Distance Bounding based on PUF*

Mathilde Igier and Serge Vaudenay

EPFL
CH-1015 Lausanne, Switzerland
http://lasec.epfl.ch

Abstract. Distance Bounding (DB) is designed to mitigate relay attacks. This paper provides a complete study of the DB protocol of Kleber et al. based on Physical Unclonable Functions (PUFs). We contradict the claim that it resists to Terrorist Fraud (TF). We propose some slight modifications to increase the security of the protocol and formally prove TF-resistance, as well as resistance to Distance Fraud (DF), and Man-In-the-Middle attacks (MiM) which include relay attacks.

1 Introduction

Wireless devices are subject to relay attacks. It is problematic because these devices are at the basis for authentication in many domains like payment with credit cards, building access control, or biometric passports [16, 19]. To ensure the security of wireless devices against relay attacks, Brands and Chaum [8] introduced the notion of *Distance Bounding* (DB) protocols in 1993. The idea is that a prover P must prove that he is close to a verifier V. Several attack models exist to make the verifier accept with a prover too far away from the verifier. The attacks described in the literature are: 1. *Distance Fraud attacks (DF)* [8]: A far away prover P tries to make V accept. No participant is close to V. 2. *Mafia Fraud attacks (MF)* [11]: A malicious actor A who does not hold the secret tries to make V accept using an honest but far away prover P. 3. *Distance Hijacking attacks (DH)* [10]: A malicious far away prover P which holds the secret tries to make V accept him using an honest prover, who holds another secret and is close to V. 4. *Terrorist Fraud (TF)* [11]: A malicious actor A who does not hold the secret tries to make V accept by colluding with a malicious far away prover P who holds the secret.

Most of the proposed protocols are vulnerable to Terrorist Fraud attacks [18, 23, 27, 28, 31]. Moreover, Hancke [17] observed that noisy-resilience in nearly all the protocols, including the SwissKnife protocol [24], allows Terrorist Fraud attacks. It was important to provide a clear model to ensure security against all types of threats. Avoine et al. [1] proposed the complete but rather informal ABKLM model. Dürholz et al. [12] provided a formal model to prove the security of the protocols. However, this model is too strong as admitted by the authors [13], and it is difficult to prove TF security in this model. Another model was proposed by Boureanu et al. [4].

Most of the proposed protocols are vulnerable to TF attacks but a few protocols provide security against all types of threats: the protocol of Fischlin and Onete [14], the SKI protocol [5, 6], DBopt protocols [7], the public-key DB protocols ProProx [35] and eProProx [34], and the anonymous DB protocol SPADE [9]. However, all these proofs are made on the assumption that in TF, the prover does not want to give his credential to the adversary for further application. This assumption is weak and does not correspond to reality. None of the

* This is the full version of the paper [20] presented at CANS'2016.

DB protocols in the plain model can provide TF security without this assumption, so, we should consider alternate models. DF, DH and TF security are easier to provide using tamper resistant hardware on the prover side because the prover cannot access his secret. Kılınç and Vaudenay [22] provide a new model for distance bounding protocols with secure hardware. In this model, the game consists of several verifier instances including a distinguished one V, hardware with their instances, instances of provers and actors. There is one distinguished hardware h with instances far away from V. The winning condition of this game is that V accepts.

- The DB protocol is DF-secure if the winning probability is negligible whenever there is no instance close to V.
- The DB protocol is MiM-secure if the winning probability is negligible whenever an honest prover is holding h (i.e. it can only be accessed by an honest and far away prover).
- The DB protocol is DH-secure if the winning probability is negligible whenever all close instances are honest.
- The DB protocol is TF-secure if the winning probability is negligible.

PUFs are tamper resistant hardware used in counterfeiting detection [30, 32] and authentication protocols [3, 15]. A PUF is a physical component which maps a challenge to a response. By definition, a PUF, as it is described in [29], has the following properties: non clonable, non emulable, a response R_i gives negligible information on a response R_j with $R_i \neq R_j$ and a PUF cannot be distinguished from a random oracle (as discussed in [2]). For simplicity reasons, we will treat PUFs as random oracles with access limited to their holder. The aim of our work is to provide a provably secure protocol using PUF in DB protocols. A TF-secure DB protocol based on PUF was proposed in [21]. Nevertheless, this protocol assumes that provers implement their protocol while using a PUF. In the model of Kleber et al. [25], the prover can implement any malicious protocol while accessing to the PUF, the protocol in [21] is trivially TF-insecure in this stronger model.[1] Kleber et al. design a protocol in [25] which is claimed to be secure in their model. However we contradict that fact in this paper and propose to modify it in order to improve the security.

Our contribution in this paper is as follows: 1. We show that the protocol proposed by Kleber et al. [25] is not secure against *Terrorist Fraud* which contradicts the claims from their authors; 2. We provide some slight modifications of this protocol which we call pufDB to improve its security; 3. We provide proofs of security for this pufDB protocol for the following attacks: *Distance Fraud*, *Mafia Fraud* and *Distance Hijacking*; 4. We prove the security of pufDB protocol against *Terrorist Fraud* when the prover is limited in the amount of bits per round he can send. The security strengthens when the distance from the prover to the verifier increases. Our protocol, pufDB provides security against Distance Fraud, Mafia Fraud and Distance Hijacking. In the case of a prover at a distance close to B, the protocol achieves a security of 2^{-10} against these three attacks in 61 rounds. In the case of a prover far to the verifier, the protocol can achieve a security of 2^{-20} against these three attacks in 28 rounds. To the best of our knowledge, pufDB is the first protocol which provides TF security even when the prover is allowed to leak his secret.

[1] In this protocol, the PUF is not used during the fast phase, so the malicious prover can give whatever is needed to complete the protocol to a close-by adversary.

2 The Kleber et al. Protocol

2.1 Details of the Protocol

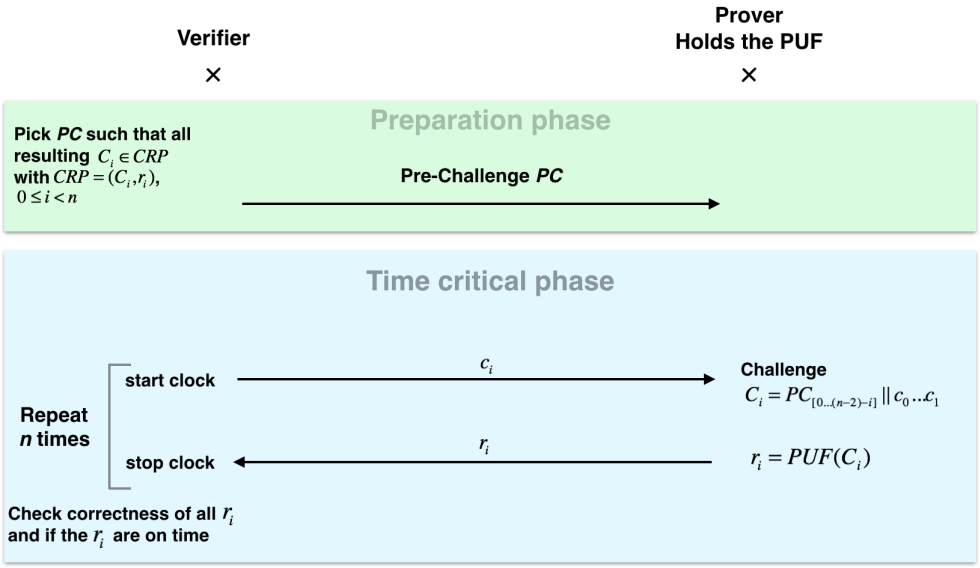

Fig. 1: *DB protocol from Kleber et al. [25]*

The verifier is called V and the prover P. The main idea of the protocol proposed by Kleber et al. [25] is to replace the PRF in P of conventional Distance Bounding protocols by a PUF. In this protocol, represented Figure 1, it is possible to use both Challenge-response PUF and a public PUF.[2] The protocol is made of two distinct phases: the preparation phase and the time critical phase.

Prior to the protocol, it is assumed that V can query the PUF and store a number of challenge-response pairs (CRP), at a round i such that $r_i = PUF(C_i)$. A CRP is defined as $(C_i, r_i), 0 \le i < n$ with n the number of rounds. There is always a set of CRPs corresponding to PC to complete the run. A set of CRPs shall not be used in protocols more than once.

