Yes	Yes	Yes	$2\delta + 3n$	$2(n - n_{void})$	$3n$
Swiss-knife	No	Option	Yes	Yes	$2(\delta+\sigma)+n$ ($+\sigma$ if mutual)	$2n$	$3n + 2\delta$
Tree-based	No	No	No	No iff $d \geq 2$	$2\delta + c$	$2n$	$\ell(2^{d+1}-2)$
RC	No	No	Yes	No rounds	σ	$2\delta_V$	$\delta_V + \delta_P$
Poulidor	No (*)	No	No	No	2δ	$2n$	$4n$
KA2	Yes	Yes	No	Yes	2δ	$2n$	$2n$
YKHL	No	Yes	No	Yes	2δ	$2n$	$3n$
SKI$_{pro}$	No	No	No	Yes	$2\delta + 2n$	$3n$	$2n$

(*) See Section 11 for a refined analysis about the adaptiveness.

Early-provide strategy with p runs (terrorist fraud). This strategy does not make sense against these protocols. Indeed, since the prover does not have the verifier's nonce before he sends its answers, he cannot compute several outputs of the the pseudo-random function.

15 PROTOCOL COMPARISON

This section summarises the properties of the introduced protocols. Because looking at individual properties is often meaningless, we use here a comparison method, formalised in Reference [6], that clusters protocols with similar features and highlights those clusters with properties that are not present in others.

15.1 Summary of Properties and Performance

Table 5 presents the properties and performance of every protocol analyzed in this article. The properties are described in Section 2. Tables 6 and 7 summarize which cryptographic building blocks are used and which security properties are expected by each considered protocol. Greyed cells in Table 7 contain results already known, while other cells contain values provided by this survey. When looking at the properties summarised in Table 5, 6, and 7, we observe that, for example, only two protocols resist all attacks with probability lower than 1: Swiss-knife and SKI. However, we are interested on a holistic and systematic analysis that accounts for (arguably) the most important features of distance bounding protocols. We provide such an approach next.

15.2 Cluster-based Comparison

Comparing distance-bounding protocols is quite a tricky task given the large number of parameters that can be considered. A given protocol P_1 can be better in terms of resistance against mafia fraud than another protocol P_2 but at the same time worse in terms of resistance against distance fraud. Thus, ranking P_1 and P_2 is very complicated. This section introduces a hierarchical clustering of the distance-bounding protocols. The key-point of the method relies on the observation that a protocol

Table 6. Cryptographic Building Blocks

Protocol	PRNG	Sym. Primitive	Commitment	Signature
BC	Yes		Yes	Yes
MAD	Yes	Yes		
HK	Yes	Yes		
MP	Yes	Yes		
Swiss-knife	Yes	Yes		
Tree-based	Yes	Yes		
RC	Yes		Yes	Yes
Poulidor	Yes	Yes		
KA2	Yes	Yes		
YKHL	Yes	Yes		
SKI	Yes	Yes		

