Secure Localization and Location Verification in Wireless Sensor Networks

Yingpei Zeng† Jiannong Cao‡
†*State Key Laboratory for Novel Software Technology*
Nanjing University, Nanjing, P.R. China
Email: {zyp,jhong}@dislab.nju.edu.cn

Jue Hong† Li Xie†
‡*Department of Computing*
Hong Kong Polytechnic University, Hong Kong
Email: csjcao@comp.polyu.edu.hk, xieli@nju.edu.cn

Abstract

Sensors' locations are important to many wireless sensor networks (WSNs). When WSNs are deployed in hostile environments, two issues about sensors' locations need to be considered. First, the attackers may attack the localization process to make the estimated locations incorrect. Second, since sensor nodes may be compromised, the base station may not trust the locations reported by sensor nodes. Researchers have proposed two techniques, secure localization and location verification, to solve the two issues respectively. In this paper we describe the attacks against localization and location verification, and survey the state of research of both secure localization and location verification.

1. Introduction

Wireless sensor networks (WSNs) are composed of small, low cost, and low power sensor nodes [1]. Many applications have been proposed for WSNs, e.g., environmental applications like volcano monitoring and military applications like battlefield surveillance [1]. In a lot of applications, WSNs are deployed in unattended and even hostile environments, where we must consider the security issues to ensure the operation of the WSNs.

Sensors' locations are important in many WSNs. First, the events detected by sensors usually should bind with locations, e.g., a truck is detected *at location loc.* Second, many network operations also depend on the locations of sensors, e.g., geographic routing [2], geographic key distribution [3], and location-based authentication [4]. Now many localization algorithms for WSNs have been proposed [5].

When WSNs are deployed in hostile environments, the attackers may attack the localization process to make the estimated locations incorrect. Incorrect locations may lead to severe consequences, e.g., wrong military decisions on the battlefield and falsely granting access rights to people. Thus it is important to ensure the correctness of sensors' locations.

We should consider the correctness of sensors' locations from two aspects. *From the aspect of sensor*, since sensors themselves need to get their correct locations (e.g., to correctly tracking an object), so we need secure location determination, which we call secure localization in the paper. *From the aspect of the base station*, the base station (BS) also needs to ensure the sensors' locations it gets are correct (e.g., to make sure an event really happened there). However when the BS needs to learn sensors' locations from sensors (i.e., in node-centric localization as we will explain later), the sensor nodes may be compromised and they intentionally report false locations. Thus we need to verify the reported locations. We call this as location verification.

In this paper we first describe the secure localization problem and the location verification problem (Section 2), and review the known attacks in them (Section 3). Then we describe and classify the state of research in both secure localization (Section 4) and location verification (Section 5). Finally we present the conclusion and several open research problems (Section 6). Different from existing review articles [6], [7], we survey the two related fields, secure localization and location verification, at the same time to provide a more comprehensive review.

2. Problem Statement

In the section we define the problems that localization, secure localization, and location verification try to solve.

2.1. Localization

Usually the sensor network contains two kinds of nodes: common nodes and beacon nodes. Common nodes do not know their locations, and beacon nodes know their locations (e.g., by GPS). Then, the localization process is to estimate the locations of the common nodes, and it can be divided into two steps (with an optional refinement step), as shown in Figure 1:

- *Information collection*: The information for localization is collected, which may include the connectivity, distances, and angles, as well as the locations of beacons. The distances between neighbor nodes can be measured by received signal strength indicator (RSSI), time of arrival (ToA), or time difference of arrival (TDoA) [8]; the distances between nodes multihop-away can be measured by DV-hop [9] or DV-distance [9]. The angles can be measured by angle of arrival (AoA) [10].

978-1-4244-5113-5/09/$25.00

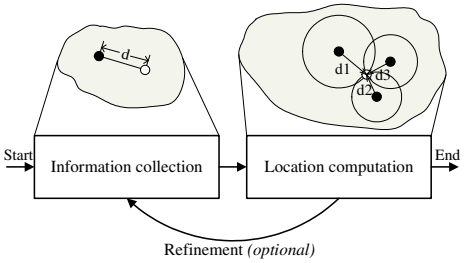

Figure 1. The localization of sensors.

