SecNav: Secure Broadcast Localization and Time Synchronization in Wireless Networks

Kasper Bonne Rasmussen
Department of Computer Science
ETH Zurich
8092 Zurich, Switzerland
kasperr@inf.ethz.ch

Srdjan Capkun
Department of Computer Science
ETH Zurich
8092 Zurich, Switzerland
capkuns@inf.ethz.ch

Mario Cagalj
FESB
University of Split
21000 Split, Croatia
mario.cagalj@fesb.hr

Abstract

We propose SecNav, a new protocol for securing wireless navigation systems. This protocol secures localization and time-synchronization in wireless networks by relying on devices' *awareness of presence* in the power-range (coverage area) of navigation stations. We perform a detailed security analysis of SecNav and show that, compared to existing secure navigation approaches, it prevents the widest range of attacks on navigation. Our implementation of SecNav, using 802.11b devices, shows that this scheme can be efficiently implemented with existing technologies.

1 Introduction

The use of location and time information in wireless networks is broad and ranges from enabling networking functions (i.e., position-based routing) to enabling applications (e.g., location-based access control, data harvesting, emergency and rescue). Researchers have therefore proposed a number of positioning [56, 57, 37, 1, 19, 4, 32] and time synchronization [47, 9, 12, 31, 48, 43, 9, 20] techniques for wireless networks, based on a wide range of technologies, including measurements of the strength and time of propagation of radio and ultrasonic signals.

Recently, researchers have shown that localization and time-synchronization techniques are highly vulnerable to signal manipulation attacks [58, 24, 53, 44, 13]. To cope with this problem, a number of solutions were proposed, some relying on bidirectional communication between the infrastructure and the nodes [26, 55, 53, 41, 13, 30, 45, 46], and some on unidirectional (broadcast) navigation signals emitted by the infrastructure [25, 24]. Bidirectional communication between the infrastructure and devices helps in reducing the set of possible attacks on localization and time synchronization, notably, through the use of security primitives like distance-bounding [2, 17, 41, 50, 33], authenticated ranging [55] and delay estimations [13]; these primitives can be used to efficiently prevent pulse-delay attacks on synchronization and signal replay attacks on localization.

In broadcast-based navigation schemes, however, such bi-directional primitives cannot be used and these scheme are therefore highly vulnerable to attacks based on navigation signal replays. Range-free secure localization scheme by Lazos et al. [25], is vulnerable to selective signal replay attacks, if jamming of navigation beacons cannot be detected by the localized devices. This problem was partially addressed by Kuhn in [24] in the context of securing range-based navigation, where the replay of individual navigation signals is prevented by a late disclosure of signal spreading codes. However, this solution is vulnerable to replays of aggregated navigation signals.

In this work, we propose SecNav, a novel secure navigation protocol, based on navigation signal broadcasts, which does not require bidirectional communication between the infrastructure and navigation devices. We show that this protocol prevents a wide range of attacks on localization and time synchronization, including location spoofing attacks using aggregated signal replays. SecNav relies on integrity coding [49] of navigation signals and on devices' awareness of their presence in the coverage area of navigation stations (e.g., within a building, university campus, or a city). We show how this coding prevents message manipulation attacks and protects the integrity and the authenticity of transmitted navigation messages. We further show how the requirement of devices' and/or users' awareness of presence in the (wider) coverage area of the infrastructure can be efficiently ensured in a number of applications. To the best of our knowledge, SecNav is also the first secure broadcast-based time synchronization system for local-area and sensor networks.

We propose two instances of SecNav: SecNav-R, which secures range-based navigation, and SecNav-F, which secures range-free navigation. Our implementation of SecNav-F using 802.11b shows that this scheme can be efficiently implemented using available technologies. Compared to SecNav-R and to distance-bounding-based secure localization approaches, SecNav-F is lightweight and does not require any high-speed processing hardware. However, given that SecNav-F is a range-free navigation scheme, it generally does have a lower accuracy than range-based schemes.

The application domain of SecNav is wide; this system can be effectively used for secure in-door and outdoor localization and synchronization of individual wireless devices, whose communication is supported by an

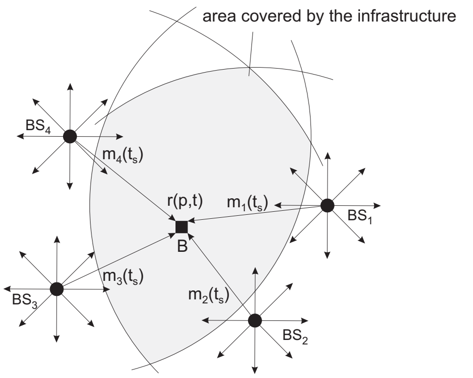


Figure 1. Broadcast Navigation: Navigation stations (BS) broadcast navigation messages ($m_i(t_s)$), based on which a receiver (B) determines its location and correct time reference. A set of locations from which navigation messages can be heard forms the infrastructure coverage area.

infrastructure, but equally for localization and synchronization in multi-hop sensor and ad-hoc networks. Although intended primarily for smaller local environments (e.g., company buildings, university campuses), with appropriate technology and legislation in place, SecNav can be equally used in larger areas.

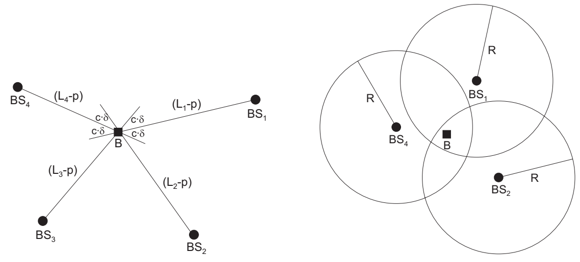
The rest of this paper is organized as follows. In Section 2, we describe our system and the attacker models and we state the observed problem. In Sections 3 and 4, we describe our secure navigation schemes. In Section 5, we present a security analysis of SecNav. In Section 6, we describe the robustness and implementation of SecNav and related measurement results. In Section 7, we present the related work. Finally, we conclude the paper in Section 8.

2 System Model and Problem Statement

Before stating our problem, we first describe the observed system.

2.1 System model

Our system consists of a set of stations forming a navigation infrastructure which provides radio signals that enable devices to determine their location and to obtain an accurate time reference. We assume that the stations are strategically located such that they *cover* a given physical space (e.g., a university campus). Here, we consider that a point in space is covered by the infrastructure if it is within the communication range of at least four infrastructure stations. We further assume that the navigation infrastructure is under the control of an authority and that the stations are protected such that they cannot be compromised by an adversary. Each navigation device is aware that there is at least one honest navigation infrastructure that covers the space in which it resides; otherwise, little can be done to enable secure navigation. This awareness is achieved through public authenticated knowledge (e.g., owners of devices are made aware of the presence of the infrastructure by local civil authorities). We note that the adversary *is not* prevented from setting-up her own navigation infrastructure covering the same space covered by the legitimate infrastructure. We observe two types of broadcast navigation systems: range-based and range-free localization systems. We first describe the range-based localization system.



(a) (b)

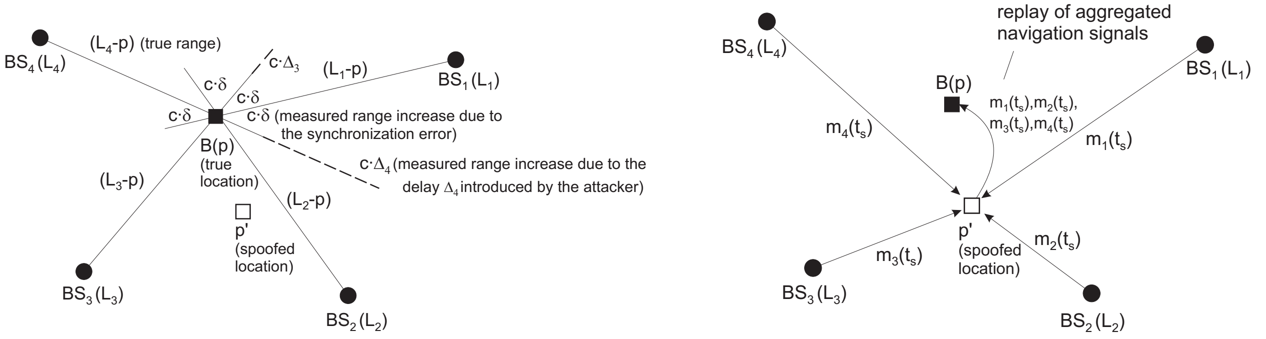
Figure 2. (a) Range-based navigation: B determines its locations and time reference by measuring pseudo-ranges, which consist of true ranges $|L_i - p|$ between B and BS_i and of a ranging error $c \cdot \delta$ caused by an offset between B's clock and clocks of navigation stations. (b) Range-free navigation: B estimates its location within the intersection of power ranges (R) of navigation stations, whose beacons it hears. B synchronizes to the infrastructure by observing the timestamps contained in navigation messages.