Table 7. Adversary Success Probabilities

Protocol	Imp	Mafia Pre-ask BB	Mafia Post-ask BB	Distance Early-reply WB(1)	Distance Early-reply WB(p)	Distance Pre & Early BB	Distance Post & Early BB	Terrorist Early-provide WB(1)	Terrorist Early-provide WB(p)
BC	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	1	1
MAD	$(\frac{1}{2})^\sigma$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	1	1
HK	$(\frac{1}{2})^n$	$(\frac{3}{4})^n$	$(\frac{1}{2})^n$	$(\frac{3}{4})^n$	5	$(\frac{1}{2})^n$	$(\frac{3}{4})^n$	1	1
MP	$(1-\frac{p_f}{2})^n \times (\frac{1}{2})^{3n}$	$(1-p_f)^n$ if $p_f < \frac{4}{5}$; $(p_f \cdot \frac{3}{4})^n$ if $p_f \geq \frac{4}{5}$	$(1-\frac{p_f}{2})^n$	$(1-\frac{p_f}{4})^n$	$((1-p_f)+p_f \cdot (1-\frac{E(d_H(v^0,v^1))}{2m}))^n$	$(1-p_f)^n$ if $p_f < \frac{2}{3}$; $(\frac{p_f}{2})^n$ if $p_f \geq \frac{2}{3}$	$(1-p_f)^n$ if $p_f < \frac{4}{5}$; $(p_f \cdot \frac{3}{4})^n$ if $p_f \geq \frac{4}{5}$	1	1
Swiss-knife	$(\frac{1}{2})^\sigma$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{3}{4})^n$	$(\frac{3}{4})^n$	$(\frac{1}{2})^n$	$(\frac{1}{2})^n$	$(\frac{3}{4})^n$	$(\frac{3}{4})^n$
Tree-based	$(\frac{1}{2})^{c+n}$	$2^{-dt}(d/2+1)^\ell$	$(\frac{1}{2})^n$	Section 9	Open Prob	$(\frac{1}{2})^{c+n}$	$2^{-n}(d/2+1)^{n/d}$	1	1
RC	$(\frac{1}{2})^\sigma$	$(\frac{1}{2})^{\delta_V}$	$(\frac{1}{2})^{\delta_P}$	$(\frac{1}{2})^{\delta_V}$	$(\frac{1}{2})^{\delta_V}$	$(\frac{1}{2})^{\delta_V}$	$(\frac{1}{2})^{\delta_V}$	1	1
Poulidor	$(\frac{1}{2})^n$	Section 11	$(\frac{1}{2})^n$	Section 11	Open Prob	$(\frac{1}{2})^n$	$\leq \Pr_{MF|pre}$	1	1
KA2	$(\frac{1}{2})^n$	$(\frac{3}{4})^{n-\alpha}(\frac{1}{2})^\alpha + \alpha(\frac{1}{2})^{n+1}$	$(\frac{1}{2})^n$	$\frac{3}{4}^{n-\alpha}$	Section 12	$(\frac{1}{2})^n$	$(\frac{3}{4})^{n-\alpha}(\frac{1}{2})^\alpha + \alpha(\frac{1}{2})^{n+1}$	1	1
YKHL	$(\frac{1}{2})^n$	Section 13	$(\frac{1}{2})^n$	$(\frac{7}{8})^n$	Section 13	$(\frac{1}{2})^n$	Section 13	1	1
SKI$_{\text{pro}}$	$(\frac{1}{2})^n$	$(\frac{2}{3})^n$	$(\frac{1}{2})^n$	Section 14	Section 14	$(\frac{2}{3})^n$	$(\frac{1}{2})^n$	$(\frac{2}{3})^n$	$(\frac{2}{3})^n$

P_1 is undeniably better than a protocol P_2 if and only if P_1 is better than P_2 for every considered parameter. In such a case, P_1 should be used, instead of P_2, whatever the considered scenario.

Seven parameters are considered for the cluster-based comparison: mafia fraud resistance ($\mathcal{Ma}$), terrorist fraud resistance ($\mathcal{T}$), distance fraud resistance ($\mathcal{D}$), the presence of a final slow phase ($\mathcal{L}$), single bit exchanges during the fast phase ($\mathcal{B}$), the number of bits exchanged by the two parties during the whole protocol[3] ($\mathcal{E}$), and the memory dependency on the prover side regarding the number of rounds ($\mathcal{Me}$). Each parameter can be assigned with a value belonging to an (obviously) ordered set. For example, it is clearly better for a protocol to have a mafia fraud success probability equal to $(\frac{1}{2})^n$ than $(\frac{3}{4})^n$. Also, it is better not to need a final slow phase in the protocol and to use binary messages than ternary messages, and so on. For each parameter, the values can be ordered: (Value 6) $\prec$ (Value 5) $\prec$ (Value 4) $\prec$ (Value 3) $\prec$ (Value 2) $\prec$ (Value 1), where (Value i) $\prec$ (Value j) means that (Value j) is better than (Value i) or, said differently, (Value j) is more convenient than (Value i) when implementing a protocol. The parameters and their values are provided in Table 8.

---

[3]In Table 8, no value is given to the constant. Since we are interested in how the number of exchanged bits scales the number of rounds, the actual constant value does not really matter.