- *Location computation*: The locations are computed with the collected information. Simple algorithms include trilateration [9], multilateration [8], and triangulation [10]. More complicated algorithms include MDS-MAP for localizing the network as a whole [11] and RobustQuad for coping with noisy measurements [12].

The optional refinement step is for iteratively computing locations, with newly calculated locations (e.g., the localized node will become new beacons [8]) or with new computation methods (e.g., in [13]–[15], new methods will be executed after obtaining nodes' coarse locations).

The localization systems can be classified into *range-based* and *range-free*. In range-based systems, the distances or angles between nodes need to be measured in the information-collection step. Range-free systems do not have such requirements.

The localization systems can also be classified into *node-centric* and *infrastructure-centric* [16], [17]. In the former sensor nodes compute their locations by themselves. In the latter the infrastructure (*we refer to the infrastructure as the BS and any other nodes the BS trusted, e.g., special mobile stations*) computes the locations of nodes.

2.2. Secure Localization

Secure localization is to make the above localization process still correct under attacks. The classification of secure localization systems may also follow the classification of general localization systems in the above subsection. We next briefly describe the adversary model in secure localization.

Adversary model: The goal of the adversary is to make the nodes (i.e., in node-centric localization) or the infrastructure (i.e., in infrastructure-centric localization) obtain false locations. He can compromise partial nodes (common nodes and beacons). He can intercept, jam, and replay signals in any transmission medium.

2.3. Location Verification

The infrastructure may not trust sensors' claimed locations in node-centric localization systems [1]. First, if the localization system is infrastructure-centric, the infrastructure will trust the estimation locations and no verification is needed, because the locations are computed by itself (the locations may also be incorrect, but securing the localization is the only thing it can do). Second, however, if the localization system is node-centric, even the locations are obtained through secure localization, the nodes may be compromised and may intentionally report false locations. Adding tamper-resistant hardware for honestly reporting locations is an approach; however it will increase the cost of node and also is proved to be problematic in practice [18].

Thus, when localization system is node-centric, location verification is needed to verify the claimed locations of sensors. A sensor node to be verified is called the *prover* and the infrastructure is called the *verifier*. We next briefly describe the adversary model in location verification.

Adversary model: The goal of the adversary is to make the verification failed, i.e., correct location claims from normal provers are verified as incorrect and are rejected, but false location claims from compromised provers are verified as correct and are accepted. Similar to secure localization systems, the adversary can compromise partial nodes (common nodes and beacons). He can also intercept, jam, and replay signals in any transmission medium.

3. Known Attacks

Many attacks can be launched against localization and location verification systems. We roughly classify them to simple attacks and complicated attacks based on the attack strength. In simple attacks, each attacker controls no more than one node and attacker are not cooperating. In complicated attacks, the attackers may compromise many nodes and are cooperating.

3.1. Simple Attacks

Range-change Attack: In this attack the attacker may change the range and AoA measurements between normal nodes and compromised nodes [43]. For example, if the measurement is RSSI-based, the attacker can increase or decrease the compromised node's transmission power. This attack has effects on both localization and location verification systems. For example, reducing the range measurement between node A and B may distort the estimated location of B if A is a beacon, and may also make A wrongly believe that B is within a given region if A is a verifier.

Impersonation: In this attack the attacker impersonates other nodes in the network. For example, in localization systems, the attacker may impersonate beacon nodes to broadcast false locations, and in location verification systems, the attacker may impersonate a victim prover to make the verifier believe the prover is at the attacker's location. This attack can be defeated by authentication.

1. Sensors using other sensors' locations may also not trust other sensors' claimed locations, however they usually trust the infrastructure. Thus it is not a problem when sensors' locations are verified by the infrastructure.

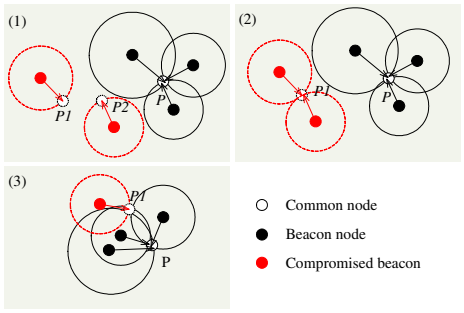

Figure 2. Three types of location-reference attacks: (1) uncoordinated, (2) collusion, and (3) pollution attacks. In the figure only P is the real location.