2.1.1 Range-based localization

Here, we consider navigation systems that have the same or similar mode of operation as the Global Positioning System (GPS) [14]. This means that in these systems, stations emit navigation signals, based on which navigation devices determine their location and time reference. Like in GPS, we assume that navigation stations are tightly synchronized and emit navigation signals simultaneously (up to a measurable drift). Each navigation signal s_i, contains a timestamp t_s of the time at which it was sent and a location L_i of the base station BS_i that sent it. Upon collecting at least four signals and registering their reception times, the navigation device calculates the distances to the stations, and determines its location p and time reference by multilateration. This is illustrated on Figure 1. The cumulative signal observed at the navigation device at time t is given by the following expression:

$$r(p,t) = \sum_i A_i(p,t) \cdot s_i(t_s - \frac{|L_i - p|}{c} + \delta) + n(\mathbf{p},t) \qquad (1)$$

where $A_i(p,t)$ and $n(p,t)$ are the strength of the signal s_i and the noise at location p and time t, respectively; δ is the de-synchronization error between the device and the navigation stations, and c is the speed of light in vacuum. Upon the reception of a navigation signal from station BS_i, the device registered its reception time t_r^i,



(a) (b)

Figure 3. Examples of attacks on localization: (a) Pulse-delay attack. Navigation messages are delayed (i.e., by Δ_3 and Δ_4) by the attacker, causing an increase of measured ranges and the computation of a spoofed location p' by the device B; (b) Replay of aggregated navigation signals. Navigation messages from location p' are relayed to the device B (at location p), which then believes that it is located at p'.

from which it computes a pseudo range $\hat{d_i}$ to BS_i as

$$\hat{d_i} = (t_r^i - t_s) \cdot c \qquad (2)$$

Each pseudo-range contains (the same) error $c \cdot \delta$ introduced by the offset δ between the device's and stations' clocks. By measuring pseudo-ranges to (at least) four stations, the device can determine it's location p and the synchronization offset δ and therefore synchronize to the stations. This is done by solving (for p and δ) the following system of (at least four) equations

$$\hat{d_i} = |L_i - p| + c \cdot \delta \qquad (3)$$

where each equation corresponds to one pseudo-range $\hat{d_i}$ measured by B to station BS_i. This is illustrated on Figure 2(a).

2.1.2 *Range-free localization*

We further consider range-free broadcast navigation systems. These systems are similar to range-based localization in that the navigation device determine its location and synchronizes to the infrastructure based on the messages that it receives from navigation stations. The main difference is that, instead of measuring distances to the stations, the device simply registers from which stations it received the messages and then estimates its location within the area defined by the intersection of the power ranges of navigation stations. This is illustrated on Figure 2(b). Examples of range-free localization schemes include the proposals of He et al. in [18] and Lazos et al. in [25]. Similarly, the device synchronizes to the infrastructure by simply adjusting its clock to the timestamp contained in the received beacons. One example of time synchronization using reference broadcast is described in [9].

2.2 Attacker model

We adopt the following attacker model. We assume that the attacker Mallory (M) controls the communication channel in a sense that she can eavesdrop messages, insert messages, modify and schedule transmitted messages. More specifically, we assume that the attacker can relay and delay transmitted messages. We do assume that the attacker cannot disable the communication channel between infrastructure nodes and navigation devices (e.g., bu using a Faraday cage to block the propagation of radio signals). However, the attacker can jam all transmissions and in that way prevent the transmission of the information contained in the message; the receiver will therefore still receive the message from the sender, superimposed by the attacker's messages. Our attacker model is similar to the the Dolev-Yao model [8] in that the attacker controls the communication channel, but it differs in that the attacker cannot trivially remove the energy of emitted signals from the channel, especially if these signals are unpredictable. We detail this in Section 5.2.

2.3 Attacks on navigation systems

Main security threats to navigation systems are caused by the **forgery** and **replay** of navigation signals. If signals can be forged by the adversary, she can present navigation devices with a set of signals corresponding to *any* location and time. With appropriate message authentication and integrity protection mechanisms, message forgery can be prevented. However, even with signal authentication, navigation systems remain insecure due to possible signal replay attacks (which cannot be prevented using traditional authentication and integrity protection mechanisms). Specifically, in systems based on time-of-arrival signals can be relayed and delayed by the attacker. The simplest form of message replay attack is the **pulse-delay attack** [53, 13]. In this attack, the attacker registers the original time-stamped signal s_i sent by the infrastructure station and replays it to the attacked receiver, but with a delay Δ_i (in some scenarios, for this attack to succeed the attacker also needs to jam or **overshadow** the original signal). Here, by signal overshadowing we mean that the original message will appear as noise in the attacker's (much stronger) signal. The computed pseudo-range at the receiver will therefore be artificially increased by $c \cdot \Delta_i$ and will be computed as follows.

$$\hat{d_i} = |L_i - p| - c \cdot \delta + c \cdot \Delta_i \tag{4}$$

If all (four) signals are appropriately delayed by the attacker, the device will estimate its location at a spoofed location p'. This is illustrated on Figure 3(a). Pulse-delay attacks have particularly severe impact on localization techniques based on time-of-arrival (TOA) and on time-difference-of-arrival (TDOA) techniques.

Another form of signal replay attack is the **replay of aggregated navigation signals** obtained from other locations. In this attack, the attacker creates a fast wormhole [21] between the location which it wants to convince the device of, and the actual location of the attacked device. The relayed signal will be stronger than the original navigation signal at devices' true location (i.e., it will **overshadow** the original signal) and will therefore make the device believe that it is located at the location from which the signal is relayed. This attack is illustrated on Figure 3(b).

Several solutions have been proposed to prevent replay attacks, based on signal spreading [24] and on authenticated ranging or distance-bounding [54, 53]. Solution based on signal spreading prevents pulse-delay attacks, but it is vulnerable to replays of aggregated navigation signals. Solutions based on authenticated ranging/distance bounding prevent both attacks, but require bi-directional communication between the infrastructure and the receivers.

2.4 Problem statement

Now we state our problem: *How can a device B, securely determine its location and time reference in the presence of an attacker M, based on signals received by the infrastructure?.* Note that in solving this problem, we focus on the above described localization systems, in which devices compute their locations and time reference based on signals emitted by the navigation infrastructure. We therefore consider scenarios in which localization and synchronization are performed passively by the devices (i.e., devices do not emit any signals in order to determine their location or to synchronize with the infrastructure).

3 SecNav-R: Secure Range-based Broadcast Navigation

In this section, we describe SecNav-R, a novel system for securing localization and time synchronization in broadcast navigation systems. SecNav-R is based on time-of-flight measurements and, in terms of location and time computations, operates as described in Section 2.1.1. In SecNav-R, navigation stations are therefore mutually synchronized, cover a given physical space, and transmit navigation signals containing station locations and timestamps. Devices that reside in the station's coverage area collect navigation signals, determine pseudo-ranges, and process them in real-time to obtain their locations and time reference. In this respect, SecNav is similar to other existing navigation systems. However, what makes SecNav-R significantly different is the fact that the navigation signals emitted by the stations are specifically encoded using *integrity-codes* [49] to eliminate the threat of replay attacks; integrity codes consist of Manchester coding and on-off keying on the physical layer that also enable straightforward detection of message overshadowing attacks. Besides integrity codes, navigation messages in SecNav-R are also protected using digital signatures, which prevent message forging attacks.

In the following section, we describe the process of encoding of the navigation signal in SecNav-R.

