Secure Positioning in Wireless Networks

Srdjan Čapkun, *Member, IEEE,* and Jean-Pierre Hubaux, *Senior Member, IEEE*

***Abstract*—So far, the problem of positioning in wireless networks has been studied mainly in a nonadversarial setting. In this paper, we analyze the resistance of positioning techniques to position and distance spoofing attacks. We propose a mechanism for secure positioning of wireless devices, that we call verifiable multilateration. We then show how this mechanism can be used to secure positioning in sensor networks. We analyze our system through simulations.**

***Index Terms*—Ad hoc networks, positioning, secure protocols, security, sensor networks, wireless.**

I. Introduction

RESEARCHERS have proposed a number of positioning and distance estimation techniques for wireless networks [3], [6], [16], [30], [41], [42]. However, they all studied these techniques in nonadversarial settings. Distance estimation and positioning techniques are, nevertheless, highly vulnerable to attacks from internal and external attackers. *Internal attackers* can report false position and distance information in order to cheat on their position. *External attackers* can modify (spoof) the measured positions and distances of wireless nodes.

In this paper, we propose a mechanism for the secure position computation and verification of positions of wireless devices. We call our mechanism verifiable multilateration (VM). This mechanism is based on the measurements of the time of radio signal propagation [i.e., time-of-flight (ToF)]; it consists of conventional multilateration with distance bounding or authenticated ranging, and it enables verification of node positions by a set of (at least three) base stations, which do not need to be tightly synchronized.

Because of its simplicity, VM can be used for securing positioning in a variety of systems. In this paper, we focus on sensor network positioning, and we show how VM can ensure secure positioning of sensors in the presence of adversaries. We call this scheme SPINE, a system for **S**ecure **P**ositioning **I**n sensor **NE**tworks. We present a security and performance analysis of SPINE.

The organization of this paper is the following. In Section II, we review positioning techniques and analyze attacks against them. In Section III, we describe a technique for radio frequency distance bounding. In Section IV, we describe our technique for position verification called VM (VM). In Section V, we present a scheme for secsecure positioning of a network of sensors. In Section VI, we present an overview of current proposals and techniques for positioning in wireless networks, based on VM. We conclude the paper in Section VII.

Manuscript received October 15, 2004; revised August 15, 2005. This paper was supported in part by the National Competence Center in Research on Mobile Information and Communication Systems (NCCR-MICS), a center supported by the Swiss National Science Foundation under Grant 5005-67322.

S. Čapkun is with the Technical University of Denmark, Lyngby DK-2800, Denmark (e-mail: capkun@ucla.edu; http://imm.dtu.dk/~sca/).

J.-P. Hubaux is with École Polytechnique Fédérale de Lausanne (EPFL), Lausanne CH-1015, Switzerland (e-mail: jean-pierre.hubaux@ucla.edu; http://lcawww.epfl.ch/hubaux).

Digital Object Identifier 10.1109/JSAC.2005.861380

II. Attacks Against Position and Distance Estimation Techniques

We now review positioning and distance estimation techniques and analyze their vulnerabilities.

A. Attacker Model

First, we briefly present our attacker model. We call an attacker *external* if it cannot authenticate itself as an honest network node to other network nodes or to a central authority. We call an attacker *internal* if the node is *compromised* or if the user controlling the node is *malicious*. We assume that malicious and compromised nodes can authenticate themselves to the authority and to other network nodes. We assume that when a node is compromised, its secret keys and other secrets that it shares with other nodes are known to the attacker. Furthermore, we assume that users have full access to their devices, meaning also to their authentication material.

Similarly, we observe two types of attacks: internal and external. Internal attacks are those in which an internal attacker reports a false position or convinces the positioning infrastructure that it is at a false position. External attacks are those in which an (external) attacker convinces an honest node and the positioning infrastructure that the node is at a different position from its true position (i.e., the attacker *spoofs* node's position).

We distinguish two types of positioning systems: node-centric and infrastructure-centric. By a node-centric positioning system we mean that a node computes its position by observing signals received from public base stations with known locations. If the positioning system is *node-centric*, internal attacks are generally straightforward: the attacker simply lies about the position that it computed. *Infrastructure-centric* positioning systems are those in which the infrastructure computes positions of nodes based on their mutual communication.

In multilateration-based approaches, an internal attacker can cheat on its position by cheating on ranging mechanisms (i.e., by reporting false signal strengths and times of signal sending/reception). In time difference-of-arrival (TDOA) systems, an attacker can cheat by sending signals to base stations at different times (in some cases, the attacker needs to have directional antennas).

Attacks by external attackers are similar to those performed by internal attackers. An external attacker can perform timing

0733-8716/$20.00 © 2006 IEEE

TABLE I
VULNERABILITIES OF THE POSITIONING AND DISTANCE ESTIMATION TECHNIQUES TO DISTANCE AND POSITION SPOOFING ATTACKS

	Internal attackers	External attackers
RSS (Received Signal Strength)	Distance enlargement and reduction	Distance enlargement and reduction
US time-of-flight (ToF)	Distance enlargement and reduction	Distance enlargement and reduction
RF time-of-flight (ToF)	**Distance enlargement and reduction**	**Distance enlargement only**
US distance bounding	Distance enlargement only	Distance enlargement and reduction
RF distance bounding	**Distance enlargement only**	**Distance enlargement only**
Civilian GPS	False position reports	Position spoofing

attacks by delaying (through jamming) or speeding-up (wormhole attacks [19]) the signals, or it can perform power level modification attacks by replaying signals between nodes and base stations at different power levels.

B. Attacks on Global Positioning System (GPS)

The GPS is today the most widespread outdoor positioning system for mobile devices. The system is based on a set of satellites that provide a 3-D positioning with an accuracy of around 3 m. GPS also provides devices with an accurate time reference. GPS, however, has several limitations: it cannot be used for indoor positioning nor for positioning in dense urban regions: in those cases, because of the interferences and obstacles, satellite signals cannot reach the GPS devices. Furthermore, civilian GPS was never designed for secure positioning. Civilian GPS devices can be "spoofed" by GPS satellite simulators, that produce fake satellite radio signals that are stronger than the real signals coming from satellites. Most current GPS receivers can be totally fooled, accepting these stronger signals while ignoring the weaker, authentic signals. GPS satellite simulators are legitimately used to test new GPS products and can be bought for $10 k–$50 k or rented for just $1 k per month. Some simple software changes to most GPS receivers would permit them to detect relatively unsophisticated spoofing attacks [43]. Nevertheless, more sophisticated spoofing attacks would still be hard to detect. Military GPS are protected from position spoofing by codes that cannot be reproduced by the attackers.

Even if a mobile node is able to obtain its correct position from the GPS satellites, the authority or another mobile node have no way to verify the correctness of node's position, unless the mobile node is equipped with a trusted software or hardware module [2], providing the correct position.

C. Attacks on Ultrasound (US) Positioning

US-based systems operate by measuring ToF of the sound signal measured between two nodes. An interesting feature of these systems is that, if used with RF signals, they do not require any time synchronization between the sender and the receiver. The limitations of the US-based systems are that, due to outdoor interferences, they can be mainly used indoors.

US-based systems are vulnerable to distance reduction and distance enlargement attacks by external and internal attacks. To reduce the measured distance between two honest nodes, two attackers can use a radio link, as it transmits the signal several orders of magnitude faster than the US. Furthermore, by jamming and replaying the signals at a later time, attackers can enlarge the measured distances between honest nodes. With US-based techniques, an internal attacker can also reduce or enlarge the measured distance by laying about the signal sending/reception times or by simply delaying its response to honest nodes. Recently, Sastry *et al.* [35] have proposed a US-based distance bounding technique which resists to distance reduction attacks from internal attacks; it does not, however, resist to attacks from external nodes.

D. Attacks on Radio (RF) Positioning

In techniques based on the received signal strength (RSS), the distance is computed based on the transmitted and RSSs. To cheat on the measured distance, an internal attacker, therefore, only needs to report a false power level to an honest node. Malicious attackers can also modify the measured distance between two honest nodes by jamming the nodes' mutual communication and by replaying the messages with higher or lower power strengths.

RF ToF-based systems exhibit the best security properties. In these systems, nodes measure their mutual distance based on the time of propagation of the signal between them. Because RF signals travel at the speed-of-light, an attacker can, by jamming and replaying the signals, only increase, but not decrease the measured ToF between the nodes. An internal attacker can further cheat on the distance by laying about the signal transmission and reception times.

An RF distance bounding technique proposed by Brands and Chaum [4] exhibits better security properties than conventional RF ToF distance estimation; it allows the nodes to upper bound their distances to other nodes, meaning that it prevents an internal attacker from reducing the measured distance. As we will show in Section III in more detail, with RF ToF distance-bounding protocols, attackers can only increase, but not decrease the measured distances to honest nodes.

