Design and Implementation of a Terrorist Fraud Resilient Distance Bounding System

Aanjhan Ranganathan[1], Nils Ole Tippenhauer[1], Boris Škorić[2], Dave Singelée[3], and Srdjan Čapkun[1]

[1] ETH Zurich, 8092 Zurich, Switzerland
firstname.lastname@inf.ethz.ch
[2] TU Eindhoven, Eindhoven, Netherlands
b.skoric@tue.nl
[3] K.U.Leuven ESAT/SCD-COSIC, Leuven, Belgium
Dave.Singelee@esat.kuleuven.be

Abstract. Given the requirements of fast processing and the complexity of RF ranging systems, distance bounding protocols have been challenging to implement so far; only few designs have been proposed and implemented. Currently, the most efficient implementation of distance bounding protocols uses analog processing and enables the prover to receive a message, process it and transmit the reply within 1 ns, two orders of magnitude faster than the most efficient digital implementation. However, even if implementing distance bounding using analog processing clearly provides tighter security guarantees than digital implementations, existing analog implementations do not support resilience against *Terrorist Fraud* attacks; they protect only against *Distance Fraud* and *Mafia Fraud* attacks. We address this problem and propose a new, hybrid digital-analog design that enables the implementation of Terrorist Fraud resilient distance bounding protocols. We introduce a novel attack, which we refer to as the "double read-out" attack and show that our proposed system is also secure against this attack. Our system consists of a prototype prover that provides strong security guarantees: if a dishonest prover performs the Terrorist Fraud attack, it can cheat on its distance bound to the verifier only up to 4.5 m and if it performs Distance Fraud or Mafia Fraud attacks up to 0.41 m. Finally, we show that our system can be used to implement existing (Terrorist Fraud resilient) distance bounding protocols (e.g., the Swiss Knife and Hancke-Kuhn protocol) without requiring protocol modifications.

Keywords: Secure Ranging, Distance Bounding, Terrorist Fraud

1 Introduction

Wireless localization solutions that emerged in the last decade [19] promise to support a broad set of security- and safety-critical applications, including people and asset tracking, emergency and rescue support [9], secure routing [16] and access control [12, 24]. Given the sensitivity of location information in those applications, this information needs to be obtained and/or verified securely.

One of the most prominent problems in the field of secure localization is that of proximity verification: how can one device (the verifier) establish its distance, either exact or as an upper bound to another device (the prover). This problem was first introduced in [4] and prompted a design of a set of distance bounding protocols [29, 30, 14, 25, 20, 26, 5, 6, 21, 22, 27]. Broader deployment of wireless networks and the attacks on proximity-based access control systems (e.g., in cars [10]), routing [15] and payment systems [11] led to an increased interest in the design and implementation of distance bounding protocols [18, 29, 25, 13]. The security of these protocols was mainly analyzed against three types of attacks: Distance Fraud attacks, Mafia Fraud attacks and Terrorist Fraud. Distance bounding protocols were further formally analyzed in a number of works [2, 5, 3].

Distance bounding protocols rely on the exchange of timed challenges and responses between the verifier and the prover. However, given that the prover is not trusted by the verifier and no assumptions can be made about its processing capabilities, the time that the prover spends in processing the verifier's challenge should be negligible compared to the measured round-trip time, which depends on the speed of light. If the verifier would overestimate the prover's processing time (i.e., the prover is able to process signals in a shorter time than expected), the prover would be able to pretend to be closer to the verifier. The challenge in implementing distance bounding protocols is therefore first to implement a prover that is able to receive, process and transmit signals in negligible time.

Although a number of protocols have been proposed, it is not clear if the proposed distance bounding protocols can be implemented with the required tight processing (and therefore security) guarantees or can be integrated within the existing RF ranging systems. For example, almost all distance bounding protocols assume that a prover will be able to receive a single bit of the challenge, XOR it or compare it with some locally stored value, and transmit the response; all within negligible time. XORs and comparisons require digital processing and the most efficient implementation in the open literature that can realize such distance bounding protocols requires 170 ns [28] and thus enables the attacker to cheat on its distance by at most 27 m. An alternative implementation of distance bounding protocols, using analog processing was proposed in [25] enabling signal reception/processing/transmission within 1 ns and thus provided a tight security guarantee of 15 cm. Instead of using XOR or comparison, this design relied on a processing function called Challenge Reflection with Channel Selection (CRCS), which can be implemented using only analog processing techniques. In [13], a design for implementing a secure distance bounding channel for the rapid bit-exchange in a near-field environment was presented. The experimental implementation used improvised wideband pulses and achieved a distance bound of 1 m in the case of Mafia Fraud attacks and 11 m for Distance Frauds.

However, even if implementing distance bounding using analog processing techniques clearly provides tighter security guarantees than digital implementations, existing analog implementations do not support resilience against *Terrorist Fraud* attacks; they are only suited for the prevention of *Distance Fraud* and *Mafia Fraud* attacks. We address this problem and propose a new, hybrid digital-analog design of a distance bounding system called *Switched Challenge Reflector with Carrier Switching* that enables the implementation of Terrorist Fraud resilient distance bounding protocols such as the Swiss Knife Protocol [17]. Our system does not introduce new processing functions at the prover (such as CRCS); instead, it uses the "bit comparison" function that is commonly used in a number of distance bounding protocols including the Hancke-Kuhn protocol [14].

In our proposed design, the verifier transmits challenges on two different carrier frequencies; the switching time synchronized with the prover. Four possible reply channels are created before activating the appropriate reflected carrier frequency. Based on the credentials held by the prover and the carrier frequency of the received challenge, an activation circuity inside the system appropriately enables the reply channel. Analysis of our prototype shows that the verifier can be cheated only up to 4.5 m in the scenario of a Terrorist Fraud attack and further only up to 0.41 m under a Distance or Mafia Fraud attacker model. Given its design, our system can be used to implement existing Terrorist Fraud resilient distance bounding protocols (e.g., the Swiss Knife protocol). Furthermore, it can be used to implement all distance bounding protocols that follow the Hancke-Kuhn construction without requiring any modifications of the protocol.