3.1 Signal Encoding

We explain the process of integrity-coding of navigation signals through an example shown in Figure 4. Notice that in order to avoid confusion we refer to the ones and zeros of the original message as *bits* and the ones and zeros of the Manchester encoded message as *symbols*. In this example base station BS_i wants to transmit a navigation message $m_i(t_s) = BS_i \| t_s \| L_i$, containing its identifier BS_i, message sending time t_s and its location L_i to navigation devices in its vicinity. Before sending the message, BS_i first appends it with the message signature $sig_{K_N}\{m_i(t_s)\}$, generated with the infrastructure private key K_N. Before emitting $m_i(t_s), sig_{K_N}\{m_i(t_s)\}$ over a radio channel, BS_i transforms this message as follows: it applies Manchester (*complementary*) encoding rule to $m_i(t_s), sig_{K_N}\{m_i(t_s)\}$ that is, each bit "1" of $m_i(t_s)$ is encoded as 10 and each bit "0" as 01. The resulting message is denoted with $c_i(t_s)$ in Figure 4. Manchester coding ensures that resulting message $c_i(t_s)$ consists of equal number of symbols "1" and "0". Finally, in order to transmit $c_i(t_s)$ over a radio channel, BS_i uses on-off keying modulation at the physical layer. Thus, for each symbol "1" of $c_i(t_s)$, the sender emits a random waveform during the *symbol period* T (a fresh random waveform is generated for each symbol). For each symbol "0" of $c_i(t_s)$, the sender is silent (does not emit any signals) during a period T (Figure 4). Here, the transmitted waveforms do not carry any information, but it is the *presence* or *absence* of signal energy in a given time slot of duration T that conveys information.

In order to retrieve the transmitted message, the navigation device (B) simply measures the energy in the corresponding time slots of duration T. Let P_r denote the average power that the receiver measures in a given time slot of duration T. Let us also denote with P_0 and P_1 pre-defined *threshold power levels.* Here, $P_1 \geq P_0$. For the given time slot, the receiver B decodes the received signals as follows:

1. if $P_r \geq P_1$, output symbol "1"
2. if $P_r \leq P_0$, output symbol "0"
3. else reject.

Here, the receiver listens on the predefined channel and for each time slot of duration T it applies the above decoding rule to obtain message $c_i(t_s)$. In section 6, we show how that this encoding and decoding can be successfully performed using 802.11b technology. Finally, the receiver uses the inverse of Manchester encoding rule

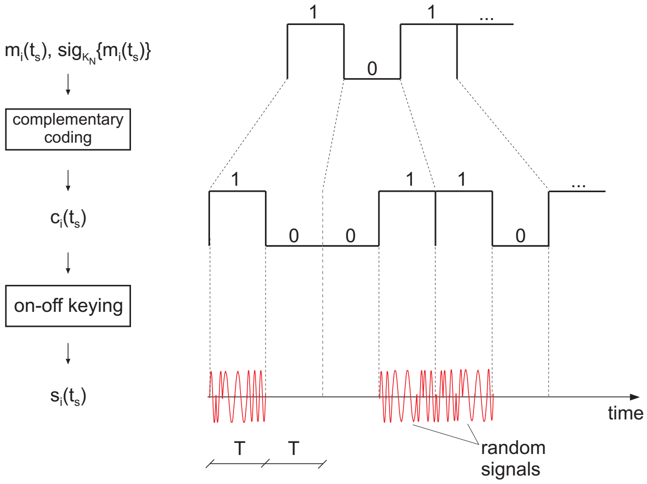


Figure 4. SecNav navigation message encoding. The navigation message is first encoded using Manchester coding resulting in message $c(t_s)$, which is then transmitted on the wireless channel using on-off keying (signal $s(t_s)$). On-off keying is implemented such that for each "1" of $c_i(t_s)$, the station emits a random waveform during the *symbol period* T (a fresh random waveform is generated for each symbol), and for each symbol "0" of $c_i(t_s)$, the sender is silent (does not emit any signals) during the period T.

(i.e., 01 → 0, 10 → 1) to retrieve the navigation message $m_i(t_s)$.

The protection of the navigation signal $m_i(t_s), sig_{K_N}\{m_i(t_s)$ here comes from the fact that simultaneous presence of two different I-coded messages $m_i(t_s)$ and $\hat{m}_i(t_s) \neq m_i(t_s)$ in the same area necessarily results in an incorrectly demodulated message at a receiver. Thus, an adversary, in order to change $m_i(t_s)$ into a fake message $\hat{m}_i(t_s) \neq m_i(t_s)$, has to change at least one bit of $m_i(t_s)$ (i.e., $\hat{m}_i(t_s)$ differs in at least one bit). This implies that the corresponding $c_i(t_s)$ and $\hat{c}_i(t_s) \neq c_i(t_s)$ will differ in at least two symbols. Moreover, at least one symbol "1" of $c_i(t_s)$ has to be converted into "0" in $\hat{c}_i(t_s)$. In other words, the adversary has to annihilate (cancel out) the waveform representing a symbol "1" of $c_i(t_s)$, otherwise the receiver cannot correctly demodulate the message received at the physical layer and it will simply reject it. By appropriately crafting waveforms representing symbols "1" (e.g., by making these waveforms random), the task of canceling them out can be made arbitrarily hard for the adversary.

Digital signatures make it even more difficult for the attacker to modify navigation messages. In the presence of digital signatures, the attacker can only attempt to convert the original message $m_i(t_s)$ into a message $\hat{m}_i(t'_s < t_s)$ that was already sent by the station in the past, and cannot forge a new message. This is important especially for secure time synchronization, as it prevents that the clocks of navigation devices are shifted ahead by the attacker; digital signatures cannot, however, prevent that the local clocks of the devices are shifted back in time. Besides adding to the security of navigation systems, digital signatures in SecNav add to the robustness of the message transmission. Signatures act as redundancy checks for the messages, and can also be used for message reconstruction, if local interference modifies messages in transmission (e.g., turns symbols 0 to 1).

Note here that we assume the navigation signal of BS_i to "always" be present in the observed area. Otherwise, an adversary could easily insert his/her fake navigation message. We elaborate further on this in Section 3.3.

To verify the integrity and authenticity of the demodulated navigation message $m_i(t_s)$, the receiver needs to

(i) verify that it resides in the infrastructure coverage area

(ii) verify that the channel on which it received the signal $s_i(t_s)$ is the channel used by the infrastructure

(iii) verify that the demodulated message $c_i(t_s)$ is valid, i.e., it contains an equal number of symbols "1" and "0"

(iv) verify that the demodulated signature $sig_{K_N}\{m_i(t_s)\}$ correspond to the demodulated message $m_i(t_s)$

If these conditions are fulfilled, device B concludes that the navigation message $m_i(t_s)$ is authentic and has been transmitted by the navigation station BS_i. Conditions (i) and (ii) are generally fulfilled by dissemination of public information; namely, the wider area that the infrastructure covers and the channels that the stations use can be made publicly available (or disseminated) by a trusted authority. Condition (iv) is fulfilled by appropriate dissemination of the infrastructure public key. Condition (iii) therefore remains the most important criterion for the verification of message authenticity and integrity. As we argued at the end of the previous section, this condition ensures that if the Manchester encoded message $c_i(t_s)$ contains an equal number of symbols "1" and "0", then it has not been modified in transmission. This is due to the on-off keying modulation and signal anti-blocking property which prevent "1" symbols from being flipped, and enable the detection of signal overshadowing attacks.