E. Conclusion

Our review of vulnerabilities of positioning systems is summarized Table I. This table illustrates that the RF ToF-based positioning solutions are best suited for secure positioning. The RF ToF distance estimation and distance bounding techniques are the most effective techniques to counter attacks. The reason is that with RF it is generally possible to perform precise non-line-of-sight distance estimations; the precision of the system can be very high (15 cm error with ultra-wideband (UWB) systems at a distance of 2 km [12]). A potential drawback of these systems is that, because they operate with the speed-of-light, the devices require fast-processing hardware.

III. DISTANCE BOUNDING AND AUTHENTICATED RANGING

Distance bounding techniques are used to upper bound the distance of one device to another (compromised) device. As we indicated in Table I, RF-based distance bounding protocols are

u : **Generate** random nonce N_u
: commitment $(c, d) = \text{commit}(N_u)$
$u \rightarrow v$: c
v : **Generate** random nonce N_v
$v \rightarrow u$: N_v *(bits sent from MSB to LSB)*
$u \rightarrow v$: $N_u \bigoplus N_v$ *(bits sent from LSB to MSB)*
v: **Measure** time t_{vu} between sending N_v and receiving $N_u \bigoplus N_v$
$u \rightarrow v$: $N_u, N_v, d, MAC_{K_{uv}}(u, N_u, N_v, d)$
v: **Verify** MAC and verify if $N_u = \text{open}(c, d)$

Fig. 1. Distance bounding protocol.

u : **Generate** random nonce N_u
: commitment $(c, d) = \text{commit}(N_u)$
$u \rightarrow v$: c
v : **Generate** random nonce N_v
$v(t_s^v) \rightarrow (t_r^u)u$: N_v
$u(t_s^u) \rightarrow (t_r^v)v$: $N_u \bigoplus N_v$
$u \rightarrow v$: $t_s^u, t_r^u, d, MAC_{K_{uv}}(u, N_u, N_v, t_s^u, t_r^u, d)$
v: **Verify** MAC and verify if $N_u = \text{open}(c, d)$
v: **Compute** $d = (t_r^v - t_s^v - t_s^u + t_r^u)s$

Fig. 2. Authenticated ranging protocol.

vulnerable to distance enlargement attacks but not to distance reduction attacks. Distance bounding protocols are used by a verifier v to verify that a claimant node u being at a distance d_{uv} from a verifier node v, cannot claim to be at a distance $d'_{uv} < d_{uv}$. These protocols were first introduced by Brands and Chaum [4] to prevent Mafia fraud attacks.

The pseudocode of the distance bounding protocol is shown in Fig. 1. In the first step of the protocol, the claimant u commits to a random value N_u. The verifier replies with a challenge nonce N_v, sends it to u in a reverse bit order and starts its timer as soon as the last bit of the challenge has been sent. The claimant u responds immediately with $N_v \bigoplus N_u$, upon receiving the challenge from v. Once the verifier has received the response from u it stops the timer and converts the challenge-response time t_{vu} to a distance d_{vu}. In the last step of the protocol, u authenticates itself to v and reveals the decommit value $\hat{d}$. The authentication and the authenticity of d is ensured with a message authentication code (MAC), using a secret key K_{vu} that u and v share. Finally, v verifies if the value N_u received in the time-measuring phase corresponds to the received (commit, decommit) pair $(c, \hat{d})$.

The commitment scheme needs to satisfy two properties: 1) a user who commits to a certain value cannot change this value afterwards (we say that the scheme is *binding*) and 2) the commitment is hidden from its receiver until the sender "opens" it (we say that the scheme is *hiding*). A commitment scheme transforms a value m into a commitment/opening pair (c, d), where c reveals no information about m, but (c, d) together reveal m, and it is infeasible to find $\hat{d}$ such that $(c, \hat{d})$ reveals $\hat{m} \neq m$. Simple commitment schemes can be realized with hash functions, which do not impose high computational requirements on sensor nodes.

The described protocol is suitable for devices that can perform rapid message exchanges, execute XOR operations rapidly, and perform encryption. In the case of RF-based distance bounding, the most important assumptions are that the claimant needs to be able to bound its processing (XOR) to a few nanoseconds, and that the verifier v needs to be able to measure time with nanosecond precision (1 ns corresponds to the time that it takes an electromagnetic wave to propagate over 30 cm). This requirement allows the node to perform distance bounding with radio signals with an uncertainty of 30 cm. We are aware that a nanosecond processing and time measurements are achievable only with dedicated hardware. Recent developments in location system show that RF time of flight systems based on UWB can achieve nanosecond precision of measured times of signal flight (and consequently of the distances). The tests with multispectral solution's UWB precision asset location system [13] consisting of active tags and tracking devices show that this system can provide two-dimensional (2-D) and three-dimensional (3-D) location of objects to within a few centimeters. The range of the system is 100 m indoor and 2 km outdoor. The used UWB tags are active and roughly the size of a wristwatch, weighing approximately 40 grams each.

In the case of a US-based distance bounding, node processing speed and clock accuracy can be of the order of milliseconds. Thus, US distance bounding can be easily implemented with off-the-shelf components such as microphones and 802.11 wireless cards [35].

Authenticated ranging protocols enable two honest and trusted parties to measure their mutual distance in an authenticated manner. Fig. 2 shows one possible realization of the authenticated distance ranging protocol, inspired by Brand's and Chaum's distance bounding protocol. Here, t_s^u, t_r^u and t_s^v, t_r^v are the message sending and reception times at nodes u and v, respectively; s is the speed-of-light.

In this protocol, unlike in the distance bounding protocol, it is not required that the claimant replies within a nanosecond time, but only that it is able to measure time with that precision. Given that the claimant and the verifier are mutually trusted, the claimant (u) reports its processing time to the verifier (v) which then computes the range based on the reported times using speed-of-light.

Like in the distance bounding protocol, in this protocol, the processing at the nodes is minimized during the ranging phase, and most processing (MAC and commitment verification) is performed *a posteriori* to the ranging.

The advantages of the ranging protocol over distance bounding are in that the nodes do not need to have high-speed hardware to perform XOR and that the channel does not need to be reserved during the ranging phase (as processing and channel access times are measured and reported). One disadvantage is that ranging is not resistant to distance reduction by internal attackers.

A very important observation here is that, essentially, authenticated ranging and distance bounding have the same resistance to external attackers: the only attack that the external attackers can successfully perform is distance enlargement. In case of internal attackers, distance bounding prevents distance reduction, whereas the authenticated ranging is vulnerable to this attack.

IV. Verifiable Multilateration (VM)

In Section II, we described security problems related to various positioning and distance estimation techniques, and in Section III, we showed how the devices can upper bound their mutual distances. We now propose a technique for position verification that we call VM. This technique enables a secure computation and verification of the positions of mobile devices in the presence of attackers. Here, by *secure position computation* we mean that base stations compute the correct position of a node in the presence of attacker, or that a node can compute its own position in the presence of an attacker; by *secure position verification* we mean that the base stations can verify the position reported by the node.

Multilateration is a technique for determining the position of a (mobile) device from a set of reference points whose positions are known, based on the ranges measured between the reference points and the device. The position of the device in two (three) dimensions can be computed if the device measured its distance to three (four) reference points. As we already detailed in Section II, distance estimation techniques are vulnerable to attacks from internal and external attacks, which can maliciously modify the measured distances. Multilateration is equally vulnerable to the same set of attacks because it relies on distance estimations.

A. Algorithm

Verifiable multilateration relies on distance bounding (or on authenticated ranging). It consists of distance bound measurements from at least three reference points (verifiers) to the mobile device (the claimant) and of subsequent computations performed by an authority. In this description, we will assume that the verification is performed with distance bounding. For simplicity, we show the algorithm for 2-D positioning; at the end of the section, we briefly comment on how a similar algorithm can be applied to the 3-D case.

The intuition behind the VM algorithm is the following. Because of the distance bounding property, the claimant can only pretend that it is more distant from the verifier than it really is. If it increases the measured distance to one of the verifiers, in order to keep the position consistent, the claimant needs to prove that at least one of the measured distances to other verifiers is shorter than it actually is, which it cannot because of the distance bounding. This property holds only if the position of the claimant is determined within the triangle formed by the verifiers. This can be explained with a simple example: if an object is located within the triangle, and it moves to a different position within the triangle, it will certainly reduce its distance to at least one of the triangle vertices. The same properties hold if an external attacker enlarges distances between verifiers and an honest claimant. This basic intuition is illustrated in Fig. 3(a).

More precisely, the VM algorithm is executed by the verifiers, as shown on Fig. 4.