Table 8. Parameters and Their Values

	$\mathcal{M}a$	$\mathcal{T}$	$\mathcal{D}$	$\mathcal{L}$	$\mathcal{B}$	$\mathcal{E}$	$\mathcal{M}e$
Val 1	$ma = \left(\frac{1}{2}\right)^n$	$t = \left(\frac{1}{2}\right)^n$	$d = \left(\frac{1}{2}\right)^n$	N	Y	$e = 2n + \mathrm{cst}$	Linear
Val 2	$\lim\limits_{n\to\infty} ma = \left(\frac{1}{2}\right)^n$	$\lim\limits_{n\to\infty} t = \left(\frac{1}{2}\right)^n$	$\lim\limits_{n\to\infty} d = \left(\frac{1}{2}\right)^n$	Y	N	$e = 3n + \mathrm{cst}$	Affine
Val 3	$\left(\frac{1}{2}\right)^n < ma < \left(\frac{3}{4}\right)^n$	$\left(\frac{1}{2}\right)^n < t < \left(\frac{3}{4}\right)^n$	$\left(\frac{1}{2}\right)^n < d < \left(\frac{3}{4}\right)^n$			$e \geq 4 \cdot n$	Non-affine
Val 4	$ma = \left(\frac{3}{4}\right)^n$	$t = \left(\frac{3}{4}\right)^n$	$d = \left(\frac{3}{4}\right)^n$				
Val 5	$\left(\frac{3}{4}\right)^n < ma < 1$	$\left(\frac{3}{4}\right)^n < t < 1$	$\left(\frac{3}{4}\right)^n < d < 1$				
Val 6	$ma = 1$	$t = 1$	$d = 1$				

The *configuration* of a protocol is an element of the Cartesian product $\mathcal{M}a \times \mathcal{T} \times \mathcal{D} \times \mathcal{L} \times \mathcal{B} \times \mathcal{E} \times \mathcal{M}e$. All the possible configurations can be deduced from Table 8, and the configuration of each protocol presented in this work is provided in Table 9. Note that there is no total order relation in $\mathcal{M}a \times \mathcal{T} \times \mathcal{D} \times \mathcal{L} \times \mathcal{B} \times \mathcal{E} \times \mathcal{M}e$, but a configuration (ma, d, t, l, b, e, me) is better than a configuration (ma′, d′, t′, l′, b′, e′, me′) if it is better for every considered parameter: ma′ ≺ ma, d′ ≺ d, . . . , me′ ≺ me. As a consequence, there exist several *best* configurations in $\mathcal{M}a \times \mathcal{T} \times \mathcal{D} \times \mathcal{L} \times \mathcal{B} \times \mathcal{E} \times \mathcal{M}e$.

A *cluster* is a set (possibly empty) of protocols that have the same configuration (ma, d, t, l, bbb, e, me). A cluster is said to be better than another one if its configuration is better. The total number of clusters is large; it is actually equal to the cardinality of $\mathcal{M}a \times \mathcal{T} \times \mathcal{D} \times \mathcal{L} \times \mathcal{B} \times \mathcal{E} \times \mathcal{M}e$, which is $6^3 \cdot 2^2 \cdot 3^2 = 7{,}776$. However, 5,774 clusters are empty, meaning that no protocol matches the configuration of these clusters. The remaining 2,002 non-empty clusters still represent a large amount of information, which is difficult to condense in an article. To further reduce this information, only *best* configurations are kept.

This process can be easily automated. A hierarchy of clusters is built and the best cluster of every branch is kept. After performing this operation, only five clusters remain: {Poulidor}, {Swiss-Knife}, {SKI$_\mathrm{shamir}$}, {RC}, and {BC, MAD}. Four of the remaining clusters are actually singletons, which means that among all the published protocols, none of them are equivalent with respect to the seven considered parameters. In the remaining cluster, {BC, MAD}, BC and MAD are equivalent, since the mutual authentication is not considered in the configurations. We can also raise that, given constraints on memory, probabilities, and so on, the best known protocol to be used belongs to these six finalists.

It is finally interesting to compare these six finalists with the evolution provided in Figure 1. A protocol that is not a finalist should not necessarily be blamed: Most of them have been useful at some point and led to more evolved protocols. However, protocols published today should be new finalists in the cluster-based comparison, possibly after considering additional parameters in the comparison.

16 CONCLUSION

Distance-bounding authentication protocols represent a new class of protocols aiming to thwart distance-based attacks whose feasibility is rendered possible by emerging technologies. This survey provides a thorough state-of-the-art of existing protocols and introduces refined security analyses. The comparisons made provide designers with new means to evaluate their performance in