In the time critical phase, only one bit can be sent from V to P in a round. However the PUF needs a big space of challenges to be secure. Therefore V transmits a pre-challenge PC to P during the preparation phase. Then, in the time critical phase, the pre-challenge is combined with the challenges c_i received by P to generate a challenge $C_i = \mathrm{PC}_0 ... PC_{n-2-i} || c_0 c_1 \dots c_i$ for the PUF. It is assumed that the hardware is such that the PUF can precompute C_i and when the prover receives the last bit of C_i he can return the response r_i in almost no time. The time critical phase consists of n transmission rounds. The verifier V starts the clock when he sends a challenge c_i and stops the clock when he receives the response r_i. In the paper,

[2] Normally, a PUF is non emulable so the verifier should first borrow the PUF to get input-output pairs. To avoid it, we can use Public-PUF also called SIMPL system (SIMulation Possible but Laborious). SIMPL systems guarantee that the response to a challenge cannot be computed faster with a simulator of the PUF than with the real PUF. Anyone can compute the right response but it takes much more time with the simulator of the PUF.

T_{max} and E_{max} are defined. T_{max} is the maximal number of responses which can arrive too late. E_{max} is the maximal number of errors admitted in the responses. (A late response is not checked.)

We note that if one c_i is incorrectly received by P, then all subsequent PUF computations will produce random outputs, independently from the expected r_i. So, this protocol is not tolerant to reception errors by P.

The protocol is claimed to be provably secure for all types of Fraud by Kleber et al. [25]. They prove the security of their protocol using the model of Dürholz et al. [12]. They only give a proof of security against Terrorist Fraud attacks. In fact, in the model defined by Kılınç et al. [22], when the protocol uses hardware, the proof that the protocol is secure against Terrorist Fraud attacks gives a proof of security against all the other types of attacks. However, when there is no additional restriction in the protocol, this protocol is insecure against Terrorist Fraud attack as we show in the section 2.2. To prove the security against Terrorist Fraud, Kleber et al. assume that the probability for the adversary to win the game is equal to $\left(\frac{1}{2}\right)^{n-E_{max}-T_{max}}$. We contradict this assumption.

2.2 A Terrorist Fraud Attack

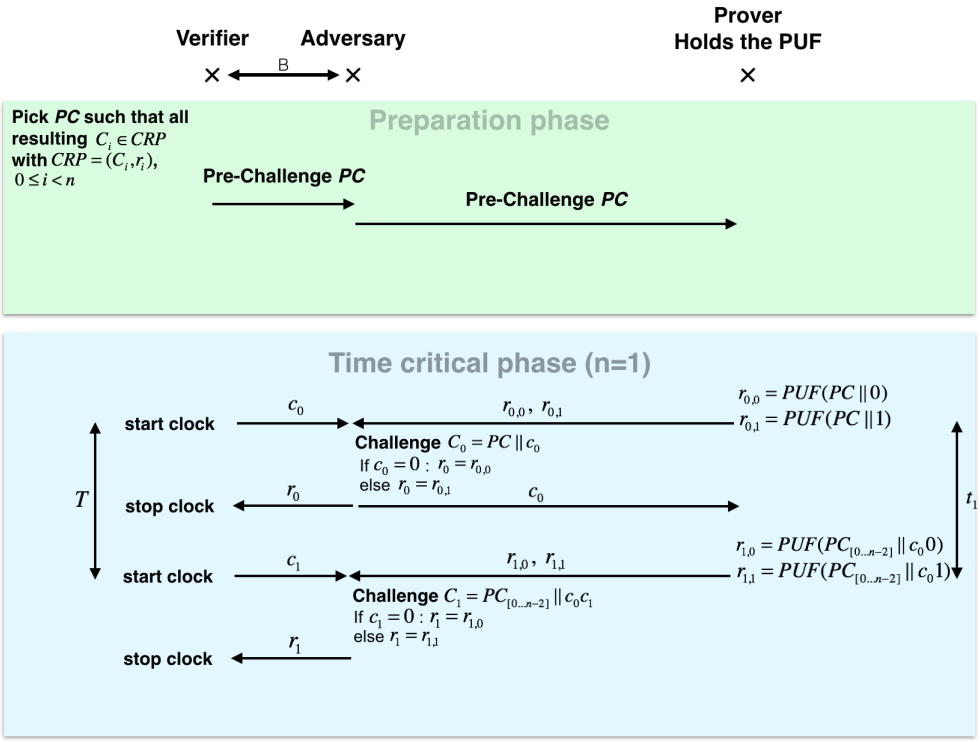

Fig. 2: *Terrorist Fraud attack,* $d_{VP} = 2B$, $T \ge t_1 + 2t_B$

Notations. d_{VP} is the distance between V and the far away prover P, t_{VP} is the signal propagation time between V and P (it is assume that $\frac{d_{VP}}{t_{VP}}$ is a constant such as the speed of

light); Similarly, d_{AP} is the distance between A and the far away prover P, t_{AP} is the signal propagation time between A and P; B is the maximal distance allowed by the protocol, t_B is the maximal signal propagation time over the distance B; Finally, T is the time between sending two consecutive challenges c_i and c_{i+1}.

In this scenario a malicious far away prover colludes with an adversary close to the verifier. In the protocol of Kleber et al. the adversary receives PC from the verifier. He can send it to the malicious prover who holds the PUF. There is no information concerning the distance d_{AP} between P and A nor about the time T in between rounds. A forwards every message from V to P. To answer a challenge c_i on time, P is missing m bits. He computes 2^m PUF values and sends them to A so that A will always be able to respond on time. For instance, if t_m denotes the time it takes for P to compute the 2^m values and to transmit them to A (without time of flight), the attack works if

$$t_{AP} + t_{VA} \leq t_B + \frac{(mT - t_m)}{2} \tag{1}$$

As an example, with $m = 1$, P has two PUF values to compute and to send and the condition is $t_{AP} + t_{VA} \leq t_B + \frac{T-t_1}{2}$. Since there is no information on d_{AP}, d_{VA} and T, we can have $d_{AP} = B$, $d_{VA} = B$ and $T \geq t_1 + 2t_B$, in that configuration Equation (1) is true. Then A can pass the round if he is in the previous configuration. He can pass all rounds with high probability, so the protocol is not secure against Terrorist Fraud. Figure 2 highlights this attack.

More concretely, we assume $m = 1$, $B = 3$m and $t_B = 10$ns. We consider V running at 1GHz and have one clock cycle between rounds, so $T = 1\mu$s. We consider a faster malicious prover P running at 10GHz so that he can evaluate two challenges with the PUF (corresponding to the possible challenges for m =1) in $t_m = 200$ns. With $d_{VA} = B$, the attack succeeds for $t_{AP} = 400$ns i.e $d_{VP} = 120$m. The attack is possible because there is a huge amount of time between the reception of r_i and the emission of c_{i+1}, but these figures clearly show it is a quite realistic scenario.

2.3 Slight Modifications of the Protocol

We choose to slightly modify the protocol of Kleber et al. [25] to improve its security. We call pufDB the new protocol. pufDB is presented on Figure 3. First, we impose a regular rhythm for sending the challenges, second, the $(n-1)$ bits of PC are sent with the same rhythm as if there were challenges in the time critical phase but expecting no answer. The prover begins to send responses when he receives the first bit of challenge c_0. With this slight change, we make sure there is no more time left for attacks in between the transmission of PC and c_0 than there is in between the transmission of each c_i and this time is bounded. Moreover, we assume that P cannot accept consecutive challenges separated by time lower than $\frac{T}{2}$, so, we cannot speed up P by sending challenges too fast.[3] Finally, another modification is that we concatenate PC with the challenges without dropping any bit. So, $C_i = PC||c_0...c_i$ is of $n+i$ bits. This guarantees domain separation for the functions computing the responses. So, to summarize, we use the three following requirements: 1. The elapsed time between sending each bit of $PC||c_0...c_{n-1}$ by V is exactly T; 2. The elapsed time in between receiving two consecutive bits by P is at least $\frac{T}{2}$; 3. PC is concatenated to $c_0...c_i$ without dropping any bit.

[3] We allow challenges to arrive faster than a period T to capture the Doppler effect when P moves towards V. With $\frac{T}{2}$ as a limit, P can move at 20% of the light speed!

