SecDEv: Secure Distance Evaluation in Wireless Networks

Gianluca Dini, Francesco Giurlanda, Pericle Perazzo
Dept. of Information Engineering
University of Pisa
Email: [name.surname]@iet.unipi.it

***Abstract*—The problem of measuring the distance between two electronic devices in the presence of an adversary is still open. Existing approaches based on *distance-bounding protocols* are subject to *enlargement attacks* that cause the target to be perceived farther than it actually is. Enlargement attacks represent a new challenge for the research field of secure localization. The contribution of this paper is twofold. First, we propose SecDEv, a secure distance-bounding protocol for wireless channels that withstands enlargement attacks based on jam-and-replay. By leveraging on the characteristics of radio frequency signals, SecDEv establishes a *security horizon* within which a distance is correctly measured and a jam-and-replay attack is detected. Second, we show how SecDEv improves the scalability of secure positioning techniques.**

***Keywords*-secure localization; secure positioning; distance-bounding protocols; distance enlargement attacks**

I. Introduction

The measurement of the distance between two electronic devices is crucial for many practical applications. Many techniques have been proposed over the years [1]. All these techniques fail in the presence of an adversary that wants to disrupt the distance measurement process. Even the well-known and widespread civilian Global Positioning System (GPS) is extremely fragile in adversarial scenarios [2]. Secure location estimation has a plethora of applications including coordination of autonomous guided vehicles [3] and geographical routing [4]. For all these applications, an insecure distance or position estimation could produce security problems such as unauthorized accesses, denial of service, thefts, integrity disruption with possible safety implications and intentional disasters.

Desmedt [5] first introduced the problem of secure location verification and showed that it cannot be solved by solely using cryptography. Brands and Chaum [6] proposed the first *secure distance-bounding* protocol. Since then, many variants have been proposed in the literature [7], [8]. These protocols leverage on both the unforgeability of authenticated messages and the upper bound of the communication speed that is the speed of light. They prevent *distance reduction*, i.e., an adversary cannot make a device appear closer than it really is. The resistance against distance reduction is an important requirement for all the application scenarios involving secure proximity verification [9], [10], [8]. A common example is the problem of proximity-based access control. Let us suppose an RFId card performing an authentication protocol with a reader. If the card correctly performs the protocol, the reader will open a door of a building. An adversary can trick the system by establishing a relay link between the reader and a far away legitimate card, owned by an unaware user. The card correctly performs the authentication protocol via the relay link, and the reader opens the entrance. This attack is known as *mafia fraud*. Along with the correctness of the authentication, the reader has to check even that the card is within a security distance. However, if such a distance measurement is made with insecure methods, the adversary can still break the system. In particular she can perform a distance reduction attack to deceive the reader into believing that the far away card is in the proximity.

The relevance of the secure proximity verification eclipsed the dual problem: the *distance enlargement* attack. By this attack, an adversary makes a device appear farther than it really is. The resistance against both reduction and enlargement attacks is important whenever we want to securely estimate a distance, rather than a proximity. Let us suppose a distributed system that monitors the movement of autonomous guided vehicles. The system relies on distance information to avoid collisions between vehicles. An example of such systems is in [3]. If an adversary is able to make a distance appear larger than it really is, the system could not take collision-avoidance countermeasures in time. This could cause collisions between vehicles, and consequent loss of money and safety threats. Secure distance estimations are extremely useful in trilateration techniques too. These techniques use the distances measurements from at least three anchor nodes, whose positions are known, to estimate the position of a fourth node. If an adversary can enlarge one or more distance measurements, she is able to disrupt the whole positioning process.

In this paper we propose SECure Distance EValuation (SecDEv), a distance-bounding protocol able to resist to enlargement attacks based on jam-and-replay tactics [11], [12], [13]. SecDEv exploits the characteristics of wireless signals to establish a *security horizon* within which a distance can be correctly evaluated (besides measurement

errors) and any adversarial attempt to play a jam-and-replay attack is detected. We also show how SecDEv improves the scalability of secure positioning techniques in terms of number of anchor nodes.

