UWB based Secure Ranging and Localization

Baptiste Pestourie

Université Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITE GRENOBLE ALPES

Spécialité : **NENT**

Arrêté ministériel : 25 mai 2016

Présentée par

Baptiste Pestourie

Thèse dirigée par **Vincent BEROULLE, Professeur des Universités, LCIS**

Co-encadrée par **Nicolas FOURTY, Maître de Conférence, LCIS**

préparée au sein du **Laboratoire LCIS**

dans **l'École Doctoral EEATS**

UWB Secure Ranging and Localization

Thèse soutenue publiquement le **04.12.2020,** devant le jury composé de :

François Spies

PR Université de Franche-Comté, Président

Adrien Van den Bossche

MCF HDR Université Toulouse 2, Rapporteur

Aurélien Francillon

PR Eurecom, Rapporteur

Assia Tria

Ingénieur HDR CEA Grenoble, Examinatrice

Vincent Beroulle

PR Université Grenoble-Alpes, Directeur de thèse

Abstract

Location services are foreseen as one of the major IoT features in the next years, and have gained a lot of interest over the last decade from the literature of Wireless Sensors Networks, (WSN) and Vehicular Ad Hoc Networks (VANet). Impulse-Radio Ultra-Wideband (UWB), standardized in IEEE 802.15.4-2003, is currently the most performant radio positioning technology with centimeter-level accuracy and is used widely in industrial applications. It has been proven in the literature that UWB positioning is not completely tamper-proof, as various physical and link layers vulnerabilities have been identified in 802.15.4. Most of the major attacks against IR-UWB are physical-level attacks, such as Early-Detection/Late-Commit (ED/LC). Considering their cost, complexity, and sometimes lack of maturity, they are not necessarily the most realistic attacks against cheap IoT systems. On the other hand, protocol-level flaws expose IR-UWB positioning against attacks that can be mounted with limited expertise and cheap hardware. Hence, the aim of this work is to identify the most critical vulnerabilities of 802.15.4 IR-UWB, evaluate real-world attacks against UWB IPS and propose low-cost countermeasures suitable for IoT applications. An open platform for IR-UWB positioning security evaluation, SecureLoc, is part of the contributions. We propose and evaluate various spoofed acknowledgment-based attack schemes against IR-UWB. Several countermeasures, at the physical, medium access control and system level, are proposed, including notably a novel weak PUF-based authentication protocol, a spoofing resilient acknowledgment scheme, a tamper-proof ranging approach, and a cooperative verification protocol for rogue node detection. All the proposed attacks and countermeasures have been implemented and evaluated on SecureLoc.

Résumé

Les services de localisation sont considérés comme une des fonctionnalités majeures de l'IoT dans les prochaines années, et font l'objet d'un intérêt croissant dans la littérature des réseaux de capteurs sans fil (*Wireless Sensors Network (WSN)*). La technologie *IR-UWB* (Impulse-Radio Ultra-Wideband), standardisée dans IEEE 802.15.4, est actuellement la technologie de localisation la plus performante avec une précision de l'ordre du centimètre et est largement déployée dans des applications industrielles. Il a été démontré dans la littérature que la localisation UWB n'est pas immunisée contre la falsification (*tampering)* ; plusieurs vulnérabilités au niveau des couches physiques et liaison de données ont été identifiées dans des travaux précédents. La plupart des attaques majeures contre l'UWB sont des attaques physiques, telles que les attaques *Early-Detection/Late-commit (ED-LC).* Du fait de leur coût et complexité, parfois doublé par un manque de maturité technologique, elles ne sont pas nécessairement les menaces les plus réalistes dans un contexte IoT. En revanche, des failles protocolaires au niveau de la couche liaison de données exposent l'IR-UWB à des attaques nécessitant peu d'expertise et de matériel. Par conséquent, les travaux introduits dans ce manuscrit sont consacrés à l'identification des menaces les plus critiques contre la technologie 802.15.4 IR-UWB, à évaluer des attaques contre cette technologie en conditions réelles, et à proposer des contremesures à bas coût appropriées à des applications IoT. Une plateforme dédiée à la localisation IR-UWB, *SecureLoc*, fait partie des contributions. Plusieurs attaques à base d'acquittements frauduleux sont proposées et évaluées. Diverses contremesures sont proposées, au niveau des couches physiques, liaison de données et système, incluant notamment un protocole d'authentification physique basé sur un *weak PUF*, une technique d'acquittement résistante contre les usurpations d'identité, un protocole d'estimation de distance *(ranging)* immune contre la falsification de position et un protocole coopératif de détection des nœuds malicieux. Toutes les attaques et contre-mesures proposées ont été implémentées et évaluées sur la plateforme SecureLoc.

Acknowledgment

I am grateful to Vincent and Nicolas for their supervision throughout my PhD. Our collaboration was very enjoyable thanks to their advices and guidance. I would also like to thank Cyril for his long-standing friendship and his comments on the manuscript, Louise and my family for their continuous support, and my LCIS colleagues for the good moments spent together.

This work has been supported by the grants provided by IDEX IRS (*Initiatives de Recherches Stratégiques)* and Cybersecurity Insitute Grenoble-Alpes.

Chapter I Introduction

The need for spatial tracking in industrial environments or public facilities has been leading the growth of indoor positioning solutions for over two decades. Manufacturing and supply chain supervision, indoor drone navigation or goods tracking in warehouse are part of the non-exhaustive list of applications benefiting from indoor positioning solutions. Historically, the Global Positioning System (GPS) is the most widely known positioning technology and is now used by the general public on a weekly basis. However, GPS does not deal with Line-of-Sight obstruction, which automatically occurs in indoor environments, and lacks robustness against multipath effects. Considering that, new solutions had to emerge regarding indoor localization, as GPS does not provide satisfying enough performances and reliability when it comes to the aforementioned applications. Positioning technologies have existed for about a century, with the introduction of the Sound Navigation Ranging (Sonar) [1] and RAdio Detection And Ranging (Radar) [2], but their cost and limitations have prevented them from being implemented in public facilities or factories until recently. Among the various technological solutions developed for indoor positioning, radio-based solutions have gained a lot of interest from the literature in the early 2000s, considering their low cost and relatively easy integration in buildings. Electromagnetic signals can be exploited for positioning, as their attenuation, phase or time-of-flight convey information about the distance traveled by the signal or its angle of arrival. This principle was already used for the first Radar prototypes in the 1930s, but the recent growth of the Internet of Thing (IoT) is leading the expansion of cheap radio transceivers in all sorts of applications, enabling low-cost integration of tracking features [3]. Many research works have been conducted on developing indoor positioning systems based on the most popular radio standards; the challenge was to reach sufficient accuracy and reliability to be functional. Solutions based on Bluetooth or WiFi can typically reach meter-level accuracy and a localization rate up to 10 Hz [4], but this is not enough for some applications, such as drone navigation. The release of the 3.1 ~ 10.6 GHz bands by the Federal Communications Commission (FCC) in 2002 was a game changer, leading to the introduction of Ultra-wideband (UWB) technologies. Two different standards have defined UWB specifications; IEEE 802.15.3 [5], designed for high-throughput, and IEEE 802.15.4 [6], designed for accurate ranging and positioning. 802.15.3 UWB was primarily intended for wireless USB and wireless HDMI, which used to be considered as very promising back in 2010, but ended up falling apart in favor of USB-C and are now very marginal technologies. On the other hand, Impulse Radio Ultra-wideband (IR-UWB), defined in IEEE 802.15.4, is based on very short pulses (2 ns width) allowing accurate time of flight ranging with an typical accuracy of about 10 cm and localization rates going up to 1000 Hz. Due to the large bandwidth used (at least 500 MHz), 802.15.4 IR-UWB is robust against multipath effects and provides an accurate and reliable solution for indoor positioning applications. An amendment for additional physical layer specifications, IEEE 802.15.4a, has been defined in 2007, and merged into 802.15.4 in

2011. Further enhancements have been brought in 2015 in IEEE 802.15.4f. Currently, the leading manufacturer on the market is Decawave, and their popular IR-UWB transceiver, the DW1000 [7], is based on 802.15.4a. Hence, the majority of UWB IPS is currently based on implementations of 802.15.4a and actual implementations of 802.15.4f are not widespread on the market as far as we are aware.

IR-UWB did not really reach a broad success until very recently. Although it is used in specific and exigent positioning applications, the fact that IR-UWB, contrary to its competitors, was not already integrated in popular IoT products was a hindrance to its growth. One of the motivations for the interest towards radio-based is indeed that the positioning features can be integrated into existing radio devices, which are primarily used for communication purposes. Despite featuring low power consumption and high throughput, IR-UWB has never been broadly integrated in general public IoT products, partially due to the cost of IR-UWB transceivers being 3 to 5 times higher than Bluetooth or WiFi transceivers. However, IR-UWB remains the leading radio solution for demanding industrial applications or indoor navigation, considering that it largely outperforms its radio competitors. Moreover, one of the powerful assets provided by IR-UWB is its inherent security when it comes to the integrity of the positions measured.

The physical layer defined for UWB-IR in IEEE 802.15.4 has indeed the particularity to support time-of-flight ranging, which is typically not possible on other radio standards due to the extremely small time granularity required. Electromagnetic signals travel nearly at the speed of light, which means that typical times-of-flight in indoor environments lie in the order of magnitude of a few nanoseconds. The modulation techniques used by most radio standards, along with other factors, do typically not allow reaching such granularity, whereas the impulse-based nature of IR-UWB leads to sub-nanosecond accuracy [6]. Due to that limitation, conventional radio positioning approaches use either the phase or the attenuation of a signal as their indicator for distance estimation, but these parameters can be potentially replicated and tampered by an attacker, by simply repeating or relaying a frame with specific physical parameters. On the other hand, considering that it is not possible to exceed the speed-of-light, reducing the time-of-flight of a signal in order to appear closer is not physically possible. As a consequence, the inherent principle of time-of-flight ranging mitigates attacks against proximity-based systems, and can easily unveil attempts to replay or relay attacks as these attacks typically induce delays much higher than the time-of-flight itself. These considerations concern the physical layer, and integrity protection can only be granted if proper security mechanisms are implemented at higher levels as well.

The potential of IR-UWB for secure positioning has recently attracted interest from several manufacturers, leading to the creation of the IEEE 4z group task in January 2019 and the Fine Ranging (FiRa) consortium in July of the same year. The 4z amendment [8] aims to improve even further the ranging performances of IR-UWB and the security of the physical layer, with companies such as Apple, NXP or Decawave involved. The FiRa consortium is "dedicated to the development and widespread adoption of seamless user experiences using the secured fine ranging and positioning capabilities of interoperable Ultra-Wideband

(UWB) technologies" [9]. IR-UWB meets a growing success and is now starting to reach the general public market; the iPhone 11 Pro, launched by Apple in August 2019, was the first smartphone to embed a UWB chip. The pandemic situation created by Covid-19 has considerably highlighted the need for more accurate ranging and positioning features on smartphones, as the development of social distancing applications has suffered from the limited performances of Bluetooth-based ranging. Major actors of the Information Technology (IT) industry have expressed their interest in IR-UWB in the first semester of 2020, and the technology is expected to be widely integrated in consumer electronics products and notably smartphones by the next two or three years [10].

As of July 2020, the transition to 4z is still on-going. The very first 4z IR-UWB transceivers have recently started to get released, and the 4z group task is still working on the amendment specifications. The majority of IR-UWB IPS are currently based on 802.15.4a, and given that the 4z guarantees retro-compliance, 4a and 4z devices will most likely have to interoperate in the next years. The work introduced in the manuscript is mostly based on the 802.15.4a standard; the impact of the transition to 4z on the presented results will be discussed in the conclusion.

The security assets brought by the time-of-flight approach on 802.15.4a IR-UWB are not vulnerability-free. Considering that IR-UWB IPS are often tracking expensive equipment (e.g., drones, forklifts), security flaws can lead to significant damages and even sometimes to human harm. Previous works have contributed to identifying the security flaws of 802.15.4a UWB, and in some cases proposed actual attacks based on these vulnerabilities. The major attacks proposed in the literature are physical attacks, which typically allow to fool a receiver into measuring an incorrect timestamp. Early-Detection/Late-Commit attacks (ED-LC) [11] in particular, which can be described briefly as a "relay backward in time" type of attack, are currently considered as one of the most significant attacks against this technology[1]. While being technically advanced and able to bypass all the security mechanisms defined in 802.15.4, ED-LC and other physical attacks proposed in the literature have been demonstrated, as far as we are aware, only in simulation environments. Also, their impact has been evaluated mostly from a peer-to-peer ranging perspective. Indoor Positioning Systems require multiple reference stations to work properly, which means that a malicious node has to be able to tamper not one but all the reference stations when mounting an attack. To remain stealthy when doing so, the malicious node must preserve the cohesiveness between the responses of the different reference stations, which is a challenging task. This induces an extra layer of complexity for an attacker compared to tampering the ranging protocols of a single station. Although countermeasures have been proposed for most of the identified vulnerabilities, these countermeasures are often only functional against the specific flaw they aim to address, are not costless in terms of hardware and computational load, and often involve link or physical layer modifications to the standard. They aim to secure the ranging protocol, i.e., the distance estimation in a peer-to-peer protocol. Yet, IPS

[1] According to meeting notes from the IEEE 4z Group Task.

always involve multiple stations, and the redundancy and cohesiveness between the reference stations can be exploited to build high-level countermeasures even if the integrity of ranging protocols cannot be guaranteed.

Based on this consideration, a fair amount of research works based on system-level approaches [4] for secure positioning has emerged in the literature of Wireless Sensors Networks (WSN) in the last fifteen years. Although some of these approaches are specific to a given technology, a significant effort has been made to propose high-level countermeasures regardless of the particular radio technology used on the IPS. One of the interesting properties of some of these approaches is that they are not only effective against attacks held by a third-party (*external attack*), but also against nodes that are legitimately part of the network and lie on their position (*internal attacks*). Indeed, 802.15.4 IR-UWB positioning is based on the assumption of honest participation of the nodes in ranging protocols, and is not more immune than other technologies against internal attacks, which are challenging to detect. Existing countermeasures against internal attacks, including system-level countermeasures, require often specific hardware to be functional (e.g., directive antennas or specific transceiver designs) and are not costless.

One of the main issues regarding the current state-of-the-art of secure positioning on IR-UWB is the lack of real-world experimentations and results when it comes to the proposed attacks and countermeasures. We already mentioned that the major physical attacks proposed in the literature have not been demonstrated outside a simulation environment. Their requirements, arguably in terms of cost and notably in terms of technical complexity, are fairly high. Their capability to produce coherent positions against an actual IPS remains to be studied. Regarding countermeasures, system-level approaches are promising in terms of flexibility, cost and portability, yet they are also rarely studied in real-world environments and mostly demonstrated through simulation or strictly analytically.

Ultimately, the goal of any position tampering attack, no matter if it is mounted internally or externally, is to fool the IPS into thinking that the measured position is legit. This mostly depends on the capability to produce tampered distances than are close enough in terms of stability and mutual agreement between the reference stations to distances obtained in non-adversarial conditions. Proper benchmarks are needed to evaluate that, especially considering that any localization algorithm deals with natural inconsistencies due to the measurement noise. Based on that consideration, we have built an open testbed for secure positioning, SecureLoc. Since off-the-shelf IR-UWB positioning systems do not give access to all the layers, SecureLoc provides an open-source testbed for secure positioning experiments. The motivation is to allow the evaluation of attacks and countermeasures in a real-world IR-UWB IPS, with full access to the layers involved in the localization process. Based on the localization experiments on SecureLoc and on an analysis of the security model of a typical UWB IPS, we propose an EBIOS [12]–based vulnerability analysis. We highlight through this vulnerability analysis that the absence of cryptography on the acknowledgment defined in 802.15.4a is currently one of the most critical vulnerabilities of IR-UWB, due to the acknowledgments being used in the ranging protocols recommended in the standard. This

vulnerability is well-known, and it has been suggested several times in the literature that it could be simply addressed by replacing acknowledgments by other types of frames supporting integrity protection in ranging protocols. Exploiting the acknowledgment vulnerability to build distance tampering attacks has often been considered as trivial, and has never been studied more in-depth to our knowledge. Yet, we demonstrate that due to the very small time granularity involved, the basic forged acknowledgment attacks suggested in the literature are far from trivial and mildly effective. We demonstrate how the effectiveness of these attacks can be enhanced, before proceeding to propose countermeasures that do not involve acknowledgment replacement in the ranging protocols or additional cryptographic operations. Finally, we propose system-level countermeasures that can detect internal attacks without requiring any dedicated hardware, along with contributions at the physical level regarding node authentication and key establishment. A more detailed list of the contributions is given chapter by chapter in the following.

The history of the evolution of positioning technologies is briefly covered in Chapter II, followed by a comparison of the localization performances of the major radio technologies. A particular focus is given on the 802.15.4 IR-UWB specifications, including its built-in security mechanisms. The most significant system-level approaches for secure positioning are introduced and classified.

SecureLoc platform is introduced in Chapter III. We show the benefits of SecureLoc as a testbed for secure positioning experiments on IR-UWB through a description of the platform architecture and functionalities. We evaluate localization performance benchmarks in non-adversarial settings with state-of-the-art localization approaches, to later evaluate the capabilities of attacks to produce tampered positions that are realistic. For that purpose, three detection factors are defined, based on the redundancy, consistency and plausibility of the distances measured. Each of these factors is evaluated in non-adversarial settings, such as defining thresholds for attack detection.

A vulnerability analysis of 802.15.4 IR-UWB indoor positioning is proposed in Chapter IV. We report the vulnerabilities of IR-UWB positioning and related attacks that have been identified in the literature. We evaluate their severity and likelihood based on the guidelines proposed by the *French National Agency of Cybersecurity (ANSSI)* in the risk manager method EBIOS [12]. Our analysis is held from the perspective of a typical industrial indoor positioning system. We discuss the most critical attacks identified.

The attacks implemented on the platform are presented in Chapter V. We first introduce an internal attack approach. Then, we propose several attack schemes based on spoofed acknowledgments. We demonstrate that the basic exploit of the acknowledgment vulnerability suggested in the literature is only effective when acknowledgments are tightly scheduled by the victim node. When acknowledgments are not scheduled, we demonstrate analytically and experimentally that basic spoofed acknowledgments attacks produce very erratic results due to the variability of the reply time. Two enhanced attack schemes are proposed, which combine forged acknowledgment with targeted jamming attacks and relay attacks. We show that these schemes allow the attacker to get finer control over the distances

obtained. When acknowledgments are tightly scheduled, we demonstrate that the attack can be extended to the whole positioning system and let the attacker take control over the victim node's position with high accuracy.

Several countermeasures are proposed in Chapter VI. We first propose to address the acknowledgment vulnerability without any additional cryptographic operation, with an approach based on randomized reply time. Then, we present two system-level countermeasures that aim to prevent from both external and internal attacks. The first one is based on a differential ranging approach, and the second one on a cooperative protocol. Finally, we introduce several interesting results at the physical layer regarding node authentication and key establishment. We demonstrate that clock skew on UWB devices has properties that are unique to each manufactured chip, similarly to a Physically Unclonable Function (PUF) [13], and propose an authentication protocol based on these properties. A Channel-Based Key Extraction (CBKE) approach based on first path power is proposed, which can be integrated within the proposed authentication protocol. By exploiting the randomness and reciprocity of wireless channels, these approaches allows two nodes to establish a secret key without using asymmetric cryptographic primitives.

Lastly, we conclude in the last chapter and discuss the perspectives related to the major contributions in this work. We give some insights on the impact of the transition to 4z on this work.

Contents

Chapter II Context

We introduce in this chapter the state-of-the-art of Indoor Positioning Systems (IPS) and their security. We first go through the history behind the emergence of modern IPS, and show how modern positioning techniques are derived from historical technologies. Radio-based IPSs have demonstrated a promising potential over the last fifteen years for IoT applications; we discuss the core principles of radio positioning techniques and compare several popular radio technologies on their cost and performances. In particular, we introduce in detail 802.15.4 IR-UWB, which is currently by far the most performant radio positioning technology. Finally, we discuss the main aspects of radio IPS security; general attack schemes against these systems are introduced along with the major high-level countermeasures proposed in the literature.

Chapter contents

II.1 Radio-based Indoor Positioning

II.1.1 General Context

Positioning an object in a 2D or 3D space is intrinsically related to the capability of measuring distances between two points. The first instances of technologies able to estimate distances remotely came in the 20th century, with the invention of Sound Navigation Ranging (Sonar) [1] and RAdio Detection And Ranging (Radar)[2]. With the imminent World War I, these new devices were oriented towards military navigation. These two technologies are both able to estimate the distance to the first obstacle in a straight Line-of-Sight[1] by using the reflection of a signal (or *echo*) on this obstacle. However, they differ by the nature of the signal used for this purpose: sonar is based on acoustic waves for the sonar whereas radar uses radio waves. Sonar exploits the propagation of sound in water to estimate the distance and is most effective in immersion. Sonar was indeed conceived for submarines and vessels. Radar on the other hand is effective in the atmosphere and has been widely used for vessels and airplanes detection.

Another major difference between these two technologies is how the signal is processed. Sound propagated at a speed of roughly 1500 m/s in water, which is low enough to measure with reasonable accuracy the signal travel time. On the other hand, radio waves travel in the air nearly at the speed of light, i.e., approximately 300 000 km/s, and estimating the time-of-flight of a signal is much more challenging given its order of magnitude. This challenge was overcome a few years before WWII as several countries developed independently their own prototypes. The foundation of the radar is based on a transceiver sending pulses at short intervals, where the received echoes were monitored by an oscilloscope with enough resolution to estimate the accurately the travel time. The first multi-static radars [2] have been developed around 1935; by combining multiple antennas, they were able to provide three-dimensional covering and estimate the direction of the detected objects. Thus, radars were accurate enough during WWII to follow even airplane or vessel displacements in real-time, but the cost and size of these devices were significant. Yet, it was the first step into the technological development of remote wireless positioning. Interestingly enough, despite being more than a century old, the concepts of time-of-flight ranging are still used in modern technologies as we will discuss throughout this manuscript.

Radio-based ranging has been pushed further with the introduction of the Global Positioning System (GPS) in the early 1970s, which is a major milestone in the history of positioning systems [14]. Contrary to the previously developed ranging devices, which were only able to give the position of an object relative to another one in a rather limited range, GPS was the first technology providing an absolute position on Earth [15]. GPS has been made possible by the progress of aerospace sciences and the growing fleet of satellites into space. GPS was indeed the first instance of a Global Navigation Satellites System (GNSS);

[1] The first radar prototype invented by Christian Hülsmeyer [139], also called Telemobiloscope, which was first demonstrated in 1904, could actually only detect the presence of an obstacle and not measure the distance. The first prototypes of radar that were able to estimate the distance came out a few years before World War II.

other GNSS have been introduced later, including Galileo, the European GNSS [16], which started its services in 2016. The basic principle of a GNSS is the following; satellites broadcast continuously radio signals that are timestamped locally by the satellite with an embedded high-accuracy [17] atomic clock. A GNSS receiver on Earth can determine its position if it receives at least the signals from four different satellites, by computing the time-of-flight to these four references and extracting geometrically its localization. The accuracy of GNSS has increased over the years; nowadays, civilian GPS can reach accuracy up to 1 m with outside[1]; differential GPS [15] and military GPS[2], up to 10 cm accuracy.

For more than a decade, smartphones have integrated GPS receivers and the general public has access to real-time localization in most places of Earth. A lot of industrial applications also benefit from the advantages provided by positioning features. A lot of these applications take place in indoor environments, such as factories [18], warehouses [19], or other public utilities like malls, train stations, or airports. The benefits of positioning features have been increasing over the last two decades with the progress of drone and robot technologies, in which navigation typically relies on self-localization. In factories, setting up a positioning system on a manufacturing chain can significantly help to anticipate problems and reduce the paralysis time involved in those. In busy warehouses, localization tags help in finding items faster and more efficiently [19]. Basically, a lot of applications can take benefits from indoor positioning systems (IPS), where the localization is relative to the monitored environment and not to the whole Earth surface.

However, there are some serious drawbacks to GNSS when it comes to this kind of smaller-scale localization systems, especially in indoor environments. The major problem is the lack of resilience of GNSS in indoor environments; GNSS does not deal very well with Non-Line-of-Sight (NLOS), which is automatically the case in any building: the multipath and attenuation effects degrade significantly the accuracy and reliability of the position estimation [20]. Also, a positioning system relying on a GNSS is by nature not self-dependent as it depends on an external entity (in this case, a satellite constellation), which can be a concern (e.g., for military applications). Also, a GNSS transceiver is dedicated mostly to localization and cannot fulfill other communication purposes.

With the advances of the IoT [21], and the development of the various wireless technologies used in IoT products, the idea has risen to extract position information from the radio data at a low, if not non-existing, additional hardware cost. Indeed, wireless communication technologies are based, similarly to radars, on electromagnetic waves. As a consequence, the principle that has been exploited for over a century for ranging can be applied to any modern radio communication system, without having necessarily to integrate specific hardware dedicated to positioning. Multiple solutions based on different radio technologies have been developed over the years and several of them have demonstrated highly superior performances to GNSS in indoor environments.

[1] With high-end GPS receiver; smartphone typically reach about 5 m accuracy[140]
[2] Military GPS is more accurate as it is dual-frequency where civilian GPS is only single-frequency

II.1.2 Ranging principles on radio Indoor Positioning Systems

A typical radio positioning system consists of two types of devices, reference stations and mobile nodes [22]. The position of a mobile node can be determined from the radio communications with several reference stations with known positions, usually at least three. Reference stations are often referred to as *anchors* and mobile nodes as (*mobile) tags*; we will use this nomenclature throughout this manuscript. So far, the configuration is similar to a GPS, except that contrary to satellites, which are mobile reference points, anchors are usually still in radio-based IPSs to achieve decent performances.

Different physical parameters can be observed on the individual links between a mobile node and each of the reference stations to extract information on the position. Usually, either a distance or an angle is extracted from a peer anchor-tag. There are three physical characteristics of the radio signals that are usually exploited to extract this information:

- **Signal attenuation:** the attenuation of a signal traveling through the atmosphere is quadratically related to the distance traveled. If the transmission power is known, a receiver can estimate the attenuation from the reception power observed and process the distance traveled. Signal attenuation, or Free-Space Path Loss (FSPL), is usually modeled with the Friis attenuation formula [23].
- **Time-of-flight:** as discussed previously, the speed of a radio signal being known, the distance between two nodes can be extracted from their mutual time-of-flight.
- **Signal phase:** signal phase contains information on the signal angle-of-arrival and time-of-flight. With multiple antennas, the angle-of-arrival of the signal can be computed from the phase difference between the antennas. Phase being a periodic function of time-of-flight, it can also be exploited to obtain indirectly the signal time-of-flight with channel aggregation techniques that we detail further.

The first two parameters allow extracting a distance, whereas the last one allows extracting an angle-of-arrival. The advantages and drawbacks of these methods are detailed below.

Signal attenuation: This is the most common approach used in radio IPS, mostly because of the low requirements involved. Any radio device capable of providing an estimation of reception power is a potential candidate for this approach. Also, since the only information needed is the reception and transmission power there is no need for a specific link layer protocol. This last point has a lot of repercussions from a privacy perspective: any radio device is subject to be roughly localized, even it was never intended in the first place, whenever it is actively emitting. This principle has been used for years for smartphones positioning in LTE networks [24]. The implications are huge in terms of privacy protection, considering that communicating directly exposes information on the emitting device position. In most implementations of wireless communication standards, an estimation of the reception power is given by a Reception Signal Strength Indicator (RSSI)[1]. The distance are

[1] The definition of RSSI tends to differ across manufacturers, and the exact calculations often vary from one chip to another [45]

usually estimated from the RSSI based on the Friis propagation model [23]; yet, this model is very limited in harsh environments and has limited performances in most real-world indoor environments [25]. As a consequence, a lot of approaches rely rather on profiling, where the RSSI response of each anchor is monitored on a set of reference points on the indoor surface covered by the IPS. After this learning phase, the RSSI responses of a given mobile tag are compared to the profile to provide an estimate of the position [26]. When available, Channel State Information (CSI) gives way more information than RSSI, such as channel scattering, power decay, or fading, and can significantly improve the accuracy compared to methods based on RSSI-only [27]. Profiling approaches do not solve the problem of dynamic environments: if the environment has changed since the learning phase (e.g., furniture being moved around, more personnel than usual in the room), the profile becomes deprecated and performances are degraded [26]. Some approaches reduce the impact of environment changes by updating dynamically their profile, which is monitored by an Artificial Intelligence (AI) [28], but it is typically not stable enough for application with high accuracy and robustness requirements. Given the low requirements of RSSI-based positioning, there have been attempts of RSSI/CSI based positioning systems with most common radio technologies, including notably Bluetooth [29], 802.11n [26], Zigbee [30], RFID [31], LoRa [32], UWB [33] and Sigfox [34].

Time-of-flight: The main challenge with time-of-flight ranging, as for radars, is to get a high enough time resolution to provide accurate distance estimations. At the speed-of-light, one meter is traveled in 3.3 ns, which is already below the duration of a single clock cycle of a typical microcontroller. This aspect is less of an issue for GNSS receivers, as the atomic clocks of the satellites do provide this high resolution. The bias induced in the distance estimation by the inconsistency of the much cheaper receiver clock is the same for each of the four reference satellites, hence can be easily corrected with a linear search. This is not the case for IoT IPS, as none of the devices performing the ranging process has a highly stable clock. Another crucial aspect is that the time-resolution of a given radio technology is proportional to the bandwidth. The bandwidth and modulation technique used on most modern standardized radio technologies are not suitable for time-of-flight estimation as they do not provide high-enough time resolution. As a consequence, time-of-flight is mostly used on UWB today.

Signal phase: Extracting the angle-of-arrival from the signal phase requires using a specific receiver design based on an array of antennas. The Angle-of-Arrival (AoA) is then computed from the phase difference within this array of antennas. The distance between each antenna being known, the angle of arrival is determined using algorithms such as Multiple Signal Classification (MUSIC) [35].

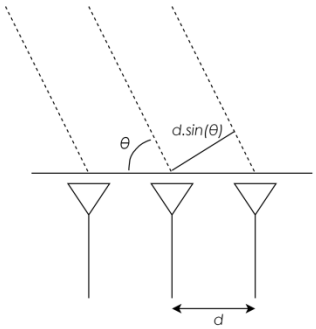


Fig. 1. Angle-of-arrival calculations from signal phase difference within an antennas array

A simplified example is shown in **Fig. 1**. For a given angle-of-arrival θ, the difference of time-of-flight between two antennas separated by a distance d can be approximated by d.sin(θ) * c. This difference is typically below the equivalent period of the signal, hence can be trivially estimated from the phase difference between the two antennas, allowing the receiver to extract the angle-of-arrival.

AoA-based methods are extremely dependent on Line-of-Sight (LoS): the angle measured is indeed only relevant if the signal followed a straight line from the emitter to the receiver. If the receiver captures a reflection of the signal, the angle measured will be significantly decorrelated from the emitter's actual position. AoA positioning is typically suited to environments where stations can be placed in spots featuring a clear LoS with the whole covered surface (e.g., on the ceiling). In that case, they typically achieve better performances than RSSI-methods [36]. Nevertheless, this is a less common solution given the additional hardware cost and the LoS constraints.

The process that computes a position from a set containing the distances between a mobile node and the reference stations is called *multilateration.* When angles are calculated instead of distances, it is defined as *multi-angulation.*

Signal phase can also be exploited to extract the time-of-flight. The phase shift of a signal between the expedient and recipient gives indeed an estimation of the time-of-flight modulo the signal period. By reproducing the phase measurement on different frequencies, which is a process known as *Channel Aggregation*, a set of arithmetic equations will be obtained as the signal period will be different for each frequency, and the time-of-flight can be computed as the solutions of this equation set. This approach has been used on WiFi [37] and LoRa 2.4 GHz chips [32, 38]. Since the measured parameter is the signal phase and not the actual time-of-flight, we classify them in this category rather than the previous one: from a security perspective, their robustness rely on the integrity of the phase measurement, which is not the case for a "true" time-of-flight ranging protocol.

II.1.3 Comparison of radio positioning technologies

We compared the positioning capabilities of the most prominent radio technologies based on the various research works held on radio IPS in the literature. We conducted this comparison from an IoT perspective, focusing on networks based on small and low-cost chips integrating wireless communication protocols. As a consequence, technologies that provide only localization and not communication are not considered. In the following, Bluetooth, Zigbee, WiFi, and UWB are compared according to the following criteria: localization accuracy, Non-Line-of-Sight (NLoS) environment performances, localization refresh rate, transmission range, communication performances, energy consumption, and cost.

The earliest attempts on implementing an indoor positioning feature in a communication system have been done with 802.15.1 Bluetooth, with RSSI-based ranging.

Bluetooth is still widely used to this day, and several independent research works have reach sub-meter accuracy for static object localization. Due to the RSSI ranging-scheme, the capability of Bluetooth IPS to track moving objects is more limited as the accuracy significantly decreases; in the case of mobile objects, or in dynamic environments, the accuracy typically lies in an interval of 1~3 m [29, 39]. Bluetooth was originally designed for star network topologies; a stack for mesh networking has been released for Bluetooth 5 in July 2017. It still requires specific implementation work for multi-hop networks and is not the most off-the-shelf technology for such networks.

802.15.4 [6] solutions using Zigbee have also been proposed, based on RSSI; the main advantages are their low-cost and low-consumption. Also, unlike Bluetooth, ZigBee networks have mesh topologies (that are more efficient and practical than star topologies for a high number of devices). Besides that, it does not have the same level of accuracy. In [30], a precision of 2.8 meters has been reached.

The most prominent beacon-based technology in indoor environments is probably RFID. RFID tags can be either passive or active. Passive tags are a very efficient solution for unpowered nodes, and previous works have implemented passive UHF (860 ~960 MHz) RFID tags localization systems [40]. Because of the lower frequencies involved compared to Bluetooth and WiFi, which give better penetration, RFID deals better with multi-path environments than Bluetooth does. Nevertheless, passive tags can initiate communications and are at the edges of the scope of this comparison. Regarding active RFID devices, the Landmarc system [41, 42], one of the most prominent works on RFID localization, gets an average precision of about 1 meter with an RSSI-based method with 915 MHz active RFID tags. It suffers mostly from the fact that exact RSSI measurements are not available on RFID tags, as the RSSI levels have only a resolution of 3 bits on the devices used. Thus, a combination of RSSI ranging and K-nearest neighbors [43] methods are used in Landmarc. Because RFID communications are based on Time-Division Multiplexing (TDM), getting reliable position estimations requires around 7.5 seconds for 500 tags, as the number of collisions is high in indoor environment. That shows that the refresh rate of the positions is pretty low. Trying to increase this rate affects the collision robustness of the system and as a

consequence, reduces its reliability. Moreover, as the positioning algorithm relies on K-nearest neighbors, the tag density needs to be high enough to make the system functional.

One of the communication technologies that have been the most studied for positioning purposes for the last decade are the Wireless Local Area Network (WLAN) standards, and more specifically Wireless Fidelity (WiFi) [13]. Accessing the users' positions is indeed extremely useful for a wide variety of applications: statistical analysis of point of interest in public places, e.g., optimization of the emitting antenna orientation to increase the data rates, or location-based access. There are research works on both signal phase-based and signal attenuation-based positioning on WiFi. AoA estimation works great in a Line-of-Sight (LoS) environment and provides not only the position but also the orientation of the localized device, which can be useful for some applications. However, the number of antennas required for each node increases as the topology of the room gets more complex. An alternative is to extract the time-of-flight from the signal phase with the channel aggregation process aforementioned [37]; this is not possible on every commercial WiFi chipset, and it requires the channels in the 5 GHz frequency bands. In [37], an accuracy in tens of centimeters has been obtained on mobile nodes.

RSSI-based WiFi IPS have a typical precision around 2~3 meters[44]; however, later researches using CSI get the same level of precision as AoA, i.e., a few decimeters accuracy, while being able to work in the presence of obstacles. Because of the learning phase, it gets limited in environments with a dynamic topography: humans moving around or furniture displacements will affect the room's RSSI profile. Moreover, it has been proven in [45] that different WiFi chipsets could provide different RSSI measurements for the same room, which implies that the same device should be used for both the learning process and the localization.

Ultra-Wideband (UWB), which came out in 2002 and has been used in several 802.15.x standards, is a promising technology for localization. The UWB physical layer defined in 802.15.4 [6] is tailored for highly accurate timestamping. UWB is based on the frequency range from 3.1 to 10.6 GHz, with a minimum bandwidth of 500MHz for each channel. Among the different modulation techniques available, Impulse Radio UWB (IR-UWB), based on very short energy pulses, allows reaching very high time resolutions (pulse have a width of 2 ns). Designed for Wireless Sensors Network (WSN), it is intended for low-cost and low-energy systems. Large bandwidth also means important data rates: UWB throughput can go up to 27 Mbps for 802.15.4 physical layer. Because of its high time resolution, UWB can achieve better precision than regular communication technologies with time-of-flight, even if AoA and RSSI [33] ranging are also possible. Latest works on time-of-flight approaches can provide a sub-decimeter precision even in the presence of obstacles [46]. Moreover, the positions update rate can exceed 1000 Hz, which is far above any other technology as they need between 0.1 and 10 seconds to refresh the positions.

The performances and characteristics of each technology are summarized and compared in **Table 1**. It is difficult to compare benchmarks that have been evaluated in different environments, considering that indoor positioning is so dependent on the quality of the

environment. As a consequence, this comparison aims rather to give an order of magnitude and a qualitative description of the performances obtained in the literature rather than precise benchmarks, which make little sense in this context.

Table 1. Comparison of the positioning performances of various radio technologies

	Accuracy		**Refresh rate**	**Range**	**Communication performances**	**Consumption**	**Cost**	**Typical use case**
	LoS	**NLoS**						
Bluetooth (*RSSI)* [29, 39]	60 ~100 cm	> 1.5 m	0.1 ~1 Hz	~30m	Less optimized for multi-hop systems	Transmission ~40 mA, rest 0.2 mA	2-3$	**Good accuracy on small tag population Interaction with smartphones, connected wearables, etc.**
Active RFID *(RSSI)* [42]	60 ~100 cm	> 1 m	0.1 ~ 10 Hz	~90m	Extremely low data rates (~1 kbps)	~0.4 µA (for active tag)	2~3$	**Budget & low-power system with little data exchange between tags**
ZigBee *(RSSI)* [30]	2 ~3 m	> 3 m	~1Hz	~30m	Low data rate (20 to 200 kbps)	Transmission ~20 mA, rest 3µA	~1$	**Home automation with low accuracy needs**
WiFi *(RSSI/CSI, AoA)* [26, 37, 47]	30 ~100 cm	> 1.5 m (CSI) [27] > 3 m (AoA) [48] ~1 m (Channel aggregation) [37]	0.1 ~1Hz	~30 m	Very high data rate (860 Mbps for 802.11ac wave 1)	Transmission ~200 mA, rest 12 mA	~2-3$	**Positioning nodes in public & open networks Smartphone positioning Low power-constraints applications**
UWB *(ToF)* [21]	~10 cm	< 50 cm	~1000Hz	10 ~ 30 m	High data rates (up to 21.6 Mbps)	Transmission 600 µA, rest, 15µA	~20$	**Highly performant IPS in industrial environment**

II.1.4 Positioning algorithms

II.1.4.1 Definition

There are different approaches to compute a position from a set of distances to reference points. The most intuitive one is to approach this multilateration problem from a purely geometrical perspective. At least two anchors are needed in a 2D space, and at least three in a 3D one. For the sake of clarity, the following figures are always represented in 2D; however, the related discussion can be extended to a 3D problem without loss of generality.

An example of a configuration with two anchors is shown below in **Fig. 2**. Two anchors A_1 and A_2 are estimating respectively the distances R_1 and R_2 to a tag T1. Geometrically, there are two possible trilateration solutions, which are symmetric over the axis (A_1, A_2). If there are not any other anchors available, the actual solution will often be extracted from the context. For example, the space below the axis (A_1, A_2) might not physically accessible, or the tag could observe environmental information indicating its presence in the upper half. Nevertheless, it is usually recommended to have at least one additional anchor to identify the right solution out of the solution pairs. This same problem also applies in three-dimensions: if only three anchors are available, two solutions, symmetric over the plan formed by the three anchors, will be geometrically possible. Some extra context information will be needed to find out the real position.

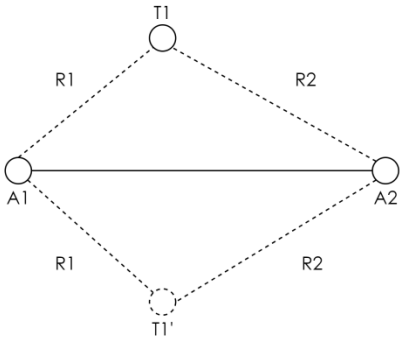


Fig. 2. 2D trilateration with two anchors

From **Fig. 2** above, it seems that the geometrical problem that multilateration has to solve is quite straight-forward. However, we have been ignoring so far the distance measurement bias. In any real-world IPS, ranging are noisy and their accuracy will vary depending on the technology used and the complexity of the environment. As a consequence, the multilateration algorithm has to compensate the individual ranging bias. This problem can be represented by the area of intersection or disjunction of circles, as shown in **Fig. 3**. In this example, three anchors A_1, A_2, and A_3 estimate their respective distances R_1, R_2, and R_3 to a mobile tag. The ideal geometrical case is shown on the left: the distances estimated are completely accurate and there is a single solution, which is the intersection of the circles of radius R_1, R_2, and R_3. However, this is not true anymore as long as a bias is introduced on the ranging outputs. If the anchors are overestimating the distance, as shown in **Fig. 3**. *(middle)*, the intersection of the ranging circles forms a surface and not a single point. In the opposite

case *(right)*, if the distances are underestimated, the ranging circles are disjointed. The area of the tag's potential position is circled in red. The surface of junction or disjunction will depend on the bias distribution. Considering this issue, the multilateration problem is less trivial than a simple geometrical resolution.

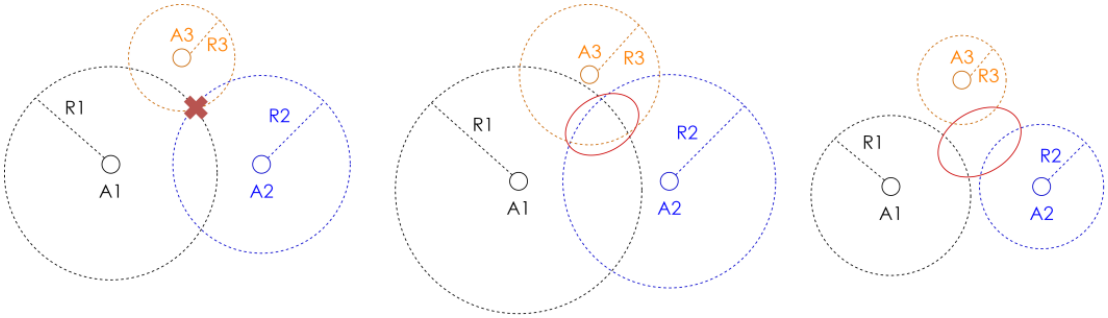


Fig. 3. Multilateration problem with different types of ranging bias: None *(left)*, over-estimation *(middle)*, under-estimation *(right)*

One parameter that is particularly relevant to solve this problem is the Ranging Deviation (RD). RD is the mean-squared error of a given position relative to the distances observed by the anchors, as detailed in the following definition.

Definition 1: For a set of anchors (A_1,…, A_N) measuring the distances (R_1,…, R_N) to a mobile tag T, let P = (x,y,z) a candidate position for T. With (D_1, …, D_N) the real distances from the anchors (A_1,…, A_N) to P, the ranging deviation RD_i of an anchor A_i of a position P is defined as:

$$RD_i(P) = (D_i - R_i)^2$$

The total ranging deviation (TRD) of P is defined as:

Eq. 1

$$TRD(P) = \sum_{i=1}^{N} (D_i - R_i)^2$$

RD is a good indicator on how coherent a given position is with the observed distances. A position with a low RD is more likely to be the real one. Also, for a given set of rangings, the minimum RD observed on the whole 2D or 3D space is a good indicator of the measurements noise. If even the minimum RD obtained is high, i.e., one cannot obtain a plausible solution, the distances obtained are not matching each other which means that the ranging measurements are being seriously degraded.

The last fundamental concept for multilateration algorithm is the idea of trajectory. The position is not only a spatial function but also a temporal one: two positions for the same

node recorded at short intervals are obviously strongly correlated to each other, as there are physical limits to the speed that a node can reach. This aspect is more application-dependent. Some knowledge about the typical speed that a node can reach will tremendously help, as there may be a slight difference between an IPS monitoring a cow breeding farm [49] and a one monitoring a kart fleet [50]. Also, a real-time localization system (RTLS) will typically run continuously and record trajectories, where some IPS might be solicited only occasionally, for example, to find a tagged object in a warehouse. In that last case, nodes are typically still and there are huge time gaps between two localizations, which means that the trajectory cannot be exploited. As a consequence, when the application context allows it, a memory effect can be used for the positioning algorithm, where the position obtained at a given time by multilateration can be filtered based on the previous ones.

II.1.4.2 **Multilateration algorithms**

Considering all the aspects aforementioned, there are three main types of multilateration algorithms:

- **Geometrical,** where the solution is extracted from a set of trilateration similarly to the case illustrated in **Fig. 2**.
- **Iterative,** where a solution minimizing the ranging deviation is obtained heuristically by series of iterations in an optimal direction.
- **Stochastic,** where the surface covered by the IPS is modeled by a probability distribution of the tag's potential presence. This probability distribution is usually obtained from the correlation between a position predictor and the ranging outputs.

In the following, we give an overview of the major approaches of the state-of-the-art for these three types of approaches.

Geometrical: geometrical approaches are simply based on the trilateration process illustrated in **Fig. 2**, where a set of 2 solutions can be obtained for each peer of anchors. In an ideal situation where the measurements are absolutely unbiased, a third anchor is enough to find out which of the two solutions is correct, and the distance information brought by any additional anchors will be completely redundant. In other words, if multiple anchors are available and there is no ranging bias, any triplet of anchors will give the same trilateration result. However, this does not hold true anymore if noise is introduced: the solutions obtained for each couple or triplet of anchors will slightly differ. When multiple anchors are available, there are two main ways to exploit the redundancy: either by using a selection process or by applying a centroid. In the first case, the closest anchors to the target, i.e., the ones that are the most likely to be accurate, are chosen and the other ones are ignored. This approach is used for example in the BAST algorithm [51]. In the second case, all the possible trilateration are computed and the final solution is computed as the centroid of all the trilateration solutions. This approach is used by Lazaro et al. in [52] for breast tumor detection with an UWB radar: a multi-static radar based on multiple UWB antennas is proposed, which estimates the reflection time to the tumor; as the propagation coefficient in

the monitored human body varies from one antenna to another, introducing a unknown bias, a weighted centroid approach is proposed to estimate the tumor's location.

Iterative: The goal behind iterative methods is to find a position minimizing the ranging deviation with a minimum effort. Bruteforcing the RD optimization problem by computing the RD for every point of the monitored space for a given granularity has a huge computational cost. However, it can be approached with conventional heuristics for least-square optimization problems, notably gradient search and Gauss-Newton algorithm [53]. An intuition for these two approaches is the following: given an arbitrary starting point, they find the direction that optimizes the RD reduction problem, iterate in that direction, and repeat the process until a satisfying solution has been obtained or the maximum number of iterations has been reached. However, the choice of the optimal direction is defined differently for the two methods.

For gradient search, the direction chosen is the opposite direction to the gradient. The iterations are defined as following:

Definition 2: Let P_0 an arbitrary starting position. With μ a positive coefficient, a single iteration of gradient descent is defined by:

$$\forall n \in N\ , P_{n+1} = P_n - \mu \nabla TRD(P_n)$$

The choice of μ is mostly a matter of compromise. If μ is too high, the gradient descent may tend to go always beyond the targeted minimum which would trigger a "zig-zag" effect. If μ is too low, the algorithm will tend to repeat several iterations in the same direction that could have been done in a single step. There two main things to retain regarding its accuracy and efficiency:

- it is based on the global ranging deviation and not the individual ranging deviation of each anchor
- it is an approximation of first-order

On the other hand, the direction is computed from the Jacobian for the Gauss-Newton method. In that approach, the iteration steps are defined as follows:

Definition 3: Let P_0 an arbitrary starting position. A single iteration of the Gauss-Newton algorithm is defined by:

$$\forall n \in N\ , P_{n+1} = P_n - \left(J_{RD_n}^T . J_{RD_n}\right)^{-1} . J_{RD_n}^T . TRD(P_n)$$

Where J_{RDn} is the Jacobian of the ranging deviation:

$$J_{RD_n} = \begin{pmatrix} \frac{\partial RD_1(P_n)}{\partial x} & \cdots & \frac{\partial RD_1(P_n)}{\partial z} \\ \vdots & \ddots & \vdots \\ \frac{\partial RD_N(P_n)}{\partial x} & \cdots & \frac{\partial RD_N(P_n)}{\partial z} \end{pmatrix}$$

Contrary to Gradient descent, Gauss-Newton is based on a second-order approximation. Because of that, it typically converges much faster than gradient descent, but convergence is not guaranteed [53]. Usually, for both gradient descent and Gauss-Newton, it is advisable to use a position extracted from the aforementioned geometrical methods rather than a random point. For an IPS that is running continuously, an alternative is to use the last recorded position as starting point, as the current position might me assumed to be very close from the previous one.

Stochastic: Contrary to the two previous categories, stochastic approaches are based on a memory effect, which means they take into account not only the current ranging but also the previous positions recorded. The principle is based on visualizing the whole map as a probability distribution of a given tag presence at a given time. Indeed, a given position might be coherent from a ranging perspective (i.e., has a low TRD) but incoherent from a trajectory perspective (i.e., implies an unrealistic speed or acceleration). Stochastic methods aggregate these two aspects by defining the presence probability distribution as a combination of these two factors. Probabilistic positioning methods in the state-of-the-art are mostly variations of particle filters [54, 55].

In the context of localization, the goal of particle filters is to find the position a node by correlating the knowledge that the IPS has of the environment to the current and past observations of the nodes within that environment. A good example is the case of an airplane flying over an environment: the airplane needs to locate itself on a map; the only parameter that it can sense is its altitude, but the relief on the whole map is known. The plane is going in a straight direction at a steady speed. With a single measurement, there are probably multiple points on the map that match its observed height. However, once the altitude measurements start being accumulated, they will form a pattern that can only possibly match a single portion of the map, allowing the airplane to locate itself.

In the case of the discussed IPS, the environment data that are known are the anchors' position, and the parameter observed is the tag-anchor distance. The principle of a particle filter applied to a time-of-flight positioning problem is illustrated below in **Fig. 4**. The particle filter starts by generating a fleet of N particles with random positions across the map *(***Fig. 4**., *left)*. Then, it associates likeliness to each of these points, based on their TRD with the current ranging outputs of the anchors: particles with a low TRD have a high likeliness, and vice-versa. The particles are then resampled: particles with high likeliness are duplicated, and particles with low likeliness are eliminated *(middle)*. Finally, each particle is moved by a vector randomly sampled from the space of realistic displacements *(right)*. This space is determined from the context information, e.g., maximum realistic speed or acceleration. The likeliness is typically estimated from an Extended Kalman Filter [56].

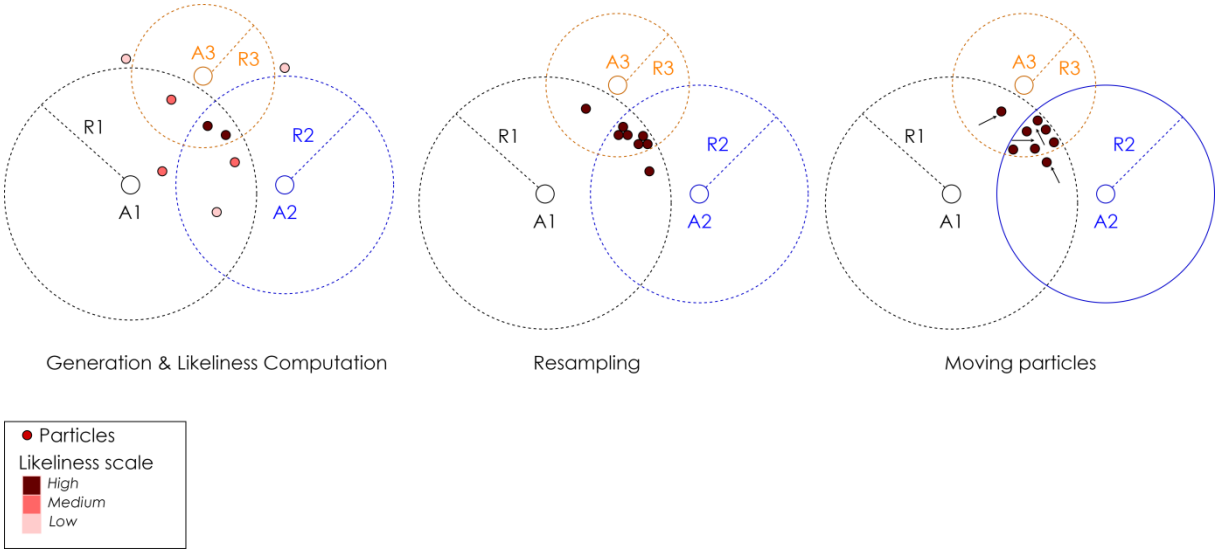


Fig. 4. Particle Filter Overview

Kalman filters can also be applied to post-process the positions obtained by iterative and geometrical multilateration. As these methods do not take into account the evolution over time of the position, IPS based on these methods that are running continuously can indeed increase their accuracy by applying a speed estimator-based filtering on the calculated positions.

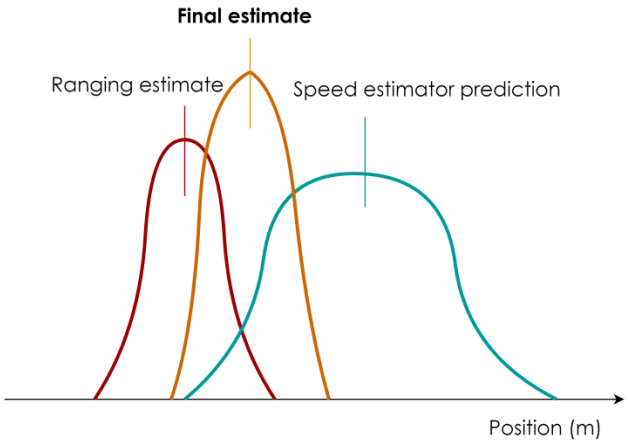


Fig. 5. Principle of Kalman filter

The principle of Kalman filters is shown above in **Fig. 5**. The figure illustrates a 1D example for clarity, where the filter must estimate the position on a single axis, but the principle is the same for 2D or 3D. The ranging estimate is the measured position extracted by the multilateration process. In the case of iterative or geometrical methods, that would be the final position returned if no further processing was applied. This position is represented as a probability distribution, here a Gaussian distribution centered on the ranging estimate. A simple example of predictor is shown on the right, based on a speed estimator: the speed is estimated from the previous positions recorded, and the position shift after each localization

is computed assuming a constant speed. The Probability Density Function (PDF) of the previous final estimate is then shifted by the computed value. Note that the modelization parameters for the ranging estimate distribution depend on multiple parameters, including the nature of the estimator, context knowledge, and empirical results. Then, the product of the two PDF is computed and the final estimate is returned as the position with the highest probability.