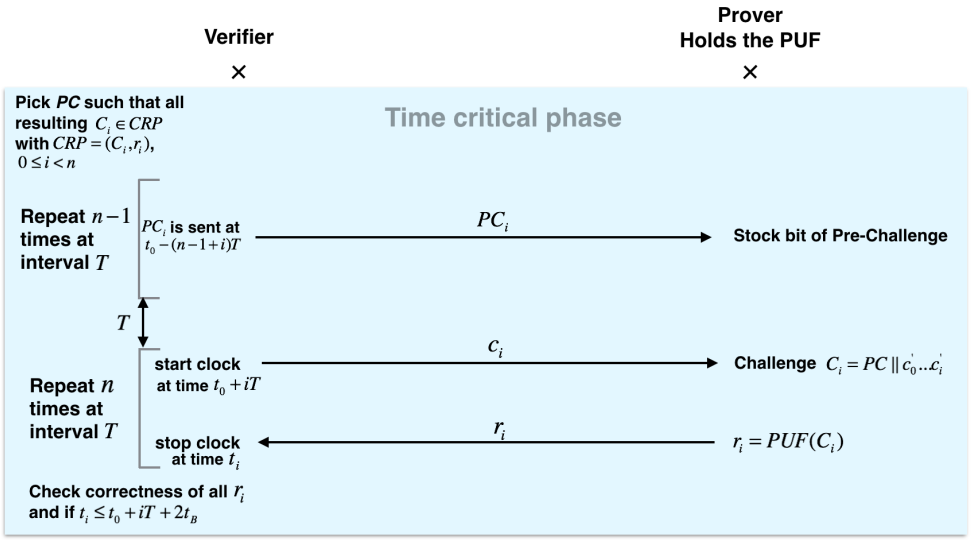

Fig. 3: *The pufDB protocol*

We denote by t_0 the time when the verifier sends c_0 to the prover. So c_i is sent at time $t_0 + iT$ and PC_i is sent at time $t_0 + (i - n + 1)T$.

Lemma 1 (Number of missing bits). *For each round i, the number of challenges which did not arrive yet to the far away prover P when it becomes critical to send the response r_i is $m = \lceil 2(\frac{t_{VP} - t_B}{T}) \rceil$. The number of possible C_i is 2^m.*

Proof. The proof is highlighted by the Figure 4. c_i is sent by V at time $t_0 + iT$ and c_i is received by P at time $t_0 + iT + t_{VP}$. The response r_i must be received at time $t_0 + iT + 2t_B$ the latest. To arrive on time, r_i is sent by P at time $t_0 + iT + 2t_B - t_{VP}$. The round of the last challenge received is called i_{last}. So, $t_0 + i_{last}T + t_{VP} \le t_0 + iT + 2t_B - t_{VP}$. Then $\lceil (i - i_{last}) \rceil = \lceil 2(\frac{t_{VP} - t_B}{T}) \rceil = m$. □

We will use the following bound:

Lemma 2 (Chernoff-Hoeffding [26]). *Given integers N and E and a probability p, we have*

$$\sum_{i=0}^{E} \binom{N}{i} p^{N-i}(1-p)^i \le e^{-2N\left(1-p-\frac{E}{N}\right)^2} \text{ for } E \le N(1-p)$$

3 Distance Fraud Analysis of pufDB

To prove resistance against Distance Fraud attacks, it is necessary to prove that a far away prover P who holds the PUF has a negligible probability to win the game presented in section 2. The idea of a Distance Fraud attack is to find a way for the far away prover P to send r_i such that it arrives on time to V. To arrive on time, the response r_i should be sent before receiving the challenge c_i. So, there are chances for the response to be wrong.

Lemma 3. *If $k = \lceil \frac{n}{m} \rceil$ and $n = (k-1)m + b$, with m from Lemma 1, then a DF-attack has a success probability limited by $\sum_{i=0}^{E_{max}+T_{max}} \binom{n}{i} \prod_{l=1}^{m} {p_l}^k$ with $p_l = \frac{1}{2} + \frac{1}{2} \times \frac{1}{2^{2^l}} \binom{2^l}{2^{l-1}}$.*

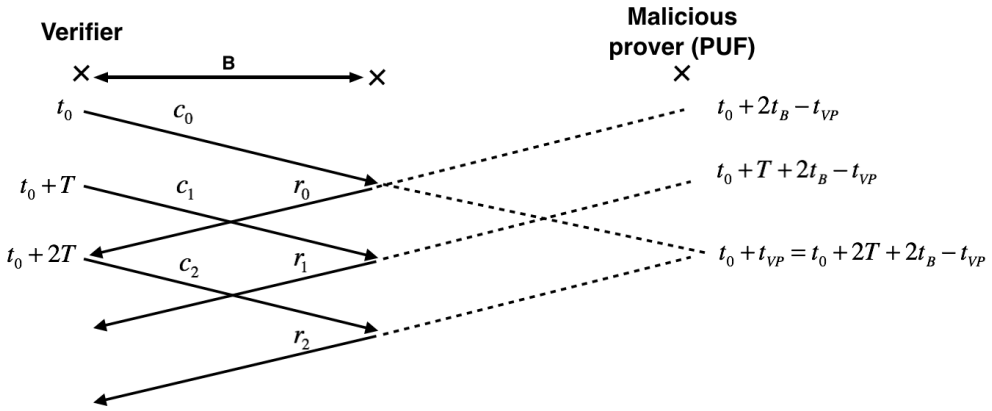

Fig. 4: *Missing data depending on distance from V to P when* $T = t_B$

This security bound does not seem so tight. Intuitively, p_m is the probability to guess correctly the output of a random (but known) boolean function on m random (but unknown bits). We would like to prove that it is

$$\sum_{i=0}^{E_{max}} \binom{n - T_{max}}{i} {p_m}^{n-T_{max}-i}(1-p_m)^{-i}$$

but we have troubles in proving it due to non independence between the rounds.

Proof. We prove security against DF attacks. We first describe what is the best possible attack. In the worst case, we assume no computational bound to the malicious prover. At each round i, P can compute each of the 2^m possible C_i (given that he is missing the last m bits of C_i) and their responses with the PUF. Then P can choose to :

- send a r_i early enough,
- send a response r_i (or none) which arrives too late. In this case, we denote $r_i = \perp$. P can do that strategy at most T_{max} times.

We remind that c_i is the challenge at round i, and $C_i = PC||c_0...c_i$. We denote by F the function realized by the PUF. We let F_i be the restriction of F over all strings of size exactly $(n+i)$ so that $r_i = F_i(C_i)$. Clearly all F_i are independent.

Let A be a DF algorithm. For $i = 0, ..., n-1$, this algorithm computes the responses r_i (or decides to be late) based on the received challenges and the function F realized by the PUF. Clearly, when A wins there exist index sets $I, J \subset \{0, ..., n-1\}$ such that $\#I \leq E_{max}$ and $\#J \leq T_{max}$ such that for all i, either $i \in J$ or $r_i = F_i(C_i) \oplus 1_{i \in I}$. As we want to upper bound the success probability, we can give advantages to the adversary. So, it is valid to allow infinite computational capabilities to A, to give more time to answer to challenges in some rounds and to allow T_{max} more errors instead of late answers. So, A wins if there exist I such that $\#I \leq E_{max} + T_{max}$ and for all i, $r_i = F_i(C_i) \oplus 1_{i \in I}$. That reduces to J empty. If I is fixed, it is equivalent to have I empty (it just changes the function F_i). So we just make the analysis for I and J empty and multiply by the number of possible I sets which is $\sum_i \binom{n}{i}$.

We let W_i be the event $r_i = F_i(C_i)$ (I and J are empty). We denote $W_{i_1,i_2} = W_{i_1} \cap ... \cap W_{i_2}$ with $i_1 \leq i_2$.

We divide the number of rounds n by m, $n = (k-1)m + b$ with $1 \leq b \leq m$. We split the n rounds into $k = \lceil \frac{n}{m} \rceil$ blocks and we allow the prover to answer to the whole block at once to make them independent. The first block goes from $i = 0$ to $i = b - 1$, we call it the block zero. The block i' (for $i' = 1, ..., k-1$) goes from $i = b + i'm - m$ to $i = b + i'm - 1$.

We called p_l the probability that a random l-bit string belongs to the largest preimage of a l-bit to 1-bit random function. We have

$$p_l = \sum_{v=0}^{2^l} \binom{2^l}{v} \frac{\max(v, 2^l - v)}{2^{l+2^l}}$$

with v the number of preimages that give 0 by the random function. $\max(v, 2^l - v)$ has symmetric properties so p_l is equal to

$$p_l = 2 \times \sum_{v=2^{l-1}+1}^{2^l} \binom{2^l}{v} \frac{v}{2^{l+2^l}} + \binom{2^l}{2^{l-1}} \frac{2^{l-1}}{2^{l+2^l}} = \frac{1}{2} + \frac{1}{2} \times \frac{1}{2^{2^l}} \binom{2^l}{2^{l-1}}$$

We denote by $C[i']$ the challenges of block $0, ..., i'$, $r[i']$ the response sent, $F[i']$ the function used to compute it, and $W[i']$ the event to win for the entire block, we have $r[i'] = A(C[i'-1], F[i'])$.

$$\begin{aligned}
\Pr(\text{win the game}) \leq & \sum_F \sum_{C_{n-1}} \Pr(F, W_{0,n-1}, C_{n-1}) \\
\leq & \sum_F \sum_{C_{n-1}} \Pr(F) \times \prod_{i'=1}^{k} \Pr(W[i'], C[i'] | W[0, ..., i'-1], C[i'-1], F[i']) \\
& \times \Pr(W[0], C[0] | PC, F[0])
\end{aligned}$$