The remainder of this paper is organized as follows. In Section II we present related works. In Section III we introduce a reference distance-bounding protocol. In Section IV we define the threat model. In Section V we introduce SecDEv as an improvement of the reference distance bounding. In Section VI we show how SecDEv improves the performance of secure positioning techniques. Finally, we draw our conclusions in Section VII.

II. Related Works

Secure localization has a vast applicability in many technological scenarios, but it has showed to be a nontrivial problem. The silver bullet is yet to be found.

Brands and Chaum [6] proposed distance-bounding protocols, in which a *verifier* node measures the distance of a *prover* node. Distance-bounding protocols do not determine the actual distance, but rather a secure upper bound on it. In this way, the actual distance is assured to be shorter or equal to the measured one, even in presence of an adversary. These protocols were created to assure the physical proximity between two devices, and consequently to contrast *mafia fraud* attack [5].

Hancke and Kuhn [8] fitted distance bounding protocols for RFId tags. Their proposal deals with a variety of practical problems such scarce resources availability, channel noise and untrusted external clock source. Though extensions for RFId's are possible, we focus on more resourceful devices. We assume the clock source is internal and trusted and the channel noise is corrected by FEC techniques.

Clulow et al. [14] focused on a wide variety of low-level attacks, which leverage on packet latencies (e.g. preambles, trailers, etc.) and symbols' modulations. PHY-layer preambles are sent before the cryptographic quantities, in order to permit the receiver to synchronize itself to the sender's clock. The preamble of the response is fixed and does not depend on the content of the challenge. A dishonest prover could thus anticipate the transmission of the response preamble to reduce the measured distance. To deal with this problem, Rasmussen and Čapkun [15] proposed full-duplex distance bounding protocols, in which the challenge and the response are transmitted on separate channels. The prover receives the challenge and meanwhile transmits the response. In this way, a dishonest prover cannot anticipate the transmission of the response, without having to guess the payload. In the present paper, we assume the prover to be honest. This permits us to simplify our reference distance-bounding protocol (cfr. Section III). In particular we use a single channel in a half-duplex fashion.

Flury et al. [10] and, more in depth, Poturalski et al. [16] analyze the PHY-protocol attacks against impulse-radio ultra-wideband ranging protocols (IR-UWB), with particular attention to 802.15.4a [17], which is the *de facto* standard. These studies concentrate only on reduction attacks, and estimate their effectiveness in terms of meters of distance reduction. We instead focus on the opposite problem, distance enlargement, which requires different countermeasures.

Chiang et al. [18] proposed the first technique able to mitigate the enlargement attack in case of dishonest prover. The verifier makes two power measurements of the prover's signal on two collinear antennas. Subsequently, it computes the difference of the two measurements. Given the standard path-loss model, if the difference is low, the signal source will be far away. Otherwise it will be near. The idea is that the adversary cannot modify the way the signal attenuates over the distance, thus the distance estimation is trusted. Obviously such proposal relies on the standard path-loss model, which is poorly reliable. The authors claim that if the path loss exponent varies between 2 and 4, an enlargement of more than twice the measured distance is impossible. In this paper, we focus on external adversaries. The problem of distance enlargement in presence of internal ones is challenging as well, but falls outside our present scope.

III. Reference Distance-Bounding Protocol

A distance-bounding protocol allows a *verifier* (V) to "measure" the distance of a *prover* (P). In its basic form, a distance-bounding protocol consists in a sequence of single-bit challenge-response rounds [6]. In each round, the verifier sends a challenge bit to the prover that replies immediately with a response bit. The round-trip time enables V to compute an upper-bound of the P distance. Then, the distance is averaged on all rounds. Many variants of distance-bounding protocols have been proposed in the literature [7], [8]. Here, we establish a *reference distance-bounding protocol*, similar to those described in [16] for external adversaries. It involves a *request* message (REQ) from the verifier, an *acknowledgment* message (ACK) from the prover, and a final *signature* message (SGN) from the prover. Such a reference protocol is vulnerable to jam-and-replay attacks, as we will show in Section IV, and SecDEv (cfr. Section V) will overcome these vulnerabilities.

