Optimal Proximity Proofs Revisited

Handan Kılınç(✉) and Serge Vaudenay

EPFL, Lausanne, Switzerland
handan.kilinc@epfl.ch

Abstract. Distance bounding protocols become important since wireless technologies become more and more common. Therefore, the security of the distance bounding protocol should be carefully analyzed. However, most of the protocols are not secure or their security is proven informally. Recently, Boureanu and Vaudenay defined the common structure which is commonly followed by most of the distance bounding protocols: answers to challenges are accepted if they are correct and on time. They further analyzed the optimal security that we can achieve in this structure and proposed DBopt which reaches the optimal security bounds. In this paper, we define three new structures: when the prover registers the time of a challenge, when the verifier randomizes the sending time of the challenge, and the combined structure. Then, we show the optimal security bounds against distance fraud and mafia fraud which are lower than the bounds showed by Boureanu and Vaudenay for the common structure. Finally, we adapt the DBopt protocol according to our new structures and we get three new distance bounding protocols. All of them are proven formally. In the end, we compare the performance of the new protocols with DBopt and we see that we have a better efficiency. For instance, we can reduce the number of rounds in DB2 (one of the instances of DBopt) from 123 to 5 with the same security.

1 Introduction

Some important applications such as NFC-based payments, RFID access cards in our daily lives provide services according to the user's location. Relay attacks are serious threats against these applications. For instance, if someone makes a payment with a card on a malicious device then the device can relay to a fake card which is paying for something more expensive [13]. Similarly, a malicious person can open a car by relaying the communication between the wireless key and the car.

In [2], the fact that the speed of communication cannot be faster than the speed of light is used to detect relay attacks. Then, Brands and Chaum [7] introduced the notion of distance bounding (DB) protocols where a prover proves that he is close enough to a verifier. Simply, in distance bounding protocols, the verifier determines the proximity of the prover by computing the round

This work was partly sponsored by the ICT COST Action IC1403 Cryptacus in the EU Framework Horizon 2020.

T. Malkin et al. (Eds.): ACNS 2015, LNCS 9092, pp. 478–494, 2015.
DOI: 10.1007/978-3-319-28166-7_23

trip communication time in challenge/response rounds. The proximity proof is disincentive against relay attacks. The literature considers the following threat models:

- Distance Fraud (DF): A malicious prover far away from the verifier tries to convince him that he is close enough.
- Mafia Fraud (MF) [12]: A man-in-the-middle (MiM) adversary between a far away honest prover and a verifier relays or modifies the messages to make the verifier accept.
- Terrorist Fraud (TF) [12]: An adversary tries to make the verifier accept with the help of far away and malicious prover without gaining any advantage to later pass the protocol on his own.
- Impersonation fraud (IF) [1]: An adversary tries to impersonate the prover to the verifier.
- Distance Hijacking (DH) [11]: A far away prover takes advantage of some honest, active provers to make the verifier accept.

Some of the distance bounding protocols [7–9,15,18,20–22] have been broken since either their security were not proven formally or they do not have any security proofs. Amongst existing distance bounding protocols, only the SKI protocol [3–5], the Fischlin-Onete (FO) protocol [14,23] and the DBopt protocol [6] are formally proven to be secure against all above threats.

Boureanu and Vaudenay [6] formalize the threat models and propose a new distance bounding protocol DBopt which has three concrete instances DB1, DB2 and DB3. They give the definition of the "Common Structure" for the distance bounding protocols. A DB protocol in common structure consists of three phases: an initialization phase and a verification phase which do not depend on communication time, and a distance bounding phase between them. The distance bounding phase consists of number of rounds. In each round, the prover responds the challenge of the verifier. The verifier checks if the responses are on time and correct. DBopt follows the common structure and all instances have the security proofs against DF and MF. All but DB3 have a security proof for TF. The common structure is defined by four parameters: the number of rounds n, the minimal number of correct rounds τ, the cardinality num_c of the challenge set, and the cardinality num_r of the response set. The optimal security bounds for DB protocols that follow the common structure are given in [6]. All instances of DBopt have optimal security bounds against MF and all but DB2 have optimal security bounds against DF.

Random delays for the messages (challenges and responses) on both the verifier and the prover side in the distance bounding phase is used for location privacy as discussed in [17,19]. In this paper, we add random delays only on the verifier side and achieve better security bounds.

The contribution of this paper is as follows:

- We define three new structures for distance bounding protocols. Differently than the common structure [6], we suggest to add properties that the prover

measures time like the verifier and the verifier sends challenge in a time that is randomly chosen.
- We show the optimal security bounds for each new structure. Compared to common structure [6], we obtain better security bounds.
- We modify DBopt protocol [6] according to the new structures and have new protocols DBoptSync, DBoptSyncRand and DBoptRand. We prove the security of them against DF, MF and IF (DH and TF resistance are unchanged compared to [6]). We reach the optimal security bounds for DF and MF for all of them in their respective structure.
- We analyse the performance of our new DB protocols and conclude that we have a better efficiency than previous works [3–6,14,23].

2 Definitions and Preliminaries

In this section, we recall the formal model of distance bounding protocols from [6].

Definition 1 (Distance Bounding Protocol). *A (symmetric) distance bounding protocol is a two party probabilistic polynomial time (PPT) protocol and consists of a tuple* $(\mathcal{K}, P, V, B)$*. Here,* $\mathcal{K}$ *is the key domain,* P *is the proving algorithm,* V *is the verifying algorithm where the inputs of* P *and* V *is from* $\mathcal{K}$*, and* B *is the distance bound. Given* $x \in \mathcal{K}$*,* $P(x)$ *and* $V(x)$ *interact with each other. At the end of the protocol, the verifier* $V(x)$ *sends a final message* Out_V*. If* $\mathsf{Out}_V = 1$*, then the verifier accepts. If* $\mathsf{Out}_V = 0$*, then the verifier rejects.*

In a DB protocol, apart from the prover and the verifier, there may exist other participants called adversaries. Each participant has instances and each instance has its own location. **P** denotes the set of instances of the prover, **V** denotes the set of the instances of the verifier and **A** denotes the set of the instances of the other participants.

Instances of an honest prover run the algorithm P denoted by $P(x)$. An instance of a malicious prover runs an arbitrary algorithm denoted by $P^*(x)$.

The verifier is always honest and its instances run the algorithm V denoted by $V(x)$.

The other participants are (without loss of generality) malicious. They may run any algorithm without no initialized key. $\mathcal{A}$ denotes a participant from **A**.

The locations of the participants are elements of a metric space.

Communication and Adversarial Model: The communication and adversarial model of a DB protocol [3] is the following:

DB protocols run in natural communication settings. There is a notion of time, e.g. time-unit, a notion of measurable distance and a location. Besides, timed communication follows the laws of physics, e.g., communication cannot be faster than speed of light.

An adversary can see all messages (whenever they reach him). He can change the destination of a message subject to constraints. Namely, a message sent by

U at time t to V can be corrupted by $\mathcal{A}$ at time t' if $t' + d(\mathcal{A}, V) \leq t + d(U, V)$ where d is a metric that shows the distance between its inputs. In addition, the adversary may have extra technology to correct the noise of the channel while honest participants cannot have it.