As we assume no bound on A, we can assume it to be deterministic without loss of generality. Clearly, $W[i']$ is the event that for indices i not in I, we have a matching between $A(C[i'-1], F[i'])$ and $F[i'](C[i'])$. In $\Pr(W[i'], C[i'] | W[0, ..., i'-1], C[i'-1], F[i'])$, $C[i'-1]$ and $F[i']$ are fixed, so only the bits of $C[i']$ from the block i' are random. The value $\Pr(W[i'], C[i'] | W[0, ..., i'-1], C[i'-1], F[i'])$ is maximized when A computes $r[i']$ with the largest preimage $b_0...b_{m-1} \rightarrow F[i'](C[i'-1]||b_0...b_{m-1})$, so independently from the functions $F[0]...F[i'-1]F[i'+1]...F[k]$. We consider the greedy adversary maximizing all these possibilities. For this adversary, the probability simplifies to

$$\Pr(\text{win the game}) \leq \left(\prod_{i'=0}^{k} \Pr(W[i']) \right)$$

We define $f_l(x_1, ...x_l) = F_{b+l-1}(C_b x_1 ... x_l)$. Let p_l be the probability that a random $x_1...x_l$ is chosen in the largest preimage of f_l. We have

$$\Pr(W[0]) \leq \prod_{l=1}^{b} p_l \quad \text{and} \quad \Pr(W[i']) \leq \prod_{l=1}^{m} p_l \text{ for } i' > 0$$

So,

$$\Pr(\text{win the game}) \leq \prod_{l=1}^{m} {p_l}^k \prod_{l=1}^{b} p_l$$

for I fixed and we have to multiply this by the number of possible I to obtain the result. □

Theorem 1. *We use m from Lemma 1. We define $q_m = \prod_{l=1}^{m} {p_l}^{\frac{1}{m}}$ for $p_l = \frac{1}{2} + \frac{1}{2} \times \frac{1}{2^{2^l}} \binom{2^l}{2^{l-1}}$, in a DF-attack, we have that*

$$\Pr(\textit{win the game}) \leq \sum_{i=0}^{E_{max}+T_{max}} \binom{n}{i} {q_m}^n$$

For $2(E_{max} + T_{max}) \leq n$ any DF-attack is bounded by

$$\Pr(\textit{win the game}) \leq e^{-n \times \left(2\left(\frac{1}{2} - \frac{E_{max}+T_{max}}{n}\right)^2 - \ln(2q_m)\right)} = bound_{DF}$$

If there exist $\alpha, \beta \in \mathbb{R}$ such that $E_{max} \leq \alpha n$, $T_{max} \leq \beta n$ and $\alpha + \beta < 0.049$ then, $bound_{DF}$ is negligible.

Here is the table of the first values of q_m:

m	1	2	3	4	5	6	7	8	9
q_m	0.75	0.7181	0.6899	0.6657	0.6454	0.6283	0.6141	0.6022	0.5921

So depending on m, q_m smoothly goes from $\frac{3}{4}$ to $\frac{1}{2}$ as m grows. p_l decreases and tends towards $\frac{1}{2}$, so q_m decreases and tends towards $\frac{1}{2}$ as well.

Proof. The first bound is a straightforward consequence of Lemma 3. Then, we apply Lemma 2. We know that p_l decreases and $p_1 = \frac{3}{4}$ so we have $q_m \leq \frac{3}{4}$, the probability to win is negligible for $\alpha + \beta < \frac{1}{2} - \sqrt{\frac{\ln(2q_m)}{2}} \leq 0.049$. □

For $m \geq 2n - 1$, we can have a better bound. The adversary has no bit to compute the PUF (not even the bits of PC), so we can redo the analysis and obtain

$$\Pr(\text{win the game}) \leq \sum_{i=0}^{E_{max}+T_{max}} \binom{n}{i} {p_n}^n \leq e^{-n \times \left(2\left(\frac{1}{2} - \frac{E_{max}+T_{max}}{n}\right)^2 - \ln(2p_n)\right)}$$

These results are unchanged when using a public PUF.

4 Mafia Fraud Analysis of pufDB

To prove resistance against Mafia Fraud attacks it is necessary to prove that if an honest far away prover P holds the PUF, an adversary close to V has a negligible probability to win the game presented in section 2.

4.1 MiM Attack

We prove security against Man-in-the-Middle (MiM) attacks. We first informally describe what is the best possible attack. A is a malicious actor. Before receiving a challenge c_i from the verifier V, he sends a guessed challenge c_i' to a far away prover P. He receives r_i' from the prover. If $c_i' = c_i$ then the adversary sends r_i' to the verifier. In this case, the adversary wins the round with probability 1.

Pre-asking gives an extra chance to pass a round. But if one c_i is incorrectly guessed, any subsequent pre-asking request will return some useless random bits. So the best strategy is to start pre-asking until there exists a round i such that $c_i' \neq c_i$, then to continue with the impersonation attack strategy. This scenario is represented Figure 5.

We have not considered replay attacks because A has no time to begin any other instance of the protocol if P does not answer at frequency larger than $\frac{T}{2}$. Actually, let V be the distinguisher verifier in a MiM attack and PC the value that he sends. As the PUF is held by a single participant, there are no concurrent sessions for P. Sending c_i to P takes at least $\frac{(n+1)T}{2}$ time but during this time, the session for V terminates. So, only one session of P receives c_i, for each i.

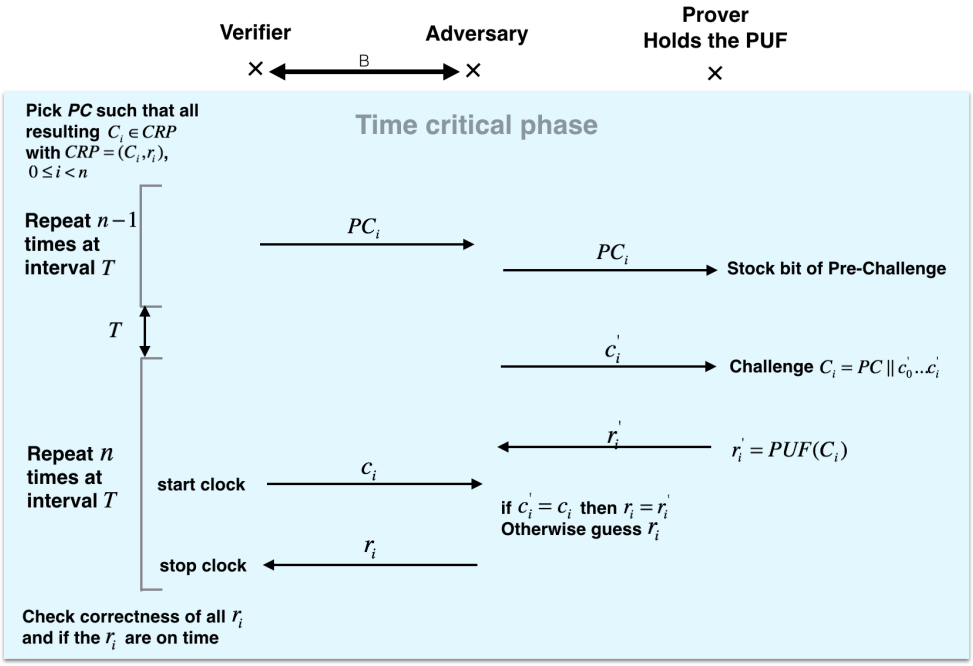

Fig. 5: *MiM attack, $i < K$*

Theorem 2. *In any MiM attack, we have*

$$\Pr(\textit{win the game}) \leq \left(\frac{1}{2}\right)^{n+1-T_{max}} \times \sum_{i=0}^{E_{max}+1} \binom{n+1-T_{max}}{i}$$

This is bounded by $e^{-2(n+1-T_{max})\times\left(\frac{1}{2}-\frac{E_{max}+1}{n+1-T_{max}}\right)^2}$ *when* $2E_{max}+T_{max} \leq n+1$. *For* $E_{max} \leq \alpha n$, $T_{max} \leq \beta n$, *and* $2\alpha+\beta < 1$, *this is negligible.*

Proof. At a round i, let $b_i = 1$ if it holds that challenge c'_i is pre-asked to P (i.e., A sent c'_i to P in his round i before receiving c_i from V in his round i) and $b_i = 0$ otherwise. Let K be the maximal (at most n) value such that $b_i = 1$ implies $c'_i = c_i$ for all $i = 0, ..., K-1$. It means that A pre-asks P with a correctly guessed c_i until the round number $K-1$. Clearly, either $K = n$ or $b_K = 1$ and $c'_K \neq c_K$.