The request and the acknowledgement convey, respectively, a and b, which are two independent, random and unpredictable sequences of bits. Note that, differently from the original version of distance-bounding protocol, the request and the acknowledgement are frames, rather than single bits. In fact, it is hard to transmit single bits over an IR-UWB channel. This is due to TLC regulation, which poses strict limits to the transmission power. In 802.15.4a [17], for example, every packet is preceded by a multi-bit synchronization preamble. The signature authenticates the acknowledgement and the request by means of a *shared secret* S. What follows is a formal description of the protocol.

REQ $\quad \mathrm{V} \longrightarrow \mathrm{P} : a$

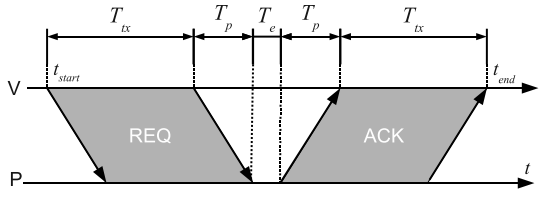


Figure 1. Round-trip time.

ACK $\mathrm{P} \longrightarrow \mathrm{V} : b$
SGN $\mathrm{P} \longrightarrow \mathrm{V} : H_S(a, b)$

The quantities a, b and $H_S(\cdot)$ are k-bit long. Therefore, the probability for an adversary to successfully guess one of these quantities is 2^{-k}. Such a probability gets negligible for a sufficiently large value of k, which we call the *security parameter*.

The verifier measures the distance between itself and the prover, by measuring the round-trip time $\hat{T}$ between the request and the acknowledgement messages. With reference to Fig. 1, we denote by t_{start} the instant when the transmission of REQ begins, and by t_{end} the instant when the reception of ACK ends. We denote by T_e the time interval from the end of REQ reception, to the beginning of ACK transmission. Since ACK does not depend on REQ, T_e does not include any elaboration time. It includes only the time for the antenna to switch from the receive mode to the transmit mode and the necessary hardware delays. We assume T_e to be small and known. Dedicated hardware can fulfill these requirements. We further denote by T_{pkt} the transmission time of the request and acknowledgement messages, and with T_p their propagation time in the medium. The round-trip time will be:

$$\hat{T} = 2T_p = (t_{end} - t_{start}) - 2T_{pkt} - T_e \tag{1}$$

Finally, we obtain a measure of the distance:

$$\hat{d} = \frac{c \cdot \hat{T}}{2} \tag{2}$$

where c is the speed of light.

The distance measurement precision depends on the capability of measuring the time interval with nanosecond precision. Localization systems based on IR-UWB can achieve nanosecond precision of measured time of flight, and consequently a distance estimation with an uncertainty of 30 cm. Also, this feature of time precision are available only with dedicated hardware.

IR-UWB protocols like 802.15.4a provides packets made up of two parts: a preamble and a payload. The preamble permits the receiver to synchronize to the transmitter and to precisely measure the time of arrival of the packet. The payload carries the information bits. In our protocol, a and b are transmitted in the payload part. We suppose the last part of the payload to carry a forward error correction code (FEC), for example some CRC bits.

In a non-adversarial scenario, the *actual distance* d will be equal to the *measured distance* $\hat{d}$. To deceive the measurement process, the adversary has to bring the verifier to measure a fake round-trip time. That is, she must act in a way that the verifier receives the acknowledgement at a different instant of time, while still receiving the correct signature. The basic idea of distance-bounding protocol is that an external adversary cannot deliver a copy of the legitimate acknowledgement *before* than the legitimate one.

On the other hand, she can deliver a copy of the acknowledgement *after* the legitimate one. In other words, she can only *enlarge* the measured distance, not *reduce* it. Thus, we are always sure that $d \leq \hat{d}$, i.e., the measured distance is a secure upper bound for the actual distance.