In fact, the adversary has very limited action because of the communication speed. For instance if the adversary relays the messages between the far away prover and the verifier, the responses arrive very late. Similarly if the adversary forces the far away prover for any online help, still he cannot succeed to respond correctly and on time. Basically, the adversary cannot break the laws of physics!

Definition 2 (DB Experiment). *An experiment* exp *for a distance bounding protocol with the tuple* $(\mathcal{K}, P, V, B)$ *is a setting* ($\boldsymbol{P, V, A}$) *with several PPT instances of participants, at some locations.*

We denote by $\mathsf{exp}(V)$ a distinguished experiment where we fix a verifier instance $\mathcal{V}$ called the distinguished verifier. Participants that are within a distance of at most B from $\mathcal{V}$ are called close-by participants. Others are called far-away participants.

Definition 3 (Common Structure [6]). *A DB protocol with the common structure based on parameters* $(n, \tau, \mathsf{num}_c, \mathsf{num}_r)$ *has some initialization and verification phases which do not depend on communication times. These phases are seperated by distance bounding phase which consists of* n *rounds of timed challenge/response exchanges. A* ***response is called on time*** *if the elapsed time between sending the challenge (by verifier) and receiving the response (by verifier) (See Fig. 1) is at most* $2B$. *Provers do not measure the time. Challenges and responses are in sets of cardinality* num_c *and* num_r, *respectively.*

When the protocol follows the specified algorithms but messages during the distance bounding phase can be corrupted during transmission, we say that the protocol is τ*-complete if the verifier accepts if and only if at least* τ *rounds have a correct and on-time response.*

In practice, the noise in the communication should be considered. We assume that there is probability of noise p_{noise} in one round of distance bounding phase. Therefore the probability that a number of τ responses are correct and on time in the case of a close-by prover is $\mathsf{Tail}(n, \tau, 1 - p_{\mathsf{noise}})$ where:

$$\mathsf{Tail}(n, \tau, \rho) = \sum_{i=\tau}^{n} \binom{n}{i} \rho^i (1-\rho)^{n-i}$$

Accordingly, the probability to fail is negligible when $\frac{n}{\tau} < 1 - p_{\mathsf{noise}}$ due to the Chernoff-Hoeffding bound [10,16].

We now give security definitions and theorems from [6] that show the optimal security bounds for the DB protocols following the common structure.

Definition 4. ***(α-resistance to Distance Fraud*** *[6]). The distance-bounding protocol* α*-resists to distance fraud if for any distinguished experiment* $\mathsf{exp}(V)$ *where there is no close participant to* $\mathcal{V}$, *the probability that* $\mathcal{V}$ *accepts is bounded by* α.

Theorem 1 ([6]). *A DB protocol following the common structure with parameters* $(n, \tau, \mathsf{num}_c, \mathsf{num}_r)$ *cannot* $\alpha-$*resists to distance fraud for* α *lower than* $\mathsf{Tail}(n, \tau, \max(\frac{1}{\mathsf{num}_c}, \frac{1}{\mathsf{num}_r}))$.

This is the optimal security bound that a DB protocol can reach against distance fraud. The DB1 and DB3 protocols from DBopt [6] reach this bound.

Definition 5 ***(β-secure Distance Bounding Protocol** [6]). We say that a distance-bounding protocol is* β*-secure if for any distinguished experiment* $\mathsf{exp}(V)$ *where the prover is honest, and the prover instances are all far away from* $\mathcal{V}$ *(the distance between the prover instances and* $\mathcal{V}$ *is more than* B*), the probability that* $\mathcal{V}$ *accepts is bounded by* β.

We recall that $\boldsymbol{\beta}$-security captures the threat models MF, MiM and IF [6].

Theorem 2 ([6]). *A DB protocol following the common structure with parameters* $(n, \tau, \mathsf{num}_c, \mathsf{num}_r)$ *cannot be* $\beta-$*secure lower than* $\mathsf{Tail}(n, \tau, \max(\frac{1}{\mathsf{num}_c}, \frac{1}{\mathsf{num}_r}))$.

This is the optimal security bound that a DB protocol can reach against mafia fraud. All instances of DBopt protocols [6] reach this bound.

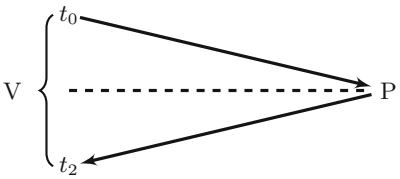

Fig. 1. The time check in the common structure is done by measuring the time difference between the curly parenthesis. t shows the time.

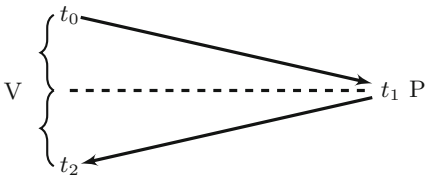

Fig. 2. The time check in the sync structure is done by measuring the time difference between the curly parentheses. t shows the time.

3 Optimal Distance Bounding Protocol with Almost Synchronized Parties

3.1 Definitions and Lemmas

Definition 6 ***(Sync Structure).*** *A DB protocol with the sync structure based on parameters* $(n, \tau, \mathsf{num}_c, \mathsf{num}_r)$ *has some initialization and verification phase which do not depend on communication times. There is an* n*-round distance bounding phase between the initialization and verification phase. The* ***challenge is on time*** *if the elapsed time between sending the challenge (by verifier) and receiving the challenge (by prover) (Corresponds first part in Fig. 2) is at most*

B. The ***response is on time*** *if the elapsed time between sending the response (by prover) and receiving the response (by verifier) (Corresponds second part in Fig. 2) is at most B. Challenges and responses are in sets of cardinality* num_c *and* num_r*, respectively.*

When the protocol follows the specified algorithms but messages during the distance bounding phase can be corrupted during transmission, we say that the protocol is τ*-complete if the verifier accepts if and only if at least* τ *rounds have a correct and on-time* ***response and challenge****.*

The important difference between "Common Structure" and "Sync Structure" is that provers now need to measure time since the verifier needs to check if the challenge arrive on time to the prover.