Table 9. Protocol Configurations

Protocols	ma	d	t	l	b	e	me
BC	$\left(\frac{1}{2}\right)^n$	$\left(\frac{1}{2}\right)^n$	1	yes	yes	$3n+$cst	Linear
MAD	$\left(\frac{1}{2}\right)^n$	$\left(\frac{1}{2}\right)^n$	1	yes	yes	$3n+$cst	Linear
HK	$\left(\frac{3}{4}\right)^n$	$\left(\frac{3}{4}\right)^n$	1	N	yes	$2n+$cst	Linear
MP $p_f = 0.5$	$\left(\frac{3}{4}\right)^n$	$>\left(\frac{3}{4}\right)^n$ and <1	1	yes	yes	$\geq 4n$	Linear
MP $p_f = 0.75$	$>\left(\frac{3}{4}\right)^n$ and <1	$>\left(\frac{3}{4}\right)^n$ and <1	1	yes	yes	$\geq 4n$	Linear
Swiss-Knife	$\left(\frac{1}{2}\right)^n$	$\left(\frac{3}{4}\right)^n$	$\left(\frac{3}{4}\right)^n$	yes	yes	$3n+$cst	Affine
Tree-Based $\ell = \sqrt{n}$	$\lim\limits_{n\to\infty} = \left(\frac{1}{2}\right)^n$	$\lim\limits_{n\to\infty} = \left(\frac{1}{2}\right)^n$	1	N	yes	$3n+$cst	Non-affine
Tree-Based $\ell = 1$	$\lim\limits_{n\to\infty} = \left(\frac{1}{2}\right)^n$	$\lim\limits_{n\to\infty} = \left(\frac{1}{2}\right)^n$	1	N	yes	$3n+$cst	Non-affine
RC	$\left(\frac{1}{2}\right)^n$	$\left(\frac{1}{2}\right)^n$	1	yes	N	$2n+$cst	Affine
Poulidor	$\lim\limits_{n\to\infty} = \left(\frac{1}{2}\right)^n$	$\lim\limits_{n\to\infty} = \left(\frac{1}{2}\right)^n$	1	N	yes	$2n+$cst	Linear
KA2 $\alpha = n$	$\lim\limits_{n\to\infty} = \left(\frac{1}{2}\right)^n$	1	1	N	yes	$2n+$cst	Linear
KA2 $\alpha = \frac{n}{2}$	$>\left(\frac{1}{2}\right)^n$ and $<\left(\frac{3}{4}\right)^n$	$>\left(\frac{3}{4}\right)^n$ and <1	1	N	yes	$2n+$cst	Linear
YKHL	$>\left(\frac{3}{4}\right)^n$ and <1	$>\left(\frac{3}{4}\right)^n$ and <1	1	N	yes	$2n+$cst	Linear
SKI$_{\text{shamir}}$	$\left(\frac{1}{2}\right)^n$	$>\left(\frac{1}{2}\right)^n$ and $<\left(\frac{3}{4}\right)^n$	$\left(\frac{1}{2}\right)^n$	N	N	$\geq 4n$	Linear
SKI$_{\text{pro}}$	$>\left(\frac{1}{2}\right)^n$ and $<\left(\frac{3}{4}\right)^n$	$\left(\frac{3}{4}\right)^n$	$>\left(\frac{1}{2}\right)^n$ and $<\left(\frac{3}{4}\right)^n$	N	N	$\geq 4n$	Linear
SKI$_4$	$>\left(\frac{1}{2}\right)^n$ and $<\left(\frac{3}{4}\right)^n$	$>\left(\frac{1}{2}\right)^n$ and $<\left(\frac{3}{4}\right)^n$	$\left(\frac{3}{4}\right)^n$	N	N	$\geq 4n$	Linear

a unified manner according to several security and resource parameters. It may be worthwhile to point out that the provided cluster-based comparison can easily be modified to better reflect specific practical considerations and/or to include other protocols. Finally, we are aware that attacks other than those considered in this article might exist. Addressing provable security of distance-bounding protocols is therefore a challenge for future research.

REFERENCES

[1] Imad M. Abbadi and Chris J. Mitchell. 2007. Digital rights management using a mobile phone. In *Proceedings of the 9th International Conference on Electronic Commerce (ICEC'07)*. ACM, 185–194.

[2] Gildas Avoine, Muhammed Ali Bingöl, Süleyman Kardaş, Cédric Lauradoux, and Benjamin Martin. 2011. A framework for analyzing RFID distance bounding protocols. *J. Comput. Secur.* 19, 2 (Mar. 2011), 289–317.

[3] Gildas Avoine, Christian Floerkemeier, and Benjamin Martin. 2009. RFID distance bounding multistate enhancement. In *Proceedings of the 10th International Conference on Cryptology in India*. LNCS, Vol. 5922. Springer, 290–307.

[4] Gildas Avoine and Chong Hee Kim. 2013. Mutual distance bounding protocols. *IEEE Trans. Mobile Comput.* 12, 5 (May 2013), 830–839.