2 Background

The goal of a distance bounding protocol is that a verifier establishes an upper bound on its distance to a prover. Although many distance bounding protocols were proposed so far [4, 23, 29, 20, 14, 30, 17], they all follow a similar pattern. The protocols consist of either two or three phases. In the first phase, the verifier and the prover agree or commit to the nonces that will be used in the rest of the protocol. In the second phase, also called the rapid bit exchange, the verifier challenges the prover with a number of single-bit challenges to which the prover replies with single-bit replies. The verifier measures the round-trip times of these challenge-reply pairs, based on which the verifier estimates its upper distance bound to the prover. The distance D between the verifier and the prover is calculated using the equation $D = \frac{c.(t_{RTOF} - t_p)}{2}$, where c is the speed of light ($3 \cdot 10^8$ m/s), t_{RTOF} is the round-trip time elapsed and t_p is the processing delay at the prover before responding to the challenge. The final phase of the protocol is used for confirmation and authentication; note that in a number of protocols this last phase is not present.

Traditionally, the security of distance bounding protocols was evaluated by analyzing their resilience against three types of attacks: *Distance Fraud*, *Mafia Fraud* and *Terrorist Fraud* attacks. In a Distance Fraud attack a dishonest prover tries to shorten the distance measured by the verifier (e.g., by sending its replies before receiving the challenges). This type of attack is executed by the dishonest prover alone, without collusion with other (external) parties.

Mafia Fraud attacks, also called relay attacks, were first described by Desmedt [8]. In this type of attack, both the prover and verifier are honest. The external attacker attempts to shorten the distance measured between the honest prover and the verifier by relaying the communications between the entities.

Finally, in the Terrorist Fraud attacks, a dishonest prover collaborates with an external attacker to convince the verifier that he is closer than he really is. All countermeasures to Terrorist Fraud make the assumption that the dishonest prover is unwilling to reveal his long-term (private or secret) key to the attacker that he collaborates with. Possible grounds for this unwillingness are impersonation, i. e., the external attacker can later use the key to impersonate the dishonest prover, and traceability, i. e., the key may later be used to implicate the dishonest prover in performing a Terrorist Fraud attack. Furthermore, from the perspective of the verifier, it is impossible to distinguish between the external attacker and the prover if the attacker knows the long term key of the prover.

Recently, another type of an attack, called the Distance Hijacking attack was introduced [7]. In this attack a dishonest prover convinces the verifier that it is at a distance at which some other honest prover resides, which differs from the actual physical distance of the dishonest prover to the verifier.

2.1 Terrorist Fraud Resilient protocols

Terrorist Fraud resilient protocols preserve the basic structure of distance bounding protocols, but bind the prover's long term secret to the nonces that are exchanged in the protocol. This prevents the prover from simply handing over the nonces to the external attacker without disclosing its long term secret.

We illustrate the operation of these protocols through an example: the *Swiss Knife* protocol. This protocol was proposed by Kim *et al.* in 2009 [17] (see Fig. 1). The protocol assumes that the verifier has a database containing prover identities (ID) and their symmetric keys (x) and that each prover possesses his own identifier and key. The protocol is executed in three phases.

Preparation phase: From its locally generated nonce N^B, a shared secret x and a constant C^B, the prover creates two m-bit strings (R^0 and R^1) using a keyed pseudorandom function f. Disclosing both R^0 and R^1 would immediately reveal m bits of x.

Rapid-bit-exchange phase: In each round i of the rapid-bit-exchange phase, the verifier sends a random single-bit challenge c_i. Upon reception of c'_i, the prover replies with the value taken from R^0_i, if $c'_i = 0$ and from R^1_i, if $c'_i = 1$. c'_i denotes the modification of c_i over the channel either due to an attack or due to transmission errors.

Concluding phase: The prover sends a Message Authentication Code (MAC) computed over the nonces and received challenges. The verifier then makes a number of checks: he tries to find an entry x in his database for which the MAC is valid; he checks if the number of transmission errors in the challenges are not too high; if the number of incorrect responses to correctly received challenges is

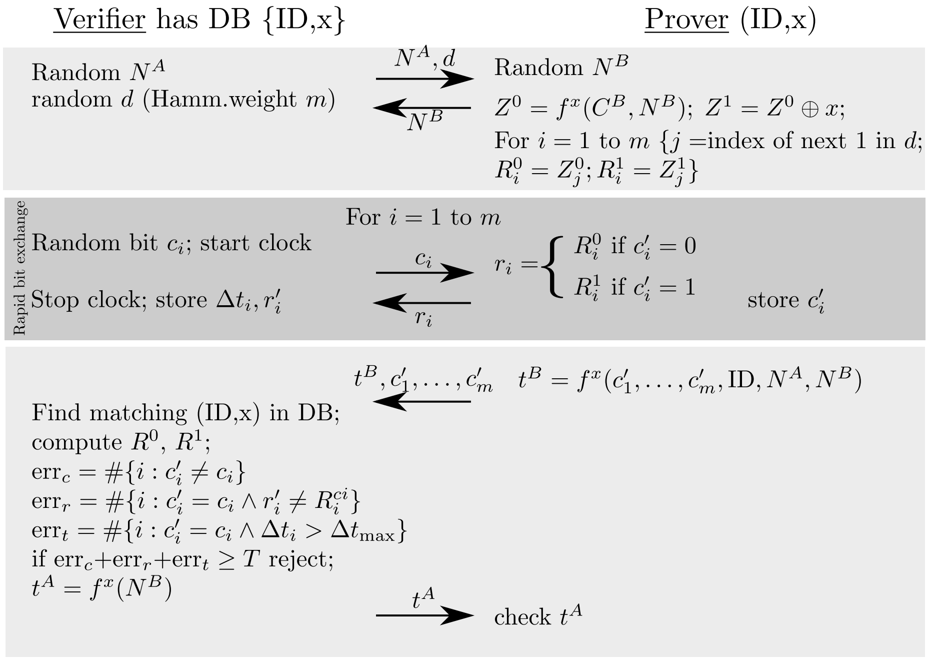
Verifier has DB {ID,x}
Prover (ID,x)
Random N^A
random d (Hamm.weight m)
N^A, d
N^B
Random N^B
$Z^0 = f^x(C^B, N^B)$; $Z^1 = Z^0 \oplus x$;
For $i = 1$ to m {j =index of next 1 in d;
$R_i^0 = Z_j^0; R_i^1 = Z_j^1$}
Rapid bit exchange
For $i = 1$ to m
Random bit c_i; start clock
c_i
$r_i = \begin{cases} R_i^0 \text{ if } c_i' = 0 \\ R_i^1 \text{ if } c_i' = 1 \end{cases}$
Stop clock; store $\Delta t_i, r_i'$
r_i
store c_i'
$t^B, c_1', \ldots, c_m'$
$t^B = f^x(c_1', \ldots, c_m', \mathrm{ID}, N^A, N^B)$
Find matching (ID,x) in DB;
compute R^0, R^1;
$\mathrm{err}_c = \#\{i : c_i' \neq c_i\}$
$\mathrm{err}_r = \#\{i : c_i' = c_i \wedge r_i' \neq R_i^{ci}\}$
$\mathrm{err}_t = \#\{i : c_i' = c_i \wedge \Delta t_i > \Delta t_{\max}\}$
if $\mathrm{err}_c + \mathrm{err}_r + \mathrm{err}_t \geq T$ reject;
$t^A = f^x(N^B)$
t^A
check t^A

Fig. 1. *The Swiss Knife protocol. Picture adapted from [17].*

not too high; and if the responses were sent in time. If all these checks pass, the verifier authenticates itself to the prover by computing a MAC on the prover's nonce N^B.