Lemma 1. *Let* exp *be an experiment,* $\mathcal{V}$ *be a participant and* t_0 *be a time. We consider a simulation* exp_{t_0} *of the experiment in which each participant* U *stops just before time* $t_0 + d(V, U)$*. We denote by* $\mathsf{View}_t^{\mathsf{exp}}(U)$ *and* $\mathsf{View}_t^{\mathsf{exp}_{t_0}}(U)$ *the view of participant* U *at time* t *in* exp *and* exp_{t_0}*, respectively. For any* $t < t_0 + d(V, U)$*,*

$$\mathsf{View}_t^{\mathsf{exp}}(U) = \mathsf{View}_t^{\mathsf{exp}_{t_0}}(U).$$

Proof. We prove by induction on t that for all participant U such that $t < t_0 + d(V, U)$, $\mathsf{View}_t^{\mathsf{exp}}(U) = \mathsf{View}_t^{\mathsf{exp}_{t_0}}(U)$. Clearly this is the case at the beginning of the both experiments. If it is the case at any time less than or equal to $t - 1$, we can now prove it is the case at time t. Let participant U be such that $t < t_0 + d(V, U)$. We know that $\mathsf{View}_{t-1}^{\mathsf{exp}}(U) = \mathsf{View}_{t-1}^{\mathsf{exp}_{t_0}}(U)$. Any incoming message m at time t from a participant U' was sent at time $t' = t - d(U, U')$. We have $t' < t_0 + d(V, U) - d(U, U') \leq t_0 + d(V, U')$. If U' is at a different location than U, we have $t' \leq t - 1$ so we can apply the induction hypothesis. Therefore $\mathsf{View}_{t'}^{\mathsf{exp}}(U') = \mathsf{View}_{t'}^{\mathsf{exp}_{t_0}}(U')$ and so the message m is the same in exp and exp_{t_0}. This applies to all instances at the same location as U, since they locally compute the same messages for each other. Hence, $\mathsf{View}_t^{\mathsf{exp}}(U) = \mathsf{View}_t^{\mathsf{exp}_{t_0}}(U)$. □

Lemma 2. *Given an experiment, if a message* c *is randomly selected with fresh coins by a participant* V *at time* t_0*, any* $\hat{c}$ *received by a participant* U *at time* $t_1 < t_0 + d(U, V)$ *is statistically independent from* c*.*

Proof. We apply Lemma 1. c is not selected at all in exp_{t_0} because V stops just before t_0 in exp_{t_0}. Since $t_1 < t_0 + d(U, V)$, $\hat{c}$ is the same in exp and exp_{t_0}. c is randomly chosen with fresh coins, so $\hat{c}$ is statistically independent from c. □

Theorem 3. *Assuming the time when* $\mathcal{V}$ *sends his challenge can be predicted by the adversary, a* $\tau-$*complete DB protocol following the* ***sync structure*** *with parameters* $(n, \tau, \mathsf{num}_c, \mathsf{num}_r)$ *can not be* β *secure (Definition 5) for* β *lower than* $\mathsf{Tail}(n, \tau, \frac{1}{\mathsf{num}_c.\mathsf{num}_r})$*.*

Remark that this bound is an improvement compared to Theorem 2 in the common structure.

Proof. We consider $\mathcal{V}$, a far-away prover P and a MiM $\mathcal{A}$ with noiseless communication. $\mathcal{A}$ relays the messages between $\mathcal{V}$ and P in the initialization and verification phases which are time insensitive. During the challenge phase, $\mathcal{A}$ should arrange the response and the challenge time. Since P is far-away, he cannot just relay the messages. Therefore he should guess the challenge and the response before receiving them. We denote that the distance between $\mathcal{V}$ and $\mathcal{A}$ by d_1 and the distance between $\mathcal{A}$ and P by d_2. So it can do the following strategy:

No-ask Strategy: $\mathcal{A}$ can guess the response and the challenge and forward them before seeing them so that they arrive on time.

We assume that $\mathcal{A}$ knows the time t_0 that $\mathcal{V}$ sends the challenge c and he chooses a distance $d \leq B$. He guesses the challenge and sends it to P at time $t_0 + d - d_2$ so that P receives it at time t_1 where $t_1 = t_0 + d$. He guesses the response and sends $\mathcal{V}$ at time $t_0 + 2d - d_1$. $\mathcal{V}$ receives the response at time $t_2 = t_1 + d$. Since $t_1 - t_0 = d \leq B$ and $t_2 - t_1 = d \leq B$, the challenge and the response rounds are on time.

As a result, $\mathcal{A}$ can be successful on the verification of the challenge and the response time with no-ask strategy if he guesses both the challenge and response correctly. The probability that he passes the verification for one round is $\frac{1}{\mathsf{num}_c.\mathsf{num}_r}$ and so the probability that the $\mathcal{V}$ accepts $\mathcal{A}$ is $\mathsf{Tail}(n, \tau, \frac{1}{\mathsf{num}_c.\mathsf{num}_r})$. □

Verifier secret: x		**Prover** secret: x
	initialization phase	
pick $L_\mu \in_U \mathcal{L}, N_V \in_U \{0,1\}^{\ell_{\mathsf{nonce}}}$ select $b \in \mathsf{GF}(q)^n$	$\xleftarrow{N_P}$	pick $N_P \in_U \{0,1\}^{\ell_{\mathsf{nonce}}}$
$a = f_x(N_P, N_V, L_\mu, b)$	$\xrightarrow{N_V, L_\mu, b}$	$a = f_x(N_P, N_V, L_\mu, b)$
$x' = L_\mu(x)$		$x' = L_\mu(x)$
	distance bounding phase for $i = 1$ to n	
pick $c_i \in_U \mathsf{GF}(q)$		
start time t_0^i	$\xrightarrow{c_i}$	receive c'_i, save time t_1^i
receive r_i, stop time t_2^i	$\xleftarrow{r'_i}$	$r'_i = \phi_{c'_i}(a_i, x'_i, b_i)$
	verification phase	
receive c'', check $\mathsf{tag} = f_x(N_P, N_V, L_\mu, T, b, c'')$	$\xleftarrow{c', T, \mathsf{tag}}$	$\mathsf{tag} = f_x(N_P, N_V, L_\mu, T, b, c')$
check $\#\{i; c_i = c''_i, r_i$ and t_2^i, t_1^i correct$\} \geq \tau$	$\xrightarrow{\mathsf{Out}_V}$	

Fig. 3. The DBoptSync distance-bounding protocol

3.2 DBoptSync with Synchronized Parties

We propose a new distance bounding protocol DBoptSync described in Fig. 3 which uses the ideas in [6]. The assumption here is that the prover P and the verifier $\mathcal{V}$ have synchronized clocks.

DBoptSync is a symmetric distance bounding protocol in which P and $\mathcal{V}$ share a secret $x \in \mathbb{Z}_2^s$ where s is a security parameter. The notations are the folllowing: n is the number of rounds, ℓ_{tag} is the length of the tag, τ is a threshold, $\mathcal{T}$ is the set of all possible time values, q is a prime power.

As in DBopt, we use the function f_x which maps different codomains depending on the input. $f_x(N_P, N_V, L_\mu, b) \in GF(q)^n$ and $f_x(N_P, N_V, L_\mu, T, b, c) \in GF(q)^{\ell_{\mathsf{tag}}}$. L_μ is a mapping defined from a vector $\mu \in \mathbb{Z}_2^s$ where $L_\mu(x) = (\mu(x), \mu(x), ..., \mu(x))$ and $\mu(x) = \mathsf{map}(\mu.x)$ such that $\mathsf{map} : \mathbb{Z}_2 \rightarrow GF(q)$ is an injection. Here $N_P, N_V \in \{0,1\}^{\ell_{\mathsf{nonce}}}$, $L_\mu \in \mathcal{L}$ where $\mathcal{L}$ includes all possible L_μ mappings, $b, c \in GF(q)^n$ and $T \in \mathcal{T}^n$.