In step 1 of the algorithm, the verifiers $v_1, \ldots, v_n$ which are in the power range of the claimant u perform distance bounding to the claimant u and obtain distance bounds $db_1, \ldots, db_n$. These distance bounds, as well as the positions of the verifiers (which are known) are then reported to the central authority. In step 2, the authority computes an estimate (x', y') of the claimant's position; this position is computed by using distance bounds from all verifiers in u's neighborhood, typically by the minimum mean square estimate (MMSE)

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

The position of u is obtained by minimizing $F(x'_u, y'_u) = \sum_{v_i \in \mathcal{T}} f_i^2(x'_u, y'_u)$ over all estimates of u

In step 3 of the algorithm, the authority runs the following two tests: 1) *δ-test*: for all v_i, does the distance between (x'_u, y'_u) and v_i differ from the measured distance bound db_i by less than the expected distance measurement error δ and 2) *point in the triangle test:* does (x'_u, y'_u) fall within at least one physical triangle formed by a triplet of verifiers. Note also that we call the triangle formed by the verifiers the *verification triangle*. If both the δ and the point in the triangle tests are positive, the authority accepts the estimated position (x'_u, y'_u) of the claimant as correct; else, the position is rejected.

The expected error δ is a system parameter that depends on the number of verifiers and on the distance estimation techniques used. This error becomes smaller as more verifiers are used to compute (x'_u, y'_u).

If both the δ and the point in the triangle tests are positive, this means that the claimant falls in at least one verification triangle v_i, v_j, v_k, and that distance bounds (db_i, db_j, db_k) are consistent with the estimated position and with each other [Fig. 3(a)]. This means that none of the distance bounds (db_i, db_j, db_k) were enlarged.

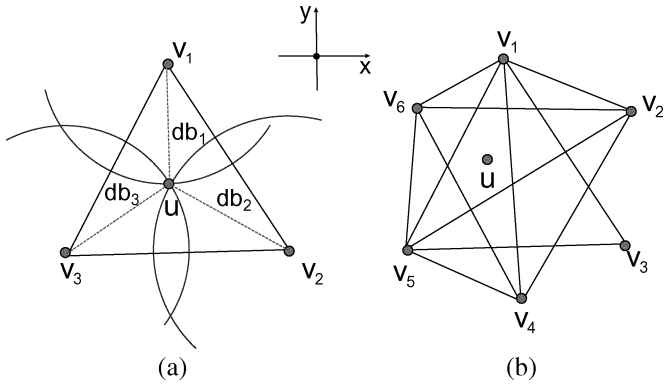

Fig. 3. Examples of verifiable multilateration (VM). (a) With three verifiers. (b) With six verifiers.

$\mathcal{T} = \emptyset$; set of verification triangles enclosing u
$\mathcal{V} = \{v_1, ..., v_n\}$; set of verifiers in the power range of u
1 *For all* $v_i \in \mathcal{V}$, perform distance bounding from v_i to u and obtain db_i
2 With all $v_i \in \mathcal{V}$, compute the estimate (x'_u, y'_u) of the position by MMSE
3 *If for all* $v_i \in \mathcal{V}$, $|db_i - \sqrt{(x_i - x'_u)^2 + (y_i - y'_u)^2}| \leq \delta$ *then*
for all $(v_i, v_j, v_k) \in \mathcal{V}^3$, *if* $(x'_u, y'_u) \in \triangle(v_i, v_j, v_k)$
then $\mathcal{T} = \mathcal{T} \cup (v_i, v_j, v_k)$
if $|\mathcal{T}| > 0$ *then* position is accepted and $x_u = x'_u$, $y_u = y'_u$
else the position is rejected
else the position is rejected

Fig. 4. Verifiable multilateration (VM) algorithm.

$\mathcal{C} = \emptyset$; set of verifiers with correctly measured bounds
$\mathcal{NC} = \emptyset$; set of verifiers whose bounds are suspicious
1 *For all* $v_i \in \mathcal{T}$
if in at least one of the verification triangles
with v_i the position of u is computed correctly
then db_i is correct, $\mathcal{C} = \mathcal{C} \cup \{v_i\}$
else $\mathcal{NC} = \mathcal{NC} \cup \{v_i\}$
2 *For all* $v_i \in \mathcal{NC}$
if v_i can create a verification triangle
with any pair $(v_j, v_k) \in \mathcal{C}^2$
then db_i *is subject to an enlargement attack*
3 With all $v_i \in \mathcal{C}$, compute the estimate (x'_u, y'_u) of the position by MMSE
4 *For all* $v_i \in \mathcal{NC}$, *if* $|db_i - \sqrt{(x_i - x'_u)^2 + (y_i - y'_u)^2}| \leq \delta$
then db_i *is subject to an enlargement attack*

Fig. 5. Detection of enlarged distances.

If any of the distance-bounds db_i differs from the estimated position (x'_u, y'_u) by more than δ, this indicates that there is a possible distance enlargement attack on one or more of the distance bounds that caused such an unexpectedly high error to occur. If a larger number of verification triangles can be formed around u, the authority can try to detect which of the distances are enlarged. Those distances can then be filtered-out and the position can be computed with the remaining set of distances. This detection is performed such that the position of u is computed independently in each triangle. If in a given triangle the computation is successful, then all the distance bounds from the verifiers forming that triangle are considered correct; otherwise, all three distance bounds are considered suspicious (see Fig. 5).

In this algorithm, the number of verification triangles and the number of enlarged distances will determine if the algorithm can detect which distance(s) is(are) enlarged. Nevertheless, in all cases, even if the number of verifiers is strictly equal to three, the VM algorithm will detect any distance enlargement attack (even if only one distance is enlarged), but it will not always be able to detect which distance it is.

VM can be also applied to 3-D positioning. For this, the system requires a minimum of four verifiers, that form a triangular pyramid, within which the secure determination of the claimant's position is possible. The algorithm is then executed in a way similar to the 2-D case.

B. Security Analysis

In this section, we analyze the security properties of VM in various scenarios. We observe VM with distance bounding or with authenticated ranging, assuming trusted or untrusted users, and with radio-based or ultrasound-based bounding/ranging.

Verifiable Multilateration With Distance Bounding: The most important properties of the VM mechanism with distance bounding can be summarized as follows.

1) A node located at position p within the triangle/pyramid formed by the verifiers cannot prove to be at another position $p' \neq p$ within the same triangle/pyramid.
2) A node located outside the triangle/pyramid cannot prove to be at any position p within the triangle/pyramid.

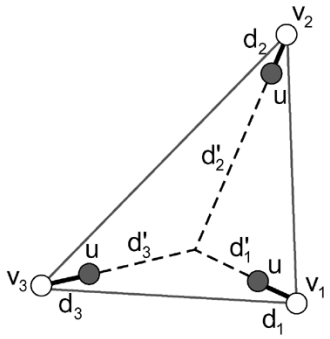

Fig. 6. Device cloning/user collusion. The attacker clones a device u and places one clone close to each verifier. The clones seem to the verifiers as a single device and can use distance enlargement to show that u is at any position within the verification triangle. Alternatively, three users collude in proving an incorrect position of node u.

3) An external attacker performing a distance enlargement attack cannot trick the verifiers into believing that a claimant located at a location p in the triangle/pyramid is located at some other position $p' \neq p$ in the triangle/pyramid.

An external attacker performing a distance enlargement attack cannot trick the verifiers into believing that a claimant is located at any position p within the triangle/pyramid, if the claimant is located outside of the triangle/pyramid.

These properties hold for verifiable multilateration based on radio distance bounding (VM-RF-DB) in environments in which the signal propagates at the speed-of-light, and for an internal attacker that controls a single device (the claimant). VM-RF-DB, therefore, resists to external attacks and to internal attacks from a single untrusted/compromised node.

However, if an attacker owns several devices and each device can authenticate to the authority as the same entity, the attacker can still successfully cheat on its position. The attacker can place three/four devices within the triangle/triangular pyramid, such that each device is close to one of the verifiers. Each of the devices can then show to its corresponding base station (by delaying the messages) that it is positioned at *any* distance larger than their actual distance (which is small). As to the base stations these devices appear to be a single claimant, the attacker can prove to be at any distance to the base stations and, thus, at any position in the verification triangle/triangular pyramid. This attack is shown on Fig. 6. Here, the attacker clones its device, or three attackers collude to appear as a single node to the verifiers. This enables the attacker (or colluding nodes) to prove that the position of the claimant is at an incorrect place within the verification triangle.