In this protocol, the values of the registers R^0 and R^1 are bound to the prover's long term secret x. If the prover would like to perform a terrorist attack, it would need to give R^0 and R^1 to the external attacker, thus disclosing x.

2.2 Implementations of Distance Bounding Protocols

The security of distance bounding protocols largely depends on the assumption that the prover's processing time is negligible compared to the measured challenge-response round-trip times. Given that the verifier does not trust the prover and cannot estimate the prover's hardware and processing capabilities, the safest assumption that the verifier can make is that the prover is able to process the challenges and transmit the replies in negligible time. If the verifier overestimates the prover's processing time (i.e., the prover is able to process signals in a shorter time than expected), the prover would be able to pretend to be closer, thus violating the distance bound. The challenge in implementing distance bounding protocols is therefore first to implement a prover that is able to receive, process and transmit signals in negligible time.

Implementations of distance bounding protocols took two distinct directions. One set of solutions focused on digital signal processing, that would enable the implementation of arbitrary processing functions at the prover. In the case of the Swiss Knife protocols, the prover's processing function is the bit comparison (interpretation of the verifier's challenge bit) and the read-out of the register value. This processing function was initially proposed in the Hancke-Kuhn protocol [14]. In the Brands and Chaum's distance bounding protocol, the prover's processing function is an XOR; upon receiving the challenge from the verifier, the prover XORs the challenge bit with a locally stored bit. In [28] Tippenhauer presented an implementation of a digital distance bounding prover that is able to receiver a challenge bit, XOR it with a locally stored bit and transmit the computed response within 170ns.

Another set of solutions focused on analog signal processing. One such solution was proposed in [25] and is based on challenge reflection. The challenge signal sent by the verifier is directly retransmitted by the prover without demodulation and remodulation of the reply signal. This resulted in a small processing delay in the order of nanoseconds. To realize this solution, the authors modified the processing function, such that it can be implemented using solely analog processing, without requiring the prover to digitize the received challenges before replying. The resulting scheme ended up being much more efficient than distance bounding implementations that rely on digital processing, but did not allow the implementation of Terrorist Fraud resilient distance bounding protocols.

This means that, so far, in the space of distance bounding protocol implementations, we could either build efficient implementations, that resist Distance Fraud and Mafia Fraud but not Terrorist Fraud attacks, or less efficient implementations that resist all three types of attacks.

3 Switched Challenge Reflector with Carrier Shifting

As discussed in Section 2, one of the open problems in distance bounding protocol design space is the realization of Terrorist Fraud resilient distance bounding with low processing delay at the prover. Prover designs based on digital signal processing techniques allow implementation of processing functions such as XOR or register read-out based on challenge bits. However, the process of demodulating the received challenge, computing the response (e.g., XOR with a shared secret), modulating and transmitting back the response incurs significant processing delay. This delay allows attackers executing Distance and Mafia Frauds to gain distance in the order of several tens of meters. Although solutions using only analog processing techniques achieved low processing delay, implementing processing functions such as register selections (critical for Terrorist Fraud resilience) gives rise to new attack scenarios. Due to the nature of analog signals and components, such solutions based on register selection are vulnerable to a new attack that we call the "double read-out" attack (detailed in Section 4) which could potentially leak the long-term shared secret. Here we present a hybrid digital-analog solution to this problem, which we call Switched Challenge

Reflector with Carrier Shifting (SCRCS). We show that a prover implementing SCRCS has low processing delay and resists not only Mafia and Distance Frauds but also Terrorist Fraud attacks without allowing any possible "double read-out" attacks.

3.1 Design Overview

In Terrorist Fraud resilient protocols [26, 17, 30], the verifier challenges the prover with randomly selected bits; in each of the m rounds, based on the received challenge bit the prover replies with a bit from one of the two local registers. The prover's processing therefore consists of receiving the challenge bit and then transmitting a bit from one of the registers, selected based on the received challenge bit. We design SCRCS to implement this functionality.

In our system the verifier challenges the prover with a challenge signal $c(t)$; if the verifier wants the prover to respond with a value from register R^0, it transmits a signal on a predefined carrier frequency ω_0 (encoding the challenge bit "0") and if it wants to query R^1, it transmits on the carrier frequency ω_1 (thus encoding the challenge bit "1").

The prover implements switched challenge reflection with carrier shifting. Figure 2 shows the two main building blocks of the prover: (i) Channel Shifter and (ii) Switched Channel Activator. The prover takes as input the challenge signal $c(t)$, which will be at the carrier frequency ω_0 or ω_1; its Channel Shifter component (details in Section 3.2) creates two copies of the received signal: at $\omega_0 + \omega_\Delta$ and $\omega_0 - \omega_\Delta$ or at $\omega_1 + \omega_\Delta$ and $\omega_1 - \omega_\Delta$ where $\omega_\Delta < (\omega_1 - \omega_0)/2$. The two created signals (e.g., the signals at $\omega_0 \pm \omega_\Delta$) are then fed into the Switched Channel Activator circuit which then, depending on the current value of the queried register, outputs $(r(t))$ only one of the two signals (e.g., the signal at $\omega_0 + \omega_\Delta$). The Switched Channel Activator circuit is constructed such that it only allows either the signals at $\omega_0 \pm \omega_\Delta$ or signals at $\omega_1 \pm \omega_\Delta$ but not both simultaneously.

The start of each rapid bit exchange round i.e., the times at which the verifier switches its challenge carrier frequency is synchronized with the prover. This is achieved by the verifier sending an initial preamble defining the exact starting time of the rounds in the rapid-bit exchange phase. This allows the prover to provide an accurate clock to the switched channel activator block (details in Section 3.3) that is responsible for enabling the appropriate reply channel.

Below we discuss our prover design in more detail.