The *initialization phase* of the DBoptSync is the same as in the DBopt protocol [6]. The *distance bounding phase* is almost the same. The difference is that P saves the each time t_1^i that he receives the challenge c_i' from $\mathcal{V}$ at round i and $\mathcal{V}$ saves the times t_0^i and t_2^i that he sends the challenge c_i and he receives response r_i', respectively. In the *verification phase*, the prover sets $T = (t_1^1, t_1^2, ..., t_1^n)$ and $c' = (c_1', c_2', ..., c_n')$ and calculates the tag $f_x(N_P, N_V, L_\mu, T, b, c')$. Then he sends the tag and the verifier does the following:

- He checks if the tag and (c', T) are compatible which means the tag he received is equal to $f_x(N_P, N_V, L_\mu, T, b, c')$. If it is compatible, he does the next step. Otherwise he rejects P.
- $\mathcal{V}$ counts the number of correct rounds. A round is correct if $c_i' = c_i$ and $r_i' = r_i$. If the number of correct rounds are less then τ, he rejects P. Otherwise he continues with the next step.
- $\mathcal{V}$ checks the challenge and response time for each correct round i. The challenge and response time is correct if $t_0^i \leq t_1^i \leq t_2^i$, $t_1^i - t_0^i \leq B$ and $t_2^i - t_1^i \leq B$, respectively. If the number of timely and correct rounds is at least τ, then $\mathcal{V}$ accepts P. Otherwise, he rejects.

We note that the timely condition in DBoptSync implies $t_2^i - t_0^i \leq 2B$, which is the only verification done in DBopt [6]. Therefore, the DBoptSync's timely condition is more restrictive.

The responses are computed depending on the concrete instance of b and ϕ_{c_i}. There are three protocols defined in [6] whose instances are given in Table 1. Hence, DBoptSync has the same instances as well.

Theorem 4 (Security). *Assuming that $\mathcal{V}$ and P are synchronized, the DBoptSync protocol with the selection of b and ϕ as in Table 1 is $\beta-$secure,*

- *(DB1 and DB2) $\beta = \mathsf{Tail}(n, \tau, \frac{1}{q^2}) + \frac{r^2}{2} 2^{-\ell_{\mathsf{nonce}}} + (r+1)\epsilon + r2^{-\ell_{\mathsf{tag}}}$ when f is a (ϵ, K)-circular PRF (See Appendix A).*
- *(DB3) $\beta = \mathsf{Tail}(n, \tau, \frac{1}{q^2}) + \frac{r^2}{2} 2^{-\ell_{\mathsf{nonce}}} + \epsilon + 2^{-\ell_{\mathsf{tag}}}$ when f is a (ϵ, K)-PRF.*

Here, r is the number of honest instances and K is a complexity bound on the experiment. β is negligible for $\frac{\tau}{n} \geq \frac{1}{q^2} + cte$ when r and K are polynomially bounded and ϵ is negligible.

Table 1. Classification of the protocols according the selection of b and ϕ in DBoptSync

Protocol	q	map	b	ϕ_{c_i}
DB1	$q > 2$	$\mathsf{map}(u) \neq 0$	no b used	$\phi_{c_i}(a_i, x_i', b_i) = a_i + c_i x_i'$
DB2	$q = 2$	$\mathsf{map}(u) = u$	Hamming weight $\frac{n}{2}$	$\phi_{c_i}(a_i, x_i', b_i) = a_i + c_i x_i' + c_i b_i$
DB3	$q \geq 2$	no map used	Hamming weight n	$\phi_{c_i}(a_i, x_i', b_i) = a_i + c_i b_i$

If ϵ, $2^{-\ell_{\mathsf{nonce}}}$ and $2^{-\ell_{\mathsf{tag}}}$ are negligible, DB1, DB2 and DB3 are **optimal** for the security according to Theorem 3.

Proof. The proof starts like in [6]. We consider a distinguished experiment $\mathsf{exp}(V)$ with no close-by participant and no adversary and $\mathcal{V}$ accepts with probability p. We consider a game Γ_0 where we simulate $\mathsf{exp}(V)$ and succeed if and only if $\mathcal{V}$ accepts P. So, the success probability of this game is p. We reduce Γ_1, Γ_2 and Γ_3 as in [6].

We reduce Γ_0 to Γ_1 whose success additionally requires that for every (N_P, N_V, L_μ) triplet there is no more than one instance $P(x)$ and one instance $V(x)$ using this triplet. Since $P(x)$ is honest and $P(x)$ and $V(x)$ are selecting N_P and N_V at random, respectively, so the success probability of Γ_1 is at least $p - \frac{r^2}{2} 2^{-\ell_{\mathsf{nonce}}}$.

Γ_2 is the reduction where Γ_1 and its success requires additionally that $\mathcal{V}$ does not accept forged tag. f_x satisfies the circular PRF assumptions (See Appendix A) as shown in [6]. It means that the tag can be forged with probability $\epsilon + 2^{-\ell_{\mathsf{tag}}}$. Therefore the success probability of Γ_2 is at least $p - \frac{r^2}{2} 2^{-\ell_{\mathsf{nonce}}} - r\epsilon - r2^{-\ell_{\mathsf{tag}}}$ (See [6] for the full proof of this step).

Now, in whole game Γ_2, we replace the oracle O_{x,f_x} by $O_{\tilde{x},F}$ and obtain a simplified game Γ_3. Γ_3's requirements for the success is the same with Γ_2. So we have $\mathsf{Pr}_{\Gamma_3}[success] \geq p - \frac{r^2}{2} 2^{-\ell_{\mathsf{nonce}}} - (r+1)\epsilon - r2^{-\ell_{\mathsf{tag}}}$.

We now detail the analysis of Γ_3 which differs from [6]. In Γ_3, P and $\mathcal{V}$ never repeat the nonces and use a random function F to select a. So, the distinguished $\mathcal{V}$ has a single matching P and these two instances pick a at random. Furthermore, acceptance implies that both instances have seen the same L_μ, T, b, c. The acceptance message of $\mathcal{V}$ also depends on the correct and timely response and challenge. In the case that $\mathcal{V}$ accepts P, P has to receive the challenge c on time and $\mathcal{V}$ has to receive the corresponding response r on time for at least τ rounds. Let's denote t_0^i the time when $\mathcal{V}$ sends c_i, t_1^i the time when P receives c_i' and t_2^i is the time when $\mathcal{V}$ receives r_i. Thanks to Lemma 2, the challenge that $P(x)$ receives is independent from the challenge that is sent by $V(x)$, since the challenge c is randomly selected by $V(x)$, the message that $P(x)$ received matches with probability $\frac{1}{q}$.

Similarly, if we exchange the roles of P and $\mathcal{V}$ in Lemma 2 and replace t_0 with t_1^i and t_1 with t_2^i, we can conclude that r that $V(x)$ receives is independent from

the response r'_i that is sent by $P(x)$ as well. The response functions on DB1, DB2 in each round i depends on challenge, a_i and x'_i. In Γ_3, a_i is random in $GF(q)^n$. Since $\phi_{c'_i}(a_i, x'_i, b_i) = a_i + g(c'_i, x'_i, b_i)$ where g is a function (See Table 1 for the details of g) we can assume that a_i is randomly selected in $GF(q)$ just when r'_i is computed. Equivalently, r_i is uniformly selected in $GF(q)$ just before being sent. So, $r_i = r'_i$ with probability $\frac{1}{q}$.