A solution that prevents this attack is to make claimant devices tamper-proof such that their authentication material is not revealed to the attacker and that they cannot be cloned; however, as shown in [2], tamper-proofness has its limitations. Another possibility is that the base stations perform device fingerprinting [37] by which they identify each device as unique. In that case, the base stations can identify a claimant device by the unique "fingerprint" that characterizes its signal transmission.[1]

[1]This process is used by cellular network operators to prevent cloning fraud; namely, a cloned phone does not have the same fingerprint as the legal phone with the same electronic identification numbers.

TABLE II
RESISTANCE OF VM TO ATTACKS. RF=RADIO COMMUNICATION, US=ULTRASONIC COMMUNICATION, DB=DISTANCE-BOUNDING, AR=AUTHENTICATED RANGING, DF=DEVICE FINGERPRINTING, UW=UNDERWATER

	external attackers	single internal attacker	single internal + external attackers	colluding internal/cloning internal
VM-RF-DB + DF	yes	yes	yes	yes
VM-RF-DB	yes	yes	yes	no
VM-US-DB	no (yes UW)	yes	no	no
VM-RF-AR	yes	no	no	no
VM-US-AR	no (yes UW)	no	no	no

Verifiable multilateration with ultrasonic distance bounding (VM-US-DB) in air, exhibits only properties (1) and (2), meaning that it protects the positioning system from an untrusted claimant, but not against an external attacker nor from colluding internal attackers (claimants). However, if the devices are under water, VM-US-DB can exhibit the same properties as VM-RF-DB; this is because, underwater, the communication is limited to ultrasonic signals. VM-US-DB can be attacked if an attacker can use surface wormholes to perform distance reduction [22].

Verifiable Multilateration With Authenticated Ranging (VM-RF-AR): VM-RF-AR exhibits only properties 3 and 4 of the VM-RF-DB. This means that this scheme provides protection against external attacks, but not against untrusted claimants (internal attackers). VM-RF-AR is, therefore, most suitable for secure positioning systems in which the infrastructure (the verifiers) and the users (the claimants) are mutually trusted. In these scenarios, VM-RF-AR resists to all distance enlargement attacks by external attackers.

Verifiable multilateration with ultrasonic authenticated ranging (VM-US-AR) exhibits the same properties as VM-RF-AR, but only in underwater communications, whereas in air, it does not provide any security at all.

The results of this analysis are summarized in Table II.

C. Maximum Attacker Impact

In this section, we analyze the impact of distance measurement errors on VM. As we have already described, for the computed position to be accepted by the verifiers, each distance bound needs to be less than δ different from the distance between the computed position and the verifier measuring that distance bound. We defined δ as the expected positioning error. Here, we define δ more precisely as 3σ, where σ is the expected standard deviation of the computed position. This means that we expect, with probability of 0.997 (the confidence interval corresponding to 3σ) that the real node position will lay in the circle of radius 3σ around the computed position x'_u, y'_u.

Within the VM this means that the maximal attacker impact on the computed position is upper-bounded by 3σ.

To estimate the expected σ of an UWB positioning system, we used the results of the distance measurements between UWB base stations and UWB tags, published by Multispectral Solutions [1]. These results show that for indoor positioning, the standard deviation of the measured distances increases with the distance length. The measured distances were from 0 to 50 m, and the standard deviation was from 0 to 1 m. From these distance measurements, we computed the standard deviation of the positions within a verifiable triangle formed by the three base stations. The results of this computation are shown on Fig. 7. The values on the x and y axis denote the measured positions,

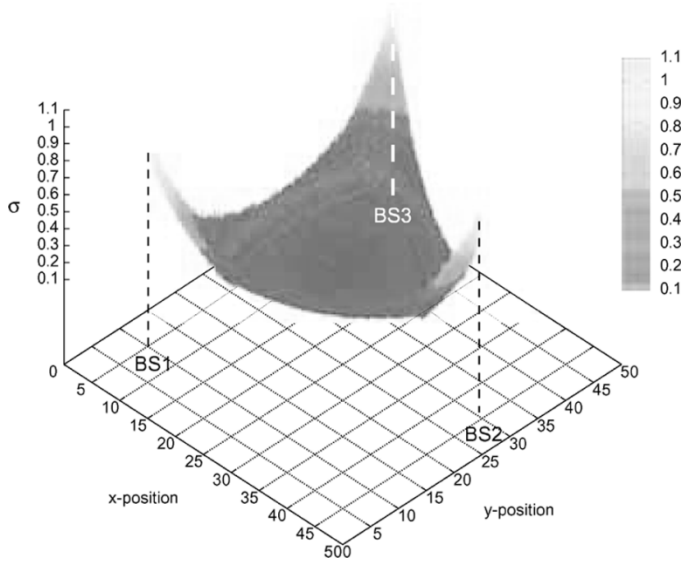

Fig. 7. Standard deviation (σ) of the position computed with MMSE of distance measurements to a UWB tag, performed by three UWB base stations. Positioning is performed within a triangle formed by the three base stations.

and the z axis denotes the standard deviation of the position. We observe that the standard deviation is the highest at positions close to the base stations. This is an expected result, due to geometric dilution of precision. It is also important to observe that the value of σ is lower than 0.5 m in the center part of the triangle, and that it increases to 1 m for positions closer to the base stations.

Given that the verifiers do not know a priory if the position that they computed is correct or not, VM cannot operate with δ, as it depends on the computed position. This is notably because we do not want to give any advantage to the attacker by allowing him to modify the distances (the position) in order to influence the choice of δ. VM, therefore, needs to operate in a "worst case" scenario with a fixed value for δ. This also means that δ needs to be chosen such that the positions which are not spoofed are not likely to be rejected, and that the positions which are spoofed are detected.

It is important to notice that by choosing δ, the verifiers are sure that the positions at which $3\sigma \leq \delta$ will not be rejected if there was no attack on the distances. This means that by choosing different δs, the verifiers will modify the verification area; for larger δ, the verification region will be larger, but so will be the maximum attacker impact.

D. Location Privacy

So far, we have described the infrastructure-based verifiable multilateration (IB-VM), in which the verifiers compute the po-

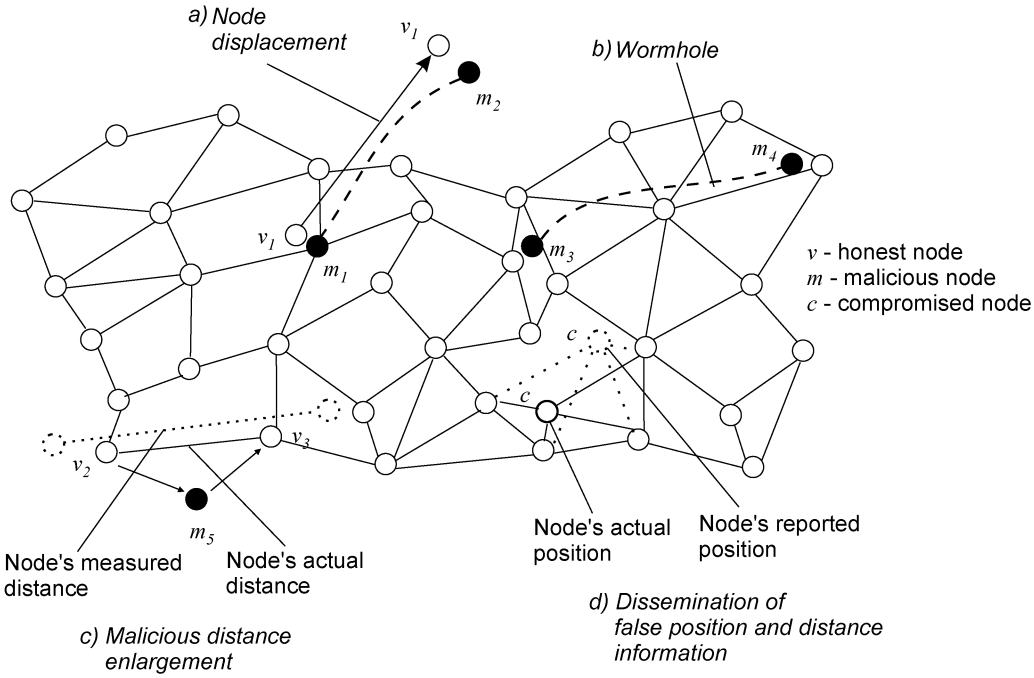

Fig. 8. Attacks on sensor network positioning.

sition of the claimant. IB-VM does not preserve the claimant's location privacy. There are two reasons for that. The first reason is that the verifiers compute the position of the claimant and, therefore, have full knowledge of where the claimant is located. The second reason is that the described distance bounding and ranging protocols are vulnerable to attacks by fake verifiers whose goal is to detect the position of a node by initiating the execution of the distance-bounding protocol. This second problem can be eliminated If the verifiers are authenticated to the claimants prior to the distance verification. In [38], Capkun *et al.*, proposed a protocol for mutually authenticated distance bounding (MAD) that enables two nodes to determine their mutual distance bounds at the time of encounter. This protocol can be used to prevent attacks by fake verifiers.