In the $b_K = 1$ and $c'_K \neq c_K$ case, there is no time to restart a session with P with the same C_K so any $i > k$ such that $b_i = 1$ brings no information about the correct r_i to answer. At each round, A can decide either to skip this round (at most T_{max} times), or guess c_i, or to guess r_i. Whenever c_K is incorrectly guessed, this error does not count and A is offered an extra chance, either to skip this round or to guess r_K, but he can no longer use the guess c_i option. So, the game is equivalent to having $n+1$ rounds, being allowed to skip at most T_{max} ones, and to make $E_{max}+1$ errors, in which either A can guess c_i or to guess r_i but not both. So,

$$\Pr(\text{win the game}) \leq \sum_{i=0}^{E_{max}+1} \binom{n+1-T_{max}}{i} \left(\frac{1}{2}\right)^{n+1-T_{max}-i} \left(\frac{1}{2}\right)^{i}$$

This prove the first bound. According to the Lemma 2, for $2E_{max}+T_{max} \leq n+1$ we obtain

$$\Pr(\text{win the game}) \leq e^{-2(n+1-T_{max})\times\left(\frac{1}{2}-\frac{E_{max}+1}{n+1-T_{max}}\right)^2} = boundMiM$$

If there exist $\alpha, \beta \in \mathbb{R}$ such that $E_{max} \leq \alpha n$, $T_{max} \leq \beta n$ and $(2\alpha+\beta) < 1$ then, $bound_{MiM} = e^{-\frac{2n}{1-\beta}\times\left(\frac{1-\beta}{2}-\alpha\right)^2+o(n)} = O(e^{-\Omega(n)})$. So, the probability to win is negligible. □

Using a public PUF just adds a negligible term in the bound.

4.2 Impersonation Attack

The adversary receives c_i from V, he picks a random r'_i and sends it to the verifier.

Theorem 3. *The probability of success of an Impersonation attack is bounded by*

$$\left(\frac{1}{2}\right)^{n-T_{max}} \times \sum_{j=0}^{E_{max}} \binom{n-T_{max}}{j}$$

For $2E_{max}+T_{max} \leq n+1$, *this is bounded by* $e^{-2(n-T_{max})\times\left(\frac{1}{2}-\frac{E_{max}}{n-T_{max}}\right)^2}$. *For* $E_{max} \leq \alpha n$ *and* $T_{max} \leq \beta n$, *and* $2\alpha+\beta < 1$, *this is negligible.*

Proof. The best strategy consists to send late responses for T_{max} selected rounds, the way these rounds are selected is not important. So, $n - T_{max}$ is the number of bits the adversary has to guess. j is the number of errors made by the adversary during these rounds.

$$\Pr(\text{win the game}) = \sum_{j=0}^{E_{max}} \binom{n-T_{max}}{j} \left(\frac{1}{2}\right)^{n-T_{max}-j} \left(\frac{1}{2}\right)^{j}$$

This prove the first bound. According to Lemma 2, for $2E_{max} + T_{max} \leq n$ we obtain

$$\Pr(\text{win the game}) \leq e^{-2(n-T_{max})\times\left(\frac{1}{2}-\frac{E_{max}}{n-T_{max}}\right)^2} = bound_{IMP}$$

If there exist $\alpha, \beta \in \mathbb{R}$ such that $E_{max} \leq \alpha n$, $T_{max} \leq \beta n$ and $(2\alpha + \beta) < 1$ then, $bound_{IMP} = e^{-\frac{2n}{1-\beta}\times\left(\frac{1-\beta}{2}-\alpha\right)^2+o(n)} = O(e^{-\Omega(n)})$. So, the probability to win is negligible. □

5 Distance Hijacking Analysis of pufDB

To prove resistance against Distance Hijacking attacks it is necessary to prove that if a far away prover P^* holds PUF_1^* and uses interaction between the verifier and an other prover P (holding PUF_2) close to V (using CRPs from PUF_1^*), then P^* has a negligible probability to win the game presented in section 2.

The PUF is responsible of the authentication. If V interacts with another prover the responses will be different than those produced by the PUF. Then, if P^* wants to use P to pass the time critical phase, he needs to make P answer to challenges from V such that V thinks that responses come from the computation of challenges with PUF_1^*. P^* cannot interact with P or V because in pufDB all the bits are sent during the time critical phase, so P^*, who is far away, has no time to make an efficient attack (according to the proof of Distance Fraud resistance).

Then the only possibility for P^* to win is that V begins an instance with P in thinking that he is doing an instance with P^*. It means that responses from PUF_2 are the same as with PUF_1^* (these responses are stored in the database of V). So, in this scenario, V sends a challenge c_i to P who sends back r_i after computation with PUF_2. If $PUF_2(PC||c_0...c_i) = PUF_1^*(PC||c_0...c_i)$ then P^* passes the round i. The probability that it happens is $\frac{1}{2}$ because PUFs are similar to random oracles and are unique (non clonable, non emulable). V is honest so the rounds are independent, moreover we consider that P is honest but he can unfortunately send exactly T_{max} responses later than allowed. So

$$\Pr(\text{win the game}) \leq \left(\frac{1}{2}\right)^{n-E_{max}-Tmax}$$

which is negligible for $\alpha + \beta < 1$, $E_{max} \leq \alpha n$ and $T_{max} \leq \beta n$.

One detail is that P^* could try to learn PUF_2 prior to the attack to estimate his chances to win. P^* can do this with many participants to select the one increasing his chances, but he can only learn a negligible fraction of the function so this just adds a negligible term in the advantage.

This is unchanged when using a public PUF.

6 Terrorist Fraud Analysis of pufDB

In Terrorist Fraud attacks, an adversary A colludes with a far away malicious prover P to make V accept. Without any limitation on the power of the verifier the protocol is insecure against TF. In our model, the prover is limited on the communication complexity. With this limitation, the prover can compute all the challenges but he has a limitation on the amount of bits he can send to A. He can compress the 2^m bits of the table of responses for each round into s bits and send to A the compressed version. From the s bits received and the challenge sent by V, A can try to recover the response.

Notation. $B_N(R) = \sum_{i=0}^{R} \binom{N}{i}$.

Lemma 4. *Given $0 \leq \delta \leq R \leq \frac{N}{2}$ with δ integer, we have*

$$B_N(R) \geq \left(1 + \frac{\delta^2 - 1}{N}\right) B_N(R - \delta)$$

Proof. For $\delta \leq i \leq R$, we have

$$\binom{N}{i} = \left(\frac{N+1}{i} - 1\right) ... \left(\frac{N+1}{i-\delta+1} - 1\right) \binom{N}{i-\delta} \geq \left(\frac{N+1}{R - \frac{\delta}{2}} - 1\right)^{\frac{\delta}{2}} \binom{N}{i-\delta}$$

For this we kept the $\frac{\delta}{2}$ last factors and lower bounded the first ones by 1. Then, for $R \leq \frac{N}{2}$,

$$\frac{N+1}{R - \frac{\delta}{2}} - 1 \geq \frac{N}{\frac{N}{2} - \frac{\delta}{2}} - 1 \geq \frac{2}{1 - \frac{\delta}{N}} - 1 \geq 1 + \frac{2\delta}{N}$$

and finally

$$\left(1 + \frac{2\delta}{N}\right)^{\frac{\delta}{2}} \geq 1 + \frac{\delta^2}{N}$$

So, by summing we obtain the result. The above is for δ even, For δ odd, we split in $\frac{\delta-1}{2}$ and $\frac{\delta+1}{2}$ and finally obtain

$$\left(1 + \frac{2\delta}{N}\right)^{\frac{\delta}{2}} \geq 1 + \frac{\delta^2 - 1}{N}$$

□

Lemma 5. *Let s and N be two positive integers. Let R be the maximum value such that $\sum_{i=0}^{R} 2^s \binom{N}{i} \leq 2^N$. We have $R \geq \frac{N}{2} - \sqrt{\frac{Ns \ln 2}{2}} - 1$.*

Proof. If $R + 1 > \frac{N}{2}$, we have $R > \frac{N}{2} - 1 \geq \frac{N}{2} - \sqrt{\frac{Ns \ln 2}{2}} - 1$.

We now assume that $R+1 \leq \frac{N}{2}$. We have $B_N(R+1) \geq \frac{2^N}{2^s}$ and, due to Chernoff-Hoeffding bound presented in Lemma 2,

$$B_N(R+1) \leq 2^N e^{-2N\left(\frac{1}{2} - \frac{R+1}{N}\right)^2}$$

So,

$$\frac{2^N}{2^s} \leq 2^N e^{-2N\left(\frac{1}{2} - \frac{R+1}{N}\right)^2}$$

We deduce

$$R \geq \frac{N}{2} - \sqrt{\frac{Ns \ln 2}{2}} - 1$$

□

We consider a prover P and a verifier V receiving a random boolean function f with an input of l bits. The prover P can give a string of s bits to the adversary who ignores f. Then the adversary receives a random l-bit input x and tries to predict $f(x)$.