To sum up, we have $p \leq \mathsf{Tail}(n, \tau, \frac{1}{q^2}) + \frac{r^2}{2} 2^{-\ell_{\mathsf{nonce}}} + (r+1)\epsilon + r2^{-\ell_{\mathsf{tag}}}$.

If ϕ and b are as in $DB3$ [6], we lose $\frac{r^2}{2} 2^{-\ell_{\mathsf{nonce}}}$ from Γ_0. In Γ_1, we apply full PRF reduction and lose ϵ to obtain Γ_2 with a random function. We lose $2^{-\ell_{\mathsf{tag}}}$ more to assume that tag is received by $\mathcal{V}$ was not forged in some Γ_3. Γ_3 succeeds with a probability bounded by $\mathsf{Tail}(n, \tau, \frac{1}{q^2})$ because of Lemma 1. In the end, we have $p \leq \mathsf{Tail}(n, \tau, \frac{1}{q^2}) + \frac{r^2}{2} 2^{-\ell_{\mathsf{nonce}}} + \epsilon + 2^{-\ell_{\mathsf{tag}}}$ for DB3. □

3.3 DBoptSync with Unsynchronized Verifier and Prover

DBoptSync assumes that the prover and the verifier have synchronized clocks. In this section, we discuss the problems of having unsynchronized clocks for P and $\mathcal{V}$ in the DBoptSync. Let's say that the time difference between the clocks of the verifier and prover is $|\delta|$[1]. For example, $\mathcal{V}$ has time t on his local clock while P has time $T = t + \delta$ on his local clock. $\mathcal{V}$ sends the challenge at t_0 according to $\mathcal{V}$'s local clock and P receives it at $T_1 = t_0 + d_1 + \delta$ according to P's local clock. Then $\mathcal{V}$ receives the response at $t_2 \geq t_0 + 2d_1$. So $\mathcal{V}$ gets the following result in the verification of timing: $T_1 - t_0 = \delta + d_1$ and $t_2 - T_1 = d_1 - \delta$. If the prover is close, the inequality $|\delta| \leq B - d_1$ should be satisfied so that P passes the protocol.

In addition, unsynchronized honest prover and verifier give advantage to the adversary since he is able to do pre-ask (for $\delta > 0$) and post-ask (for $\delta < 0$). Indeed, if the honest prover is far at a distance up to $B + |\delta|$ and at least $\max(B, |\delta|)$, $\mathcal{A}$ passes the protocol with probability $\mathsf{Tail}(n, \tau, \max(\frac{1}{\mathsf{num}_c}, \frac{1}{\mathsf{num}_r}))$.

Note that $t_2^i - T_1^i \leq B$ and $T_1^i - t_0^i \leq B$ imply that $t_2^i - t_0^i \leq 2B$ which is what is described in DBopt [6]. So, the security result of [6] apply to our protocol even if the clocks are not synchronized.

Pre-ask: $\mathcal{A}$ guesses the challenge before it is released and asks for the response to P on time so that he can later on answer. If P and $\mathcal{V}$ are synchronized, this strategy never works because $\mathcal{A}$ relays the response from P to $\mathcal{V}$ where the distance between them is more than B. However the following happens if P and $\mathcal{V}$ are not synchronized and $\delta > 0$.

We consider $d_1 + d_2 \in [\max(B, |\delta|), B + |\delta|]$. $\mathcal{V}$ sends the challenge c at t_0. $\mathcal{A}$ guesses the challenge $\hat{c}$ and sends it to P at t_A to be determined which is before receiving the challenge from $\mathcal{V}$. P receives $\hat{c}$ at $T_1 = t_A + d_2 + \delta$ that is local time of P. P sends response r and $\mathcal{A}$ relays it and $\mathcal{V}$ receives r at $t_2 = t_A + 2d_2 + d_1$.

[1] If the difference between clocks is not constant it can be still considered as a constant during the protocol since the distance bounding phase takes very short time (order of nanoseconds).

$T_1 - t_0 = t_A + d_2 + \delta - t_0$. By selecting $t_A = t_0 + d_1 - 2\delta$, $T_1 - t_0 = d_1 + d_2 - \delta \in [0, B]$. So the challenge is considered on time.

$t_2 - T_1 = t_A + 2d_2 + d_1 - t_A - d_2 - \delta = d_1 + d_2 - \delta \in [0, B]$. So the response is considered on time.

Post-ask: $\mathcal{A}$ guesses the response at the same time he forwards the challenge to P. If P and $\mathcal{V}$ are synchronized, this strategy never works because $\mathcal{A}$ relays the challenge from $\mathcal{V}$ to P where the distance between them is more than B. However the following happens if P and $\mathcal{V}$ are not synchronized and $\delta < 0$.

We consider $d_1 + d_2 \in [-\delta, B - \delta]$. $\mathcal{V}$ sends the challenge c, then $\mathcal{A}$ relays c and P receives it at $T_1 = t_0 + d_1 + d_2 + \delta$. Without waiting the response from P, $\mathcal{A}$ guesses response and sends it at time t_A. So $\mathcal{V}$ receives it at $t_2 = t_A + d_1$.

$T_1 - t_0 = t_0 + d_1 + d_2 + \delta - t_0 = d_1 + d_2 + \delta \in [0, B]$. So the challenge is on time.

By selecting $t_A = t_0 + d_1 + 2d_2 + 2\delta$, we have $t_2 - T_1 = d_1 + d_2 + \delta \in [0, B]$. So the response is on time.

Therefore, there is an attack when the distance between P and $\mathcal{V}$ is in between $\max(B, |\delta|)$ and $B + |\delta|$.

As a result, we have the security bound of Theorem 4 if the distance between P and $\mathcal{V}$ is more than $B + |\delta|$ even though P and $\mathcal{V}$ are not synchronized. However if P is in the distance between B and $B + |\delta|$, we have the weaker security bound as in Theorem 2.

One of the important problems in DBoptSync with unsynchronized P and $\mathcal{V}$ is correctness, since the close-by P cannot pass the protocol, when $d(P, V) \leq B - |\delta|$. Therefore if the verification fails in DBoptSync, $\mathcal{V}$ can do the time verification of DBopt [6] which is checking if $t_2 - t_0 \leq 2B$, but in this case we have a weaker security which is as in DBopt. We stress that this does not require to restart the protocol. We rather obtain a variant of DBoptSync which Out_V can take 3 possible values: "reject", "DBopt accept", or "DBoptSync accept". Applications can decide if a "DBopt accept" is enough depending on the required security level.

4 Randomizing Sending Time of the Challenge

We think of a new modification to distance bounding protocols that are in either "Common Structure" or "Sync Structure". Before, we assumed that the sending time t_0^i of the challenge for each round i in distance bounding phase was known by the adversary. Now, we suggest a new modification where the verifier randomizes the sending time $t_0^i \in [T, T + \Delta]$ where T and Δ are public and t_0^i is uniformly distributed (as real numbers) so that the exact t_0^i cannot be accurately known by the adversary before seeing the challenge.