Still, even with mutual claimant-verifier authentication IB-VM does not fully protect the claimant's location privacy, because the infrastructure knows the location of the claimant. This problem can be solved through node-based verifiable multilateration (NB-VM). In this protocol, the claimant performs distance bounding to the verifiers, and computes its location within the verification triangle in the same way as in the protocol in Fig. 4. Here, the claimant trusts the verifiers about their positions, but not does allow them to find out its position. However, the verifiers could try to infer the claimant's position based on the readings of the strengths of the signals received from the claimant. This, and similar attacks on node's location privacy have been previously investigated [15], [20], [21], [31], [33], [34], but thwarting these attacks is out of the scope of this paper.

V. Secure Positioning in Sensor Networks

In this section, we review attacks on sensor network positioning systems. We then propose solutions for secure positioning in sensor networks.

A. Threat Analysis

Threats on positioning in sensor networks are more severe than if positioning is performed directly by trusted base stations. This is because of the distributed nature of the majority of sensor network positioning algorithms. Namely, to reduce the number of trusted base stations needed to position all the sensors in the network, most of the positioning algorithms rely on sensor cooperation for computing node positions.

One of the most obvious threats to sensor networks is the physical displacement of nodes. An external attacker can physically displace nodes from their original positions to other positions in the network, or can temporarily or permanently remove the nodes from the network while this remains undetected to the nodes or to the network authority. If the network is not properly protected, an attacker can create the impression to the displaced node and to its neighbors that the node did not move; a simple approach for the attacker is to create a communication link to the new position of the honest node. This attack, that we call the *node displacement* attack is illustrated in Fig. 8, case *a*).

Even without displacing the nodes, external attackers can still perform a number of attacks on node positions and network topology. An example of this behavior is the *wormhole attack* shown in Fig. 8, case *b*), by which the attacker establishes links between nodes that are not in each others' power range. Besides the establishment of new links, attackers can permanently or temporarily jam the communication between pairs of nodes and, thus, by remove links that would normally exist. Furthermore, an attacker can jam and replay the messages between the nodes and, therefore, enlarge the distances between the nodes (Fig. 8, case *c*).

Attacks by internal attackers are simpler to perform and can be more harmful than those performed by external attackers. Internal attackers can modify the computed network topology by reporting nonexisting links, or by not establishing or not

reporting the links that would normally be established. The *false position and distance dissemination attack* is illustrated in Fig. 8, case *d*).

B. System Model

Our secure positioning system consists of a set of sensor nodes and a set of reference nodes (landmarks) with known locations. Nodes and verifiers communicate using radio transmissions. We assume that the radio link between neighbors is bidirectional.

We assume that the sensor nodes have distance-measuring capabilities, but are not equipped with GPS receivers. The nodes are able to measure the distances to their neighbors or to the landmarks by using time-of-arrival or round-trip time measurements with radio signals. The nodes are also able to bound their processing delays to a few nanoseconds.

We assume that the network is operated by an authority. The authority controls the network membership and assigns a unique identity to each node. Each node is able to generate symmetric cryptographic keys and, more generally, to accomplish any task required to secure its communications. All network nodes can establish pairwise secret keys. This can be achieved by manually preloading all keys into the nodes in a network setup phase, by probabilistic key predistribution schemes [7], [11], or through an on-line key distribution center [18].

We observe two scenarios. In the first scenario, sensors are positioned directly by the landmark stations. In the second scenario, the sensor cooperate to compute their positions and the number of landmarks is significantly smaller.

C. Direct Sensor Positioning

If the sensors are being positioned directly by the landmark stations (verifiers), secure positioning is straightforward through the application of VM. This can be enabled by a network of landmarks which can be fixed, with predetermined positions, randomly distributed over the area of interest, or even mobile. The number of verifiers needed to cover an area, such that position verification can be performed in the whole area, depends on the number of verifiers and their (and mobile nodes') power ranges. So far, we have assumed that the power range of each verifier can cover the verification triangle and that the position verification is, thus, enabled in the whole triangle. This is, however, not true in general; the verification triangle is the largest possible region in which three verifiers can verify node positions. If the power ranges of the verifiers are such that they do not cover the whole triangle, the verification region can be smaller than the verification triangle. Only if the verifiers are in each others' power ranges will the verification region be equal to the verification triangle.

For this reason, the optimal way to cover an area of interest is to place verifiers within the area such that they form regular triangles with sides equal to their power ranges. In this case, the number n of verifiers needed to cover a square area of $L \times L$ is

$$n = \frac{\left[\frac{2L}{R} + 3\right]\left[\frac{2L}{R} + 1\right]}{2}$$

where L is the area width and length and R is the power range of the verifiers and mobile nodes. In this way, each verifier (except for the boundary verifiers) will be a verifier in six triangles (i.e., in a hexagon).

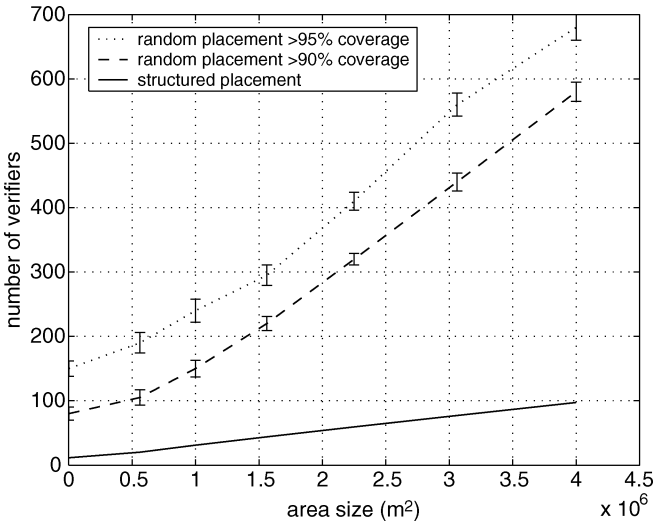

Fig. 9. Number of verifiers required to cover an area ($L \times L$) with verification triangles. The power range is 250 m.

We now consider the case in which, instead of being predeployed on fixed locations, the verifiers are uniformly distributed over the area of interest. We performed simulations to determine the number of verifiers necessary to cover the area. This coverage will depend on the sizes and the positions of the verification triangles formed by the verifiers. Our simulations were performed on areas of variable sizes (from 500×500 to 2000×2000 m^2 with verifiers power ranges of 250 m). To avoid boundary effects, the verifiers were uniformly distributed in the area and in a boundary region outside the area, whose width was 10% of the area width.

The results of an average of 100 simulations are shown in Fig. 9 and are displayed with confidence intervals of 95%. As expected, an optimal placement of verifiers is much more efficient than their random placement, in terms of number of verifiers.

However, for security purposes, in some scenarios, it might be advantageous for the verifiers to be randomly placed, to randomly move within the area of interest and, thus, not to have their positions known at all times. Verifier mobility could also prevent the cloning attack. If the sensors are mobile, their trajectories can be reconstructed based on the verified positions [17].

D. Cooperative Positioning: SPINE

In some application scenarios, a larger number of landmarks cannot be deployed, or is expensive to deploy. For those scenarios, we propose a secure cooperative positioning mechanism called SPINE: Secure Positioning for sensor NEtworks algorithm. SPINE is based on VM. The algorithm is executed in three phases: 1) the sensors measure distance bounds to their neighbors; 2) the distance bounds are verified through VM; and 3) the positions of the nodes are computed by the sensors using a distributed algorithm, or by the central authority, using a centralized positioning algorithm. SPINE algorithm is shown on Fig. 10.

BDV stands for *Basic Distance Verification (BDV)* (Fig. 11); it relies on VM. The BDV of the distance between v and u is performed by: 1) forming verification triangles around u with

$\mathcal{VD} = \{\emptyset\}$; set of verifiable distance bounds
$\mathcal{NV} = \{\emptyset\}$; set of non-verifiable distance bounds
$\mathcal{DB} = \{$all distance bounds$\}$
For all distances $db_i \in \mathcal{DB}$
  if db_i can be verified with BDV *then* $\mathcal{VD} = \mathcal{VD} \cup \{db_i\}$
  else $\mathcal{NV} = \mathcal{NV} \cup \{db_i\}$
Compute the positions of the nodes with $db_i \in \mathcal{VD}$
Compare the computed positions with $db_i \in \mathcal{NV}$

Fig. 10. SPINE algorithm.