The choice between these different approaches usually depends on the application context, notably the following points:

1. Computation power available, which is often a concern in decentralized systems where the mobile node has to compute its position itself with a cheap microcontroller. The highest the localization frequency is, the highest the computation load will be.
2. Continuity and frequency of the localization service, i.e., whether the IPS is keeping a continuous track of the mobile node or not, and at which sampling frequency.
3. Presence of sensors giving alternative information on localization, e.g., on the motion of the node (accelerometer) or on the surrounding environment (vision, local temperature, etc.).
4. Ranging redundancy, i.e., how many anchors are involved in the multilateration process.

We summarize the strength and weaknesses of each approach below.

Geometrical methods are computationally efficient and very effective on a small set of anchors. However, they do not scale up very well on wider anchors population as the number of potential trilateration is quadratic to the number of anchors, which makes the process quite inefficient at dealing with the information redundancy.

Iterative methods have a higher computation cost, but they are tailored to solve multi-dimensional problems efficiently and tackle better with high anchor densities. They can exploit alternative sensors to orientate the iterations more efficiently, although it will have an impact mostly on the computation power rather than the actual accuracy. Both iterative and geometrical methods do not exploit the continuity of the trajectory and will require additional position filtering on top of them, e.g., Kalman Filters. Stochastic methods, on the other hand, are based on this very aspect as they have a memory effect. They are also efficient at aggregating different kinds of position-related information as these indicators can be aggregated in the presence of likeliness estimator. However, their computation cost is high. They also tend to be weaker if the IPS designer has no clue about the nature and properties of the objects that are tracked.

Table 2. Pros and cons overview of the main multilateration approaches

	Power-constrained positioning	Continuous, high-frequency positioning	Exploitation of alternative sensors	Exploitation of ranging redundancy
Geometrical	++	-	-	-
Iterative	-	-	+	+
Probabilistic	--	++	++	+

II.1.5 802.15.4 UWB, a technology tailored for Indoor Positioning

802.15.4 UWB-IR is currently by far the most performant radio technology for indoor positioning, as shown in **Table 1**. We detail in the following the physical and Medium Access (MAC) layer of 802.15.4 UWB.

II.1.5.1 Physical layer

UWB has been introduced in 802.15.3 2003, with a physical layer designed for high throughput. A physical layer design for Time-of-Flight (ToF) ranging has then been proposed in IEEE 802.15.4 2003 [6]. This physical layer has been further defined in the amendment 802.15.4a in 2007, which has been merged into 802.15.4 in 2011[1]. Currently, UWB is mostly known for its positioning capacity with the 802.15.4 physical layer. The large bandwidths involved and the impulse-radio (IR) communication scheme allow highly accurate Time-of-Flight ranging with very little delay and good robustness to multipath effects.

Two modes are defined in the standard for the physical layer, High/Low Rate Pulse Repetition Frequency (HRP/LRP). HRP has been designed to support accurate ToF measurements and is intended for localization applications. As a consequence, we focus on HRP mode in this manuscript.

Decawave is currently the leader on the 802.15.4 UWB market; the DWM1000 [7], one of their UWB transceivers, is currently by far the most widely spread UWB transceiver in off-the-shelf indoor positioning systems. DWM1000 chips came out in 2014 and implement the 802.15.4-2011 physical layer. The research work in this manuscript is also based on DWM1000; as a consequence, we also give occasionally details in this section on certain implementation details of 802.15.4 PHY on DWM1000.

Note: In the following, we refer several times to the synchronization between an emitter and a receiver in the context of transmissions. Here, the term synchronization does not refer to them sharing the same clock reference but rather implies they are timestamping the same event (e.g., they have the same reference for the leading peak of a given frame). The transmitter and receiver do not actually need to be synchronized from a MAC perspective to perform a ranging protocol, as we will see later.

[1] For that reason, 802.15.4-2011 is also often referred to as 802.15.4a.

An UWB frame is formed by a preamble, a Start Frame Delimiter (SFD), a physical header (PHR), and a payload, as shown in **Fig. 6**. We define them in the following.

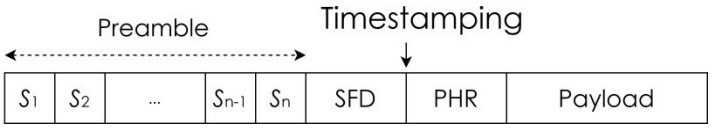


Fig. 6. UWB Frame Format

Preamble: Each transmission starts with a preamble, which is a sequence formed by a symbol repetition. This symbol consists of a sequence of polarized pulses, forming the so-called preamble code with a ternary alphabet {-1; 0; +1}. Two Pulses Repetition Frequencies (PRF) are available, 16 Mhz and 64 Mhz. The time length of a symbol is roughly 1 μs (997 ns for 64 MHz and 1003 ns for 16 Mhz).

Multiple preamble codes are proposed in 802.15.4. The transmitter and receiver have to agree on the preamble code to communicate successfully; for two devices of the same manufacturer it will be pre-programmed in the firmware most of the time, but if this information is not known, the receiver can simply try the different preamble codes successively as they are only two per channel/PRF combination. For each channel/PRF combination, the two preambles proposed in the standard have been designed to have a low intercorrelation so that they can be used simultaneously with reduced collisions.

The receiver typically detects the preamble by cross-correlating in chunks: on the DWM1000, the length of a chunk can be set from 8 to 64 preamble symbols. Larger chunks tend to give better performances but may lead the receiver to miss frames with short preambles; hence, this parameter should be set according to the defined preamble length.

The preamble length can be set from 16 to 4096 symbols[1]. The choice of the preamble length depends on the quality of the link, notably the distance and the harshness of the environment (Line-of-Sight conditions). Longer preambles are more reliable but induce a lower data rate and a higher consumption per frame.

Given that the preamble is based on a symbol repetition, the receiver does not need to receive all the preamble symbols: the preamble is designed for synchronization purposes and does not carry information. Since the cross-correlation is done in chunks, the first symbols are typically lost as they will start in the middle of a chunk. For that reason, it is difficult to retrieve the exact number of elapsed symbols when the preamble is detected by the receiver [11]. The emitter might also increase or decrease the preamble length without disrupting the communication. Hence, the receiver does not need to get the exact symbols count to identify the end of the preamble; this is actually notified by the Start Frame Delimiter (SFD).

SFD: The preamble is followed by the SFD. The SFD symbols are drawn from the same ternary alphabet as preamble codes, but the symbol set is different, so it can be discriminated from the preamble. The SFD is a crucial element in a ranging protocol, as the end of the SFD

[1] Preamble length of 16 symbols are not supported by DWM1000 chips

is the event timestamped by both the emitter and receiver. This means that in a ranging protocol, the timestamps recorded for each frame do not correspond to their start but to the end of the beginning of their PHR (which follows the end of the SFD): preamble starting time can be retrieved by subtracting the preamble and the SFD duration the timestamp recorded, assuming that the emitter did respect the pre-agreed preamble length, which is difficult to verify given the imprecision of the preamble symbols count aforementioned. This is a very important consideration in the later security analysis, as this aspect is sometimes partially neglected in MAC-level security analysis [57].

The length of the SFD depends on the data rate of the payload: 8 symbols for 110 kpbs and 64 symbols for higher rates. The SFD is followed by the data segment of the frame, which is modulated with Binary Pulse Position Modulation (BPPM). Thus, the receiver switches to BPPM demodulation after timestamping the SFD to receive the Physical Header (PHR).

PHR: the purpose of the PHR is to notify the receiver of the parameters used for the transmission of the payload, which are essentially the data rate and the payload length. The receiver can extract from the SFD length whether the emitter is using a low data-rate (8-symbols SFD) or a high data-rate (64 symbols SFD). For a low data-rate (110 kbps), the PHR is also sent at a data rate of 110 kbps; for a high data rate (850 kbps or 6.5 Mbps), the PHR is always sent at 850 kbps. The PHR has a length of 19 bits and is immediately followed by the payload.

Payload: Three data rates are available for the payload, 110 kpbs, 850 kpbs, 6.5 Mbps and 27 Mbps. The length of the payload shall not exceed 128 bytes according to the standard[1]. Note that the data rate applies only to the payload, and does not consider the extra time required for the preceding preamble, SFD, and PHR. For example, for a preamble length of 1024 symbols and a payload data rate of 27 Mbps, the total throughput is about 14.4 Mbps, which is 41% lower.

II.1.5.2 **MAC layer**

Standard Ranging Protocols

From a localization perspective, the specific feature that provides the physical layer of 802.15.4 UWB compared to other wireless technologies is the capability to accurately timestamp the reception and transmission of frames. From this feature, ranging protocols can be built in the MAC layer.

Given that the speed of the radio signals is known, two nodes can measure their mutual distance if they can estimate the time-of-flight between them. However, it is difficult to estimate time-of-flight with a single frame for two reasons:

[1] DWM1000 chips support non-standard preamble length up to 1024 bytes.

- If only one timestamp is provided, the two nodes need to share the same clock reference; hence, they will need a clock synchronization protocol. This is quite a huge constraint in the context of WSN, and any slight inaccuracy in the synchronization protocol will generate high degradations in the overall accuracy of the ranging process.
- As defined in the physical layer description, the transmission timestamp is learned by the emitter while the frame is already being emitted; hence, the timestamp of a given frame cannot be embedded in its payload and has to be transmitted in the following frame. An alternative is to use a scheduling feature to target a specific pre-agreed transmission timestamp; this feature has been discussed in [58], and is actually available on the DWM1000. However, the accuracy of the scheduling process of the DWM1000 is lower than the one of the timestamping process, as we discuss later in section III.3.2.

Considering these two issues, conventional ranging protocols are built upon round-trip time-of-flight estimation: in a round-trip configuration, the offset between the respective clocks of the two nodes is compensated and there is no need for a synchronization protocol. Also, they often include an additional frame to send the measured timestamps: by doing so, they avoid relying on a scheduling process and increase the overall accuracy.

Two ranging protocols are proposed in 802.15.4, the **Two-Way Ranging (TWR)**, and **Symmetric Double-Sided Two Way Ranging (SDS-TWR)** protocols. They are both shown in **Fig. 7.**

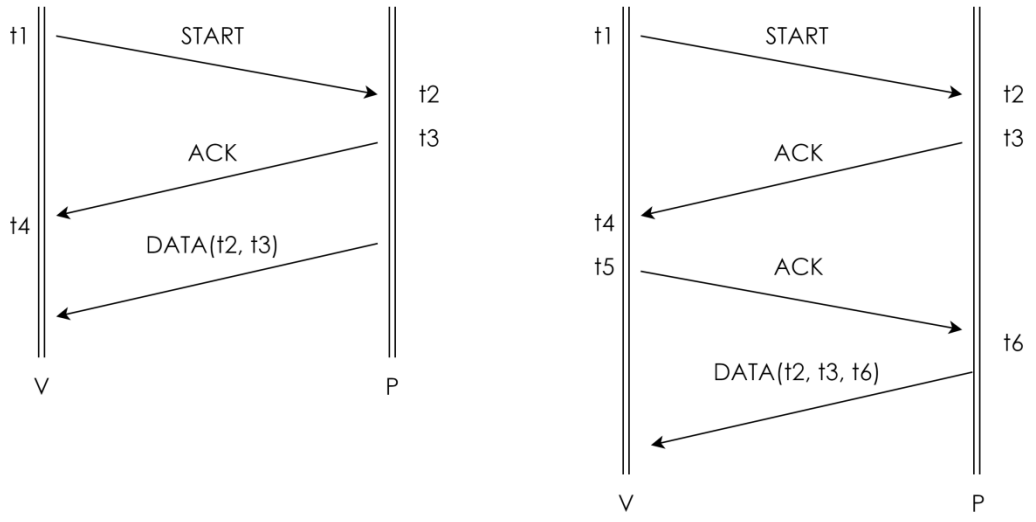


Fig. 7. Two-Way Ranging (TWR) *(left)* and Symmetric Double Sided-TWR *(right)* ranging protocols

In a TWR protocol, the verifier V initiates the protocol by sending a START frame at t1 to a prover P, which receives the frame at t2. Then, P proceeds to acknowledge at t3; the acknowledgment (ACK) is received by V at t4. Finally, P needs to transmit its timestamps to

V, as they are not synchronized. Thus, P proceeds to send a DATA frame containing t2 and t3. Note that sending only (t3 – t2) is enough. Indeed, with *c* the speed of light, the distance is calculated as:

Eq. 2

$$d = c * tof(P \rightarrow V) = c * \frac{(t4 - t1) - (t3 - t2)}{2}$$

Basically here, the round-trip time-of-flight is the difference between the total time waited by V, *(t4 – t1)*, and the reply time of P, *(t3 – t2)*. Considering that the protocol can be completed within a very short time span (<10 ms), the distance variation between the start of the end of TWR is negligible, and the distance between the two nodes can be extracted as half of the round-trip time multiplied by the speed of light. TWR compensates for the offset between the prover and verifier respective clocks, but does not compensate their difference of speed. A crystal oscillator has a certain finite crystal tolerance, and there is always a slight error between the nominal frequency and the real frequency of a given clock, referred to as clock **drift.** The difference of drift between the respective clocks of two different devices is called clock **skew** in the context of synchronization protocols. Clock drift and clock skew are typically expressed in parts per million (ppm). For the estimation of a given period t, a clock with a drift *e* gives the following biased time estimation:

$$\hat{t} = (1 + e)t$$

Hence, if P and V have the respective clock drifts e_p and e_v, the error introduced by the clock skew on the distance estimate $\hat{d} - d$ is given by [59]:

Eq. 3

$$\hat{d} - d = d * e_v + (t3 - t2) * c * (e_P - e_V) \approx (t3 - t2) * c * (e_P - e_V)$$

The distance-dependent part is negligible as the drift is in the order of magnitude of a few ppm, hence the approximation in the right part of the equation. However, the part that depends on the reply time *(t3 –t2)* can lead to tremendous errors and depends on the clock skew between the two devices *($e_v – e_p$)*. For instance, a clock skew of 10 ppm will generate an error of approximately 1.75 m for a reply time of 1 ms, which is far above the advertised 10 cm accuracy of UWB. Because of the clock skew, the estimation of *(t3 – t2)* from P might be slightly biased in V's time referential.

Considering this issue, SDS-TWR has been proposed in 802.15.4 to allow the two nodes to compensate their mutual skew. The two first frames of SDS-TWR are exactly similar to TWR. The difference with TWR is that instead of sending the final DATA frame straight after the ACK, P waits for V to acknowledge as well. V's acknowledgment is sent at t5 and received by P at t6. Doing so, the ranging protocol becomes symmetric, as both sides get to execute a TWR protocol. In that case, the distance is given by [60]:

Eq. 4

$$d = \frac{[(t4 - t1\,) - (t3 - t2)] + [(t6 - t3) - (t5 - t4)]}{4} = \frac{t6 - t5 + 2(t4 - t3) + t2 - t1}{4}$$

For SDS-TWR, the error the induced by the clock skew between the two devices is proportional to the difference between their respective reply times and is given by [59]:

$$\hat{d} - d \cong (e_v - ep) * \frac{[(t3 - t2) - (t5 - t4)]}{4}$$

Hence, SDS-TWR is optimal when the reply times of the two devices are equal. Although requiring an additional frame compared to TWR, SDS-TWR reduces by the error induced by the clock skew between the two devices. Also, in a configuration where both devices need to learn their mutual distance, SDS-TWR lets both parties compute the distance by simply using an additional final DATA frame from V to P containing *(t4, t1, t5)*. In that case, the two nodes obtain the distance with a total of 5 frames, where they would have needed 6 with two successive TWR.

Proposed ranging protocols in the literature

As far as TWR and SDS-TWR are proposed in 802.15.4, there is no obligation for standard-compliant devices to restrain to use of these two protocols only. Although TWR variants are the most ranging protocol common in commercial UWB system and off-the-shelf indoor RTLS (e.g., [61][62]), multiple alternatives have been proposed in the literature. We provide below a non-exhaustive list of some of these protocols, that we classified in three categories.

1. TWR/SDS-TWR variants:

Several works have proposed modifications to the standard TWR and SDS-TWR protocols to improve their performances.

Regarding TWR, in [58], the authors propose to exploit the scheduling feature (or *delayed send*) of the DWM1000 to get rid of the DATA frame in the TWR protocol. With that scheme, the prover and verifier agree on a reply time t_{reply} before the ranging protocol, and the verifier acknowledges at t3 = t2 + t_{reply} using the delayed send feature. In that configuration, the verifier already knows *(t3 – t2)* = t_{reply}, and there is no need for DATA frame, leading to a 2 messages-TWR (2M-TWR) protocol. We refer to this protocol as *lightweight TWR (LTWR)* throughout this manuscript. Note that this approach is more lightweight yet less accurate than regular TWR: we demonstrate later in section III.3.2that the scheduling process is not as accurate as the timestamping process on the DWM1000, which leads to a lower average accuracy on LTWR.

Concerning the clock drift correction problem, it is possible to compensate the error by estimating the clock skew between P and V during the ranging protocol. Indeed, as shown in Eq. 3, the error induced by the clock drift depends mostly on their relative skew. Two methods are provided on DWM1000 to estimate the clock skew with another device [63]; in [64] the authors exploit one of them to introduce a skew compensation in the distance calculation for TWR, which gives:

Eq. 5

$$d = \frac{(t4 - t1) - (1 + \sigma)(t3 - t2)}{2} \quad with\ \sigma = \left(e_v - e_p\right)$$

We will see in section III.3.1the basics of these two skew estimation methods and their difference of performances.

Regarding SDS-TWR, in [65], the authors propose a novel distance calculation method for SDS-TWR, which does not require doing any modification to the protocol. They demonstrate that the distance can be computed as[1]:

$$d = c * \frac{(t4 - t1)(t6 - t3) + (t3 - t2)(t5 - t4)}{t6 + t5 - t2 - t1}$$

With that calculation method, the error induced by the skew is given by:

$$\hat{d} - d = e_V * d$$

Given the typical value of the crystal clock drift, this timing error is negligible as it will be largely below 1 cm. With that approach, the clock drift compensation does not rely on the reply times of both devices being equal and provides higher performances while simplifying the requirements for the protocol.

2. Sequential ranging protocols:

Standard ranging protocols allow the two nodes to obtain a single observation of their mutual distance. As this distance estimation is biased due to the noise induced by the environment, the accuracy of the estimation can be largely improved if multiple observations are aggregated, even if the two nodes remain still. Thus, even in an ad-hoc network where the ranging/localization feature might be solicited only intermittently, it is recommended to use a sequence of ranging protocols instead of a single one, unless the power constraints are extremely high. In an RTLS, there is typically a continuous ranging stream, with a certain associated localization frequency. That being considered, it makes sense to design ranging

[1] The equation has been adapted to the timestamp notation proposed in this manuscript

protocols that are optimized for sequential use. In the following, we give a short overview of various protocols that have been designed for this purpose.

The intuition behind these protocols is simple: since data frames are not timestamped and are only required to transmit the timestamps measured for the previous frame, one can simply get rid of them if multiple ranging protocols succeed each other within a very short lapse of time: the timestamps of a ranging n can be transmitted in a START or ACK[1] frame of the next ranging (n+ 1), which avoids dedicating a frame to timestamps transmission. This is somehow equivalent to replace the DATA frame in **Fig. 7** by the START frame of the next ranging embedding the timestamps of the previous one. The SDS-TWR Multiple Acknowledgments (SDS-TWR-MA) proposed in [66] apply this very principle by performing a rapid succession of N SDS-TWR rangings where the DATA frame with the timestamp of the previous ranging replace the ACK frame; hence, only 2 frames are needed for each distance acquisition.

The authors in [67] propose a "Burst"-based approach, called Burst Mode SDS-TWR (BM-SDS-TWR) in which the verifier sends a succession of k ranging requests, followed by a reply of k acknowledgments from the prover. Thus, instead of using a strict alternation of requests and acknowledgments, the approach proposed is to send all the requests at once, and reply with all the acknowledgments at once. SDS-TWR-MA and BM-SDS-TW have very similar performances; as for the same total number of frames n, with $t_{reply}(V_i)$ and $t_{reply}(P_i)$ the respective reply times of V and P at the iteration i, that gives the following clock drift-induced error:

$$\hat{d} - d = \frac{c}{4} * \sum_{i=1}^{n} \left(e_p - e_v\right) \left(t_{reply}(V) - t_{reply}(V_i)\right)$$

The clock skew is assumed to be constant in both works, and the efficiency of the proposed protocols in the case of a varying skew is not discussed. With the assumption of a static skew, the two protocols have similar skew correction performances.

II.1.5.3 **Multi-party ranging protocol:**

So far, we have only studied ranging protocols from a peer-to-peer prospective. Yet, a localization system requires multiple anchors to provide a valid position estimate in a 2D or 3D space; as a consequence, a tag should perform rangings with multiple anchors to get localized, and not just a single one. Considering that, several approaches have been proposed in the literature to define ranging protocols that are efficient in a multi-party configuration.

[1] Note that embedding timestamps in an ACK frame is not standard-compliant, as acknowledgment are not supposed to carry a payload. However, this is completely possible from a physical perspective, as acknowledgments are managed the same way as other types of frames.

The intuitive approach when estimating the distances of a mobile tag M to N anchors $\{A_1, \ldots, A_N\}$ is to cycle through the anchors with a sequence of ranging protocols. In [68], a cyclic succession of SDS-TWR protocols, called Sequential SDS-TWR (SSDS-TWR) is proposed. This approach allows each tag-anchor distance to be sampled at the same frequency but does not optimize the number of frames required for the process as it is a sequence of peer-to-peer protocols. In order to define more efficient multiparty ranging protocols, different broadcast-based schemes have been proposed in past works.

Parallel Double-Sided TWR (PDS-TWR) [69], is a broadcast approach where a mobile tag M can perform the equivalent of N SDS-TWR protocols with N anchors with a total of $6 + 2 * N$ frames, instead of the 4N frames that are needed for N individual SDS-TWR protocols. In that configuration, the tag is the verifier in the ranging protocol, hence, is the one collecting the distances. Compared to regular SDS-TWR, all the frames transmitted by the tag in the protocol are broadcasted. The protocol is illustrated in **Fig. 8**. The tag starts by sending a broadcasted start frame, received by the N anchors; to avoid frame overlapping, all the anchors acknowledges after a different reply time, and the N acknowledgment timestamps are collected by the tag. Then, the tag sends a broadcast acknowledgment, timestamped individually by the N anchors; finally, the anchors send their timestamps in a data frame, again with sparse reply times to prevent frames from overlapping.

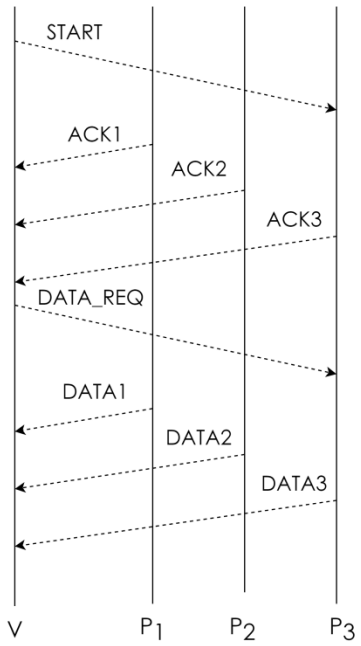


Fig. 8. Parallel Double Sided Two Way Ranging Protocol (PDS-TWR)

The whole protocol is equivalent to a succession of individual SDS-TWR with each anchor; yet, the tag spares the repetition of its START frame and ACK with that approach. The drawback compared to SSDS-TWR is that since the reply times are different for every anchor, the impact of the clock drift-induced error will vary from one anchor to another. To fix that,

the authors propose to use a skew compensation in the calculations based on the skew estimation provided by the DWM1000. There is still a slight loss of accuracy compared to SDS-TWR as the error in the skew estimation process will generate higher distances error on the anchors with high reply times.

Another example of a broadcast-based ranging scheme can be found in the Crazyflie project. Crazyflie is a drone developed by Bitcraze [62] tailored for academic purposes, which features notably an optional navigation feature based on a UWB IPS, the Loco Positioning System [70]. Three ranging protocols are implemented on Loco, including a TWR variant, and one of them is a broadcast-based Time Difference of Arrival (TDOA) protocol, as defined in [71]. This approach does not require transmitting any timestamp and is one-way as the drone does not need to reply. It is the most efficient in terms of bandwidth use, but leads to lower accuracies than the methods previously introduced. Instead of computing the absolute distance to each of the anchors, the tag computes the difference of distance within each anchor pair. For two anchors A_1 and A_2, the tag does not measure the distances d_{A1} and d_{A2} but instead $\Delta d_{A1,2} = d_{A2} - d_{A1}$. It is assumed that the time lapse between the respective transmissions of A_1 and A_2 is known by all parties, which means that the tag can retrieve the differential distance from the two timestamps. A toy example with 3 anchors gives the following. The mobile tag M embedded on the drone receives one frame from each anchor and compute the differential distances $\Delta d_{A1,2}$, $\Delta d_{A2,3}$ and $\Delta d_{A1,3}$. Thus, it can express each distance as a function of d_{A1}:

$$d_{A2} = \Delta d_{A2,1} + d_{A1};\ \ d_{A3} = \Delta d_{A1,3} + d_{A1}$$

Then, the value of d_{A1} can be retrieved during the multilateration algorithm. Since all the measured distances obtained are the sum of a measurement and an unknown component d_{A1}, the multilateration problem is not the resolution of circles intersection anymore but hyperbolas intersection [71]. Hyperbolas intersection resolution is more sensitive to measurement noise and more complex to solve. The resolution can be achieved by solving a least-square reduction problem on the parameter d_{A1}, but the computational cost is higher than classic multilateration and more biased by the noise. Hence, the TDOA broadcast scheme sacrifices accuracy for bandwidth efficiency: it has by far the lowest communication cost with only N frames needed to get the N anchor-tag distances but features an overall lower accuracy than other ranging schemes. This protocol is mostly recommended when working on large scales, i.e., when the number of tags and anchors is high (>20). It is also an approach that remains functional if the mobile tag can only receive and not emit.

Overall, the choice of the ranging protocol mostly depends on the size of the indoor positioning system and the localization frequency. TWR and SDS-TWR protocols are not the most efficient protocols in large-scale systems but feature good and homogenous accuracy performances with skew correction as all the anchors use the same reply time; also, they are also functional with low localization frequency unlike burst-based schemes. Considering that they are standardized and largely deployed in industrial applications, we focus on them in

the following chapters. An experimental comparison of TWR and SDS-TWR is presented In section III.3.2.

II.2 Security of radio Indoor Positioning Systems

II.2.1 General attack schemes against Indoor Positioning Systems

When it comes to the ranging security, the vulnerabilities prone to expose a ranging protocol to distance tampering depends largely on the physical parameter that is exploited for distance estimation and the specific vulnerabilities of the wireless radio technology used. As a consequence, the types of attacks that can be used against ranging protocols are very different from one technology to another. However, all the radio positioning technologies have in common their anchors/mobile tags model which has significant consequences on the security of the positioning process. Indeed, hijacking the position of a node (i.e., taking control over the position of this node) requires tampering the distances of all the anchors involved in the positioning process, regardless of the specific attack that is used against the ranging protocol. As a general rule, if an attacker wants to target a specific position he or she should tamper all the distances measured by the anchors to match that given position. In that case, the distance measured by each anchor should be either decreased or increased. In most IPS, the mobile tags wander in the area that is **within** the polygon formed by the anchors. Indeed, anchors are typically placed on the edges of the indoor environment; also, localization performances will be degraded for tags leaving the anchor polygon. A toy example is shown in **Fig. 9** for three anchors, forming a triangle. Any position shift induced by the attacker within the anchors' polygon cannot be solely based on only distance reductions or only distance enlargements, as this would not be geometrically possible. In **Fig. 9**, this real position of the tag is shown in green, and the position targeted by the attacker in red. When the position targeted is way outside the triangle, the attacker can get there using only distance enlargements. However, as we just stated, this would not be allowed in most IPS simply because of the physical boundaries of the monitored environment, and the attack detection would be trivial. If the position targeted is within the triangle, the attacker must use at least one enlargement and one reduction attacks. Hence, he or she must master both.

Distance-enlargements attacks and distance-reduction-attacks are often very different, and depending on the technology one might be much harder to mount than the other. In the case of UWB, we will see that the state-of-the-art largely differs for both types of attacks.

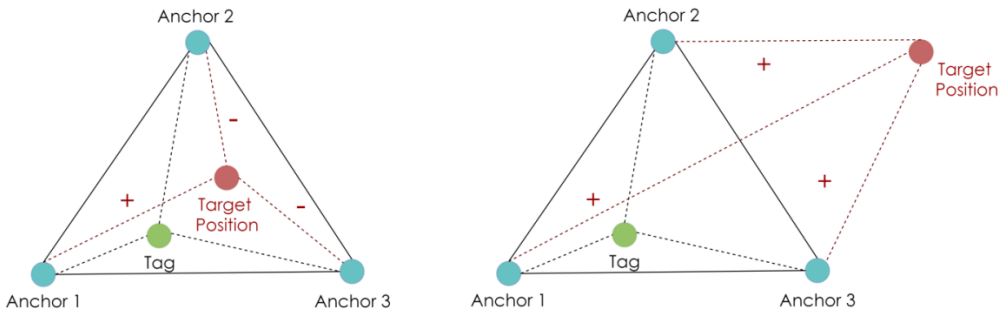


Fig. 9. The distance enlargement/reduction problem in a 3 anchors configuration

Another peculiar aspect of IPS security is despite being based on communication protocols, ranging protocols are closer to a sensing application than a communication one. Indeed, the valuable data in a ranging protocol is not so much the payload of the frames exchanged but rather the physical properties of these frames. These properties can be either, the signal received power, the signal phase, or the signal time-of-flight, depending on the ranging method. One major consequence in terms of security is that the critical data are carried by the physical layer, and not by the MAC and upper layers. In classic communication protocols, security is usually provided by cryptographic means. Privacy can be protected by encryption; integrity and authenticity, by digital signatures. Yet, cryptography does not provide any asset regarding the physical layer security: it does not prevent an attacker from monitoring physical parameters of the messages exchanged (e.g., received power, phase, Time-of-Flight…), nor does it detect modifications of these properties when they do not affect the payload. For example, in the case of RSSI-based ranging, if an attacker replays the frame of a victim node with a higher transmission power, the victim node will look closer to the anchor; yet, the attacker does not need to understand or modify the payload, as he or she is simply repeating the same message. Hence, classic cryptographic mechanisms cannot alone assess the integrity or authenticity of the positions measured.

Because of that, replay and relay attacks are two of the major threats against IPS security. In the context of a ranging protocol between a verifier V and a prover P, replay and relay are two different types of spoofing attacks: from V's perspective, it seems that V is performing is estimating to the distance to P, while it is actually estimating the distance to a malicious node M.

Replay attacks [72] are usually done in two steps, shown in **Fig. 10** *(left)*. First, the attacker starts jamming P such as hiding V from P and prevents P from receiving V's original message[1]. Meanwhile, the attacker is still listening V's messages, even though he or she is not able to decipher them. Then, the attacker replays the message on M, which means that V calculates the distance based on the physical parameters of M's message, which relate to M's position. Note that M can also easily tamper these physical parameters, e.g., M can increase its transmission power beyond typical values.

Relay attacks [73, 74] consist of creating an illicit link between two nodes that are out of range of each other, as shown in **Fig. 10** *(right)*. To some extent, they are a variant of replay attacks in which the frame is replayed to a node that should not even receive it in the first place. Thus, the jamming process is not even required. The existence of this link is unknown from V; the attacker simply uses M as a bridge between V and P, which means there is no need for M to decipher or modify the messages. From V's point of view, it looks like P is within its range.

[1] Replay attacks do not necessarily imply that the original frame has been denied, but in the case of an IPS not doing so will reveal the attack as both the original and tampered distances will be received. The replay scheme defined here is sometimes referred to as *jamming and replay*.

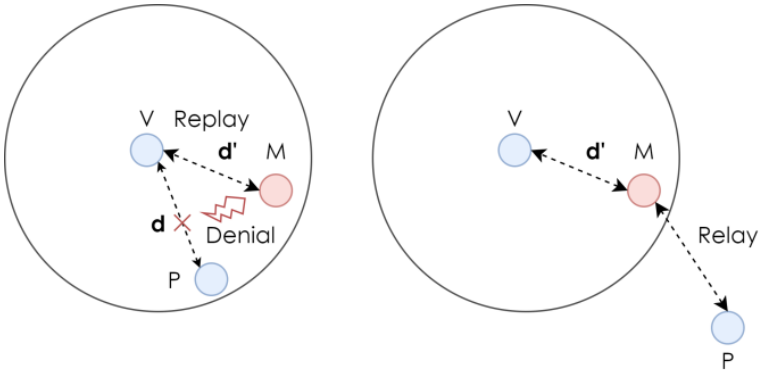


Fig. 10. Basic principles of replay and relay attacks

In most communication protocols, such Man-in-the-Middle (MITM) attacks are not harmful if proper encryption and authentication mechanisms are implemented. In that case, the attacker will indeed not be able to understand nor tamper the messages exchanged. The main impact of that unsolicited intermediates M is that the physical profile of the message received corresponds to the attacker and not to the legit expedient, which does not expose the payload if the payload is encrypted and signed. For an IPS on the other hand, the fact that the physical profile of the expedient has been spoofed is extremely critical as it is the main source of information for distance estimation. Thus, message authentication codes do not prevent from replay and relay attacks on most positioning technologies. However, time-of-flight ranging, and by extension UWB positioning, are more robust against these attacks than other ranging schemes. This is one of the major security features of UWB. Indeed, replay and relay attack induce delays on the reply times, which are typically much larger than the order of magnitude of time-of-flight on short distances; as a consequence, they typically generate absurdly large distances and can be easily spotted. We detail this aspect further in Chapter V.

II.2.2 System-level countermeasures for IPS: a brief survey

Different approaches to securing localization systems with high-level approaches have been proposed in the literature; we have classified them into four categories: statistical, sector-based, distance-bounding, and cooperative.

Statistical approaches aim to detect deviant behaviors by comparing the current locations to a forecasting statistical model. One of the first attempts of this approach can be found in [75]: Li et al. proposed to implement a median-based fingerprinting localization, which allows discarding the corrupted data sent by the malicious device as they typically diverge from the legit data. That ensures the reliability of the localization performed as long as the attacker is outnumbered, but it does not protect against an attacker than can alter the ranging measurements of honest devices. Following the same principle, a trust indicator is proposed in [76], which quantifies how deviant are the data provided by a given node. The concept of trust indicator has been proposed many times and under many different forms in the WSN security literature [77], for both safety and security applications. In a typical WSN,

sensors are generally not completely isolated from each other, which there is an information redundancy as multiple nodes might provide observations of the same environmental data. Thus, the reliability of the data perception of each node can be evaluated from how strongly it relates to the global perception. If one node deviates too much from the observations of the rest of the network, it can be assumed that this node is facing either security or safety issues. In the context of localization systems, all the anchors are sensing a common parameter, which is the position of one or multiple other nodes. This parameter is estimated from the distance between the station and the localized node, which depends on the station placement; yet, indicators such as the TRD give an estimate of the coherency of the distance returned by an anchor compared to the other ones. This approach has been extended to multi-hop positioning systems (i.e., IPS which coverage area exceeds the radio range) in Distributed Reputation-based Beacon Trust System (DRBTS) [78], in which a trust indicator for multi-hop sensor networks is proposed. In this model, every beacon monitors the other beacons within a 1-hop range. Deviant behaviors are determined by a voting scheme. Beacons with a too low trust indicator end up being discarded.

Sector-based approaches determine the intersection between the area of emission of different anchor nodes and define the location of the target node as the center of this intersected area. One of the major work on this topic is SerLoc [79], which is based on directive antennas, where the beam orientation is kept secret. Due to the directivity of the antennas, the map is divided into sectors formed by the complex intersection of the beams: each sector is a unique combination of beams, and the nodes within that sector will only hear the beacons a certain combination of anchors. Hence, if the position claimed by a node is in a different sector than the real one, either because the node is cheating or because it is victim of distance tampering, there will be at least one missing reply to a beacon or one unsolicited reply. As a consequence, the attack will be spotted by the anchors. Thus, position tampering attacks can only target the sector of the victim node to remain undetected. This principle is illustrated for 3 anchors in **Fig. 11** *(left)*. In that example, a victim tag T receives the beacons from $\{A_1, A_3\}$ in its sector. The malicious tag M is in another sector and receives $\{A_1, A_2\}$. If we assume that M has spoofed T and is replying in its name, A_2 will receive a reply that it should never have received as the position estimated for T (which actually corresponds to M) is not within its beam. A3, on the other hand, will never receive a reply. Hence, M can only spoof successfully T if they are located within the same sector, shown in green.

For that reason, the level of security brought by this approach depends a lot on the number of sectors and their organization: the narrower the sectors, the more secure it is. Statistical and sectors-based approaches are intended mostly for networks spanning over large areas, which could be found in indoor environments such as factories or malls. They usually reduce the computational burden for each node compared to other methods, as they are based on a centralized model where the stations run the major part of the security protocol. However, they require specific station placements to set up a proper sector configuration, and they reduce the localization performances as they exclude deliberately stations from the multilateration process.

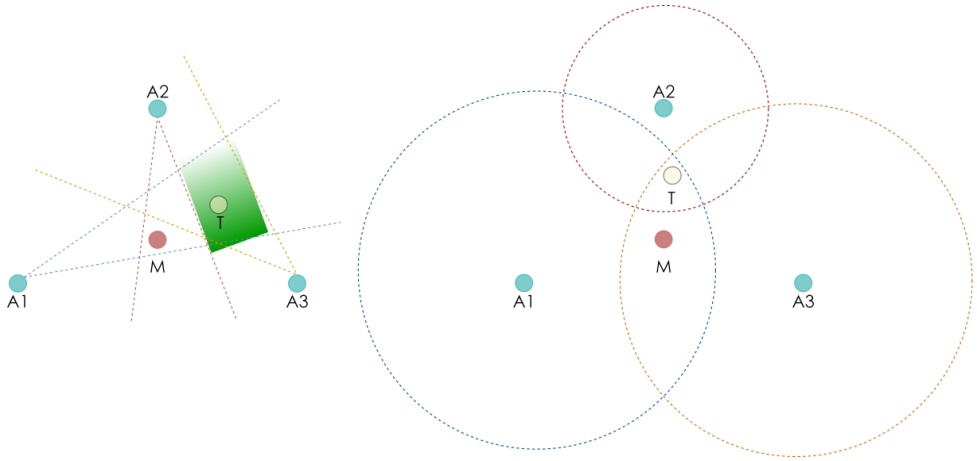


Fig. 11. Principles of sector-based and distance-bounding approaches

Distance-bounding (DB) approaches follow a similar intersection-based concept, but implement it differently than sector-based approaches. The primary goal of DB is to set an upper boundary on the distance measured based on physical observations. Often, this upper boundary is simply induced by the short-range of the technology used. Indeed, DB protocols are commonly implemented using short-range technologies like NFC or passive RFID [80]; some approaches have also been using audio signals to verify the presence of a node in a given area [81]. An alternative that can be applied to technologies with longer range is to simply use lower transmission power to reduce the maximum range: in [82], the authors proposed to send periodically challenges with lower transmissions power to verify the proximity of a device. This approach has been patented in 2009 for 802.11n networks [83].

DB approaches are represented in **Fig. 11** *(right)*. In this example, the legit tag T is within the distance bounds of all three anchors, but the malicious tag M is not in the distance-bound of A_2. Thus, if M is spoofing T, the DB protocol initiated by A_1 will fail, revealing the attack.

All these approaches are binary as they are essentially presence tests. As a consequence, they rely solely on the estimation of their maximum range, which depends on a lot of factors and cannot be estimated nor controlled very accurately. However, several research works have proposed more fine-grained distance-bounding approaches, especially for UWB, where the upper distance boundary is independent from the communication range. These DB approaches are based on challenges that have to be completed by the challenged device as fast as possible [30]. If a malicious device is farther than it pretends to be, its reply will take an abnormally long time as it can not travel faster than the speed of light. The verification of the reply time can be efficient only if the incertitude on the processing time required for the challenge can be kept in the same order of magnitude as the time-of-flight, which is the major challenge in DB protocols conception. The first DB protocol for UWB has been proposed in [84]: in this approach, the challenger sends a sequence of random Binary Pulse Position Modulated (BPPM) bits to the challenged device. Upon reception of each bit, the

recipient must compute an XOR on the received bit and sends the result to the challenger. A non-coherent receiver is proposed to minimize the processing time: compared to a coherent receiver, no digital processing is involved, which allows handling the challenge using only a hardware circuit with low-delay UWB amplifiers, squaring devices, detectors, and XOR gate. The authors estimate the processing delay to 3.72 ns; in the worst case scenario, they evaluate the attacker advantage to 3.6 m. The proposed approach is analyzed from a theoretical prospective and is not implemented and evaluated; also, it is not based on the physical layer of 802.15.4.

Multiple works have later developed further DB protocols on UWB; on one of the most recent research works on the topic [85], the first full implementation of a UWB DB protocol has been proposed, based on very short BPPM symbols for the challenge bits. In their evaluation, the authors estimate the maximum attacker advantage to 15.6 m, based on the processing time involved. These solutions require specific transceiver designs, which are currently not standardized, and are far from costless.

So far, DB protocols appear to be mostly based on the physical layer of the technologies involved. Yet, DB protocols can only be fully exploited if they are monitored by the system layers, which is the reason they are reported here as a system-level countermeasure. Indeed, in a peer-to-peer protocol, DB only provides a partial security asset: it can prevent from distance-reduction attacks[1], but does not bring any guarantee when it comes to distance enlargement. This is not necessarily a problem, as from a system's perspective, we already demonstrated in **Fig. 9** that at least one distance enlargement attack is required to tamper successfully a position when the anchors are placed on the edges on the monitored environment. Hence, if the distance-bounding initiated by the anchors are managed properly from the system layer, DB can prevent from position tampering attacks within the anchor polygon. However, they either suffer from a lack of resolution in the case of transmission power-based solutions or from a consequent additional hardware cost in the case of specialized transceivers.

Cooperative approaches are based on the fact that an attacker needs to know where the locators are to fake a coherent displacement. While most localization systems rely solely on stations or anchors to localize the nodes, cooperative algorithms like Secure Positioning in Sensor Networks (SPINE) [86] involve both the nodes and stations in the ranging processes. When moving nodes are involved into the multilateration process, spoofing successfully an honest node or even lying on its own position is not possible anymore, because the position and identity of the verifiers are not predictable by the attacker. Cooperative approaches require complex verification mechanisms, but compared to the previous methods they do not let any attack opportunity as long as the density and dynamic of the network are high enough. Consequently, they require advanced verification protocols and more computational resources than other approaches, as all the nodes are potential verifiers.

[1] As long as the distance shift is superior to the accuracy of the upper distance bound estimation

A summary of the comparison between these methods is provided in **Table 3**. It should be considered that even if these methods can be applied to any technology compared in **Table 1** (assuming the technical requirements are fulfilled); indeed, the application criteria that are involved in the choice of the technology are obviously not always independent from the ones involved in the security approach.

Table 3. Comparison between the major system-level approaches

Security Approach / **Characteristics**	**Statistical**	**Sector-based**	**Distance-bounding**	**Cooperative**
Global Security Level: integrity and authenticity.	Poor : not deterministic	Average : does not work against an attacker in proximity	Average : focused on long-range attacks	High
Network Specifications : density, coverage, dynamic,	Requires high density, wide coverage and low dynamic [75]	Requires a decent number of sectors to work efficiently. Works great when confronted to a high dynamic. [79]	Requires large areas and low dynamic [31]	Requires a high density, deals better with high-dynamic networks[86]
Resources constraints: computational burden, consumption.	Low: Centralized model. Computational burden for the nodes is at its lowest. [76]	Low: Centralized model. Computational burden for the nodes is at its lowest. [79]	Moderate, as long as the specific hardware required is available.	High computational burden for the nodes since they are also involved as verifiers. [86]
Technical requirements	None	Requires directive antennas	Requires specific hardware [30]	Tags can perform ranging between themselves
Developer challenges	**Building a statistical model of the system**	**Getting the proper anchor configuration for sectors repartition**	**Implementing the specific hardware required for distance bounding challenges**	**Defining the protocol that appoints the verifiers for each localization**

Chapter III SecureLoc Platform

SecureLoc is an open platform for UWB positioning security evaluation that has been developed throughout the PhD. In this section, we present the motivations behind this platform, give an overview of its architecture and main features and discuss the evaluation results of several major ranging and localization protocols conducted on the platform. We define three indicators for malicious behaviors detection and evaluate pratical thresholds for each based on our experiments.

Chapter Contents

III.1 Motivation

SecureLoc's hardware is based on the OpenWino project developed in IRIT Toulouse [87]. OpenWino is an open-source software and hardware framework tailored for fast IoT prototyping, and includes notably the DecaWino project. Decawino [88] nodes are based on a Arduino board embedding a DWM1000 UWB transceiver, which together with the Decaduino library allows fast development of UWB applications. Decaduino is indeed an Application Programming Interface (API) for the DWM1000 operations which provide a higher-level access to the functionalities provided by the DWM1000 physical layer. Several of the ranging protocols introduced in the state-of-the-art (e.g., [69, 88–90]**)** have been developed on DecaWino nodes.

Based on these previous works, SecureLoc platform has been developed on top of the DecaWino framework. There are several motivations behind the development of this testbed:

- **Open source:** As a continuity of the research work on Decaduino held in IRIT Toulouse [58], the openness factor is one of the main motivation. Most off-the-shelf IPS on the market are not fully open, which is not suitable to research activities, especially when it comes to security where having access to all the layers is crucial. The physical layer is based on DWM1000 but all the upper layers are fully open and can be freely modified and adapted to different types of applications. The schematics for Decawino nodes are available at [91]. Regarding the software, both Decaduino and SecureLoc source code are distributed under a GPL v3 license.
- **Inter-operability:** although primarily-based on UWB, SecureLoc is a modular platform and can the filtering and multilateration mechanisms can be interfaced with other ranging technologies.
- **Security focus:** SecureLoc is focused on the security aspects and features implementations of different attacks against UWB and simulation tools that can emulate the effects of state-of-the-art attacks that are currently not mature enough to be implemented in an IoT context.

The first step into developing SecureLoc was to provide a realistic implementation of the localization layer. As far as Decaduino provides all the ranging functionalities needed for positioning, getting a proper IPS required building a centralized architecture on top the Decawino nodes to collect the ranging data and compute multilateration. One of the major goals of this framework, besides providing the basic positioning functionalities, is to collect and log a maximum of physical and MAC layer data. The architecture of the platform is detailed in the next section.

One of the major criteria regarding the features of the platform is to work on a realistic representation of an UWB positioning system. Since our aim is to attack UWB positioning to better secure them, the attack results can only be valuable if the localization layer is representative of the state-of-the-art of indoor positioning. Considering this, a special effort was made to compare some of the major approaches to multilateration on the platform and estimate their performances in a laboratory environment. The motivation was not to provide

novel multilateration approaches or aim for ground-breaking performances, but rather to build solid reference benchmarks in order to later evaluate properly the impact of attacks on performance degradations.

Finally, one of the major contributions of SecureLoc is the implementations of real-world attacks for security evaluation. The literature regarding attacks against 802.15.4 UWB, that we present in depth in section IV.1.5, is majorly focused on high-complexity physical attacks, which feasibility have not be entirely proven in real-world settings. These attacks are often not mature enough yet to be implemented in real-world settings, and involved costs that are way beyond the cost of the potential IoT targets. One of the major aim of the work held on SecureLoc was to demonstrate several low-cost attacks against UWB positioning, which are more realistic in an IoT context, and to provide implementations of these attacks. The attack implementations provided on the platform can be deployed and automated easily and were part of the test vectors for the proposed countermeasures. Regarding the physical layers proposed in the literature, since they are not mature enough yet to be implemented we developed a simulation layer to reproduce the theoretical outcomes of the attacks, which is described in section III.3.6.

III.2 SecureLoc Architecture

III.2.1 Hardware and software overview

The architecture of SecureLoc is displayed below in **Fig. 12**.

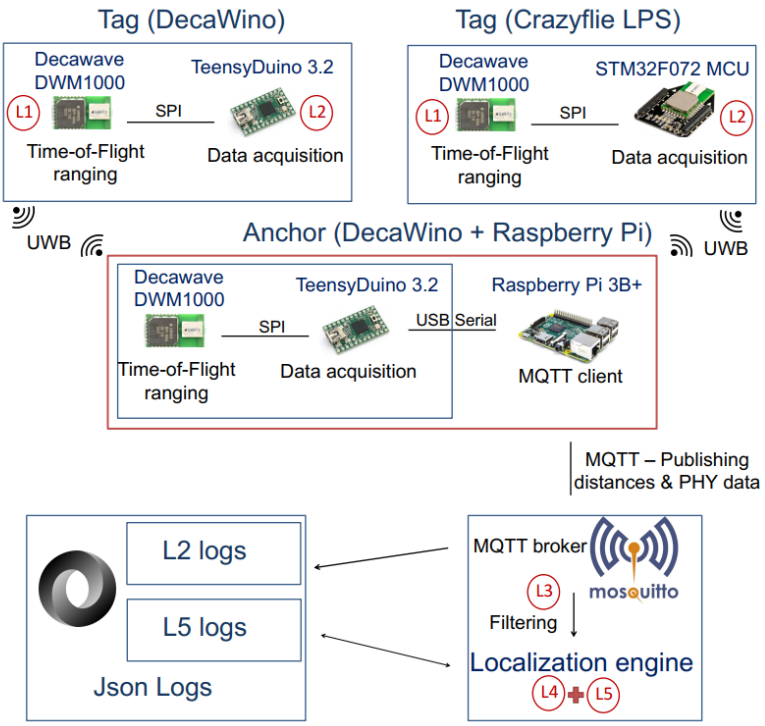


Fig. 12. SecureLoc Architecture

Like most classic indoor IPS, SecureLoc is based on an Anchor/Tag mode. Both anchor and tags are based on Decawino nodes. Indeed, DWM1000 transceivers, which implement the UWB physical layer (L1), need to be driven through the SPI port by a host microcontroller. On Decawino nodes, the host microcontroller, which implements the UWB MAC layer (L2), is a Teensyduino 3.2. We classify the functionalities of SecureLoc in 5 different layers, detailed in **Table 4**. The ranging data of the Decawino on the anchors are sent through USB serial to a Raspberry Pi 3B+. Then, they are dumped to a server through MQTT [92] protocol using an Ethernet link. MQTT gives an easy to use and powerful classification by topics of the various types of physical data (e.g., ToF or RSS) sent by the tags. The tag locations are computed and displayed by a Python 3D engine based on Panda 3D library [11]. The distance filtering can be computed on both the Raspberry Pi and the localization engine; however, this last option is preferable for experiments, as it allow logging both raw and filtered data. The platform is also compatible with the Loco Positioning System (LPS) of Crazyflie [70]; as a consequence, experiments with drone navigation can be done on the platform.

Table 4. SecureLoc layers

L1	Physical layer	Impulse radio transceiver, high-resolution clock
L2	MAC layer	Ranging protocols to estimate the distance between two nodes.
L3	Ranging filtering	Sliding Window filter; dynamic correction
L4	Multilateration	Localization algorithms: Gauss-Newton, Weighted Centroid…
L5	Position filtering	Saturation filter

III.2.2 An efficient prototyping tool: easy and fast firmware deployment

The communications can be ciphered and authenticated with an AES-CCM*-128 block cipher. Hence, each anchor requires a unique ID and a secret AES key. The Arduino framework provides a basic IDE to compile for source code compilation and for flashing Teensyduinos, but this tool is quite limited for IPS experiments, as they require compiling an individual hex file for each anchor and plugging the anchor to the experimenter's laptop. Considering these limitations, we developed a custom compilation tool that allows deploying and evaluating localization and/or security scenarios quickly on the platform. The principle of the deployment tool is shown in **Fig. 13**.

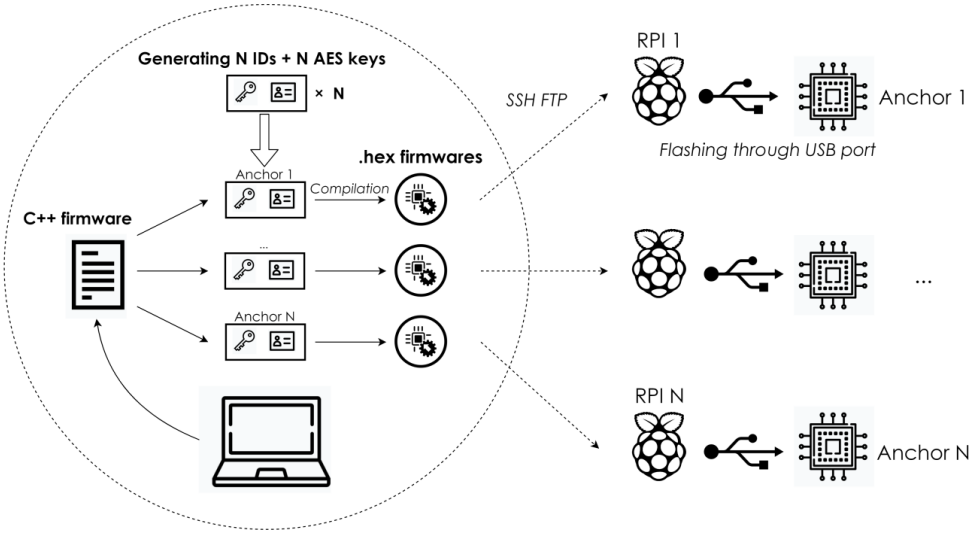


Fig. 13. Deployment tool principle

The first thing that this deployment tool addresses is the generation of multiple anchor binaries from a single C++ source project. The behaviors of the anchors are defined by a Finite State Machine (FSM), for the Time-Division Multiplexing Access (TDMA) management and the ranging protocols. All the anchors are following the same FSM; they all have dedicated time slots based on their IDs a secret AES key for each tag they have to interact with, that we refer in the following as their AES key set. One anchor of the anchor is designed as master Anchor, and has the duty to restart a new ranging cycle when a timeout occurs due to a node missing or due to interferences. Besides that, all the anchors follow the same behavior, and can be programmed with the same source project: the compilation tool automatically compiles one binary for each anchor, each having a unique ID and AES key set. A configuration file describes the platform setup, notably the number of anchors, their coordinates, and the number of tags. These data are automatically parsed and extracted by the deployment tool, and appended to the binaries of each anchor. Then, the binaries are sent by File Transfer Protocol (FTP) to the RPI; finally, the Teensyduino bootloader is called by SSH on every Raspberry and the anchors are flashed. Although not represented in **Fig. 13**, each RPI can handle up to 4 anchors (one on each USB port). Thus, as long as the configuration is up to date the experimenter can deploy any anchor firmware from a single command, without having to handle the platform details configuration.