4.1 Definitions and Lemmas

Definition 7 (Rand Structure). *A DB protocol with the rand structure based on parameters* $(n, \tau, \mathsf{num}_c, \mathsf{num}_r, \Delta)$ *has the same properties with the common*

structure in Definition 3. Additionally, the verifier chooses randomly a sending time in the interval $[T, T+\Delta]$ for each challenge in the distance bounding phase.

Definition 8 (SyncRand Structure). *A DB protocol with the rand structure based on parameters $(n, \tau, \mathsf{num}_c, \mathsf{num}_r, \Delta)$ has the same properties with the sync structure in Definition 6. Additionally, the verifier chooses randomly a sending time in the interval $[T, T+\Delta]$ for each challenge in the distance bounding phase.*

Theorem 5. *A DB protocol following either the "Rand Structure" or the "SyncRand Structure" with parameters $(n, \tau, \mathsf{num}_c, \mathsf{num}_r, \Delta)$ cannot α-resists to distance fraud (DF) for α lower than* $\mathsf{Tail}(n, \tau, \max(\frac{1}{\mathsf{num}_c}, \frac{1}{\mathsf{num}_r}).\frac{2B}{\Delta})$.

Proof. We construct a DF following the early reply strategy: A malicious prover guesses the challenge c_i or the response r_i before it is emitted, and then already sends the response at time T_1^i (We use capital T since the prover does not have to be synchronized with the verifier). Therefore the prover has to guess proper time T_1^i to send the response because the verifier checks the inequalities $t_2^i - t_0^i \leq 2B$ for the "Rand Structure" and $T_1^i - t_0^i \leq B$ and $t_2^i - T_1^i \leq B$ for the "SyncRand Structure". t_2^i is the time that the verifier receives the response so it depends on the sending time T_1^i of response by the prover. It means that $0 \leq t_2^i - t_0^i = T_1^i + d - t_0^i \leq 2B$ where d is the distance between the prover and the verifier. So we can conclude that if $t_0^i \in [T_1^i + d - 2B, T_1^i + d]$ then P passes i^{th} verification. The probability that it happens is $\frac{2B}{\Delta}$. Once c is received, the prover can deduce t_0^i and use $t_1^i = \frac{t_0^i + t_2^i}{2}$ in the "SyncRand Structure" since verifier needs to know it to check if the response and challenge are on time. Therefore the probability that prover succeeds the round i is $\max(\frac{1}{\mathsf{num}_c}, \frac{1}{\mathsf{num}_r}).\frac{2B}{\Delta}$ since he also have to guess correctly c or r. We can conclude that P succeeds at least τ rounds with probability at least $\mathsf{Tail}(n, \tau, \max(\frac{1}{\mathsf{num}_c}, \frac{1}{\mathsf{num}_r}).\frac{2B}{\Delta})$. □

Note that in the "Rand Structure", there is no change on the optimal β which is given in Theorem 2. As for the "SyncRand Structure", the new bound is as follows.

Theorem 6. *A τ-complete DB protocol following the "SyncRand Structure" with parameters $(n, \tau, \mathsf{num}_c, \mathsf{num}_r, \Delta)$ cannot be β-secure for β lower than* $\mathsf{Tail}(n, \tau, \frac{1}{\mathsf{num}_c.\mathsf{num}_r}.\frac{B}{\Delta})$.

Proof. We consider $\mathcal{V}$, a far away prover P and MiM $\mathcal{A}$ with noiseless communication. As showed in Theorem 3, $\mathcal{A}$ can use No-ask strategy to pass the protocol. Differently, he needs to guess proper time t_A^i to send guessed challenge to P. P receives the challenge from $\mathcal{A}$ at time t_1^i where $t_1^i = t_A^i + d_2$. If $\mathcal{A}$ passes i^{th} round, the following inequality $0 \leq t_1^i - t_0^i \leq B$ should be satisfied. It means that $0 \leq t_A + d_2 - t_0 \leq B$. If t_A satisfies this inequality then t_0 should be in the interval $[t_A + d_2 - B, t_A + d_2]$. The probability that it happens is $\frac{B}{\Delta}$. Therefore the probability that prover succeeds the round i is $\frac{1}{\mathsf{num}_c.\mathsf{num}_r}.\frac{B}{\Delta}$ since he also have to guess correct c and r. We can conclude that P succeeds at least τ rounds with probability at least $\mathsf{Tail}(n, \tau, \frac{1}{\mathsf{num}_c.\mathsf{num}_r}.\frac{B}{\Delta})$. □

As a result of all the structures, "SyncRand Structure" gives the best optimal security bounds for both β-security and α-resistance. See Table 2 for the review of the optimal bounds for all of the structures.

Table 2. The review of optimal security bounds according to defined structures

Structure	DF	MF
Common	$\mathsf{Tail}(n,\tau,\max(\frac{1}{\mathsf{num}_c},\frac{1}{\mathsf{num}_r}))$	$\mathsf{Tail}(n,\tau,\max(\frac{1}{\mathsf{num}_c},\frac{1}{\mathsf{num}_r}))$
Sync	$\mathsf{Tail}(n,\tau,\max(\frac{1}{\mathsf{num}_c},\frac{1}{\mathsf{num}_r}))$	$\mathsf{Tail}(n,\tau,\frac{1}{\mathsf{num}_c}.\frac{1}{\mathsf{num}_r})$
Rand	$\mathsf{Tail}(n,\tau,\max(\frac{1}{\mathsf{num}_c},\frac{1}{\mathsf{num}_r}).\frac{2B}{\Delta})$	$\mathsf{Tail}(n,\tau,\max(\frac{1}{\mathsf{num}_c},\frac{1}{\mathsf{num}_r}))$
SyncRand	$\mathsf{Tail}(n,\tau,\max(\frac{1}{\mathsf{num}_c},\frac{1}{\mathsf{num}_r}).\frac{2B}{\Delta})$	$\mathsf{Tail}(n,\tau,\frac{1}{\mathsf{num}_c}.\frac{1}{\mathsf{num}_r}\frac{B}{\Delta})$

4.2 DBoptSyncRand and DBoptRand with Randomized Sending Time

We construct new distance bounding protocols DBoptSyncRand and DBoptRand. DBoptSyncRand follows the same steps as in DBoptSync and DBoptRand follows the same steps as in DBopt [6]. Differently in both of the protocols, the verifier randomizes the send time $t_0^i \in [T, T+\Delta]$ where T and Δ are public and t_0^i is uniformly distributed (as real numbers) for each round i in the distance bounding phase.

In Sect. 5, we consider $\Delta = 100B$. For instance, $\Delta = 1\mu s$ and $B = 10ns$ (this corresponds to 3 m according to speed of light). n rounds take n μs which is reasonable.