3.2 Channel Shifter

The channel shifter receives the incoming challenge signal $c'(t)$ and applies filters creating four possible reply channels. Figure 3 illustrates in detail the operation of channel shifter module. The received challenges are mixed with an offset frequency ω_Δ ($\omega_\Delta < (\omega_1 - \omega_0)/2$). Based on the carrier frequency on which the challenge is transmitted, the mixer output signal consists of two out of four possible frequency components ($\omega_0 \pm \omega_\Delta$ or $\omega_1 \pm \omega_\Delta$). A set of low-pass and

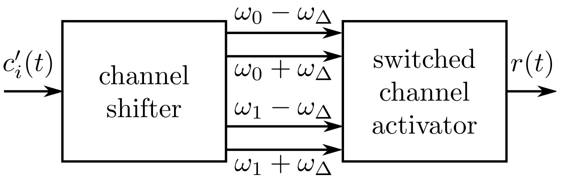
$c'_i(t)$
channel shifter
$\omega_0 - \omega_\Delta$
$\omega_0 + \omega_\Delta$
$\omega_1 - \omega_\Delta$
$\omega_1 + \omega_\Delta$
switched channel activator
$r(t)$

Fig. 2. Overview of the switched challenge reflector with carrier shifting.

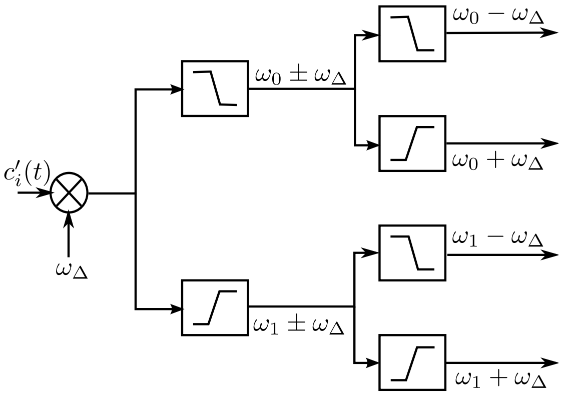
$c'_i(t)$
ω_Δ
$\omega_0 \pm \omega_\Delta$
$\omega_1 \pm \omega_\Delta$
$\omega_0 - \omega_\Delta$
$\omega_0 + \omega_\Delta$
$\omega_1 - \omega_\Delta$
$\omega_1 + \omega_\Delta$

Fig. 3. The channel shifter. The incoming signal $c(t)$ contains the challenges on either carrier frequency ω_0 or ω_1. After mixing $c(t)$ with ω_Δ, the signal is filtered appropriately to generate the four possible response channels: $\omega_0 - \omega_\Delta$,$\omega_0 + \omega_\Delta$,$\omega_1 - \omega_\Delta$,$\omega_1 + \omega_\Delta$.

high-pass filters separate the frequency components resulting in four possible reply channels. These are then fed into the switched channel activator block.

3.3 Switched Channel Activator

The switched channel activator module enables the appropriate reply channel based on amount of energy detected in each of the four signals output by the channel shifter. The module consists of two clocked registers R^0 and R^1, a channel activation circuitry and a memory element to store which channel was activated every round as shown in Figure 4. Both the memory and registers R^0 and R^1 are clocked with the signal CLK, which signals the start of each round in the rapid bit-exchange phase of the protocol. The output $r(t)$ depends on the carrier frequency of $c'(t)$ and the content of R^0 and R^1 during the current round. For example, if the challenge is sent on ω_i, the output is on the channel $\omega_i + (2R^i - 1)\omega_\Delta$. The channel activation circuitry detects the carrier frequency of the challenge signal based on energy detection. Once a channel is activated, it will disable the other channel's activation circuit (i.e. $O_1 = \overline{EN_0}$).

Channel Activation: Figure 5 shows the internals of the channel activation circuitry. The channel activation mechanism ensures that only one of the output channels is activated in each round of the rapid-bit exchange. After this initial

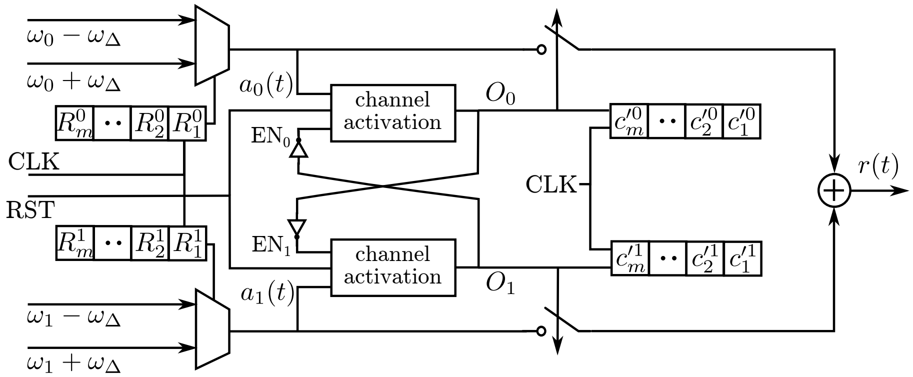
$\omega_0 - \omega_\Delta$
$\omega_0 + \omega_\Delta$
R^0_m R^0_2 R^0_1
CLK
RST
R^1_m R^1_2 R^1_1
$\omega_1 - \omega_\Delta$
$\omega_1 + \omega_\Delta$
$a_0(t)$
$a_1(t)$
channel activation
channel activation
EN_0
EN_1
O_0
O_1
CLK
c'^0_m c'^0_2 c'^0_1
c'^1_m c'^1_2 c'^1_1
$r(t)$

Fig. 4. Switched channel activator. The registers R^0 and R^1 select which two of the four reply channels are used in this round. The channel in which sufficient energy is encountered first gets enabled. After a channel is activated, it stays active until the end of this rapid bit-exchange round while the other channels remain de-activated until the end of this round.

activation, the channel then stays active for the remainder of the current round, reflecting all challenges on this frequency. This selection requires an initial energy and carrier detection, which takes δ_a time in each round of the rapid bit exchange. After δ_a, the correct reply channel is activated and reflects $c'(t)$ with very low delay (incurred by mixing and filtering). The selection of the reply channel is based on the first carrier frequency which contained energy above the threshold T^E. After each round in the rapid bit exchange, all reply channels are deactivated by asserting the RST signal until energy is encountered again in the next round.

Security of Terrorist Fraud resilient protocols relies on the fact that extracting the contents of both the registers R^0 and R^1 compromises the long term shared secret. In fully digital implementation of provers it is not possible to read-out both the register contents simultaneously. However, in our design due to the nature of analog signals and components, there is a possibility of extracting both register contents. We explain this in detail in Section 4. The important role of the channel activation module is to prevent an attacker from executing such *double read-out* attacks by ensuring only one reply channel is active at any given point in time of a particular round.