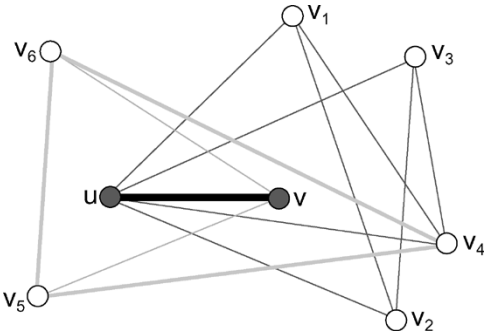

Fig. 11. Basic distance verification (BDV). To verify a distance, a set of triangles is formed around the distance.

v and its neighbors; 2) by forming verification triangles around v with u and its neighbors; and 3) by forming verifiable triangles around u and v. In our example, the following triangles are formed around v: $\triangle(u, v_1, v_2)$, $\triangle(u, v_3, v_4)$, $\triangle(u, v_1, v_4)$, and $\triangle(u, v_2, v_3)$; only a single triangle $\triangle(v, v_5, v_6)$ is formed around u. Finally, a triangle $\triangle(v_4, v_5, v_6)$ is formed around both u and v. After forming the triangles, the measured distance bounds db_{uv} (from u to v) and db_{vu} (from v to u) are verified in all triangles, by performing VM over u and v, respectively. This is done in such a way that the nodes forming a triangle define a local coordinate system, in which they then compute the position of u or v, or the positions of both u and v. The computation of the position of u and v is performed with VM through which the distance bounds db_{uv} and db_{vu} are then verified. Verification of the distance bound is successful within BDV only if in all verification triangles the measured distance bounds db_{uv} and db_{vu} match the computed positions (with a tolerance of δ). The algorithm is executed as shown on Fig. 12.

The set $\mathcal{VD}$ contains those distance bounds that can be verified by at least one triangle. The distance bounds that cannot be verified are included into a set $\mathcal{VD}$ of nonverified distances. Once the selection process is finished, the positions of the nodes can be computed by using only verified distances from the set $\mathcal{VD}$. Finally, the computed positions of the nodes are compared with the nonverified distances from $\mathcal{NV}$.

The computation of the positions of the nodes can be performed by a number of centralized or distributed range-based positioning algorithms (see Section VI). Note here that the BDV algorithm can be executed locally as the nodes forming a triangle are in each other's power ranges.

The effectiveness of any of the used positioning techniques (and consequently of SPINE) depends on the number of node neighbors (node density) and on the number and the spatial distribution of landmarks. The number of node neighbors is crucial

u measures db_{uv} and v measures db_{vu}
1 Triangles are formed with v and its neighbors;
  $\mathcal{US}$ is the set of triangles enclosing u
2 Triangles are formed with u and its neighbors;
  $\mathcal{VS}$ is the set of triangles enclosing v
3 Triangles are formed with neighbors of u and v;
  $\mathcal{UVS}$ is the set of triangles enclosing u and v
4 *In all* $\triangle_\ell \in \mathcal{US} \cup \mathcal{VS} \cup \mathcal{UVS}$ compute d^ℓ_{uv} with VM
5 *If for all* $\triangle_\ell \in \mathcal{US} \cup \mathcal{VS} \cup \mathcal{UVS}$, $|d^\ell_{uv} - db_{uv}| \leq \delta$ and $|d^\ell_{vu} - db_{uv}| \leq \delta$, *then* $\{db_{uv}, db_{vu}\}$ are verified
  else db_{uv}, db_{vu} cannot be verified

Fig. 12. Basic distance verification (BDV) algorithm.

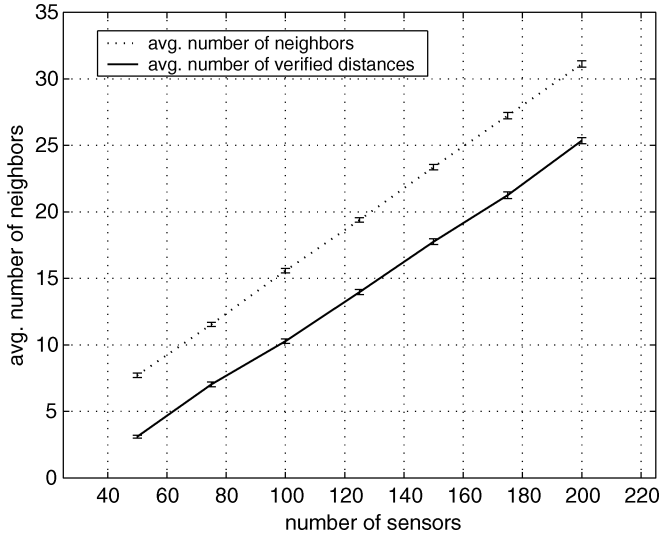

Fig. 13. Average number of neighbors per node and average number of verifiable distances adjacent to a node.

to ensure that the positions of most of the nodes can be computed. The requirements for secure positioning are higher: it is necessary that the network is sufficiently dense to ensure that the positions of most nodes can be *securely* computed.

To show the difference between the density requirements for secure and nonsecure positioning, we observe average number of distance bounds to the neighbors that can be verified with BDV (the distances that are used for secure positioning), and the average number of node neighbors (the distances used for nonsecure positioning). We performed simulations on an area of 100 × 100 m, with 50 to 500 uniform randomly distributed nodes with power ranges of 25 m. The results are presented in Fig. 13 with 95% confidence intervals.

As expected, the results show that to perform secure positioning equivalently to nonsecure positioning (meaning with approximately the same number of distances), a higher density of nodes is required. For nonsecure positioning, an average of ten distances per node (ten neighbors) is reached already with $80\ \text{nodes}/100 \times 100\ \text{m}^2$, whereas for secure positioning, an average of ten verifiable distances requires at least $110\ \text{nodes}/100 \times 100\ \text{m}^2$.

We further computed the average percentage of nodes covered by at least one verification triangle. These results are shown in Fig. 14. This figure is important as it shows that at node den-

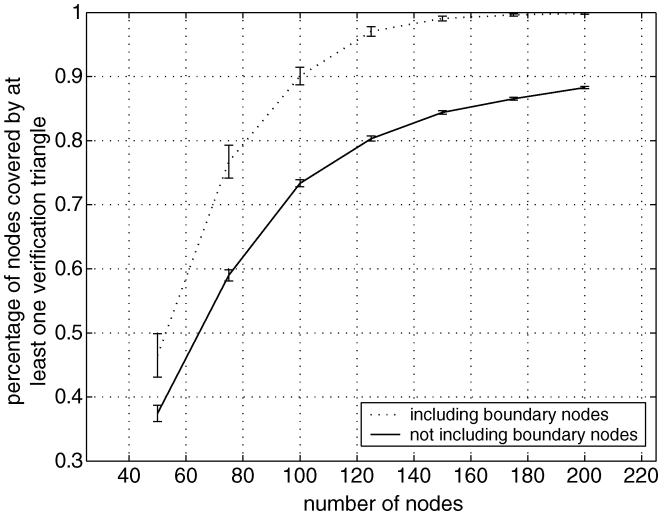

Fig. 14. Average percentage of nodes covered by at least one verification triangle, with and without boundary nodes.

sity of $120\ \text{nodes}/100 \times 100\ \text{m}^2$, most of the nodes are covered by at least one verification triangle, meaning that their adjacent distances and their position can be verified. As expected, the figure shows that the boundary nodes are not covered by verification triangles. This is an important indication that the landmark stations need to be specifically placed at the boundaries of the area to protect boundary nodes from attacks by enabling the formation of verification triangles around them.

E. Security Analysis

The resistance of SPINE relies on the resistance of BDV to attacks; it depends on the ability of the attacker to modify the verified distances, but also on the positioning algorithm used to compute node positions with verified distances.

Here, we primarily analyze the resistance of BDV to attacks. We then discuss security implications of using BDV with several positioning algorithms.

The resistance of BDV to attacks depends on the number and on the mutual dependence of triangles that are formed around the distance. To spoof a distance verified by a single triangle, it is sufficient for an external attacker to enlarge two distances (the distance d_{uv}, and one additional distance between the nodes forming a triangle). This is illustrated on Fig. 15, where distances d_{uv} and d_1 are enlarged. By enlarging these two distances, all the distances in the verification triangle remain mutually consistent. This attack can be performed by an external attacker.

If only a single node in a triangle is compromised, this node can enlarge distances to the claimant and to other nodes forming the verification triangle. This is illustrated on Fig. 16. In this example, node v is compromised, and enlarges distances to u, v_2, and to v_3 such that all the distances in the verification triangle remain mutually consistent. Similarly to the attack on Fig. 15, if the attacker controls one compromised and one external node, it can enlarge the measured distance even if the compromised node is not adjacent to the distance. This essentially means that a single-triangle BDV resists only to attacks that enlarge only a single distance.