IV. Threat Model

We assume that the adversary (M) is an external agent, meaning that she does not know the shared secret (S) ant it cannot be stolen. Techniques like trusted hardware and remote attestation can help defending against these possibilities [19]. The objective of M is to deceive the verifier into measuring an enlarged round-trip time:

$$\hat{T} = 2T_p + \Delta T \tag{3}$$

in order to make it infer an enlarged measured distance:

$$\hat{d} = \frac{c \cdot \hat{T}}{2} = d + \frac{c \cdot \Delta T}{2} \tag{4}$$

We do not deal with distance reduction attacks. Since our protocol is an enhancement of the reference distance-bounding protocol of Section III, it offers the same guarantees against distance reduction attacks.

A. Adversary's Capabilities

M can eavesdrop, transmit or jam any signal in the wireless channel. The principle of a jammer is to generate a radio noise at a power comparable or higher than the legitimate one. In case of IR-UWB channels, a jammer could send periodic UWB pulses, in such a way to disrupt the synchronization process [20]. Alternatively, she could simply send random pulses in the payload part, in such a way the receiver discards the packet as corrupted after the FEC test. In both cases, the goal of the jammer is to disrupt the reception of the message.

M can transmit or jam *selectively*, in such a way that only a target node receives. In the meanwhile, M can correctly eavesdrop other signals. To do this, she can place a transmitting device nearby the receiver, and a listening one nearby the transmitter. Alternatively, she can use a single device with two directional antennas. One of them transmits to the receiver, while the other listens to the transmitter.

Another possibility is the *overshadowing* attack. In this attack, M injects a fake signal with higher power than the original one. The original signal becomes entirely overshadowed

by the attacker's signal. Ideally, original signal is treated as noise by the receiver. In this paper, we do not deal with this attack, and we focus only with jam-and-replay attacks. The overshadowing attack is indeed interesting and deserves a full analysis, that we are planning to do in future work. Here we only points out that it is not simple to be performed in a real-world IR-UWB protocol. In fact, the verifier does not receive only the fake signal, but the legitimate signal too. Even if the former is much stronger in power, the latter is still a valid IR-UWB signal, which interferes with the packet synchronization and reception. Sending an overshadowing signal is probably not enough. The adversary should also attenuate the legitimate signal with some complementary technique, such as electro-magnetic shields or similar.

We assume that M has no physical access to the prover or the verifier. This has two consequences: (i) she cannot tamper with the nodes and steal their secret material, and (ii) she cannot attenuate the wireless signals with electro-magnetic shields or Faraday cages.

B. Jam-and-Replay Attacks

In the distance-bounding protocol of Section III, the adversary can enlarge the measured round-trip time in the following way (Fig. 2a).

1) M listens to the radio channel, until she hears a REQ signal.
2) M waits for the ACK signal.
3) M jams the ACK signal and eavesdrop it in the meanwhile.
4) After a time ΔT, M replays it.

The adversary must replay the ACK signal selectively, in such a way that only the verifier receives it. Otherwise, the prover will also receive the replayed signal, and could infer that the protocol is under attack.

It is important to highlight that M has to wait for the legitimate ACK to end, before starting the transmission. This is because she must avoid signal collision.

The adversary can perform a similar attack on the REQ signal (Fig. 2b). Even in this case, M has to wait for the end of the legitimate REQ before starting her transmission.

We state the following:

Proposition 1 *In a jam-and-replay attack on REQ/ACK, the adversary must enlarge the round-trip time of a quantity ΔT not smaller than T_{pkt}, i.e., $\Delta T \geq T_{pkt}$.*

Proposition 1 represents the fundamental limitation of the jam-and-replay attacks. SecDEv will leverage on this to withstand them. Note that this limitation comes from the properties of the radio-frequency channel, and does not depend on how many devices the adversary controls. For the sake of simplicity, Figg. 2a and 2b show a single adversary.

V. SecDEv Protocol

SecDEv is a distance-bounding protocol, which measures the correct distance between a verifier V and a prover P in presence of an adversary M performing a jam-and-replay attack. It is similar to the reference distance-bounding protocol (cfr. Section III), except that the length of REQ and ACK do not depend only on the security parameter, but also on a *security horizon*.