Regarding tags, since they are not wired to a RPI they have to be flashed directly from the experimenter's workstation. Similarly to anchors, the binaries can be generated from a single source project as their individual IDs and AES key set[1] are also automatically generated.

[1] The pairwise keys generated for the anchors are stored by the deployment tool and associated to the corresponding tag during compilation

III.2.3 Real-time monitoring and data collection

The positions are computed in real-time and displayed in a 3D engine, based on Panda 3D. A sample of the 3D engine rendition is shown below in **Fig. 14**, where 4 anchors and 2 tags are being used. The distances measured by the anchors are displayed above them, as well as the positions of the tags.

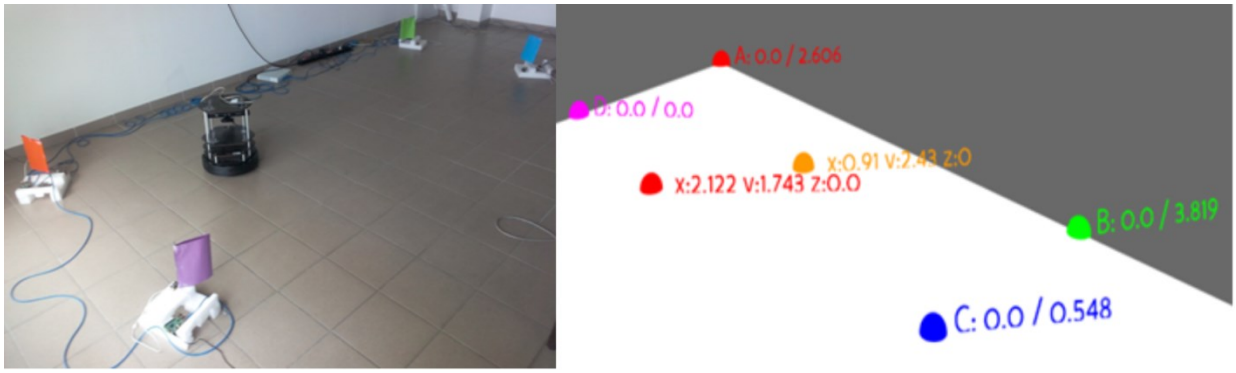


Fig. 14. SecureLoc Platform *(left)*, 3D engine overview (*right)*

During a ranging protocol, an anchor estimates not only the distance but also various physical parameters that are detailed below. Once the protocol is completed, the anchors sends the ranging results to its host Raspberry PI through their USB serial link. Each parameter measured during the ranging protocol is sent on a dedicated MQTT stream. The localization engine subscribes to all the streams and collects the data sent by the anchors. These data are then exploited by the localization algorithm and logged into JSON (*Javascript Object Notation)* files. The content of each JSON sample is detailed in **Table 5**.

The position is appended to each ranging sample by the localization engine right after the anchor data are received. As a consequence, it corresponds to the position measured *before* the distance update. This is necessary as the localization frequency is independent from the ranging frequency, and the position may not be updated immediately upon reception of the MQTT data. For a high-frequency RLTS, this makes virtually no difference as the last position estimate will be typically less than a tenth of a second old.

The distance computed locally by the anchors as it is a very simple and low-cost calculation; considering that the timestamps are transmitted to the upper layers, the localization engine can also compute the distance itself or add further corrections on it.

Table 5. Content of a JSON sample

Parameter	Level	Description
Position	*SYS*	The position that was estimated for the tag when the sample has been received.
Localization method		Localization algorithm that has been used to compute the logged position; filter parameters (Sliding windows/Saturation filters)
Anchor ID	*MAC*	ID of the anchor that performed the ranging.
Tag ID		ID of the tag which was being localized.
Protocol		Ranging protocol that has been used.
Timestamps	*PHY*	Timestamps measured by both parties during the ranging protocol.
Distance		Distance calculated from the timestamp; the calculation depends on the ranging protocol used.
Skew		Clock skew estimated by the anchor during the exchanges.
RSSI		Received signal strength intensity of the acknowledgment frame estimated by the anchor.
First path power (FPP)		Estimation that the DWM1000 provides for the first path power of the acknowledgment frame.
SNR[1]		Signal to noise ratio of the acknowledgment frame estimated by the anchor.
Temperature		Internal temperature of the anchor.

III.3 Benchmarking the performances at each layer

We provide in the following a detailed description of each layer, along with the performances evaluated for each of the protocols, filters and algorithms involved.

III.3.1 L1: Physical layer

The DWM1000 implements the physical layer defined in 802.15.4-2011, which has been described in section II.1.5.1. In a ranging protocol, the most valuable piece of information provided is the distance estimated; yet, various physical data can help to improve the distance estimation or reinforce the security of the protocol. We discuss below the physical parameters logged by the localization engine detailed in section 3.

[1] The absolute SNR estimated by the DWM1000 is not calibrated and cannot be converted into a proper dB estimation; it is rather intended for relative comparison between two different frames, as its absolute value lacks physical meaning [63]

III.3.1.1 **Time-of-flight-related specifications**

Timestamps

The timestamps provided by the DWM1000 are extracted from a 40-bits timer with a resolution of 15.65 ps. This timer is not incremented bit by bit, as a crystal clock would never been able to reach such a frequency (63.87 GHz), and is actually incremented by steps of 512. This step size is equivalent to 8 ns, which is the period of the 124.88 MHz system clock of the DWM1000.

However, the timestamps extracted upon reception of a frame have an accuracy that is much better than the resolution of the system's clock. 802.15.4 defines the minimum UWB bandwidth to 499.2.MHz; the corresponding Nyquist frequency, which determines the minimum sampling rate that the receiver should support, is equal to twice the bandwidth, hence 998.4 MHz [6]. Hence, on the DWM1000, the Phase Locking Loop (PLL) has indeed a frequency of 998.4 MHz, which means that the period between two samples during reception is approximately 1 ns. As a consequence, the PLL frequency is equals to 8 times the system frequency. One sample corresponds to 512 / 8 = 64 ticks on the time-of-flight timer, which is equivalent to a distance of 30 cm.

During reception, the samples are accumulated and a Leading Edge Detection (LDE) algorithm is applied to estimate the index of the leading peak of the PHR on the accumulator [93]. The timestamp is then computed from the system clock's value and tap index, and a correction is applied to compensate for the delay induced by the signal path in the Analog Front-End (AFE). As a consequence, the timestamps estimated during ranging protocols have a resolution of about 1 ns.

Skew

Upon reception of a frame, the clock skew with the expedient can be estimated by two different methods [63]:

1. In the time domain, based on the time length of the preamble symbol received compared to the receiver's reference.
2. In the frequency domain, where the Carrier Recovery Integrator (CRI) provides an estimation of the frequency offset between the two clocks, by comparing the emitter's carrier frequency to the receiver reference.

We implemented both methods and compared their mean and standard deviation. The evaluation has been done on 2000 frames, on channel 2[1] and with a preamble length of 128 symbols. The skew measurements are shown in **Fig. 15**. The average value returned by both methods are very close, the difference being inferior to 0.2 ppm, but the CRI skew estimation is much more stable, with a standard deviation of 0.21 ppm versus 0.8 ppm for the symbol length-based estimation.

[1] The choice of the channel does not affect significantly the skew estimation. Channel 2 has been used for the major part of the experiments in this work.

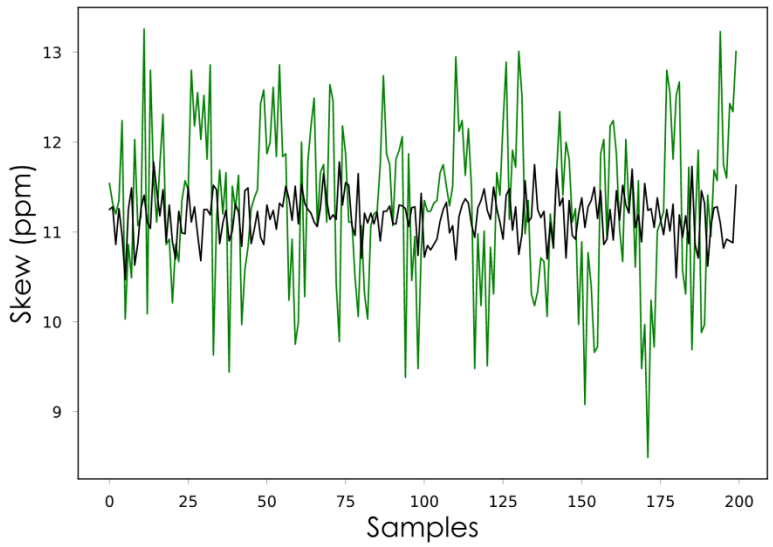


Fig. 15. Comparison between the frequency-based *(black)* and time-based *(green)* skew estimations

When two nodes estimate their respective skew while exchanging frames, they should theoretically obtain opposite values; hence, the sum of their respective skew estimations should be equal to 0. In practice, there are small discrepancies between the skew estimation on both sides, but the absolute value of the sum of their respective skew estimations is typically below 1 ppm.

Link quality-related parameters:
Several parameters can help to evaluate the link quality on the DWM1000.

Received Signal Strength Indicator (RSSI)

Considering that the physical layer is impulse-based, the RSSI estimated on the DWM1000 is linearly related to the Channel Impulse Response (CIR). The RSSI is indeed calculated as follows:

$$RSSI\ (dBm) = 10 * \log_{10}(\frac{CIR * 2^{17}}{N^2}) - A$$

Where N is the number of preamble symbols accumulated and A is a constant equals to 113.77 for a PRF of 16 MHz, and 121.74 for a PRF of 64 MHz.

First Path Power (FPP)

Considering that the short length of the UWB pulses greatly reduces the overlap between the first path and the signal reflections, it is possible to differentiate the different paths to some extent. The role of the Leading-Edge Detection algorithm (LDE) algorithm is to identify the position of the first peak in the accumulator, and the received power for this first path can be estimated from the amplitude reported in the accumulator. Considering that each pulse has a length of 2 ns and that the period between 2 samples accumulated is 1 ns, FPP is

estimated from the amplitude reported in the accumulator at the index returned by the LDE algorithm and the two following samples, which are referred in the following as F_1, F_2 and F_3.On the DWM1000, the following formula is recommended for FPP computation:

$$FPP(dBm) = 10 * \log_{10}\left(\frac{F_1^2 + F_2^2 + F_3^2}{N}\right) - A$$

III.3.2 L2: Ranging protocols

The two standard ranging protocols introduced in section II.1, TWR and SDS-TWR, have been implemented and compared on SecureLoc, as defined in **Fig. 7**. Regarding the distance calculation, we implemented the skew correction proposed in [64], defined in Eq. 5. For TWR, with σ_P the skew evaluated by P, the distance d between P and V is obtained by:

$$d_{TWR} = \frac{(t4 - t1) - (1 + \sigma_V)(t3 - t2)}{2}$$

For SDS-TWR, the distance is computed as:

$$d_{SDS-TWR} = \frac{[(t4 - t1) - (1 + \sigma_V)(t3 - t2)] + [(1 + \sigma_V)(t6 - t3) - (t5 - t4)]}{4}$$

A measurement mode has been implemented on SecureLoc, which can be used to characterize the performances of ranging protocols. A set of reference points is defined. Their coordinates are submitted to the localization engine. For each reference point, the evaluated ranging protocol is repeated multiple times and the experimenter is notified that he can move the tag to the next reference point by an audio signal. The accuracy of the distance estimation for each reference point is automatically evaluated by the localization engine, and the mean, standard deviation, maximum and minimum obtained are returned. The protocol can be repeated for different ranging protocols while keeping the same configuration and reference points so as to get a fair comparison. We used the measurement mode of SecureLoc to characterize the performances of the two standard ranging protocols and compare them.

For the following experiments, we set the number of measurement per reference point to 500. The IPS was spanning over a surface of 8.64 m² (4.8 × 1.8), a total of 32 reference points uniformly distributed on the platform have been tested. The platform configuration is shown in **Fig. 16**.

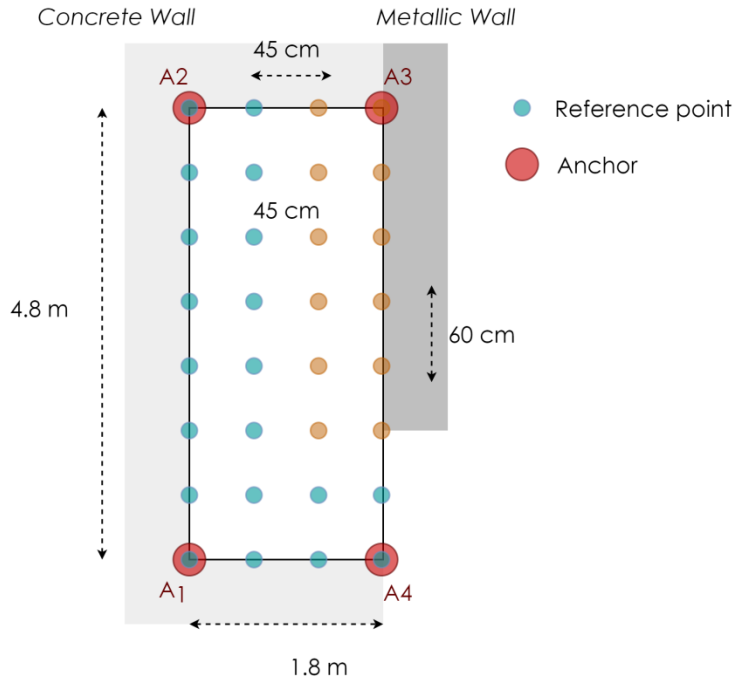


Fig. 16. Platform configuration during ranging characterization

The presence of a metallic wall in the upper right of the platform was noticeably affecting the quality of the distance measurements on the nearby area; the reference points where that effect can be observed are shown in orange (*in the upper right*).

Ranging and localization performances are typically characterized in both optimistic and pessimistic conditions, which are mostly defined from the quality of the tag-anchor links in terms of Line-of-Sight and multipath effects. An indoor environment is too complex to fully model the signal path; hence, it is relatively difficult to compare ranging performances that have been evaluated in two different environments, unless they have been collected in an anechoic chamber. However, it is possible to evaluate qualitatively whether an environment can be considered as clear or complex, based on the two following parameters:

- Line-of-Sight conditions between the tag and the surrounding anchors: if obstacles are on the way, the absorption and reflection effects introduced by these items will degrade the performances.
- Presence of reflective materials around the nodes, which will increase the multipath effects.

Note that the definition of Line-of-Sight, from an RF perspective differs from the definition of an *optical Line-of-Sight*. An optical Line-of-Sight simply implies that a straight line can be drawn between the two nodes without encountering any obstacle. However, an RF Line-of-Sight implies that a defined area around the optical Line-of-Sight, known as Fresnel Zone, should be clear of obstacles. Typically, even if the room is free of obstacles the floor might induce perturbations if the nodes are not placed high enough. The radius R of the Fresnel zone is given by [94]:

$$R = \frac{1}{2}\sqrt{\lambda D}$$

With λ is the wavelength and D the distance between the two nodes.

For a distance of 4.8 m (*highest distance between two anchors in the previously introduced platform configuration)*, that gives a radius of 33 cm for the bottom of channel 1 (3244 Mhz), down to 22 cm for the top band of channel 7 (6998 MHz) [95].

Considering that, we made our measurements in two different settings. In the **clear environment** settings, the platform was empty and the area affected by the metallic wall was excluded from the characterization. Also, the tags were placed 30 cm above the tiled floor, to avoid reflections from the tiles. As a consequence, there was a clear Line-of-Sight between the monitored tags and the anchors, and the multipath effects were limited. We used two other configurations to aim to emulate a **harsh environment.** In the first one, the anchors were placed only 5 cm above the tiled floor, introducing a strong Fresnel reflection that increases multipath effects**.** In the second, anchors were also placed 5 cm above the floor and metallic obstacles were introduced at multiple places on the platform, such as obstructing the Line-of-Sight for each tag-anchor pair.

The performances in the clear environment have been evaluated only on the blue reference points shown in **Fig. 16**, which are the less subject to reflections, while the whole reference points set has been used in the harsh environment settings. The measurements have been repeated with six different mobile tags. The global performances are shown in **Table 6**; the results displayed are the average of the performances obtained for each tag.

Table 6. Accuracy and Standard Deviation (SD) of TWR and SDS-TWR in different types of environment

	TWR			**SDS-TWR**		
Environment	Clear	Harsh (LoS)	Harsh (NLoS)	Clear	Harsh (LoS)	Harsh (NLoS)
Before linear correction						
Accuracy (cm)	17.2	24.5	41.7	16.9	25.4	40.2
SD (cm)	12.3	18.5	23.7	10.7	17.7	20.6
After linear correction						
Accuracy (cm)	12.2	19.4	32.5	12.2	19.6	33.5
SD (cm)	12.1	18.2	23.4	10.5	17.4	20.3

We can observe that there is no significant difference between TWR and SDS-TWR, besides a slightly better standard deviation for SDS-TWR. This corroborates the conclusions of similar experiments in previous works regarding the comparison performances between the two protocols, where little improvements had been noticed when using SDS-TWR over TWR [88]. This is not necessarily surprising considering that the main purpose of SDS-TWR is to limit the effects of the clock skew on the ranging accuracy, which are already

compensated by the skew correction applied on the distance. Considering that the bandwidth cost of SDS-TWR is higher, we use TWR as the default protocol on SecureLoc.

We observed variations across the performances of different chips, even though they had all the same hardware architecture and software, and were evaluated in the same conditions. Indeed, the extremely high-resolution of the ToF measurements makes the process sensitive to very subtle hardware imperfections that occur naturally in the manufacturing process. For that reason, the DWM1000 chips are calibrated individually during production [63], but it does not completely remove the little performance variations across different chips.

As a consequence, we applied a first-order correction on the distances estimated, for which the parameters are calibrated individually for each tag and stored by the localization engine. We used a least-square based linear approximation algorithm on the ranging results obtained in clear environment settings. The R-squared[1] of the linear approximation was superior to 0.9 for all the tags tested, which shows that the first-order model is reasonable. We applied a linear correction based on this first order model and obtained improved performance: for TWR, we have an accuracy of 12.2 cm in the most favorable settings, 19.4 cm in the intermediate one, and 32.4 cm for in the hardest configuration.

The distribution of the slew and offset obtained across the tested tag population is shown in **Table 7**.

Table 7. Slew/Offset distribution across different chips for the linear correction

	Slew	**Offset (m)**
Min	0.94	-0.21
Max	1.07	0.26
Average	0.99	0.07
Standard deviation	**0.04**	**0.12**

We observe clearly here the importance of individual calibration. Some chips tend to over-evaluate the distance where some others are under-evaluating. Considering the standard deviation obtained for the slew and offset, applying the same linear correction to all the chips would lead to an error on a distance of 10m of at least $0.04 * 10 + 0.12 = 0.52$ m. For that reason, applying an individual linear correction at the distance filtering layer (L3) can greatly improve the performances.

Finally, we evaluated the performances of the delayed send (i.e., frame scheduling) function of the DWM1000 which is used in Lightweight TWR (LTWR) protocols [58]. As a reminder, in the LTWR protocol only two frames are used instead of three as the reply time is pre-agreed by the two devices, which allows getting rid of the DATA frame. As far as saving one frame represents a considerable bandwidth gain, one point which has not been discussed in depth in previous works is the accuracy differences between TWR and LTWR. Indeed, in a normal TWR the accuracy of the protocol depends on the accuracy of the

[1] R-squared, or R^2, is the ratio between the expected variation and the total variation in a linear regression. A R^2 close to 1 implies the model tested fits the data, whereas a R^2 close to 0 shows that the model is completely inappropriate.

timestamping process, where for LTWR it depends on the accuracy of the scheduling process, i.e., the capacity of the transceiver to respect the pre-agreed reply time.

Scheduled frames are also timestamped, which means that one can very easily verify if the transmission starting time was correct. We monitored the accuracy of the scheduling process for different reply times. The scheduling process is well-centered, as the mean reply time obtained is less than 300 ps away from the targeted value on the nodes tested. However, there are fluctuations and the standard deviations obtained are not negligible. The results are displayed in **Fig. 17**; for all each reply times tested, the standard deviation has been calculated on a total of 1000 frames. The minimum reply time was 230 µs as it roughly the minimum reply time that a Decawino can achieve with optimized settings; we went up to 10 ms, which is already way above typical reply times[1].

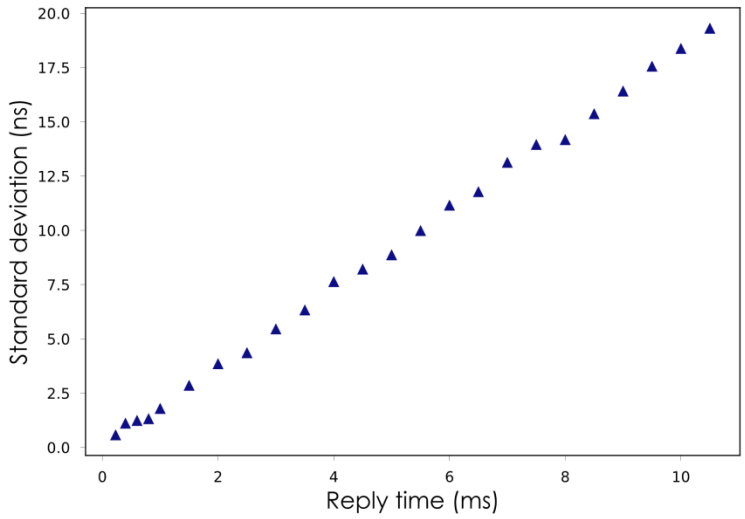


Fig. 17. Relation between the Delayed Send standard deviation and the reply time

The relation between the reply time and the scheduling process standard deviation is clearly linear. After applying a least-squares linear approximation and convert the time error into equivalent TWR distance we obtain a standard deviation of 32.9 cm/ms, which is considerable regarding the typical performances of normal TWR. With the minimum reply time of 230 µs, that is equivalent to a standard deviation of 7.6 cm, which can be a decent trade for a bandwidth cost reduction of about a third. LTWR should be avoided if tight reply times cannot be achieved. Nonetheless, scheduled acknowledgment can also be used within normal TWR as they have the following benefits:

- Tighter TDMA scheduling as the prover's reply time is controlled much more precisely
- If the DATA frame is lost, degraded distance estimation can still be obtained from the known pre-agreed reply time.

[1] Even with the longest preamble, a node can reply within 6 ms. Inactive wait does not bring anything to the process while increasing the clock drift-related error.

- If the localization frequency is high enough, the timestamps can be transmitted in the next START frame to avoid using a dedicated DATA frame. In that case, a first degraded estimation can be exploited temporarily by the localization algorithm in the short time lapse between the two rangings, and the value can be corrected upon reception of the next START.

III.3.3 L3: Distance Filtering

The L3 layer fulfills mostly two tasks: calibrating the distance estimates and reducing the noise measurement.

We already demonstrated that DWM1000 chips require individual calibration. Following the example of the previous section, we calibrated individually each node with a first-order correction based on the ranging measurements. This first-order correction can be stored in the localization engine or directly in the flash memory of each Decawino. However, in a typical industrial application, where the nodes cannot be calibrated in a laboratory environment, calibrating individually the nodes might not be possible. An alternative is to do that calibration dynamically during runtime. A dynamic calibration approach is proposed in [51], which is also based on DWM1000. The authors propose the Best Anchor Trilateration Selection (BAST) localization algorithm, in which part of the proposed algorithm relies on a dynamic correction. After each multilateration returning a position $P(x,y,z)$ for a given tag T, for each anchor A_n, the distance returned by A_n, $R(A_n->T)$, is compared to the actual distance between A_n and P, $d(A_n->P)$, leading to an estimation of the measurement error $\hat{e} = R(A_n->T) - d(A_n->P)$. The idea here is that the final position estimate, which is obtained by the aggregation of the responses of multiple anchors, is less biased than the individual responses of the anchor. Hence, a correction based on the estimation of E can be applied to the next ranging, and the value of $\hat{e}$ can be updated after each multilateration. In terms of architecture, this requires a feedback loop from the top layers (L4 or L5) to L3, as such approaches cannot not be applied without feedback on the position.

We implemented an approach similar to the one used for BAST. For a given anchor performing a ranging R_k with a tag T, with P_k the multilateration solution, we compute the estimation of the error at the iteration k + 1, $\hat{e}_{k+1}$, as:

$$\hat{e}_{k+1}(A_n) = \hat{e}_k(A_n) + a * (R_k(A_n \rightarrow T) - d(A_n \rightarrow P_k))$$

The coefficient a is the increment ratio, comprised between 0 and 1. For a = 1, the approach is equivalent to the one proposed in BAST. However, because of the variability of the ranging outputs, it tends to produce unstable corrections and to decrease the performances. As this approach requires feedback from the positioning layers L4 / L5, its performances depend on the multilateration approach used. Different multilateration approaches are compared in the next section. We did not observe significant differences in the efficiency of the dynamic ranging correction when switching from a multilateration method to another. The choice of a is crucial regarding the efficiency of this method. If a is

too low, the method will be too slow to converge, leading it to be unable to adapt fast enough to the devices' motion. On the other hand, if a is too high, the approach might diverge and lead to chaotic results. Based on the dataset collected during the ranging experiments, we obtained the best accuracy improvement with a = 0.05, with an accuracy gain of 3.1 cm on average.

Regarding the variability of the distance estimation, the results obtained during the ranging characterization show that the distance estimation is not completely stable over time even when the tag is still, considering that the standard deviation is over 10 cm. The usual approach is to use some type of averaging filter; as using an averaging window would considerably decrease the overall refresh frequency of the distance, and considering that the distance variations are limited in time given the physical limits for speed, it is more usually appropriate to use a moving average filter.

Another thing is that it is not uncommon for ranging protocols to fall completely off and give unrealistic values (e.g., negative). These unrealistic distances should be removed and not included in any averaging process. To improve the impact of the moving average, we combined it with extremum elimination. The principle of the sliding window filter implemented on SecureLoc is shown in **Fig. 18**. The last N distance samples are accumulated, and sorted. A certain fraction of the maximum and minimum values are eliminated, and the average of the remaining values is returned. This method demonstrated better results than basic moving average. A comparison of the accuracy of the filtered distance returned by the sliding window filter is given in **Table 8**, based on the data collected during the ranging measurements.

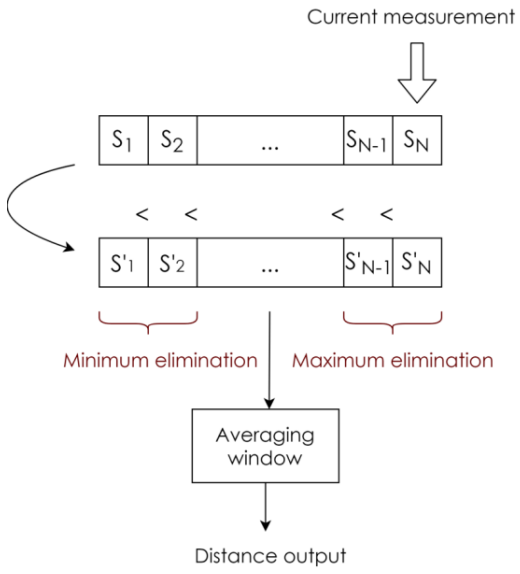


Fig. 18. Sliding window principle

The sliding window has been tested with a ranging refresh frequency of 50 Hz. Increasing the length of the windows tends to improve the accuracy, although a plateau is reached around 20 samples. For a length of 20, an elimination ratio of 60% gives the best results with an average ranging accuracy after filtering of 13.2 cm.

Table 8. Sliding Window parameters comparison accuracy comparison (cm)

Ranging	**TWR**				**SDS-TWR**			
	Window's length							
Elimination ratio	**12**	**16**	**20**	**24**	**12**	**16**	**20**	**24**
20%	29.5	22.5	20.8	22.8	27.5	22.0	20.7	20.5
30%	26.4	22.3	19.0	20.5	25.8	22.7	28.4	18.5
40%	24.2	21.1	18.5	18.6	24.1	20.8	26.0	15.7
50%	30.2	18.2	15.5	15.2	28.3	16.2	**23.7**	13.7
60%	X	16.4	**13.2**	13.8	X	16.9	14.4	24.3
70%	X	20.7	14.3	15.7	X	20.5	28.1	28.2

Clearly, this filtering method induces a delay between a given event (e.g. node starting moving) and the observation of this event in the localization engine. As a consequence, it should be chosen accordingly to the typical velocity of the nodes and the application needs in terms of reactivity. For applications involving mainly still nodes and/or slow movements, the SW filter is defaulted to a length of 20 samples and an elimination ratio of 60%. For a ranging frequency of 50 Hz, that gives a delay of 0.4s between the position measured by the localization engine and the real one.

III.3.4 L4: Localization performances

We implemented four multilateration approaches on SecureLoc, which are representative of the state-of-the-art of localization algorithms introduced previously in II.1:

- Weighted centroid
- Discretized iterative multilateration
- Gauss-Newton algorithm
- Particle filter

In the following, we discuss the principle and limitations of each.

Weighted centroid (WCL): This geometrical approach is based on the centroid method introduced in section II.1, where the solution is defined as the centroid of the trilateration solutions of all possible anchor pairs. However, we implemented an alternative: instead of

distributing the weights equally among all anchors, each trilateration solution is weighted based on its Ranging Deviation (RD, section II.1.4). RD reflects how much a given position fits the distance estimates of the anchors. A 2D trilateration only requires 2 anchors to provide 2 possible solutions; calculating the RD of these two solutions allows not only finding which of the two the correct one is, but also estimating how coherent with the rest of the system the solution is. Hence, we proposed a weighted centroid method where the weight of each trilateration solution is defined as the inverse of its RD [96]. Hence, with $\{S_1, \ldots, S_N\}$ the set of trilateration solutions[1], the final solution $P(x,y)$ is defined as :

Eq. 6

$$P = \sum_{i=1}^{N} \frac{TRD^{-1}(S_i)}{SumTRD^{-1}} * S_i, \qquad with\ SumTRD^{-1} = \sum_{k=1}^{N} TRD^{-1}(S_k)$$

With TRD the Total Ranging Deviation function.

This method is a good compromise between centroid and selection approaches. The anchors that deviate the most will have less impact on the final solution, similarly to selection-based approaches. The redundancy of information is exploited more efficiently as none of the anchors are completely ignored. Weighted centroid has a low computational cost on small numbers of anchors; as the number of trilateration solutions is exponential when the number of anchors increases, proceeding to a selection first may be needed depending on the computational power available. We compare weighted and unweighted centroid in further experiment results.

Gauss-Newton (GN): we implemented the standard Gauss-Newton algorithm, one of the predominant iterative localization approach in the literature and industry, as defined in Definition 2. A decrement ratio of $\mu = 0.8$ has been used for the iterations and the maximum number of steps has been fixed to 10. Gauss-Newton is one of the most efficient heuristics for the TRD minimization problem II.1.4.2, yet is more costly than centroid approaches on small numbers of anchors due to the Laplacian calculations. However, it scales very well as the computational power required is linear to the number of anchors, contrary to the weighted centroid approach.

Particle filters (PF): Particle filter is the predominant stochastic approach in the field of localization. We implemented a particle filter-based multilateration approach as defined in section II.1 (**Fig. 4**), with a fleet of 200 particles. The particle displacements were sampled randomly from a uniform distribution corresponding to speeds from 0 to 10 m/s. The likelihood is estimated with an Extended Kalman Filter (EKF) [97]. Considering that particle filters have a memory effect, as it aims to follow a trajectory rather than identifying isolated positions, particle filters can correct past positions based on the events that occurred after. In

[1] The number of trilateration solutions is equal to the number of possible unordered anchor pairs. For example, for 4 anchors there are 12 solutions.

our implementation, the length of the trajectory tracked was set to 20 samples, which means that for the *n*-th calculated position P_n, the previous positions, up to P_{n-20} can be updated. Hence, the benefits of particle filters can vary significantly according to the delay between the moment the position is measured and the moment this position is actually exploited. For example, in the case of drone navigation, the estimated position will be typically exploited straight away by the drone command. On the other hand, in the case of an IPS aimed toward data science, the position processing is usually not done in real-time, and there is room for correction of the past positions. We refer to the correction of past positions as *backtracking*; in our experiments, the performances have been evaluated with and without backtracking.

We collected a set of ranging data on the same reference points as the ones shown in **Fig. 16**. Four anchors and six tags have been used for the measurements, and the performances have been calculated on over 10 000 distance samples. All the ranging data have been obtained with the TWR protocol. We used the *log and replay* feature of SecureLoc to compare the performances of the different multilateration algorithm: as the localization engine allows replying the ranging events, all the multilateration approaches can be compared on the same dataset. The results are displayed in **Table 9**.

Table 9. Comparison between the average accuracy of various multilateration approaches (in cm); *: without backtracking; **: with backtracking

	CL	**WCL**	**GN**	**PF***	**PF****
Clear environment	26.4	21.2	25.8	25.4	18.9
Harsh environment *(LoS)*	38.9	30.7	32.8	32.2	25.1
Harsh environment *(NLoS)*	47.2	37.4	38.6	36.9	29.7

With 4 anchors, WCL performs slightly better than GN, and features a lower computational cost. The gain of accuracy due to the trilateration solution weighting is consequent as there is a significant accuracy difference between WCL and CL. When backtracking is disabled, WCL is the most performant overall, with 21.2 cm accuracy in a clear environment. However, we can observe a clear accuracy improvement with PF when backtracking is enabled, as it outperforms WCL for all types of environments, with 18.9 cm accuracy in the best case and 29.7 cm in the worst case. However, as discussed earlier, backtracking is not an option for some applications.

III.3.5 L5: Position filtering

The purpose of the positon filtering layer is to check if the position variations over time make sense from a trajectory prospective: typically, the position of a node over a short time lapse are limited considering that there are physical limits to a speed that a node can reach.

PFs are also part of L5 layer as they are actually based on this notion of trajectory. This is not the case for geometrical and iterative layers, which do not have memory effects. As a consequence, further processing can be applied on the positions. We used an acceleration

saturation filter on top of GN and WCL approaches: as the position is monitored in real-time, the speed of each tag can be easily derived from its position and by extension the acceleration can be derived from the speed. Because of the oscillations induced by the measurement noise, some acceleration peaks can be observed due to the displacements observed between two samples, which are introduced by the randomness of the noise. A saturation filter is a simple way to prevent that: a maximum acceleration threshold is defined, based on the application context; if the acceleration exceeds this threshold, the acceleration is saturated to the threshold and the speed and position are recomputed from the corrected acceleration value. We implemented an acceleration filter an experimented with different acceleration thresholds; the accuracy results are displayed in **Table 10**.

Table 10. Acceleration filter accuracy gain (cm) for different thresholds

Step (cm/s²)	**Acceleration threshold**			
	0.5	**1**	**1.5**	**2**
WCL	3.2	2.8	2.5	2.2
GN	2.7	2.4	2.1	1.9

Saturation can improve the accuracy by up to 3.2 cm for WCL, which is the most accurate non-stochastic approach: in that case, WCL reaches an accuracy of 18 cm in clear environments, which is slightly better than PF even with backtracking.

III.3.6 Simulation layer

In order to allow experiments on larger scales and more complex attacks, several simulation features have been implemented on SecureLoc. These simulation features can be used in real-world settings, and allow mounting attack scenarios mixing real and simulated data. They mainly bring three main functionalities:

Distance tampering simulation: Considering that several of the major attacks proposed in the literature have never been demonstrated outside a simulation environment, it is not possible yet to implement them in real conditions due to their current lack of maturity. As a consequence, the effect of distance tampering attacks can be simulated on SecureLoc. Upon reception of a distance sample on MQTT, the distance can be modified by the simulation layer based on three parameters:

Success rate: The probability for the attack to induce any modification of the distance. If the attack fails, the distance remains unchanged.

Denial rate: the probability for the attack to make the ranging protocol fail in any way (i.e., the protocol cannot complete and no distance is obtained).

Distance distribution: the probability density function of the distance shift in case of success. The distance shift induced by a given attack is typically modeled as a Gaussian distribution. The offset targeted by the attacker (positive if the attacker aims a distance enlargement and negative otherwise) is reflected by the distribution mean. The steadiness of the attack depends is reflected by the standard deviation of the distribution.

Although the localization engine will process the positions based on the distances tampered by the simulation layer, the untampered distances are still being stored by the simulation layer. This is indeed required for the two following points, which concern node simulation.

Anchor simulation: Virtual anchors can be created on the platform. They aim to simulate the behavior of real anchors. Based on the position estimated by the localization engine of the tags, the distances between the position provided for the virtual anchor and the tags are computed by the simulation layer. Then, a function emulating the typical measurement noise of anchors (modeled based on the results of section III.3.2) is applied to the distance calculated. Finally, MQTT frames containing the distances computed are generated by the simulation layer and sent on the MQTT bus. From the perspective of the localization engine, virtual and real anchors are basically the same. This feature is typically useful to estimate the impact of a higher number of anchors on the robustness of the IPS against attacks.

Tag simulation: Similarly to virtual anchors, virtual tags can be created on the platform. Virtual tags can be defined either as still nodes or randomly moving nodes (in which case the velocity if fixed). The distance of the virtual tag to each anchor is computed by the localization engine, and processed through the measurement noise emulation function. For each anchor, a MQTT frame containing the distance to the virtual tag is created and sent on the MQTT bus. From the perspective of the localization engine, both virtual and real tags are processed the same way. This feature is typically useful for cooperative protocols, and is exploited in section VI.2.2

At its current stage, the simulation layer does not allow link-quality indicators (i.e., RSSI, FPP, SNR…) simulation and focus on the distance data only. Also, the impact on the virtual nodes on the scheduling scheme is not emulated, although they would need dedicated slots in a real-world configuration. These features may be implemented and used in security experiments in future works.

III.4 Approach and threat model

III.4.1 Detection factors at the system-level

Regarding the three levels of WSN described earlier, it may be intuitive to build security similarly to protocols: from bottom to top. However, because of the peculiarities of IPS compared to other types of WSN, taking the opposite approach can be very effective.

The ranging layer can be assumed to be vulnerable on most radio positioning technologies. There are indeed physical and MAC attacks against ranging protocols, and in particular against UWB ranging protocols as we will present in section IV.1.2Having the countermeasure placed at the same layer as the attack helps to prevent the propagation of the attack to the upper protocols. It also reinforces the independence of the security scheme from the implementation of the higher layers. However, the cost of physical or MAC

countermeasures is often prohibitive, and they are usually specific to one attack and do not necessarily protect from others.

Hence, before implementing complex and costly countermeasures at the physical and link layers, one might want to consider if these attacks can be detected at the system layer with little effort. The nature of the system layer, and by extension the security assets that it can bring, largely depends on the application. A ranging-only application (e.g., wireless car key) will sometimes feature only one anchor, and does not beneficiate from anchor redundancy. A power-constrained IPS which localize a node only upon occasional solicitations (e.g., object tracking in warehouses) has typically multiple anchors but does not provide a continuous track of the nodes. As a consequence, the consistency and plausibility of the track cannot be necessarily verified. However, a typical Real-Time Localization System (RTLS) does provide both anchor redundancy and continuous track. In that configuration, there are three major detection factors that an attacker has to bypass at the system layer:

- **Consistency:** the ranging output of an anchor tracking a tag is quite stable for UWB devices, as we evaluated previously a standard deviation of 18 cm for a static node. It the attacker is not able to repeat the tampering process with a success rate close to 100 %, the output of the ranging process will contain both tampered and untampered values which will lead to a much higher standard deviation than usual. This is equally true if the control of the attacker on the tampered distance is inconsistent, i.e., if the tampered distance is varying randomly in a more inconsistent way than normal rangings.
- **Redundancy:** the attacker should be able to replicate and adapt the attack for each anchor involved in the IPS. As we have seen in **Fig. 3**, in a normal multilateration process, the distances returned by the anchors should give a solution with a low TRD. If the attacker is not able to get all the tampered ranging to match together, the multilateration process obtains an incoherent solution characterized by a high TRD, and the attack can be detected.
- **Plausibility:** localization systems aim to measure a physical data, the position of a device, which obeys several physical constraints:
- Boundaries, such as the walls of the monitored room.
- Bounded n-order derivatives, notably a maximum plausible speed and acceleration, depending on the nature of the objects monitored.
- Continuity, i.e., a given node cannot teleport itself.

If one of these physical assets is broken by the attacker, the position tampering process can be easily spotted. In other words, the attacker should be able to produce realistic measurements and realistic trajectories.

These detection factors rely on simple comparison with thresholds, e.g., the maximum speed of a drone. As a consequence, they have a negligible computational cost and mostly need basic information about the IPS, such as room dimensions or tags maximum speed. This information can be either given by the application designer when possible, or profiled

statistically by the IPS itself, similarly to statistical system countermeasures like DRBTS [78]. If the system layer of a given IPS allows using one or several of these three factors, it can beneficiate from an low-cost detection scheme for attacks that cannot imitate perfectly a normal ranging output. The availability of these different detection factors is summarized in **Table 11** below for different categories of applications, based on the previous discussion.

Table 11. Availability of the detection factors for different positioning application

Application	**Proximity check**	**Intermittent positioning**	**1D-tracking**	**2D/3D RTLS**
	E.g., Car key	Warehouse tracking	Object tracking on a rail	Drone/Robots navigation
Consistency	~ (*)	~ (*)	✓	✓
Redundancy	×	✓	×	✓
Plausibility:	~	~	✓	✓
Boundaries	✓	✓	✓	✓
Bounded derivatives	×	×	✓	✓
Continuity	×	×	✓	✓

✓: Available; ×: Unavailable; ~: Partially available, depends on application details

(*): Even for an application which only localize upon solicitation, series of rangings, rather than single ranging process, might be used to improve accuracy, in which case the consistency of the output can be checked.

At that point, these detection factors cannot be assimilated to the system-level countermeasures introduced in section II.2.2. Indeed, the high-level approaches proposed in the literature, although they are based on the presence of a system layer (i.e., multiple anchors, continuous tracking) to be effective, require often modifications in the physical or MAC layers. For example, sector-based approaches need directional antennas, and cooperative approaches require a specific scheduling scheme for the tag to tag verifications. That constraint apart, they are able to detect attacks than can actually produce realistic distance outputs, regardless of how these attacks are done at the MAC and/or physical level. The three detection factors proposed, on the other hand, do not need any action on the lower layers, but aim only to detect weaker attacks that cannot get close enough to a fully realistic ranging output. As a consequence, they rather represent a costless first barrier against attacks, as they can be implemented by simple thresholds. We use these detection factors as a basis to evaluate the threat represented by a given attack. Indeed, the capability of an attack to bypass these low-cost detection factors is an excellent indicator of how realistic this attack is, i.e., how close from a normal untampered distance it is. In the next section, we propose thresholds for these detection factors based on the performances evaluated on SecureLoc.

III.4.2 Evaluating thresholds for each detection factor on SecureLoc

The evaluation we conducted on ranging and localization protocols in section III.2 provided benchmarks of the IPS performances in both clear and harsh environments in non-

adversarial settings. As the testbed has been designed to give an accurate representation of typical industrial UWB IPS, these benchmarks can give an order of magnitude of the typical performances of these systems without adversaries. If an attack is not able to produce consistent, redundant and plausible tampered distances, then, simple thresholds comparison are enough to detect the attack. We evaluated these thresholds for each of the detection factors proposed in the previous section:

Consistency: regarding the ranging performances evaluated in **Table 6**, the worst standard deviation obtained was 20.2 cm for TWR in a NLoS harsh environment. In real-world settings, a node cannot estimate its own accuracy as the real position is not known; however, it can estimate the standard deviation of the distances obtained. We estimated a boundary for the standard deviation by pushing the NLoS conditions further compared to the previous experiments. For the experiment, the link between two nodes estimating continuously their mutual distance has been gradually degraded up to the complete disruption. The two nodes were placed at 5m from each other and performing TWR continuously while being gradually electromagnetically isolated from each other[1]. The evolution of the quality of their mutual link was monitored through the Line-of-Sight Indicator (LOSI) provided on the DWM1000. According to DWM1000's User Manual [63], a good rule of thumb for the LOSI is that a LOSI above 10 dBm implies a NLOS link and a LOSI below 6 dBm a LoS link. We observed during the experiments that the link disruption occurs typically when for LOSI values between 20 and 24 dB. The standard deviation obtained for different LOSI intervals is shown below in **Table 12**.

Table 12. Ranging Standard deviation in different LOSI intervals

LOSI (dBm)	**0 - 6**	**6 - 10**	**10 - 15**	**15 - 20**	**> 20**
Standard Deviation (cm)	12.3	17.5	20.8	29.5	**39.2**

We can observe that even when reaching the complete link disruption the ranging standard deviation does not exceed 40 cm. Thus, if the standard deviation of the distance output for a given tag is beyond that, it is likely to expect that the node might be under attack.

Redundancy: the lack of redundancy of an attack can be expressed by the incoherencies between the distances returned by different positions, for example if they are not geometrically possible. As explained earlier, because of the measurement noise there is always a slight deviation to a geometrically perfect solution in a multilateration process, but that deviation can be estimated through the Ranging Deviation (RD). As the goal of most localization algorithms is to find the position that minimizes the Total Ranging Deviation (TRD) with a minimum computational effort, the typical RD obtained in different types of

[1] The gradual isolation was obtained by increasing progressively the proximity with a human body, which is one the easiest and most practical way to obtain relatively fine-grain isolation; note that simply closing the hand on a UWB transceiver is enough to completely isolate it from nearby nodes.

environment can be easily evaluated from the different experiments as this parameter is used actively by all multilateration algorithms to compute their solution.

Below are displayed the average RD obtained for each multilateration approach in the three types of environment tested.

Table 13. Ranging Deviation results for each localization algorithm in three types of environment

TRD (cm²)	**CL**	**WCL**	**GN**	**PF***	**PF****
Clear environment	17.2	14.4	13.7	14.5	13.1
Harsh environment *(LoS)*	32.3	29.8	28.7	28.4	26.7
Harsh environment *(NLoS)*	**44.7**	39.6	38.2	37.2	33.5

Even in the harshest conditions, the average RD hardly exceeds 50 cm; if the monitored RD is on average beyond that threshold, it is a sign that either there is an error in the configuration (e.g., the anchor positions used in the localization algorithm are not correct), either one or several anchor/tag links are being attacked.

Plausibility: Plausibility is mostly evaluated in terms of trajectory, i.e., how realistic is the speed of a monitored node. As far as this is a relatively trivial problem, we already observed that the noise measurements induce fast and small amplitude oscillations in the measured position. Hence, the ranging noise introduces a bias in the speed acceleration estimation. From a security perspective though, the concern is not so much to provide an accurate and up-to-date speed estimate but rather to verify if the speed trend falls into reasonable bounds. Hence, instead of defining the speed as simply $S_k = (P_k - P_{k-1}) * T_S$, where P_k is the k-th position measured and T_S is the localization period, the speed can be simply estimated by averaged chunks of N positions as:

$$S_k = \sum_{i=0}^{N-1} P_{k-i} - \sum_{i=N}^{2N-1} P_{k-i}$$

This simple approach induces a delay in the speed estimation and cannot monitor high-frequency speed variations accurately, but is enough from a security perspective to verify that the speed does not exceed realistic limits. With N = 10 samples, we estimated the measurement noise of this simple speed estimation scheme on both still tags and tag embedded on TurtleBots [98] moving at speed of 0.65 m/s. The error induced by the ranging errors on the speed was below 0.2 m/s in both settings, which more than enough to verify the plausibility of the tag trajectory. The value of the threshold needs obviously to be set according to the application as it entirely depends on the typical motion of the monitored assets.

III.5 Results summary on localization performances

SecureLoc integrates all the layers involves in a typical indoor positioning system. A special effort has been made to reflect the typical state of the art approaches used in modern IPS on SecureLoc, and to evaluate and compare these methods on their performances. The accuracy improvements brought by the filters and correction methods discussed earlier, either on the ranging itself or on the multilateration process are summarized below in **Table 14**. The table should be read from top to bottom; a method considered on a given entry is implicitly enabled on all the entries **below** as well. The performances results are separated into two categories, as some of them target the ranging protocol while others target the multilateration protocol.

Table 14. Summary of accuracy improvements

		Method	**Ranging accuracy**	**Localization accuracy**
MAC-level	**L2**	TWR output *(reply time 1 ms)*	20 ~ 150 cm	
		With skew correction on TWR	24.5 cm	
		With static linear correction	18.5 cm	
		With shorter reply times (~230 μs)	17.1 cm	
System-level	**L3**	Sliding window filter	13.2 cm	
	L4	Particle filter (PF) in clear environment		32.2 cm
		PF with backtracking added		25.1 cm
	L5	Saturation filter		22 cm
		Ranging dynamic correction (BAST)	10.1 cm	18.9 cm

The typical localization error in a clear environment can be brought below 20 cm with the particle filters approach when all filters are enabled. These are reasonable performances considering UWB typical accuracy. We used these benchmarks as a reference on the typical performances of an UWB IPS in non-adversarial settings in the next chapters, which focus on the security work of this thesis.

Chapter IV Vulnerability Analysis of 802.15.4 IR-UWB Indoor Positioning Systems

Various works have reported vulnerabilities regarding 802.15.4 IR-UWB positioning. We discussed in the first chapter some general attack schemes against radio positioning technologies, including notably replay and relay attacks. Regarding IR-UWB in particular, specific vulnerabilities in the physical and link layer of IEEE 802.15.4 have been identified in the literature, and attacks have also been proposed for some of them. Yet, the severity of these vulnerabilities and attacks are often studied from the perspective of peer-to-peer ranging protocols, and their impact against a complete Indoor Positioning System is not always discussed. We propose in this chapter a vulnerability analysis of 802.15.4 IR-UWB from the perspective of a typical industrial IPS, based on the methods defined by the *French National Cybersecurity Agency (ANSSI)* in EBIOS [12]. The main attacks and vulnerabilities known on 802.15.4 IR-UWB are reported. A security model is proposed, with the definition of the main threats against industrial IPS. Several attack scenarios are discussed, and evaluated on their likelihood and on the severity of the threats they induce. We finally conclude on the most critical threats identified, which are internal attacks, Early-Detection/ Late-commit (ED-LC), and spoofed acknowledgments.

Chapter contents

IV.1 Overview

IV.1.1 Introduction- EBIOS methodology for vulnerability analysis

In section II.2, we discussed the main aspects of the security of radio IPS, and highlighted that the different technologies that can support positioning features share a lot of common challenges when it comes to security. We underlined that the main security problem faced by ranging protocols is the protection of the physical integrity of the frames exchanged. Classic cryptographic means can protect the privacy and authenticity of the content of these frames, but not their physical properties, which are exploited by ranging protocol. Radio IPS are exposed to any type of attack that can circumvent the physical properties exploited by the ranging protocol (attenuation, phase or time-of-flight), including notably replay and relay attacks.

This consideration led naturally a lot of research works to investigate system-level countermeasures, which are independent of the particular ranging method and radio technology implemented on the IPS. Indeed, the nature of the configuration of a classic IPS brings a strong security asset, which is the capability to easily verify the cohesiveness of the stations involved in the multilateration process. To successfully tamper a position, and remain undetected, an attacker does not simply need to master distance tampering attack; he must be able to produce a set of distances that leads to a coherent geometrical solution. Based on this observation, several system-level approaches have been proposed in the literature, to further enhance the security assets brought by the system layer. We classified these approaches in section II.2.2 in four categories, which are statistical, sector-based, distance-bounding and cooperative approaches. Since each technology has its specific vulnerabilities, these works have been built with the goal to verify the position integrity even when the ranging protocol integrity cannot be guaranteed.

For a typical IR-UWB IPS that does not implement any of these proposed system countermeasures, we formalized the security assets brought by the system-level layers in three main detection factors: consistency, redundancy and plausibility. We consider that an attack should be able to bypass each of these three detection factors to be considered as a critical threat for an IR-UWB IPS.

The testbed developed for the UWB security experiments, SecureLoc, has been designed to reflect the main components of an industrial UWB IPS. We implemented standard and state-of-the art ranging and positioning protocols benchmarked their performances in our experimental environment. From that, we proceeded to quantify the three proposed detection factors by drawing thresholds for each from the worst-case performances. The goal behind these benchmarks is to create a framework for proper UWB vulnerability analysis. So far, we have only described the security challenges that are common to most radio positioning technologies, and have not investigated the vulnerabilities related to the ranging protocol of each. We proceed in this section to discuss the security features of 802.15.4 UWB, the known vulnerabilities and attack vectors in the literature, and the countermeasures proposed for these if any. The research works on this topic have been conducted with

different application intents, and with different security assumptions. Their results, in terms of attack success rate or accuracy, have not necessarily been compared to performances benchmark of real-word IPS to evaluate the detectability of these attacks. Also, their level of maturity is variable, as some of them have been only demonstrated through theoretical analysis, others have been simulated, and very few have been implemented and evaluated. As a consequence, as far as these attacks and vulnerabilities are known, we identified a lack of works in the literature giving a global, objective and exhaustive picture of the state of 802.15.4a/f UWB IPS security state.

The on-going 802.15.4z amendment aims some of the security flaws identified, particularly at the physical level, but as far as we are aware there is not any vulnerability analysis in the literature that gives a global overview of the state of 802.15.4 UWB security.

In the current context, with the release of the new 4z generation, a vulnerability analysis of 802.15.4a/f is a valuable tool for the following reasons:

- Analyzing the exposure of UWB devices of the current generation, that will remain on the market even after the release of 802.15.4z (e.g. DWM1000, which is based on 802.15.4 and used in most UWB industrial IPS today)
- Helping to identify the vulnerabilities that have to be fixed in 802.15.4z, and targeting the most critical ones.
- Designing a proper threat model for the case of 4z devices are interacting with 4a/4f devices: the 4z group task has indeed already announced that 802.15.4z will guarantee interoperability with previous standards. As a consequence, this use case may happen and 4z nodes will have to cooperate with potentially tampered 4a/4f nodes.

The vulnerability analysis method proposed in this section is based on EBIOS (*Expression des Besoins et Identification des Objectifs de Sécurité)*, the methodology defined by the National Cybersecurity Agency of France (ANSSI, *Agence Nationale de la Sécurité des Systèmes d'Information* [12]), with some adaptations to the specific research context discussed here. It should follow the main guidelines of most international vulnerabilities analysis report writing convention. EBIOS is tailored for industrial platforms where the risk can be directly associated to a cost evaluated on the equipment and human resources involved. In an academic context, as the security research work is usually not associated with utterly specific applications, giving an exact estimate of the cost associated with a given risk is more challenging. As a consequence, we define some general metrics based on typical industrial applications with IR-UWB positioning. We give several examples of applications in the following sections and discuss the possible threats for each.

IV.1.2 Context: 802.15.4 security mechanisms

Time-of-Flight, compared to signal attenuation or phase, brings strong security assets when it comes to replay or relay attacks, or any kind of man-in-the-middle (MITM) attack. Indeed, signal attenuation or signal phase do no bring any valuable information on whether the frame received has been physically emitted by the legit expedient or not. This is not the case for time-of-flight ranging; replay and relay attacks induce a delay which is far from negligible. In a replay attack, the attacker must first listen to the frame then replay it, which induces a delay of a least the length of a bit[1]. In a relay attack, the attacker has to demodulate the frame and transmit it at the other end of the link. At the speed of light, a distance of 100 m, which is already an extremely optimistic range for UWB, is traveled in 330 ns. Any delay superior to that would induce distance shifts that exceed the maximum range of UWB; yet, a relay even with loads of optimization would induce a delay of a few microseconds. Hence, relay and replay attacks induce extremely unrealistic distance enlargements, and are quite trivial to detect.