[5] Gildas Avoine, Cédric Lauradoux, and Benjamin Martin. 2011. How secret-sharing can defeat terrorist fraud. In *Proceedings of the 4th ACM Conference on Wireless Network Security (WiSec'11)*. ACM, 145–156.

[6] Gildas Avoine, Sjouke Mauw, and Rolando Trujillo-Rasua. 2015. Comparing distance bounding protocols: A critical mission supported by decision theory. *Comput. Commun.* 67 (2015), 92–102.

[7] Gildas Avoine and Aslan Tchamkerten. 2009. An efficient distance bounding RFID authentication protocol: Balancing false-acceptance rate and memory requirement. In *Proceedings of the Information Security Conference.* LNCS, Vol. 5735. Springer, 250–261.

[8] Paramvir Bahl and Venkata N. Padmanabhan. 2000. RADAR: An in-building RF-based user location and tracking system. In *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies (INFO-COM'00)*. 775–784.

[9] Asli Bay, Ioana Cristina Boureanu, Aikaterini Mitrokotsa, Iosif-Daniel Spulber, and Serge Vaudenay. 2012. The bussard-bagga and other distance-bounding protocols under attacks. In *Proceedings of the China International Conference on Information Security and Cryptology (Inscrypt'12)*. LNCS, Vol. 7763. Springer.

[10] Samy Bengio, Gilles Brassard, Yvo Desmedt, Claude Goutier, and Jean-Jacques Quisquater. 1991. Secure implementation of identification systems. *J. Cryptol.* 4, 3 (1991), 175–183.

[11] Thomas Beth and Yvo Desmedt. 1990. Identification tokens - or: Solving the chess grandmaster problem. In *Proceedings of the Conference on Advances in Cryptology (CRYPTO'90)*. LNCS, Vol. 537. Springer, 169–177.

[12] Ioana Boureanu, Aikaterini Mitrokotsa, and Serge Vaudenay. 2015. Practical and provably secure distance-bounding. *J. Comput. Secur.* 23, 2 (2015), 229–257.

[13] Ioana Cristina Boureanu, Aikaterini Mitrokotsa, and Serge Vaudenay. 2012. On the pseudorandom function assumption in (secure) distance-bounding protocols. In *Proceedings of the Conference on Progress in Cryptology (LATIN-CRYPT'12)*. LNCS, Vol. 7496. Springer, 100–120.

[14] Ioana Cristina Boureanu, Aikaterini Mitrokotsa, and Serge Vaudenay. 2013. Practical & provably secure distance-bounding. In *Proceedings of the Information Security Conference (ISC'13)*. LNCS, Vol. 7807. Springer, 248–258.

[15] Ioana Cristina Boureanu, Aikaterini Mitrokotsa, and Serge Vaudenay. 2013. Secure & lightweight distance-bounding. In *Proceedings of the 2nd International Workshop on Lightweight Cryptography for Security & Privacy.* LNCS, Vol. 8162. Springer, 97–113.

[16] Ioana Cristina Boureanu, Aikaterini Mitrokotsa, and Serge Vaudenay. 2013. Towards secure distance bounding. In *Proceedings of the 20th International Workshop on Fast Software Encryption (FSE'13)*) LNCS, Vol. 8424. Springer.

[17] Stefan Brands and David Chaum. 1993. Distance-bounding protocols. In *Proceedings of the Conferene on Advances in Cryptology (EUROCRYPT'93)*. LNCS, Vol. 765. Springer, 344–359.

[18] Laurent Bussard and Walid Bagga. 2005. Distance-bounding proof of knowledge to avoid real-time attacks. In *Proceedings of the Conference on Security and Privacy in the Age of Ubiquitous Computing (IFIP'05)*, Vol. 181. Springer, 223–238.

[19] W. Camp. 2007. Digital rights management based on device proximity. (May 2007). US Patent No. 11/164, 289.

[20] Srdjan Čapkun, Levente Buttyán, and Jean-Pierre Hubaux. 2003. SECTOR: Secure tracking of node encounters in multi-hop wireless networks. In *Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN'03)*. ACM, 21–32.

[21] Srdjan Čapkun and Jean-Pierre Hubaux. 2006. Secure positioning in wireless networks. *IEEE J. Select. Areas Commun.* 24, 2 (Feb. 2006), 221–232.