Every s-bit string given to the adversary defines an element g of a code C of size 2^s, we denote $g = help(f)$. The adversary answers to the input x by $g(x)$ and wins if $f(x) = g(x)$. Let $p_{help} = \Pr(f(x) = help(f)(x))$ where the probability is over the random choice of f and x. We have $p_{help} = 1 - \frac{1}{N}E(d(f, help(f)))$, where d denotes the Hamming distance and $N = 2^l$. Clearly the function $help$ is optimal when $help(f)$ is a closest element $g \in C$ to f. We define $p_C = 1 - \frac{1}{N}E(d(f, C))$. We want to bound $p_{l,s} = \max_{help} p_{help} = \max_C p_C$ over a code C of size 2^s.

Lemma 6. *Let s and l be two positive integers and $N = 2^l$. We define*

$$p_{l,s} = 1 - \frac{1}{N}E(\min_C d(f, C))$$

where f is a random boolean function of l-bit input and the minimum is over sets C of up to 2^s elements. We define

$$p^*_{l,s} = 1 - \frac{1}{2^N}\sum_{i=0}^{R+1} \frac{i}{N} N'_i \quad , \quad \bar{p}_{l,s} = \frac{1}{2} + \frac{1}{\sqrt{N}} \times \left(\sqrt{\frac{s \ln 2}{2}} + \sqrt{\frac{2}{2^s} + \frac{1}{N}}\right) + \frac{1}{N}$$

where R is the maximum value such that $\sum_{i=0}^{R} 2^s \binom{N}{i} \leq 2^N$ and $N'_i = 2^s \binom{N}{i}$ for $0 \leq i \leq R$, $N'_i = 0$ for $i > R+1$, and $N'_{R+1} = 2^N - 2^s \sum_{i=0}^{R} \binom{N}{i}$. We have $p_{l,s} \leq p^*_{l,s}$. For $s \leq \frac{2^l}{2}$, we also have $p^*_{l,s} \leq \bar{p}_{l,s}$.

Proof. Let N_i be the number of f such that $d(f, C) = i$. We have $p_C = 1 - \frac{1}{2^N}\sum_{i=0}^{N} \frac{i}{N} N_i$, clearly, $0 \leq N_i \leq 2^s \binom{N}{i}$ and $\sum_{i=0}^{N} N_i = 2^N$. Let R be the maximum value such that $\sum_{i=0}^{R} 2^s \binom{N}{i} \leq 2^N$. Let $N'_i = 2^s \binom{N}{i}$ for $0 \leq i \leq R$, $N'_i = 0$ for $i > R+1$, and $N'_{R+1} = 2^N - 2^s \sum_{i=0}^{R} \binom{N}{i}$. Clearly, the vector of all N'_i satisfies the same constraints as the vector of all N_i and we have $p_C \leq 1 - \frac{1}{2^N}\sum_{i=0}^{R+1} \frac{i}{N} N'_i$ for all C. So $p_{l,s} \leq 1 - \frac{1}{2^N}\sum_{i=0}^{R+1} \frac{i}{N} N'_i = p^*_{l,s}$. This prove the first bound.

For $l < 6$ we can check that $p^*_{l,s} \leq \bar{p}_{l,s}$ by direct inspection, so the result is true. We now assume that $l \geq 6$. Now, the average of i appearing N'_i times for $i = 0, ..., R+1$ is greater than the average of i appearing N'_i times for $i = 0, ..., R$. So,

$$p^*_{l,s} \leq 1 - \frac{1}{B_N(R)}\sum_{i=0}^{R} \frac{i}{N} N'_i = 1 - \frac{2^s}{B_N(R)}\sum_{i=0}^{R} \frac{i}{N}\binom{N}{i}$$

As expected, if $s = 2^l$, it means that the adversary has all the information to choose a response then $R = 0$ so $p^*_{l,s} = 1$, but we assume that $s \leq \frac{2^l}{2}$ to prove $p^*_{l,s} \leq \bar{p}_{l,s}$. In the particular case when $s = 0$, we have $R = N$ and the above bound gives $p^*_{l,s} \leq \frac{1}{2}$. So, we can exclude this case as it proves our result. Since $s > 0$, we have $\sum_{i=0}^{R} \binom{N}{i} \leq \frac{2^N}{2^s} \leq \frac{2^N}{2}$ so $R \leq \frac{N}{2}$. For $N \geq 4$ we have $\bar{p}_{l,2} \leq \bar{p}_{l,1}$. So, assuming $p^*_{l,2} \leq \bar{p}_{l,2}$, as $p^*_{l,s}$ can only increase with s, we have $p^*_{l,1} \leq p^*_{l,2} \leq \bar{p}_{l,2} \leq \bar{p}_{l,1}$ which proves the result for $s = 1$. Hence we just have to prove the bound for $s \geq 2$ and $l \geq 6$.

We take δ the smallest positive integer bigger than $\sqrt{\frac{2N}{2^s}+1}$ (i.e $\sqrt{\frac{2N}{2^s}+1} \leq \delta \leq \sqrt{\frac{2N}{2^s}+1}+1$), we assumed that $2 \leq s \leq \frac{2^l}{2}$ and $l \geq 6$. We can prove that $\sqrt{\frac{2N}{2^s}+1}+1 \leq \frac{N}{2} - \sqrt{\frac{Ns\ln 2}{2}} - 1$ so $\delta \leq R$ due to Lemma 5. For $i > R-\delta$, the terms in the sum are lower bounded by $\frac{R-\delta+1}{N}\binom{N}{i}$. For $i \leq R-\delta$, the terms are positive. So, we have

$$p^*_{l,s} \leq 1 - \frac{2^s}{B_N(R)} \sum_{i=R-\delta+1}^{R} \frac{i}{N}\binom{N}{i} \leq 1 - 2^s \times \frac{R-\delta+1}{N} \times \frac{B_N(R) - B_N(R-\delta)}{B_N(R)}$$

Using Lemma 4 we obtain

$$p^*_{l,s} \leq 1 - 2^s \times \frac{R-\delta+1}{N} \times \left(1 - \frac{1}{1+\frac{\delta^2-1}{N}}\right)$$

We can prove that $\sqrt{\frac{2N}{2^s}+1}+1 \leq \sqrt{N+1}$ for $s \geq 2$ and $l \geq 4$ so we have $\delta \leq \sqrt{N+1}$. We know that $\frac{1}{1+\frac{\delta^2-1}{N}} \leq 1 - \frac{\delta^2-1}{2N}$ for $\delta \leq \sqrt{N+1}$ so

$$p^*_{l,s} \leq 1 - 2^s \times \frac{R-\delta+1}{N} \times \frac{\delta^2-1}{2N}$$

We have $\delta \geq \sqrt{\frac{2N}{2^s}+1}$ and $\delta \leq \sqrt{\frac{2N}{2^s}+1}+1$, so we obtain

$$p^*_{l,s} \leq 1 - \frac{R - \sqrt{\frac{2N}{2^s}+1}}{N}$$

Using the inequality $R \geq \frac{N}{2} - \sqrt{\frac{Ns\ln 2}{2}} - 1$ from the Lemma 5 we obtain $p^*_{l,s} \leq \bar{p}_{l,s}$. □

Theorem 4. *We use m as defined in Lemma 1. We assume that the malicious prover is limited to s bits of transmission per round to the adversary in a TF-attack. We use $q_{m,s} = \prod_{l=1}^{m} {p_{l,s}}^{\frac{1}{m}}$ and we have*

$$\Pr(\textit{win the game}) \leq \sum_{i=0}^{E_{max}+T_{max}} \binom{n}{i} {q_{m,s}}^n$$

where $p_{l,s}$ is defined in Lemma 6. For $2(E_{max}+T_{max}) \leq n$ a TF-attack has a success probability bounded by

$$\Pr(\textit{win the game}) \leq e^{-n\times\left(2\left(\frac{1}{2}-\frac{E_{max}+T_{max}}{n}\right)^2 - \ln(2q_{m,s})\right)} = \textit{bound}_{TF}$$

If there exist $\alpha, \beta \in \mathbb{R}$ such that $E_{max} \leq \alpha n$, $T_{max} \leq \beta n$ then the protocol is secure when $\alpha + \beta < \frac{1}{2} - \sqrt{\frac{\ln(2q_m)}{2}}$.

Using a public PUF just adds a negligible term in the bound.