The relay-resilience of UWB, combined to its high accuracy, has contributed making it the leading technology for indoor positioning-related security applications. Along with that, several security mechanisms are defined in 802.15.4 a/f physical and MAC layer specifications.

IV.1.2.1 Physical Security

802.15.4 Physical layer contains one integrated security mechanism, which is the *Dynamic Preamble Selection*. This mechanism primarily intended to prevent a malicious tag from hijacking a ranging protocol by forging one of the frames involved in the protocol, and is also referred to as *private ranging mode*. In normal settings, preamble codes are typically static: the two nodes pick a preamble code according to their channel and PRF and stick to it throughout their communications. Since the preamble occurs before the payload, nothing prevents an attacker to forge a frame that has a legit preamble as he can easily learn the preamble code. If the payload of the legit prover is signed, the attacker will not be able to produce a valid payload; still, the verifier will have to demodulate the payload, which happens after the timestamping event, to detect the fraud. Dynamic Preamble Section (DPS) aims to prevent an attacker from producing a valid preamble by adding unpredictability to the preamble code. The dynamic preamble is simply an extension of the normal preamble code, and can take 8 different values [6]. The choice among these 8 values does not affect the performances regarding the channel or PRF. DPS is meant to be used conjointly to encryption: whenever sending a frame, the chosen preamble extension of the frame is always encrypted and included in the payload. The extension should be picked randomly to limit the prediction capability of the attacker. There not many examples of evaluation of this feature in the literature as it is quite unanimously considered as a weak and rather limited

[1] And requires a specific physical layer for the attack; with a standard physical layer the delay is superior to the length of a frame

security mechanism [11, 99]. A non-exhaustive list of justifications for that statement is the following:

- The size of the extension set, which is only 8, is way too low to be considered secure as the attacker has still one chance out of 8 to guess the right code.
- If one frame is not received by its recipient, the extension for the next one will remain unknown from the recipient and the link will be disrupted. The expedient can use an un-extended preamble to reestablish the link, yet an attacker could deliberately deny frames so as to force the expedient to use a static preamble, then, proceed to forge a frame.
- With numerical analysis, it is actually possible for the attacker to guess the extension symbols before they are completed, which allows him to replay the frame with very little delay [11]
- As the preamble usually takes the most time out of a whole frame, the benefits to detect the attack right after the preamble instead of after the payload are quite limited. Most of the time, DPS do not bring a lot of more than a classic cryptographic signature or a message integrity code.

IV.1.2.2 **MAC security**

There are several points related to security in the 802.15.4 MAC layer, including notably security suites and time slots allocation. We detail each of these points below.

Security Suites

Regarding security, 9 security suites are described in 802.15.4, displayed in **Table 15**.

Table 15. 802.15.4 security suites comparison

Name	Description	Access Control	Confidentiality	Integrity	Replay Protection
Null	No security				
AES-CTR	Encryption only	X	X		X
AES-CBC-MAC-32	Authentication	X		X	
AES-CBC-MAC-64	Authentication	X		X	
AES-CBC-MAC-128	Authentication	X		X	
AES-CCM-32	Encryption Authentication	X	X	X	X
AES-CCM-64	Encryption Authentication	X	X	X	X
AES-CCM-128	Encryption Authentication	X	X	X	X

The most advanced one is AES-CCM, providing both message integrity and authenticity, as well as replay protection (optionally).

Numerous security flaws have been reported on 802.15.4 security specifications in [99]. The most critical points observed are the following:

- No directives in the standard regarding key establishment and key management, which are critical points, especially when using symmetric cryptography.
- Undefined behavior in specific cases (e.g., unexpected device reset, Access Control Lists (ACL[1]) tables full) regarding the nonce states; nonce may be reused in some implementations.
- Weak group key mechanisms, which are unpractical and unsecure
- Unsecured acknowledgments**.** 4 types of frame are available on the standard: Beacon, MAC command, Data, Acknowledgement. If the three first ones can be secured, that is not the case of acknowledgement frames. This implies that anyone can read and forge acknowledgements frames. This a problem given that acknowledgments are used in the UWB ranging protocols suggested in the standard (TWR and SDS-TWR).

Also, ACL tables are limited to 255 entries, which is relatively low considering the predominant use of pairwise keys[2].

Guaranteed Time Slots (GTS) Allocation

Two types of devices are defined in IEEE 802.15.4: reduced-function device (RFD) and full-function device (FFD). FFD devices can be used as Personal Area Network (PAN) coordinators. The standard provides the option of allocating Guaranteed Time Slots (GTS), which in that case will be handled by a PAN coordinator.

Although there are several known GTS slots-related vulnerabilities [100], the allocation scheme define in 802.15.4 is rarely implemented in real-world IPS. Indeed, the time division management in UWB IPS is typically handled locally and piloted in a centralized manner; usually, the network is not meant to allow interaction with any 802.15.4 device passing by. Hence, we do not consider GTS slots vulnerabilities in the following.

Key establishment and key management

Key establishment and key management are two sensitive aspects of security; a strong cryptographic algorithm will not be efficient if the key establishment and key management are poorly handled.

Most embedded systems are secured with symmetric cryptography (when they are), given the computation cost of asymmetric cryptography, which means that a single key is used to encrypt and decipher when a node sends a message to another. The problematic of key establishment is to define a safe way to share the key between two nodes.

[1] Access Control Lists store notably the AES keys for each on-going link

[2] Group keys are intended to reduce the number of keys that a node has to manage; given their inefficiency, 802.15.4 nodes have to rely only on pairwise keys, which requires more ACL entries.

With asymmetric methods, this is less of a problem; the public key can be safely shared and used further to encrypt a private key. For symmetric cryptography, three main approaches for session key sharing are available:

- Using asymmetric cryptography primitives exclusively for the key establishment. Those primitives are not further used once the key is shared.
- This method is flexible, but embedding an asymmetric cryptography has a memory and a development cost that has to be taken into account.
- Hard-writing a long-term key(s) for every node. The long-term key will be used to establish a session key, or can be used directly as a session key which is less secured as if the key is broken the communications will be hijacked for the lifetime of the node. This is the less flexible method and implies that every new node on the network should be flashed specifically for that network. Such an approach cannot be applied in an open network.
- Sending the key in plaintext. This is obviously an unsecured method, which mostly relies on the fact that the window of vulnerability is shorter. This method is used in some ZigBee systems for example [13].

There are different types of symmetric session keys:

- **pairwise keys**, which are shared between only two nodes.
- **group keys,** which are shared by a specific group of nodes of a network, which means that any node can decipher any message of any other node in a given group.
- **network keys** that are shared among the whole network, usually for broadcast purposes. Any node of the network can decipher any message encrypted with the network key.

The choice among those types of keys depends on a balance between security and memory/computation cost.

Pairwise keys are the most secure as they ensure that no one besides the two communicating node can decipher the messages; however, in a network-based exclusively on pairwise keys, the number of keys that a node needs to store becomes huge quickly, and may exceed the size of ACL tables. Also, in the case of multi-hop communications that number becomes quickly exponential considering that the key of every node of the route taken is required.

Group keys help to decrease the memory cost of pairwise keys; they can be combined with pairwise keys for multi-hop communications: handling the intermediate encryptions with group keys drastically reduces the key storage cost, without taking away the privacy if the first encryption of the chain has been done through a pairwise key. However, in the case of single-hop communications the sole use of group keys reduces considerably the security level.

Network keys allow to encrypt a broadcast with a single key, which avoids a tremendous computation cost when sending a ciphered broadcast in a network. Similarly to group keys they can be used for intermediate encryption for multi-hop communications. Besides that, they should not be used for regular single-hop communications given their poor level of security.

Keys are stored in the ACL tables. If these tables ever happen to be full, some of their entries will be flushed and the keys lost in the process will have to be re-established. In the case of 802.15.4, given the maximum size of ACL tables of 255 entries, and the weaknesses of group key mechanism leading to rely mostly on pairwise keys, this could likely happen in a large-scale networks.

IV.1.3 Vulnerability analysis scope

IV.1.3.1 Overview

UWB IPSs are subject to two main categories of vulnerabilities: the ones that are related to radio IPS in general, and the ones that are specific to the IR-UWB specifications in IEEE 802.15.4. We focus on physical (PHY) and Medium-Access Control (MAC) layer vulnerabilities in this section. Indeed, the system layer proposed in III.3.4, which is responsible for the IPS and node positions management, is typically based on third-party software that may be very different from an application to another. As far as this third-party software will induce further vulnerabilities, considering that they are not part of the 802.15.4 in any way and that they are extremely application-dependent we do not consider the higher-level vulnerabilities in this section. Thus, given the elements of context presented in the previous sections, this section examines the vulnerabilities of UWB IPS at four different levels:

- The vulnerabilities related to the PHY layer of localization systems
- The vulnerabilities related to the MAC layer of localization systems
- The vulnerabilities related to the 802.15.4 standard PHY layer
- The vulnerabilities related to the 802.15.4 standard MAC layer

Note that the two first points deal with inner MAC and PHY properties of localization as detailed the sections above, rather than specific implementation or standards properties that may vary from one technology to another. These ones will be studied for IEEE 802.15.4 in the two last points, for both MAC and PHY layers.

IV.1.3.2 Applicative considerations

Estimating the impact of threats and their severity is a huge part of vulnerability analysis, and defining metrics is required to do so. However, providing metrics for the severity or impact of threats has a limited relevance without additional hypothesis regarding the applications studied.

Therefore, this section intends to give a brief background on the most common applications with UWB localization to help understand the metrics introduced in the next section.

From an applicative point of view, the most important characteristics of UWB technology are the benefits of its performances and the inconvenient of its price and transmission range. We already discussed applications based on UWB, some examples are:

- **Personnel & vehicles tracking**: improving the efficiency of the manufacturing by localizing in real-time the actors and anticipating abnormal states or behaviors
- **Robots navigation**: guiding robots in a complex environment, where they might execute tasks together with human worker.
- **Location-based access**: tracking valuable resources that are shared by several workers and raising an alarm if they are taken out the authorized area.

From these examples, several typical characteristics of applications based on UWB positioning are the following:

- High accuracy requirements
- Real-time requirements: the entity carrying the tag is mobile and should be tracked with high accuracy and with minimum delay
- The equipment tracked with UWB is highly valuable

Regarding security, it can be observed that:

- Safety is directly related to security and is highly dependent on integrity and authenticity. In the applications given in example, altering a node's position or spoofing a node's identity can lead to potential damages or robbery.
- The goods concerned are often highly valuable material, and human safety can be in the balance
- Integrity and authenticity are more critical than privacy, although privacy remains a concern. Considering that the applications studied take place typically in industrial environment, privacy concerns are often related to GDPR. Due to the sort-range of the technology, privacy-related vulnerabilities will expose the localization data to local eavesdroppers, who need to be present in the monitored environment in the first place. Fully remote privacy leaks would be induced by flaws in the local infrastructure (e.g. company network), which are not discussed here, as they are not related to UWB. Considering the direct risks induced by integrity and authenticity on safety, we rate the severity of privacy-related vulnerabilities lower than the two others in the following Metrics section.
- Considering the communication capabilities of UWB, UWB can also be used to send commands to or get data from the tracked objects (e.g. robots). As a consequence, there is a difference between circumventing the position of a node and impersonating a node. The latter case is more critical as the attacker will also be able to tamper or forge data that are not-related to localization.

Thus, we made the following hypothesis for the further analysis:

- The system cannot be run safely if one or several tags positions are missing
- In case of damages, the loss caused by the damages is most likely higher than the price of the whole localization system
- One or several modifications of tag's positions can lead to damages and/or human harm.
- Rogue command execution through UWB can lead to damages and/or human harm.

IV.1.4 Metrics

Given the hypothesis stated in the previous section, the metrics that are used in this analysis are defined as following:

Severity: Severity metrics are defined in **Table 16**. As a general rule, keeping secret the position estimated by the localization system does not ensure that this position cannot be retrieved by alternative localization means (as a reminder, any radio technology provides indirectly information on the position through reception power as explained earlier). Also, positions are not meant to be kept secret in the majority of use cases (e.g., indoor navigation or manufacturing chain supervision). Thus, confidentiality-related threats are ranked as relatively minor here. This aspect might change for the future 4z applications that are more user-oriented. A partial position unveiling refers to any leak that gives information on the prover's position with a degraded accuracy; total unveiling refers to the case where the attacker obtains the prover's position with a similar accuracy to verifiers(s). The loss of the tag's position is ranked as a higher threat as it can potentially be a danger for the infrastructure itself. Position tampering is the highest threat; the attacker can either have an erratic control over the tampered position (high severity) or a fine-grain control (very high severity), which is the worst case.

Table 16. Severity Metrics

Level	Qualification	Privacy issues	Authenticity/Integrity Issues
1	Low	Partial information on the position of the device has leaked	
2	Limited	Position related information of the device has leaked	Human intervention is required
3	Medium	Non-position related information of the device has leaked	System is paralyzed
4	High		Potential untargeted damages or human harm
5	Very High		Potential targeted damages or human harm

Likelihood: Likelihood metrics are defined in Table 17. Attacks with the highest likelihood are the one that can be mounted by attackers with very basic technical skills and with basic hardware (e.g., using the same hardware as the victim). If the attack can be set up quickly "on the field", i.e. the attacker can simply walk in and trigger the attack within a few minutes, its likelihood is considered as quite high. On the other hand, attacks, requiring high expertise, expensive equipment (e.g. oscilloscopes/spectrum analyzers) and that are difficult to mount outside a laboratory environment, have typically a much lower likelihood. Also, attacking the position requires necessarily to be able to increase and decrease distances to obtain a coherent set of rangings. Hence, an attack that cannot do both is weak against multilateration. Based on these metrics and the list of attacks reported in the previous section, we evaluate the attack that can occur at each security level in Table 5. Note that an attack that works at a given security level also works for the security levels below.

Table 17. Likelihood metrics

Level	Qualification	Description
1	Very Low	Cost of the attack largely superior to the attacker's potential gain Has never been achieved in real-world conditions
2	Low	Cost of the attack superior to the attacker's potential gain Requires very specific conditions for success (i.e. low success rate) and expert skills
3	Moderate	Cost of the attack inferior to the attacker's potential gain Requires expert skills
4	High	Cost of the attack largely inferior to the attacker's potential gain Requires technical proficiency
5	Very High	Negligible cost; requires only basic technical skills

Security Level: Security levels are defined according to the security specifications of the standard and are defined in **Table 18**. At the highest security level, all the standard security mechanisms are enabled.

Table 18. Security levels metrics

Level	Description
0	None of the security suites of 802.15.4 are implemented
1	AES-CTR
2	AES-CBC
3	AES-CCM
4	AES-CCM + replay protection enabled Key establishment: key sent in plaintext
5	AES-CCM + replay protection enabled Key establishment: key hard-written or established through symmetric cryptography mechanisms

These metrics are relevant as long as the hypotheses are verified; the severity scale should be adapted for a system following a very different applicative scheme. Nevertheless, the analysis regardless of the metrics used will remain relevant.

IV.1.5 Threats

We present in **Table 19** the major threats identified for industrial UWB IPS. The list of impacts is based on the typical applications discussed earlier and is not exhaustive. The table structure is based on EBIOS recommendations.

Table 19. Threats against UWB IPS

Threat	Description	Impacts	Severity
Tag spoofing	One or several node identities have been spoofed	- Position of the spoofed tag is wrong - Spoofing device can access unauthorized area - Loss of control of the device tracked by the spoofed node	High
Anchor spoofing	One or several anchor identities have been spoofed	- Anchor can get partial or total information on tags position - Tags position within the spoofed anchor area can be altered - Chaotic control of the devices tracked in the anchor area	Very High
Position alteration	The position of the node seen by the localization system has been altered externally	- Loss of the device tracked - Control of the tracked device becomes chaotic	High
Position cheating	The node is cheating on this own position	- Access to unauthorized area - Loss of the device tracked - Control of the tracked device becomes chaotic	High
Rogue commands execution	Rogue command execution on a node by an attacker	- Device hijacking - Tampered reports	Very High
Anchor Denial	Anchor is unable to work properly	- Accuracy/Reliability of the system is lowered - At some point the system may not be accurate enough anymore to be run safely	Medium
Node hiding	Node's position is lost	- Device's track is lost - Exposure to theft -Potential damages due to loss of control (e.g. vehicle)	Limited
Node hijacking	Node's position remain unchanged or within a certain area while the node has been physically removed from that area	- Exposure to theft - Potential access to unauthorized area	High
Position partial spoiling	The node's position can be seen approximatively by an attacker	- Privacy issues (e.g. industrial secret) - GDPR	Low
Position total spoiling	The node's position can be seen by an attacker with the same level of accuracy that the system provides	- Privacy issues - GDPR	Limited
Privacy broken	Non-localization related private content can be read by unauthorized parties	- Privacy issues - GDPR	Medium

The focus should be given on the most critical threats identified, here the ones with high (level 4) and very high (level 5) severity levels, which should be considered in priority in the security conception.

In **Table 19**, anchor spoofing and rogue command executions are considered as the highest threats. Anchors are involved in the verification of all the mobile tags; a compromised anchor can potentially compromise the whole IPS. Rogue command execution implies that an unauthorized actor could potentially take full control over a device, e.g., a drone or a robot; such as scenario could lead to potential damages and induce a risk for surrounding human workers.

Tag spoofing, position alteration, position cheating and node hijacking are classified as threats with high severity. Tag spoofing or position alteration imply that an unauthorized actor can take a partial control over a monitored device: all the actions that are driven by the localization system (e.g., vehicle following a defined path) will be indirectly under the attacker's control.

Position cheating and node hijacking can lead the system to miss an event that should have triggered an alarm. Indeed, if a node is lying on its position, it may access unauthorized areas; if a node is hijacked, the attacker might be able to withdraw the node from its normal environment without being detected. Based on this threat analysis, we analyze the attack vectors in the following.

IV.2 Attack Vectors

This section introduces the state-of-the art of attacks against UWB. Based on the security works in the literature regarding 802.15.4 UWB positioning, we identify and list the known vulnerabilities of this technology. Then, we proceed to give the details of the attacks vectors that can exploit these vulnerabilities. Finally, we discuss scenarios based on these attack vectors.

IV.2.1 Known Vulnerabilities of UWB positioning

This section reports the vulnerabilities of 802.15.4 UWB positioning identified in the literature in two categories, PHY-related and MAC-related.

IV.2.1.1 Physical-layer vulnerabilities:

Binary Pulse Position Modulation: 802.15.4 UWB physical layer is based on On-Off Keying (OOK) modulation and Binary Pulse Position Modulation (BPPM). In a BPPM scheme, the value of the bit depends on the position of the pulse in the timeslot: the first half encodes a '0' while the second half encodes a '1'. In practice, the energy received for each half of the timeslot is compared to determine the bit value. This modulation scheme is not resilient to malicious energy detection: if for example a bit with a value of '1' is encode with an energy E1 in the second half of the timeslot an energy E0 > E1 is injected in the first half then the bit value will be switched to 0 [11].

Leading peak detection: when detecting the SFD, a backward search of the leading peak is executed [101]. The output of this backward search is vulnerable to energy injection as it can modify the leading peak. The SFD detection being the event timestamped this will affect the distance measured [101].

Private preamble codes predictability: The private ranging mode let the two nodes chose one code among only 8 possibilities. That let an external device a fairly decent chance to guess the right code. Also, using multiple transmitters or using some digital processing tool can overcome easily this mode [11].

IV.2.1.2 **MAC-layer vulnerabilities:**

Trust in the prover: 802.15.4 ranging protocols require honest participation from the prover. The protocol is not tamper-proof to dishonest participation, such as fake timestamps. As a consequence, ranging protocols do not give any asset to the verifier regarding the honesty of the prover's replies.

Unsecured acknowledgment: this is one of the major MAC vulnerabilities of 802.15.4 UWB [102]. The standard proposed four types of frame: beacon, control, data, and acknowledgments. Acknowledgment frames do not support any security suite in the standard, while they are being used in TWR protocols. The 2015-802.15.4 has proposed slight improvements by dividing acknowledgment in two sub-categories, immediate and enhanced acknowledgment. The first one is similar to the acknowledgment format defined in the previous standard (i.e., supporting no security), while the latter one does support cryptography mechanisms, although there are no prescription to use them in ranging protocols. The private ranging mode is claimed in the standard to be sufficient to guarantee the integrity of ranging protocols.

ACL tables Overflow: There are at most 255 entries in the ACL tables in the standard [103]. If, as mentioned earlier, this is enough for pairwise keys in classic applicative contexts, these tables can overflow quickly if new keys (that could be gibberish) are deliberately added. The policy to follow when ACL tables are full is not defined in the standard, but one would expect the most ancient entries to be flushed.

Possible nonce reuse/ nonces reset to known values: this point is more obscure; there is lack of specifications in the standard regarding nonces. As pointed in [103], this grey area around nonces could lead to resetting nonce to known values (e.g. 0) when the device has been abruptly powered off; also, some flaws in ACL tables management could lead to nonce reuse. This point is more speculative and we are not aware of any implementation that may be concerned. Thus, we do not discuss this particular potential vulnerability in this paper.

IV.2.2 State-of-the-art of attacks against UWB

Several attack vectors that can exploit the vulnerabilities presented in the previous section have been proposed in the literature, especially at the physical layer. We report in this section the major attacks discussed in the literature; some of them have been only theoretically conceptualized, others have been simulated, some have been implemented. In the different attacks description we refer as P and V the prover and verifier of a ranging protocol.

Physical-layer attacks:

Jamming attacks: jamming attacks are a threat for any radio system. The attacker deliberately generates collisions with the transmissions of the victim by emitting loudly continuously.

Replay attacks [104]*:* the attacker replays a previous frame of its victim. If replay protection is disabled, a replayed DATA frame would for example seem legit; if the attacker has recorded some DATA frames when the tag was at another position, the timestamps in the replayed frame will correspond to this previous position and allow the attacker to modify the distance measured. If replay protection is enabled, the attacker can only replay the last frame of the victim, and has to annihilate these frame (e.g., with jamming), otherwise the replay counter will be incremented.

Relay attacks: the attacker establishes an illicit link between P and V, while they are out of range of each other. The attacker only has to relay the frame between the two nodes through a long range link; he does not need to modify or even understand the frames relayed. The relay is intrinsically a distance-increasing attack, due to the extra delay induced. Actually, given the short times involved in ToF ranging (signals travel at the speed of light) the delay induced by a relay is usually equivalent hundreds of kilometers, even when optimized for speed. We are not aware of experiment reports regarding relay attacks on UWB, yet, realistic relay attacks would require the relay delay to be kept at least below 330 ns (~100 m at the speed of light). A single preamble symbol spans over 1 μs, which means that even if the attacker would be able to repeat the victim's symbols straight after its emission, he or she would produce unrealistically large distances. The only alternative is to have a way to anticipate the symbols and bits sent by the victim, which is actually possible with the Early Detection/Late-commit (ED-LC) [11] attacks detailed below. Nevertheless, classic relay attacks produce very erratic and unrealistic large distances on IR-UWB.

Overshadowing attacks [101]: overshadowing attacks are distance-increasing attacks that target the leading peak detection vulnerability. The attacker uses a replay attack, where the replayed signal is slightly behind and has a higher transmission power than the legit one. Doing that, the attacker shifts the leading peak of the first path forward in time, increasing the time-of-flight measured. Research works on this have shown that this attack does not work in every case and that the probability to get a consistent distance output is low. This concept has been validated through simulation but never implemented. It has been shown in [101] that overshadowing attacks are very dependent on the delay value targeted by the attacker. The attacker will have to target specific delay values to add on the time-of-flight;

otherwise, his chances of succeeding are close to 0. Even so, these attacks are very unstable and probabilistic by nature. As a consequence, the control of the attacker on the distance produced is very limited: it is not possible to target specific distance shift values and the attack might not produce the same distance shift every time. These attacks also require a Gaussian channel between the attacker and the victim, which is unlikely in real conditions. As a consequence, they are typically not considered as a major threat.

ED-LC Attacks [11]: these attacks aim primarily to decrease the distance measured. The UWB physical layer is based on BPPM for the payload: each bit is coded on two-times slots, where a pulse in the first slot encodes a '0' and a pulse in the second slot encodes a '1'. In practice, the energy E_0 of the first slot and E_1 of the second slot are compared to estimate the position of the pulse.

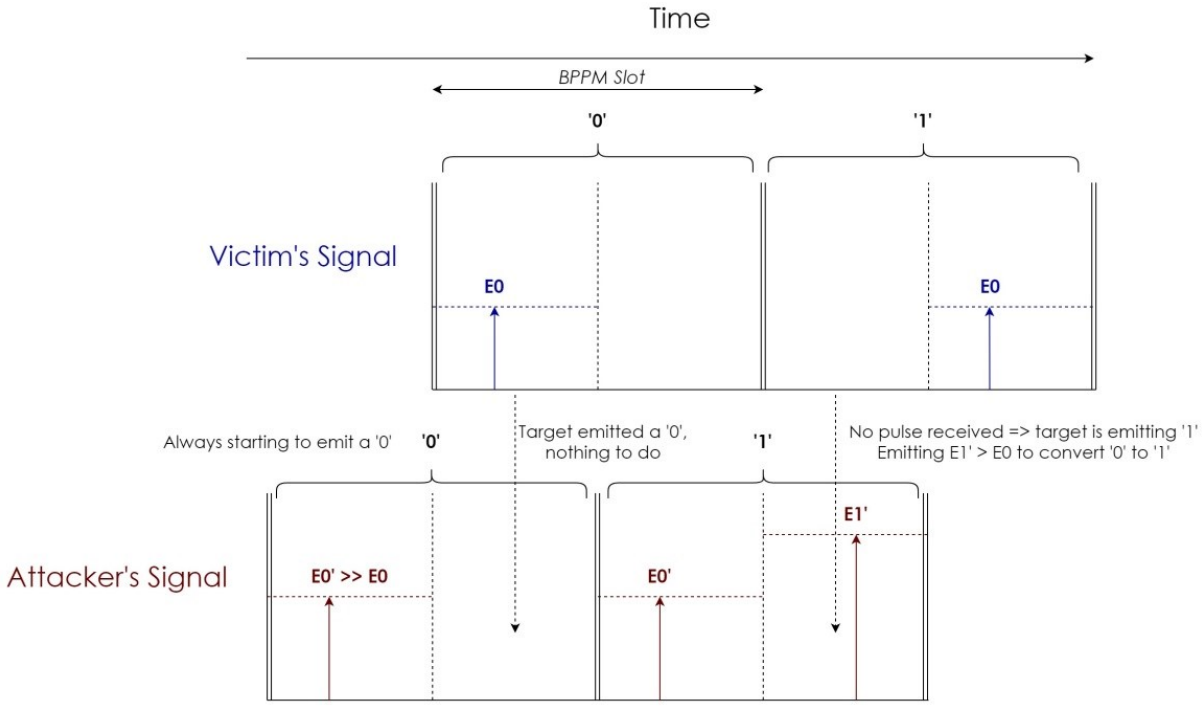


Fig. 19. Late-Commit principle illustrated on the bit sequence '01'

BPPM is vulnerable to the so-called Late Commit (LC) attacks, illustrated in **Fig. 19**: a '0' can be modified to a '1' if an energy $E_1' > E_0'$ is emitted in the second time slot. In other terms, a rogue device can start emitting about half a slot period before knowing the payload, by sending always '0' first and switching to '1' in the second slot when needed. LC allows an attacker completing a replayed frame before the actual expedient, leading to a replay 'backward in time' attack, as long as the attacker transmits with an energy $E_0' >> E_0$, as shown in **Fig. 19**. One of the key elements to achieve such attacks is to anticipate the SFD such as starting the PHR before the victim, which will lead to pull the timestamp backward in time. An Early-Detection (ED) mechanism for this purpose is proposed in [11]. Combined

together, ED and LC could potentially allow distance reduction attacks; in [11], the authors show that an attacker can obtain a distance decrease up to 140 m with a 99% probability through simulation. As far as we are aware, no implementation has been proposed.

MAC-level attacks:

Spoofed acknowledgment [57]*:* acknowledgment frames are not secured and can be forged by an attacker; if the attacker sends a spoofed acknowledgment during the ranging protocol, the acknowledgment reception time will be tampered, leading to a fraudulent time-of-flight. It is suggested in [57] can easily mount a distance decreasing attack by acknowledging before the victim. In that case, the attacker can control the reduced distance through his/her reply time.

ACL overflow with Sybil attacks [103]*:* A Sybil attacks consist of a single node claiming multiple identities [105]; one of the consequence is that the malicious node can establish a key for each of its fake identity with its victim such as filling the target's ACL tables. This will likely lead to flushing the first entries of the ACL tables and force the victim to reestablish a session key with the anchors. If, as discussed in section IV.1.2, these keys are sent in plaintext, the attacker will be able to tamper the ranging protocols (e.g., modifying the timestamps reported) and hijack the victim's position.

Internal attacks [103]*:* refers simply to a node lying in the ranging protocol, i.e. reporting fake timestamps. By extension all the previously introduced attacks are *external* attacks as they involve an external party.

IV.2.3 Attack vectors evaluation for each security level

Attacks with the highest likelihood are the one that can be mounted by attackers with very basic technical skills and with basic hardware (e.g., using the same hardware as the victim). If the attack can be set up quickly "on the field", i.e. the attacker can simply walk in and triggers the attack in a few minutes, its likelihood is quite high. On the other hand, attacks, requiring high expertise in the field, expensive equipment (e.g. spectrum analyzers) and are difficult to mount outside a laboratory environment, have typically a low likelihood. Also, in a positioning system attacking the position requires necessarily to be able to increase and decrease distances to obtain a coherent set of rangings; hence, an attack that cannot do both is weak against multilateration.

Based on these metrics and the list of attacks reported in the previous section, we evaluate the attack that can occur at each security level in **Table 20**. For each attack vector, the related threats are listed in the right column along with their severity. Note that an attack that works at a given security level also works for the security levels below.

Table 20. Attack vectors evaluation

SL	Attack Vector	Likelihood	Related Threats	Severity
1	Timestamps tampering in DATA frame	Very High	Biased reports	Very High
2	Eavesdropping	Very High	Complete position leakage	Low
3	Replay attacks	High	- Biased ToF measurements - Biased reports	Very High
4	Sybil + eavesdropping	Moderate	- Complete position leakage - Verifier spoofing - Prover spoofing - Biased reports	High
	MITM on key establishment	Moderate	- Complete position leakage - Verifier spoofing - Prover spoofing - Biased reports	High
5	Spoofed acknowledgments	High	- Prover spoofing - Biased ToF measurements	Very High
	Overshadowing attacks	Very Low	Biased ToF measurements	Very High
	Isolation + replay attacks	Moderate	Unrealistic ToF measurements	Low
	RSSI localization	Moderate	Partial position leakage	Medium
	Spoofed acknowledgments	Very High	- Prover spoofing - Biased ToF measurements	High
	Jamming	Moderate	- Tag enclaving - Anchor denial	Medium
	Relay	Moderate	Unrealistic ToF measurements	Limited
	Internal attacks	High	Biased reports	Very High
	ED/LC	Low	Biased ToF measurements	High

Replay and spoofed acknowledgment attacks require only small software modifications on a regular node and can be set up quickly, hence their high likelihood. On the other hand on the spectrum, physical attacks (ED-LC and overshadowing) require high expertise, complex equipment (a modified physical layer is required on the malicious nodes, and these attack would likely be calibrated with an oscilloscope) and are very probabilistic in terms of success (especially overshadowing). Replay protection or private ranging mode only work if the original message has been received, and are efficient if the attacker annihilates the original frame before replaying. The delay induced in the process leads to unrealistic time-of-flight measurements.

At level 4, it is assumed that the key establishment scheme is not secure. This is often the case for cheap IoT devices that do not support asymmetric cryptography, for which it is not uncommon to send the key in plaintext. In that case, the key is obviously exposed to eavesdroppers when the session starts. A simple MITM attack lets an attacker learning the

key, and forging fake timestamps, e.g., in a DATA frame. Often, such unsecure key establishment schemes rely on the fact that the window of vulnerability is short: if the key is intended to be shared in an environment that is considered secure (i.e., eavesdropper-free). However, even if the attacker arrives after the key establishment, it is possible for him to compromise a node. Indeed, the attacker can overflow the ACL tables of a given verifier with a Sybil attack. In that case, the key of the session with the target prover will be erased, and the verifier and prover will have to reestablish a key in plaintext, which this time, will be picked up by the attacker.

Finally, even with the higher level of security an attacker can always retrieve information on a node position based on attenuation estimation through RSSI as discussed in the previous sections, which implies that position data can leak even at the highest security level.

IV.2.4 A simplified attack taxonomy for RTLS

Regarding the previous discussion, we can simplify down the attack taxonomy introduced in the previous section to a much simpler model specific to RTLS. We will distinguish only two main types of attacks: **DoS attacks** and **tampering attacks**.

Concerning DoS, we propose two categories:

- **Fine DoS**, which basically refer to tampering attacks that produce unrealistic distance and/or position outputs. Given the three detection factors aforementioned, they cannot be possibly confused with actual legit positioning data, but they still prevent the IPS to localize one or several nodes.
- **Full DoS,** which refers to the conventional DoS attacks in the WSN literatures. This category contains the classic wide-band and pulse-band jamming attacks [106] and link layer disruption schemes [103] (e.g., attack on the replay counter).

Concerning tampering attacks, we consider in this category any attack that can lead the IPS to record a realistic-looking fake position for a given node, regardless of how the attacker generates that behavior. As a consequence, a spoofing attack falls also into this category: if an attacker is able to successfully impersonate a victim node, then the position evaluated by the IPS will correspond to the position claimed (real, or not) by the malicious node and not the honest one. Based on the previous considerations regarding the realism of an attack, this can be determined by the capability of the attack to bypass the system's detection factors.

In the context of RTLS, any distance tampering attack that cannot bypass these detection factors is reclassified as in the fine DoS category. Based on the results presented in the concerned research paper and the benchmarks evaluated on SecureLoc, we evaluated the capability of each of the aforementioned attack vectors to bypass these detection factors.

Table 21. Capability of each attack vector to bypass system-level detection factors; ✓: undetected; ×: detected; ~: depends on other factors; ?: lack of technical evidence in the literature

	Consistency	**Redundancy**	**Plausibility**
Overshadowing [101]	~	×	×
ED-LC [11]	✓	?	?
Replay [72]	~	×	×
Relay [73]	~	×	×
Internal attack [57]	✓	✓	✓
Timestamps tampering [57]	✓	✓	✓
Spoofed acknowledgment [57]	✓	?	✓

Overshadowing attacks can only increase the distance, hence, will not be able to bypass the redundancy factor as at least one distance reduction will be needed. Also, based on the success rate presented in [101], the attack can be considered consistent only in very favorable settings for the attacker. ED-LC attacks are primarily intended for distance reduction as this is considered by the authors as way more challenging than distance enlargement; nothing in their realization prevents them from generating distance enlargements as well. Yet, the simulation results obtained show that the attacker should target specific distance values to maximize its chances. However, bypassing the redundancy factor needs from the attacker to produce precise values on each of the anchor involved; as a consequence, it is unlikely that the attack could provide both consistency and redundancy. Replay and relay attacks, as already discussed, produce unrealistic distance enlargements; they might be able to produce a relatively consistent output under favorable assumptions, but this output will be far beyond realistic boundaries. Forged acknowledgments are considered to be a trivial way to produce an accurate distance enlargement; the opportunity to use them as a distance reduction attack has not been discussed in the literature. Finally, internal attacks bypass all three detection factors as the lying node has simply to modify its timestamps and can control accurately the distance obtained by each of the surrounding anchor. Based on these considerations, we classified the attacks previously listed in the three proposed categories:

Table 22. Simplified attack taxonomy

Distance tampering	**Fine DoS**	**Full DoS**
Internal attack; Spoofed acknowledgments; ED-LC (*assuming their practical feasibility*)	Relay; replay; Overshadowing;	Jamming attacks

Obviously, the capability of forged acknowledgments and ED-LC to produce realistic tampered positions remains to be proven. We will investigate in depth in this work the capabilities of acknowledgment attacks and discuss the hypothesis under which they can achieve realistic outputs. Regarding, it is difficult to provide more accurate benchmarks as they have never been implemented to our knowledge. Nevertheless, we will bring several elements observed throughout our research regarding their feasibility.

IV.3 Conclusion: most critical attacks

In the previous sections, we have reported the known vulnerabilities of 802.15.4 IR-UWB positioning along with the attack vectors exploiting them that have been proposed in previous works. We have discussed the threats related to these vulnerabilities and evaluated their severity based on the metrics proposed in section IV.1.4. The costs, complexity and maturity of each attack discussed have been compiled into a likelihood estimate. The security levels at which these attacks can occur are discussed in section IV.2.3 and shown in **Table 20**. A simplified attack taxonomy for RTLS has been proposed, based on only three categories, which are, fine DoS, full DoS and distance tampering.

The most critical attacks are the one that are functional still at the highest level of security defined in the standard. They are shown in **Table 23** with their severity and likelihood ratings.

Table 23. Most critical risks at the highest security level (level 5)

Vulnerability	Attack Vector	Severity	Likelihood
Prover trust	**Internal attacks**	Moderate	High
Unsecured acknowledgments	**Spoofed acknowledgments**	High	Moderate
BPPM integrity	**ED/LC**	High	Limited

Regarding ED-LC, there is currently a lack of evidence of their feasibility in real-world settings. The evaluation has been made based on the assumption that their results in a real-world IPS would be comparable to the on obtained in simulation in [11], which is not verified yet. As a consequence, their likelihood is limited, but because of their capability to bypass 802.15.4 security mechanisms and their severity they are among the most critical attack vectors.

On the other hand, internal attacks and forged acknowledgment are relatively simple to mount. This is not a surprising result for internal attacks as they are considered as very challenging to detect [86, 99], and can be very harmful in applications in which the mobile tags are not trusted. However, forged acknowledgments have drawn less interest in the literature. Although their principle has been described in previous works [99], they have never been implemented or evaluated. As they represent a quite low-cost attack, hence are quite realistic in an IoT context, we investigated in depth the potential of these attacks and worked on standard-compliant countermeasures against them.

Chapter V Attacks against 802.15.4 IR-UWB Indoor Positioning Systems

We present in this section our work regarding the development of attacks against 802.15.4 IR-UWB. We propose an internal attack scenario that can be easily mounted against IR-UWB IPS. Regarding external attacks, we show that the basic forged acknowledgment scheme, often described as trivial in the literature, involve several major challenges for the attacker that have been neglected [102]. We identify the conditions and hypothesis under which these attacks can be harmful. We propose several novel attack schemes based on forged-acknowledgment, and propose notably an attack that can let an attacker take control over the position of a tag with a single malicious node when the tag is using scheduled acknowledgments. All these attacks have been implemented and evaluated on SecureLoc; we present our results regarding their performances and accuracy.

Chapter contents

V.1 Internal attacks

The simplest attacks against indoor positioning, and the most challenging to detect, are the internal attacks. An internal attack occurs when a tag that legitimately belongs to the IPS is trying to lying on its own position, typically by cheating during the ranging protocols. Internal attacks are fairly trivial to mount, as most ranging protocols are built upon the assumption of honest participation of the prover. In a ranging protocol, the prover has either to comply with a pre-agreed reply time (e.g., LTWR, as seen in section III.3.2), or return its timestamps (e.g., TWR), or both. In both cases, the node can easily cheat: a rogue node can use a reply time superior or inferior to the pre-agreed value, or modify its timestamps in order to increase or decrease the estimation of its reply time. Moreover, the rogue node has a complete control over the distance shift produced, as the relation between the time shift applied and the distance is known.

Hence, a cheating node can produce any distance it wants in a ranging protocol, which is why internal attacks are so challenging to detect. As long as the rogue node knows the position of the anchors, extending a distance tampering to a position tampering attack is a straight-forward problem. An example with three anchors is shown in **Fig. 20**.

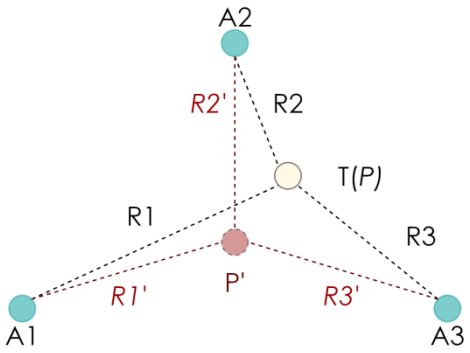


Fig. 20. Internal attack with three anchors

In this example the rogue tag is at the position P and wants to reach the position P'. The rogue tag T knows its distances to each of the three anchors, R_1, R_2, R_3. T can simply compute the distances from P' to the anchors, R_1', R_2', R_3'. Then, it can estimate the distance shift to apply to each anchor A_n as $\Delta d = (R_{n'} - R_n)$, and estimate the corresponding time shift to apply as $\Delta t = \Delta d / c$. Applying this time shift on the reply time estimated by the verifier is trivial, although it is done differently for TWR and LTWR. For TWR or any other ranging protocols requiring from the prover to submit its timestamps, the rogue tag can simply report fake timestamps in order to increase its reply time by the calculated time shift. For LTWR or any ranging protocol based on pre-agreed reply time, the rogue tag does not respect the pre-agreed reply time and simply acknowledges earlier or later than it should, based on the calculated time shift. In both cases, the rogue tag has a tight control over the distance shift applied, which makes the fraud challenging to detect. Indeed, as long as the rogue tag is

aware of the anchors positions, it can calculate and apply precisely the distance shifts to apply as shown in **Fig. 20**, and reach a perfectly coherent tampered position.

As discussed in the vulnerability analysis, the position of the anchors cannot be assumed to be secret; also, even if a tag is not informed of its own position, it might still learn it from an independent positioning mean. As a consequence, this attack can easily bypass the three detection factors.

We implemented internal attacks on SecureLoc in order to later evaluate countermeasures proposed in Chapter VI. The internal attack is based on a modified tag firmware, in which the timestamps reported in the DATA frame are tampered by the rogue tag. The time shift is calculated as explained in **Fig. 20**, and applied on t3. We assume that the rogue tag is informed of the distances computed by the anchors during the ranging protocol. The position targeted by the rogue tag can be controlled through the serial port, in which case the trajectory can be controlled through a graphical interface by the attacker. It can also be automated, by making the rogue tag targeting a still position a generating random motions. In the latter configuration, the rogue tag is autonomous and the serial link is not needed. Considering that the rogue tag needs to be aware of its own position, it needs to compute locally its real position. As the rogue tag knows the distance shifts applied on each anchor, it can retrieve the untampered distances (i.e.) to each anchor and apply multilateration algorithms to locate itself. We implemented the sliding window filter defined in section III.3.3 and the weighted centroid algorithm defined in section III.3.4 on the rogue tag firmware to do so.

This basic principle of internal attack can be applied to Sybil attacks (see section IV.2.2). If the TDMA scheme used allows a new incomer to request a slot, then it is possible for a malicious tag to create multiple identities, and claim a different position for each using an internal attack. In that case, the Sybil attack is not easy to detect as each of the created identities is located at a different place, making difficult to discriminate the real from the fake nodes.

V.2 Spoofed acknowledgment attacks

As far as internal attacks are relatively easy to mount and complex to detect, mounting external attacks against an 802.15.4 IR-UWB IPS implementing the security mechanisms defined in the standard (security level 5 in section IV.1.4) is more challenging. We already covered the major external attacks proposed in the literature, and demonstrated in section IV.3 that the vulnerability on acknowledgments in IEEE 802.15.4a leads to attacks with high likelihood. This vulnerability expose ranging protocols based on acknowledgments to distance tampering attacks. We conduct in in-depth analysis of spoofed acknowledgment attacks in this section, and show the limitations of the typical acknowledgment-based tampering scenario discussed in previous works. We propose several novel attack schemes based on spoofed acknowledgment, and highlight the factors that contribute to mitigate their effects, in order to propose low-cost countermeasures in the next chapter.

V.2.1 Principle

As highlighted in Chapter IV, the absence of integrity and authenticity mechanisms on acknowledgments exposes most common ranging protocols to distance tampering attacks. In the TWR protocol illustrated in Figure 2, the DATA frame containing (t2, t3) supports encryption and authentication but not the acknowledgment. The t4 recorded by the verifier can be lead to wrong distance estimation if the acknowledgment received is not from the original prover but has been actually spoofed by a rogue device (**Fig. 21**).

In such a case, t2 and t3 will be the authentic timestamps of the prover, however (t4 – t1) will be altered, as the prover records the reception time of the spoofed acknowledgment t4' instead of t4. Despite looking fairly simple, this attack is actually hard to tune in real life settings.

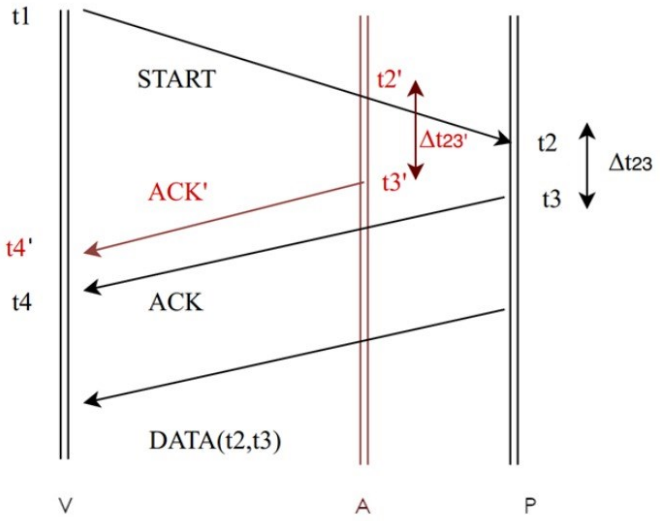


Fig. 21. Spoofed acknowledgment attack; Prover (P), Attacker (A) and Verifier (V)

To explain this attack in a simple context, we can assume that the attacker (A) is roughly at the same distance from the prover (P) as the verifier (V). Thus, the time-of-flight for both is similar. If the attacker wants to shorten the distance measured by Δd, the rogue acknowledgment ACK' should be received earlier than the prover, at $t4' = t4 - \Delta t$, with $\Delta t = \Delta d / (2^* c)$ according to (1). The time-of-flight being similar for the rogue and victim nodes, that implies that t3' must verify $t3' = t3 - \Delta t$. Hence, the reply time of the rogue node should be $\Delta t_{23'} = \Delta t_{23} - \Delta t$. Such attacks have been mentioned in previous works [103], but never implemented and evaluated to our knowledge. It has indeed been assumed in the literature that (1) tampering a ranging protocol with a forged acknowledgment was trivial and (2), that such attacks could be easily prevented by adding signatures on acknowledgments or replacing acknowledgments by other types of frames supporting integrity protection in the ranging protocols. Considering these assumptions, the acknowledgment problem was considered as trivial enough to not be investigated further.

However, the results we obtained after performing an in-depth analysis and evaluation of spoofed acknowledgment attacks (SAA) [102], which are presented in the following sections, contradict these assumptions. Regarding (1), we demonstrate that the inherent slight indeterminism of the devices processing time in a ranging protocol have a major impact on the consistency of the attack, leading to a low probability of hitting a realistic distance (i.e., within the maximum range of UWB). We verify this observation experimentally, and show that the tampered distance was in most cases unrealistic, making this attack. Regarding (2), adding signatures on acknowledgments is not standard-compliant and involves an additional cryptographic cost. Replacing acknowledgments by another type of frame supporting integrity and replay protection considerably increases the communication and computational cost of the ranging protocol. It can also conflict with certain features of UWB chips (e.g., the fixed format of acknowledgments allows them to be automated and handled directly by a hardware circuit).

V.2.2 Attacker's success rate analysis

We demonstrated in **Fig. 21** that in a SAA the attacker has to learn the reply time of the victim Δt_{23}. Δt_{23} on a node with the faster settings (Teensyduino overclocked at 96 MHz, SPI communication with the DWM1000 at 20 MHz) is on average 230 µs. After profiling that processing time, we observed that 85% is dedicated to UWB communications, 15% to the SPI Master-Slave communications and 5% to the program running on the Teensyduino. However, Δt_{23} has some variability due to UWB and SPI communications. We measured the probability distribution of Δt_{23} (**Fig. 22**), and obtained a Gaussian distribution with a standard deviation of 1.7 µs. An average variability of 1.7 µs is roughly equivalent to a distance shift of half a kilometer. Given that UWB industrial systems have a maximum range of about 100 m in the most optimistic case, this variability on the distance shift is far above realistic values. Hence, such an attack will not produce realistic values most of the time, but there is still a certain probability that it could happen. In the following, we estimate that probability analytically.

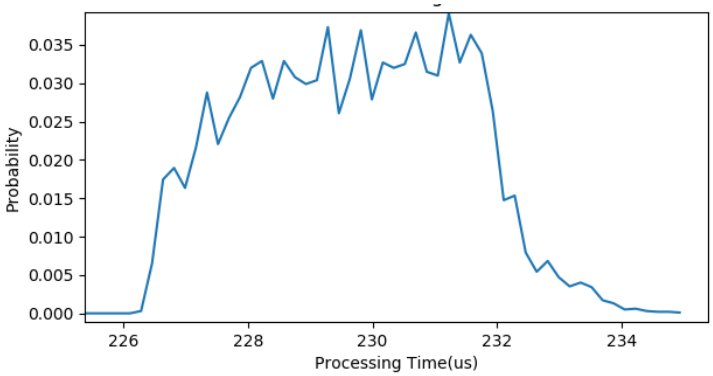


Fig. 22. Δt23 probability distribution (SPI speed = 20 Mhz, Preamble Length = 64 Symbols)

From that distribution, we calculate in **Table 24** the probability that the distance-shift |Δd| induced by the attack is below a Maximum Realistic Distance (MRD) threshold for two case: (1) when the attacker uses a Decawino with the same settings and program as the victim node (i.e., has the same probability distribution), and (2) when the attacker uses presumably ideal hardware, i.e., with no variability in the processing time, with Δt23' equal to the center of the Gaussian distribution in Figure 4 (230 ns).

Table 24. Probability (|Δd| < MRD) to get a realistic distance shift for different MRD threshold

MRD threshold	**10 m**	**50 m**	**100 m**
(1) Attacker with a DecaWino node	4.9%	16.1%	30.8%
(2) Attacker with ideal hardware (no variability)	6.0%	20.6%	36.1%

We observe that if the success probabilities are obviously higher with the assumption of ideal hardware with perfectly steady reply times, they actually remain close to the probabilities calculated for an attacker using a regular Decawino node. This implies that enhancing the stability of the reply time on the malicious node would make little difference in the attack success rate. The unpredictability of the process will remain due to the inconsistency of the victim node. The only alternative for the attacker is to use shorter frames and wait for the victim's acknowledgment to start in order to properly adjust the timing of the forged acknowledgment, which we discuss later in section V.2.2.

V.2.3 Overlap effect

So far, we only considered the absolute value of the distance shift. Two cases can actually occur when using an acknowledgment attack: distance enlargement or distance reduction. If an intuitive justification for a distance reduction would be that it happens when the rogue device acknowledges faster that the victim, which has been discussed in [11], the explanation for distance enlargements is less obvious.

The standard deviation of Δt_{23} that we estimated at 1.7 μs is actually shorter than the duration of two preamble symbols, which means that most of the time a part of the preambles of the spoofed and the authentic acknowledgment will occur simultaneously, as shown in **Fig. 23**. Preamble symbols have a low inter-correlation such as allowing multiple preambles emitting at the same time with minimized losses [6], which explains why both frames are not lost.

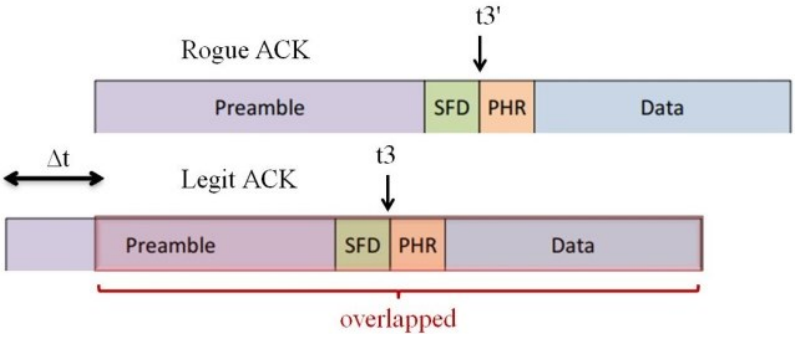


Fig. 23. Preamble overlap

Thus, either the receiver will tune on the earliest preamble and ignore the last one, which is known as capture effect either the latest preamble will overlap the earliest one [19]. This second scenario leads to t4 being delayed and the distance being increased.

Note that the probabilities calculated in **Table 24** are based on the assumption that the rogue acknowledgment systematically overlaps the legit one. We show in the next section that this overlap probability is actually highly dependent on the received signal intensity by the anchor of each acknowledgment.

We evaluated the ranging output on a victim prover targeted by a spoofed acknowledgment attacks mounted on a malicious Decawino node, with different configurations. The difference of signal intensity between the rogue and legit acknowledgment has a clear incidence on the output of the attack. Three cases can be observed: distance enlargement, distance reduction, or distance unchanged. The scenarios leading to these outputs are shown in Figure 6.

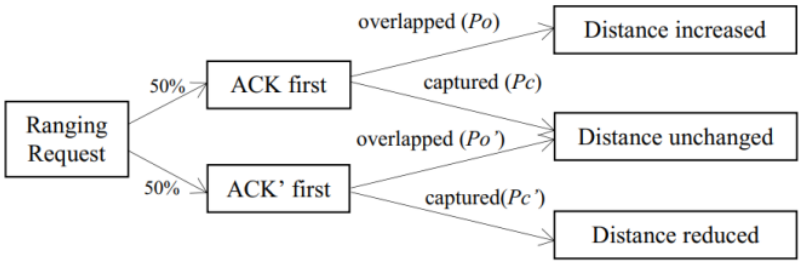


Fig. 24. Possible SAA scenarios

We showed previously that the time-of-flight is negligible compared to the variability of the processing time. Hence, as the rogue and victim node run both the same program, they have both a 50% probability to be faster (Figure 6). The authentic distance will be measured if the legit acknowledgment is received, i.e., if ACK arrives first and is captured, or if ACK arrives last and overlaps ACK'. If ACK' is first and captured, t4' is inferior to t4 and the distance is decreased; if ACK' is last and overlaps ACK, t4' is superior to t4 and the distance is increased.

Increasing the transmission power of the rogue device increases the probability of a distance modification. We increased gradually the difference of Received Signal Strength

Indicator (ΔRSSI) measured by the anchor between the legit and spoofed ACK, and estimated the overlap probability on a set of 1000 ACK for different ΔRSSI values. The results are shown in **Table 25**.