Let us consider the Equation 3 for a general enlargement attack and apply the Proposition 1, we obtain the constraint $\hat{T} \geq 2T_p + T_{pkt}$. Hence:

$$\hat{T} \geq T_{pkt} \tag{5}$$

Equation 5 assures us that a measured round-trip time smaller than T_{pkt} has not been affected by any jam-and-replay attack. We can translate T_{pkt} in a distance d_M, that we call *security horizon*:

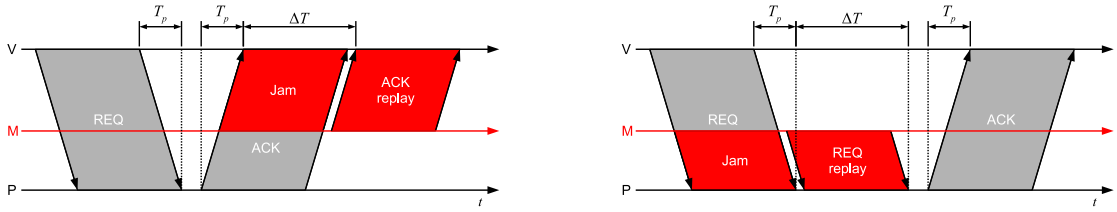
$$d_M \triangleq \frac{cT_{pkt}}{2} \tag{6}$$

In terms of distances, Equation 5 becomes:

$$\hat{d} \geq d_M \tag{7}$$

Equation 7 is our test to distinguish between trusted and untrusted distance measurements. V can extend the packet transmission time to enlarge the security horizon (cfr. Eq. 6), in order to securely measure longer distances. T_{pkt} is enlarged by introducing padding bits after the nounce. Padding bits have not to be unpredictable. They can have a well-known value (e.g. all zeroes), since they serves only to prolong the packet transmission time. V decides on the length of the REQ padding, and P has to respond with the same padding length in the ACK. Therefore, both messages have the same length, to withstand both jam-and-replay on REQ and on ACK.

Let us explain the protocol in detail. We assume that the wireless channel is characterized by the parameter tuple: $\{T_{pre}, R_{pld}, T_e\}$. T_{pre} is the transmission time of the preamble part. R_{pld} is the bit rate of the payload part. T_e is the reaction time of the prover node. In addition, we define the following triplet of protocol parameters: $\{k, S, d_M\}$. k is the security parameter. A higher value for k implies a higher security level, but has an impact on power consumption, as we will see in the following. S is a secret bit sequence shared between V and P. Its length is longer than or equal to k. d_M is the security horizon that distinguishes between trusted and untrusted measured distances. If the actual distance d is longer than d_M, the measured distance cannot be trusted because it may be affected by a jam-and-replay attack. In such a case, the protocol can be executed again with a longer d_M. Alternatively, the distance d can be first estimated in an insecure manner, and then securely confirmed with $d_M > d$.



(a) Jam-and-replay on ACK.

(b) Jam-and-replay on REQ.

Figure 2. Jam-and-replay attack.

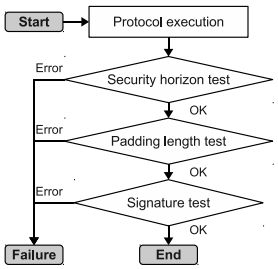


Figure 3. SecDEv algorithm.

A higher value for d_M allows us to measure longer distances, but has an impact on power consumption.

We further define the following quantities. N_{pad} and N_{fec} are respectively the number of bits of the padding and the FEC code. Since the number of bits of a and b is k, the total transmission time will be:

$$T_{pkt} = T_{pre} + (k + N_{pad} + N_{fec})/R_{pld} \tag{8}$$

If with $N_{pad} = 0$, the T_{pkt} identifies the minimum value of d_M. Thus, if the actual distance is smaller than this value, there is not need of padding bits. Otherwise, we determine N_{pad} with the following formula:

$$N_{pad} = \left\lceil \left(\frac{2d_M}{c} - T_{pre} \right) \cdot R_{pld} \right\rceil - k - N_{fec} \tag{9}$$

Using the Equation 9, we can set every value of d_M. Note that T_{pkt} grows with d_M. A larger security horizon causes longer messages, accordingly higher energy consumptions per protocol execution. An implementer must choose d_M as a trade-off between ranging capabilities and power consumption.