Synchronization between the verifier and prover: Synchronization between the verifier and the prover is essential for easy verification of the reflected signal later in the concluding phase of the protocol. As discussed in Section 3.1, a preamble sequence transmitted by the verifier is used to establish this synchronization and to generate the switched channel activator's CLK signal. Using this clock, channels are reset at the start of each round of the rapid bit-exchange. It is important to note that the processing time of the preamble does not have strict limitations or security implications. The prover can take some determinis-

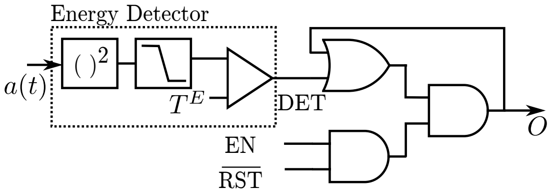

Fig. 5. Internals of channel activation. We obtain a DC component of the squared signal to detect energy in the channel and store the value for this round in a latch-like circuit. The channel activation can be disabled by pulling EN (enable signal) low and is automatically reset at the beginning of each round of the rapid-bit exchange (RST).

tic time δ_p to process the preamble, as long as the challenge data sequence starts at a time greater than δ_p after the preamble.

4 Security Analysis

We investigate the security impact of our proposed distance bounding system with respect to each of the three attack scenarios. In addition, we consider a fourth attack: *double read-out attacks* on Terrorist and Mafia Fraud resilient systems with multiple registers at the prover side.

4.1 Resilience Against Distance Fraud Attacks

In Distance Fraud attacks, the malicious prover is further than D away from the verifier. In order to shorten the measured distance, he will have to send the reply signal $r(t)$ earlier than an honest prover. To achieve this goal, the prover has two options: (a) predict the challenge signal $c(t)$, including the carrier frequency used for each round, or (b) reflect $c(t)$ in with less delay than expected.

The probability to correctly predict the challenge signal $c(t)$ for m rounds of rapid bit exchange depends on the nature of the baseband data signal modulated on the challenge carrier. In the worst case, a constant data signal is modulated on the carrier, which enables the malicious prover to predict it. In this case, our system matches the security analysis of the distance bounding protocol it is used in, as the malicious prover only has to predict which of the registers R^0 and R^1 gets queried in each round. If the baseband signal in $c(t)$ contains data which is unpredictable for the prover, the chance to send a early correct $r(t)$ is strictly smaller than predicted by the overlying protocol. An exact specification depends on the nature of the baseband data signal.

In the following, we analyze the security impact of timing parameters (see Figure 6).

Reflection delay (δ_r): Even if the malicious prover can reflect the challenge with less delay than expected, this will only yield an improvement in the order of nanoseconds. In our implementation, the reflection delay δ_r once the channel

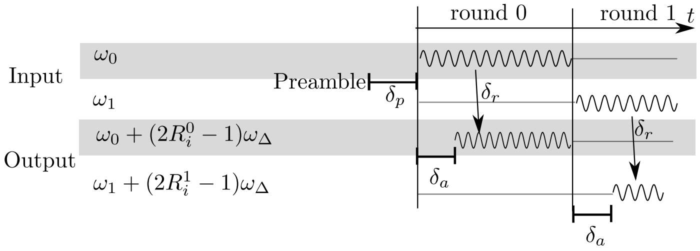

Fig. 6. Timing related variables for challenge reflection : In each round, channel activation adds an initial delay δ_a. After channel activation, the challenges are reflected with a very small delay δ_r. The start time of each round depends on the initial preamble synchronization by the prover.

is activated is around 3 ns. This means the attacker can only gain a distance advantage of 50 cm by reducing δ_r to 0.

Activation delay (δ_a): If the prover is able to shorten δ_a, the correct channel can be activated sooner. Nevertheless, this will not shorten the reflection delay δ_r, and therefore not influence the measured distance for this attack case.

Round start time (δ_p): In our design, we assume that the prover was able to establish the exact start time for each round due to a synchronization preamble sent earlier. This time is required to successfully run the protocol—if the timing is changed, the protocol will most likely fail, instead of returning a wrong distance measure.

If the malicious prover (or external attacker) advances the local round start time of the prover, the channel might be activated by the previous round's carrier frequency. This leads to incorrect reflection of the challenge in 50% of the rounds. If the round start time at the prover is delayed, the prover will not switch to the correct reply channel early enough. Since we have a strict requirement for δ_a, the channel activation delay, this will also cause the protocol to fail. Therefore, changing the round start time does not give an advantage to either malicious prover or external attacker.

4.2 Resilience Against Mafia Fraud Attacks

In the Mafia Fraud, an external attacker close to the verifier tries to impersonate the prover. To successfully impersonate the prover, the attacker can either (a) guess the content of the registers R^0 and R^1 in advance (with probability as predicted in the original protocols), or (b) try to send *early challenges* to the honest prover, to obtain the actual content of registers in advance. Since our system allows the prover to record the received challenges, these can be sent to the verifier in the concluding phase of the protocol later. If the protocol performs this reconciliation on the received challenges, the attacker will have to correctly predict the challenge carrier frequencies used in each round of the rapid-bit-exchange to avoid detection. If no reconciliation phase is supported by the protocol (as in [14]), the attacker's chances are better as discussed in the original protocol.

As the Mafia Fraud is an external attack, the attacker cannot influence the processing delays δ_p, δ_a and δ_r of an involved honest prover. The same reasoning as in the Distance Fraud attack holds good for the round start time. Any modification to the round start time will only result in failure of the protocol execution.

4.3 Resilience Against Terrorist Fraud Attacks

In a Terrorist Fraud attack, an attacker close to the verifier tries to impersonate the prover. The prover will support the attacker, if this does not compromise his long-term secret. In our rapid-bit-exchange scheme, the content of both registers R^0 and R^1 is needed by the attacker to successfully impersonate the prover. But as both register values combined allow the attacker to derive the long-term secret, the prover will not be able to provide these.

Another possibility is for the attacker to early detect the current round's challenge carrier frequency, forward it to the malicious prover and obtain that round's register value. In this case, the long term secret of the malicious prover would not be revealed. To estimate the impact of this attack, we consider a strong attacker and prover with both zero processing time for incoming challenges and messages. In this setting, the attacker could use the channel activation time at the start of each round to forward the current round's challenge carrier frequency. In this setting, the attacker could shorten the measured distance by up to $\delta_a/2$. As this delay is typically short ($< 30\,\text{ns}$ in our implementation), the maximal gain is only in the range of few meters ($\approx 2.5\,\text{m}$ for $30\,\text{ns}$ and instantaneous processing).