Theorem 7 *(Security). Assuming that $\mathcal{V}$ and P are synchronized, the sending time of the challenge is randomized and the time interval $[T, T+\Delta]$ to send the challenge is public. Then the DBoptSyncRand protocol is $\beta-$secure for*

- *(b and ϕ as in DB1 and DB2 [6]) $\beta = \mathsf{Tail}(n,\tau,\frac{1}{q^2}.\frac{B}{\Delta}) + \frac{r^2}{2}2^{-\ell_{\mathsf{nonce}}} + (r+1)\epsilon + r2^{-\ell_{\mathsf{tag}}}$ when f is a (ϵ, K)-circular PRF [6].*
- *(b and ϕ as in DB3 [6]) $\beta = \mathsf{Tail}(n,\tau,\frac{1}{q^2}.\frac{B}{\Delta}) + \frac{r^2}{2}2^{-\ell_{\mathsf{nonce}}} + \epsilon + 2^{-\ell_{\mathsf{tag}}}$ when f is a (ϵ, K)-PRF.*

Here, r is the number of honest instances of the prover and K is a complexity bound on the experiment and ϕ is response function. β is negligible for $\frac{\tau}{n} \geq \frac{1}{q^2} + cte$ and r and K polynomially bounded and ϵ is negligible.

If ϵ, $2^{-\ell_{\mathsf{nonce}}}$ and $2^{-\ell_{\mathsf{tag}}}$ are negligible, DB1, DB2 and DB3 are **optimal** for the security according to Theorem 6.

Proof. The proof is the same as Theorem 4 until game Γ_3. The success of Γ_3 depends on the correct and timely response and challenge. Lemma 2 shows that the challenge and the response have to be independent so that they arrive on time and these independent response and challenge can be correct with probability $\frac{1}{q^2}$ (See the proof of Theorem 4). Additionally, the independent challenge $\hat{c}$ is

on time when the sending time is randomized, if $\hat{c}$ is sent on proper time. This proper time can be correct with probability $\frac{B}{\Delta}$ as showed in Theorem 6. Therefore the probability of one successful round is $\frac{1}{q^2}.\frac{B}{\Delta}$.

Consequently, success probability Γ_0 is at least $\mathsf{Tail}(n,\tau,\frac{1}{q^2}.\frac{B}{\Delta}) + \frac{r^2}{2}2^{-\ell_{\mathsf{nonce}}} + (r+1)\epsilon + r2^{-\ell_{\mathsf{tag}}}$ for DB1 and DB2. For DB3, it is at least $\mathsf{Tail}(n,\tau,\frac{1}{q^2}.\frac{B}{\Delta}) + \frac{r^2}{2}2^{-\ell_{\mathsf{nonce}}} + \epsilon + 2^{-\ell_{\mathsf{tag}}}$. □

Theorem 8 *(DF-resistance). The DBoptSyncRand and DBoptRand protocols are $\alpha-$resistant to distance fraud for*

- *(DB1 and DB3)* $\alpha = \mathsf{Tail}(n,\tau,\frac{1}{q}.\frac{2B}{\Delta})$.
- *(DB2)* $\alpha = \sum_{\substack{i+j\geq\tau \\ i,j\leq n/2}}^{n} \binom{n/2}{i}(\frac{2B}{\Delta})^i(1-\frac{2B}{\Delta})^{\frac{n}{2}-i}\binom{n/2}{j}(\frac{B}{\Delta})^j(1-\frac{B}{\Delta})^{\frac{n}{2}-j}$.

DB1 and DB3 are **optimal** for the DF-resistance according to Theorem 5, while DB2 cannot reach the optimal bounds for DF.

Proof. We consider distinguished experiment $\mathsf{exp}(V)$ with no close-by participant. Due to the Fundamental Lemma in [6], the response r_i is independent (in the sense of Fundamental Lemma in [6]) from c_i. For DB1 and DB2, r_i is correct with probability $\frac{1}{q}$. Since r_i has to be arrived on time, the proper time has to be chosen. As stated in Theorem 5 the sending time is chosen correctly with probability $\frac{2B}{\Delta}$. So the probability of success in one round i is $\frac{1}{q}.\frac{2B}{\Delta}$.

In DB2, half of the rounds where $x' = b_i$ are correct because of the hamming weight of b. Therefore, the only necessity in these rounds is sending the response in correct time which can be chosen well with probability $\frac{2B}{\Delta}$. For the remaining rounds ($\frac{n}{2}$ rounds), at least $\tau - \frac{n}{2}$ rounds should pass correctly. The correct response is chosen with the probability $\frac{1}{2}$ and correct time with the probability $\frac{2B}{\Delta}$. □

5 Performance

Three new protocols DBoptSync DBoptSyncRand and DBoptRand have different success probabilities for distance fraud and mafia fraud. DBoptSync and DBoptSyncRand have better bound against mafia fraud compared to DBopt while DBoptRand has the same security against mafia fraud with DBopt. In addition, DBoptRand and DBoptSyncRand have the same and better success probability for distance fraud compared to DBopt but DBoptSync is same with DBopt.

Assuming a noise level of $p_{\mathsf{noise}} = 0.05$ and $\frac{B}{\Delta} = 0.01$, we get the results in Tables 3 and 4. We find τ in terms of rounds n such that $\mathsf{Tail}(n,\tau,1-p_{\mathsf{noise}}) \approx 99\,\%$ for $\tau-$completeness. Table 3 shows the required number of rounds for distance fraud i.e. $\alpha \leq s$. Table 4 shows the number of rounds required for the security i.e. $\beta \leq s$. We used Theorems 4, 7 and 8 and theorems in [6] to compute the required number of rounds to achieve security level.

Table 3. Number of required rounds to be secure against distance fraud where s is the security level in DB protocols. The bold protocols improve DBopt

	$s = 2^{-10}$				$s = 2^{-20}$			
	DB1	DB1	DB2	DB3	DB1	DB1	DB2	DB3
	$(q = 3)$	$(q = 4)$			$(q = 3)$	$(q = 4)$		
DBoptSync	14	12	69	24	24	20	123	43
DBoptSyncRand	3	3	2	3	6	6	2	6
DBoptRand	3	3	2	3	6	6	2	6
DBopt	14	12	69	24	24	20	123	43

Table 4. Number of required rounds to be secure against mafia fraud where s is the security level in DB protocols. The bold protocols improve DBopt

	$s = 2^{-10}$			$s = 2^{-20}$		
	DB1	DB1	DB2-DB3	DB1	DB1	DB2-DB3
	$(q = 3)$	$(q = 4)$		$(q = 3)$	$(q = 4)$	
DBoptSync	7	6	12	12	8	20
DBoptSyncRand	3	1	3	5	5	5
DBoptRand	14	12	24	24	20	43
DBopt	14	12	24	24	20	43

As we can see in Tables 3 and 4, we can use DB2 with 5 rounds (instead of 123) in DBoptSyncRand and reach a pretty good security. If synchronized clocks are not realistic, we can see that we have a much better DF-security with DBoptRand with the same number of rounds.

6 Conclusion

We define new structures for DB protocols which are not used before. The first structure is the "Sync Structure" where the prover measures the time as well as the verifier. We modify the DBopt [6] according to sync structure and we get DBoptSync which has better security against mafia fraud. Then we add new modification which is randomizing the sending challenge time to both "Common Structure" and "Sync Structure" and get the second and third structures "Rand Structure" and "SyncRand Structure", respectively. Similarly, we modify the DBopt and DBoptSync protocols based on these structures and get better security bounds against distance fraud for the DBoptSyncRand and DBoptRand protocols and mafia fraud for DBoptSyncRand protocol. We give the optimal security bounds against distance fraud and mafia fraud for all DB protocols that follows the new structures.