Proof. The first bound comes from the same technique as Th. 1 but using $p_{l,s}$. Then, we apply Lemma 2. If there exist $\alpha, \beta \in \mathbb{R}$ such that $E_{max} \leq \alpha n$, $T_{max} \leq \beta n$ then

$$bound_{TF} \leq e^{-n \times \left(2\left(\frac{1}{2}-\alpha-\beta\right)^2 - \ln(2q_{m,s})\right)}$$

which is negligible when $\alpha + \beta < \frac{1}{2} - \sqrt{\frac{\ln(2q_m)}{2}}$. □

For $l \leq 7$ and any s, we upper bound $p_{l,s}$ by $p^*_{l,s}$ because this formula is more precise. For $l \geq 8$, it is not possible to use the same formula because of heavy computations so we use $\bar{p}_{l,s}$ when $s \leq \frac{2^l}{2}$. For $s > \frac{2^l}{2}$ we use the bound 1 for $p_{l,s}$. Here is the table of upper bounds of $q_{m,s}$ for different values of s such that $s \leq \frac{2^m}{2}$:

m	6	8	10	20	30	50	500	1000
$q_{m,1}$	0.6283	0.6127	0.6021	0.5550	0.5362	0.5214	0.5021	0.5010
$q_{m,2}$	0.7170	0.6793	0.6534	0.5780	0.5509	0.5300	0.5029	0.5015
$q_{m,4}$	0.7969	0.7402	0.6998	0.5981	0.5636	0.5372	0.5036	0.5018
$q_{m,50}$		0.9413	0.8766	0.6816	0.6151	0.5662	0.5063	0.5031
$q_{m,100}$		0.9787	0.9214	0.7066	0.6302	0.5745	0.5070	0.5035
$q_{m,200}$			0.9599	0.7322	0.6456	0.5829	0.5077	0.5038
$q_{m,2^{10}}$				0.7917	0.6812	0.6020	0.5094	0.5047

So depending on m, $q_{m,s}$ smoothly goes from 1 to $\frac{1}{2}$ as m grows. From the definition of $p_{l,s}$, it is clear that $p_{l,s}$ decreases and tends to $\frac{1}{2}$. So, $q_{m,s}$ decreases and tends towards $\frac{1}{2}$ as well.

We have the following relation:

$$\text{Packet transmission time} = \frac{\text{Packet size}}{\text{Bit rate}}$$

The adversary succeeds to send s bits when $\frac{d_{AP}}{c} + \frac{s}{\text{Bit rate}} \leq T$ with $\frac{d_{AP}}{c}$ the packet traveling time is in ns, this is negligible compared to T in μs. So, we get the relation $s \leq \text{Bit rate} \times T$. For wireless communication, the maximal bit rate is of order 1Gbps and we define $T = 1\mu$s. So the prover can send maximum $s = 1000$ bits to the adversary. So the maximal s is $s = 2^{10}$.

For a noisy communication such that $E_{max} = 5\%n$ and $T_{max} = 0$ with $s = 2^{10}$, if the prover is close to the verifier ($m \leq 18$), pufDB cannot be proven secure against TF-attacks.

If the prover is close to the verifier then he can help the adversary in doing the authentication himself or in giving directly the device to the adversary. So, we can assume that the prover is quite far from the adversary proportionally to the distance allowed. For instance, if we consider that $d_{VP} = 3000$m, $B = 3$m, $T = 1\mu$s and the speed of the light $c = 3.10^8$m.s^{-1} we get $t_B = 10$ns and $m = 20$. For $s = 2^{10}$, we obtain $q_{m,s} = 0.7917$ so the protocol achieves a security level of 2^{-10} in 110 rounds, and 2^{-20} in 307 rounds.

If we can lower T to $T = 100$ns and $t_B = 10$ns then the prover can send at most $s = 2^7$ bits to the adversary and we have security for a noisy communication with $E_{max} = 5\%n$ and $T_{max} = 0$ for $m \geq 15$ which corresponds to $t_{VP} > 71 t_B$.

7 Conclusion

Until pufDB, none of the existing protocol has provided Terrorist Fraud resistance in the plain model without assuming that the adversary would not share his secret, which is not a realistic

assumption. The protocol of Kleber et al. is not secure against Terrorist Fraud attacks. pufDB is an improvement of this protocol. We prove security against Distance Fraud, Mafia Fraud and Distance Hijacking. In the case of a prover at a distance close to B (i.e $m = 1$), the protocol achieves a security of 2^{-10} against these three attacks in 61 rounds. In the case of a prover far from the verifier (i.e $m > 2n - 1$), the protocol can achieve a security of 2^{-20} against these three attacks in 28 rounds. We further prove the security against TF using a reasonable limitations on the number of transmission per round.

The following table presents the performance of the protocol against the different types of attacks and the sufficient number of rounds to reach a level of security of 2^{-10} and 2^{-20}, we consider a noise such that $E_{max} = 5\%n$ and $T_{max} = 0$. In the row of DF with $m = 1$, we also indicate the necessary number of rounds due to the attack in Appendix A. For DF, the bound is given in Section 3. The best MF-attack described in Appendix B reaches the upper bound given in Th. 2. The upper bound for DH given in Section 5 is already the best DH-attack possible. So the number of rounds in the MF and DH rows are necessary and sufficient.

Attack	m	n (security level 2^{-10})	n (security level 2^{-20})
DF	1	[33-37]	[76-114]
	2	32	76
	9	14	34
	$m > 2n - 1$	11	25
MF	all m	13	28
DH	all m	10	21
TF ($s = 2^7$)	26	17	51
	40	14	34
	93	12	28
TF ($s = 2^{10}$)	20	78	259
	30	19	56
	120	12	28

Table 1: pufDB security

We compare with other distance bounding protocols. The parameters in pufDB, SKI [5, 6], FO [14, 33] and DBopt [7] are taken such that the protocols achieve 99% completeness with a noise of 5% as it is described in [7]. (I.e., we adjust $E_{\max}$ to have 99% completeness and obtain different figures than in the previoous table.) If we take the worst case for pufDB (i.e. $m = 1$), pufDB needs more rounds than the previous protocols to achieve the same security level. However, for m large, pufBD is more efficient than SKI and FO to achieve security against DF, DH and MF and it almost reaches the optimal bounds of DBopt.

Protocol	n (security level of 2^{-10})	n (security level of 2^{-20})
SKI	48	91
FO	84	151
DBopt (DB2,DB3)	24	43
pufDB ($m = 1$)	345	474
pufDB ($m > 2n - 1$)	26	45

Table 2: Efficiency of the protocols against DF, DH and MF for completeness 99% under noise 5% ($T_{max} = 0$)

Acknowledgement. The authors thank Negar Kiyvash and Daniel Cullina for their valuable help in the proof of Lemma 6.

A A DF-attack found for $m = 1$

T_{eff} is the counter of the number of time P chooses to answer late. If $T_{eff} < T_{max}$, P can choose to answer late at the round i or to compute each of the 2^m possible C_i and to send for r_i the most probable response to be accepted. For $m = 1$ the probability that the 2 possible challenges C_i output the same bit is $\frac{1}{2}$. If the 2 different C_i output the same bit r_i then the malicious prover can send r_i to the verifier and we bound the winning probability by 1. If the C_i output different results and if $T_{eff} < T_{max}$ then P can choose to send a random response r_i late, the winning probability is 1. P can give at most E_{max} wrong responses.

Let us define ω the number of rounds in which the two challenges C_i give opposite results. Then $\omega - T_{max}$ is the number of rounds when the C_i return opposite results for which P did not choose to use the joker of answering late. So $\omega - T_{max}$ is the number of rounds P has to guess.

Let us define j the number of errors in the case when the possible C_i give opposite results.

For DF, when the adversary is the most powerful ($m = 1$), we have the probability for this attack to succeed equal to:

$$P[\text{win}] = \sum_{\omega=0}^{T_{max}} \binom{n}{\omega} \frac{1}{2^n} + \sum_{\omega=T_{max}+1}^{n} \binom{n}{\omega} \frac{1}{2^n} \left(\frac{1}{2}\right)^{\omega - T_{max}} \sum_{j=0}^{\min(E_{max}, \omega - T_{max})} \binom{\omega - T_{max}}{j}$$

For $E_{max} = 5\%n$, $T_{max} = 0$ we have a security of 2^{-10} in 41 rounds and a security of 2^{-20} in 84 rounds.

B A MF-attack

A is a malicious actor. Before receiving c_i from V, he sends a guessed challenge c'_i to a far away prover P. He receive r'_i from P. If $c'_i = c_i$ then the adversary sends r'_i to V. Pre-asking gives an extra chance to pass a round. But if c_i is incorrectly guessed, any subsequent pre-asking request request will return some useless random bits. So, the best strategy is to pre-ask until $c'_j \neq c_j$, then to continue with the impersonation attack strategy.

We define K the number of rounds passed before that $c'_j \neq c_j$.

We define $k_0 = n - T_{max} - K$ the number of rounds the adversary has to guess.

The adversary succeed if he makes at most E_{max} errors while guessing.

If $K \geq n - E_{max} - T_{max}$ then the adversary wins. The adversary has to make guesses when $K < n - E_{max} - T_{max}$ i.e $k_0 > E_{max}$. ω is the number of error made by the adversary.