Reducing the preamble processing delay δ_p will not yield an advantage to the attacker, while a reduction of the reflection delay can reduce the measured distance as discussed above.

4.4 Double Read-out Attacks

The double read-out attack targets a potential implementation weakness of analog provers with multiple registers. If the attacker manages to simultaneously query (read-out) the values from both registers of the prover, he would be able to reconstruct the prover's long term secret in Terrorist Fraud resilient protocols. In the case of Mafia Fraud resilient protocols, this would allow the attacker to mount a Mafia Fraud attack instead.

Analog implementations e.g., those that would build on CRCS [25] would typically allow a double read-out attack, since they would not prevent the verifier (and the attacker) to transmit the challenge signals on both carrier frequencies simultaneously. To prevent this attack, a digital component is needed (e.g., a channel activation component) that prevents that both register values are transmitted by the prover simultaneously.

More precisely, consider our SCRCS scheme without the channel activation part, i.e. we assume that only the challenge signal and the values of R^0 or R^1 are used to determine the reply channel. In this setting, the attacker could craft a challenge signal which alternates between two challenge carrier frequencies within each round of the rapid bit-exchange and obtain the content of both registers, allowing him to derive the prover's long term secret. Although this attack will most likely be detected by challenge reconciliation in the concluding phase (the MAC'ed c' sent by the prover), the long term secret would still be revealed to the attacker.

In our system, this attack is prevented by the channel activation circuit—this circuit will only allow one register to be read in each round (see Figure 4 and Figure 5). To show that both registers can never be read at the same round, we first show that signal O_i, once activated, can only be deactivated by $\overline{\mathrm{RST}}$. In Boolean logic, we can write $O_i = (\mathrm{DET}_i \vee O_i) \wedge \overline{\mathrm{RST}} \wedge \mathrm{EN}_i$, with $\vee$ as boolean OR and $\wedge$ as AND. Therefore, once O_i is high, it only transitions to false (low) if either $\overline{\mathrm{RST}}$ or EN_i are low. Using $j = |i-1|$ we can write $\neg\mathrm{EN}_i = O_j$. Therefore, once O_i is true (high) and assuming that $\overline{\mathrm{RST}}$ is high, O_i can only turn false if O_j is also true. Using the equation above, one can write $\mathrm{EN}_i = \neg[(\mathrm{DET}_j \vee O_j) \wedge \overline{\mathrm{RST}} \wedge \mathrm{EN}_j]$. Since O_i is true and $\mathrm{EN}_j = \neg O_i$, O_j will always return false. Summarizing, this result shows that a channel can only be deactivated if both channels are true, which cannot happen once one channel is activated. Therefore, both registers cannot be read in the same round.

In addition, our design also prevents unintentional double read-out by the verifier, which might occur if the round start timing of the prover is not aligned well with the verifier. As discussed above, our channel activation will cause the protocol to fail in this case, instead of unintentionally revealing the long-term secret of the prover.

5 Implementation and Analysis

In this section we describe our prototype implementation of the prover and the results of our experiments. We implement our design using commercially available RF modules [1]. The analog components of the prover implementing the switched challenge reflection with carrier shifting is shown in Figure 7. The two carrier frequencies $\omega_0 = 3.5\,\mathrm{GHz}$ and $\omega_1 = 5\,\mathrm{GHz}$ used for transmitting the challenge signal $c(t)$ are generated using function generators and given as input to the prover.

5.1 Channel Shifter

As described in Section 3.2 the channel shifter is implemented using a mixer and six filters (3 low-pass and 3 high-pass). In Figure 7, components 1–4d constitute the channel shifter module. The received signal is amplified and mixed (2) with an intermediate frequency $\omega_\Delta = 500\,\mathrm{MHz}$ generated by a voltage controlled oscillator (1).

Depending on the received carrier frequency (ω_0 or ω_1), the mixer output contains either the frequency components $\omega_0 \pm \omega_\Delta$ or $\omega_1 \pm \omega_\Delta$. This signal now

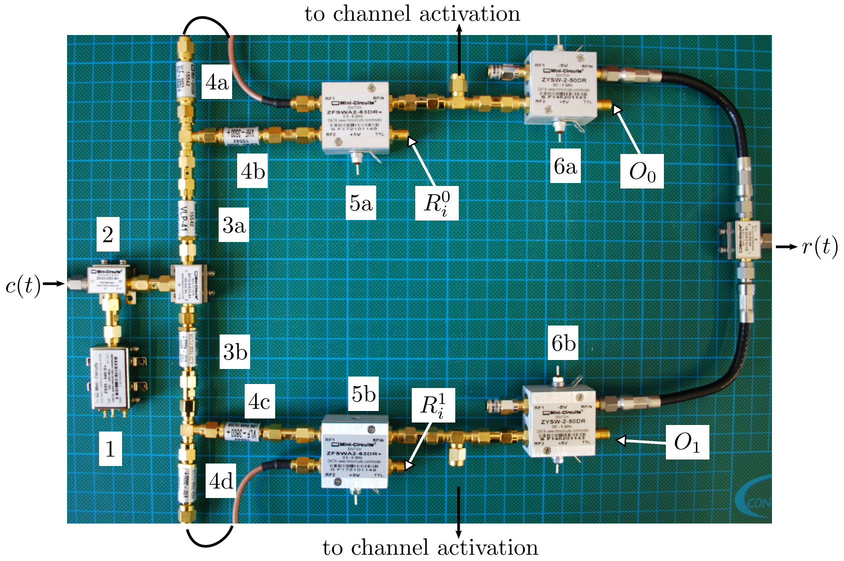

Fig. 7. Experimental Setup: 1: voltage controlled oscillator; 2: mixer; 3a,3b, 4a, 4b, 4c, 4d: filters that constitutes the channel shifter module; 5a, 5b: switches whose output depends on the contents of registers R_i^0 and R_i^1; 6a, 6b: switches that activate the reply channel based on the channel activation circuit outputs (O_0, O_1).

passes through the combination of low-pass and high-pass filters separating the signal into four possible reply channels. For example, if $c(t)$ was transmitted on ω_0, the filters 3a, 4a and 4b (see Figure 7) create the signals with frequency components $\omega_0 + \omega_\Delta$ and $\omega_0 - \omega_\Delta$. Similarly for ω_1, filters 3b, 4c and 4d output $\omega_1 + \omega_\Delta$ and $\omega_1 - \omega_\Delta$. These shifted signals are then fed to the switched channel activator block.