A Circular-Keying PRF

The notion of circular-keying in pseudorandom functions introduced in [4,5]. It is necessary to use circular-keying PRF in our protocols to prove security against MiM attacks. Circular-keying PRF has an extra assumption to the PRF $(f_x)_{x\in GF(q)^s}$ to handle reuse of a fixed x outside of a PRF instance f_x.

Definition 9 (Circular PRF [6]). *Let be s, n_1, n_2 and q some parameters. An oracle $O_{\tilde{x},F}$ is defined as $O_{\tilde{x},F}(y, L, A, B) = A \cdot L(\tilde{x}) + B \cdot F(y)$, using dot product over $GF(q)$, given $L : \{0,1\}^s \to GF(q)^{n_1}$ and $F : \{0,1\}^* \to GF(q)^{n_2}$. We assume that L is taken from a set of functions with polynomially bounded representation. Let $(f_x)_{x\in GF(q)^s}$ be a family of functions from $\{0,1\}^*$ to $\{0,1\}^{n_2}$. The family f is a (ϵ, K)-circular-PRF if for any distinguisher having K complexity, if the probability of distinguishing $O_{x,f_x}, x \in \{0,1\}^s$ from $O_{\tilde{x},F}$ is bounded by $\frac{1}{2} + \epsilon$. Additionally, we require two conditions on the list of queries:*

- *for any pair of queries (y, L, A, B) and (y', L', A', B'), if $y = y'$, then $L = L'$.*
- *for any y, if (y, L, A_i, B_i), $i = 1, 2, ..., \ell$ is the list of queries using this value y, then $\forall\ \lambda_1, \lambda_2, ..., \lambda_\ell \in GF(q)$*

$$\sum_{i=1}^{\ell} \lambda_i B_i \Rightarrow \sum_{i=1}^{\ell} \lambda_i A_i = 0$$

over the $GF(q)$-vector space $GF(q)^{n_2}$ and $GF(q)^{n_1}$.

References

1. Avoine, G., Tchamkerten, A.: An efficient distance bounding RFID authentication protocol: balancing false-acceptance rate and memory requirement. In: Samarati, P., Yung, M., Martinelli, F., Ardagna, C.A. (eds.) ISC 2009. LNCS, vol. 5735, pp. 250–261. Springer, Heidelberg (2009)
2. Beth, T., Desmedt, Y.: Identification tokens or: solving the chess grandmaster problem. In: Menezes, A.J., Vanstone, S.A. (eds.) Advances in Cryptology-CRYPTO 1990. LNCS, vol. 537, pp. 169–176. Springer, Heidelberg (1991)
3. Boureanu, I., Mitrokotsa, A., Vaudenay, S.: Secure and lightweight distance-bounding. In: Avoine, G., Kara, O. (eds.) LightSec 2013. LNCS, vol. 8162, pp. 97–113. Springer, Heidelberg (2013)
4. Boureanu, I., Mitrokotsa, A., Vaudenay, S.: Towards secure distance bounding. In: Moriai, S. (ed.) FSE 2013. LNCS, vol. 8424, pp. 55–68. Springer, Heidelberg (2014)
5. Boureanu, I., Mitrokotsa, A., Vaudenay, S.: Practical and Provably Secure Distance-Bounding. IOS Press, Amsterdam (2015)
6. Boureanu, I., Vaudenay, S.: Optimal proximity proofs. In: Lin, D., Yung, M., Zhou, J. (eds.) Inscrypt 2014. LNCS, vol. 8957, pp. 170–190. Springer, Heidelberg (2015)
7. Brands, S., Chaum, D.: Distance bounding protocols. In: Helleseth, T. (ed.) EUROCRYPT 1993. LNCS, vol. 765, pp. 344–359. Springer, Heidelberg (1994)
8. Bussard, L., Bagga, W.: Distance-bounding proof of knowledge to avoid real-time attacks. In: Sasaki, R., Qing, S., Okamoto, E., Yoshiura, H. (eds.) Security and Privacy in the Age of Ubiquitous Computing. IFIP AICT, vol. 181, pp. 223–238. Springer, Heidelberg (2005)

9. Capkun, S., Buttyan, L., Hubaux, J.-P.: Sector: secure tracking of node encounters in multi-hop wireless networks. In: ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN), pp. 21–32 (2003)
10. Chernoff, H.: A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations. Ann. Math. Stat. **23**, 493–507 (1952)
11. Cremers, C., Rasmussen, K.B., Schmidt, B., Capkun, S.: Distance hijacking attacks on distance bounding protocols. In: 2012 IEEE Symposium on Security and Privacy (SP), pp. 113–127. IEEE (2012)
12. Desmedt, Y.: Major security problems with the unforgeable (Feige-) Fiat-Shamir proofs of identity and how to overcome them. In: Congress on Computer and Communication Security and Protection Securicom 1988, pp. 147–159. SEDEP, Paris (1988)
13. Drimer, S., Murdoch, S.J.: Keep your enemies close: distance bounding against smartcard relay attacks. In: USENIX Security (2007)
14. Fischlin, M., Onete, C.: Terrorism in distance bounding: modeling terrorist-fraud resistance. In: Jacobson, M., Locasto, M., Mohassel, P., Safavi-Naini, R. (eds.) ACNS 2013. LNCS, vol. 7954, pp. 414–431. Springer, Heidelberg (2013)
15. Hancke, G.P., Kuhn, M.G.: An RFID distance bounding protocol. In: First International Conference on Security and Privacy for Emerging Areas in Communications Networks, SecureComm 2005, pp. 67–73. IEEE (2005)
16. Hoeffding, W.: Probability inequalities for sums of bounded random variables. J. Am. Stat. Assoc. **58**(301), 13–30 (1963)
17. Mitrokotsa, A., Onete, C., Vaudenay, S.: Location leakage in distance bounding: why location privacy does not work. Comput. Secur. **45**, 199–209 (2014)
18. Munilla, J., Peinado, A.: Distance bounding protocols for RFID enhanced by using void-challenges and analysis in noisy channels. Wirel. Commun. Mob. Comput. **8**(9), 1227–1232 (2008)
19. Rasmussen, K.B., Čapkun, S.: Location privacy of distance bounding protocols. In: Proceedings of the 15th ACM Conference on Computer and Communications Security, pp. 149–160. ACM (2008)
20. Reid, J., Nieto, J.M.G., Tang, T., Senadji, B.: Detecting relay attacks with timing-based protocols. In: Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security, pp. 204–213. ACM (2007)
21. Singelée, D., Preneel, B.: Distance bounding in noisy environments. In: Stajano, F., Meadows, C., Capkun, S., Moore, T. (eds.) ESAS 2007. LNCS, vol. 4572, pp. 101–115. Springer, Heidelberg (2007)
22. Tu, Y.-J., Piramuthu, S.: RFID distance bounding protocols. In: First International EURASIP Workshop on RFID Technology, pp. 67–68 (2007)
23. Vaudenay, S.: On modeling terrorist frauds. In: Susilo, W., Reyhanitabar, R. (eds.) ProvSec 2013. LNCS, vol. 8209, pp. 1–20. Springer, Heidelberg (2013)