3.2. Complicated Attacks

Wormhole attack: In this attack the attacker records packets at one location in the network, tunnels them to another location, and replays them [19]. In localization systems, wormhole attack will make the beacons in one side appear at another side and make the information collected erroneous. In location verification systems, the attack may tunnel the packets of a prover to another location and make the verifier believe that the prover is at the false location.

Sybil attack: In this attack the attacker has obtained several node identities, and then he can make one compromised node masquerade as several nodes at the same time. For example, in localization systems, one compromised node may masquerade as several beacons (their identities are compromised by the attacker), and send false information.

Location-reference attack: This attack is against localization systems in which each common node gets a location-reference set for localization (e.g., in [8], [9], [13]), and the attack is to change partial location references [20]. According to the smart level, the attack can be classified into three types: *uncoordinated attack*, *collusion attack*, and *pollution attack*. Exemplary scenarios are shown in Figure 2. *In uncoordinated attack*, different bad location references are to mislead the common node to different false locations, e.g., $P1$ and $P2$ in the figure. *In collusion attack*, all bad location references are to mislead the common node to a random but the same false location. This attack is more powerful, but it is still can be defeated when normal location references are in the majority [21]. *In pollution attack*, all bad location references are to mislead the common node to a specially chosen false location, which also conforms to partial normal location references. This attack may succeed even when normal location references are in the majority [22].

4. Solutions for Secure Localization

Many secure localization systems have been proposed. As we mentioned they can be classified into two types, node-centric and infrastructure-centric.

Based on their goals, existing solutions can be classified into three methods: 1) Prevent the adversary from producing erroneous information (*the prevention method*), 2) Detect and revoke the nodes that producing erroneous information (*the detection method*), and 3) Filter the received erroneous information in location computation (*the filtering method*).

4.1. Node-centric Secure Localization

The prevention method: Researchers proposed several solutions following the prevention method [23]–[27]. In SeRLoc [23], Lazos et al. used trusted nodes called locators to replace beacons. The locators are equipped with sectored antennas and have longer transmission range. When a node hears multiple locators, it computes the center of gravity of the sectors corresponding to locators as its location. The same authors latter proposed an improved method HiRLoc [24], which can achieve higher accuracy through rotatable antennas and variable transmission power.

In [25], [26], Capkun et al. proposed SPINE based on the verifiable multilateration (VM) technique introduced in the same paper. In VM, if a node is inside the triangle formed by three nodes with known locations, through distance bounding [28], its location can be uniquely determined. In SPINE, all the distance measurements are verified by VM triangles around them formed by sensor nodes, so nodes cannot produce erroneous distance measurements.

In [27], combining the techniques in SeRLoc [23] and VM [25], [26], Lazos proposed ROPE. In ROPE each node obtains its exact location by VM when it is inside at least one triangle formed by locators, and still estimates its location by center of gravity when it is not inside any triangle. In [29], Zeng et al. proposed SHOLOC to prevent the compromised nodes from reducing the hop counts in hop-count based localization algorithm. Their method is to represent the value of hop count by the number of hash operations on a nonce.

The detection method: Two solutions have been proposed in this category [30]–[32], and they both focus on detecting malicious beacons. In [30], Liu et al. proposed to use detecting beacons to detect malicious beacons broadcasting false locations. The detecting beacons pretend to be common nodes and send requests to beacons. Next they compare the distances computed by using their locations and the replied locations with the measured distances. If the distances are inconsistent, the beacons being checked are malicious and will be revoked.

In DRBTS [31], [32], Srinivasan et al. generalized the solution by Liu et al. [30] by employing beacons to maintain reputations for their neighbor beacons. Each beacon computes reputations of its neighbor beacons based on the overheard location reply as well as the reputation value heard from other beacons. Common sensor nodes will only use beacons trusted by other beacons to compute its location.

The filtering method: Many works have been done for filtering the impact of erroneous information [20], [21], [33]–[37]. They all focus on filtering the bad location references in location-reference set. In [20], [33] Liu et al. proposed ARMMSE and a voting-based algorithm. The ARMMSE is to obtain a subset of location references, which satisfies that the mean square error of the location computed by the subset is below a threshold. In the voting-based algorithm, each location reference votes to the divided cells according to its observation, and the centroid of the current cell(s) with the highest vote is the estimated location.