5.2 Channel Activation

The channel activation circuitry constitutes an important part of the prover design to prevent double read-out attacks, as explained in Section 4. The circuit is implemented using a mixer squaring the signal followed by a low-pass filter and a switch. The output of the low-pass filter is the control voltage for the switch. The switch, with one input connected to 5 V and the other grounded acts as a threshold detector whose output is a logic high when its control voltage is above T^E.

We measured the time delay of the channel activation circuitry from the moment the signal is available for energy detection (output of switches 5a, 5b) until the channel is actually activated or deactivated (depends on control signals O_0, O_1 to switches 6a, 6b). Figure 8 shows the control voltage V_{ctrl} and the channel signal. We can see that the switching delay δ_a is approximately 30 ns. As discussed in Section 4 the delay δ_a does not have any security implications in the scenarios

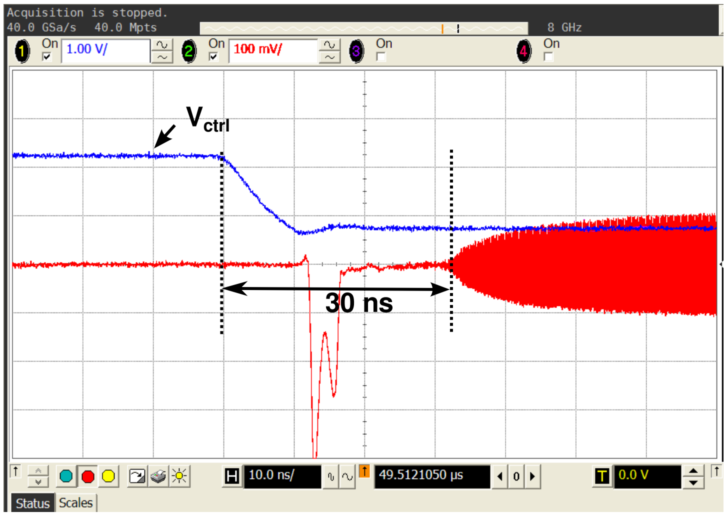
Acquisition is stopped.
40.0 GSa/s 40.0 Mpts

Fig. 8. Delay in switching channels.

of Distance and Mafia Frauds. In the case of Terrorist Fraud an attacker can shorten the distance only up to 4.5 m for $\delta_a = 30$ ns.

5.3 Challenge Reflection Delay

The time taken by the prover to process and reflect back the challenge (δ_r) directly impacts the maximum distance advantage an attacker gains as discussed in Section 4. The challenge signal $c(t)$ is pulse modulated using a 2 μs pulse in order to capture and estimate the delay more accurately. The challenge is processed by the prover circuit, and the delay is estimated by tapping into the signal at the circuit's input and output. An oscilloscope with high sampling rate of 40 GSa/s is used to visualize the delay of the signals. Figure 9 shows both input challenge signal and the prover output with a delay of approximately 2.75 ns. This implies that a dishonest prover can gain a maximum distance of 0.41 m by implementing SCRCS with 0 ns delay. The measured delay is independent of the carrier frequency on which the challenge is transmitted and same for both the carrier frequencies (ω_0 and ω_1).
Table 1 summarizes all the delays and the attack scenarios in which they are applicable. It is important to note that these delays would be further reduced by implementing the system as an integrated circuit.

6 Summary

In this paper, we designed and implemented a distance bounding system that is resilient to the three well-known distance modification attacks: Distance, Mafia

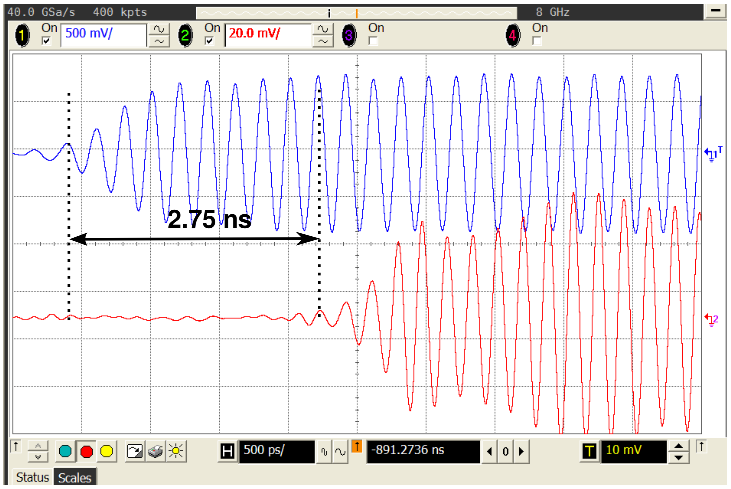

Fig. 9. Prover path delay: The total delay incurred due to mixing, filtering and channel activation switch is estimated to be 2.75ns.

Delay	Max. distance gained	Attack Scenario
$\delta_r = 2.75\,\text{ns}$	0.41 m	DF, MF and TF
$\delta_a = 30\,\text{ns}$	4.5 m	TF
δ_p	-NA-	-NA-

Table 1. Summary of prover delays and the attack scenarios under which they are applicable. Reducing or enlarging round start time δ_p would only cause the protocol to fail.

and Terrorist Frauds. Our mixed digital-analog realization allows challenge processing delays of the order of few nanoseconds, thereby limiting the maximum distance an attacker can cheat on. To the best of our knowledge, this is the first implementation of a distance bounding system that is secure against all the three forms of attacks, while having a low processing delay. We introduced a new attack called the “double read-out” attack and showed how our proposed system is secure against it.

With the example of the Swiss Knife protocol, we illustrated how our system design allows implementation of existing Terrorist Fraud resilient protocols and also other distance bounding protocols that are based on the Hancke-Kuhn construction model. We conclude from the delay measurements of our prover prototype that the attacker will be able to decrease distance by not more than 4.5 m in the Terrorist Fraud scenario. This was derived from the processing delay of 2.75 ns and delay incurred during channel activation. This bound further reduced to 0.41 m for the Distance and Mafia Fraud cases. We plan to explore realizing a complete prototype system including the verifier and analyze its security and performance under different real-world environments and applications.

7 Acknowledgements

This work was funded by the Swiss National Science Foundation (SNSF) under the grant 200020_129605 and NCCR-MICS. The work of Dave Singelée is supported in part by the IAP Programme P6/26 BCRYPT of the Belgian State, by the European Commission under contract number ICT-2007-216676 ECRYPT NoE phase II, by the Flemish IWT SBO project MobCom, and by the Research Council K.U.Leuven: GOA TENSE.