So, the probability for the adversary to win with the most powerful attack we known is:

$$\begin{aligned}
P_{max}[\text{win}] &= \frac{1}{2^{n-E_{max}-T_{max}}} + \frac{1}{2^{n+1-T_{max}}} \sum_{k_0=E_{max}+1}^{n-T_{max}} \sum_{\omega=0}^{E_{max}} \binom{k_0}{\omega} \\
&= \frac{1}{2^{n-E_{max}-T_{max}}} + \frac{1}{2^{n+1-T_{max}}} \sum_{w=0}^{E_{max}} \left[\binom{n-T_{max}+1}{\omega+1} - \binom{E_{max}+1}{\omega+1} \right] \\
&= \frac{1}{2^{n-E_{max}-T_{max}}} + \frac{1}{2^{n+1-T_{max}}} \sum_{w=0}^{E_{max}} \binom{n-T_{max}+1}{\omega+1} - \frac{2^{E_{max}+1}}{2^{n+1-T_{max}}} + \frac{1}{2^{n+1-T_{max}}} \\
&= \left(\frac{1}{2}\right)^{n+1-T_{max}} \sum_{w=0}^{E_{max}+1} \binom{n+1-T_{max}}{w}
\end{aligned}$$

This is equal to the bound found in Th. 2. For $E_{max} = 5\%n$, $T_{max} = 0$ we have a security of 2^{-10} in 13 rounds and a security of 2^{-20} in 48 rounds.

References

1. G. Avoine, M.A. Bingöl, S. Kardaş, C. Lauradoux, and B. Martin. A framework for analyzing RFID distance bounding protocols. *Journal of Computer Security*, pages 289-317, 2001.
2. L. Bolotnyy and G. Robins. Physically Unclonable Function-based Security and Privacy in RFID Systems. *IEEE International Conference on Pervasive Computing and Communications*, 2007.
3. L.B. Bolotnyy and G. Robins. Physically Unclonable Function-based Security and Privacy in RFID Systems. *IEEE International Conference on Pervasive Computing and Communications (PerCom 2007)*, pages 211-220, 2007.
4. I. Boureanu, A. Mitrokotsa, and S. Vaudenay. Practical & Provably Secure Distance-Bounding. *The 16th Information Security Conference, Dallas, Texas, USA*, pages 13–15, November 2013.
5. I. Boureanu, A. Mitrokotsa, and S. Vaudenay. Secure & Lightweight Distance-Bounding. *Proceedings of LIGHT- SEC 2013, volume 8162 of LNCS*, pages 97-113, 2013.
6. I. Boureanu, A. Mitrokotsa, and S. Vaudenay. Towards Secure Distance Bounding. *20th anniversary annual Fast Software Encryption (FSE 2013), LNCS*, 2013.
7. I. Boureanu and S. Vaudenay. Optimal Proximity Proofs. *10th International Conference on Information Security and Cryptology (INSCRYPT 2014)*, pages 13-15, December 2014.
8. S. Brands and D. Chaum. Distance-bounding protocols. in: Advances in Cryptology. *Eurocrypt'93*, pages 344-359, 1993.
9. X. Bultel, S. Gambs, D. Gérault, P. Lafourcade, C. Onete, and J-M Robert. Spade: A prover-anonymous and terrorist-fraud resistant distance bounding protocol. *Wisec*, 2016.
10. Cremers C., K.B Rasmussen, B. Schmidt, and S. Čapkun. Distance Hijacking Attacks on Distance Bounding Protocols. *2012 IEEE Symposium on Security and Privacy*, pages 113-127, May 2012.
11. Y. Desmedt. Major security problems with the 'unforgeable' (Feige)-Fiat- Shamir proofs of identity and how to overcome them. *In Proceedings of the 6th worldwide congress on computer and communications security and protection (SecuriCom)*, pages 147-159, March 1988.
12. U. Dürholz, M. Fischlin, M. Kasper, and C. Onete. A formal approach to distance bounding RFID protocols. *Proceedings of the 14th Information Security Conference ISC 2011, LNCS*, pages 47-62, 2011.
13. M. Fischlin and C. Onete. Subtle kinks in distance-bounding: an analysis of prominent protocols. *Proceedings of WISEC 2013*, pages 195-206, 2013.
14. M. Fischlin and C. Onete. Terrorism in distance bounding: Modelling terrorist-fraud resistance. *Proceedings of ACNS 2013, Lecture Notes in Computer Science*, pages 414-431, 2013.
15. K.B. Frikken, M. Blanton, and M.J. Atallah. Robust Authentication Using Physically Unclonable Functions. *ISC 2009 LNCS volume 5737*, pages 262-277, 2009.
16. G. Hancke. A Practical Relay Attack on ISO14443 Proximity Cards. *http://www.cl.cam.ac.uk/gh275/relay.pdf*, 2005.

17. G. Hancke. Distance Bounding for RFID: Effectiveness of Terrorist Fraud. *Conference on RFID-Technologies and Applications RFID-TA'12*, pages 91-96, 2012.
18. G. Hancke and M.G. Kuhn. An RFID Distance Bounding Protocol. *Conference on Security and Privacy for Emerging Areas in Communications Networks SecureComm'05*, pages 67-73, 2005.
19. M. Hlaváč and T. Rosa. A Note on the Relay Attacks on e-passports. The case of Czech e-passports. *Cryptology ePrint Archive, Report 2007/244*, 2007.
20. M. Igier and S. Vaudenay. Distance Bounding based on PUF. *CANS'16*, 2016.
21. S. Kardaş, M.S. Kiraz, M.A. Bingöl, , and H. Demirci. A Novel RFID Distance Bounding Protocol Based on Physically Unclonable Functions. *7th International Workshop, RFIDSec 2011*, pages 78-93, 2011.
22. H. Kılınç and S. Vaudenay. Optimal Distance Bounding with Secure Harware. under submission.
23. C.H. Kim and G. Avoine. RFID Distance Bounding Protocol with Mixed Challenges to Prevent Relay Attacks. *Cryptology and Network Security, 8th International Conference CANS 2009, Kanazawa, Japan, Lecture Notes in Computer Science 5888*, pages 119-133, 2009.
24. C.H. Kim, G. Avoine, F. Koeune, F-X. Standaert, and O. Pereira. The Swiss-Knife RFID Distance Bounding Protocol. *Information Security and Cryptology ICISC'08, Seoul, Korea, Lecture Notes in Computer Science 5461*, pages 98–115, 2009.
25. S. Kleber, R.W. Van Der Heijden, H. Kopp, and F. Kargl. Terrorist Fraud Resistance of Distance Bounding Protocols Employing Physical Unclonable Functions. *IEEE International Conference and Workshops on Networked Systems (NetSys)*, 2015.
26. C. McDiarmid. On the method of bounded differences. *Surveys in Combinatorics, London Math. Soc. Lectures Notes 141, Cambridge Univ. Press, Cambridge 1989*, pages 148-188, 1989.
27. J. Munilla and A. Peinado. Distance Bounding Protocols for RFID Enhanced by Using Void- challenges and Analysis in Noisy Channels. *Wireless Communications and Mobile Computing, vol. 8*, pages 1227-1232, 2008.
28. V. Nikov and M. Vauclair. Yet Another Secure Distance-Bounding Protocol. *Proceedings of SECRYPT'08, Porto, Portugal*, pages 218-221, 2008.
29. U. Rührmair, Sölter. J., and F. Sehnke. On the Foundations of Physical Unclonable Functions. *Cryptology ePrint Archive, Report 2009/277*, 2009.
30. S. Shariati, F. Koeune, and F-X. Standaert. Security Analysis of image-based PUFs for Anti-Counterfeiting. *Communication and Multimedia Security, volume 7394 of Lecture Notes in Computer Science LNCS*, pages 26-38, 2012.
31. D. Singelée and B. Preneel. Distance Bounding in Noisy Environments. *Security and Privacy in Ad-hoc and Sensor Networks ESAS 2007, Lecture Notes in Computer Science 4572*, pages 101-115, 2007.
32. P. Tuyls and L. Batina. RFID-tags for anti-counterfeiting. *Pointcheval, D. (ed.) CT-RSA 2006. LNCS, vol. 3860*, pages 115-131, 2006.
33. S. Vaudenay. On modeling Terrorist Frauds. *Provsec'13, Lecture Notes in Computer Science 8209*, pages 1-20, 2013.
34. S. Vaudenay. On privacy for RFID. *Advances in Cryptology- ASIACRYPT2007, Vol. 4833 of the series Lecture Notes in Computer Science*, pages 68-87, 2015.
35. S. Vaudenay. Sound Proof of Proximity of Knowledge. *Provsec'15, Vol. 9451 of the series Lecture Notes in Computer Science*, pages 105-126, 2015.