Table 25. ACK attack output probabilities for different ΔRSSI

Case	1	2	3	4	5
ΔRSSI	-4dB	-2 dB	0	+2 dB	+4dB
Enlargement	0%	0.8%	1.8%	20.5%	45.5%
Reduction	0%	21.2%	47.8%	47.1%	47.5%
Unchanged	100%	78%	50.4%	32.4%	7%

We can observe that the overlap success rate is strongly related to the difference of reception power. We define as **reception power advantage**, or RSSI advantage, the difference of reception power between the victim's and the attacker's acknowledgments. When the attacker's reception power advantage is low (ΔRSSI= -4dB), none of the attacks has been successful, giving an overlap probability for the attacker of 0%. Symmetrically, that implies that the overlapping probability for the victim node is also 0% in the opposite configuration. When the attacker's reception power advantage is high (ΔRSSI= +4dB), enlargement and reduction are almost equiprobable and the actual distance is received only with a 7% probability, giving an overlap probability of 93% for the attacker. Therefore, the attacker should increase its reception power advantage to optimize the attack results. This can be achieved by boosting transmission power beyond regulations, or by bringing the rogue node closer to the anchor, the time-of-flight variation induced being negligible for the success rate of the attack.

In the case of a successful distance modification, we measured the percentage of realistic distance shifts ($|\Delta d| < MRD = 100$ m), on a set of 1000 distance shifts. The rogue tag was placed close to the anchor for this experiment. The results are given in **Table** 26.

Table 26. Probability (|Δd| <MRD) for various Maximum Realistic Distances (MRD) when a distance shift has been obtained

MRD threshold	10m	50m	100m
Analytical Estimation	4.9%	16.1%	30.8%
Experimental results	4.1%	13.8%	25.9%

The probabilities presented previously in **Table 24**, shown in the first line of **Table 26**, have been calculated with the assumption of a systematic overlap of the rogue acknowledgment. The actual overlap probability of the attacker is 93%, which explains why the experimental results in **Table 26** are slightly below the probabilities estimated in **Table 24**. Besides that, these results are coherent with the analysis conducted earlier, with a probability of 25.9% for MRD = 100 m.

After measuring the probability that a distance shift is realist, we can estimate the global probability for a successful enlargement or reduction attack. The requirements for a

successful enlargement or reduction attack are that (1) it produces the desired distance shift (enlargement or reduction) and (2) that this distance shift is below 100 m. Given that the rogue tag is close to the anchor, when the rogue acknowledgment is first (50% chance), the legit acknowledgment will always be ignored as seen previously, implying Po' = 0. Hence, combining the probabilities given by Figure 6 and Table V, the probabilities that an attack achieves a successful enlargement (P_e) or reduction (P_r), i.e., generates a distance shift $|\Delta d| < 100$ m, are given by:

$$Pe = \frac{1}{2} * Po * P(|\Delta d| < MRD = 100\,m)$$

$$Pr = \frac{1}{2} * Pc' * P(|\Delta d| < MRD = 100\,m)$$

With Po = 86% and Pc' = 1 – Po' = 100%. With the measurements provided, we get P_e = 11.14 % and P_r = 12.95 %.

Summary

The two main conclusions regarding the basic SAA are the following:

1. A realistic distance shift Δd is necessarily below 100 m, given the range of UWB-IR technology. The corresponding time-of-flight (ToF) shift Δt = t4' – t4 should verify Δt < ToF(100 m) = 0.33 μs. A preamble symbol has a length of about 1 μs, and at the receiver level several symbols are required before the preamble detection occurs, as detailed in section II.1.5.1. As a consequence, to get a realistic attack, the time gap between the legit and rogue acknowledgments should be much lower than the preamble detection time, which also means that they will strongly overlap. Hence, if the attacker is targeting a realistic distance shift, ACK and ACK' will start overlapping before the preamble detection on the verifier's side. The behavior resulting from this collision has not been considered in previous works.
2. The accurate ToF estimation occurs after the SFD is received, which means that a receiver cannot determine accurately when a frame started being emitted during the preamble phase. That implies for an attacker to predict the processing time of its target, Δt23. In most systems, the prover always executes the same piece of code after receiving the START frame, and it does not look like an actual issue at first sight, but time-of-flight measurements work on a much higher time resolution than usual systems do. Clock drifting, internal (Serial Peripheral Interface) and external (UWB) communications inconsistencies lead to a slight indeterminism on Δt23. Basically, even if the attacker aligns its own processing time with the average processing time of the victim, and even assuming that the attacker does not suffer from processing time indeterminism on its own hardware, the distance output produced by the attack in **Fig. 21** will often be off by hundreds of meters. This is a real issue for an attacker as such inconsistencies are strongly unrealistic.

The experimental results regarding these two points are the following:

1. Regarding the overlap of ACK and ACK', complete collisions are very rare (less than 10% of cases) and most of the time the victim captures the one with the highest received energy. An attacker can increase the odds to overlap the victim's acknowledgment with a higher transmission power. This can be achieved either by getting closer to the victim or emitting beyond the regulations. We estimated that with a RSSI difference of +4 dB observed on the receiver between ACK' and ACK, ACK' has a 93% to be captured.
2. Regarding the indeterminism of $\Delta t23$, when evaluating our implementation we obtained a success rate (i.e., a realistic distance shift) of 26%. We evaluated that if the attacker has ideal hardware (i.e., no indeterminism on its own processing time and unlimited transmission power) this probability could go up to 36%.

At this stage, using regular hardware, this attack has only one on four chances to hit a plausible distance and even when it happens the distance remains random. Regarding the attack classification discussed earlier, this attack falls more into the category of fine DoS attacks rather than distance reduction or enlargement. However, this basic attack scenario can be exploited more efficiently in more complex attack schemes.

V.2.4 Solving the prediction problem

Considering that the reply time prediction problem is non-trivial for the attacker when communications are not tightly scheduled, an alternative that an attacker could potentially consider is to estimate the legit acknowledgment arrival time based on the detection of its preamble.

Until then, we assumed that the attacker is using the same preamble length as the victim, which implies that the rogue acknowledgment has the same duration as the legit one. The receiver needs indeed to receive several preamble symbols to detect the transmission, the exact number being relatively random depending on the transceiver/receiver synchronization. Each symbol being equivalent to a distance of roughly 150 m in a TWR protocol, the delay induced by the preamble detection time will lead to largely unrealistic distance enlargements if the attacker has to wait for the victim's preamble. However, it becomes possible if the attacker uses shorter preamble than the victim. The protocol that the attacker can apply in that case is shown in **Fig. 25**. The attack is seen from the perspective of the verifier receiving the acknowledgments. The potential difference of time-of-flight between the rogue and legit acknowledgments (e.g., if the attacker is closer to the verifier than the victim) is not represented as the impact is negligible compared to the order of magnitude of the reply time the preamble detection time. For the sake of simplicity, we assume that the attacker is as the same distance from the verifier than the prover. In **Fig. 25**, the attacker waits the victim's preamble and timestamps the detection time T_{pre}. The attacker knows the prover's preamble length P, as well as the time-of-flight between the prover and verifier *tof*(P->V), which he or she can learn from previous observations. As a consequence,

with T_{sym} the preamble symbol's duration and T_{SFD} the SFD's duration, it can estimate the acknowledgment arrival time Δ as:

$$t4 = T\_pre + P * T\_sym + T\_SFD + tof(P \rightarrow V)$$

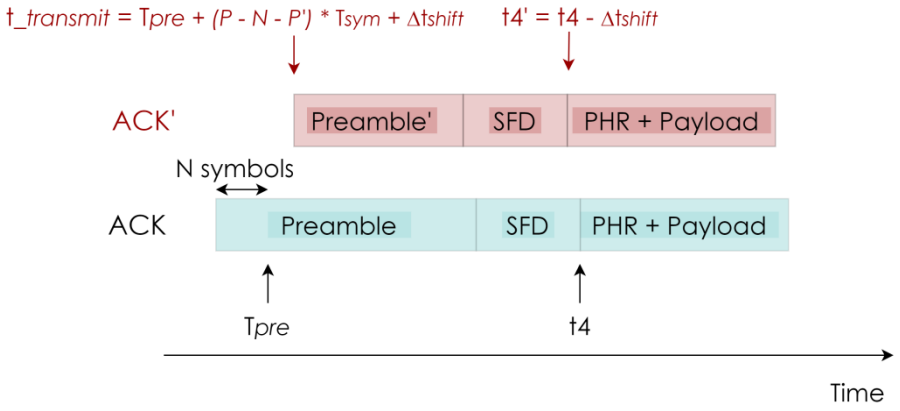


Fig. 25. SAA based on victim's preamble detection

From that, if the attacker uses a preamble length P' < P, detects the victim's preamble after N symbols and wants to achieve a time shift Δt_{shift}, he or she can schedule the departure time of the forged acknowledgment as:

$$t_{transmit} = T_{pre} + (P - N - P') * T_{sym} + \Delta t_{shift}$$

As far this approach might appear as an efficient solution for the prediction problem, there are four major flaws that make it very limited, which we observed experimentally:

1. As discussed before, the attack cannot learn the exact number of symbols elapsed N, even with superior hardware [11]. It can only infer a value based on the previous observations, but given that N is highly random the prediction will be biased. As a consequence, the attacker is not estimating t4 but t4 modulo[1] T_{sym}.
2. The whole point of the preamble in 802.15.4 physical layer is to synchronize tightly the receiver on the transceiver's preamble symbols, such as obtaining an accurate timestamp at the end of the SFD. The timestamp obtained at the preamble detection, even if the correct N is inferred, is far less accurate than the SFD timestamp, leading to further inconsistencies.
3. Waiting for the preamble detection means that the verifier will be also able to detect the preamble. As a consequence, even if the following overlap is successful, the verifier will be aware that a frame has been denied prior to the acknowledgment reception, which is not normal for a TDMA scheduling scheme. If that interference is

[1] Defined here as a floating-point modulo operator

not deliberate (e.g. presence of an unaware UWB device that is not part of the network nearby), switching to a different preamble code should fix the problem.

4. If the verifier has already detected the preamble, the verifier is locked on the prover's timing, which makes the whole overlap process much harder given the impulse scheme of UWB. The reception power advantage required to obtain overlap success rate similar to the one obtained with the basic SAA are much higher.

To illustrate these flaws, we actually mounted this attack and recorded the tampered distances obtained. The implementation of the physical layer on the DWM1000 does not allow using any preamble length; as a consequence, we used a preamble length of 128 symbols on the prover's tag and 64 symbols on the malicious tag. In the following scenario, the malicious tag and the prover are both placed at 5 m from the verifier. The malicious tag has a boosted transmission power leading him to a reception power advantage of +4 dB, and aims for a distance increase of 1 m. The malicious tag implements the SAA protocol described above in **Fig. 25**. As the malicious tag cannot estimate the number of symbols elapsed N upon preamble detection, it simply infers that N = 12: this is the mean value of N that we measured on DWM1000 nodes[1]- hence, the most likely. We recorded the results of a 1000 TWR protocols between the prover and the verifier, during which the malicious node was interfering with the proposed attack.

These results corroborate the four flaws aforementioned, considering that:

1. The distances are randomly shifted by an integer multiple of D_{sym}, with D_{sym} corresponding to the equivalent distance of the symbol period T_{sym} (~1 μs), i.e., roughly 150 meters[2]. As a consequence, most distances are very unrealistic. This is due to the impossibility for the attacker to estimate the exact number of preamble symbols elapsed.
2. However, if we ignore this random symbol shift and calculate the distance d(P->V) obtained modulo D_{sym}, it appears that besides the symbol count error the attack is relatively steady. We obtained a mean of 2.13 m and a standard deviation of 8.1 m for a distance d(P->V) modulo D_{sym}. This is still far from the 10 cm accuracy obtained in normal conditions, and gives an estimation of the accuracy of the preamble detection timestamp provided on the DWM1000. Also, 26% of the tampered distances are completely off even with after applying the symbol shift correction, i.e., more than 30 m away from the target distance.
3. The prover's preamble has been detected every time by the verifier, even when the legit acknowledgment is later overlapped.
4. The attacker actually fails the overlap process quite often. 50.7 % of the distances received were legit as the forged acknowledgment did not overlap the legit acknowledgment. For the same reception power advantage of 4 dB, we had an

[1] The preamble detection time depends on various parameters, notably the chunk size defined for the autocorrelation, which can be defined to different values on DWM1000. This aspect has a minor importance here.

[2] As TWR is based on a round-trip, D_{sym} is actually half of the distance traveled by light in 1 μs

overlap rate of 93% in the previous experiments. This shows that the overlapping process is harder when the prover's preamble has already been detected.

Regarding this last point, we estimated the reception power advantage required from the malicious tag to achieve an overlap close to 100%. We reached 95% for a reception power advantage of 10 dB which can be achieved by setting the prover's power very low, the attacker's power to the maximum (way beyond regulations) and placing the attacker close to the verifier. As a consequence, this attack has a higher hardware cost compared to the basic scenario proposed earlier in section V.2.1.

There are a lot of similarities between the proposed scenario and the ED-LC attacks in [11] regarding the four challenges faced by the attacker. Concerning the first point, they learned only the arrival time modulo T_{sym} and use early SFD detection (ED) to learn when to stop the preamble. Even with the ED attack, the malicious tag will start the SFD later than the victim; to fix that problem, they propose to use a shortened, yet standard-compliant SFD. However, they do not address the second point, as the feasibility of a preamble timestamping circuit that would reach the same level of accuracy as the regular 802.15.4 PHR timestamping is not discussed.

Considering these results, preamble detection-based SAA cannot achieve the accuracy and consistency required to bypass the system's layer basic detection factors. Nevertheless, that could allow to obtain a more stable output than the basic SAA described in the previous sections, with a 8.1 m standard deviation on the tampered distance, using the ED scheme proposed in [11], assuming their hardware feasibility.

V.2.5 SAA against Delayed Send feature

Basic TWR protocol does not rely on pre-agreed reply times, which means that the prover can theoretically wait as much time as it wishes before sending ACK and before sending DATA without compromising the protocol. However, if LTWR is used, or if acknowledgment frames are sent with the *delayed send* option, a pre-agreed reply time will be respected with a high accuracy by the acknowledgment expedient. In that case, we measured previously in section III.3.2 the indeterminism on the reply time, which is tremendously reduced with a standard deviation in the order of magnitude of a few nanoseconds. Hence, if the delayed sent feature is used by the prover for the ACK frame, the attacker can tamper the distance accurately: **Table 27** shows the results obtained for different distance shifts for two different configurations, shown in **Fig. 26**. The prover is placed at a distance d from the verifier; in the first configuration, V is close to P (1 m), in the second configuration V is further way from P (10 m). The attacker is applying successively different distance shifts (1 m, 10 m and 50 m) to the ranging protocols of the victim such as reaching a target distance Δd.

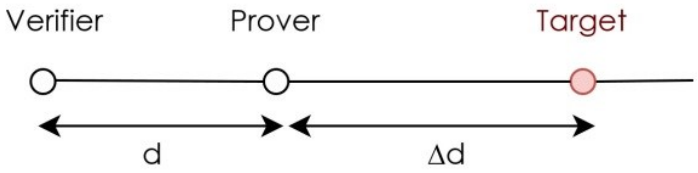


Fig. 26. Prover (P) and Verifier (V) configuration during the attack

We measured the average accuracy of the attack with these different configurations, the accuracy being defined as the absolute difference between the distance targeted by the attacker and the distance obtained. The experiments have been done with 10 different Decawino-based prover/verifier pair on SecureLoc. The reply time was set to 1 ms. 1000 samples have been collected for each pair and each configuration for a total of 10000 samples per configuration. The results are shown in **Table 27**.

Table 27. Average Accuracy of SAA with Delayed Send enabled, obtained on 10 different prover/verifier pairs on SecureLoc

Distance shift Δd	V 1 m away from P	V 10 m away from P
1 m	44 cm	48 cm
10 m	43 cm	48 cm
50 m	45 cm	49 cm

We can observe that the accuracy of this attack is nor dependent on the distance between P and V, nor on the distance shift added by the attacker. The error on the distance shift is below 50 cm, and if that error is superior to the usual noise level in clear environments, it is actually on a similar scale as the noise observed in complex environments (e.g., without Line-of-Sight, in presence of metallic obstacles), as seen in section III.3.2

V.2.6 Enhanced Spoofed Acknowledgment Attack

We showed that the main problem faced by the attacker is the slight indeterminism of the processing time of its victim. Only a small fraction of the attacks will end up in realistic distances; we demonstrate in this section how this small fraction can be exploited to mount more realistic attacks against unscheduled transmissions.

Perturbations and frames loss are common place with UWB, even in non-adversarial conditions. If a mobile tag goes back and forth from a place where the link with an anchor is lost, rangings will fail intermittently. More generally, when approaching an obstacle affecting the quality of communications, there is an intermediate state before the complete link disruption where only a small fraction of the rangings succeed, which can be observed for Line of Sight Indicator (LOSI) superior to 20 dB. Also, distances obtained through the ranging protocols are filtered (L3 layer) before being processed by the multilateration layer (L4). As far as it reduces the impact of noise on the position calculated, it can also benefit to

an attacker by improving the steadiness of the attack. Thus, the attacker can exploit that to improve the basic and limited spoofed acknowledgment attack introduced in section V.2.1

We estimated previously that a rogue device using a SAA has a chance of 26% to hit a distance shift below 100 m when acknowledgments are not tightly scheduled. If the rogue device is able to deny the 74% remaining ranging, then it can obtain realistic distances. TWR is completed once the DATA frame containing (t2, t3) is received. Hence, the challenges for the attacker are to:

1. Evaluate the distance shift directly after ACK and ACK' overlap, before the DATA frame is received by the prover.
2. Deny the DATA frame if the estimated distance shift is not satisfying, i.e., in the interval desired by the attacker (for this example, [0, 100 m]).

We refer to the attacks built on that scheme as *Selective SAA*. We developed two attack protocols for that purpose, one based on DATA jamming and one based on a relay attack. The first scenario can tamper the distances measured between a prover and a verifier that are in range of each other. The second one can tamper an out-of-range prover in order to make the prover seem to be close to the verifier.

Selective SAA with DATA jamming - Principle: This attack requires two malicious nodes, ARX and ATX, shown in **Fig. 27**. ARX is placed close to P and ATX close to V. The role of ATX is to send the forged acknowledgment ACK', and the role of ARX is to receive the victim's acknowledgment ACK. For clarity's sake, we assume in the following explanation that ARX and ATX are respectively placed close enough to P and V so the *tof*(ATX->V) and *tof*(ARX->P) can be neglected, hence *ToF*(P->V) ≈ *ToF*(ATX->ARX).

ToF(ATX->ARX) has been measured prior to the attack by ATX and ARX with a regular ranging process. The four steps of the attack scenario are detailed below and illustrated in **Fig. 27**:

1. This attack starts exactly like the basic SAA. V sends START at t1; P receives START at t2 and replies with ACK at t3. Meanwhile, ATX sends ACK' at t3' = t2' + Δt23', using a scheduled transmission. The value of Δt23' should be as close as possible to the average processing time of P, i.e., Δt23. V captures ACK' and not ACK because ATX is very close. On the other hand, ARX receives ACK and not ACK' because P is very close.
2. As previously, the time-of-arrival at V's location of ACK is notated t4.
   As *ToF(*P->ARX) is negligible, ARX receives ACK at t3 = t4 - ToF. ATX has now to learn t4 to evaluate if the attack is going to produce a realistic distance shift, hence needs to learn t3 to compute t4 = t3 + *ToF*(P->V). To do that, immediately after receiving ACK, ARX proceeds to transmit t3 to ATX as fast as possible. For that purpose, ARX switches to the shortest UWB preamble length (64 symbols on the DWM1000) in order to have the shortest frame possible. Then, ARX sends a frame with empty payload at t3 + T_{delay} with the delayed send feature, where T_{delay} is very short and known by ATX.

3. ATX receives the empty frame at t3 + *ToF*(ARX->ATX) + T*delay* = t4 + T*delay*. Hence, ATX can retrieve t4, calculate Δt_{shift} = t4' – t4, and estimate the distance shift Δd_{shift} as $\Delta d_{shift} = \Delta t_{shift} * c$.
4. If Δd is not within its target interval [D_{min}, D_{max}], ATX starts jamming the DATA frame, preventing the TWR protocol from completing.

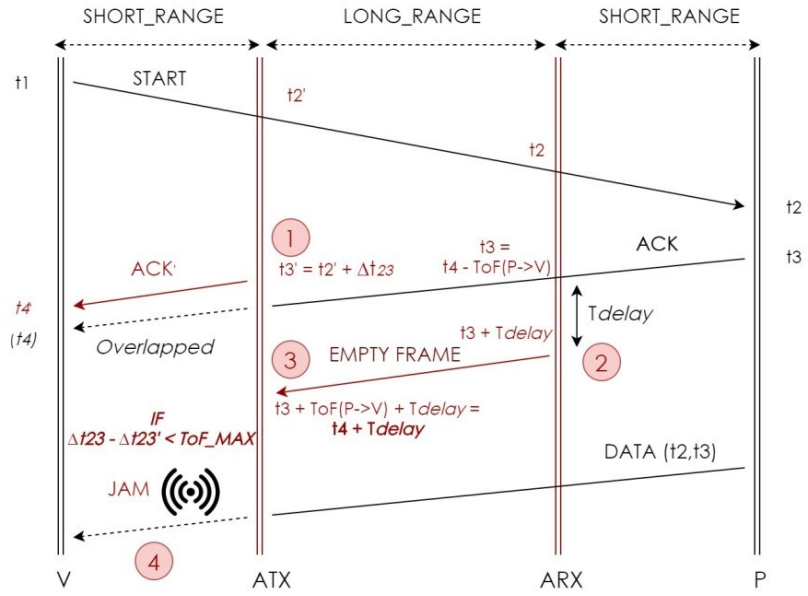


Fig. 27. Selective SAA with DATA jamming

Setup- We implemented this attack on two Decawino nodes. We set T*delay* = 92.5 µs which was the shortest delay that ARX could achieve to send its frame to ATX. To perform jamming on V, we set ATX transmission power to the maximum available on the DWM1000 (amplification coarse grain +18 dB, fine grain +15.5 dB), which is way beyond the regulations. After that, ATX proceeds to disrupt the DATA frame, using a dummy frame with a long preamble length (e.g., 2048 or 4096) to overlap the legit frame. We discuss the efficiency of this jamming method below, but one can consider that an attacker could use specific hardware for that purpose that would make that task easier [105].

Selective SAA with Relay - Principle: This scenario combines the SAA with the well-known relay attack, defined in section II.2.1.This scheme is represented in **Fig. 28**; the START frame is relayed by ARX to ATX, and the replies of P are relayed back by ATX to ARX.

As we already discussed, basic relay attacks are a limited threat against UWB ranging considering the delay they induce on the reply time. Even with speed optimization, the delay induced by the round trip is equivalent to hundreds of kilometers at the speed of light, leading to an unrealistically high distance and making the attack quite obvious. However, this delay problem is fixed if the attacker does not have to wait for the ACK to arrive from the other end of the relay: the attacker can simply forge itself the ACK frame and imitate the

processing time of the victim. Meanwhile, the TWR protocol is reproduced at the other end of the relay after START is relayed, and ARX can relay the ciphered DATA frame back to ATX, which is then replayed to V by ARX. DATA frame are not timestamped, so the delay induced by the relay is not detected by the verifier. Following this principle, we proposed the following attack protocol, shown in **Fig. 28**. In this scenario, P and V are out of range of each other. ARX is placed close to V and ATX close to V. For clarity's sake, ARX and ATX are assumed to be very close to their targets, so we can neglect the ToF between them. This protocol is still effective if it is not the case, with minor adjustments in the calculations to include an additional time-of-flight.

1. ARX relays the START frame immediately after receiving it at t1. Then, ARX sends an ACK' at t2' = t1 + Δt23'. The value Δt23' is known by both ATX and ARX and should as close as possible to the average processing time of P.
2. ATX replays the START frame at the other end of the relay at t1 + T_{relay}. P is too far from V to receive the original START. Hence, from P's perspective, the START frame seems to have been emitted by V. P receives START at t2 and replies with an ACK at t3 = t2 + Δt23, which cannot be heard by V. Given the proximity assumption we have t1' = t2. Then P sends the DATA frame containing (t2, t3).
3. ATX receives ACK at t4' = t3 because of its proximity with P. Then, ATX computes Δt23= t3 – t2 = t4' – t2'. Finally TX calculates Δt_{shift}= Δt23' - Δt23 and extracts the distance shift $\Delta d_{shift} = \Delta t_{shift} * c$. If Δd_{shift} is in the target interval [D_{min}, D_{max}], ATX relays the DATA frame, otherwise it does not. If DATA is relayed, ARX replays DATA to P; otherwise, V never receives DATA and the ranging protocol fails.

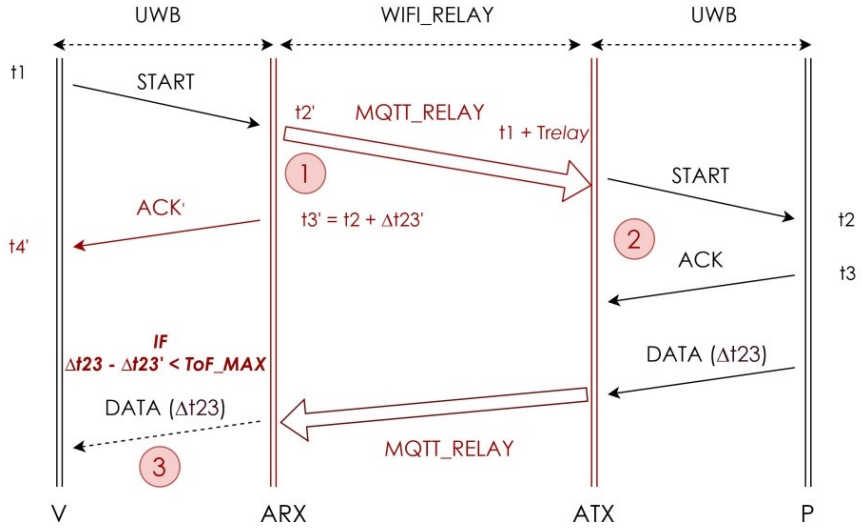


Fig. 28. Selective SAA with relay

Setup- We also mounted this relay attack on two Decawino nodes connected to two Raspberry Pi 3 through a USB serial link. The relay was implemented with an 802.11n long-range link with MQTT protocol, implemented on the Raspberry Pi.

The results obtained for these two attacks are shown in **Table 28**. Any attack producing a distance within the interval defined by the attacker is considered as a success, whereas in the for distances outside this interval the attack is considered as failed. Jammed or unrelayed ranging protocols are considered as denied. In the scenario combining SAA with jamming, the prover P and verifier V were placed 10 m from each other. In the scenario combining SAA with relay, P and V were not in the same room. In both configurations, ATX was placed close to V and ARX close to P as defined earlier. The performances of these two attacks have been estimated on 2000 TWR protocols between P and V.

Table 28. Success rate of the attacker with Selective SAA

Interval		**SAA + Jamming**			**SAA + Relay**		
D_{min}	**D_{max}**	**Success**	**Denied**	**Fails**	**Success**	**Denied**	**Fails**
25 m	50 m	5.1 %	92 %	2.9%	6.3 %	92.9 %	0.8 %
50 m	100 m	10.1 %	85.2 %	3.7 %	12.1 %	87.2 %	0.7 %
10 m	100 m	20.6%	74.2 %	5.2 %	23.7%	75.8 %	0.5 %

The success rate obviously depends on the length of the interval. With an interval length of 90 m we get a 20.6% success rate for selective SAA and 23.7% for relay, while with an interval length of only 25 m these success rates are very low with respectively 5.1% and 6.3%. Selective SAA have slightly lower success rates and higher fail rates due to a less powerful denying process. The DATA frame can actually sometimes pass the jamming process. One has to be aware that this jamming process is done on a cheap Decawino tag, and the attacker could simply use more advanced hardware. For the relay on the other hand, the denying process consists simply of not sending the DATA frame, and cannot fail. The few distances outside the interval set by the attacker were mistakenly considered as valid by the attacker due to the slight inaccuracy in the calculation process, and are actually distances very close to the boundaries of the interval.

These attacks allow an attacker get a finer control on the distances obtained, especially if these distances are processed by a median and/or average based filter (L3), which will tend to return the center of the distances interval picked by the attacker. Considering that, the filters at L3 may help the attacker get a higher accuracy on the distance shift; nevertheless, these attacks lead to a much higher standard deviation in the distances obtained than the one observed in non-adversarial settings. As a consequence, it is obvious that these attacks cannot bypass the system detection factors: they could potentially be used against ranging-only applications, but is it is relatively trivial to detect them in an IPS.

However, in the case of scheduled transmissions studied in section V.2.5, the accuracy of basic SAA is high enough to be undistinguishable from untampered rangings. In that case SAA can be extended to the multilateration layer. We define an attack scenario against multilateration in the following sections.

V.2.7 Unveiling nodes' positions

In this section, we show how SAA can be exploited to take control over the victim's position when scheduled transmissions are enabled. We first explain how the attacker can retrieve its target's position, which will contribute to make the attack realistic. Then, we show how he can extend SAA to the whole positioning system.

If the victim uses precisely scheduled transmissions, the attacker can potentially take an accurate control over the distances between the anchors and the victim. The attacker can then proceed to attack the multilateration layer and take control over the position of a tag by tampering the ranging of every anchor. However, to do that in a realistic manner, the attacker has to know the current position of the tag. Indeed, a sudden position jump would be suspicious, and a more realistic scenario is to move the tag's position slowly (i.e., at a reasonable speed) away from its current place. The real position of the tag can be retrieved by the attacker by different ways:

- As part of the applicative context (e.g., a robot performing a scheduled task at a given spot)
- By using another localization mean (e.g., RSSI analysis or vision-based positioning)
- By eavesdropping the ranging processes of all the anchors.

We show here how the attacker can proceed in the last case. We assume that the attacker ignores the target position and the scheduled processing time Δt23 of the victim, and aims to retrieve the distance between P and V by eavesdropping their TWR protocol. The configuration is shown in **Fig. 29**. We notate t2′ the START reception time and t4′ the ACK reception time of A. Based on **Fig. 29**, for an attacker A, a verifier Vi and a prover P, we have the following relations for the timestamps t2′ and t4′:

$$\Delta t24' = t4' - t2' = t1 + tof(V_i \rightarrow P) + \Delta t23 + tof(P \rightarrow A) - (t1 + tof(V_i \rightarrow A))$$

Giving:
Eq. 7

$$tof(V_i \rightarrow P) + tof(P \rightarrow A) = \Delta t24'_i - \Delta t23 + tof(V_i \rightarrow A)$$

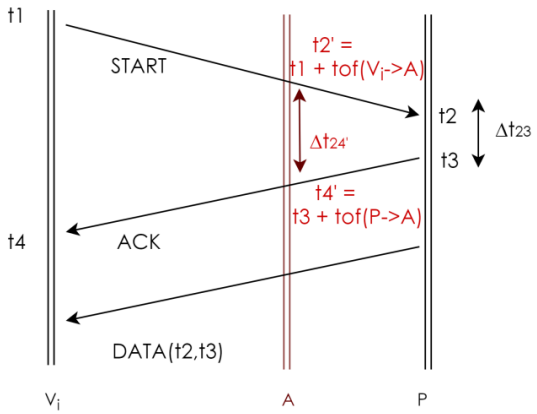


Fig. 29. Eavesdropping on TWR protocol

We assume here that the displacements of P between the ranging of the first anchor and the ranging of the last anchor are negligible; hence, the distance between P and A is the same at the time of the ranging protocol is the same for each anchor. Thus, *tof*(P->A) is the same for every anchor V_i. A is aware of its own position as well as the anchor's positions so *tof*(V_i-> P) is known. Δt_{23} is known by A and $\Delta t_{24}'$ is measured by A.

Thus, A can obtain *tof*(V_i ->P) + *tof*(P->A) for every anchor V_i. Assuming there are N anchors, A disposes of the distance set $D = \{d(P \rightarrow V_i) + K, i \in \mathbb{N}, i \leq N\}$, where K = d(P->A) is an unknown constant. Then, A can compute its position as following.

Given a chosen multilateration algorithm M (e.g., Gauss-Newton), A must find x such as minimizing the Total Ranging Deviation (TRD) of $M(\{d(P_i \rightarrow V) + x, i \in \mathbb{N}, i \leq N\})$. This can be achieved with a reasonable complexity by applying a line search algorithm [107] on x. Doing so, A obtains d(A->P) as the solution of the line search and its position as the solution of $M(\{d(P_i \rightarrow V) + d(A \rightarrow P), i \in \mathbb{N}, i \leq N\})$.

Hence, A can retrieve its target's position by simply eavesdropping the ranging process. This means that the attack can be mounted even without precise information on the target's position. As discussed in section II.2.1, this also highlights that cryptographic means are not enough to assert the privacy of the position: here, the attacker does not need to be able to decipher any of the frames.

V.2.8 Attacking the multilateration layer

We demonstrated in section V.2.5 that an attacker can control the distance output of the TWR protocol with an error below 50 cm if the prover uses scheduled transmissions on ACK frames. In that case, a malicious node can take control over its victim rangings. We explain in the following how this attack can be extended to the multilateration layer and allow a single malicious node hijack an honest one.

In order to control the prover's position, an attacker has to modify the distances measured by all the anchors involved in the verification process. We propose the following attack scenario against the multilateration layer when the prover uses scheduled transmissions, with the following assumptions:

- Anchor positions are known by the attacker: anchors are fixed stations and usually placed in clear spots such as minimizing perturbations. As a consequence, when considering the threats facing a localization system, anchor positions should never be considered as confidential.
- The attacker knows its own position.
- The attacker knows the target's position – he or she can get this information following the protocol described in section V.2.7.

Principle- This attack scenario is shown in **Fig. 30**. The prover P is a static node at the moment of the attack, at the position (x,y,z); the attacker aims to make P appear at the target position T(x',y',z') using a single node ATX. To reach this target position, with a configuration with N anchors, the attacker must modify the anchor ranging outputs

(R_1, ... , R_N) to (R'_1, … , R'_N). The positions of the anchors are known so for any anchor V_i, R'_i can be calculated easily by the Euclidian distance:

$$Ri' = \sqrt{((x' - x_{Vi})^2 + (y' - y_{Vi})^2 + (z' - z_{Vi})^2}$$

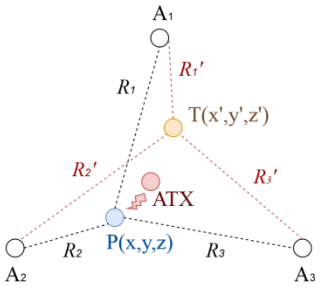


Fig. 30. Attack Scenario against the multilateration layer

Then, ATX has to apply the distance shift $\Delta d_i = (R'_i - R_i)$ for every ranging of the anchor i. As the scheduling is based on Time-Division Multiplexing, one node ATX is enough to attack all the anchors.

Setup- We implemented ATX on a regular Decawino node like we did for the previous experiments. The attack is monitored from an external laptop that has a serial connection with ATX. The distance set (R_1',…, R_n') to the target position is calculated by the external laptop and the distance shifts (Δd_1,…, Δd_n) are sent through the serial port to ATX. The trajectory is controlled in real-time on the laptop by the attacker. Note that the laptop is mostly used to provide a control interface to the attacker so the trajectory can be controlled in real-time. The calculations involved in (4) have a low-computational cost and can be handled by ATX if the trajectory is pre-programmed; hence, an external monitoring device is not necessarily needed.

We show one example of the results of this attack in **Fig. 31**. In this scenario, the platform was a rectangle of length 4.8 m and width 1.8 m; 4 anchors were placed in the corners of the rectangle. The victim node was static all the way through and placed in the center of the platform. The 2D trajectory that was targeted by the attacker is shown in green. The trajectory obtained is shown in blue. The localization frequency was set to 20 Hz, and the localization algorithm used was particle filter with backtracking[1] (section III.3.4). The trajectory has been collected on a total period of two minutes, with a ranging frequency of 20 Hz, leading to a total of 2400 distance samples. We evaluated the average accuracy of the attack, which is defined here as the Euclidean distance between the position obtained and the position wanted. We obtain an average accuracy of 47 cm, with a minimum of 12 cm and a maximum of 67 cm.

[1] The ranging data have been replayed with other multilateration algorithms, which did not show significant differences on the overall results obtained by the attacker.

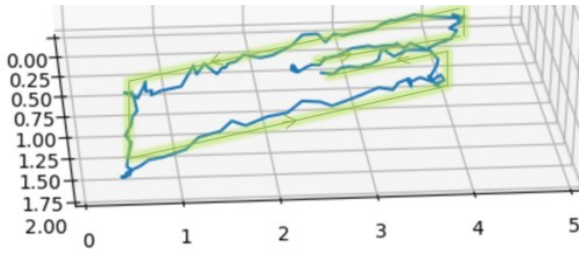


Fig. 31. Trajectory targeted by the attacker *(green)* and trajectory obtained *(blue);* coordinates in m

These results demonstrate that this attack is accurate enough to produce a controlled and coherent trajectory. We also estimated on the same dataset the average values obtained for each detection factor. Regarding consistency and redundancy, the standard deviation of the distance estimates was 19.4 cm, and the average ranging deviation, 31.2 cm. Based on the benchmarks presented in section III.4.2, these values are actually realistic as they are largely below the typical values in harsh environments (respectively 39.2 cm and 44.7 cm). Regarding plausibility, the speed estimated remained below 2.3 m/s throughout the experiments, which is a reasonable speed in most applications. Hence, this attack is able to bypass the system-level detection factors; as a consequence, the acknowledgment vulnerability should absolutely be addressed when scheduled acknowledgments are used.

V.2.9 Conclusion

The transmission scheduling features available on the DWM1000 bring powerful facilities when it comes to tightly controlled reply times, yet are particularly exposed against spoofed acknowledgment attacks. We have shown that a single malicious node can control the distances obtained by the anchors when this feature is enabled, and can extend the attack to the multilateration layer such as taking control over the position of a victim tag. When the acknowledgment is not scheduled, the natural variability of the computations involved for the reply are largely enough on a Decawino node to make the tampered distances very erratic. In that case, we proposed two enhanced scheme that can let the attacker filter out the unrealistic or unwanted distances. The first one is based on a complementary jamming process and the second one on a relay. Although these attacks are efficient at the ranging-level, they are not performant enough to bypass the detection factors discussed in the previous section. Nevertheless, SAA remain a serious threat at the system-level when scheduled communications are involved.

The most intuitive and straight-forward solution to the acknowledgment vulnerabilities in 802.15.4a is either to replace them with another type of frame supporting integrity or replay protection. As far as this solution is indeed effective, it prevents from automating the acknowledgment (which is a feature available on DWM1000 chips), and adds an unneglectable cost due to the cryptographic operations and longer frame length. Based on this consideration, we propose in the next section cryptography-less countermeasures that are efficient against the acknowledgment vulnerability.

Chapter VI Countermeasures for IR-UWB Indoor Positioning Systems

We have shown in that internal attacks are easy to mount against IR-UWB positioning systems, yet difficult to detect. Also, the ranging protocols proposed in IEEE 802.15.4 are exposed to tampering attacks due to absence of authentication on acknowledgments. Replacing acknowledgments by other types of frames supporting authentication is not the most ideal solution, regarding the additional cost involved and the loss of the benefits of automatic acknowledgment functionalities. As a consequence, we introduce in this chapter an acknowledgment strategy that can secure LTWR protocols or TWR with scheduled acknowledgments against spoofing attacks. Then, we propose system-level countermeasures against internal attacks that do not involve specific hardware or modifications of the ranging protocols proposed in the standard. Finally, we expose several novel results at the physical-level which can address the concerns identified in Chapter IV regarding node authentication and key establishment in IR-UWB 802.15.4a networks.

Regarding the proposed acknowledgment strategy, we present a novel scheme for scheduled TWR and LTWR integrity protection that can prevent from spoofed acknowledgment attacks without any cryptographic operation on the acknowledgment frames. This scheme is based on reinforcing the reply time unpredictability with reply time randomization. We notably propose an approach that allows randomizing the reply time without having to share any prior sharing any secret between the prover and the verifier. We demonstrate that this approach increases the robustness against overlapping and allows detecting the attack in a short amount of time, while being extremely low-cost.

Regarding system-level countermeasures, we propose a position fraud detection approach based on differential ranging, which can be mounted with two anchors or more and that is efficient against internal attacks. Similar approaches have been proposed in the literature, but never for TWR and often without practical evaluation. We propose a scheme that requires only passive listening from one or multiple anchors, and can be mounted on top of standard TWR protocols. Then, we propose a cooperative verification scheme against internal attacks, which is based on unpredictable selection of tags as verifiers. By masking the identity and position of at least one verifier in the localization process, this approach introduces inconsistencies in the distance claims of internal attackers and allows detecting position frauds.

Regarding physical-level countermeasures, we first present a novel authentication protocol for IR-UWB nodes, which is based on clock skew. We demonstrate that the crystal oscillators used as reference clocks on DWM1000 chips have unique skew properties, dependent on the temperature, which can be exploited in a weak Physical Unclonable Function (PUF) manner. We introduce the basic concepts of PUFs and physical authentication protocol, and propose an original authentication protocol, which can

characterize the skew/temperature unique curve of a node. Then, we show that the proposed protocol is also a good candidate for Channel-based Key Extraction (CBKE) approaches, which allow two nodes extracting a secret from their mutual radio channel even in the presence of eavesdroppers.

All the proposed approaches have been implemented and evaluated on the platform, with both real and simulated attacks. They are characterized on their discrimination performances, time to alarm, and the smallest magnitude of position violation they can detect.

Chapter contents

VI.1 Countermeasures against the acknowledgment vulnerability

VI.1.1 A randomized reply time-based approach

As demonstrated in section V.2.2, the major challenge to overcome for an attacker when it comes to any type of forged acknowledgment attack is the prediction problem. As the transmission time of the victim is only known accurately when the SFD of its acknowledgment is received, the attacker must start emitting the forged acknowledgment before actually learning the transmission timestamp. As a consequence, the transmission timestamp of the victim acknowledgment must be predicted by the attacker, and the consistency of the attack depends on the accuracy of the prediction.

As a brief reminder, there are three types of configuration to consider in the case of acknowledgment attacks against TWR protocols:

- **Unscheduled TWR**, which is the typical TWR protocol without any scheduling.
- **Scheduled TWR (STWR),** in which similarly to LTWR, the acknowledgment is scheduled with a reply time Δt_{23}, but a DATA frame is still used.
- **Lightweight TWR (LTWR),** in which the acknowledgment is scheduled with a pre-agreed reply time Δt_{23}. In that case there is no DATA frame, and the START frame is the only frame containing a payload in the protocol. This scheme can only work with scheduled acknowledgments.

The two major challenges for the attacker are the overlapping process and the reply time prediction problem. The main conclusions regarding these two challenges were the following:

Overlap: If the two acknowledgments overlap before the preamble detection by the prover, a reception power advantage of 4 dB is enough to get an overlap success rate close to 100%. This advantage can be easily obtained on a cheap UWB device such as a Decawino node. On the other hand, if the preamble detection of the victim acknowledgment has already occurred, an advantage superior to 10 dB on every anchor is required, which requires often more advanced hardware. Also, as the victim preamble has been already detected, the verifier will be aware of the overlap.

Reply time prediction: The natural inconsistencies of the reply time are enough, even on a Decawino node optimized for fast and consistent reply time, to largely throw off the attacker's prediction. However, in the case of scheduled transmissions, the prediction problem becomes trivial as the reply time is known and tightly controlled, and tampering attacks can get an accurate control over the position which can bypass the system detection factors. Hence, scheduled TWR and LTWR are very vulnerable to spoofed acknowledgment attacks.

Considering that, we proposed a countermeasure to the scheduled TWR/LTWR vulnerability, which is based on increasing the prediction error of the attacker by randomizing the reply time Δt_{23}.

VI.1.2 Distance tampering detection

Instead of using always a constant value for the reply time, the prover can use a Δt_{23} picked sampled randomly in a given interval at each ranging. When doing so, the attacker faces the same reply time indeterminism problem as for unscheduled TWR, and loses the ability to produce consistent outputs. As a consequence, the fraud will be easily detectable considering that the standard deviation of the ranging output will be considerably higher than usual.

Considering that, there are three main assets the randomizing process should fulfill:

1. The verifier must be able to retrieve the random reply time used by the prover, otherwise, it will not be able to compute the distance properly, and the benefits of using scheduled transmissions will be lost.
2. A node other than the verifier should not be able to retrieve the reply time; also, the variations of the random function should be high enough to generate detectable inconsistencies on the tampered distance. That means that the variations of the tampered distance induced by the randomization process should be high enough to be detected by the consistency detection factor.
3. The reply time should remain relatively steady even with the randomization process, i.e., the standard deviation of the reply time distribution should be much lower than its mean. If the reply time varies drastically from one ranging to another, this security scheme can indeed create conflicts with TDMA scheduling.

First approach— Ciphered random reply time: This approach is the most intuitive and straight-forward. Since the START frame supports both encryption and authentication, it can be used to exchange a secret between the prover and verifier in a ranging process. In this approach, the verifier randomly samples Δt_{23} before each ranging protocol and sends the encrypted value to the prover in the start frame. Δt_{23} is sampled from a uniform distribution $U([\mu - S_{max}, \mu + S_{max}])$ where μ is the average reply time required for the application, and $\mathbf{S_{max}}$ is the maximum time shift allowed from the center of the distribution. **Fig. 32** shows this principle applied for LTWR protocols.

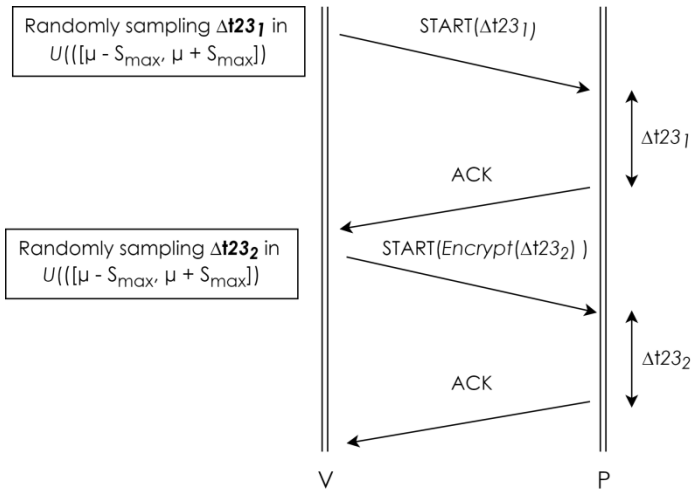


Fig. 32. Ciphered reply time strategy for LTWR

The equivalent distance of S_{max} should be chosen high enough so that the variability of the randomization process can be discriminated from the natural inconsistency of distance measurements. Given that the distribution is uniform, the standard deviation of a $\Delta t23$ is equal to $S_{max}/2\sqrt{12}$ [108]. Hence, a value in the order of magnitude of 10 ns for S_{max} is typically enough, as 10 ns corresponds to a standard deviation of about 1.45 m in a TWR protocol, which is significantly higher than the standard deviation observed in non-adversarial conditions.

The main advantage of this approach is that the verifier is aware of the exact sampled reply time value as it samples $\Delta t23$ itself. The main drawback is that the start frame should be ciphered for this scheme to be secure; otherwise, the attacker can easily learn the secret reply time. Ciphering the start frame is not necessary in a regular ranging, for which integrity protection is sufficient, as the start frame does not carry any secret payload. If the ranging process is scheduled TWR and not LTWR, an alternative is to inform the prover of the reply time for the next ranging in the ciphered DATA frame.

Discussion— This approach is not completely costless. For applications that did not need to protect the privacy of start or data frames in the first place, encrypting one of those two frames will be required to mount this countermeasure, implying an additional cryptographic cost. For these applications, this approach might not be the most suitable considering this additional cryptographic cost. Besides that, there is also a payload extension of 8 bytes[1] for the reply time value on one of those two frames for each ranging.

In the following, we demonstrate that this scheme allows detecting acknowledgment spoofing attack, and show that it also increases the minimum reception power advantage required for successful overlaps. Then, we propose an enhanced scheme that does not require sharing the chosen reply time beforehand.

[1] Timestamps are typically encoded on 64 bits on UWB devices [63]

VI.1.3 Increasing the robustness against overlapping

Detecting the attack does not prevent the attacker to use spoofed acknowledgments as fine DoS attacks (section IV.2.4), as it will generate gibberish distances and prevent the verifier from obtaining the real distance. In a classic spoofed acknowledgment against scheduled TWR, when the reply time is constant and not randomized, there are two possible attack outcomes, which we discussed in section V.2.2:

- **Distance reduction,** when the attacker emits the forged acknowledgment before the legit one. In that case, as long as the attacker is transmitting before the victim and gets a higher reception power on the verifier side, he or she will trigger an overlap with a high success rate.
- **Distance enlargement,** when the attacker emits the forged acknowledgment after the legit one. In that later case, if the forged acknowledgment arrives too late the verifier might have enough time to detect the victim preamble, which two consequences:
    1. The verifier will be aware of the collision; collisions should not occur recurrently considering the established TMDA scheme
    2. Overlapping after the victim's preamble detection is much harder and requires from the attacker a much higher reception power advantage (~10 dB)

We have demonstrated that the reception power advantage required for overlapping can be obtained on cheap UWB devices if the overlap starts before the preamble detection. However, if the preamble detection has already occurred, the attacker will need a more advanced and costly hardware setup to achieve an overlap success rate close to 100%, and the attack will be eventually revealed by the repetitive collisions. We define as $T_{detection}$ the time required by the verifier for preamble detection.

Attacker's game— The goals of the attacker are defined as following. The attacker makes a prediction t3′ about the starting time of legit acknowledgment, t3. The objectives of the attacker's game are twofold. First, t3′ should verify $t3' < t3 + T_{detection}$, which grants the attacker an easy and stealth overlap. If that condition is not verified, the legit acknowledgment's preamble will be detected and the game is lost. Otherwise, if that first objective is complete, the attacker's must maximize his or her gain, defined as $|t3' - t3|$. Indeed, this gain defines the accuracy of the attacker on the distance obtained.

Hence, when the reply time is randomized, there is a certain probability that the error on t3 prediction of the attacker is superior to the preamble detection time $T_{detection}$. We illustrate that in **Fig. 33**. We notate $\mathbf{P}_{overlap}$ the probability for the attacker to achieve an overlap with **little effort** (i.e., with cheap hardware and a small reception power advantage) and **without being detected**. This probability is shown in red in **Fig. 33**. The probability that the victim's acknowledgment starts as a given time is shown in yellow *(hatch lines)*; considering that it is drawn from a uniform distribution $\mathcal{U}([\mu - S_{max}, \mu + S_{max}])$, this probability is constant over the time interval defined for the reply time. When the victim's acknowledgment starts, the attacker has a timespan $T_{detection}$ to start the rogue acknowledgment. Past that delay, the attacker loses the game. The earlier the prediction is, the higher are the chances of the

attacker to overlap the legit acknowledgment before preamble detection. In that regard, the attacker should emit no later than **(t2 + μ - S_{max} + $T_{detection}$).** However, assuming that S_{max} >> $T_{detection}$, that strategy diverges from the attacker's second objective, which is to be accurate on the t3 prediction to maximize the gain. Indeed, considering the uniformity of the distribution the attacker should rather target the center of the distribution, **(t2 + μ).**

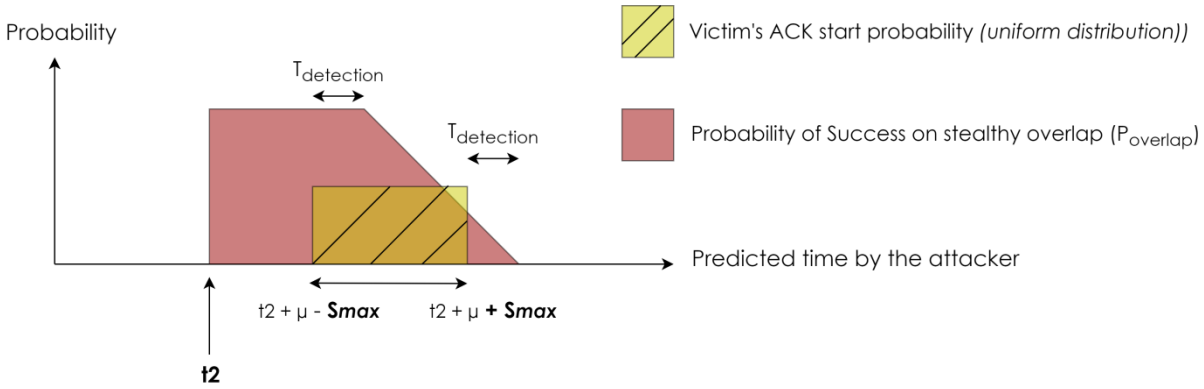


Fig. 33. Probability of success of the attacker over its time prediction

Considering that divergence, we propose in the following two different schemes to increase the TWR robustness against acknowledgment overlap.

1. Ciphered random reply time with increased standard deviation— This is the same approach as the one introduced in section VI.1.2, but this time with a much larger sampling interval for the normal distribution used in the randomizing process. Given that preamble detection takes typically a few microseconds, a value in the order of magnitude of 100 μs for the maximum time shift allowed S_{max} is typically enough to reduce drastically the probability for the attacker to hit the overlap ideal timespan.

2. Reply time with random modulo: This approach aims to randomize the reply time without having to share the sampled value between both parties. The idea is the following: the prover and verifier agree on a modulo value T_{mod} which verifies $D_{mod} = tof(T_{mod}) > MRD$, with D_{mod} the distance travelled in T_{mod} at the speed of light and MRD the Maximum Realistic Distance. The value of MRD, as discussed in section V.2.2, can be either the maximum range of UWB (which hardly exceeds 100 m in the most optimistic case, especially indoor) or the longest possible distance within the walls of the monitored environment if it is lower. Put differently, the modulus value chosen for the time-of-flight should correspond to an equivalent distance that is not physically obtainable in a UWB ranging.

From that, this approach works as following. The prover samples an integer value k from a uniform discrete distribution, $U([-N_{max}, +N_{max}])$. We discuss the choice of N_{max} later in the simulation results. The prover computes the reply time Δt_{23} as:

$$\Delta t_{23} = \mu + k * T_{mod}$$

With μ the typical reply time that is suitable for the application, similarly to the previous approach.
When the ranging protocol is complete, the verifier V computes the time-of-flight based on the assumption that $\Delta t_{23} = \mu$, and obtains the distance $d_{shifted}$, the shift depending on the random reply time picked by the prover. Given that the distance is positive and cannot exceed D_{mod}, there is only one possible distance solution *d* that verifies:

$$0 < d < D_{mod} \; and \; d \equiv d_{shifted}[\mathrm{D_{mod}}]$$

This unique solution is the reminder of the Euclidean division of $d_{shifted}$ by T_{mod}. Hence, the verifier retrieves the real distance d and the random value sampled by P, k, by simply computing this Euclidean division as:

$$d_{shifted} = D_{mod} * k + d, k \in Z$$

Here is an example to illustrate the protocol. Alice performs a TWR protocol with Bob. Alice is 10 m away from Bob, and D_{mod} is defined as 100 m. After receiving the start frame, Bob picks randomly k = 4. So far, the reply time of Bob is unknown to Alice. After completion, Alice obtains a distance $d'_{shifted}$ of 410 m. Alice retrieves d and k by computing the Euclidean division of $d'_{shifted}$ by D_{mod} as $d' = k * D_{mod} + d$, which gives here $410 = 4 * 100 + 10$, allowing Alice to obtain d = 10 m and k = 4.