References

1. Mini-Circuits, *www.minicircuits.com*
2. Avoine, G., Bingöl, M.A., Kardaş, S., Lauradoux, C., Martin, B.: A framework for analyzing RFID distance bounding protocols. J. Comput. Secur. 19(2), 289–317 (Apr 2011)
3. Basin, D., Capkun, S., Schaller, P., Schmidt, B.: Lets Get Physical: Models and Methods for Real-World Security Protocols. In: Theorem Proving in Higher Order Logics, Lecture Notes in Computer Science, vol. 5674, pp. 1–22. Springer Berlin / Heidelberg (Aug 2009)
4. Brands, S., Chaum, D.: Distance-bounding protocols. In: Workshop on the theory and application of cryptographic techniques on Advances in cryptology. pp. 344–359. EUROCRYPT '93, Springer-Verlag New York, Inc. (May 1993)
5. Bussard, L., Bagga, W.: Distance-Bounding Proof of Knowledge to Avoid Real-Time Attacks. In: Proceedings of 20th International Conference on Security and Privacy in the Age of Ubiquitous Computing. pp. 223–238 (May 2005)
6. Capkun, S., Buttyn, L., Hubaux, J.P.: Sector: secure tracking of node encounters in multi-hop wireless networks. In: Workshop on Security of Ad Hoc and Sensor Networks (SASN). pp. 21–32. ACM (Oct 2003)
7. Cremers, C., Rasmussen, K.B., Schmidt, B., Capkun, S.: Distance Hijacking Attacks on Distance Bounding Protocols. In: Proceedings of the 33rd IEEE Symposium on Security and Privacy (May 2012)
8. Desmedt, Y., Goutier, C., Bengio, S.: Special Uses and Abuses of the Fiat-Shamir Passport Protocol. In: CRYPTO. pp. 21–39 (Aug 1987)
9. Fischer, C., Gellersen, H.: Location and Navigation Support for Emergency Responders: A Survey. IEEE Pervasive Computing 9, 38–47 (Jan 2010)
10. Francillon, A., Danev, B., Ĉapkun, S.: Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars. In: Proceedings of the 18th Annual Network and Distributed System Security Symposium. The Internet Society (Feb 2011)
11. Francis, L., Hancke, G., Mayes, K., Markantonakis, K.: On the security issues of NFC enabled mobile phones. International Journal of Internet Technology and Secured Transactions 2 (Dec 2010)
12. Gupta, S.K.S., Mukherjee, T., Venkatasubramanian, K., Taylor, T.B.: Proximity Based Access Control in Smart-Emergency Departments. In: Proceedings of the 4th Annual IEEE International Conference on Pervasive Computing and Communications Workshops. pp. 512–516 (Mar 2006)
13. Hancke, G.P.: Design of a secure distance-bounding channel for RFID. J. Netw. Comput. Appl. 34(3), 877–887 (May 2011)

14. Hancke, G.P., Kuhn, M.G.: An RFID distance bounding protocol. In: Proceedings of the 1st International Conference on Security and Privacy for Emerging Areas in Communication Networks. pp. 67–73 (Sep 2005)
15. Hu, Y.C., Perrig, A., Johnson, D.B.: Packet leashes: A defense against wormhole attacks in wireless networks. In: INFOCOM (2003)
16. Hu, Y.C., Perrig, A., Johnson, D.B.: Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks. Wireless Networks 11(1-2), 21–38 (2005)
17. Kim, C.H., Avoine, G., Koeune, F., Standaert, F.X., Pereira, O.: Information security and cryptology — icisc 2008. chap. The Swiss-Knife RFID Distance Bounding Protocol, pp. 98–115. Springer-Verlag, Berlin, Heidelberg (2009)
18. Kuhn, M., Luecken, H., Tippenhauer, N.O.: UWB Impulse Radio Based Distance Bounding. In: Proceedings of the 7th Workshop on Positioning, Navigation and Communication. pp. 28–37 (Mar 2010)
19. Liu, H., Darabi, H., Banerjee, P., Liu, J.: Survey of Wireless Indoor Positioning Techniques and Systems. IEEE Transactions on Systems, Man, and Cybernetics 37(6), 1067–1080 (Nov 2007)
20. Munilla, J., Ortiz, A., Peinado, A.: Distance bounding protocols with void-challenges for RFID (2006), printed handout at the Workshop on RFID Security (RFIDSec)
21. Peris-Lopez, P., Castro, J.C.H., Estévez-Tapiador, J.M., van der Lubbe, J.C.A.: Shedding Some Light on RFID Distance Bounding Protocols and Terrorist Attacks. CoRR abs/0906.4618 (2009)
22. Peris-Lopez, P., Castro, J.C.H., Estévez-Tapiador, J.M., Palomar, E., van der Lubbe, J.C.A.: Cryptographic puzzles and distance-bounding protocols: Practical tools for RFID security. In: IEEE International Conference on RFID. pp. 45–52 (Apr 2010)
23. Poturalski, M., Flury, M., Papadimitratos, P., Hubaux, J.P., Boudec, J.Y.L.: Distance Bounding with IEEE 802.15.4a: Attacks and Countermeasures. IEEE Transactions on Wireless Communications 10(4), 1334–1344 (Apr 2011)
24. Rasmussen, K.B., Castelluccia, C., Heydt-Benjamin, T.S., Čapkun, S.: Proximity-based Access Control for Implantable Medical Devices. In: Proceedings of the 16th ACM conference on Computer and Communications Security. pp. 410–419. ACM (Nov 2009)
25. Rasmussen, K.B., Čapkun, S.: Realization of RF Distance Bounding. In: Proceedings of the 19th USENIX Security Symposium. pp. 389–402 (Aug 2010)
26. 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 (Mar 2007)
27. Singelée, D., Preneel, B.: Distance bounding in noisy environments. In: Proceedings of the 4th European conference on Security and privacy in ad-hoc and sensor networks. pp. 101–115. Springer-Verlag, Berlin, Heidelberg (Jul 2007)
28. Tippenhauer, N.O.: Physical-Layer Security Aspects of Wireless Localization. Ph.D. thesis, ETH Zurich, Switzerland (2012), draft version
29. Tippenhauer, N.O., Čapkun, S.: ID-based Secure Distance Bounding and Localization. In: Proceedings of the 14th European Conference on Research in Computer Security. pp. 621–636. Springer-Verlag, Berlin, Heidelberg (Sep 2009)
30. Tu, Y.J., Piramuthu, S.: RFID Distance Bounding Protocols. In: First International EURASIP Workshop on RFID Technology. Vienna, Austria (September 2007)