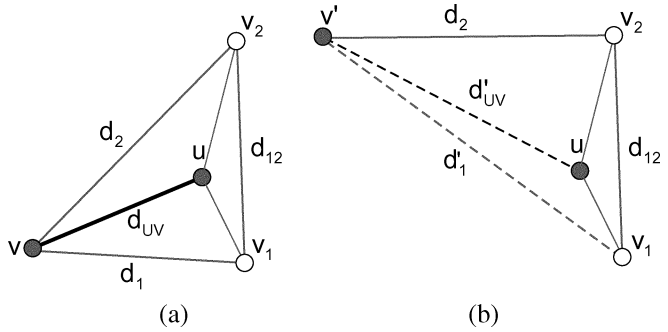

Fig. 15. Example of distance enlargement attack by external nodes on a single-triangle BDV. Distance d_{uv} (a) before enlargement and (b) after enlargement.

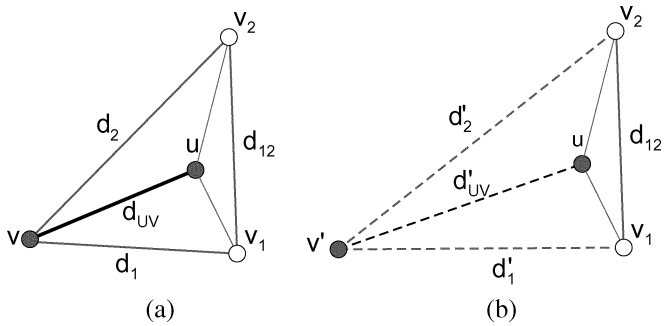

Fig. 16. Example of a distance enlargement attack by a compromised node (v) on a single-triangle BDV. Distance d_{uv} (a) before enlargement and (b) after enlargement.

If k verification triangles can be formed around a distance, the resistance of BDV to attacks can be expressed in terms of k. If the triangles are node-disjoint, then BDV resists to up to $2k$ distance enlargements. This is intuitive, as the distance is verified by k disjoint triangles, and an attacker needs to spoof the verification process in each of the triangles to successful cheat on the measured distance.

If the triangles are node-joint and edge-disjoint, then BDV also resists to up to $2k$ distance enlargements by external attackers, but it does not resist attacks by a single compromised node adjacent to the spoofed distance. Essentially, if all triangles have a common (compromised) node, the distance adjacent to that node can be successfully spoofed. We note here, however, that the triangles formed around a distance are almost never node-joint, given that some are formed with u and its neighbors around v, others are formed with v and its neighbors around u, whereas the third set of triangles is formed by the neighbors of u and v around the two nodes.

If the triangles are edge-joint, then BDV resists to up to $k + 1$ distance enlargements by external attackers. If the nodes are positioned favorably for the attacker, the attacker can enlarge the joint edge and enlarge one additional edge from every triangle. We note here that this attack will not always be possible.

Colluding internal (and external) attackers are the most serious threat to BDV. These attackers can modify arbitrarily the distances and help each other in providing consistently incorrect distance and position information. The number of such attackers needed to cheat on distances depends on the number of nodes forming triangles around a particular distance. Typically,

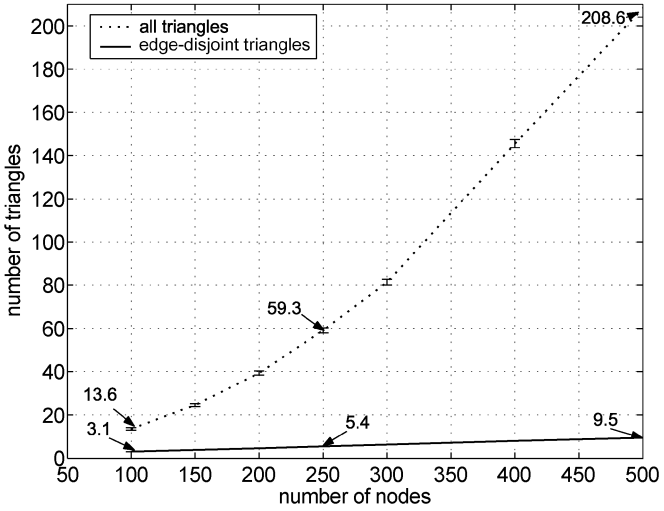

Fig. 17. Average number of verification triangles and an average number of edge-disjoint verification triangles that can be formed around a distance.

colluding internal attackers need to share mutual authentication material and need to be placed close to the verifiers to perform a successful attack.

We performed simulations on a network of sensors with densities from 50 to $500\ \mathrm{nodes}/100 \times 100\ \mathrm{m}^2$ and a power range of 25 m. We computed the average number of verification triangles and an average number of edge-disjoint verification triangles that can be formed around a distance (Fig. 17). The results show that BDV, depending on the node density and node positions, can resist to attacks up to 100 distance enlargements.

To compromise the computation of the position of a single node, an attacker needs to modify the computation and the verification of the (verified) distances surrounding the node. Furthermore, the attacker needs to make all the modified distances and positions consistent with the positions of other nodes in the network. The difficulty for the attacker here is in distance enlargement. Essentially, when the attacker enlarges distances, it makes some nodes to appear further from each other, but also makes some unavoidably to appear closer. This is why in a very dense network, the attacker could only scale-up all the distances in the network, but it would not be able to, by changing a smaller number of distances, successfully modify the computed positions of the nodes.

VI. Related Work

In the last decade, a number of indoor positioning systems were proposed, based notably on infrared [41], ultrasound [30], [42], received radio signal strength [3], [6], [16], and ToF radio signal propagation techniques [12], [25]. These positioning techniques were then extended and used for positioning in wireless ad hoc networks [5], [8]–[10], [28], [29], [32], [36], [39].

Recently, a number of secure distance and location verification have been proposed. Brands and Chaum [4] proposed a distance bounding protocol that can be used to verify the proximity of two devices connected by a wired link. Sastry *et al.* [35] proposed a new distance bounding protocol, based on ultrasound and radio wireless communication. In that work, the authors also propose to make use of multiple base stations to narrow down the area in which the nodes lie. However, as this proposal is based on ultrasound distance bounding, it can, therefore, be used only for the verification of nodes' positions, and only if external nodes have no access to the area of interest. In [19], the authors propose a mechanism called "packet leashes" that aims at preventing wormhole attacks by making use of the geographic location of the nodes (geographic leashes), or of the transmission time of the packet between the nodes (temporal leashes). Kuhn [23] proposed an asymmetric security mechanism for navigation signals. That proposal aims at securing systems like GPS [14]. Lazos *et al.* [24] proposed a set of techniques for secure positioning of a network of sensors based on directional antennas and distance bounding. Li *et al.* [26] propose statistical methods for securing localization in wireless sensor networks. Liu *et al.* [27] propose techniques for the detection of malicious attacks against beacon-based location discovery in sensor networks, based on consistency of received beacons. In [40], Capkun *et al.* propose a secure localization scheme based on hidden and mobile base stations, which makes use of the unpredictability of the base station locations.

Recently, a number of proposals have been made to protect the anonymity and location privacy of wireless devices [15], [20], [21], [31], [33], [34].

VII. Conclusion

In this paper, we have analyzed positioning and distance estimation techniques in adversarial settings. We have shown that most proposed positioning techniques are vulnerable to position spoofing attacks from internal and external attackers. We have further shown that positioning and distance estimation techniques, based on radio signal propagation, exhibit the best properties for position verification. We have proposed a novel mechanism for position verification, called VM. VM enables for the secure computation and verification of node positions in the presence of attackers. We have further proposed SPINE, a system for secure positioning in a network of sensors, based on VM. We have shown that this system resists against distance modification attacks from a large number of attacker nodes.

Our future work includes a detailed analysis and possible implementation of distance bounding and position verification techniques. Furthermore, we intend to investigate the applicability of our basic distance verification scheme to a number of existing positioning algorithms.

References

[1] Multispectral [Online]. Available: http://www.multispectral.com/
[2] R. Anderson and M. Kuhn, "Tamper resistance—A cautionary note," in *Proc. 2nd Usenix Workshop on Electronic Commerce*, 1996, pp. 1–11.
[3] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. INFOCOM*, vol. 2, 2000, pp. 775–784.
[4] S. Brands and D. Chaum, "Distance-bounding protocols," in *Proc. Workshop on the Theory and Application of Cryptographic Techniques on Advances in Cryptology*, 1994, pp. 344–359.
[5] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization for very small devices," *IEEE Pers. Commun. Mag.*, vol. 7, no. 5, pp. 28–34, Oct. 2000.
[6] P. Castro, P. Chiu, T. Kremenek, and R. Muntz, "A probabilistic room location service for wireless networked environments," in *Proc. 3rd Int. Conf. Atlanta Ubiquitous Computing (Ubicomp)*, vol. 2201, Sep. 2001, pp. 18–34.