Fig. 3 shows the algorithm executed by V. After the protocol execution, V tests whether the measured distance is within the security horizon, that is, if $\hat{d} < d_M$. If this test fails, the measured distance is discarded as untrusted. Then, V tests the length of the ACK padding. If it contains less bits than the REQ one, the measured distance is discarded as untrusted. This is to avoid a jam-and-replay attack on REQ (cfr. Fig. 2b), in which M tries to lower ΔT by replaying REQ with a smaller padding. In such a case, P will respond with an ACK with a smaller padding too, and the attack will not pass the padding length test. Finally, V tests the validity of the cryptographic signature.

VI. Experimental Results

We combined SecDEv with multilateration technique to securely localize the prover. We analyzed the efficiency of this solution in terms of covered area and we compared it with *verifiable multilateration* [12], which is the state-of-the-art technique for secure positioning in wireless networks. Verifiable multilateration involves at least three distance measurements from different verifiers. The distance measurements are performed by means of distance bounding protocols, which are supposed to withstand reduction attacks. Verifiable multilateration deals with possible enlargement attacks by forcing an additional check to the final position estimation. In order to be trusted, the position must be inside the polygon formed by the verifiers, otherwise it is discarded as untrusted. Intuitively, this reduces the coverage area of the positioning technique.

In other words, classic multilateration is more scalable in terms of number of verifiers needed to cover a specific area. To quantify this, we have tested the performance of classic multilateration in terms of number of verifiers needed to cover a working area, and we have compared our results with those of verifiable multilateration, taken from [12]. We supposed that every verifier covers a circular area with radius $250\,\mathrm{m}$.

We neglect planned distributions [12], because in a real deployment, environment may impose constraints on the verifier positioning. Thus, we consider that the verifiers are uniformly distributed over the area of interest.

In order to evaluate the two techniques under the same conditions, our simulation were performed on areas of variable sizes. The verifiers were uniformly distributed in the area and in a boundary region outside the area, whose width was 10% of the area width. We use the boundary region to avoid the boundary effects [12] in the verifiable multilateration.

Fig. 4 shows how many verifiers are required to cover 95% and 90% of the working area. VM and CM curves are respectively verifiable multilateration with distance bounding and classic multilateration with SecDEv. The number of verifiers is the average of 100 simulations with confidence

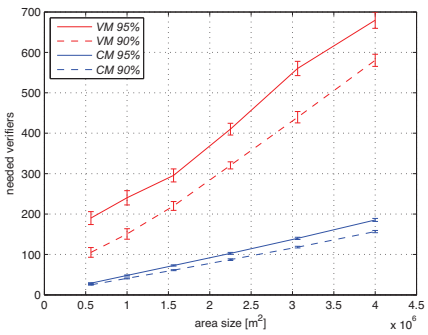


Figure 4. Verifiers required to cover an area.

intervals of 95% calculated for different values of working area from $0.5km^2$ to $4km^2$. The chart shows that classic trilateration needs far less verifiers, because it has not the limitation of the verification triangles. This gives strong motivation to fight distance enlargement attacks.

VII. Conclusions

We proposed SecDEv (SECure Distance EValuation), a distance-bounding protocol able to resist to enlargement attacks based on jam-and-replay tactics. SecDEv exploits the characteristics of wireless signals to establish a security horizon within which any adversarial attempt to play a jam-and-replay attack is detected. We also showed how SecDEv improves the scalability of secure positioning techniques in terms of number of anchor nodes.

References

[1] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1067–1080, Nov. 2007.

[2] R. G. Johnston, "Think GPS cargo tracking = high security? think again," Los Alamos National Laboratory, Tech. Rep., 2003.