3.2 Computing the Location and Time Reference

Upon the reception of the navigation message, the navigation device registers the message reception time t^i_r (the reception time of the first symbol of the message), verifies its authenticity and integrity (as described earlier) and extracts from the message its sending time t_s. From four message sending and reception times, the sta-

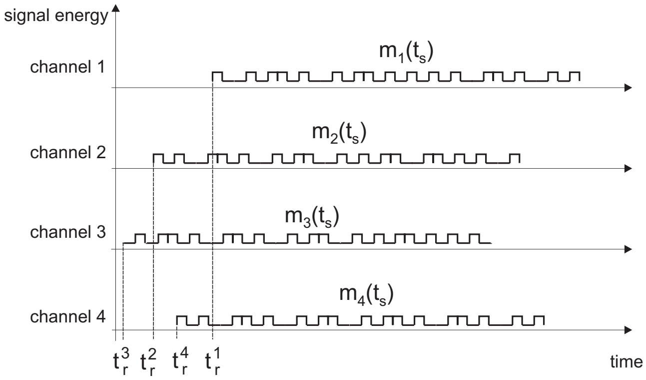


Figure 5. Navigation message reception. Example diagram of message reception at a navigation device. Each message was transmitted by a different navigation station, at the same time (t_s). Messages are then received at times $t_r^1, t_r^2, t_r^3, t_r^4$ by the navigation device. Based on the sending and reception times, the device then computes the current time and its location.

tion computes four pseudo-ranges.

$$\begin{aligned}(t_r^1 - t_s)\cdot c &= |L_1 - p| + c\cdot\delta\\(t_r^2 - t_s)\cdot c &= |L_2 - p| + c\cdot\delta\\(t_r^3 - t_s)\cdot c &= |L_3 - p| + c\cdot\delta\\(t_r^4 - t_s)\cdot c &= |L_4 - p| + c\cdot\delta\end{aligned}$$

By solving this system of equations, the station computes its location p and the time difference δ between its clock and the clocks of the stations. Here, t_r^i are the navigation signal reception times, L_i are the locations of navigation stations and c is a speed of light in vacuum. An example diagram of message reception times at the receiver is shown on Figure 5. Here, we assume that, similarly to GPS receivers [14], navigation devices in our system can receive navigation signals simultaneously on at least four channels (one channel for each navigation station).

Thus far, we have observed that each station BS_i transmits a single navigation signal $s_i(t_s)$ at time t_s. However, in our system, the absence of legitimate navigation signals in the infrastructure coverage area would enable an attackers to insert messages and provide false reference to navigation devices in that area. To prevent this, in our scheme each navigation station is required to keep the channel busy by either transmitting valid navigation messages in uninterrupted sequence or by transmitting I-coded sequences that will prevent the attacker from forging any meaningful messages on that channel. Note, however, that in this case there has to be a way for the navigation station BS_i to inform the receiver B about the beginning and the end of any message $c_i(t_s)$ emitted over the channel. In our navigation system SecNav, this is achieved by means of the *incongruous-delimiter* (*I-delimiter*). In the following section, we show how navigation stations (BS_i) and navigation devices (B) can use I-delimiters in order to synchronize securely with respect to the beginning and the end of the transmission of the given message $c_i(t_s)$.

3.3 SecNav Message Synchronization via Incongruous-Delimiter (I-delimiter)

Assuming that the station transmits sequences of navigation messages, we implement the navigation message delimiters which enable navigation stations (BS_i) to recognize the start and the end of each message (even if the messages vary in length). We introduce message delimiters through the following example. Let us assume that the station wants to transmit the following two codewords consecutively

$$\begin{aligned}c_i(t_s) &= 1010011001\\c_i(t_s+\Delta t) &= 1010010101\end{aligned}$$

which, under Manchester encoding rule, correspond to navigation messages $m_i(t_s) = 11010$ and $m_i(t_s) = 11000$, respectively. The station BS_i simply emits (using on-off keying - see Figure 4) the following sequence

$$\overbrace{111000}^{\text{delimiter}}\ \overbrace{1010011001}^{c_i(t_s)}\ \overbrace{111000}^{\text{delimiter}}\ \overbrace{1010010101}^{c_i(t_s+\Delta t)}\ \overbrace{111000}^{\text{delimiter}}$$

Here, the "delimiter=111000" is a specially constructed symbol-string such that any *successfully demodulated codeword*[1] received between any two consecutive "delimiters" is authentic. This is true as the delimiter sequence 111000 cannot appear as a part of any correctly encoded message nor can it be forged by an adversary, given that the adversary cannot convert "1" symbols to "0" (see Section 5.2). This effectively prevents the adversary from "shifting" delimiters in time and thus forging transmitted navigation messages without being detected.

In the following section, we present the range-free SecNav (SecNav-F).

4 SecNav-F: Secure Range-Free Broadcast Navigation

Secure Range-Free Broadcast Navigation (SecNav-F) relies on the same message (Integrity) coding as SecNav-R. Navigation messages in SecNav-F have the same format as those in SecNav-R and are equally separated by I-delimiters (Section 3.3).

In SecNav-F, every navigation station BS_i transmit navigation messages $m_i(t_s) = BS_i \| t_s \| L_i$ containing its identifier BS_i, message sending time t_s and its location L_i to navigation devices in its vicinity. This message is appended with the message signature $sig_{K_N}\{m_i(t_s)\}$, generated with the infrastructure private key K_N. Before emitting $m_i(t_s)$, $sig_{K_N}\{m_i(t_s)\}$ over a radio channel, BS_i transforms this message using integrity coding as shown on Figure 4.

---

[1] In our example, by "successfully demodulated codeword" we mean the codeword for which the transformation ($10 \rightarrow 1, 01 \rightarrow 0$) exists.

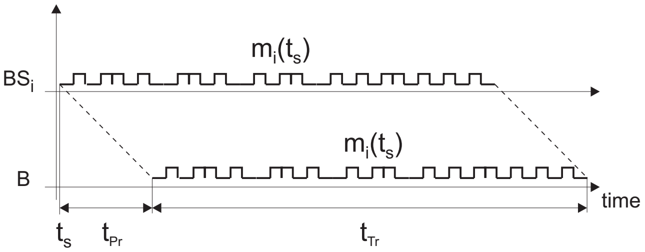


Figure 6. Secure Time Synchronization with SecNav-F. To synchronize with the infrastructure station BS_i, the receiver (B) sets its local clock $Cl_B = t_s - t_{Pr} - t_{Tr}$.

The navigation device collects messages from stations for a predefined time period of duration Δt. Upon receiving messages from at least three (four in the case of 3D localization) navigation stations, the device starts their verification and the computation of its location. The duration of Δt is set by the wireless device and it depends on device's speed of displacement.

The device first demodulates navigation messages and verifies their integrity and authenticity by performing the same four message verification steps as in SecNav-R: (i) verifies that it resides in the infrastructure coverage area, (ii) verifies that the channel on which it received the signal $s_i(t_s)$ is the channel used by the infrastructure, (iii) verifies that the demodulated message $c_i(t_s)$ is valid, i.e., it contains an equal number of symbols "1" and "0" and (iv) verifies that the demodulated signature $sig_{K_N}\{m_i(t_s)\}$ correspond to the demodulated message $m_i(t_s)$. If these verifications are successful the navigation device computes its location (x_B, y_B) within the area defined by the stations' ranges. This is illustrated on Figure 2(b). One example of such computation is the Minimum Mean Square Estimate (MMSE), which computes the devices location as follows:

$$\text{Let } f_i(x'_B, y'_B) = R - \sqrt{(x_i - x'_B)^2 + (y_i - y'_B)^2}$$

$$\boxed{\begin{array}{l}\text{The location } (x_B, y_B) \text{ is then obtained by minimizing} \\ F(x'_B, y'_B) = \sum_{BS_i \in S} f_i^2(x'_B, y'_B) \\ \text{over all estimates } (x'_B, y'_B)\end{array}}$$

where $L_i = (x_i, y_i)$ is the location of station BS_i, R is the power range of stations and S is the set of stations whose messages B received within Δt.

Note that for localization purposes, in SecNav-F, navigation stations do not need to be mutually synchronized and that navigation messages do not need to be sent simultaneously. Stations, however, do send navigation messages continuously.

In SecNav-F, for a navigation device to synchronizes to the infrastructure it is sufficient that it receives messages from at least one of the navigation stations. It then adjusts its local clock Cl_B as follows:

$$Cl_B = t_s - t_{Pr} - t_{Tr}$$

where t_s is the timestamp contained in the navigation message, t_{Tr} is the message transmission time (which depends on the message length and on the transmission speed) and t_{Pr} is the message propagation time (which depends on the distance between the station and the device); t_{Pr} is typically few nanoseconds, and it can therefore be neglected in most applications. Time Synchronization in SecNav-F is illustrated on Figure 6.

5 Security Analysis

As we already noted, in SecNav, attacks on localization are prevented by the construction of the codes used to encode navigation signals and by devices' awareness of presence in the coverage area of the infrastructure. In the following security analysis, we will assume that devices and/or users are aware of their presence in the infrastructure coverage area.