[22] Humphrey Cheung. 2004. How To: Building a BlueSniper Rifle. Retrieved from http://www.tomsguide.com/us/how-to-bluesniper-pt1,review-408.html.

[23] Omar Choudary and Frank Stajano. 2011. Make noise and whisper: A solution to relay attacks. In *Security Protocols.* LNCS, Vol. 7114. Springer, 271–283.

[24] Jolyon Clulow, Gerhard P. Hancke, Markus Kuhn, and Tyler Moore. 2006. So near and yet so far: Distance-bounding attacks in wireless networks. In *Proceedings of the European Workshop on Security and Privacy in Ad-Hoc and Sensor Networks (ESAS'06)*. LNCS, Vol. 4357. Springer, 83–97.

[25] John H. Conway. 1976. *On Numbers and Games.* Number 6 in London Mathematical Society Monographs. Academic Press, New York, NY.

[26] Cas J. F. Cremers, Kasper Bonne Rasmussen, Benedikt Schmidt, and Srdjan Čapkun. 2012. Distance hijacking attacks on distance bounding protocols. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P'12)*. 113–127.

[27] Yvo Desmedt. 1988. Major securitv problems with the unforgeable (Feige)-Fiat-Shamir proofs of identity and how to overcome them. In *Proceedings of the Worldwide Congress on Computer and Communications Security and Protection (SecuriCom'88)*. 147–159.

[28] Yvo Desmedt, Claude Goutier, and Samy Bengio. 1988. Special uses and abuses of the fiat-shamir passport protocol. In *Proceedings of the Advances in Cryptology (CRYPTO'87)*. LNCS, Vol. 293. Springer, 21–39.

[29] Saar Drimer and Steven J. Murdoch. 2007. Keep your enemies close: Distance bounding against smartcard relay attacks. In *Proceedings of the USENIX Security Symposium (USENIX'07)*. 1–16.

[30] Ulrich Dürholz, Marc Fischlin, Michael Kasper, and Cristina Onete. 2011. A formal approach to distance bounding RFID protocols. In *Proceedings of the 14th Information Security Conference (ISC'11)*. LNCS, Vol. 7001. Springer, 47–62.

[31] Uriel Feige, Amos Fiat, and Adi Shamir. 1987. Zero knowledge proofs of identity. In *Proceedings of the 19th ACM Symposium on Theory of Computing (STOC'87)*. 210–217.

[32] Amos Fiat and Adi Shamir. 1986. How to prove yourself: Practical solutions to identification and signature problems. In *Proceedings of the Conference on Advances in Cryptology (CRYPTO'86)* LNCS, Vol. 263. Springer, 186–194.

[33] Marc Fischlin and Cristina Onete. 2013. Subtle kinks in distance-bounding: An analysis of prominent protocols. In *Proceedings of the 6th ACM Conference on Security and Privacy in Wireless and Mobile Networks (WISEC'13)*. 195–206.

[34] Marc Fischlin and Cristina Onete. 2013. Terrorism in distance bounding: Modeling terrorist-fraud resistance. In *Proceedings of the International Conference on Applied Cryptography and Network Security (ACNS'13)*. LNCS, Vol. 7954. Springer, 414–431.

[35] Aurélien Francillon, Boris Danev, and Srdjan Čapkun. 2011. Relay attacks on passive keyless entry and start systems in modern cars. In *Proceedings of the Network and Distributed System Security Symposium*.

[36] Lishoy Francis, Gerhard Hancke, and Keith Mayes. 2013. A practical generic relay attack on contactless transactions by using NFC mobile phones. *Int. J. RFID Secur. Cryptogr.* 2, 1 (Dec. 2013), 92–106.

[37] Lishoy Francis, Gerhard P. Hancke, Keith Mayes, and Konstantinos Markantonakis. 2010. Practical NFC peer-to-peer relay attack using mobile phones. In *Proceedings of the Workshop on RFID Security*. LNCS, Vol. 6370. Springer, 35–49.

[38] Mohammad Ghavami, Lachlan B. Michael, and Ryuji Kohno. 2004. *Ultra Wideband Signals and Systems in Communication Engineering*. Wiley, New York, NY.

[39] Tzipora Halevi, Haoyu Li, Di Ma, Nitesh Saxena, Jonathan Voris, and Tuo Xiang. 2013. Context-aware defenses to RFID unauthorized reading and relay attacks. *IEEE Trans. Emerg. Top. Comput.* 1, 2 (Dec. 2013), 307–318.