[7] H. Chan, A. Perrig, and D. Song, "Random key predistribution schemes for sensor networks," in *Proc. IEEE Symp. Res. Security and Privacy*, May 2003, p. 197.
[8] H. Chan, M. Luk, and A. Perrig, "Using clustering information for sensor network localization," in *Proc. IEEE Conf. Distrib. Comput. Sensor Syst.*, Jun. 2005.
[9] L. Doherty, K. Pister, and L. El Ghaoui, "Convex position estimation in wireless sensor networks," in *Proc. INFOCOM*, Apr. 2001, pp. 1655–1663.
[10] T. Eren, D. Goldenberg, W. Whiteley, Y. Yang, A. Morse, B. Anderson, and P. Belhumeur, "Rigidity, computation, and randomization in network localization," in *Proc. INFOCOM*, 2004, pp. 2673–2684.
[11] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proc. ACM Conf. Comput. Commun. Security*, 2002, pp. 41–47.
[12] R. Fontana, "Experimental results from an ultra wideband precision geolocation system," *Ultra-Wideband, Short-Pulse Electromagnetics*, May 2000.
[13] R. Fontana, E. Richley, and J. Barney, "Commercialization of an ultra wideband precision asset location system," in *Proc. IEEE Conf. Ultra Wideband Syst. Technol.*, Nov. 2003, pp. 369–373.
[14] I. Getting, "The global positioning system," *IEEE Spectrum*, pp. 36–47, Dec. 1993.
[15] M. Gruteser and D. Grunwald, "Enhancing location privacy in wireless LAN through disposable interface identifiers: A quantitative analysis," in *Proc. WMASH*, 2003, pp. 46–55.
[16] J. Hightower, G. Boriello, and R. Want, "SpotON: An indoor 3D location sensing technology based on RF signal strength," Univ. Washington, Seattle, WA, Tech. Rep. 2000-02-02, 2000.
[17] L. Hu and D. Evans, "Localization for mobile sensor networks," in *Proc. MobiCom*, 2004, pp. 45–57.
[18] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Ariadne: A secure on-demand routing protocol for ad hoc networks," in *Proc. MobiCom*, Sep. 2002, pp. 12–23.
[19] ——, "Packet leashes: A defense against wormhole attacks in wireless networks," in *Proc. INFOCOM*, San Francisco, CA, Apr. 2003, pp. 1976–1986.
[20] L. Huang, K. Matsuura, H. Yamane, and K. Sezaki, "Enhancing wireless location privacy using silent period," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2005, pp. 1187–1192.
[21] J. Kong and X. Hong, "ANODR: Anonymous on demand routing with untraceable routes for mobile ad-hoc networks," in *Proc. MobiHoc*, 2003, pp. 291–302.
[22] J. Kong, Z. Ji, W. Wang, M. Gerla, R. Bagrodia, and B. Bhargava, "Low-cost attacks against packet delivery, localization and time synchronization services in underwater sensor networks," in *Proc. WiSe*, 2005, pp. 87–96.
[23] M. G. Kuhn, "An asymmetric security mechanism for navigation signals," in *Proc. Inf. Hiding Workshop*, 2004, pp. 239–252.
[24] L. Lazos, S. Čapkun, and R. Poovendran, "ROPE: Robust position estimation in wireless sensor networks," in *Proc. IPSN*, Apr. 2005, pp. 324–331.
[25] J.-Y. Lee and R. Scholtz, "Ranging in a dense multipath environment using an UWB radio link," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 9, pp. 1677–1683, Dec. 2002.
[26] Z. Li, W. Trappe, Y. Zhang, and B. Nath, "Robust statistical methods for securing wireless localization in sensor networks," in *Proc. Int. Conf. Inf. Process. Sensor Netw.*, 2005, pp. 91–98.
[27] D. Liu, P. Ning, and W. Du, "Attack-resistant location estimation in sensor networks," in *Proc. Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2005, pp. 99–106.
[28] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *Proc. SenSys.*, 2004, pp. 50–61.
[29] D. Niculescu and B. Nath, "DV based positioning in ad hoc networks," *J. Telecommun. Syst.*, vol. 22, no. 4, pp. 267–280, 2003.
[30] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket location-support system," in *Proc. MobiCom.*, 2000, pp. 32–43.
[31] A. R. Beresford and F. Stajano, "Location privacy in pervasive computing," *Pervasive Computing*, pp. 46–55, Jan.–Mar. 2003.
[32] D. Niculescu and B. Nath, "Ad hoc positioning system using AoA," in *Proc. INFOCOM*, 2003, pp. 1734–1743.
[33] I. W. Jackson, "Anonymous addresses and confidentiality of location," in *Proc. Int. Workshop on Information Hiding*, 1996, pp. 115–120.
[34] Y.-C. Hu and H. J. Wang, "Location privacy in wireless networks," in *Proc. ACM SIGCOMM Asia Workshop*, 2005.
[35] N. Sastry, U. Shankar, and D. Wagner, "Secure verification of location claims," in *Proc. WiSe*, Sep. 2003, pp. 1–10.
[36] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," in *Proc. MobiCom.*, 2001, pp. 166–179.
[37] D. Shaw and W. Kinsner, "Multifractal modeling of radio transmitter transients for classification," in *Proc. IEEE Confe. Commun., Power, Comput.*, May 1997, pp. 306–312.
[38] S. Čapkun, L. Buttyán, and J.-P. Hubaux, "SECTOR: Secure tracking of node encounters in multi-hop wireless networks," in *Proc. SASN*, Washington, Oct. 2003, pp. 21–32.
[39] S. Čapkun, M. Hamdi, and J.-P. Hubaux, "GPS-free positioning in mobile ad-hoc networks," *Cluster Comput.*, vol. 5, no. 2, pp. 157–167, Apr. 2002.
[40] S. Čapkun, M. Srivastava, and M. Čagalj, "Secure positioning with hidden and mobile base stations," NESL-UCLA, Tech. Rep., [Online]. Available: http://nesl.ee.ucla.edu/, Jun. 2005.
[41] R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The active badge location system," *ACM Trans. Inf. Syst.*, vol. 10, no. 1, pp. 91–102, 1992.
[42] A. Ward, A. Jones, and A. Hopper, "A new location technique for the active office," *IEEE Pers. Commun.*, vol. 4, no. 5, pp. 42–47, Oct. 1997.
[43] J. S. Warner and R. G. Johnston, "Think GPS cargo tracking=High security? Think again," Los Alamos National Lab., Los Alamos, NM, Tech. Rep., 2003.

Srdjan Čapkun (M'00) received the Dipl.Ing. degree in electrical engineering from the University of Split, Croatia, and the Ph.D. degree in communication systems from École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland.

He is an Assistant Professor at the Technical University of Denmark, Lyngby. He spent a year as a Postdoctoral Researcher at the University of California, Los Angeles (UCLA). His current research focuses on the design and the analysis of security protocols for wireless networks.

Dr. Čapkun is a member of the Association for Computing Machinery (ACM).

Jean-Pierre Hubaux (M'91–SM'95) was born in Belgium. He joined the faculty of École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland, in 1990. He was promoted to Full Professor in 1996. His research activity is focused on mobile networking and computing, with a special interest in wireless ad hoc and sensor networks. He served as the General Chair for the Third ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2002), held on the EPFL campus. He has been serving on the program committees of numerous conferences and workshops, including INFOCOM, MobiCom, MobiHoc, SenSys, WiSe, and VANET. He has held visiting positions at the IBM T. J. Watson Research Center and at the University of California at Berkeley. After completing his studies in electrical engineering at Politecnico di Milano, he worked ten years in France with Alcatel, where he was involved in R&D activities, primarily in the area of switching systems architecture and software. He has been strongly involved in the definition and launching phases of a new National Competence Center in Research named "Mobile Information and Communication Systems" (NCCR/MICS), since its genesis in 1999; this center is often nicknamed "the Terminodes project." In this framework, he has notably defined, in close collaboration with his students, novel schemes for the security and cooperation in fully self-organized mobile ad hoc networks; in particular, he has devised new techniques for key management, key establishment, and secure positioning in such networks. He has also made several contributions in the areas of power management in sensor networks and of group communication in ad hoc networks.

Dr. Hubaux is an Associate Editor of the IEEE TRANSACTIONS ON MOBILE COMPUTING, *Foundations and Trends in Networking*, and the *Journal on Ad Hoc Networks*.