Retrieving the value of k picked by the prover is necessary for LTWR to guarantee the security of the proposed scheme, as we explain in the following. Considering that the prover never sends its timestamps in LTWR protocols, the verifier cannot verify that the value extracted for k is indeed the one that has been picked by the prover. Thus, an attacker could simply choose an arbitrary value for k, apply the time shift $k * T_{mod}$ on its own reply time, and tamper successfully the distance. However, to remain undetected by the consistency factor (section III.4.1), he or she needs to overlap successfully all the TWR protocols. Considering the analysis in **Fig. 33**, if the attacker respects the protocol and samples k in the defined random uniform distribution $U([-N_{max}, +N_{max}])$, the probability to trigger overlap successfully will not be maximized, and he or she will be revealed by the consistency detection factor. On the other hand, since the attacker can choose any value of k, he or she can simply cheat and pick the minimum reply time $\sigma - N_{max} * T_{mod}$: in that case, the forged acknowledgment will always precede the legit acknowledgment and maximize $P_{overlap}$.

To counter that, the verifier can simply monitor k over time and verify that k is indeed drawn from the defined uniform distribution. If the attacker tries to cheat in that manner, the values chosen for k will always be negative and definitely not correspond to a random variable. The verifier can simply monitor the mean and standard deviation of k, which should respectively be close enough to 0 and $2(N_{max} * T_{mod})/\sqrt{12}$. We evaluate that more precisely in the results with confidence intervals. By doing so, the attacker cannot bypass the consistency detection factor, which makes this security scheme secure even for LTWR.

Regarding the choice of T_{mod}: an additional security asset can be brought by using $T_{mod} = T_{sym}$, where T_{sym} is the exact duration of a preamble symbol. In a TWR protocol, T_{sym} corresponds to an equivalent distance shift of roughly 150 m, which is high enough to fulfill the condition $T_{mod} >$ MRD (~100 m). As mentioned before, preamble detection does not allow the receiver to learn the exact number of symbols that have been already sent [11]. As a consequence, the attacker cannot infer the value of t3 from the preamble detection timestamp $t_{preamble}$, but only t3 modulo T_{sym}. If $T_{mod} = T_{sym}$, that property is maintained. However, if they are different, there is only one solution t3 that satisfies:

$$t3 \equiv t_{preamble}[T_{mod}] \; and \; t3 \equiv t_{preamble}[T_{sym}]$$

As a consequence, it is more secure to use $T_{mod} = T_{sym}$, as it will prevent the attacker from inferring t3 by timestamping the preamble symbols.

VI.1.4 Performances analysis with Monte-Carlo simulations

We estimated with Monte-Carlo simulations the performances of the modulo-based approach for different values of S_{max}. We evaluated these performances in two scenarios.

In the first scenario, it is assumed that the attacker applies the same randomization process than the victim (as defined in the previous section) on its own reply time. As a consequence, the random properties of its reply time will look legit for the verifier, but a significant part of the legit acknowledgment preambles will be detected by the verifier due to the randomness.

We show the success rate in **Fig. 34**. We neglect the impact of the distance shift chosen by the attacker has it is extremely small compared to the timescale involved. Even if the reply times of both the attacker and victim are randomized the same way, the attacker has an extra advantage due to the preamble detection delay, as illustrated previously in **Fig. 33**. As a consequence, when S_{max} is relatively low the success probability of the attacker is higher; however, when S_{max} is increased this advantage gets progressively negligible and ultimately the attacker's success is almost equivalent to a coin flip, as the attacker has basically one chance out of two to emit before the victim.

The attacker's success rate is 100% for the lowest values of N_{max}; indeed, the amplitude of the randomized component of the reply time is inferior to the average preamble detection time. After $N_{max} = 4$, the success rate starts quickly decreasing and converge eventually to 50%. The success rate of the attacker can be reduced to 80% (i.e., 20% of untampered distances) with $N_{max} = 10$, which is enough to detect the attack with the consistency detection factor. For reference, as the process if based on a random uniform function, the standard deviation of the reply time is about 5.8 μs for $N_{max} = 10$.

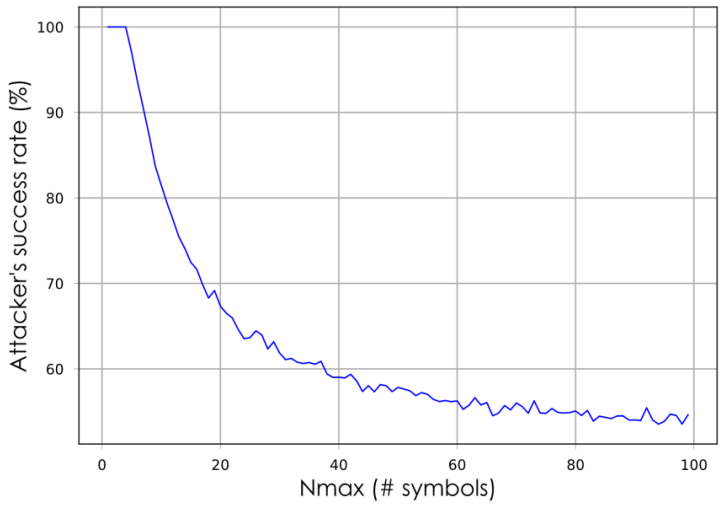


Fig. 34. Success rate of the attacker against modulo-based reply time randomization for different N_{max}; N_{max} defines the length of the interval in which the reply times are sampled

In the second scenario, it is assumed that the attacker prioritizes the success rate over bypassing the randomness verification. We assume favorable conditions for the attacker, as we consider that a success rate of 80% is enough to remain undetected. The randomness of the reply time is controlled by the verifier based on the mean and standard deviation. The verifier records the symbol shift chosen by the prover and computes continuously the mean and standard deviation of the accumulated values. The attack detection is based on 99% confidence intervals, i.e., the verifier reports an attack if either the observed mean or standard deviation do not lie into the 99% confidence intervals of the defined random uniform function. If they do, that means that the chances that the values were indeed drawn from the defined uniform distribution are high (i.e., the reply time distribution is not tampered). The higher the number of accumulated samples is, the lower are the tolerated differences between the observed mean and standard definitions and the theoretical ones. With μ and σ the mean and standard deviation of the symbol shift, n the number of accumulated samples, and z= 2.58 the z-score for a 99% confidence-level, the respective confidence intervals of the mean and standard deviation, I_μ and I_σ are given by [109]:

$$I_\mu = [-\sigma_{max}, +\sigma_{max}] \; with \; \sigma_{max} = z * \sigma/\sqrt{n}$$

$$I_\sigma = [-2 * \frac{\sigma_{max}}{\sqrt{12}}, +2 * \frac{\sigma_{max}}{\sqrt{12}}]$$

We evaluate in the following the average time to alarm, i.e. the average number of TWR protocols required for attack detection. The verifier computes the 99% confidence intervals after each sample and detects an attack if either the observed mean or standard deviation does not lie in these intervals. Then, we monitored the average time to alarm required by the verifier to detect the fraudulent reply time distribution used by the spoofer.

The results are displayed in **Fig. 35**, and have been obtained with 10000 Monte-Carlo simulations for each value of N_{max}.

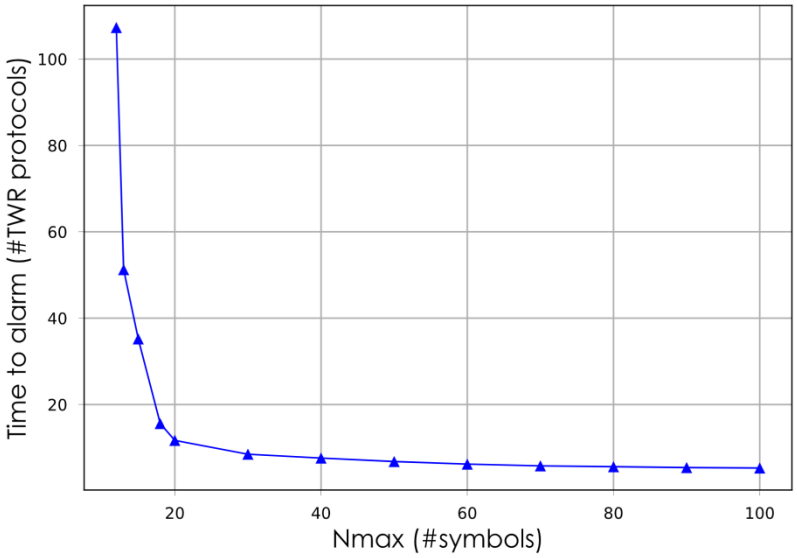


Fig. 35. Average time to alarm for different value of N_{max}

When N_{max} is too low, we have seen previously in **Fig. 34** that the attacker already benefited from an 80% success rate without even having to cheat. This leads obviously to tremendous times to alarm as the fraud can barely be detected. However, the time to alarm decreases rapidly between values N_{max}= 12 and N_{max} = 20, as it goes from 107.3 to 8.5 average TWR protocols. The impact of increasing N_{max} is less notorious beyond that. The corresponding time to alarm in time units obviously depends on the ranging frequency; for a ranging frequency of 10 Hz, that is equivalent to 0.85 s on average.

VI.1.5 Conclusion

We proposed two randomized reply time schemes for acknowledgment. The first one, based on pre-agreed reply times, is functional even with a minimized variability on the reply time (in the order of a nanosecond), but requires at least one ciphered frame in the protocol and does not do anything regarding the robustness of the legit acknowledgment against overlap tentatives. The second scheme proposed does not require the two parties to share the chosen reply time beforehand; instead, the random part of the reply time is defined as a multiple of a time-of-flight that is high enough to be physically impossible, T_{mod}. This allows the verifier to easily retrieve the real distance by applying a simple Euclidean division. This second scheme also increases the robustness against overlaps; indeed, the attacker will often not be able to overlap before the preamble detection due to the reply time unpredictability. A reception power advantage of 4 dB was enough for the attacker to get a high success rate on unsecure acknowledgments, while at least 10 dB are required with the proposed scheme. The

main asset of this approach is that even if the attacker has a strong reception power advantage the attack will be still detected; indeed, the randomness of the reply time allows the verifier detecting some of the discarded preambles of the victim node. We showed that the attacker cannot get a sufficient success probability to bypass the consistency detection factor if he or she complies to the proposed randomization scheme. If he does not, we demonstrated that with $N_{max} = 20$ (i.e., the sampled reply time can deviate from up to 20 times T_{mod} from the average reply time), the verifier can already detect tampered randomization process after 8.5 ranging protocols on average. $N_{max} = 20$ corresponds to a standard deviation of the reply time of 11.6 μs; hence, the reply time remains relatively steady. As a consequence, this is a great low-cost approach for spoofed acknowledgment attack detection, which does not require authentication primitives on the acknowledgments and can be mounted with little computational effort.

VI.2 System-level Countermeasures

The section introduces the countermeasures proposed at the system-level. These countermeasures have the higher level of abstraction, which means they are relatively independent from the MAC and physical layers implementations and can be extended to other technologies than UWB.

We introduce two main contributions in this section, which are *differential TWR* and *cooperative approaches for malicious nodes detection*. In a TWR protocol, differential TWR allows one or multiple external listeners (typically, other anchors) to verify the authenticity of the distance claimed by a prover. Despite being derived from a MAC-level protocol, differential TWR is based on the anchors' common knowledge of the system's configuration, and does not require any active participation (i.e., transmission of dedicated frames) at the link layer. Considering that, and given that differential TWR can only be used in a multi-anchors system, they are classified here as system-level countermeasures. On the other hand, cooperative verification introduces mutual distance verification between tags: as contrary to the anchors, tag positions are varying over time and are not necessarily publicly known, a cheater is more likely to generate an incoherent distance when performing a ranging with an unknown mobile tag.

We discuss the threat model, principles, use case, performances and limitations of these two methods in the following.

VI.2.1 Differential TWR

VI.2.1.1 Principle

Differential ranging has been already introduced as part of the ranging protocols state-of-the-art in section II.1.2. In a differential ranging protocol, a node does not estimate an absolute distance but a difference between two distances instead. For the sake of clarity, we will refer to the ranging protocols that are not differential (i.e., most conventional ranging protocols) as *direct ranging.* For three nodes A, B and C, direct ranging schemes allow A to

estimate the distances d(AB) and d(AC); a differential ranging scheme allows only A to learn d(AB) – d(BC), d(AC) – d(BC) and d(AB) – d(AC). As discussed previously, differential ranging schemes are more lightweight and less accurate than direct ones. In the case of Crazyflie Loco Positioning System [62], which features a differential ranging protocol, they are mostly intended for high nodes density where direct ranging schemes cannot reach sufficient position refresh rates due to the high number of nodes.

In this section, we propose to use the differential ranging approach as a security mechanism. The main strength of the differential approach is that it involves two or more verifiers using the same frame as reference. Hence, if the prover tries to cheat, it will not be able to apply a custom shift for each verifier as they are both timestamping the same frame. Using differential ranging has already been proposed in [110, 111]; however, the proposed protocols were not based on standard TWR and were only evaluated through simulation.

Our approach is TWR-based and has been implemented and evaluated on SecureLoc. We propose to use differential ranging not as substitute but as a complement of direct TWR. We demonstrate in the following that an external node that has some knowledge of the system can perform a differential ranging by simply listening to a TWR protocol in which it was not primarily involved.

We assume a similar configuration as the one described for the internal attack, which is described below in **Fig. 36**. The tag T is at the position P; the distances that should obtained by the anchors without any attack, are notated R_i for every anchor A_i.

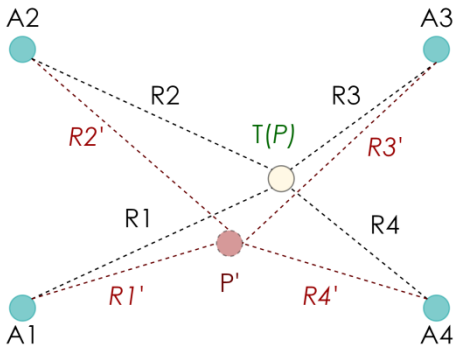


Fig. 36. Platform configuration for differential TWR

Assuming that an anchor A_i performs a TWR protocol with the tag T, an anchor A_j can verify the integrity of the distance measured by A_i by simply listening to the TWR protocol. We introduce a listener A_j in the conventional TWR protocol between A_i and T, as shown below in **Fig. 37**.

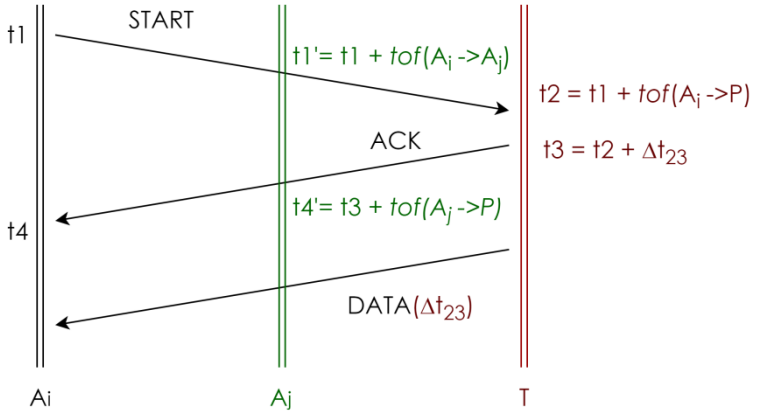


Fig. 37. Differential TWR protocol

Note that the following discussion applies equally to TWR and LTWR; the DATA frame shown in **Fig. 37** is optional. If TWR is used, Δt23 is extracted from the timestamps returned in the DATA frame: if LTWR is used, Δt23 is simply the pre-agreed reply time. Whether LTWR or TWR is used has no impact in the overall principle of differential TWR, and we focus in the following on classic TWR without loss of generalities.

As the positions of the anchors are known, the mutual time-of-flight between of any pair of anchor (A_i, A_j) is known and static. Although A_j is simply listening to the ranging protocol of A_i in **Fig. 37**, A_j is a regular anchor and has also dedicated slots for direct TWR with the tag T. Thus, it can be assumed that A_j has performed a direct ranging protocol with T recently, hence has an up-to-date estimation of the distance $d(A_i \rightarrow P)$. A_j receives START at t1' and ACK at t4'. Thus, we have the following relation:

$$\Delta t_{14'} = t4' - t1' = \left[t1 + tof(A_i \rightarrow P) + \Delta t_{23} + tof\left(A_j \rightarrow P\right)\right] - \left[t1 + tof\left(A_j \rightarrow A_i\right)\right]$$

Giving:

Eq. 8

$$\Delta t_{14'} - \Delta t_{23} = tof(A_i \rightarrow P) + tof\left(A_j \rightarrow P\right) - tof(A_j \rightarrow Ai)$$

As highlighted above, $tof(A_j \rightarrow P)$ and $tof(A_j \rightarrow A_i)$ are known. We refer to $(\Delta t_{14'} - \Delta t_{23})$ as the *differential time-of-flight*. We notate the corresponding distance d_{diff}, with $d_{diff} = c * (\Delta t_{24'} - \Delta t_{23})$.

Hence, from Eq. 8, A_j can extract $tof(A_i \rightarrow P)$ and obtains a differential estimation of the distance between A_i and P, $\tilde{d}(A_i \rightarrow P)$, defined as:

$$\tilde{d}(A_i \rightarrow P) = d_{diff} + d\left(A_j \rightarrow Ai\right) - \hat{d}\left(A_j \rightarrow P\right)$$

With $\hat{d}\left(A_j \rightarrow P\right)$ the most last distance estimate obtained by A_j in its most recent **direct** ranging protocol with P.

Then, A_i and A_j can proceed to compare their obtained distances. If P participated honestly during the protocol, A_i and A_j should theoretically obtain the same distance. In practice, a small difference is tolerated because of the measurement noise. Assuming that the time-of-flight between the two anchors is exactly known, the average error on the differential measurement is twice more than the direct measurement as it is a difference between two distance estimates. If P is cheating, we can demonstrate that this approach allows detecting the attack at the system-level.

Proof

If P is dishonest (i.e., P is mounting an internal attack), P aims to apply customized distance shifts on the distance estimates of each anchor, as studied previously in section V.1. In the following, we assume that a direct ranging is occurring between P and an anchor A_i, with another anchor A_j listening to the protocol and performing differential TWR. P must apply the distance shifts Δd_i and Δd_j on respectively A_i and A_j to reach its dream position. Assuming that P did already apply the distance shift Δd_j during its last direct TWR with A_j, leading Aj to obtain the tampered distance $\hat{d}(A_j \rightarrow P) = d(A_j \rightarrow P) + \Delta d_j$, P must now apply a distance shift Δd_i during the TWR with A_i. Instead of replying reply after Δt_{23}, P cheats and applies Δd_i by sending fraudulent timestamps in the DATA frame. Note that Δd_i or Δd_j can be equal to zero.

After completion of the TWR protocol between A_i and P, A_i obtains $\hat{d}(A_i \rightarrow P)$ and A_j $\tilde{d}(A_i \rightarrow P)$. Since P has applied a distance shift during its TWR of A_i, A_j is also affected as a listener and obtains $\hat{d}_{diff} = d_{diff} + \Delta d_i$ based on Eq. 8. With $d(A_i \rightarrow P)$ the real distance from A_i to P, $\hat{d}(A_i \rightarrow P)$ the direct distance estimate to P obtained by A_i, and $\hat{d}(A_j \rightarrow P)$ the most recent direct distance estimate of A_j, we have:

Eq. 9

$$\tilde{\boldsymbol{d}}(\mathbf{A_j} \rightarrow \mathbf{P}) = \hat{d}_{diff} + \Delta d_i + d(A_j \rightarrow Ai) - \hat{d}(A_j \rightarrow P)$$

$$= d_{diff} + \Delta d_i + d(A_j \rightarrow Ai) - (d(A_j \rightarrow P) + \Delta d_j)$$

$$\tilde{\boldsymbol{d}}(\mathbf{A_i} \rightarrow \mathbf{P}) = d(A_i \rightarrow P) + (\Delta d_i - \Delta d_j)$$

If P was honest, A_i and A_j should have obtained the same distance. When A_i and A_j compare their respective results, A_i has obtained the distance $\hat{d}(A_i \rightarrow P) = d(A_i \rightarrow P) + \Delta d_i$ and A_j has obtained $\tilde{d}(A_i \rightarrow P) = d(A_i \rightarrow P) + (\Delta d_i - \Delta d_j)$. For noiseless measurements, the two estimates are equal only if only if $\Delta d_j = 0$. If A_j and A_i are verifying mutually with differential TWR their respective distance estimates, it becomes impossible for the tag to apply a distance shift without being revealed. The only way to bypass differential TWR would be to attack only one anchor per ranging cycle, which implies that at least half of the distances would be untampered; such a scheme is defeated by the consistency detection factor.

If we consider noise, the condition $\Delta d_i = 0$ does not hold true anymore as small discrepancies between the differential and the direct distance estimates might be simply induced by the measurement noise. In that case, based on Eq. 9, the attacker needs to keep the distance shift in the same order of magnitude as the typical ranging error to bypass differential TWR. As a consequence, if the distance shift is small enough the attacker could potentially remain undetected. We evaluate in the following the maximum position shift that an attacker can aim for based on the performances of SecureLoc.

VI.2.1.2 **Evaluation**

Similarly to the accuracy benchmarks displayed in section III.3.2, we evaluated the typical differential ranging error in three types of non-adversarial environment: clear, harsh with LoS, harsh without LoS. Then, we compared these results to the differential ranging error in an adversarial environment, in which the monitored tag was mounting an internal attack. We repeated the evaluation for different position shift (i.e., the Euclidean distance between the real position and the claimed position of the rogue tag). In the experiments, every anchor was performing differential TWR during the direct TWR of the other anchors. Thus, we formalize the definition of differential ranging error as follows:

Definition 4: for N anchors, with $\widetilde{d_k}(A_i \to T)$ the differential distance measured by an anchor A_k during the direct ranging between an anchor A_i and a tag T, and $\hat{d}(A_i \to T)$ the direct distance estimate of A_i during for the same ranging protocol; we define the Average Differential Ranging Error (ADRE) of a given anchor A_i for a tag T as:

$$ADRE_i\,(T) = \frac{1}{N-1} \sum_{\forall k \leq N / k \neq i} (\widetilde{d_k}(A_i \to T) - \hat{d}(A_i \to T)\,)^2$$

We also define the Maximum Differential Ranging Error (MDRE) of an anchor A_i for a tag T as:

$$MDRE_i(T) = Max(\left\{\left(\widetilde{d_k}(A_i \to T) - \hat{d}(A_i \to T)\right)^2, \forall k \leq N \,/\, i \neq k\right\})$$

Based on that, we define the Total Average Differential Ranging Error (TADRE) of a tag T as:

$$TADRE(T) = \frac{1}{N(N-1)} \sum_{i=1}^{N} ADRE_i(T)$$

And finally, the Global Maximum Differential Ranging Error (GMDRE) of a tag T as:

$$GMDRE(T) = Max(\{MDRE_i, \forall i \leq N\,\})$$

Setup—the experiments have been done with 4 anchors, placed in the corners of a 4.8 m by 1.8 m rectangle (similar to the configuration used in Chapter II, shown in **Fig. 16**). 4 mobile tags have been tested, each on 10 mn measurement sessions. The mobile tags were placed on a remote controlled car, which was moving randomly across the platform at a speed of roughly 4 m/s. The configuration (types of obstacles and anchor placement) for the three types of environment was the same as the ones used in section III.3.2. In the first experiments, the tags were participating honestly in the ranging protocols. In the next series, they were mounting an internal attack: a random 2D-vector with a pre-defined norm was randomly sampled by the rogue tag. The position claimed by the rogue tag was its real position shifted by the sampled vector. Hence, the position shift was always equal to the pre-defined position shifting vector norm. The shifting vector was re-sampled every 5 seconds. We repeated the evaluation with multiple position shift values, from 10 cm to 3 m. The attack scenarios have all been tested in a clear environment.

Results—We evaluated the average TADRE et GMDRE on the samples with honest tags first. The results obtained for the three types of environment are shown below in **Table 29**.

Table 29. Average TADRE and GMDRE for different types of environment

	Clear environment	**Harsh LoS Environment**	**Harsh NLoS Environment**
TADRE	23 cm	36 cm	62 cm
GMDRE	29 cm	46 cm	74 cm

These results give an estimation of the typical error induced by measurement noise in a non-adversarial differential TWR. We used these benchmarks as threshold and compared them to the TADRE and GMDRE obtained for internal attacks with increasing position shifts. The results are shown in **Fig. 39** and **Fig. 38**. The red lines represent the average values obtained in non-adversarial settings presented in **Table 29**.

We can observe that for important position shifts, differential TWR is largely able to discriminate tampered distances from normal distances. GMDRE shows better discrimination performances than TADRE. Indeed, for a position shift of 75 cm GMDRE, is already able to differentiate the internal attack from the perturbations induced by a harsh NLoS environment. On the other hand, this is only the case above 1 m for TADRE. Hence, the worst case (i.e., the higher differential ranging error obtained on the whole set of anchor) is a better indicator for internal attack detection that the average case. This can be explained by the fact that the error by the noise is essentially random, whereas the error induced by the attack is the result of a deterministic geometrical process. For instance, if the position shift vector is directed toward an anchor, the attack-induced error will be maximized for this anchor, i.e., equal to the position shift; this is quite unlikely for a random process.

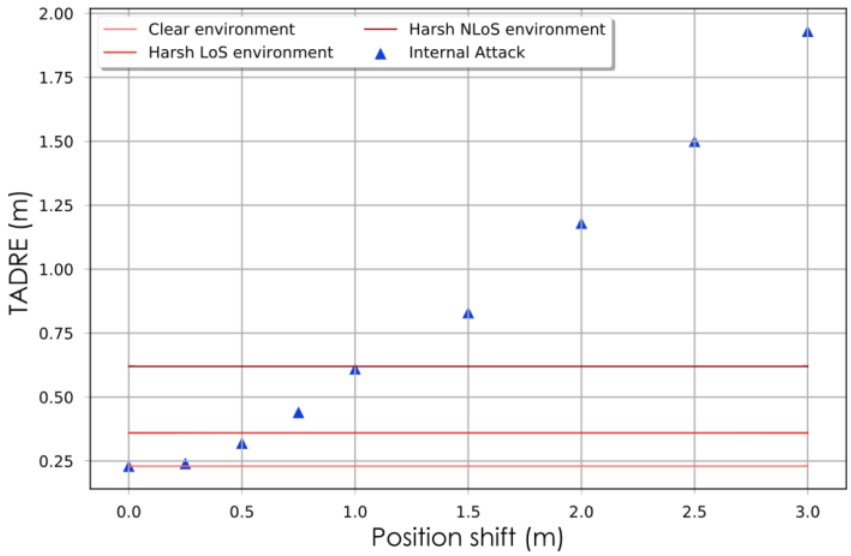


Fig. 38. Average TADRE for different position shifts in an internal attack

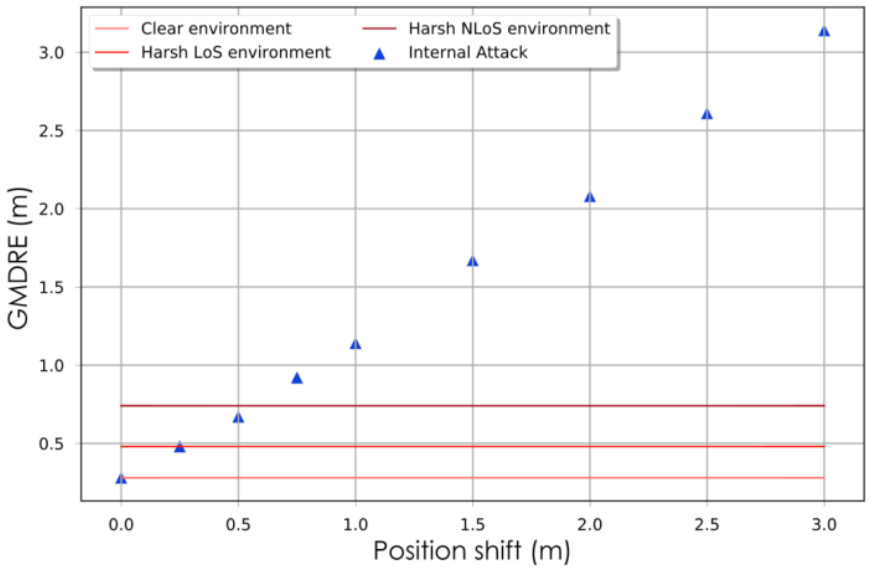


Fig. 39. Average GMDRE for different position shifts in an internal attack

Thus, GMDRE is a good detection factor for differential TWR. An attacker can aim up to 75 cm shift without being detected; this can be reduced further if the complexity of the environment is identified, as it allows setting a finer threshold: for instance, if the environment is clear position shifts above about 25 cm can be detected. In any case, the room for internal attack when differential TWR is enabled is quite limited, which makes differential TWR a strong countermeasure against internal attacks.

Discussion

The major strength of differential TWR is that it relies only on reception and not on transmission; as a consequence, it can be used without applying any change to standard TWR. Also, only anchors are involved into differential TWR, which means that it does not add any constraint on the tags. This is a consequent advantage as tags are typically highly power-constrained. Thus, it is a costless countermeasure for the tags.

Anchors are typically less power-constrained than tags as they are usually not battery-powered. Yet, in a scenario where anchors are actually power-constrained differential TWR has a consequent cost: we already highlighted that reception is the more energy-consuming operation on UWB transceivers. If all the anchors listen to the respective ranging of their peers, the energy consumption of each is almost multiplied by the number of anchors. In that case, the number of differential TWR per cycle can be reduced: we already demonstrated that if each anchor monitors the direct ranging protocols of one single peer, the attack can be detected. Also, considering the consistency detection factor, occasional instead of systematic verifications (e.g., one every five ranging cycles) can help decreasing the cost.

One additional drawback is that differential TWR restrains the use of preamble code-based or frequency-based multiplexing. Although this is not commonly used on 802.45.4 UWB, two anchors could transmit simultaneously on two different channels or with two different preamble codes[1]. In that configuration, differential ranging prevents an anchor from communicating on another media as it has to listen to the ranging protocol of a peer. As a consequence, it can reduce the IPS multiplexing capacities. Nevertheless, it is quite rare for IPS to use non time-based multiplexing, and this will not be a concern for most applications.

Regarding the security of these approaches, it is suggested in [111] that an attacker could use directive antennas to send two beacons slightly delayed to two different listeners. This scheme could theoretically bypass differential ranging, but such an attack would be highly complex and require advanced hardware. First, the beam of the antennas needs to be oriented to the right target anchor and to be narrow enough to prevent any other anchor from receiving it. Second, the tag needs to send the same frame in several directions with fine-controlled delays, which would most likely require multiple UWB transceivers and an advanced synchronization protocol. As a consequence, such attacks have a moderate likelihood from a vulnerability analysis perspective, considering their cost and complexity.

[1] Preambles codes are orthogonal to each other; 2 different preamble codes can be transmitted simultaneously on the same channel without collisions. This does not apply to the PHR and payload; as a consequence, the two frames should be shifted enough to avoid the payloads overlap when using this approach.

VI.2.2 Cooperative approaches for malicious nodes detection

VI.2.2.1 A brief overview of cooperative approaches for security in WSN

The capability of an attacker to successfully bypass the system detection factors derives essentially from its knowledge of the system. In an internal attack, the rogue tag has to learn the position of the anchors to produce coherent distances shifts. As a consequence, hiding the position of verifiers is a good way to induce inconstancies in the tampered distances of a cheating tag, as it will prevent it from calculating the proper distance shift to reach its target position. This applies equally to internal and external attacks: no matter if the concerned tag is tampering the distances itself or is actually attacked by an external device, in both cases the actor behind the attack must be able to produce coherent distance shifts to remain stealth.

Adding hidden mobile stations has been proposed in [111], but there are multiple drawbacks to this solution. First of all, hiding an anchor (i.e., keeping the position of that anchor secret) is quite unpractical or even impossible for most applications. Also, the hidden stations have to be passive; otherwise they could be located from their transmissions. This implies a relatively high cost for this countermeasure: the hidden stations do not bring much value in terms of localization performances (passive localization is necessarily based on differential approaches, which are less accurate) and are mostly dedicated to security. If the hidden anchors are static, this approach is quite flawed as compromising the position of a hidden anchor once can expose the whole system for an extended amount of time. If the hidden anchors are mobile, as proposed in [111], this scheme is even more unpractical and the mobile stations accuracy is decreased even further as the error on their own position adds up to their measurement error.

A cooperative approach to secure positioning called Secure Positioning for sensor Network algorithm (SPINE), has been proposed by Capkun et al. in [86]. In SPINE's model, it is assumed that secure positioning has to be achieved with a very constrained number of anchors, and that all nodes can perform distance bounding. The novelty of SPINE is to involve the mobile nodes into the distance integrity verification protocol.

SPINE is based on verifiable multilateration (VM), which is also introduced in [86]: if three verifiers form a triangle that encapsulates a prover, they can assess the integrity of the position claimed by the verifier with distance-bounding (DB, section II.2.2) protocols, as at least one of the three distance will be enlarged if the prover is cheating. This triangle-based verification process is referred to as Verifiable Multilateration (VM). There are three main steps in SPINE: (1), all the nodes measure their respective distances to their neighbors; (2), the distances bounds are checked with VM; (3), the positions of the nodes are computed. Depending on the number of anchors and the node density, all the distances cannot necessarily be verified. The authors show in [86] that 80 nodes/10000m² is typically enough for unsecure positioning, whereas at least 110 nodes/10000m² are required for SPINE. As SPINE involves DB, which requires a specific transceiver design, it cannot be implemented on regular IR-UWB devices.

There are not many other examples of cooperative verification schemes in the literature of indoor positioning systems. However, there are a lot of works on this topic in the more general literature of WSN. Cooperative operations are inherent to WSN: multi-hop communications, as they involve nodes relaying information, are by nature cooperative. Also, the data measured by the sensors data are typically shared among nodes to increase their individual efficiency. However, fake reports and jamming are significant threats against these cooperative schemes, as they can propagate fake information across the whole network or induce an exponential computational load. Thus, several cooperative approaches have been proposed to enhance the security of these networks against these threats. Some of them are inspired from game theory: they define a game based on identifying and dismissing rogue data at the lowest consumption cost possible [112, 113]. Other approaches define trust indicators based on the correlations of the data measured by a given sensors with its nearest neighbors [114].

We propose in the following a cooperative verification protocol for IR-UWB IPS, which does not require using DB protocols. Similarly to SPINE, our approach is based on involving mobile tags in the verification process; ultimately, the goal is to have at least one verifier which position is unknown by the attacker. We already demonstrated because of the redundancy factor, the attacker will be revealed by a Total Ranging Deviation (TRD) increase if he fails to guess the position of one verifier or more. Thus, as long as the positioning algorithm manages to obfuscate the position of at least one verifier, DB is not required to achieve secure positioning.

VI.2.2.2 **Attacker's model**

Anchors positions are public knowledge in IPS, but this is usually not the case of tags' positions. Indeed, depending on the centralized or decentralized nature of the IPS, the position of a mobile tag is known by the tag itself and/or the central authority, but does not need to be known by others when it comes to positioning. It could be needed by other nodes for other purposes than positioning (e.g., two robots performing coordinated tasks), but in that case the frame containing the position can simply be ciphered. As a consequence, an attacker is typically not aware of the position computed by the IPS.

However, we have shown in section V.2.7 that an attacker can learn the position of a tag by simply eavesdropping to its ranging protocols and performing least-squares regression. Also, based on the previous section, the attacker could also use multiple rogue nodes and perform differential ranging on a target tags. In both cases, the position obtained has a degraded accuracy compared to the one computed by the IPS. Hence, it is assumed that the attacker can learn the positions of all mobile tags with a degraded accuracy.

The attacker's goal is to tamper the position of a node. This node can be under control of attacker and be part of the network, or be an honest node victim of spoofing attacks. In other words, the proposed countermeasure aim to detect any fraud on the position integrity, regardless of whether that fraud comes from the localized node *(internal attack)* itself or from an attack of external node *(external attack).*

VI.2.2.3 **Proposed algorithm**

In this approach, we assume the following TDMA scheduling scheme. We refer to the timespan dedicated to a single TWR protocol as TDMA slot. Hence, in our model a TDMA slot does not correspond to a single frame but to a complete TWR protocol, which requires a total of three frames. Indeed, it is better for the localization accuracy to keep the reply time as short as possible. Thus, even if the localization frequency is moderate, the TWR will be completed as fast as possible, and there will be simply large time gaps between two TWR protocols as illustrated in **Fig. 40**. The TDMA scheduling protocol always cycle through all the anchors first before switching to a different tag. This is shown for three anchors in **Fig. 40**; anchor 1, 2 and 3 perform successively ranging protocols on tag 1, before proceeding to localize tag 2. We define as ranging cycle the series of TWR protocols performed by A_1, A_2, …, A_N on a single tag T_i. When a given ranging cycle $R_k(T_n)$ is completed, the anchors proceed to localize T_{n+1}. This is typically more efficient than doing the opposite as it ensures that the timespan between the measurement of the first anchor and the last anchor is short enough to reduce the error induced by the potential motion of the tag between two distance estimates.

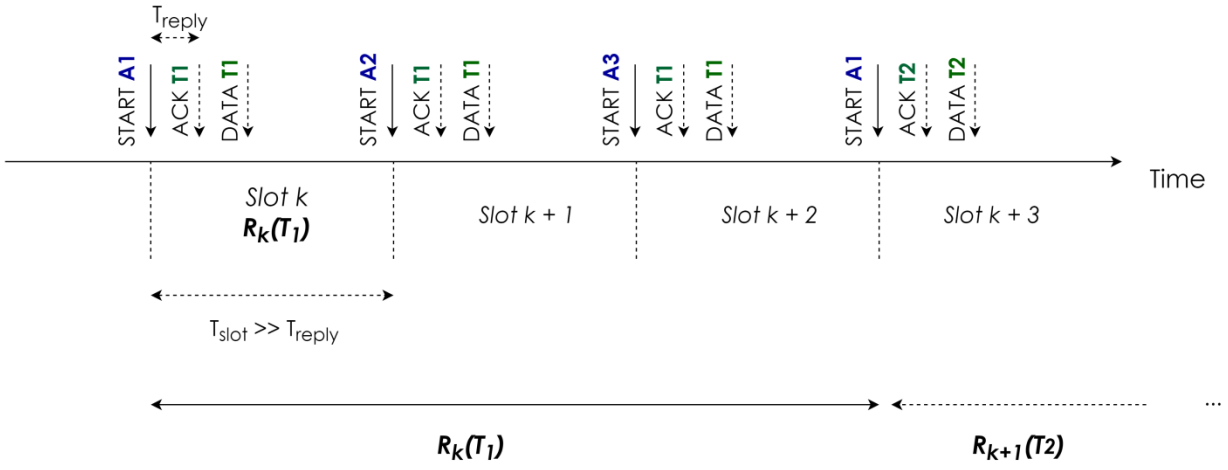


Fig. 40. Proposed TDMA model for cooperative verification

The anchors are assumed to know their total number. Thus, for N anchors, A_1 starts a new cycle when A_N has completed its ranging.

Principle— We notate N the number of anchors and K the number of mobile tags. In a normal unsecure ranging cycle, all anchors in {A_1, …, A_N} proceed to localize a given tag Ti which belongs to the tag set {T_1, …, T_K}. The k-th ranging cycle targeting a tag T_i is notated $R_k(T_i)$, similarly to the notation introduced in **Fig. 40**. In the proposed cooperative verification protocol, we introduce an additional verifier, which we refer to as ghost anchor. This ghost anchor is a mobile tag T_j that has been designated randomly and secretly. In the TDMA scheduling protocol, one slot is dedicated specifically to the ghost anchor.

Details— For this approach, the START frame is ciphered and authenticated with AES-CCM*. The first anchor A_1 has extra duties in this approach and has the task to designate ghost anchors and manage the cooperation verifications. We refer to the first anchor as *master anchor.* The protocol is done in several steps, illustrated here for the ranging cycle of a tag *R*(Ti).

Pre-steps: For each ranging cycle $R_k(T_i)$, a ghost anchor may be associated to that ranging with a certain Probability P_v. In that case, the ghost anchor is randomly picked in the tag set $\{T_1, \dots T_N\}$-$\{T_i\}$.

If cooperative verification is enabled, the master anchor must generate two additional strings:

1. A random reply time that will be used for the ciphered random reply time approach described in section VI.1.3. The reply time is sampled from a uniform distribution $U([\sigma - Smax, \sigma + Smax])$. Refer to section VI.1.3 for details.
2. Optionally, a 128 bits AES-key.

The master must always have pre-computed the associated ghost anchor for at least the K next ranging cycles.

Algorithm— The master anchor initiates each ranging cycle and proceeds to the following operations for each ranging cycle $R_k(T_i)$:

1. Check if T_i is a verifier in one of the next (K- 1) ranging cycles. If not, proceed directly to step 5. Otherwise, proceed to the next steps.
2. Compute the number of slots that T_i has to sleep before waking up as a verifier. This is necessary to avoid T consuming unnecessary energy. This number of slots is calculated relatively to the slot of the master anchor, i.e., the transmission of the START frame from A1 to T_i.
3. Generate a random reply time that will be used for the ciphered random reply time approach described in section VI.1.3. The reply time is sampled from a uniform distribution $U([\sigma - Smax, \sigma + Smax])$. Refer to section VI.1.1 for details.
4. If integrity protection is enabled, generate a 128 bits- AES key for (T_i, T_j). We explain further under which conditions integrity protection is enabled or disabled.
5. Check if cooperative verification is enabled for this cycle. In the START frame for T_i, add the following 51-bytes payload extension:

```
boolean prover_flag ;  /* 1 byte */
unsigned long prover_reply_time ; /* 8 bytes */
byte_array p_AES_key; /* 16 bytes */
byte sleep_slots; /* 1 bytes */
boolean verifier_flag ;  /* 1 byte */
unsigned long verifier_reply_time ; /* 8 bytes */
byte_array verifier_AES_key; /* 16 bytes */
```

If the prover flag is set to *true*, T_i is notified that a cooperative verification will occur at this cycle; hence, T_i should stay awake one extra slot for the additional ghost anchor verification. The prover reply time and AES key that are provided in the extension are for the current ranging cycle and will be used by T_i to complete the ranging protocol with the ghost anchor.

If the prover flag is set to *false*, the tag is notified that there is no cooperative verification for this cycle. In that case, the master anchor replaces both the reply time and the AES key by gibberish data to preserve the length of the extension. This is necessary to prevent the attacker from guessing anything about the cooperative verification process by looking at the payload length. As a reminder, the START frame is ciphered so as to prevent an attacker from guessing anything about the actual content of the payload. Similarly, if cooperative verification is enabled but not integrity protection, the AES key is also gibberish and will not be used by the tag.

If the verifier flag is set to *true*, T_i learns that it will proceed to **ghost anchor verification** in one of the following ranging cycles. T_i calculates the time of its verification based on the number of slots to wait provided by the master anchor. The verifier reply time and AES key provided are used by T_i as ghost anchor when it wakes up and proceeds to the ghost verification, detailed below.

Ghost anchor verification: ghost anchor verification is based on Lightweight TWR (LTWR), with the ciphered randomized reply time approach defined in section VI.1.1. If integrity protection is enabled, a Message Authentication Code (MAC) is appended to the START frame, generated by AES-CBC-MAC 128. Otherwise, the START is sent in plaintext.

The ghost anchor reports the results of the verification in the DATA frame during its next ranging with the master anchor. The results include the distance measured and also the potential fraud detected (i.e., detection of an overlapped preamble) by the ciphered reply time protection. The distance obtained is compared to the distance between the two claimed positions of the ghost anchor and the prover. If the two distances are too far away, or if overlap frauds are detected, an alarm is triggered. If an attack is occurring, alarms will be triggered repetitively.

The algorithm is illustrated for an example with 3 anchors and 3 tags in **Fig. 41**.

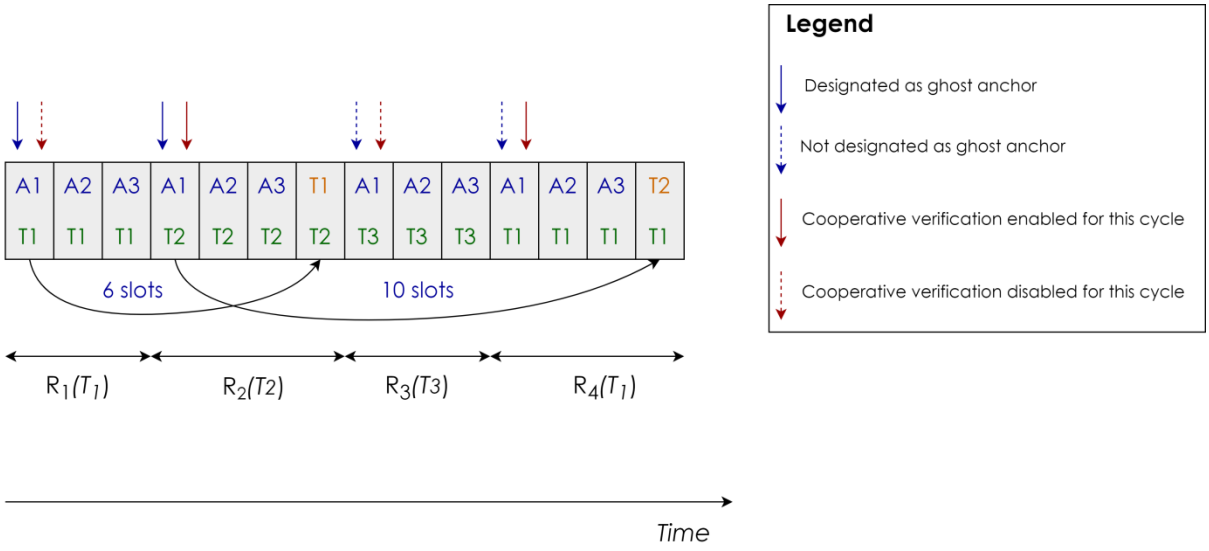


Fig. 41. Cooperative verification example (K = 3, N = 3)

In the first ranging cycle, Tag 1 is informed that it must act as ghost anchor in 6 slots. Cooperative verification is disabled for this ranging cycle; as a consequence, the master anchor starts a new ranging cycle right after A3's slot. During R2, Tag 2 is also informed that it must act as ghost anchor in 10 slots. This time, cooperative verification is enabled. Hence, Tag 2 stays awake for an extra slot to complete a LTWR protocol with the ghost anchor. Tag 1 wakes up after A3's slot and proceeds to Tag 2's verification. Tag 2 is obviously not informed of the verifier's identity. During R3, cooperative verification is disabled and Tag 3 is not prompted to perform any verification, hence the ranging cycle does not include any ghost anchor slot. Finally, cooperative verification is enabled during R4; T2 wakes up after A3' slot and proceed to verify Tag 1's position.

It is not necessary to perform systematic cooperative verifications (i.e., $P_v = 1$) because of the consistency detection factor. The attack can only be efficient if it is applied for an extended period. Hence, P_v can be defined based on the acceptable average time-to-alarm for the application considered. For instance, if P_v is set to 10%, an IPS with a localization frequency of 10 Hz will have a time-to-first-alarm of one second on average. This needs to be adjusted based on the typical false positive rate; indeed, one alarm might not be enough to safely decide if an attack is occurring. We discuss this aspect in the results based on the obtained false positive and false negative rates.

Trust in the verifier—so far we assumed that a compromised tag is either an internal attacker and or a victim of an external attack. From the cooperative's protocol perspective, these two scenarios are equivalent as they are both reflected by a tampered position of the concerned tag. However, since tags are not trusted, the fraud could also be imputed to the ghost anchor, which is potentially malicious. In that case, the ghost anchor might simply lie on the measured distance to create a false alarm. Thus, there are 4 possibilities in a

cooperative verification regarding the honesty of the verifier V (ghost anchor) and the prover P:

1. *P and V are both honest.* If a fraud is detected, it comes necessarily from an external attack in that case
2. *V is honest but not P*. In that case P is mounting an internal attack.
3. *P is honest and not V*. In that case, V has lied on the measured distance and is trying to dismiss P by triggering false alarms.
4. *P and V are both dishonest (and allies)*. In that case, they can easily collaborate to return a coherent distance to the central authority and the fraud will not be detected.

The fourth point implies that if the attacker has deployed multiple nodes, the mutual verification among these nodes will be under its control and he will be able to mask the frauds. In that scenario, a given tag T_i mounting an internal attack might be identified as honest by a subset of the tag populations (the honest tags) and identified as rogue by the rest of the tags (the rogue tags deployed by the attacker).

As a consequence, in that scenario the decision relies currently on a **majority-based vote:** a node will be identified as malicious if more than half of the ghost anchors are reporting frauds. This implies that the attacker will be able to bypass the countermeasure if he can circumvent half of the network or more.

Discussion on the attacker's strategy—Regarding that voting scheme, it might look advisable to keep out of the vote the nodes that have already been identified as potentially suspicious, or to give them a lower ponderation. This is often the case in trust-based approaches for security. However, that feature would expose the system much more to false alarms: the attacker could exclude honest nodes from the vote simply by reporting bogus frauds on malicious nodes. Also, a node that is mounting an internal attack will not necessarily report bogus frauds as a ghost anchor and vice-versa. Consequently, if the attacker's strategy is to dedicate some malicious nodes exclusively to internal attacks and some others exclusively to fake ghost anchor reports, excluding suspicious nodes from the vote would be inefficient and counter-productive. We can illustrate that with a simple example. Let's assume that there are 7 honest nodes and 3 malicious nodes in an IPS. If all nodes get to vote equally, the malicious nodes will always be a minority and any attempt to coordinate an internal attack will be dismissed by the IPS as in the best case they will get 3 votes against 7. Now, if we assume that the system excludes from the vote the nodes that are identified as malicious by two other nodes or more, a vulnerability window is opened. Indeed, the attacker's strategy can be the following. First, the three malicious tags start by being honest on their positions, and are detected as trustworthy. Then, these they proceed to report fake frauds as ghost anchor by simply sending bogus ghost anchor reports, and proceed to exclude one by one honest nodes from the vote. When the number of voting honest nodes is reduced to 2, the attacker gets the majority and can start mounting internal attacks. Hence, the current decision scheme is based on an equally balanced majority-based vote.

Regarding integrity protection— we did not discuss the choice of integrity protection so far. Cooperative verification security does not rely on the integrity of the ranging protocols, which means that the protocol is still secure even if the attacker can tamper accurately the distance estimates of the ghost anchor. If this verification is not cryptographically protected, the tampering process is much easier, yet it does not allow the attacker remaining undetected.

However, not protecting the integrity creates a breach for false alarms. Indeed, the attacker might not be able to tamper the distance measured by the anchors themselves and choose to attack the less protected verifications of the ghost anchors. In that case, victim nodes' positions are actually correct; yet, the attacker will trigger alarms and make the victim node appear as compromised.

One strategy to approach that problem is to use integrity protection as a second barrier. Integrity protection can be disabled by default, and enabled in case of fraud detection. In that case, if the attacker was only able to tamper the verifications of the ghost anchor, he or she will lose that capability and false alarms will stop. If frauds are still being detected, it can be assumed that the alarms were legit.

VI.2.2.4 **Results**

As the capability of an attacker to tamper successfully a given position relies on the accuracy of its guess on the ghost anchor position, we first evaluated the efficiency of the countermeasure to detect attacks for different level of knowledge of the attacker. In the following experiments, it is assumed that the attacker can predict the position of the ghost anchor with a given accuracy. We implemented an internal attack in which the attacker was informed of the positions of the ghost anchors. However, a variable error was introduced in the provided positions, which we refer to as *prediction accuracy*.

First experiments: Attacker's knowledge—We monitored the average absolute difference between the distance returned by the ghost anchor and the distance estimated by the IPS between the ghost anchor and the prover, which we define as *ghost anchor error*. Similarly to the previous experiments, we first benchmarked this error in non-adversarial conditions for the three types of environments aforementioned. Then, we evaluated this parameter with an internal attack, while varying the prediction accuracy from 1 m to 0.

The experiments have been done with 4 anchors and 6 mobile tags. The verification probability P_v was set to 20%. The platform dimensions were 1.8 m by 4.8 m. Each of the measured average error has been evaluated on 200 samples.

The results are shown below in **Fig. 42**.

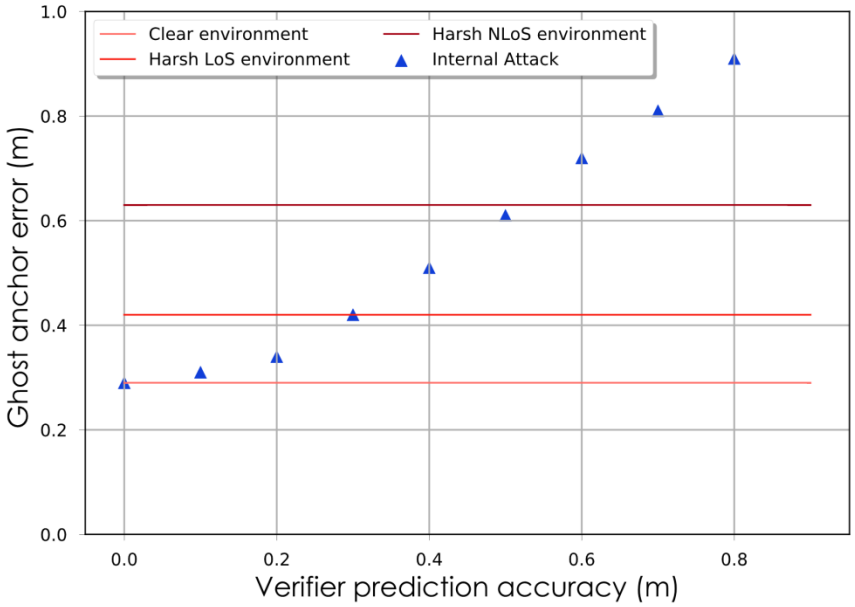


Fig. 42. Ghost anchor error for different prediction accuracy

Not surprisingly, we can observe that the relation between the prediction accuracy and the ghost anchor error is close to linear. In non-adversarial settings, the average ghost anchor error is 29 cm for a clear environment, 42 cm for a harsh LOS environment and 63 cm for a harsh NLOS environment, which are shown as red lines in **Fig. 42**. In adversarial settings, if the attacker can pretend to predict with about 50 cm on average the position of the ghost anchor in a harsh NLOS environment the IPS will not necessarily detect the attack. If the accuracy of the prediction is worse than that, the average error clearly exceeds the non-adversarial values.

Second experiments: Confusion performances and time to alarm— We evaluated the average time-to-alarm and confusion performances in the incoming experiments. This time, we assume that the attacker has circumvented part of the network, and controls the distance reports of a certain number of nodes. We evaluated the confusion performances in function of the percentage of the tags population controlled by the attacker.