As we already described in Section 2, navigation systems are vulnerable to a range of attacks by manipulation of navigation signals.

In SecNav, message **forgery**, manipulation and **replay** is prevented through permanent transmissions of navigation signals on the communication channel. By permanent presence of legitimate navigation messages on all four communication channels and over the entire infrastructure coverage area, the attacker is prevented from inserting false navigation messages, without being detected. If the attacker inserts its (false) navigation message, this message will interleave with navigation messages sent by the infrastructure. The receivers will therefore reject the received superposition of two messages because the ratio of the number of symbols 1 and 0 in that message will be different from the one expected at the receivers. Essentially, any message forged by the attacker, replayed, or simply modified in transmission will be equally rejected at the receiver as it will change the ratio of the number of 1s and 0s in the received message. Following the same reasoning, we can conclude that the **replay of aggregated navigation signals** will be equally prevented. These aggregated navigation signals will interleave with legitimate navigation signals sent by the infrastructure and will cause the receivers to reject the received signals. If the device is unknowingly displaced from the infrastructure coverage area, message forgery is still prevented by the use of digital signatures. However, in our scenario, we assume that the devices are aware of their presence in the coverage area of the infrastructure, e.g., on campus; in Section 5.1 we detail how this is ensured.

Since message replay and forgery are prevented in SecNav, attacks on localization and time-synchronization by pulse-delays are equally prevented. E.g., if pulse-delay is attempted by **jam-and replay**, this will be detected at the receivers as the messages replayed by the attacker will be superimposed to the legitimate messages sent by the infrastructure. Given that to detect symbols 0 and 1 on the channel, receivers measure strengths of the received signals (as opposed to their signal-to-noise ratio), attacks by message **overshadowing** will be equally detected. Pulse-delay attacks with message overshadowing will therefore be equally detected.

5.1 Awareness of presence

Although SecNav effectively prevents attacks by manipulation of navigation signals, there are some physical attacks that SecNav cannot prevent. One example of such an attack is when the attacker cuts-off node's communication to the navigation infrastructure (either by displacing the node out of infrastructure coverage area, or by placing it into a Faraday cage) and then feeds it with false navigation signals. If the users control their devices, that kind of attacks are highly unlikely. If, however, the devices are autonomous, such attacks are hard to prevent, irrespective of the navigation system, unless the devices can detect displacement and/or encapsulation.

SecNav relies on the devices awareness of presence in the infrastructure coverage area. In the case of user-centric applications, the knowledge of this coverage area can be made known to the user by a trusted authority; the user can then use the system only within the intended area. In the case of autonomous devices, what suffices is that they are once initiated within the infrastructure coverage area, where they can securely obtain their location; the devices can then be programmed never to leave the intended area. Attacks involving physical removal of the autonomous devices from the coverage area can be prevented by requiring devices to occasionally check its proximity to the navigation stations by means of e.g., authenticated ranging [54] or distance bounding [2] (these techniques do, however, require occasional bidirectional communication between the infrastructure and mobile devices). Alternatively, these attacks can be prevented by the use of motion detectors and/or inertial navigation systems [6].

5.2 Preventing the attacker from erasing symbol "1"

So far, we showed that any message manipulation by the attacker will result in the ratio between the number of "0"s and "1"s being changed in the message, resulting in the message rejection at the receiver. Here, we assumed, notably, that the attacker is not able to convert symbol "1" into "0", but is only able to convert symbol "0" into "1". This scheme provided integrity protection of transmitted messages and implicitly also enabled the verification of their authenticity.

In order to erase the signal (symbol "1") from the channel, the attacker needs to be able to predict the shape of the signal at the receiver and send the inverted signal to the receiver to cancel it out (see Figure 4). There are several major factors that make it difficult for the attacker to erase the signal from the channel: the randomness of the channel, the randomness of the signal generated at the sender and the mobility of the navigation device.

To prevent the attacker from erasing the signal, we implement the following scheme: the sender randomizes the signals corresponding to symbols "1". Specifically, to prevent signal erasure, each symbol "1" of the I-coded message c is transmitted as a random signal of duration T_S. Note that we can randomize amplitude, phase, frequency etc. Given the randomness of this signal, it is difficult for the attacker to flip symbol "1" to "0" as it would need to predict the shape of the random signal in order to cancel it. In [49] the authors present a detailed analysis of the effects of the randomness of the radio signal, on the attacker's ability to erase it from the channel.

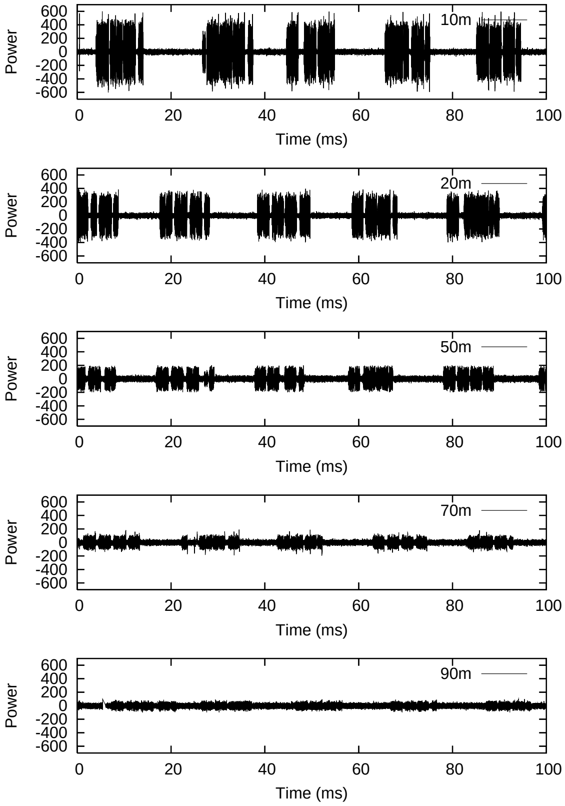


Figure 7. Signal strength at 10, 20, 50, 70 and 90 meters. Note that the "symbol width" is much wider ($\sim 10ms$) than it needs to be to make the bits easy to identify.

5.3 Denial of Service Attacks

Although an attacker cannot flip symbol "1" to "0", it can simply flip any symbol "0" into "1", by transmitting on navigation channels. If the attacker continuously transmits on these channels, it will permanently disrupt the navigation system. SecNav is not designed to prevent Denial of Service Attacks (DoS) and will not resist to continuous jamming. However, casual interference and sporadic jamming will not affect the operation of SecNav; a short discussion on this is presented in Section 6.2. Prevention of jamming on SecNav is part of our future work.

6 Implementation issues

In this section, we show results of our SecNav implementation feasibility study. Our study focused on the robustness of SecNav signals to interference and casual

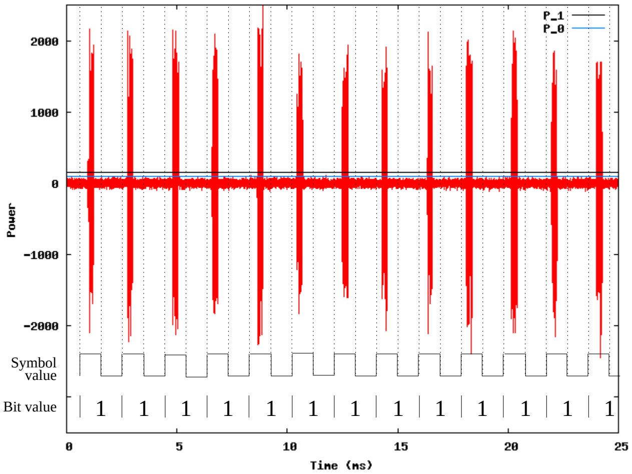


Figure 8. SecNav signal with 2 *ms* bit duration, resulting in SecNav signal transmission rate of 500 *bits/s*. The signal is shown with the corresponding symbol- and bit-values underneath.

jamming. We also investigated the maximal reach of navigation signals, the bit-width and navigation station deployment issues.