[40] Gerhard P. Hancke. 2006. Practical attacks on proximity identification systems (short paper). In *Proceedings of the IEEE Symposium on Security and Privacy IEEE (S&P'06)*. IEEE, 328–333.

[41] Gerhard P. Hancke. 2010. Design of a secure distance-bounding channel for RFID. *J. Netw. Comput. Appl.* 34, 3 (May 2010), 877–887.

[42] Gerhard P. Hancke and Markus Kuhn. 2005. An RFID distance bounding protocol. In *Proceedings of the Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm'05)*. IEEE, 67–73.

[43] Gerhard P. Hancke and Markus Kuhn. 2008. Attacks on time-of-flight distance bounding channels. In *Proceedings of the 1st ACM Conference on Wireless Network Security (WiSec'08)*. ACM, 194–202.

[44] Gerhard P. Hancke, Keith Mayes, and Konstantinos Markantonakis. 2009. Confidence in smart token proximity: Relay attacks revisited. *Comput. Secur.* 28, 7 (Jun. 2009), 615–627.

[45] Orhun Kara, Süleyman Kardaş, Muhammed Ali Bingöl, and Gildas Avoine. 2010. Optimal security limits of RFID distance bounding protocols. In *Proceedings of the Workshop on RFID Security*. LNCS, Vol. 6370. Springer, 220–238.

[46] Süleyman Kardaş, Mehmet Sabir Kiraz, Muhammed Ali Bingöl, and Hüseyin Demirci. 2011. A novel RFID distance bounding protocol based on physically unclonable functions. In *Proceedings of the Workshop on RFID Security*. LNCS, Vol. 7055. Springer, 78–93.

[47] Ziv Kfir and Avishai Wool. 2005. Picking virtual pockets using relay attacks on contactless smartcard systems. In *Proceedings of the Conference on Security and Privacy for Emerging Areas in Communication Networks (SecureComm'05)*. IEEE, 47–58.

[48] Chong Hee Kim and Gildas Avoine. 2009. RFID distance bounding protocol with mixed challenges to prevent relay attacks. In *Proceedings of the International Conference on Cryptology and Network Security (CANS'09)*. LNCS, Vol. 5888. Springer, 119–133.

[49] Chong Hee Kim and Gildas Avoine. 2011. RFID distance bounding protocols with mixed challenges. *IEEE Trans Wireless Commun.* 10, 5 (2011), 1618–1626.

[50] Chong Hee Kim, Gildas Avoine, François Koeune, François-Xavier Standaert, and Olivier Pereira. 2008. The swiss-knife RFID distance bounding protocol. In *Proceedings of the International Conference on Information Security and Cryptology (ICISC'08)*. LNCS, Vol. 5461. Springer, 98–115.

[51] Sjouke Mauw, Jorge Toro Pozo, and Rolando Trujillo-Rasua. 2016. A class of precomputation-based distance-bounding protocols. In *Proceedings of the 1st IEEE European Symposium on Security and Privacy (Euro S&P)*.

[52] Sjouke Mauw, Jorge Toro Pozo, and Rolando Trujillo-Rasua. 2016. Optimality results on the security of lookup-based protocols. In *Proceedings of the Workshop on RFID Security*. LNCS. Springer.

[53] Aikaterini Mitrokotsa, Christos Dimitrakakis, Pedro Peris-Lopez, and Julio C. Hernandez-Castro. 2010. Reid et al.'s distance bounding protocol and mafia fraud attacks over noisy channels. *IEEE Commun. Lett.* 14, 2 (Feb. 2010), 121–123.

[54] Jorge Munilla, Andres Ortiz, and Alberto Peinado. 2006. Distance bounding protocols with void-challenges for RFID. In *Proceedings of the Workshop on RFID Security*. Ecrypt.

[55] Jorge Munilla and Alberto Peinado. 2008. Distance bounding protocols for RFID enhanced by using void-challenges and analysis in noisy channels. *Wireless Commun. Mobile Comput.* 8, 9 (Jan. 2008), 1227–1232.

[56] Jorge Munilla and Alberto Peinado. 2008. Security analysis of tu and piramuthu's protocol. In *Proceedings of the Conference on New Technologies, Mobility and Security (NTMS'08)*. IEEE, 1–5.

[57] Jorge Munilla and Alberto Peinado. 2009. Enhanced low-cost RFID protocol to detect relay attacks. *Wireless Commun. Mobile Comput.* 10, 3 (Mar. 2009), 361–371.