Decision scheme for attack detection— For a given tag T, we notate V the total number of ghost anchors that have verified the position of T at least once, and V_{min} the minimum value of V required before the central authority takes a decision. For instance, one ghost anchor verification is typically not enough to take a decision as the detection of a fraud detection might be due to measurement noise. If the votes for malicious and non-malicious behavior are equal, the node is considered as malicious. We first proceeded to evaluate the most appropriate value for V_{min} in a non-adversarial harsh NLoS environment. The threshold for distance fraud detection was set to 80 cm, i.e., any ghost anchor error superior to 80 cm is considered as fraudulent. On the virtual tags, the measurement noise was emulated based on

the performances evaluated in III.3. We obtained a 99.9% confidence interval for $V_{min} \geq \mathbf{0.3K}$, with K the total number of anchors, which was set to 20 in our setup, leading to $V_{min} \geq 6$. Hence, setting V_{min} to 30% of the total number of tags is enough to maintain the false positive rate below 0.01% even in harsh conditions. We used $V_{min} = 6$ in the following experiments.

Setup- The simulation layer of SecureLoc has been used for these experiments**.** The experiments have been held with 8 physical tags and 12 virtual ones. The platform configuration was the same as the previous experiments, with 4 anchors on a surface of 4.8 by 1.8 m. Internal attacks are simulated on both real and virtual tags, i.e., the simulation layer is computing the distance shifts required for the internal attacks and tampers the MQTT samples accordingly. Regarding the attacker's model, this is strictly equivalent to mounting the real attack on a rogue node (as we did previously) considering that the distance shift calculations are exactly the same. Simulating the attack instead of using a real implementation is only needed to properly emulate the behavior of the virtual tags, as they need to be informed of both the tampered and real position.

The behavior of each virtual tag can be set to honest or malicious; in the case of a malicious behavior, the following actions can be enabled on the virtual tag:

- Mounting an internal attack
- Lying on its ghost anchor reports to help dissimulating the internal attack of another malicious node
- Lying on its ghost anchor reports to trigger false alarms on honest nodes

In the following experiments, we gradually increase the number of malicious tags on the network from 0 to 10 (50% of the network). This time, the malicious tags are not aware of the positions of the honest nodes, which implies that they can only take blind guesses when receiving ghost anchor verifications from these ones. However, malicious tags know the positions of each other. Basically, if a malicious tag verifies the position of an ally, it can compute the proper distance to report in order to dissimulate an internal attack. Half of the malicious tags (rounded up) were mounting an internal attack; all the malicious tags were collaborating to hide the internal attacks of their peers. Half of the honest tags were targeted by false alarms attacks, as the malicious tags were deliberately reporting bogus distance reports in their ghost anchor verifications. Note that these proportions (50-50) are quite arbitrary as they have little impact in the confusion performances; they were mostly chosen to maximize the accuracy of the characterization.

The True Negative Rate (TNR) and False Positive Rate (FPR) obtained are shown in **Fig. 43**. In each scenario, the TN and FP rates have been evaluated on 2000 localizations.

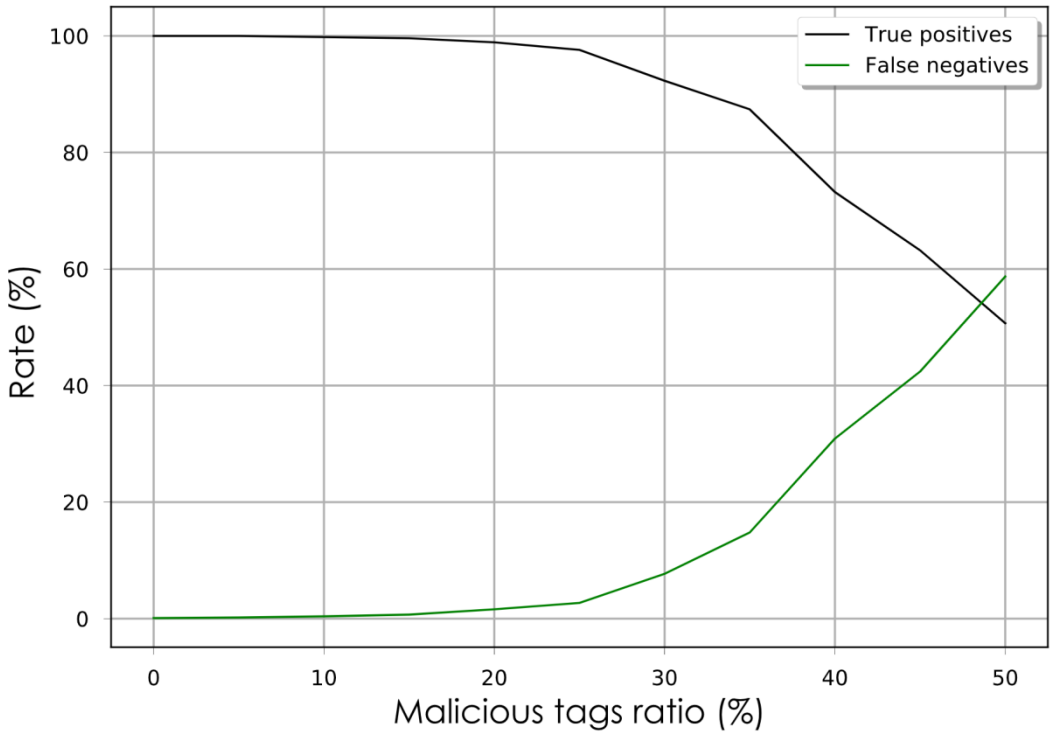


Fig. 43. True positive and false negative rates for different ratios of compromised nodes

The confusion performances are good when the attacker's network coverage is below 20%, attacker's coverage being defined as the proportion of the total number of nodes compromised by the attacker. Past 20%, the TPR start to decrease and the FNR to increase noticeably. For a coverage of 50%, which is the theoretical boundary for the protocol's efficiency, the false negatives rate is 58.7%, which is actually more than a balanced coin flip. This is due to the occasional natural false alarms: there are a few erroneous fraud reports from the honest nodes due the measurement noise, which give an additional advantage to the attacker.

We also extracted the average time to alarm from these data. The time to alarm defines the time elapsed between the starting time of an attack and its detection by the protocol. An attack is considered as detected when the central authority deciding to classify the node as malicious. We define by *time to first alarm* the time elapsed between the attack start and the first fraud report incoming from a ghost anchor. They are not given in time units but in localization units, as the corresponding time depends on the number of tags, number of anchors and localization frequency. A time to alarm of ten localizations should be interpreted as the malicious tag having to be located on average ten times before detecting the fraud. As the relation between the number of cooperative verifications depends directly on the verification probability, the relation between the time to alarm and the verification probability is linear (e.g., an attack will be detected ten times faster for P_v = 100% than for P_v = 10%). We estimated the time-to-alarm on a total of 8000 position log entries, based on the

same platform setup as for **Fig. 43**. The results are given in **Fig. 44** for a verification probability of 100% (i.e., systematic verifications). The time to alarm results for other verification probabilities can simply be obtained by multiplying the time-to-alarm by the verification probability.

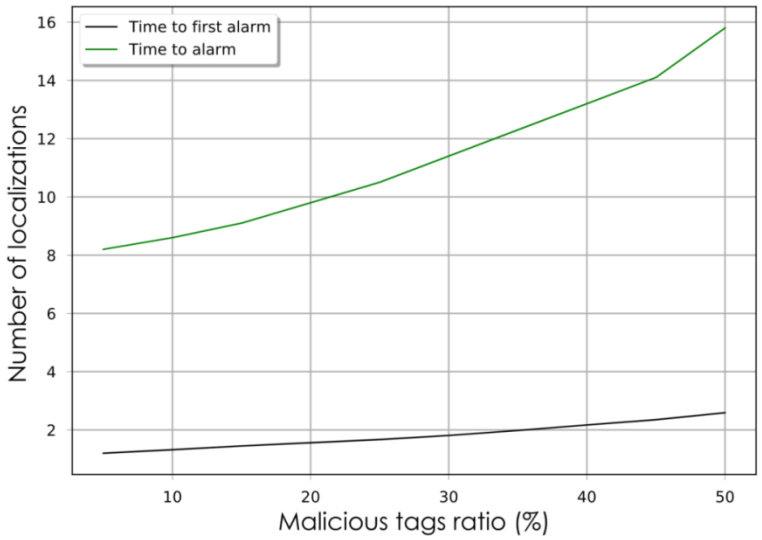


Fig. 44. Time to alarm and time to first alarm for different ratios of compromised nodes, given as a number of localizations; the actual time depends on the localization frequency

As the probability for a malicious tag to be picked as ghost anchor is linearly related to the attacker's coverage, the time to first alarm has a linear trend: indeed, the first alarm usually occurs the first time an honest ghost anchor is picked. As malicious tag have an additional advantage due to the possibility to occasional bypass the protocol with a blind guess, the average time to first alarm is 2.5 and not 2 for an attacker's coverage of 50%.

Regarding time to alarm, as $V_{min} = 6$, a certain number of localizations have to occur before attack detection even for a low attacker's coverage; for one malicious node, about 8 localizations are required on average. Contrary to time to first alarm, time to alarm grows faster when the attacker's coverage increases. The time-to-alarm goes up to 15.8 localizations for 50% coverage.

If we take for reference the benchmark at 20% coverage, the time to alarm is 9.8 localizations. The localization frequency was set to 10 Hz (i.e. each tag is located 10 times a second), which gives an average time to alarm of 0.98 s. For 4 anchors and 4 tags, a localization frequency of 20 Hz corresponds to a ranging slot length of 1.35 ms.

VI.2.2.5 **Discussion - Perspectives for improvement**

This cooperative scheme relies on the difficulty for an attacker to guess the identity and position of a verifier. If an attacker can guess the position of a verifier with an accuracy better than 80 cm, he or she can potentially bypass the verification.

We already mentioned that the attacker could try to get the positions of all the mobile tags of the system. This is theoretically possible if the attacker deploys enough malicious nodes, as he or she could mount an alternative IPS based on differential ranging. Assuming an accuracy better than 80 cm with this differential ranging scheme (which is much less accurate the direct ranging schemes used by the legit IPS), the attacker would have simply to guess which tag has been picked as ghost anchor at each verification. We identified three main schemes that could allow the attacker to do so:

- Using directive antennas
- Analyzing the reception power from the designated ghost anchor to guess the most likely candidate based on signal attenuation
- Comparing the skews of the ghost anchor to the previous recorded skew of all tags

Regarding reception power, a transmission power randomization scheme could be investigated to reduce the amount of information that an attacker can extract from signal attenuation. However, directive antennas and skew fingerprinting remain valid threats even with that countermeasure. Ultimately, the cost of such attacks is quite high, as it involves multiple nodes, specific hardware and a quite advanced localization performances with only differential ranging. A perspective to push the security of that scheme further is to investigate a passive verification scheme, in which the ghost anchor does not emit anything. The drawbacks are the loss of accuracy in the verification and the implication of a second ghost anchor, as differential ranging will require at least two verifiers.

In conclusion, the cooperative approach can provide a powerful alternative to differential ranging against internal attacks. For differential ranging, the attack opportunities were depending on the capability to send the same frame in multiple directions with fine-controlled delays. For cooperative approaches, it depends mostly on the attacker's coverage and its ability to mount an alternative IPS to uncover the positions and identities of the verifiers. The proposed cooperative protocol involves extra duties for the tags, as they are involved in more ranging protocols than usual. However, it can detect both external and internal attacks without any physical requirement, and are efficient even if the physical and MAC layers are both flawed as the measured distances are always assumed to be vulnerable.

Besides the presented work on system-level countermeasures, our experiments led us to interesting side results regarding the physical layer of IR-UWB. In the next section, we introduce the last contributions of this manuscript, which deal with node authentication and key establishment at the physical-level on commercial IR-UWB chips.

VI.3 A clock-skew based authentication protocol

VI.3.1 Context

Node authentication cannot be secured exclusively with cryptographic primitives for 802.15.4: the cryptographic primitives for authentication in 802.15.4 might be exposed by the vulnerabilities identified in the literature, and authentication codes do not prevent from relay or replay[1] attacks.

External attacks always involve at some point a spoofer: for both physical attacks and spoofed acknowledgment attacks, the verifier will receive at least part of one frame from an impersonator at some point during the attack. Such impersonations attacks are not prevented by cryptographic signatures, as they are based on enhanced replay and/or replay mechanisms.

A promising approach to the authentication problem for IoT devices that has emerged in the early 2000s' is to extract authentication primitives from certain physical imperfections of devices. These imperfections are intrinsic to the manufacturing process and are usually random and unique for each device, which is very strong asset for any authentication protocol. The main challenge is to accurately and reliably evaluate them, which is particularly difficult when the authentication is done remotely, as it is the case in WSN [115]. The notion of Physical Unclonable Function (PUF) has pushed the physical authentication concept further by building cryptographic primitives from the physical imperfection randomness [116]. Most PUFs are stimulated by a challenge, which allow characterizing the physical device, and can be consequently used to authenticate devices.

We developed a physical authentication protocol for 802.15.4 IR-UWB systems based on the clock skew variations across devices. This protocol is based on the observation that every clock has a unique and reproducible imperfection that can be evaluated from the skew measured during communications. Similar approaches have been explored by previous research works, which proposed clock skew fingerprinting in 802.11 multi-hop networks for Access Points (APs) identification [117][118]. However, we observed two main limitations in these works regarding the security and the reliability of the authentication process:

- The clock skew is extracted from the presumably honest participation of the node in a synchronization protocol. An attacker aware of this countermeasure can cheat in the synchronization protocol to match the skew of a chosen target.
- The clock skew is exploited as a static value. However, internal temperature has an influence on clock frequency [119]. That implies that a temperature variation of a legit node might lead to an authentication failure, while an attacker could actually use a temperature gradient to spoof a legit node.

As far as we are aware, clock skew-based authentication has never been proposed for UWB devices in previous works. UWB is actually a good candidate for such approaches as it

[1] Replay protection counters are only effective if the original message has been received. If the original frame has been denied by an attacker, the replayed frame will look legit even with replay protection mechanisms.

does not suffer from the first limitation mentioned above. Indeed, the UWB physical layer, tailored for precise Time-of-Flight, allows estimating the skew of a node directly in the physical layer. The verifier only needs one frame from the prover to get this information. Hence, the authenticated device cannot tamper the value measured for its skew. Regarding the second limitation, we propose to induce a temperature gradient to take into account the relation between skew and temperature.

Before diving into the details of the protocol, we explain the similarities of this approach with weak PUFs and characterize the relation between skew and temperature in the following sections.

VI.3.2 From physical authentication to PUFs

There are multiple approaches to physical authentication in wireless systems based around identifying unique imperfections of Analog Font Ends (AFE). Each chip has some unique and random AFE imperfections inherent to the manufacturing process that can be evaluated remotely by other chips during communications based on the received signal. However, Wang et al. pointed in [115] that AFE imperfections-based authentication generally suffer from reliability problems: the precision of the measurements that is achievable does typically not provide sufficient entropy, as the probability to find two devices that the probe cannot differentiate is unneglectable.

This idea to use random physical imperfections has also been exploited in the field of PUFS. The concept behind PUFs has been first introduced as Physical One-Way Function in 2001 by Pappu in 2001 [120], and later formalized by Gassend et al. as Physical Unclonable Function [121]. Several attempts to propose larger definitions of PUFs have been made in the literature, leading to the concepts of weak and strong PUFs [116], defined in the following.

The intuition behind PUFs is to exploit unique and unclonable properties of a physical process to build authentication or encryption primitives. Building a PUF requires two entities, a physical device and a probe to evaluate one or multiple unclonable properties of this device, as shown in **Fig. 45**. The physical device is stimulated by a challenge and the output of this challenge is estimated by the probe. PUFs are often classified in two categories, weak and strong PUFs [116]. Strong PUFs are typically used in applications where the probe is held by the prover (P) and not the verifier (V): in that case the physical device P is a black box from V's perspective, and V cannot make the difference between the actual physical device P and an adversary numeric process modeling this device which simply contains known Challenge-Response Pairs (CRPs).

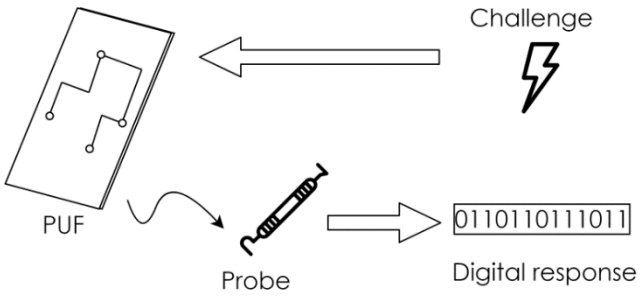


Fig. 45. Illustration of PUF use

As a consequence, the same challenge should not be used twice in the potential presence of adversaries [122]. On the other hand, a weak PUF has a small CRP space, sometimes reduced to a single CRP, hence is predictable. Contrary to strong PUFs, weak PUFs cannot be used for cryptographic operations. However, they are very effective as an authentication mean, on which we focus in the following. Weak PUFS are meant to reuse the same challenge multiple times, which means that the verifier has to hold the probe to evaluate the physical device. If the physical device is indeed unclonable, then the authentication process is secure, even if the same challenge is reused. A weak PUFs should fulfill the following requirements [116]:

- Unclonability of the physical device: the probability that two devices with the same physical architecture provide the same outputs to the challenge is negligible. Hence, the PUF should be tamper-evident, i.e., any response from an attempted counterfeit should be differentiable from the real one.
- Efficiency of the evaluation process: the output of the challenge can be estimated in a polynomial time by the probe.
- Reproducibility of the challenge: for a given argument, the challenge should always give the same output. In practice, this output is noisy for most PUFs and will require to be processed by a fuzzy extractor [122].

In that regard, the skew-based authentication process proposed in this section has weak PUF properties and satisfies the typical use case of such PUFs. The physical process evaluated is the UWB chip system clock deviation. Since the skew evaluation is done directly in the physical layer the verifier has a direct probe on the verifier, which is a powerful asset.

VI.3.3 Skew and Temperature Characterization

We define in the following the skew as the deviation of a clock to a reference, in ppm, over a certain period of time. As far as there is a lot of literature regarding clock skew optimization in the context of IC design, the control that a physical device has on its clock skew during runtime is limited. It is well-known that oscillator clock frequency increases with the temperature [123]. However, few papers have characterized the relation between skew and temperature during runtime. In [119], this relation has been identified and shown to be very close to a linear relation on crystal clocks, similar to the 38.4 MHz crystal clocks embedded in DWM1000 chips. We have run similar experiments to characterize the relation

between skew and temperature on several DWM1000 chips. The results for three different tags are shown in **Fig. 46** below:

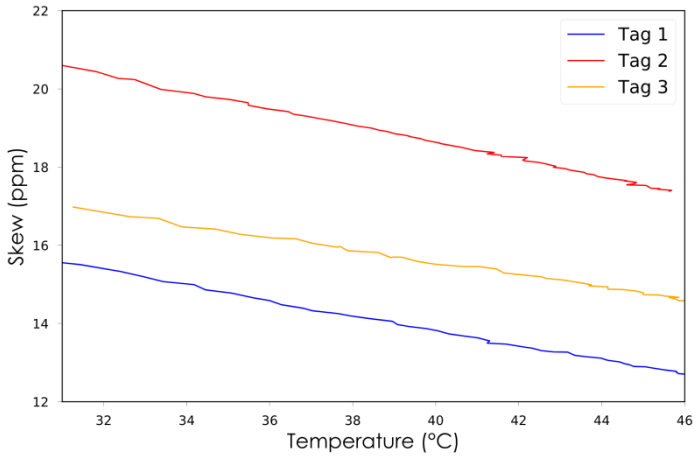


Fig. 46. Skew/Temperature Relation

Two interesting properties can be observed from these results: similarly to [119], this relation can be indeed well approximated by a first order polynomial, and the coefficients of this approximation are unique to each device. As a consequence, this skew can be defined as following:

$$skew(T°) = slew * T°C + offset$$

We proceeded to characterize the slew and offset of different DWM1000 clocks on a population of 12 tags. One of the device has been arbitrary chosen as the tester for the measurements. The tester has been stabilized at an internal temperature of 45°C to prevent its own skew variations from interfering with the measurement process. The measurement process has been reproduced multiple times on each device and at different room temperatures, which were controlled by a climatic chamber. The coefficients obtained were reproducible. All the tested devices have been monitored on continuous transmission with the tester on duration of 120 s. The measurements have been linearized with a least squared polynomial fit to evaluate the slew and offset defined in (1) for each device. The distribution of these two parameters is characterized in **Table 30** below. Given that the offset are defined relatively to an arbitrary tester device, the offset distribution has been centered to a mean of 0.

Table 30. Distribution of the skew/temperature parameters on SecureLoc testbed (12 DecaWino chips)

Skew	**Min**	**Max**	**Mean**	**Standard deviation**
Slew (ppm/°C)	0.141	0.243	0.187	**0.0312**
Offset (ppm)	-13.2	12.9	0	**6.84**

The measurement process has been reproduced at different times and different room temperatures. The slew distribution and the offset distribution have standard deviations of respectively 0.0312 ppm/°C and 6.84 ppm. We observed that each device crystal clock has a unique deviation that is randomly introduced during manufacturing, and that can be characterized in a reproducible manner. On the full temperature operating range of DWM1000 (-40 ~ +85 °C), the skew variations of the crystal clock are estimated to the order of +- 30 ppm [63]. On the typical temperature range of indoor IPS (15 ~ 25 °C), we estimate them to +- 10 ppm in the worst case. As a consequence, skew fingerprinting cannot be secured if it is not correlated with the IC internal temperature. Indeed, the probability that the characteristic skew/temperature of two random devices have an intersection point within their temperature operating range is far from negligible. As a consequence, a secure skew verification process should evaluate the skew on a certain temperature range, which requires inducing a temperature gradient on the tested device.

Internal temperature is mostly related to the duty cycle for IoT devices. In the context of wireless communications, the duty cycle (DC) refers to the fraction of time in which a device is transmitting (Tx) or receiving (Rx). In wireless sensor networks, communications represent the most part of the power consumption of nodes. We neglect the power consumption unrelated to communications in the following.

The power consumption during reception on UWB transceivers is about twice superior to the power consumption during transmission[63]. We empirically observed that a node emitting continuously needs about four times more time than a node receiving continuously to reach the same temperature. Considering that, for simplicity sake, we approximate the duty cycle of mobile tags in the following as the average percentage of time dedicated to reception. As we mainly aim to give an order of magnitude of the duty cycle rather than an exact estimation, which would require a lot of implementation details, this approximation is satisfying in the context of this study.

UWB IPS typically relies on Time-Division Multiplexing (TDMA). For a fixed localization frequency (i.e., how many times each mobile tag is located per second), it is trivial to calculate the corresponding duty cycle, as the inverse of the product of the localization frequency by the time length of a frame. In real-world IPS, the duty cycle is typically below 10% even for high-performance systems (e.g., ~5% for Crazyflie drones IPS [62]). We evaluated the temperature stabilization point of DWM1000 mobile tags for different duty cycles and different external temperatures. We did the experiments in a climatic chamber with temperatures from 5°C to 45°C. The temperature rise on the device after being turned on is shown in **Fig. 48** for different duty cycles; we observe that at the maximum duty cycle,

the temperature increases by 24°C in 2 minutes, with about 8°C during the first 5 seconds. On the other hand, at a duty cycle of 1% the temperature only increases by 3°C in 2 minutes. The absolute temperature rise at the maximum duty cycle is displayed in **Fig**. 48. The relative temperature rise is shown in **Fig. 49**. We observe clearly in this last one that with the exception of the coldest external temperature (5°C), the temperature rise is similar for different experiments.

As a consequence, on the range of typical indoor temperatures (15°C ~ 25°C), the relations between the different temperature curves shown in can be expressed by a simple offset. The temperature quickly reaches an asymptote when the device is running for an extended amount of time, e.g., 52.3°C after 4 hours for a duty cycle of 100% and an external

Fig. 48. For different duty cycles (DC) at Text = 23°C

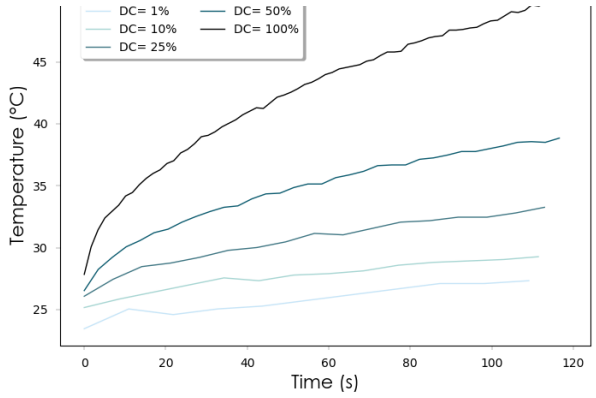
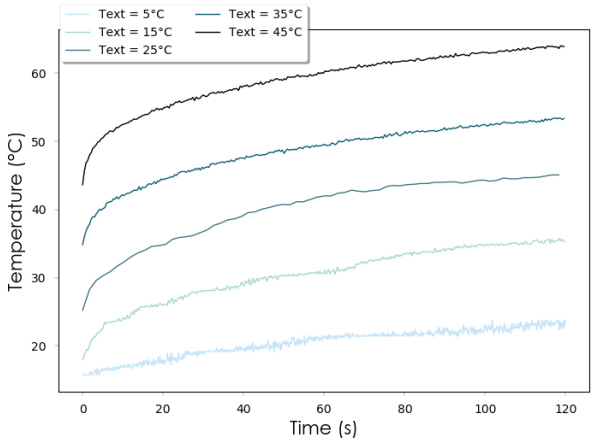


Fig. 47. At different external temperatures



temperature of 25°C.

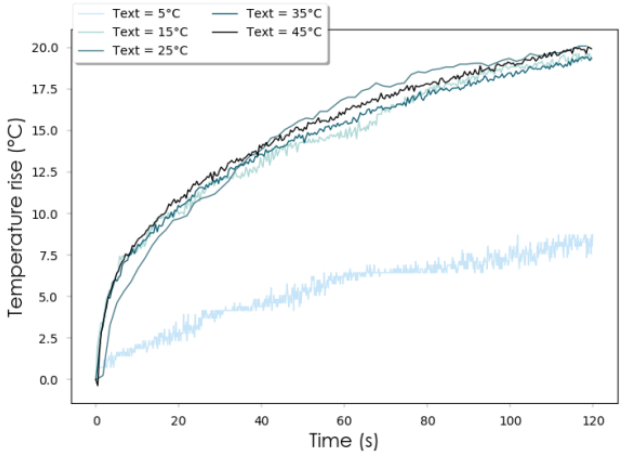


Fig. 49. At different external temperatures (DC 100%), relatively to the starting temperature

The skew/temperature linear relation characterized in this section shows that the clock deviation satisfies several weak PUF properties. First, the parameters of this relation are randomly introduced during the manufacturing process, leading to each device having unique parameters. The capability of an attacker to produce an accurate counterfeit is limited: this behavior depends indeed more on the physical nature of crystal clocks than on the IC design used, and the parameters that allow to modify this behavior during runtime are also difficult to master. Although digital crystal clock trimming at runtime is possible [124], this feature has a granularity of only 1.5 ppm on DW1000 [63] which is far above the resolution of about a tenth of a ppm observed here. Voltage and mechanical stress have also an influence on crystal clock frequencies, yet, we are not aware of any work exploiting that to get the type of fine-grained sub-ppm modification required to produce such counterfeits. Regarding voltage, we did not observe any significant difference on the skew between power supplies of 3.7 and 5V. Concerning mechanical stress, they require a direct access to the clock and would most likely produce unstable skew variations. As a consequence, the skew/temperature relation has good unclonability properties. That relation can be estimated directly into the physical layer by any UWB device, which means that the verifier holds a direct probe, and this probe is accurate enough to differentiate devices as shown in **Table 30**. Finally, the temperature gradient induced when switching to a maximized duty cycle can be exploited as a challenge to characterize the PUF. Thus, we propose and evaluate an authentication challenge in the following section.

VI.3.4 Skew authentication protocol

Considering the observations of the previous section, building a challenge between a prover P and a verifier V to characterize the prover's skew has three main requirements:

1. Providing accurate measurements of P's skew
2. Inducing a temperature rise on P, ideally fast enough to keep the challenge relatively short
3. Providing an accurate estimation of the distance between P and V

The third requirement is necessary to guarantee the distance integrity asset. Estimating the distance during the challenge prevents a spoofer to simply dodge the authentication challenge. For example, let Alice an anchor and Bob a mobile tag, with none of them being aware that Bob has been spoofed: Alice thinks she is performing ranging protocols with Bob but is actually estimating its distance to Bob's impersonator, Eve. If Alice starts now an authentication challenge with Bob, and cannot estimate the distance during the challenge, then Eve can simply turn off the attack during the challenge, let Bob successfully complete the challenge, and restart immediately after. In that case, Alice will not detect the spoofing attack. However, if Alice can actually estimate the distance to Bob during the protocol, then this distance will not match the ones that were received before, and the attack will be revealed. If Eve tries instead to complete the authentication protocol by herself, then the challenge will be failed and the attack will be revealed as well. As a consequence, the

authentication challenge should include ranging features. Based on these considerations, we propose a challenge in the following, shown in **Fig. 50**.

Principle— The challenge is based on a burst of frames in a "Ping-Pong" manner, where P and V have to exchange N frames as fast as they can. In this protocol, V is emitting frames with long preamble length (4096 symbols), whereas P uses short preamble length (128 symbols); they both use empty payloads. As a consequence, V's frames are about 30 times longer, which means that although P and V will both send N/2 frames in total, P spends more than 95% of its time on reception. Thus, the duty cycle of P is close to 100% and the temperature gradient is maximized. V records P's skew after each frame received and obtains a vector of N/2 skew samples after the challenge's completion. This output forms the signature of P. After validating the authenticity of P, the distance measured can be compared to the values that have been measured before the challenge to verify that P was not being tampered.

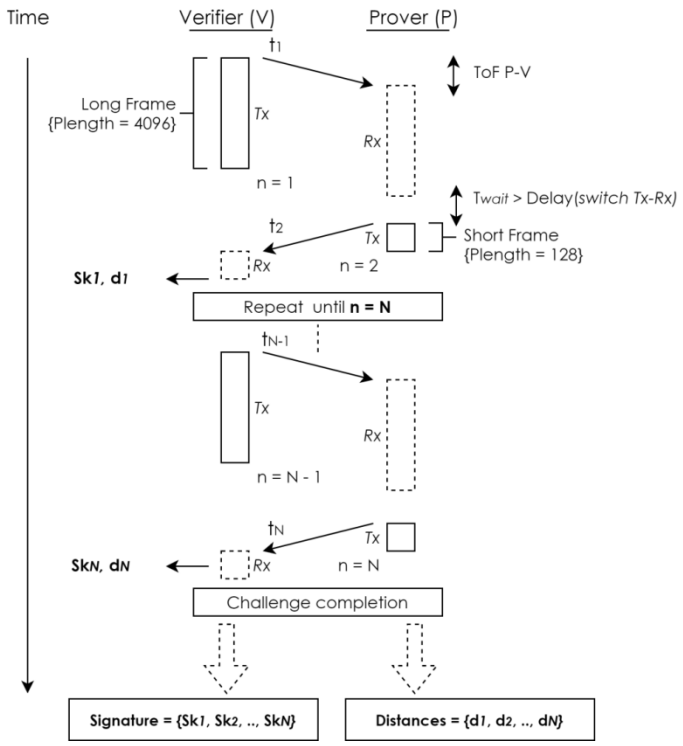


Fig. 50. Authentication challenge – Fast frames exchange with skew (Sk) and distance (d) sampling

Details– The challenge can be done on any frequency channel. This does not affect noticeably the accuracy of the skew estimation. However, it is preferable to switch a different channel than the one used for ranging to avoid perturbing other nodes. We used channel 2 in our implementation. After each message n from V, P uses the delayed transmission feature of

DWM1000, which allows scheduling precisely a transmission, to reply after T_{wait}. A short T_{wait} is preferable to optimize the challenge, we used T_{wait} = 30 µs in our implementation. The reply from P is received by V at t_{n+1}. This is equivalent to a LTWR protocol and V can compute the distance after each reply of P. Considering that the distance is estimated continuously throughout the challenge, it is not an issue if the prover is moving during the protocol.

The number of exchanged messages N was set to 1500 in our implementation; increasing N can improve the accuracy of the signature characterization but increases the cost in time and consumption of the challenge. For 1500 frames, the challenge length is 3.6s and the temperature increase about 5°C. We discuss in section VI.4.2 the impact of the challenge on the overall consumption.

Concerning the skew samples of the verifier, there is a correction applied so the temperature variations of V do not affect the accuracy of the protocol. V's skew/temperature offset and slew have been characterized and their values are known by V. The offset between the external temperature of the challenge and the reference signature of P is denoted by ΔT_{ext}; the difference between the internal temperature of V and the reference temperature of 45°C is denoted by ΔT_{int}. The skew corrected value $\widetilde{Sk}$ is extracted from the measured skew value $\widehat{Sk}$ as defined in Eq. 10:

Eq. 10

$$\widetilde{Sk} = \widehat{Sk} + \Delta T_{ext} * slew_P + \Delta T_{int} * slew_V$$

ΔT_{int} is estimated by V from its internal temperature sensor. Evaluating ΔT_{ext} is more complex; in most indoor environment, the external temperature of two UWB nodes will be very similar. If V has an external temperature sensor, the estimation of ΔT_{ext} is trivial, assuming that V and P are in the same room; if it does not, it can be estimated from the skew variations of other trusted nodes nearby. We discuss the impact of the accuracy of ΔT_{ext} estimation in the following section. After applying the extractor, the Root Mean-squared Error (RMSE) between the evaluated signature and the reference is computed. If the RMSE is above a given threshold, the challenge is failed, otherwise, the authentication succeeds.

VI.3.5 Performances

We evaluated the performances of this authentication challenge in a climatic chamber, on its reproducibility and discrimination capabilities, which are two major parameters to consider in the evaluation of physical authentication protocols [125] and weak PUFs [13].

The reproducibility for a given RMSE threshold is directly reflected by the false positive rate (FPR). If the challenge output is inconsistent, then it will deviate often from the reference signature and generate false alarms. The discrimination performances are reflected by the true positive rate (TPR): the higher is the distance (in terms of RMSE) between the signatures of different tags is high, the higher is the impersonator detection rate. The value of the RMSE threshold for the attack detection is a compromise between the TPR and FPR; it is typically chosen from a Receiver Operating Characteristic (ROC) curve which characterizes the relation between FPR and TPR. We established the ROC curve of the proposed authentication protocol in the following experiments.

Setup– 6 tags have been used as provers for the first experiment series, which aim to demonstrate the feasibility of the approach. All the possible victim-impersonator pairs have been tested for a total of 36 combinations. In the following results, the different provers have been tested using the same verifier tag. For each measurements session, the tags were placed in a climatic chamber during 4 hours. The authentication challenge was triggered every minute; during the rest of the time, the verifier and prover are performing ranging process at a duty cycle of 10%. The reference signature has been established at 20°C. A total of 1200 challenge signatures have been collected on temperatures ranging from 14°C to 26°C, by steps of 3°C. We induced a variable error in the external temperature variation estimation ΔT_{ext}, involved in the skew correction process defined in Eq. 2, to estimate the impact of ΔT_{ext} estimation accuracy in the challenge performances.

Results– The ROC curves obtained are shown below in **Fig. 51**. As explained in section 4.1, the verifier needs to compensate the skew offset induced by the change of external temperature ΔT_{ext} between the reference and the challenge signatures, if any. Each ROC curve corresponds to a different magnitude of error for ΔT_{ext} estimation, that we notate in the following $e(\Delta T_{ext})$, ranging from a perfect temperature compensation to a completely off one.

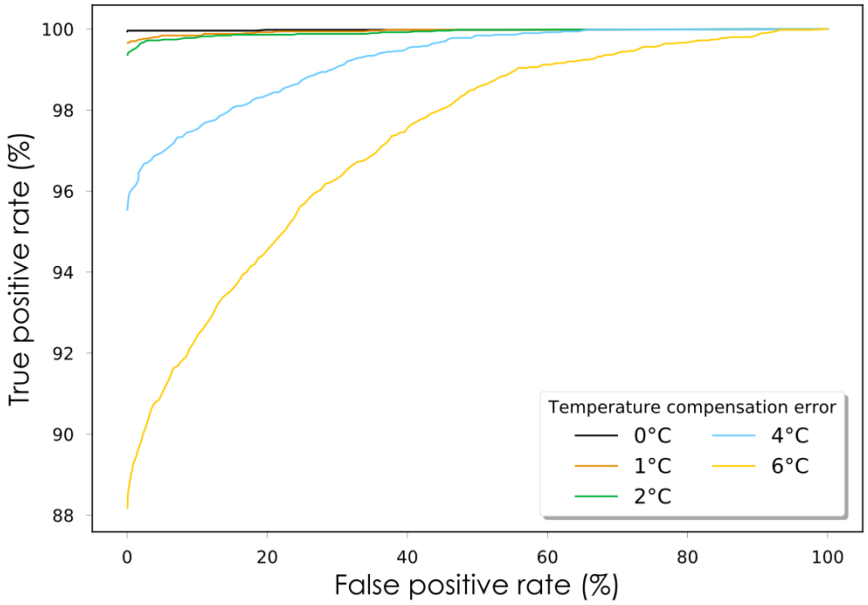


Fig. 51. Authentication ROC curves for different temperature compensation ($-\Delta T_{ext}$) accuracy values

We observe that the ROC curves obtained on the interval [0°C, 2°C] for $e(\Delta T_{ext})$ are close; more significant degradations are observed for 4 and 6°C. An accuracy of 2°C in the external temperature estimation process is very achievable with a basic external temperature sensor or by monitoring the skew variations of trusted nodes. For $e(\Delta T_{ext})$ = 2°C, we have a TPR of 99.1% for a FPR of 0.16%, corresponding a RMSE threshold of 0.19 ppm. Regarding the reproducibility details, the average RMSE distance of a challenge signature to the reference one, for a random temperature challenge external temperature in [14°C, 26°C], is 0.066 ppm. The worst case obtained during the experiment was 0.172 ppm, for Text = 14°C, and the best case 0.018 ppm, for Text = 20°C. Regarding the discrimination capabilities, the average RMSE distance between the signatures of 2 different nodes is 3.86 ppm. The smallest RMSE distance observed for a pair of random node signatures among a population of 12 tags was 1.41 ppm and the highest one 8.89 ppm. Hence, the worst case in terms of reproducibility is still below the threshold identified from the ROC curve, and the worst case in terms of discrimination way beyond. These FPR and TPR results are promising for future larger scale experiments, on a wider node population. It is also good to notice that given the short-range of UWB, the number of tags in range of each other in a real-world IPS rarely exceeds twenty. We discuss the possible applications of this authentication protocol in the next section.

VI.3.6 Discussion on use cases and limitations

The challenge we proposed displays good discrimination performances and allows a verifier to check whether if it is interacting with the authentic physical device or not. As a consequence, it is powerful to detect identity thefts (i.e., spoofing attacks), and identity duplication (i.e., Sybil attacks).

This authentication challenge provides the assets that the prover is the legit physical device and is located at a certain distance at a given time. In that sense, it is very different from a cryptographic signature: cryptographic signatures certify that the content of the frame has been produced by the prover, but do not provide guarantee on who physically emitted the frame, hence, are not secured against relays, replay attacks or other variants (e.g., ED-LC or selective SAA). Nevertheless, this challenge should be used sparingly: given that the prover is required to operate at full-load during the challenge, using it extensively would lead to over-consuming the node resources. For example, with the current challenge duration of 3.6 s, a node working at a duty cycle of 1% will burn a hundred times more energy than usual during the challenge. This means that a single challenge will burn about 6 minutes of its total autonomy. As a consequence, the best way to use this protocol is to trigger the challenge every time a suspicious or sensitive activity is detected. This could be for example a still node starting moving, a sudden instability in the position measurements (typically induced by ED-LC [11] or acknowledgment attacks [102]), or abnormally high transmission powers (also induced by the aforementioned attacks). Also, a lightweight skew supervision can also be implemented in addition to the challenge. Monitoring the skew of mobile tags during the regular ranging process can be enough to detect a spoofing attack if the spoofer has not tuned its own skew before. However, as there is no temperature rise induced, an attacker aware of the countermeasure can remain undetected if he can increase or decrease its own temperature (e.g., with an external source) to match the skew of the victim.

There are some limitations to this challenge. We demonstrated that the protocol is able to deal with moderate external temperature changes, but it will not work properly in environment with complex temperature constraints, such as quickly varying or very cold temperatures, as the temperature gradient induced by the protocol will be less reproducible. Small duty cycle changes do not have much impact as the temperature gradient variation will be negligible, but in a case of a drastic change the reference signature should be updated. An attacker could also try to use an additional heat source to match the signature of a legit node; however he/she would need to generate very quick and fine-grained temperature variations to manage to do so. Such attacks would require some custom hardware and a fair amount of time to tune the attack to the victim skew's profile.

Finally, the proposed protocol is very similar to key extraction protocols based on the randomness of radio channels. We show in the following section that the proposed protocol could also allow two nodes to extract a secret key as an additional feature.

VI.4 Channel-based Key Extraction

There is a wide literature on the topic of channel-based key extraction as the first research works released towards this concept are from the early 1970s. *The wiretap channel* by A.D Wyner [126], released in 1973, is considered as one of the major and earliest references and radio channels secrecy. The major contribution demonstrated by Wyner can be summarized as following: given a wiretapper listening to a Discrete Memory Channel, DMC_1, through a second DMC, DMC_2, there exists a data rate C up to which perfect secrecy can be achieved on DMC_1, even if the codebooks used by the two parties are known by the wire-tapper. Radio channels have a spatial component; if a node A is sending a frame that is received by the nodes B and C, the channels A->B and A->C are different even though B and C are both receiving the same message. If we interpret Wyner's results for that example, that means that there exists a certain quantity of information that A and B can hide form C, even if C is able to decode the frame sent by A. Wyner's results are strongly related to the concept of channel capacity introduced by Shannon in 1948 [127], which allow evaluating the maximum theoretical data rate achievable on a channel for a given noise distribution. From a communication perspective, higher levels of noise tend to reduce the channel capacity. From a security perspective, as studied in the work of Wyner, they can actually increase the potential secrecy of the channels.

If we fast forward in time and look at more recent works on the topic, one application field of this concept that has been popularized with the development of the IoT is the concept of Channel-based Key Extraction (CBKE) [128]. CBKE approaches aim to bring a solution for a recurrent problem for IoT devices, which is key establishment. As discussed in section IV.1.2.2, key establishment is often a critical part of IoT devices security. Indeed, key establishment protocols are based on asymmetric cryptography, which memory and computational cost are higher than symmetric cryptography. A lot of cheap microcontrollers are able to run symmetric cryptography primitives today, but asymmetric primitives often require higher-end microcontrollers, even though it gets increasingly more accessible. Thus, the security of key establishment is always a problem on devices that do not support asymmetric cryptography.

The idea behind CBKE, shown in **Fig. 52**, is to use the channel secrecy to extract a key even in the presence of an eavesdropper, based on a common observation of a random process [129]. The two nodes that want to establish a key can observe their channel noise; due to the reciprocity of the wireless channel, they should observe the same thing [130]. An eavesdropper that is far enough away has uncorrelated channels with the two nodes, hence, observe a different noise. In practice, this decorrelation distance is proportional to the wavelength. Indeed, a distance in the order of magnitude of a wavelength is often considered as enough to prevent an adversary from learning the secret, although this assumption has been criticized [131]. For reference, the wavelength at a 2.4 GHz frequency is 12.5 cm; typically, a distance of a few decimeter is enough to achieve total spatial decorrelation between two channels [132].

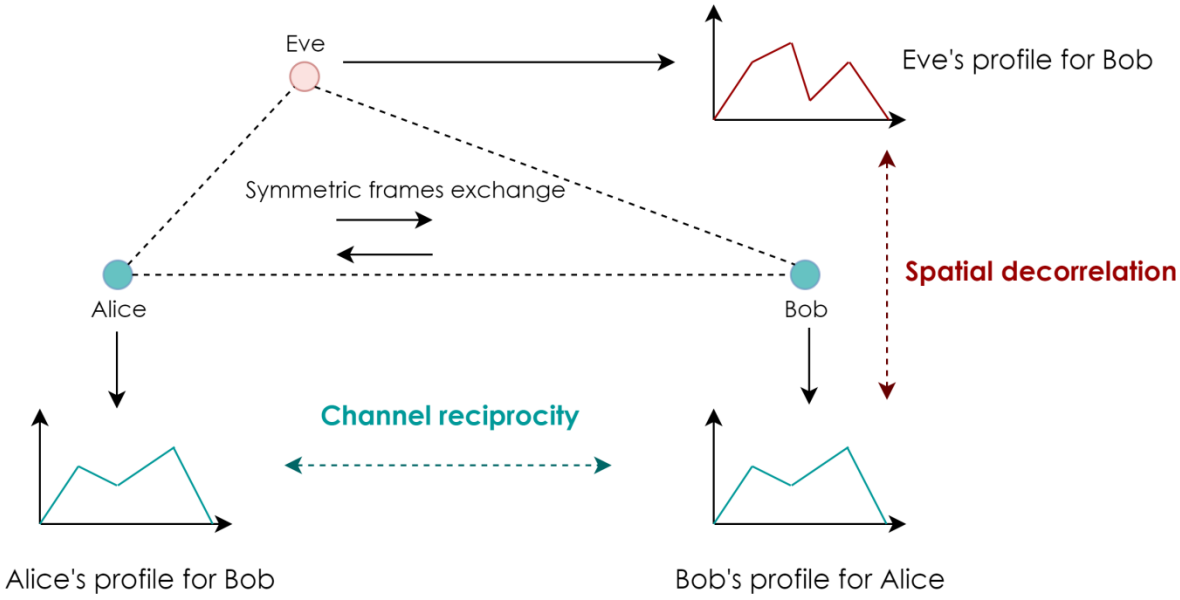


Fig. 52. Principles of Channel-based Key Extraction

There are different requirements to build a CBKE protocol. We assume in the following a CBKE protocol between a node A and a node B. The first thing to consider is that the only way for A or B to characterize their channel is to receive frames from each other. As a consequence, it is impossible for A and B to do a channel observation simultaneously considering that the channel is half-duplex: one node has to transmit and the other one to receive, they cannot do both as the same time.. Hence, they should (1) apply a symmetric protocol, i.e., repeat exactly the same steps on both sides, and (2) keep the delay between their respective observations as short as possible. Typically, this can be achieved with a fast frame exchange in a "Ping-Pong" manner, which is very similar our proposed skew-based authentication protocol. Considering that, the proposed authentication protocol is a good candidate for CBKE, as the key extraction could potentially be implemented on top of the protocol without any active modification. We previously explained that an attacker performing distance tampering could simply shut off the attack during the authentication protocol, in which case the identity of the prover will be validated but not the distance measured. In that particular situation, it might be assumed that the distances were tampered because of a cryptographic flaw, e.g., the AES key of the prover-verifier pair has been compromised. That being considered, the verifier and prover would benefit from extracting a new key from the authentication protocol. Another strong asset of the skew authentication protocol is that it allows the detection of active attackers (i.e., attacker that are interfering and not only listening). CBKE approaches are based on the assumption of passive attackers but are typically not secure against active ones. Hence, integrating a CBKE protocol into the proposed authentication protocol would allow protecting the key establishment protocol from active attackers.

CBKE protocols have been proposed on various wireless technologies, including 802.11 [128], Bluetooth [133] and UWB [134]. There are four main steps in CBKE protocols: channel

probing, quantization, reconciliation and privacy amplification. Channel probing extracts a digitalized profile of one or several given channel parameters. Quantization builds a binary string from this profile. Reconciliation allow the two parties to correct the divergences induced by the measurement noise in their respective quantization outputs, typically by exchanging error correction codes. Finally, privacy amplification allows reducing the information leaked due to the reconciliation process, usually by eliminating the bits that are considered as the most exposed from the extracted string. As a consequence, it is necessary to extract a longer string than actually needed in the quantization process.

In the following, we do not propose a full-implementation of a CBKE protocol, but rather demonstrate the potential of a first path power-based CBKE approach for UWB. We focus mainly on the channel probing process; a simple quantization scheme is proposed to evaluate the performances of the proposed approach in practical settings for an 128-bits key establishment.

VI.4.1 Performance comparison of different observable parameters for channel profiling

The first thing is to define is the parameter(s) that should be observed by the two nodes to characterize the channel noise. Regardless of the radio technology chosen, there are three recurrent candidates proposed in the literature: Channel-Impulse Response (CIR), SNR, and RSSI. UWB being impulse-based, it is considered as a high-potential technology for CBKE [135]. One additional metric that is specifically available on UWB is the First Path Power (FPP).

As the technology is impulse-based, the RSSI is defined directly from the CIR estimate (CIRE) on the DWM1000, and the relation between CIRE and RSSI is linear, which means they are strictly equivalent from a CBKE perspective. CIRE (and *a fortiori* RSSI) is calculated as the average squared magnitude of the CIR samples for a single impulse. This includes both the first path and the multiple reflections of the impulse. Firth Path Power (FPP), on the other hand, is based on the average squared magnitude of the first path CIR samples exclusively. We already discussed how the difference between RSSI and FPP is used as a Line-of-Sight indicator in section III.2.3. Basically, this indicator reflects the quantity of power contained in the secondary paths, which is likely to be higher in an NLoS situation.

In the following, we compare the reciprocity of channel probing for RSSI, SNR and FPP. We did the experiments with both static and dynamic scenarios. In the static scenarios, the two nodes are still, as well as the elements of the surrounding elements. In the dynamic scenarios, one of the nodes was moved slowly toward the other. We also experimented with LoS and NLoS configurations. The CBKE protocol used for the experiments is based a rapid symmetric frame exchange of 2000 frames (1000 on each side) on channel 2, similar to the

skew authentication protocol in **Fig. 50**; the preamble length is set to 64[1] and the payload is empty. The two nodes are measuring the RSSI, SNR and FPP throughout the protocol and report their samples when the protocol is completed. They are placed at 3 m from each other. The CBKE protocol is repeated 10 times with slightly different positions in each scenario. An example of profiles obtained for FPP in the least and most favorable configurations is shown in **Fig. 53**.

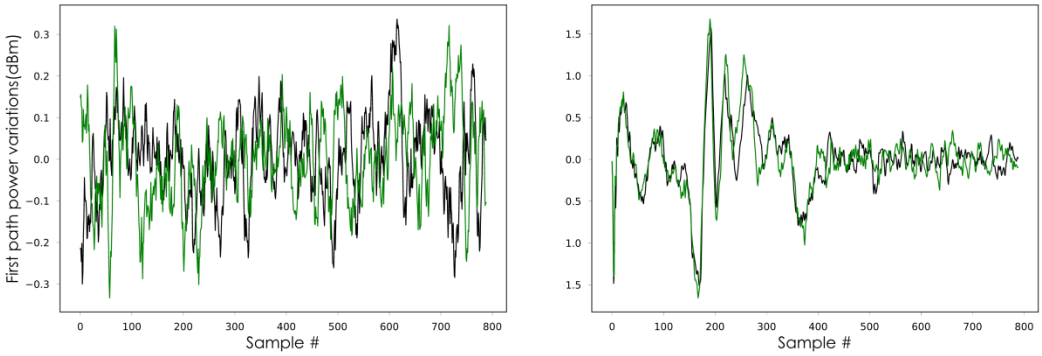


Fig. 53. Examples of FPP profiles obtained by the two parties in the worst (*left)* and best (*right)* configurations; the profiles have been centered on their mean

Table 31. CBKE results for RSSI, SNR and FPP

	Pearson Correlation Coefficient			**Mutual Information (bits)**		
Scenario	**RSSI**	**SNR**	**FPP**	**RSSI**	**SNR**	**FPP**
Static LoS	0.06 *(0~0.13)*	0.12 *(0.02~0.21)*	**0.19** *(0.03~0.29)*	0.09 *(0.07~0.14)*	0.17 *(0.06~0.25)*	**0.19** *(0.08~0.31)*
Dynamic LoS	0.84 *(0.72~0.93)*	0.68 *(0.51~0.84)*	**0.85** *(0.73~0.95)*	1.07 *(0.84 ~1.29)*	0.81 *(0.57~1.05)*	**1.32** *(0.91~1.61)*
Static NLoS	0.33 *(0.07~0.52)*	0.75 *(0.67~0.82)*	**0.90** *(0.81~0.98)*	0.12~0.54	0.85 *(0.71~1.02)*	**1.57** *(0.97~1.92)*
Dynamic NLoS	0.62 *(0.47~0.85)*	0.89 *(0.84~0.92)*	**0.98** *(0.97~0.99)*	0.77 *(0.48~1.03)*	1.13 *(0.99~1.22)*	**2.63** *(1.86~3.47)*

Reciprocity—For each channel parameter, we estimated in **Table 31** the average Pearson correlation coefficient [109] and Mutual Information (MI) [136] in each configuration . A correlation coefficient close to 1 indicates a strong linear relation between the profiles obtained by both parties, whereas a value close to 0 indicates a total decorrelation. Mutual Information basically gives an estimate of the information that one node can extract from a single sample on the sample obtained by the other. It can be seen as the upper bit rate boundary that the quantization and reconciliation process can theoretically reach, assuming a perfectly optimized quantization process. For each scenario, we give the average correlation coefficient and mutual information obtained; the minimum and maximum observed are displayed in parenthesis.

[1] The impact of preamble length on the following results has been independently tested and showed no significant impact on the results. As a consequence, keeping it as short as possible is advisable to minimize the delay between the respective measurements of the two parties.

The worst configuration is the static LoS scenario, as there are little sources of variability in the channel. The variability observed in the node's respective profiles are mostly induced by the measurement noise and not the channel variability, leading to low correlation coefficient and MI. However, it can be observed that introducing motion (dynamic scenarios) or channel degradation (NLoS scenarios) significantly increases these two factors. FPP demonstrates overall the best performances. SNR and FPP follow the same trend, but FPP systematically showcased better performances. This can be explained by the fact that SNR is estimated from the peak value of the first path on DWM1000, which estimation is a slightly random as the 1 GHz sampling frequency only lets the chip collecting 2 or 3 samples of the first path impulse. FPP on the other hand, is based on the 3 samples surrounding the first path and not just the central one, and is less subject to measurement noise. Interestingly, both SNR and FPP tend to perform better in NLoS situations, whereas in dynamic scenarios RSSI is actually better in LoS channels, which might be due to the fact that RSSI is more dependent on the secondary path than the two others. In the most favorable configuration, FPP-based channel probing has an average correlation coefficient of 0.98 and an average MI of 2.63 bits.