6.1 Transmission range and signal coding

To receive SecNav signals, the receiver must be able to reliably detect the presence and absence of signal on the channel. As described in Section 3.1, symbol "1" is decoded at the receiver if $P_r \geq P_1$, where P_r is the average received signal strength at the receiver during the symbol duration period T, and P_1 is the predefined signal strength threshold. Equally, if the average power $P_r \leq P_0$, the receiver detects symbol "0". Figure 7 displays the results of measurements of SecNav navigational signals at the receiver whose distance from the sender was changed from 10 to 90 meters (LoS). In this experiment, the signals were sent using a standard built-in Atheros 5212 wireless network card and received by a software radio [59]. Both the transmitter and the receiver were equipped with built-in omnidirectional antennas, whose gains were not enhanced. The transmission power at the sender was set to $100mW$. These results show that the receiver can clearly distinguish symbol "1" from environmental noise (i.e., symbol "0") up to almost $100m$. Note that these results can be further enhanced if higher transmission powers and antenna gains are used at the stations.

Besides the reach of navigation signals, our experiments also included the estimation of the maximal rate of SecNav navigation signals, using 802.11b devices. Using available MadWifi [60] drivers we were successful in transmitting SecNav signals with bit durations of approx. 2 *ms* (i.e., symbol durations of approx. 1 *ms*). The results of this experiment are illustrated on Figure 8. This experiment showed that with off-the-shelf cards and drivers, the data rate of SecNav signals can be 500 *bits/s*, which is sufficient to transmit one navigation message/second. This means that the devices will be able to synchronize to the infrastructure each second, and determine their location in the worst case every few seconds (accounting for possible clock drift between navigation stations). Appropriate modifications of the wireless card drivers of the sender (and receiver) will allow the rate of the SecNav signals to be further increased; this is part of our ongoing work.

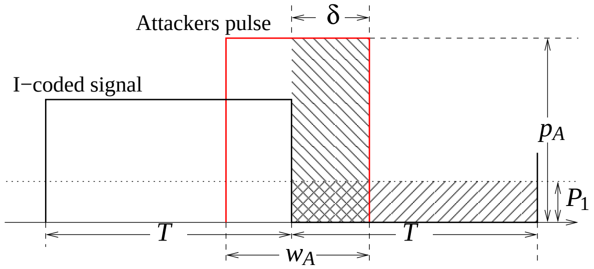


Figure 9. Visualization of equation (5). The signal marked "Attackers pulse" is the attackers jamming pulse with width w_A and hight p_A. Jamming is only successful if the shaded area $\delta \cdot p_A$ is bigger than, or equal to, the shaded area $T \cdot P_1$.

6.2 Robustness

In this section, we analyze the robustness of SecNav to jamming and interference. Because SecNav uses on-off keying it is generally not robust to continuous interference or jamming. If the receiver receives sufficient signal power within a predefined time slot, it will interpret it as the symbol "1". This means that any interference whose average power within a time slot T is above a predefined threshold P_1 will be also interpreted as the symbol "1" at the receiver. Continuous interference or jamming will therefore turn each SecNav navigation message into an unusable stream of "1" symbols. However, due to the Manchester coding and the digital signatures, SecNav messages still provide robustness to casual interference or jamming, and enable message reconstruction. If, due to interference, the navigation message ...100110..., followed by its digital signature $sig_{K_N}\{...100110...\}$, is received at the receiver as ...101110..., the receiver will be able to detect that the error occurred in symbol three or four (i.e., a symbol string 11 cannot be decoded into a valid bit using Manchester coding). Upon detection, the receiver can reconstruct the message by turning symbols three and four into 01 and 10 and observing which of the messages (...101010... or ...100110...) corresponds to the received signature. Here, on-off keying enables immediate and simple recognition of transmission errors, and Manchester coding enables the receiver to reconstruct the message.

If interference or jamming appears in random bursts, those bursts might overlap with a part of the signal that was transmitted as a "1" and thus cause no interference. The probability of successful jamming (or harmful interference) depends on the length of the jamming burst and the length of the time slots (i.e., the symbol-width) of the I-coded signal. The jamming burst scenario is shown in Figure 9 and the probability of successful random jamming (interference) is given by:

$$P_{interf} = \begin{cases} 0 & \text{for} \quad w_A < \delta \\ \frac{T+w_A-2\delta}{2T} & \text{for} \quad \delta \le w_A \le T+2\delta \\ 1 & \text{for} \quad w_A > T+2\delta \end{cases} \qquad (5)$$

where T it the width of the time slot, w_A is the width of the attackers pulse and δ is the width of the pulse required to change a 0 to a 1. δ is a function of the threshold P_1, the time slot width T and the power of the attackers transmission p_A:

$$\delta = \frac{P_1 T}{p_A} \qquad (6)$$

As we can see from (5), narrow pulses of interference (less than δ) are simply ignored. Equation (6) shows that there are two ways a SecNav message can be made more resistant to jamming. The first is increasing the threshold P_1 to force the attacker to either use more power or "hit" the zeros in the SecNav transmission more accurately. The second way to increase robustness is to increase the width of the time slot T. To see this more clearly we can rewrite (5) and (6) to:

$$P_{interf} = \frac{1}{2} + \frac{w_A}{2T} - \frac{P_1}{p_A} \quad \text{for} \quad TP_1 \le p_A w_A \le T(2P_1+1) \qquad (7)$$

Here it is clear that a bigger T forces the attacker to spend more power, however, a larger time slot has the unwanted consequence of lowering the bit rate. An in depth analysis of the optimal setting for these parameters is part of our ongoing work.

Regardless of the values of the above mentioned parameters the receiver will detect any successful jamming or interference with 100% probability. Thus, while the attacker can jam the system, he can not do so without being detected.

6.3 Deployment of navigation stations

In order to compute its location, each client needs to receive SecNav navigation signals from several (e.g., four) stations. However, given that SecNav signals emmited on the same channel mutually interfere, the deployment of navigation stations need to ensure that no location is covered by more than one station emmiting on the same channel, and that each location is covered by at least three (four in 3D) stations (emmiting on different channels).

Ideally we would like the entire navigation area to be covered by all the channels in use (i.e., if there are 4 channels being used, the entire area should be covered by all 4). Unfortunately this is very hard to achieve if the area is too big to be covered by a single navigation station, since multiple stations on the same frequency interfere with each other. If we have a precisely confined area and a lot of time it is possible to pre-measure the entire space and, using directional antennas, create an environment almost free of interference. A more interesting case, however, is the one where we want to cover an arbitrary area with navigation stations without the use of directional antennas or variable power ranges. We make only two assumptions: (i) that we can approximately predict the radius of the transmission of the navigation stations and (ii) that the area covered by each navigation station is ruffly circular.

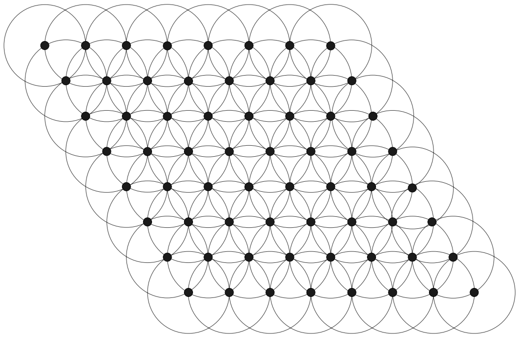


Figure 10. Navigation stations placed in a honeycomb (hexagonal) grid. The circles indicate their transmission range.

Under the above assumptions we can use a honeycomb (hexagonal) grid [15] such as the one shown in Figure 10. In this setup every point in the coverage area is covered by at least 3 (and at most 4) navigation stations. This means that the client will know that something is wrong if it receives less than 3 or more than 4 beacons.

Figure 11 is a close up of a part of the coverage area. A node will conclude that it is within the shaded area shown in Figure 11(a) if it receives 3 beacons and within the shaded area in Figure 11(b) if it receives 4 beacons. Since every point in the space is covered by at least 3 beacons, a node that receives only 3 beacons can conclude that non of them are malicious. This is because an attacker can only add new signals, not remove the ones that are already there. If the node receives 4 beacons, as shown in Figure 11(b), either of signals from the far left or far right nodes might be a fake signal sent by the attacker, thus the area of uncertainty is larger.