[58] Ventzislav Nikov and Marc Vauclair. 2008. Yet Another Secure Distance-Bounding Protocol. *Cryptology ePrint Archive, Report 2008/319.* (2008).

[59] Pedro Peris-Lopez, Julio C. Hernandez-Castro, Juan M. Estevez-Tapiador, Esther Palomar, and Jan C. A. van der Lubbe. 2010. Cryptographic puzzles and distance-bounding protocols: Practical tools for RFID security. In *Proceedings of the IEEE Annual International Conference on RFID 2010*. 45–52.

[60] Pedro Peris-Lopez, Julio C. Hernandez-Castro, Juan M. Estevez-Tapiador, and Jan C. A. van der Lubbe. 2009. Shedding Some Light on RFID Distance Bounding Protocols and Terrorist Attacks. *CoRR* abs/0906.4618 (2009).

[61] Kasper Bonne Rasmussen. 2011. *Primitives for Secure Localization and Location Verification*. Ph.D. Dissertation. ETH Zurich.

[62] Kasper Bonne Rasmussen, Claude Castelluccia, Thomas S. Heydt-Benjamin, and Srdjan Čapkun. 2009. Proximity-based access control for implantable medical devices. In *Proceedings of the ACM Conference on Computer and Communications Security*. ACM, 43–53.

[63] Kasper B. Rasmussen and Srdjan Čapkun. 2010. Realization of RF distance bounding. In *Proceedings of the USENIX Security Symposium (USENIX'10)*. USENIX, 389–402.

[64] Jason Reid, Juan Gonzalez Nieto, Tee Tang, and Bouchra Senadji. 2007. Detecting relay attacks with timing based protocols. In *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security (ASI-ACCS'07)*. ACM, 204–213.

[65] Dave Singelée and Bart Preneel. 2007. Distance bounding in noisy environments. In *Proceedings of the Security and Privacy in Ad-hoc and Sensor Networks (ESAS'07)*. LNCS, Vol. 4572. Springer, 101–115.

[66] Mohammad Reza Sohizadeh Abyaneh. 2011. Security analysis of two distance-bounding protocols. In *Proceedings of the Workshop on RFID Security*. LNCS, Vol. 7055. Springer, 94–107.

[67] Frank Stajano, Ford-Long Wong, and Bruce Christianson. 2010. Multichannel protocols to prevent relay attacks. In *Proceedings of the 14th International Conference on Financial Cryptography and Data Security (FC'10)*. LNCS, Vol. 6052. Springer, 4–19.

[68] Rolando Trujillo-Rasua. 2013. Complexity of distance fraud attacks in graph-based distance bounding. In *Mobile and Ubiquitous Systems: Computing, Networking, and Services*. Springer.

[69] Rolando Trujillo-Rasua, Benjamin Martin, and Gildas Avoine. 2010. The poulidor distance-bounding protocol. In *Proceedings of the Workshop on RFID Security*. LNCS, Vol. 6370. Springer, 239–257.

[70] Rolando Trujillo-Rasua, Benjamin Martin, and Gildas Avoine. 2014. Distance bounding facing both mafia and distance frauds. *IEEE Trans. Wireless Commun.* 13, 10 (2014), 5690–5698.

[71] Yu-Ju Tu and Selwyn Piramuthu. 2007. RFID distance bounding protocols. In *Proceedings of the International EURASIP Workshop on RFID Technology*.

[72] Pascal Urien and Selwyn Piramuthu. 2014. Elliptic curve-based RFID/NFC authentication with temperature sensor input for relay attacks. *Decis. Support Syst.* 59 (Oct. 2014), 28–36.

[73] Serge Vaudenay. 2013. On modeling terrorist frauds. In *Proceedings of the International Conference on Provable Security (ProvSec'13)*. LNCS, Vol. 8209. Springer, 1–20.

[74] Serge Vaudenay. 2015. Private and secure public-key distance bounding - application to NFC. In *Proceedings of the Conference on Financial Cryptography and Data Security (FC'15)*. LNCS, Vol. 8975. Springer, 207–216.

[75] Brent R. Waters and Edward W. Felten. 2003. *Secure, Private Proofs of Locations*. Technical Report TR-667-03. Princeton Computer Science.

[76] Dae Hyun Yum, Jin Seok Kim, Sung Je Hong, and Pil Joong Lee. 2011. Distance bounding protocol for mutual authentication. *IEEE Trans. Wireless Commun.* 10, 2 (2011), 592–601.