Security—One aspect of CBKE security that is rarely discussed in the literature is the capability of an eavesdropper to model the secret profile obtained by the two parties. In the context of a CBKE protocol between Alice and Bob, the capability of an eavesdropper Eve to obtain partial information about the secret key is typically estimated through a straight comparison between the profile obtained by Eve and the ones obtained by Alice and/or Bob. Yet, Eve could try to model Alice and Bob channel based on physical equations. The general consensus is that the complexity of radio channels is such that doing so would be unrealistic in any real-world environment. The fact that channel noise is considered as random actually comes directly from the model complexity. However, that does not mean that parts of the channel components could not be modeled. Notably, we have seen that RSSI-based ranging is commonly used for positioning; this is only possible because there exists a determinist relation between received power and distance. Even if that relation does not capture all the noise induced by interferences and multi-path effect, it still allows characterizing the global trend of RSSI. If Alice is moving toward Bob, it is obvious that the average RSSI will tend to increase; as a consequence, if the quantization scheme is based on a simple threshold comparison, Eve could easily infer that the proportions of '1' will be superior in the second half of the protocol compared to the first one. Considering that RSSI-based CBKE approaches rely on motion to create entropy, this can actually be a significant flaw in terms of randomness properties. This problem can actually be seen from a frequency analysis prospective. The part of the radio channel that can be easily modeled is the one that comes from observable phenomenon in the environment (e.g., Alice moving toward Bob). The channel variations induced by these events are simply low-frequency components of the channel profile. Beyond a given frequency (e.g., vary fast and chaotic drone displacements),

it is unlikely that an eavesdropper would be able to model accurately the channel variations induced by the environment observable dynamic.

As a consequence, we applied Butterworth High Pass Filter (HPF) of second order on the collected FPP profiles to eliminate these potentially vulnerable low-frequency components. We proceeded to check if that approach has an incidence on the reciprocity of the protocol, as it was not excluded that the reciprocity previously observed was stronger on the low-frequency components than on the higher-frequency ones.

The cut frequency of the HPF was set to 10 Hz, which seems to be a reasonable boundary regarding the eavesdropper's modeling capabilities. We focused on FPP profiles collected in dynamic scenarios. For the lowest coefficient correlation (0.73 ~0.8), degradations were observed, with up to 20% of the original mutual information being lost in the filtering process. However, for high correlation coefficients (>0.9), we observed that MI degradations were negligible, as it lies typically below 1%. In terms of security, the effect of applying a HPF can be clearly observed through the autocorrelation curve of the FPP profile. Autocorrelation, by comparing a given signal to a delayed version of itself, quantifies how much future states of a signal can be inferred from its past states. Autocorrelation is part of the NIST randomness tests [137]; a random function should have a low autocorrelation for all lags except 0 (0 corresponding to the comparison of the signal to a non-delayed version of itself, leading to maximized correlation). This is typically estimated through confidence intervals. The correlograms of the FPP profile before and after applying the HPF are shown in **Fig. 54**. The NIST 95% confidence interval boundaries are shown in **red**. To pass the randomness tests, the autocorrelation curve[1] should always be below the confidence interval limit with a tolerance for occasional outsiders (except in the very first lags as they are the most sensitive). The node was moved back and forth when the profile has been characterized. The effect of the motion can clearly be seen in the left correlogram, as a periodic autocorrelation can be observed, which steps outside of the 95% confidence interval. After applying the HPF, this periodic autocorrelation is removed, while the mutual information between the two nodes is preserved. However, we observed that the autocorrelation was still exceeding the 95% confidence interval boundary in the very first lags. On average, we observed that the 95% confidence interval is reached after 5 lags in the collected measurements.

[1] Note that only the positive side of the autocorrelation function is plotted here. The function being symmetric in our case, the negative lags simply mirrors the positive one, hence is not represented.

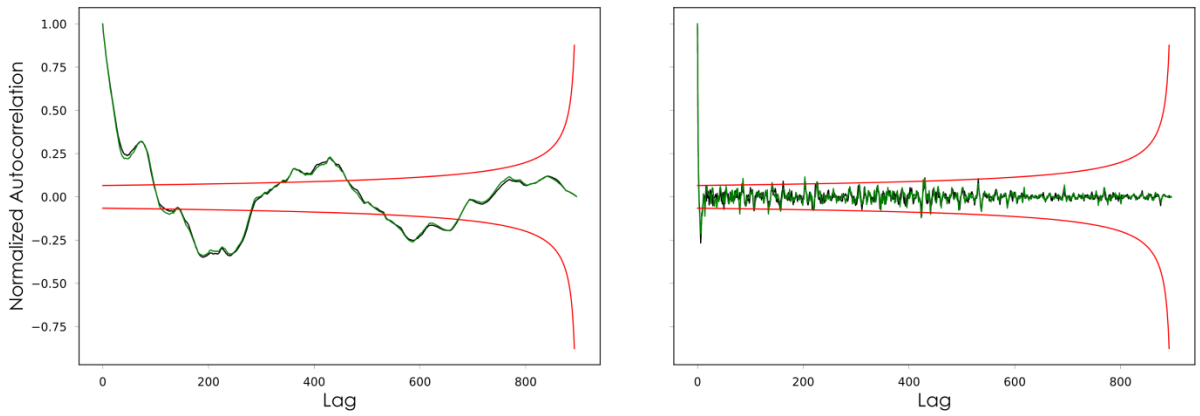


Fig. 54. Correlogram before *(left)* and after *(right)* applying the HPF; Alice and Bob's profiles are shown respectively in green and black, the NIST 95% confidence interval boundary in red

That 5-lags delay can be explained by a second crucial aspect of CBKE security, which is coherence time. Coherence time defines the timespan in which the impulse response of the channel is assumed to stay the same. If two channel measurements occur at an interval that is below the coherence time, these two measurements are correlated. The fact that the reciprocity is strong in dynamic NLoS scenarios shows that the delay between the respective measurements of Alice and Bob (about 550 μs) is below the coherence time of the channel; otherwise, the reciprocity would be tremendously degraded. Yet, if the delay between two measurements of Alice is also below the coherence time, then, these two measurements are correlated and cannot be considered as cryptographically secure. Considering the measurement noise, aggregating several measurements with an averaging or median filter is recommended to clean up the profile measured. However, maintaining a delay between two measurements cluster is essential to maximize the randomness of the process from the eavesdropper's perspective. To reduce the measurement noise, we applied a median filter[1] on clusters of 4 samples. Based on the previous experiments, we estimated the coherence time to a length of 5 samples (i.e., 5 frames on each side), which is equivalent to about 5.5 ms. Rather that applying a passive delay between each cluster (i.e., the nodes stop emitting temporarily), we let the nodes continuing the frames exchanges but exclude from the quantization process the samples collected during the gaps. This is necessary to compute the HPF correctly, as having missing samples between clusters would considerably mess up the process. Also, sampling will be continuous anyway if this approach is used in the skew authentication protocol proposed. After setting the cluster gap to a length of 6 samples, the

[1] We chose median filtering over mean filtering as the performances in terms of both reciprocity and autocorrelation were slightly better. For the sake of readability, we do not dive into the details of this comparison considering the minor impact of the following.

autocorrelation curve is within the 95% confidence interval[1] in both LoS and NLoS scenarios; an example is shown in **Fig. 55**.

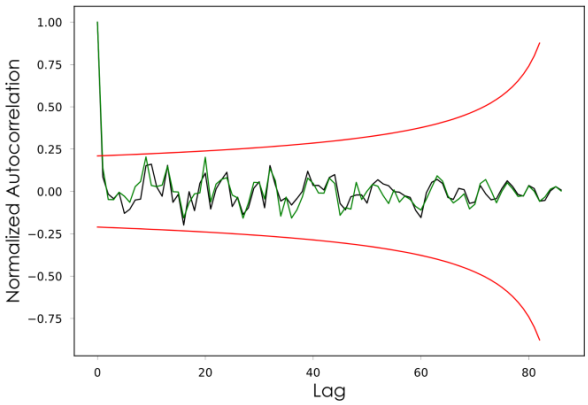


Fig. 55. Autocorrelation with a cluster gap of 6 samples

The FPP profile follows a normal probability distribution. The standard deviation depends on the exact LoS conditions and the intensity of the motion applied, but typically ranges from 2.5 to 6 dBm in NLoS dynamic environments. Hence, FPP-based CBKE showcases good randomness properties for cluster gaps superior to 6.5 ms.

So far, we obtained three main results regarding the proposed FPP-based approach. First, in NLoS (i.e., LOSI > 10 dB) and/or dynamic configurations (i.e., one of the two nodes is moving), good reciprocity can be achieved with Mutual Information ranging from 0.85 to 2.63 bits per channel use. Second, applying a high-pass filter on the collected profiles can considerably reduce the capacity of an attacker to model the profile based on environment observations, without affecting significantly the reciprocity of the protocol. Finally, the distribution of FPP samples features good randomness properties when the profile when small delays are respected in-between each acquisition. When each acquisition is given by the output of a median filter of cluster of 4 samples, and delay of 6 samples are applied between each cluster, the FPP distribution obtained pass the NIST 95% confidence interval autocorrelation test.

Performances against eavesdroppers— We evaluated the information leaked to a nearby eavesdropper in the later experiments. We let two nodes Alice and Bob perform CBKE protocols in the configurations presented in **Table 31**. An eavesdropper Eve was placed 2 cm away from Bob, which is critically close. We monitored the Pearson correlation coefficient and Mutual information obtained by Alice. The worst case obtained is a Pearson correlation coefficient of 0.17 and 0.15 bits of MI, in a LoS dynamic configuration. The average correlation coefficient is 0.07 (i.e., strong decorrelation) and the average mutual information 0.12 bits. Regarding the MI obtained for Eve, it is 8 to 20 times lower than the MI between Alice and Bob based on the results in **Table 31**. This shows that even in very close proximity of attackers the proposed approach achieves decent secrecy.

[1] Note that the confidence interval depends on the length of the signal; shorter signals have higher confidence interval limits as the autocorrelation is inversely proportional to the square root of the signal's length.

Quantization scheme and bit-rate performances— Finally, we applied a basic quantization scheme to evaluate a first real-world estimate of the throughput and bit error rate of FPP-based CBKE. We applied a quantization based on a comparison with two thresholds, which allows extracting 2 bits per cluster. Considering that the profiles are centered, the first threshold is simply 0, giving a '1' if the sample is positive and '0' otherwise. The second threshold is defined as 0.667σ with σ the standard deviation, as for a normal distribution this threshold gives equals probability to get an absolute value above or below (50%). The absolute value of each sample is compared to this second threshold, giving a '1' if it is above and '0' otherwise. A guard area is defined around each threshold; if a sample falls into this guard area, the node classifies the sample as undefined (i.e., the node has low confidence of the quantization of that specific sample). This is basically an anticipation of the privacy amplification process; samples close the thresholds are the most likely to generate errors. The strategy here is that the nodes extract more bits than they actually need, and eliminates later the bits for which one or both of them have low confidence.

The guard area was defined on a width of $\sigma / 8$, with σ the standard deviation of the FPP profile measured, which was the value giving the best results in our first experiments. The bit error rate and undefined bit rate (i.e., the proportion of bits that are classified by either Alice or Bob as undefined) are given below in **Table 32**.

Table 32. Bit Error Rate (BER) and undefined bit rate for various correlation coefficients

Correlation Coefficient	0.6~0.7	0.7~0.8	0.8~0.9	0.9~1
BER	31%	16%	7.5%	1.4 %
Undefined bit rate	41%	36%	33%	29%

The proportion of '0' and '1' (bit balance) was verified systematically, and was always comprised between 48.9 and 51.2% (50.1 % on average). For high correlation coefficients (>0.9), we can observe that the BER can be brought down to 1.4%, with a rate of undefined bits of 29%. For a 128 bits AES-key, this means that extracting about 200 bits would be largely enough to establish the key with basic reconciliation and privacy amplification scheme. Considering that the cluster gap is 12 frames, and that each cluster contains 8 frames (i.e., 4 FPP measurements for each party), the total delay between two channel readings is 11 ms. This gives a theoretical effective bit rate of 116.4 bit/s, leading to a total time for an AES key establishment of 1.1 s.

In conclusion, after applying a basic quantization scheme the protocol can reach a BER of 1.4% in NLoS dynamic conditions while respecting a satisfying bit balance, and allows a potential effective bit rate of 116.4 bits/s. This is potentially enough to achieve key establishment in a single skew authentication protocol. We discuss in the following section the integration of this CBKE approach within the skew authentication protocol previously discussed.

VI.4.2 Discussion and perspectives

One of the constraints regarding integrating a FPP-based CBKE protocol within the skew authentication protocol proposed in section VI.4.2 is the fact that the reciprocity requires both parties to use the same preamble length, as otherwise the frames exchange would not be symmetric. This constraint reduces the temperature gradient created by the skew authentication protocol; as a reminder, the temperature gradient is optimized when short preambles are used on the prover and long preambles on the verifier. Nevertheless, using the same preamble length on both sides does not prevent fundamentally the approach from working, even if the temperature gradient will be lower. Also, because of the coherence time constraints the samples collected by the authentication between each cluster will be ignored as they cannot be used securely for cryptographic operations.

Another huge limitation is that the CBKE protocol will not be functional in LoS static environment, as shown in **Table 31**. A major strength of this approach in an IPS context is that the node could try to pick the anchor with the most favorable CBKE conditions, i.e., the most degraded or the most dynamic channel. This can be easily estimated from the LoS indicator and the FPP standard deviation, which is significantly higher in dynamic settings.[1] In most real-world environments, it is quite unlikely that the node would enjoy a static LoS link with all of the anchors.

The main advantage to combine the proposed skew authentication protocol with CBKE approaches is that it reinforces the security of the CBKE process against active attackers. CBKE protocols are typically only effective against passive attackers; the skew verification scheme allows detecting active attackers interfering and makes the key establishment more secure.

We also plan to explore approaches based on the channel multipath profile, which have been proposed in theoretical works in the literature [135, 138]. Instead of using only the first path (FPP) or the average magnitude of all paths (RSSI), it is possible to use the delay and/or the magnitude of secondary paths as entropy source as suggested in previous theoretical works. This could potentially increase significantly the key establishment bit rate.

[1] Note that dynamicity can simply be induced from moving elements the room, even if the two nodes are still.

Chapter VII Conclusion

We introduced in this manuscript the research work that has been conducted during the PhD towards secure localization with 802.15.4 IR-UWB. Indoor Positioning Systems (IPS) feature unique security challenges due to the fact they exploit radio transceivers as distance sensors. There are numerous IPS applications in which malfunction could lead to significant safety issues (e.g., manufacturing chain supervision). Rather than focusing solely on the security of peer-to-peer ranging protocols, we chose to study IR-UWB IPS security from a system perspective, i.e., by taking into consideration the presence of multiple anchors. The geometrical nature of positioning, along with anchors redundancy, allows indeed to protect the positions integrity at the system-level, even if ranging protocols are compromised.

A special effort has been made to support most of the contributions with real experiments, as we believe it is sometimes missing from previous security works on the topic. We have shown that attacks at the ranging-level can be sometimes easily detected at the positioning-level, based on the consistency, redundancy and plausibility factors. These factors have been benchmarked on SecureLoc based on several state-of-the-art localization methods, such as building a framework to evaluate the stealth of the considered attacks. The criticality of the known vulnerabilities of 802.15.4 IR-UWB has been analyzed from the perspective of indoor positioning systems based on the EBIOS methodology. An in-depth study of the acknowledgment vulnerability has been presented, which discuss the overlap effects involved in Spoofed Acknowledgment Attacks (SAA) and show that they suffer drastically from reply time inconsistencies. Enhanced attack schemes based on SAA have been proposed; we notably showed that using tightly scheduled transmission exposes IR-UWB IPS to precise position tampering attacks. We proposed to induce deliberate inconsistencies in the reply times to secure TWR and LTWR ranging protocols against SAA without replacing acknowledgments or adding cryptography support on them. Two system-level countermeasures against internal attacks have been introduced, respectively based on differential ranging and cooperative verification protocols. We obtained a sensibility of 75 cm for position violations detection with differential ranging. Regarding cooperative countermeasures, we show that an adversary should be able to predict with an accuracy better than 50 cm the positions of all the verifiers to bypass detection; we obtained an average time-to-alarm of 0.98s for a localization frequency of 10 Hz and an adversarial coverage of 30% of the network. Regarding the physical layer, a novel skew-based authentication protocol has been proposed. This protocol induces a temperature stimulus that allows characterizing the unique clock skew response on each chip, and has a TPR of 99.1% for a FPR of 0.16% when the external temperature can be estimated with 2°C accuracy. Finally, we introduced our first results regarding channel-based key extraction; we demonstrated that the proposed approach can potentially allow the establishment of a 128 bits AES key in 1.1 s. This CBKE approach can be integrated within the proposed skew-based

authentication protocol, allowing to potentially achieving both authentication and key establishment.

Regarding the perspectives of this work, one of the major ones is to extend the proposed approaches to decentralized systems. The model assumed in this manuscript is based on the typical IPS approach in the industry, in which the positions are supervised and exploited directly or indirectly by a local infrastructure. In terms of security, this leads to a close network model in which the anchors are more trusted than the tags. Also, the model is infrastructure-centric, which means that the positions are primarily needed and computed by the central server, while in a node-centric model each tag computes by itself is own position. In that case, the tag can often only trust itself and system-level approaches cannot necessarily be applied or must be adapted.

Concerning system-level approaches, we plan to investigate them on larger scales as they largely depend on the node density, network configuration and complexity of the environment. Regarding the physical-layer contributions, the performances of the proposed authentication protocol need to be studied on larger scales and under more complex temperature constraints. We also plan to provide a complete implementation of the proposed CBKE approach and to integrate this approach within the skew-based authentication protocol. At the crossroads of our works on the system and physical layer, we already started to investigate approaches based on Artificial Intelligence (AI) for attack detection. Although IR-UWB is primarily-based on time-of-flight, signal attenuation is also a vector of information regarding location as for any other radio technology. If the accuracy of attenuation-based positioning does not match remotely the performances of time-of-flight based approaches, correlating all the location-related RF indicators can provide a precious security indicator. For example, the RSSI or FPP measured for a malicious node lying on its position will not match its claimed position if the node has not taken care to adjust its transmission power. In practice though, building propagation model that are accurate enough to allow such verifications is very challenging due to the complexity of signal propagation in, for instance, an office environment. Using AI to learn empirically this model is an alternative that has a lot of potential. Promising first results towards such approaches have been obtained during a Master internship in our laboratory, and we plan to investigate more in that direction in the future.

Another problem that arises from the detection methods discussed in the manuscript is the behavior that the system should adopt after detecting an attack on one or several nodes. Indeed, detecting a fraud will inform the central authority that a given positions is fake, but does not inform the system about what the actual position is. In the case of internal attacks, the room for attack mitigation is limited considering that if a node does not want to transmit its position, it can simply remain silent. On the other hand, in the case of the external attack, it could be considered to try extracting the real position even with the malicious intervention of a third-party. When facing external attacks, both differential ranging approaches and cooperative protocols can still extract a certain amount of legit data on the real position of the victim node, even if these data are significantly degraded compared to a non-adversarial

scenario. One perspective regarding further enhancements of these approaches is to identify and filter out the specific data that are corrupted in order to extract the real position. This is another problem for which AI-oriented approaches could bring additional value.

The contributions in this manuscript are based on the IEEE 802.15.4a standard, and new specifications regarding the physical layer have already been defined by the IEEE 4z Group Task for the 4z amendment. In terms of ranging performances, one of the major features brought by the 4z amendment is called the simultaneous ranging mode. The idea is to enable a new multiplexing technic based on preamble overlapping. The preamble sequences defined in 4z allow multiple preambles to overlap, allowing several frame to be sent at the same time as long as the payloads remain separated from each other. This approach allows exploiting the time spent for preambles more efficiently. Regarding security, Dynamic Preamble Selection (DPS), which, as discussed earlier in section II.1.5.1, was quite unanimously considered as a limited security mechanism in 4a/4f, has been removed in 4z. It is replaced by a much more secure dynamic preamble approach: one part of the preamble is encrypted by an AES cipher [8], making spoofed acknowledgements or other attacks requiring static preambles impossible if the attacker does not know the AES key. Preambles are not based on BPPM modulation and are not vulnerable to LC; as a consequence, this feature will automatically prevent from ED-LC attacks. This feature makes possible to authenticate acknowledgments directly at the physical layer. Nevertheless, the computational cost for cryptography will still apply; as a consequence, we believe that the acknowledgment strategy based on unpredictable reply times proposed in this manuscript will remain a valuable lightweight alternative. Regarding internal attacks, it does not seem at this stage that 802.15.4z will adopt specific protocols in order to prevent nodes from lying, which means that protocols proposed in both previous works and this manuscript are still significant for future 4z-based IPS. Moreover, interoperability between 4z and previous generations will be guaranteed, in the case of communications with 4a/4f systems. Also, the current 4a/4f devices will remain on the market and it is unlikely that all the future 4z devices on the market will implement the complex and costly cryptographic mechanisms required for the physical layer. As a consequence, secured 4z devices will still potentially be exposed to spoofed acknowledgment or ED-LC attacks when communicating with 4a/4f devices.

As a final note, it is likely to expect that decentralized and unsupervised positioning approaches will grow as UWB chips should be more widely integrated in smartphones. Compared to more industry-centered applications, these approaches have specific security challenges that remain to be addressed, including on the topic of user privacy. Thus, we firmly believe that the secure positioning research community is going to grow and expand in the next years, to better respond to the new challenges aroused by IoT location services.

References

[1] Ainslie, M. 2010. *Principles of Sonar Performance Modelling*.

[2] Peyton Z. Peebles 1988. *Radar Principles*.

[3] Xia, F., Yang, L.T., Wang, L. and Vinel, A. 2010. Internet of Things. *International Journal of Communication Systems*. 23, 5 (2010), 633–652. DOI:https://doi.org/10.1002/dac.

[4] Pestourie, B., Beroulle, V. and Fourty, N. 2018. Guidelines for the Choice of a Wireless Secure Positioning and Communication System. *Proceedings - 2018 International Workshop on Secure Internet of Things, SIoT 2018* (2018), 1–7.

[5] Standards Committee of the IEEE Computer Society, M. 2016. *IEEE 802.15.3 -2016*.

[6] Standards Committee of the IEEE Computer Society, M. 2016. *IEEE Std 802.15.4™-2015, IEEE Standard for Low-Rate Wireless Networks*.

[7] Decawave DWM1000. Retrieved from: *https://www.decawave.com/product/dwm1000-module/*.

[8] Sedlacek, P., Slanina, M. and Masek, P. *An Overview of the IEEE 802.15.4z Standard and its Comparison to the Existing UWB Standards*.

[9] FiRa Consortium: *https://www.firaconsortium.org/*.

[10] Impact Of COVID-19 on Ultra Wideband (UWB) Market Rapidly Increasing in Size Globally : Latest Report with Current Trends, Future Estimations and Opportunity Analysis: *https://www.marketwatch.com/press-release/impact-of-covid-19-on-ultra-wideband-uwb-market-rapidly-increasing-in-size-globally-latest-report-with-current-trends-future-estimations-and-opportunity-analysis-2020-06-05?mod=mw_more_headlines&tesla=y*. Accessed: 2020-08-26.

[11] Flury, M., Poturalski, M., Papadimitratos, P., Hubaux, J.P. and Le Boudec, J.Y. 2010. Effectiveness of distance-decreasing attacks against impulse radio ranging. *WiSec'10 - Proceedings of the 3rd ACM Conference on Wireless Network Security* (2010), 117–128.

[12] ANSSI 2010. *Expression des besoins et identifications des objectifs de sécurité*.

[13] Rührmair, U., Sölter, J. and Sehnke, F. 2009. On the Foundations of Physical Unclonable Functions. *Cryptology ePrint Archive*. (2009), 1–20.

[14] Yunck, T.P., Liu, C.H. and Ware, R. 2000. A history of GPS sounding. *Terrestrial, Atmospheric and Oceanic Sciences*. 11, 1 (2000), 1–20. DOI:https://doi.org/10.3319/tao.2000.11.1.1(cosmic).

[15] Parkinson, B.W. and Telecom, S. 1996. *Global Positioning System: Theory and Applications, Volume I*. American Institute of Aeronautics & Astronautics.

[16] Galileo: *https://www.gsa.europa.eu/european-gnss/galileo/galileo-european-global-satellite-based-navigation-system*. Accessed: 2020-06-12.

[17] Groves, P.D. 2013. *Principles of GNSS, inertial, and multisensor integrated navigation systems, 2nd edition*. Cambridge University Press.

[18] Flatt, H., Koch, N., Röcker, C., Günter, A. and Jasperneite, J. 2015. A context-aware assistance system for maintenance applications in smart factories based on augmented reality and indoor localization. *IEEE International Conference on Emerging Technologies and Factory Automation, ETFA*. 2015-Octob, (2015). DOI:https://doi.org/10.1109/ETFA.2015.7301586.

[19] Zhao, Z., Fang, J., Huang, G.Q. and Zhang, M. 2016. IBeacon enabled indoor positioning for warehouse management. *2016 4th International Symposium on Computational and Business Intelligence, ISCBI 2016*. (2016), 21–26. DOI:https://doi.org/10.1109/ISCBI.2016.7743254.

[20] Puričer, P. and Kovář, P. 2007. Technical limitations of GNSS receivers in indoor positioning. *2007 17th International Conference Radioelektronika*. (2007), 205–209. DOI:https://doi.org/10.1109/RADIOELEK.2007.371487.

[21] Zanella, A., Bui, N., Castellani, A., Vangelista, L. and Zorzi, M. 2014. Internet of things for smart cities. *IEEE Internet of Things Journal*. 1, 1 (2014), 22–32. DOI:https://doi.org/10.1109/JIOT.2014.2306328.

[22] Mautz, R. 2012. Indoor Positioning Technologies. *ETH Zurich, Department of Civil, Environmental and Geomatic Engineering, Institute of Geodesy and Photogrammetry*. (2012).

[23] IEEE 1969. *IEEE standard definitions of terms for antennas*. IEEE.

[24] Vaghefi, R.M. and Buehrer, R.M. 2014. Cooperative RF pattern matching positioning for LTE cellular systems. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC*. 2014-June, (2014), 264–269. DOI:https://doi.org/10.1109/PIMRC.2014.7136172.

[25] Bilodeau, J.-S., Bouzouane, A., Bouchard, B. and Gaboury, S. 2018. An experimental comparative study of RSSI-based positioning algorithms for passive RFID localization in smart environments. *Journal of Ambient Intelligence*

and Humanized Computing. 9, (2018), 1327–1343.

[26] Torres-Sospedra, J., Moreira, A., Knauth, S., Berkvens, R., Montoliu, R., Belmonte, O., Trilles, S., João Nicolau, M., Meneses, F., Costa, A., Koukofikis, A., Weyn, M. and Peremans, H. 2017. A realistic evaluation of indoor positioning systems based on Wi-Fi fingerprinting: The 2015 EvAAL-ETRI competition. *Journal of Ambient Intelligence and Smart Environments*. 9, 2 (2017), 263–279. DOI:https://doi.org/10.3233/AIS-170421.

[27] Yang, Z., Zhou, Z. and Liu, Y. 2013. From RSSI to CSI: Indoor localization via channel response. *ACM Computing Surveys*. 46, 2 (2013). DOI:https://doi.org/10.1145/2543581.2543592.

[28] Ahmadi, H. and Bouallegue, R. 2017. Exploiting machine learning strategies and RSSI for localization in wireless sensor networks. *International Wireless Communications and Mobile Computing Conference* (2017).

[29] Feldmann, S., Kyamakya, K., Zapater, A. and Lue, Z. *An indoor Bluetooth-based positioning system: concept, Implementation and experimental evaluation*.

[30] Niu, J., Wang, B., Shu, L., Duong, T.Q. and Chen, Y. 2015. ZIL: An Energy-Efficient Indoor Localization System Using ZigBee Radio to Detect WiFi Fingerprints. *IEEE Journal on Selected Areas in Communications*. 33, 7 (Jul. 2015), 1431–1442. DOI:https://doi.org/10.1109/JSAC.2015.2430171.

[31] Hekimian-Williams, C., Grant, B., Liu, X., Zhang, Z. and Kumar, P. *Accurate Localization of RFID Tags Using Phase Difference*.

[32] Kojakian, V., Bloch, C., Chapuis, V., Da Silva, A., Stenta, M., Genon-Catalot, D., Fourty, N., Pestourie, B., Dalce, R., Van Den Bossche, A., Val, T., Vey, Q. and Spies, F. 2019. Firefighter indoor localization (POUCET). *2019 IEEE Radio and Antenna Days of the Indian Ocean, RADIO 2019*. (2019). DOI:https://doi.org/10.23919/RADIO46463.2019.8968931.

[33] Waadt, A.E., Wang, S., Kocks, C., Burnic, A., Xu, D., Bruck, G.H. and Jung, P. 2010. Positioning in multiband OFDM UWB utilizing received signal strength. *Proceedings of the 2010 7th Workshop on Positioning, Navigation and Communication, WPNC'10*. (2010), 308–312. DOI:https://doi.org/10.1109/WPNC.2010.5653193.

[34] Janssen, T., Aernouts, M., Berkvens, R. and Weyn, M. 2018. Outdoor Fingerprinting Localization Using Sigfox. *IPIN 2018 - 9th International Conference on Indoor Positioning and Indoor Navigation*. September 2018 (2018), 24–27. DOI:https://doi.org/10.1109/IPIN.2018.8533826.

[35] Gupta, P. and Kar, S.P. 2015. MUSIC and improved MUSIC algorithm to estimate direction of arrival. *2015 International Conference on Communication and Signal Processing, ICCSP 2015* (2015), 757–761.

[36] Ahmed, A.U., Arablouei, R., de Hoog, F., Kusy, B., Jurdak, R. and Bergmann, N. 2018. Estimating angle-of-arrival and time-of-flight for multipath components using wifi channel state information. *Sensors (Switzerland)*. 18, 6 (2018), 1–18. DOI:https://doi.org/10.3390/s18061753.

[37] Vasisht, D., Kumar, S. and Katabi, D. 2016. Decimeter-level localization with a single WiFi access point. *Proceedings of the 13th USENIX Symposium on Networked Systems Design and Implementation, NSDI 2016*. (2016), 165–178.

[38] Semtech SX1280: *https://www.semtech.com/products/wireless-rf/24-ghz-transceivers/sx1280*. Accessed: 2020-06-20.

[39] Cantón Paterna, V., Calveras Augé, A., Paradells Aspas, J. and Pérez Bullones, M.A. 2017. A Bluetooth Low Energy Indoor Positioning System with Channel Diversity, Weighted Trilateration and Kalman Filtering. *Sensors (Basel, Switzerland)*. 17, 12 (2017). DOI:https://doi.org/10.3390/s17122927.

[40] Fortin-Simard, D., Bouchard, K., Gaboury, S., Bouchard, B. and Bouzouane, A. 2012. Accurate passive RFID localization system for smart homes. *Proceedings - 2012 IEEE 3rd International Conference on Networked Embedded Systems for Every Application, NESEA 2012* (2012).

[41] Ni, L.M., Liu, Y., Lau, Y.C. and Patil, A.P. 2004. LANDMARC: Indoor Location Sensing Using Active RFID. *Wireless Networks* (2004), 701–710.

[42] Han, K. and Cho, S.H. 2010. Advanced LANDMARC with adaptive k-nearest algorithm for RFID location system. *Proceedings - 2010 2nd IEEE International Conference on Network Infrastructure and Digital Content, IC-NIDC 2010*. (2010), 595–598. DOI:https://doi.org/10.1109/ICNIDC.2010.5657852.

[43] Altman, N.S. 1992. An introduction to kernel and nearest-neighbor nonparametric regression. *American Statistician*. 46, 3 (1992), 175–185. DOI:https://doi.org/10.1080/00031305.1992.10475879.

[44] Wang, X., Gao, L., Mao, S. and Pandey, S. 2016. CSI-based Fingerprinting for Indoor Localization: A Deep Learning Approach. *IEEE Transactions on Vehicular Technology*. 66, 7 (Mar. 2016), 763–776.

[45] Lui, G., Gallagher, T., Li, B., Dempster, A.G. and Rizos, C. *Differences in RSSI Readings Made by Different Wi-Fi Chipsets: A Limitation of WLAN Localization*.

[46] Tiemann, J., Schweikowski, F. and Wietfeld, C. 2015. Design of an UWB indoor-positioning system for UAV navigation in GNSS-denied environments. *2015 International Conference on Indoor Positioning and Indoor*

Navigation, IPIN 2015 (2015).

[47] Peng, R. and Sichitiu, M.L. 2006. *Angle of Arrival Localization for Wireless Sensor Networks*.

[48] Schüssel, M. 2016. *Angle of Arrival Estimation using WiFi and Smartphones*.

[49] Huhtala, A., Suhonen, K., Mäkelä, P., Hakojärvi, M. and Ahokas, J. 2007. Evaluation of Instrumentation for Cow Positioning and Tracking Indoors. *Biosystems Engineering*. 96, 3 (2007), 399–405. DOI:https://doi.org/10.1016/j.biosystemseng.2006.11.013.

[50] FAPESP: 2018. *https://agencia.fapesp.br/ultra-wideband-technology-is-used-for-indoor-vehicle-location/27128/*. Accessed: 2020-05-22.

[51] Courtay, A., Le Gentil, M., Berder, O., Scalart, P., Fontaine, S., Carer, A. and Fontaine, S. 2017. Sélection d'ancres pour localisation en intérieur par réseaux radios UWB. (2017), 1–4.

[52] Lazaro, A., Girbau, D. and Villarino, R. 2010. Weighted Centroid Method for Breast Tumor Localization. *Progress In Electromagnetics Research B*. 24, July (2010), 1–15. DOI:https://doi.org/10.2528/PIERB10063004.

[53] Kelley, C.T. 1999. Iterative Methods For Optimization. *Society for Industrial and Applied Mathematics*. (1999), 69–88. DOI:https://doi.org/10.1007/978-94-007-2300-9_4.

[54] Wang, X., Zhang, C., Liu, F., Dong, Y. and Xu, X. 2017. Exponentially weighted particle filter for simultaneous localization and mapping based on magnetic field measurements. *IEEE Transactions on Instrumentation and Measurement*. 66, 7 (2017), 1658–1667. DOI:https://doi.org/10.1109/TIM.2017.2664538.

[55] Luo, J. and Qin, S. 2018. A fast algorithm of simultaneous localization and mapping for mobile robot based on ball particle filter. *IEEE Access*. 6, (2018), 20412–20429. DOI:https://doi.org/10.1109/ACCESS.2018.2819419.

[56] Huang, Y., Zhang, Y., Xu, B., Wu, Z. and Chambers, J.A. 2018. A New Adaptive Extended Kalman Filter for Cooperative Localization. *IEEE Transactions on Aerospace and Electronic Systems*. 54, 1 (2018), 353–368. DOI:https://doi.org/10.1109/TAES.2017.2756763.

[57] Sastry, N. and Wagner, D. 2004. Security considerations for IEEE 802.15.4 networks. *Proceedings of the 2004 ACM Workshop on Wireless Security, WiSe* (2004), 32–42.

[58] Bossche, A. Van Den, Dalce, R., Fofana, N.I., Val, T., Bossche, A. Van Den, Dalce, R., Fofana, N.I., Val, T. and An, D. 2017. DecaDuino : An Open Framework for Wireless Time-of-Flight Ranging Systems. *2016 Wireless Days (WD)* (2017).

[59] Jiang, Y. and Leung, V.C.M. 2007. *An Asymmetric Double Sided Two-Way Ranging for Crystal Offset*.

[60] Standards Committee of the IEEE Computer Society, M. 2006. *IEEE Std 802.15.4-2011, IEEE Standard for Local and metropolitan area networks—Part 15.4: Low-Rate Wireless Personal Area Networks (WPANs)*.

[61] Sewio: *https://www.sewio.net/*. Accessed: 2020-05-27.

[62] Bitcraze: *https://www.bitcraze.io/crazyflie-2/*. Accessed: 2020-04-12.

[63] DecaWave inc. 2017. *DW1000 USER MANUAL*.

[64] Dalce, R., Van Den Bossche, A. and Val, T. 2015. Reducing localisation overhead: A ranging protocol and an enhanced algorithm for UWB-based WSNs. *IEEE 81st Vehicular Technology Conference (VTC Spring)*. (2015), 1–5. DOI:https://doi.org/10.1109/VTCSpring.2015.7146048.

[65] Neirynck, D., Luk, E. and McLaughlin, M. 2017. An alternative double-sided two-way ranging method. *Proceedings of the 2016 13th Workshop on Positioning, Navigation and Communication, WPNC 2016*. October 2016 (2017). DOI:https://doi.org/10.1109/WPNC.2016.7822844.

[66] Kim, H. 2009. Double-sided two-way ranging algorithm to reduce ranging time. *IEEE Communications Letters*. 13, 7 (2009), 486–488. DOI:https://doi.org/10.1109/LCOMM.2009.090093.

[67] Baba, A.I. and Atia, M.M. 2011. Burst mode symmetric double sided two way ranging. *IFIP Wireless Days*. 1, 1 (2011). DOI:https://doi.org/10.1109/WD.2011.6098183.

[68] Dalce, R., Van Den Bossche, A. and Val, T. 2013. Indoor self-localization in a WSN, based on time of flight: Propositions and demonstrator. *2013 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2013*. (2013). DOI:https://doi.org/10.1109/IPIN.2013.6817852.

[69] Dalce, R., Van Den Bossche, A. and Val, T. 2016. A study of the ranging error for Parallel Double Sided-Two way ranging protocol. *IEEE Vehicular Technology Conference*. 0, (2016). DOI:https://doi.org/10.1109/VTCFall.2016.7880884.

[70] Loco Positioning System: *https://www.bitcraze.io/products/loco-positioning-system/*. Accessed: 2020-06-18.

[71] Ledergerber, A., Hamer, M. and D'Andrea, R. 2015. A robot self-localization system using one-way ultra-wideband communication. *IEEE International Conference on Intelligent Robots and Systems*. 2015-Decem, (2015), 3131–3137. DOI:https://doi.org/10.1109/IROS.2015.7353810.

[72] Sharma Vandana and Hussain Muzzamil 2016. Mitigating Replay Attack in WirelessSensor Network Through Assortment of Packets. *Proceedings of the First International Conference on Computational Intelligence and*

Informatics (2016).

[73] Francillon, A., Danev, B. and Capkun, S. 2011. Relay Attacks on Passive Keyless Entry and Start Systems in Modern Cars. *Proceedings of the Network and Distributed System Security Symposium* (San Diego, 2011).

[74] Hancke, G. A Practical Relay Attack on ISO 14443 Proximity Cards.

[75] Li, Z., Trappe, W., Zhang, Y. and Nath, B. 2005. Robust statistical methods for securing wireless localization in sensor networks. *4th International Symposium on Information Processing in Sensor Networks* (2005), 91–98.

[76] Li, P., Yu, X., Xu, H., Qian, J., Dong, L. and Nie, H. 2017. Research on secure localization model based on trust valuation in wireless sensor networks. *Security and Communication Networks*. 2017, (2017). DOI:https://doi.org/10.1155/2017/6102780.

[77] Souissi, I., Ben Azzouna, N. and Ben Said, L. 2019. A multi-level study of information trust models in WSN-assisted IoT. *Computer Networks*. 151, (2019), 12–30. DOI:https://doi.org/10.1016/j.comnet.2019.01.010.

[78] Srinivasan, A., Teitelbaum, J. and Wu, J. *DRBTS: Distributed Reputation-based Beacon Trust System*.

[79] Mabrouki, I. and Belghith, A. 2013. E-SeRLoc: An enhanced serloc localization algorithm with reduced computational complexity. *2013 9th International Wireless Communications and Mobile Computing Conference, IWCMC 2013*. (2013), 153–158. DOI:https://doi.org/10.1109/IWCMC.2013.6583551.

[80] Mauw, S., Smith, Z., Toro-Pozo, J. and Trujillo-Rasua, R. 2018. Distance-Bounding Protocols: Verification Without Time and Location. *Proceedings - IEEE Symposium on Security and Privacy*. 2018-May, (2018), 549–566. DOI:https://doi.org/10.1109/SP.2018.00001.

[81] Cho, J., Yu, J., Oh, S., Ryoo, J., Song, J. and Kim, H. 2017. Wrong Siren! A Location Spoofing Attack on Indoor Positioning Systems: The Starbucks Case Study. *IEEE Communications Magazine*. 55, 3 (2017), 132–137. DOI:https://doi.org/10.1109/MCOM.2017.1600595CM.

[82] Anjum, F., Pandey, S. and Agrawal, P. 2005. Secure localization in sensor networks using transmission range variation. *2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems, MASS 2005*. 2005, (2005), 195–203. DOI:https://doi.org/10.1109/MAHSS.2005.1542800.

[83] Anjum, I.F. and Us, N.J. 2009. Secure wireless user localization using transmission range variation. 2009.

[84] Kuhn, M., Luecken, H. and Tippenhauer, N.O. 2010. UWB impulse radio based distance bounding. *Proceedings of the 2010 7th Workshop on Positioning, Navigation and Communication, WPNC'10*. (2010), 28–37. DOI:https://doi.org/10.1109/WPNC.2010.5653801.

[85] Tippenhauer, N.O., Luecken, H., Kuhn, M. and Capkun, S. 2015. UWB rapid-bit-exchange system for distance bounding. *Proceedings of the 8th ACM Conference on Security and Privacy in Wireless and Mobile Networks, WiSec 2015* (Jun. 2015).

[86] Čapkun, S. 2006. Secure positioning in wireless networks. *IEEE Journal on Selected Areas in Communications*. 24, 2 (Feb. 2006), 221–232. DOI:https://doi.org/10.1109/JSAC.2005.861380.

[87] Van Den Bossche, A., Dalce, R. and Val, T. 2016. OpenWiNo: An open hardware and software framework for fast-prototyping in the IoT. *2016 23rd International Conference on Telecommunications, ICT 2016*. 2016, May (2016). DOI:https://doi.org/10.1109/ICT.2016.7500490.

[88] Van Den Bossche, A., Dalcé, R., Gonzalez, N. and Val, T. 2018. LocURa: A New Localisation and UWB-Based Ranging Testbed for the Internet of Things. *IPIN 2018 - 9th International Conference on Indoor Positioning and Indoor Navigation*. September (2018), 1–6. DOI:https://doi.org/10.1109/IPIN.2018.8533778.

[89] Dalce, R., Van Den Bossche, A. and Val, T. 2014. An experimental performance study of an original ranging protocol based on an IEEE 802.15.4a UWB testbed. *Proceedings - IEEE International Conference on Ultra-Wideband*. (2014), 7–12. DOI:https://doi.org/10.1109/ICUWB.2014.6958942.

[90] Fofana, N.I., Bossche, A. Van Den, Dalcé, R. and Val, T. 2016. Prototypage et analyse de performances d'un système de ranging pour une localisation par UWB. *16ème Colloque Francophone sur l'Ingénierie des Protocoles* (Paris, France, 2016).

[91] OpenWino: *https://wino.cc/*. Accessed: 2020-01-01.

[92] MQTT: *http://mqtt.org/*. Accessed: 2020-06-18.

[93] Ledergerber, A. and D'Andrea, R. 2020. A Multi-Static Radar Network with Ultra-Wideband Radio-Equipped Devices. *Sensors (Basel, Switzerland)*. 20, 6 (2020). DOI:https://doi.org/10.3390/s20061599.

[94] Tomasi, W. 1987. *Advanced electronic communications systems.* Prentice Hall PTR.

[95] Decawave 2014. *APS006 application note: Channel effects on communication range and timestamp accuracy in DWM1000 systems*.

[96] Pestourie, B., Beroulle, V. and Fourty, N. 2019. Open 802.15.4 IR-UWB Modular Platform for Localization Protocols Evaluation. *2019 IEEE Radio and Antenna Days of the Indian Ocean, RADIO 2019* (2019).

[97] Won, S.H.P., Melek, W.W. and Golnaraghi, F. 2010. A kalman/particle filter-based position and orientation

estimation method using a position sensor/inertial measurement unit hybrid system. *IEEE Transactions on Industrial Electronics*. 57, 5 (2010), 1787–1798. DOI:https://doi.org/10.1109/TIE.2009.2032431.

[98] TurtleBot: *https://clearpathrobotics.com/turtlebot-2-open-source-robot/*. Accessed: 2020-06-18.

[99] Jakobsson, M., Perrig, A. and ACM Digital Library. 2004. *Proceedings of the 2004 ACM Workshop on Wireless Security : 2004, Philadelphia, PA, USA, October 01-01, 2004*. ACM Press.

[100] Jung, S.S., Valero, M., Bourgeois, A. and Beyah, R. 2015. Attacking and securing beacon-enabled 802.15.4 networks. *Wireless Networks*. 21, 5 (2015), 1517–1535. DOI:https://doi.org/10.1007/s11276-014-0855-2.

[101] Compagno, A., Conti, M., D'Amico, A.A., Dini, G., Perazzo, P. and Taponecco, L. 2016. Modeling Enlargement Attacks Against UWB Distance Bounding Protocols. *IEEE Transactions on Information Forensics and Security*. 11, 7 (2016), 1565–1577. DOI:https://doi.org/10.1109/TIFS.2016.2541613.

[102] Pestourie, B., Beroulle, V. and Fourty, N. 2019. Security Evaluation with an Indoor UWB Localization Open Platform: Acknowledgment Attack Case Study. *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC* (2019).

[103] Sastry, N. and Wagner, D. 2004. Security considerations for IEEE 802.15.4 networks. *Proceedings of the 2004 ACM Workshop on Wireless Security, WiSec* (2004), 32–42.

[104] Singh, M., Leu, P., Abdou, A.R. and Capkun, S. 2019. UWB-eD: Distance enlargement attack detection in ultra-wideband. *Proceedings of the 28th USENIX Security Symposium* (2019), 73–88.

[105] Amin, Y.M. and Abdel-Hamid, A.T. 2016. Classification and analysis of IEEE 802.15.4 PHY layer attacks. *2016 International Conference on Selected Topics in Mobile and Wireless Networking, MoWNeT 2016* (2016).

[106] Amin, Y.M. and Abdel-Hamid, A.T. 2016. A Comprehensive Taxonomy and Analysis of IEEE 802.15.4 Attacks. *Journal of Electrical and Computer Engineering*. 2016, (2016). DOI:https://doi.org/10.1155/2016/7165952.

[107] More, J.J. and Thuente, D.J. 1994. Line Search Algorithms Sufficient Decrease. *ACM Trasaction on Mathematical Software*. 20, 3 (1994), 286–307.

[108] Cramer, H. 1970. *Random Variables and Probability Distributions*. Cambridge Tracts in Mathmeatics.

[109] Smithson, M. 2002. *Confidence Intervals*. Sage University Paper.

[110] Wang, Y., Ma, X. and Leus, G. 2010. An UWB ranging-based localization strategy with internal attack immunity. *2010 IEEE International Conference on Ultra-Wideband, ICUWB2010 - Proceedings*. 2, (2010), 651–654. DOI:https://doi.org/10.1109/ICUWB.2010.5615052.

[111] Čapkun, S., Rasmussen, K.B., Čagalj, M. and Srivastava, M. 2008. Secure location verification with hidden and mobile base stations. *IEEE Transactions on Mobile Computing*. 7, 4 (2008), 470–483. DOI:https://doi.org/10.1109/TMC.2007.70782.

[112] Casado-Vara, R., Prieto-Castrillo, F. and Corchado, J.M. 2018. A game theory approach for cooperative control to improve data quality and false data detection in WSN. *International Journal of Robust and Nonlinear Control*. 28, 16 (2018), 5087–5102. DOI:https://doi.org/10.1002/rnc.4306.

[113] Shamshirband, S., Patel, A., Anuar, N.B., Kiah, M.L.M. and Abraham, A. 2014. Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks. *Engineering Applications of Artificial Intelligence*. 32, (2014), 228–241. DOI:https://doi.org/10.1016/j.engappai.2014.02.001.

[114] Qin, D., Yang, S., Jia, S., Zhang, Y., Ma, J. and Ding, Q. 2017. Research on Trust Sensing Based Secure Routing Mechanism for Wireless Sensor Network. *IEEE Access*. 5, (2017), 9599–9609. DOI:https://doi.org/10.1109/ACCESS.2017.2706973.

[115] Wang, X., Hao, P. and Hanzo, L. 2016. Physical-layer authentication for wireless security enhancement: Current challenges and future developments. *IEEE Communications Magazine*. 54, 6 (2016), 152–158. DOI:https://doi.org/10.1109/MCOM.2016.7498103.

[116] Rührmair, U., Sölter, J. and Sehnke, F. 2009. *On the Foundations of Physical Unclonable Functions*.

[117] Lanze, F., Panchenko, A., Braatz, B. and Zinnen, A. 2012. Clock skew based remote device fingerprinting demystified. *GLOBECOM - IEEE Global Telecommunications Conference* (2012), 813–819.

[118] Alaca, F. and Van Oorschot, P.C. 2016. Device fingerprinting for augmenting web authentication: Classification and analysis of methods. *ACM International Conference Proceeding Series* (Dec. 2016), 289–301.

[119] Time with focus on NTP and Slovenia: 2004. *http://www.ijs.si/time/*.

[120] Ravikanth, P.S., Benton, S.A. and Instinte, M. 2001. *Physical One-Way Functions*. MIT.

[121] Gassend, B., Clarke, D., Van Dijk, M. and Devadas, S. 2003. Delay-based circuit authentication and applications. *Proceedings of the ACM Symposium on Applied Computing* (2003), 294–301.

[122] Dodis, Y., Ostrovsky, R., Reyzin, L. and Smith, A. 2008. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. *SIAM Journal on Computing*. 38, 1 (2008), 97–139. DOI:https://doi.org/10.1137/060651380.

[123] Schmid, T., Charbiwala, Z., Shea, R. and Srivastava, M.B. 2009. Temperature compensated time synchronization. *IEEE Embedded Systems Letters*. 1, 2 (2009), 37–41. DOI:https://doi.org/10.1109/LES.2009.2028103.

[124] Mann, E.N. 2000. Crystal oscillator with eprom-controlled frequency trim. 2000.

[125] Wang, X., Member, S., Hao, P. and Hanzo, L. 2016. *Physical-Layer Authentication for Wireless Security Enhancement: Current Challenges and Future Developments*.

[126] A.D.Wyner 1975. The Wire-Tap Channel. *Bell System Technical Journal*. (1975). DOI:https://doi.org/https://doi.org/10.1002/j.1538-7305.1975.tb02040.x.

[127] Shannon, C.E. 1948. A Mathematical Theory of Communications. *The Bell Technical Journal*. (1948), 379–423.

[128] Mathur, S., Trappe, W., Mandayam, N., Ye, C. and Reznik, A. 2008. Radio-telepathy: Extracting a secret key from an unauthenticated wireless channel. *Proceedings of the Annual International Conference on Mobile Computing and Networking, MOBICOM*. (2008), 128–139. DOI:https://doi.org/10.1145/1409944.1409960.

[129] Wang, T., Liu, Y. and Vasilakos, A. V. 2015. Survey on channel reciprocity based key establishment techniques for wireless systems. *Wireless Networks*. 21, 6 (2015), 1835–1846. DOI:https://doi.org/10.1007/s11276-014-0841-8.

[130] Patwari, N., Croft, J., Jana, S. and Kasera, S.K. 2010. High-rate uncorrelated bit extraction for shared secret key generation from channel measurements. *IEEE Transactions on Mobile Computing*. 9, 1 (Jan. 2010), 17–30. DOI:https://doi.org/10.1109/TMC.2009.88.

[131] Edman, M., Kiayias, A., Tang, Q. and Yener, B. 2016. On the Security of Key Extraction From Measuring Physical Quantities. *IEEE Transactions on Information Forensics and Security*. 11, 8 (2016), 1796–1806. DOI:https://doi.org/10.1109/TIFS.2016.2543687.

[132] Honma, N. and Murata, K. 2020. Correlation in MIMO antennas. *Electronics (Switzerland)*. 9, 4 (2020). DOI:https://doi.org/10.3390/electronics9040651.

[133] Premnath, S.N., Gowda, P.L., Kasera, S.K., Patwari, N. and Ricci, R. 2014. Secret key extraction using Bluetooth wireless signal strength measurements. *2014 11th Annual IEEE International Conference on Sensing, Communication, and Networking, SECON 2014*. (2014), 293–301. DOI:https://doi.org/10.1109/SAHCN.2014.6990365.

[134] Habib, S.M., Ries, S. and Max, M. 2010. Secret key generation exploiting Ultra-wideband indoor wireless channel characteristics. *Security and Communication Networks*. December 2014 (2010), 1–18. DOI:https://doi.org/10.1002/sec.

[135] Wilson, R., Tse, D. and Scholtz, R.A. 2007. Channel identification: Secret sharing using reciprocity in ultrawideband channels. *2007 IEEE International Conference on Ultra-Wideband, ICUWB*. 2, 3 (2007), 270–275. DOI:https://doi.org/10.1109/ICUWB.2007.4380954.

[136] Kraskov, A., Stögbauer, H. and Grassberger, P. 2004. Estimating mutual information. *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*. 69, 6 (2004), 16. DOI:https://doi.org/10.1103/PhysRevE.69.066138.

[137] NIST randomness tests: *https://www.itl.nist.gov/div898/software/dataplot/refman1/auxillar/autoband.html*. Accessed: 2020-07-20.

[138] Madiseh, M.G., McGuire, M.L., Neville, S.W. and Shirazi, A.A.B. 2008. Secret key extraction in ultra wideband channels for unsynchronized radios. *Proceedings of the 6th Annual Communication Networks and Services Research Conference, CNSR 2008*. (2008), 88–95. DOI:https://doi.org/10.1109/CNSR.2008.52.

[139] Blanchard, Y. 2006. Le centenaire de l'invention du radar par Christian Hülsmeyer, retour sur un anniversaire. *Revue de l'Electricité et de l'Electronique*.

[140] GPS Accuracy: *https://www.gps.gov/systems/gps/performance/accuracy/*. Accessed: 2020-06-22.

Index of abbreviations

AES	Advanced Encryption Standard
AFE	Analog Front-End
AoA	Angle of Arrival
BPPM	Binary Pulse Position Modulation
CBKE	Channel-based Key Extraction
CIR	Channel Impulse Response
CSI	Channel State Information
DB	Distance-bounding
FPP	First Path Power
GMDRE	Global Maximum Differential Ranging Error
GN	Gauss-Newton (Algorithm)
LDE	Leading Edge Detection
LOSI	Line-of-Sight Indicator
LTWR	Lightweight Two-Way Ranging
MAC	Medium Access Layer
MDRE	Maximum Differential Ranging Error
MITM	Man-in-the-middle
MRD	Maximum Realistic Distance
NLoS	Non Line-of-Sight
PF	Particle Filter
PHR	Physical Header
PRF	Pulse Repetition Frequency
PUF	Physical Unclonable Function
RD	Ranging Deviation
ROC	Receiver Operating Characteristics
RSSI	Received Signal Strength Indicator
RTLS	Real-time Localization System
SAA	Spoofed Acknowledgment Attack
SFD	Start Frame Delimiter
TADRE	Total Average Differential Ranging Error
TDMA	Time Division Multiplexing Access
ToF	Time-of-Flight
TRD	Total Ranging Deviation
TWR	Two-Way Ranging
VM	Verifiable Multilateration
WCL	Weighted Centroid-based Localization
WSN	Wireless Sensors Network

List of figures

List of tables