Using a honeycomb grid we can cover an arbitrary area with enough signals to provide a granularity of A_{4b} in the case of 4 beacons and A_{3b} in the case of 3 beacons, where A_{3b} and A_{4b} are given by:

$$A_{3b} = R^2\left(\sqrt{3} - \frac{\pi}{2}\right); \qquad A_{4b} = R^2\left(\frac{9\sqrt{3}-4\pi}{6}\right) \qquad (8)$$

where R is the transmission range of the navigation stations. With a range of $R = 100m$ we get areas of uncertainty corresponding to squares of about 40×40 meters in the best case and 70×70 meters in the worst case.

7 Related work

In the last decade, a number of indoor localization systems were proposed, based notably on infrared [56],

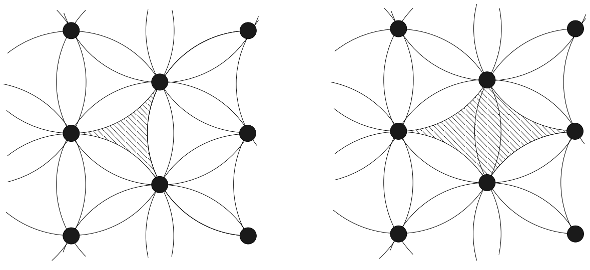

(a) (b)

Figure 11. An example of the localization granularity using seven frequencies and a symmetrical deployment scheme (all transmission ranges are the same). (a) shows the case with three beacons received (b) shows the case with four beacons received.

ultrasound [57, 37], received radio signal strength [1, 19, 4] and radio time-of-flight [27, 11] techniques. These localization techniques were also extended to wireless ad hoc networks [7, 3, 52, 35, 42, 34, 10, 32].

Recently, a number of secure distance and location verification schemes have been proposed. Brands and Chaum [2] proposed a distance bounding protocol that can be used to verify the proximity of two devices connected by a wired link. Sastry, Shankar and Wagner [41] proposed a distance bounding protocol, based on ultrasonic and radio wireless communication.

Kuhn [24] proposed an asymmetric security mechanism for navigation signals, based on hidden message spreading codes. Lazos et al. [25] proposed a set of techniques for secure positioning of a network of sensors based on directional antennas. Both these approaches, however, remain vulnerable to attacks including the replay of aggregated navigation signals. Čapkun and Hubaux [53, 54] propose a technique called verifiable multilateration, based on distance-bounding, which enables a local infrastructure to verify positions of the nodes. Lazos et al. [26] propose an extension of their work in [25] that copes with the replay of navigation signals. In [55], Čapkun et al. propose a secure localization scheme based on hidden and mobile base stations. Although these techniques prevent message replays, they assume bi-directional communication between the infrastructure and the devices and require that stations and devices are equipped with fast processing ($O(ns)$) hardware. We pose no such requirements in SecNav-F. For SecNav-R, we require similar processing speed as found in GPS receivers. Li et al. [28] and Liu et al. [29] propose statistical methods for securing localization in wireless sensor networks. In [51], the authors propose a secure localization scheme based on RSS-based ranging. These techniques assume a limited attacker that can only modify a fraction of navigation messages exchanged between the nodes, or has limited processing speed. We do not make such assumptions in SecNav.

Similarly to localization, time synchronization has equally been thoroughly studied, especially in the context of sensor networks [47]. In this context, there are several prototype implementations, such as RBS [9], TPSN [12], FTSP [31], that can achieve synchronization precision of a few microseconds. Time synchronization techniques have been shown to be vulnerable to signal manipulation attacks, similar to those that affect localization [13]. Several solutions emerged that detect such attacks; in [13] Ganeriwal et al. propose and implement a secure time synchronization scheme for sensor networks that effectively detects pulse-delay attacks. A related solution was later proposed by Sun et al. in [46]. In [30], authors analyze the impact of malicious attacks on time synchronization to sensor network applications and middleware services such as shooter localization. All these solutions assume bi-directional communication between a reference node and the synchronizing node (or between the infrastructure and the synchronized nodes). In SecNav, we do not make such assumptions; SecNav achieves secure time-synchronization through broadcast communication from reference nodes to the synchronizing nodes and is therefore well suited for securing reference-broadcast time-synchronization schemes. An overview of secure localization and secure time synchronization in wireless networks can be found in [36].

8 Conclusion

In this work, we proposed SecNav, a novel secure navigation protocol based on navigation signal broadcasts. We showed that this protocol prevents a wide range of attacks on localization and time synchronization, including message forgery and replay; SecNav is the first navigation system that effectively prevents location spoofing attacks using aggregated signal replays.

We proposed two instances of SecNav: SecNav-R, which secures range-based navigation, and SecNav-F, which secures range-free navigation. Our implementation of SecNav-F using 802.11b shows that this scheme can be efficiently deployed using available technology.

9 References

[1] P. Bahl and V. N. Padmanabhan. RADAR: An In-Building RF-Based User Location and Tracking System. In *Proceedings of the IEEE Conference on Computer Communications (InfoCom)*, volume 2, pages 775–784, 2000.

[2] S. Brands and D. Chaum. Distance-bounding protocols. In *Workshop on the theory and application of cryptographic techniques on Advances in cryptology*, pages 344–359. Springer-Verlag New York, Inc., 1994.

[3] N. Bulusu, J. Heidemann, and D. Estrin. GPS-less low cost outdoor localization for very small devices. *IEEE Personal Communications Magazine*, 7(5):28–34, October 2000.

[4] P. Castro, P. Chiu, T. Kremenek, and R. Muntz. A Probabilistic Room Location Service for Wireless Networked Environments. In *Proceedings of the Third International Conference Atlanta Ubiquitous Computing (Ubicomp)*, volume 2201. Springer-Verlag Heidelberg, September 2001.

[5] Haowen Chan, Mark Luk, and Adrian Perrig. Using Clustering Information for Sensor Network Localization. In *Proceedings of IEEE Conference on Distributed Computing in Sensor Systems (DCOSS 2005)*, June 2005.

[6] Averil Chatfield. *Fundamentals of High Accuracy Inertial Navigation (Progress in Astronautics and Aeronautics)*. AIAA, 1997.

[7] L. Doherty, K. Pister, and L. El Ghaoui. Convex position estimation in wireless sensor networks. In *Proceedings of the IEEE Conference on Computer Communications (InfoCom)*, April 2001.

[8] D. Dolev and A. C. Yao. On the Security of Public Key Protocols. *IEEE Transaction of Information Technology*, 29(2):198Ű–208, 1983.

[9] J. Elson, L. Girod, and D. Estrin. Fine-grained network time synchronization using reference broadcasts. *SIGOPS Operating System Review*, 36(SI):147–163, 2002.

[10] T. Eren, D. Goldenberg, W. Whiteley, Y.R. Yang, A.S. Morse, B.D.O. Anderson, and P.N. Belhumeur. Rigidity, computation, and randomization in network localization. In *Proceedings of the IEEE Conference on Computer Communications (InfoCom)*, 2004.

[11] R.J. Fontana. Experimental Results from an Ultra Wideband Precision Geolocation System. *Ultra-Wideband, Short-Pulse Electromagnetics*, May 2000.

[12] S. Ganeriwal, R. Kumar, and M. B. Srivastava. Timing-sync protocol for sensor networks. In *Proceedings of the ACM Conference on Networked Sensor Systems (SenSys)*, pages 138–149. ACM Press, 2003.

[13] Saurabh Ganeriwal, Srdjan Čapkun, Chih-Chieh Han, and Mani B. Srivastava. Secure time synchronization service for sensor networks. In *WiSe '05: Proceedings of the 4th ACM workshop on Wireless security*, pages 97–106, New York, NY, USA, 2005. ACM Press.

[14] I. Getting. The Global Positioning System. *IEEE Spectrum*, December 1993.

[15] Sasthi C. Ghosh, Bhabani P. Sinha, and Nabanita Das. An efficient channel assignment technique for hexagonal cellular networks. In *ISPAN '02: Proceedings of the 2002 International Symposium on Parallel Architectures, Algorithms and Networks*, page 361, Washington, DC, USA, 2002. IEEE Computer Society.

[16] M. Gruteser and D. Grunwald. Enhancing location privacy in wireless LAN through disposable interface identifiers: a quantitative analysis. In *Proceedings of WMASH*, 2003.

[17] G. P. Hancke and M. G. Kuhn. An rfid distance bounding protocol. In *IEEE SecureComm*, 2005.

[18] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher. Range-free localization schemes for large scale sensor networks. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 81–95. ACM Press, 2003.