[3] G. Dini, F. Giurlanda, and L. Pallottino, "Neighbourhood monitoring for decentralised coordination in multi-agent systems: A case-study," in *Computers and Communications (ISCC), 2011 IEEE Symposium on*, 2011, pp. 681–683.

[4] Y. Yu, R. Govindan, and D. Estrin, "Geographical and energy aware routing: a recursive data dissemination protocol for wireless sensor networks," UCLA Computer Science Department, Tech. Rep., 2001.

[5] Y. Desmedt, "Major security problems with the 'unforgeable' (Feige)-Fiat-Shamir proofs of identity and how to overcome them," *SecuriCom*, pp. 15–17, 1988.

[6] S. Brands and D. Chaum, "Distance bounding protocols," in *EUROCRYPT'93*, 1993, pp. 344–359.

[7] L. Bussard and W. Bagga, "Distance-bounding proof of knowledge to avoid real-time attacks," *IFIP/SEC*, pp. 223–238, 2005.

[8] G. P. Hancke and M. G. Kuhn, "An RFId distance bounding protocol," in *Proceedings of IEEE/Create-Net SecureComm 2005*, I. C. S. Press, Ed., 2005, pp. 67–73.

[9] A. Francillon, B. Danev, and S. Capkun, "Relay attacks on passive keyless entry and start systems in modern cars," in *NDSS*, 2011.

[10] M. Flury, M. Poturalski, P. Papadimitrios, J.-P. Hubaux, and J.-Y. Le Boudec, "Effectiveness of distance-decreasing attacks against impulse radio ranging," in *Proceedings of the third ACM conference on Wireless network security (WiSec2010)*, 2010, pp. 117–128.

[11] J. Kong, Z. Ji, W. Wang, M. Gerla, R. Bagrodia, and B. Bhargava, "Low-cost attacks against packet delivery, localization and time synchronization services in under-water sensor networks," in *Proceedings of the 4th ACM workshop on Wireless security*, ser. WiSe '05, 2005, pp. 87–96.

[12] S. Čapkun and J.-P. Hubaux, "Secure positioning in wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 221–232, feb 2006.

[13] N. Tippenhauer and S. Čapkun, "ID-based secure distance bounding and localization," in *Computer Security – ESORICS 2009*, ser. Lecture Notes in Computer Science, M. Backes and P. Ning, Eds. Springer Berlin / Heidelberg, 2009, vol. 5789, pp. 621–636.

[14] J. Clulow, G. P. Hancke, M. G. Kuhn, and T. Moore, "So near and yet so far: Distance-bounding attacks in wireless networks," in *Proceedings of European Workshop on Security and Privacy in Ad-Hoc and Sensor Networks (ESAS)*, 2006, pp. 83–97.

[15] K. B. Rasmussen and S. Čapkun, "Location privacy of distance bounding protocols," in *Proceedings of the 15th ACM conference on Computer and communications security*, ser. CCS '08. ACM, 2008, pp. 149–160.

[16] M. Poturalski, M. Flury, P. Papadimitrios, J.-P. Hubaux, and J.-Y. Le Boudec, "Distance bounding with IEEE 802.15.4a: Attacks and countermeasures," *IEEE Transactions on Wireless Communications*, pp. 1334–1344, 2011.

[17] Z. Sahinoglu and S. Gezici, "Ranging in the IEEE 802.15.4a standard," in *Proceedings of IEEE Wireless and Microwave Technology Conference*, 2006, pp. 1–5.

[18] J. T. Chiang, J. J. Haas, J. Choi, and Y.-c. Hu, "Secure location verification using simultaneous multilateration," *IEEE Transactions on Wireless Communications*, vol. 11, no. 2, pp. 584–591, feb 2012.

[19] W. Hu, H. Tan, P. Corke, W. C. Shih, and S. Jha, "Toward trusted wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 7, no. 1, pp. 1–25, aug 2010.

[20] M. Poturalski, M. Flury, P. Papadimitrios, J.-P. Hubaux, and J.-Y. Le Boudec, "The cicada attack: degradation and denial of service in ir ranging," in *Proceedings of 2010 IEEE International Conference on Ultra-Wideband*, 2010, pp. 1–4.