[19] J. Hightower, G. Boriello, and R. Want. SpotON: An indoor 3D Location Sensing Technology Based on RF Signal Strength. Technical Report 2000-02-02, University of Washington, 2000.

[20] L. Hu and D. Evans. Localization for mobile sensor networks. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*. ACM Press, 2004.

[21] Y.-C. Hu, A. Perrig, and D. B. Johnson. Packet Leashes: A Defense against Wormhole Attacks in Wireless Networks. In *Proceedings of the IEEE Conference on Computer Communications (InfoCom)*, San Francisco, USA, April 2003.

[22] L. Huang, K. Matsuura, H. Yamane, and K. Sezaki. Enhancing Wireless Location Privacy Using Silent Period. In *Proceedings of the IEEE Wireless Communications and Networking Conference(WCNC)*, 2005.

[23] J. Kong and X. Hong. ANODR: ANonymous On Demand Routing with Untraceable Routes for Mobile Ad-hoc Networks. In *Proceedings of MobiHoc*, 2003.

[24] M. G. Kuhn. An Asymmetric Security Mechanism for Navigation Signals. In *Proceedings of the Information Hiding Workshop*, 2004.

[25] L. Lazos and R. Poovendran. SeRLoc: Secure Range-Independent Localization for Wireless Sensor Networks. In *Proceedings of the ACM Workshop on Wireless Security (WiSe)*, 2004.

[26] L. Lazos, S. Čapkun, and R. Poovendran. ROPE: Robust Position Estimation in Wireless Sensor Networks. In *Proceedings of IPSN*, 2005.

[27] J.-Y. Lee and R.A. Scholtz. Ranging in a Dense Multipath Environment Using an UWB Radio Link. *IEEE Journal on Selected Areas in Communications*, 20(9), December 2002.

[28] Z. Li, W. Trappe, Y. Zhang, and B. Nath. Robust Statistical Methods for Securing Wireless Localization in Sensor Networks. In *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN)*, 2005.

[29] D. Liu, P. Ning, and W. Du. Attack-Resistant Location Estimation in Sensor Networks. In *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN)*, 2005.

[30] Michael Manzo, Tanya Roosta, and Shankar Sastry. Time synchronization attacks in sensor networks. In *SASN '05: Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks*, pages 107–116, New York, NY, USA, 2005. ACM Press.

[31] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi. The flooding time synchronization protocol. In *SenSys*

'04: Proceedings of the 2nd international conference on Embedded networked sensor systems, pages 39–49, New York, NY, USA, 2004. ACM Press.

[32] M. Maroti, P. Volgyesi, S. Dora, B. Kusy, A. Nadas, A. Ledeczi, G. Balogh, and K. Molnar. Radio interferometric geolocation. In *Proceedings of the ACM Conference on Networked Sensor Systems (SenSys)*, pages 1–12, New York, NY, USA, 2005. ACM Press.

[33] C. Meadows, R. Poovendran, D. Pavlovic, L. Chang, and P. Syverson. Distance Bounding Protocols: Authentication Logic Analysis and Collusion Attacks. Technical report, March 2007.

[34] D. Moore, J. Leonard, D. Rus, and S. Teller. Robust distributed network localization with noisy range measurements. In *Proceedings of the ACM Conference on Networked Sensor Systems (SenSys)*, pages 50–61. ACM Press, 2004.

[35] D. Niculescu and B. Nath. DV Based Positioning in Ad hoc Networks. *Journal of Telecommunication Systems*, 22(4):267–280, 2003.

[36] R. Poovendran, C. Wang, and S. Roy (eds.). *Secure Localization and Time Synchronization for Wireless Sensor and Ad Hoc Networks*. Springer, March 2007.

[37] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket location-support system. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 32–43. ACM Press, 2000.

[38] A. R. Beresford and F. Stajano. Location Privacy in Pervasive Computing. *Pervasive Computing*, January-March 2003.

[39] I. W. Jackson. Anonymous Addresses and Confidentiality of Location. In *Proceedings of International Workshop on Information Hiding*, 1996.

[40] Y.-C. Hu and H. J. Wang. Location Privacy in Wireless Networks. In *Proceedings of the ACM SIGCOMM Asia Workshop*, 2005.

[41] N. Sastry, U. Shankar, and D. Wagner. Secure Verification of Location claims. In *Proceedings of the ACM Workshop on Wireless Security (WiSe)*, pages 1–10. ACM Press, September 2003.

[42] A. Savvides, C.-C. Han, and M. B. Strivastava. Dynamic fine-grained localization in Ad-Hoc networks of sensors. In *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom)*, pages 166–179. ACM Press, 2001.

[43] M. Sichitiu and C. Veerarittiphan. Simple, accurate time synchronization for wireless sensor networks. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)*, 2003.

[44] S.Sedighpour, S.Čapkun, S. Ganeriwal, and M.Srivastava. Implementation of Attacks on Ultrasonic Ranging Systems - Demo. In *Proceedings of the ACM Conference on Networked Sensor Systems (SenSys)*, 2005.

[45] Kun Sun, Peng Ning, and Cliff Wang;. Secure and resilient clock synchronization in wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 24(2):395–408, 2006.

[46] Kun Sun, Peng Ning, Cliff Wang, An Liu, and Yuzheng Zhou. Tinysersync: Secure and resilient time synchronization in wireless sensor networks. In *Proceedings of the ACM Conference on Computer and Communications Security*, 2006.

[47] B. Sundararaman, U. Buy, and Kshemkalyani A. Clock Synchronization for Wireless Sensor Networks: A Survey. Technical report, March 2005.

[48] J. van Greunen and J. Rabaey. Lightweight time synchronization for sensor networks. In *WSNA '03: Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, pages 11–19, New York, NY, USA, 2003. ACM Press.

[49] M. Čagalj, S. Čapkun, RamKumar Rengaswamy, Ilias Tsigkogiannis, M. Srivastava, and Jean-Pierre Hubaux. Integrity (I) codes: Message Integrity Protection and Authentication Over Insecure Channels. In *Proceedings of the IEEE Symposium on Research in Security and Privacy*, Oakland, California, USA, 2006.

[50] S. Čapkun, L. Buttyán, and J.-P. Hubaux. SECTOR: Secure Tracking of Node Encounters in Multi-hop Wireless Networks. In *Proceedings of the ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN)*, Washington, USA, October 2003.

[51] S. Čapkun, S. Ganeriwal, F. Anjum, and M. Srivastava. Secure RSS-based Localization in Sensor Networks. Technical report, September 2006.

[52] S. Čapkun, M. Hamdi, and J.-P. Hubaux. GPS-free Positioning in Mobile Ad-Hoc Networks. *Cluster Computing*, 5(2), April 2002.

[53] S. Čapkun and J.-P. Hubaux. Secure positioning of wireless devices with application to sensor networks. In *Proceedings of the IEEE Conference on Computer Communications (InfoCom)*, 2005.

[54] S. Čapkun and J.-P. Hubaux. Secure Positioning in Wireless Networks. *IEEE Journal on Selected Areas in Communications*, 24(2), February 2006.

[55] S. Čapkun, M. Čagalj, and M. Srivastava. Secure Localization with Hidden and Mobile Base Stations. In *Proceedings of the IEEE Conference on Computer Communications (InfoCom)*, 2006.

[56] R. Want, A. Hopper, V. Falcao, and J. Gibbons. The Active Badge Location system. *ACM Transactions on Information Systems*, 10(1):91–102, 1992.

[57] A. Ward, A. Jones, and A. Hopper. A New Location Technique for the Active Office. *IEEE Personal Communications*, 4(5), October 1997.

[58] J. S. Warner and R. G. Johnston. Think GPS

Cargo Tracking = High Security? Think Again. *Technical report, Los Alamos National Laboratory*, 2003.

[59] Website. Gnu radio – the gnu software radio. http://www.gnu.org/software/gnuradio/index.html.

[60] Website. Madwifi: Multiband atheros driver for wireless fidelity. http://madwifi.org/.

[61] Y. Zhang, Wei Liu, Yuguang Fang, and Dapeng Wu. Secure localization and authentication in ultra-wideband sensor networks. *IEEE Journal on Selected Areas in Communications*, 24(2), February 2006.