In [21] Li et al. proposed to use LMS [38] to filter the bad location references. Different from traditional methods that minimize the mean square error, LMS method is to minimize the median of square errors: $loc_0 = \arg\min_{loc_0} med_i[dist(loc_i, loc_0) - d_i]^2$, where $< loc_i, d_i >$ is the i location reference and loc_0 is the estimated location.

In [34], [35], Misra et al. proposed a method to filter compromised beacons when distance bounding [28] is used and the attackers can only enlarge the distance measurements. Their method is to compute the geometric center of the intersection of circles corresponding to location references.

In [37] Zhong et al. proved that when there are no more than $\frac{n-3}{2}$ compromised beacons (i.e., $k \leq \frac{n-3}{2}$) [2], we can definitely compute the location of node with an error bound, where n and k are the number of total and normal beacons respectively. However, such result is proved under the condition that ϵ (i.e., the measurement error) is *ideally small*; in [22] Zeng et al. showed that the adversary can still seriously distort the estimated location when $k \leq \frac{n-3}{2}$ holds and ϵ is *practically small*. In [37] Zhong et al. also proposed two algorithms to compute the location, based on finding a region inside $k + 3$ rings.

4.2. Infrastructure-centric Secure Localization

Infrastructure-centric localization systems usually follow the prevention method, since they have reliable infrastructure (no vulnerable beacon nodes). Capkun et al. [16], [17] proposed a method to localize nodes based on covert base stations (CBS). First, the public base station (PBS) sends a nonce. When a node replies to the nonce, all the CBS will compute its location together based on the TDoA method. Then if the sum of the actual time differences deviates from the supposed values over a threshold, an attack is detected and the estimated location is rejected.

Zhang et al. [39] proposed SLS for UWB sensor networks. The authors assume that there is a set of trusted anchors which can perform group movement in the deployment field. In SLS, first, each anchor performs an algorithm called K-Distance to measure the distance between the anchor and the node to be localized. Second, anchors send the measured distances to the anchor leader to compute node's location. Third, SLS employs a location validity test by checking whether the location is inside the polygon formed by all the anchors. This test is more general than VM [25], [26] since the polygon is not limited to triangle.

Anjum et al. [40] proposed SLA to securely localize nodes based on transmission range (TR) variation. Here the anchors are assumed to be reliable and can vary their TR to several values. In the localization the BS let anchors transmit different nonces with different TRs. Each sensor then sends its received nonces to the BS. The BS computes sensors' locations based on the unique sets of nonces.

2. It is equal to say the condition $g \geq k + 3$ should hold, where g is the number of compromised beacons. In [34] a similar result is proved.

Table 1. Secure localization systems comparison.

	Prevention method	**Detection method**	**Filtering method**	**Additional hardware**
Node-centric	[23]–[27], [29]	[30]–[32]	[20], [21], [24], [29], [33]–[37]	[23]–[27], [30]
Infrastructure-centric	[16], [17], [39], [40]			[16], [17], [39], [40]

4.3. Comparison of Secure Localization Solutions

We list the classification of existing solutions in Table 1. Secure localization systems purely following the filtering method usually do not need any additional hardware. Some solutions following other methods also do not need additional hardware [29], [31], [32]. In contrast, infrastructure-centric secure localization systems always need to deploy new reliable infrastructure; however, they also have advantages, e.g., there is no need for location verification.

The three methods are from radical to conservative, and they may operate in defend-in-depth manner. For example, a system can use the the prevention, detection, and filtering methods as the first, second, and last lines of defense respectively. In fact, some schemes already combine more than one method in their design [24], [29].

5. Solutions for Location Verification

Based on the goals of verification, we classify the existing location-verification solutions into two types: *in-region* [4], [27], [28], [41], [42] and *single-position* [16], [17], [43]–[48]. In-region and single-position solutions verify that whether nodes (provers), are inside given regions (e.g., inside a cafe), and are at given positions respectively.

5.1. In-region Verification

Several solutions are proposed based on the distance-bounding technique [28]. Brands and Chaum [28] first

proposed distance bounding to make the prover (P) unable to reduce its distance to the verifier (V). The bounding process is: V sends bit α_i to P, and P sends bit $\beta_i = \alpha_i \oplus m_i$ to V immediately after he receives α_i. After that, V can compute an upper-bound on its distance to P based on the maximum of delay times between sending out a bit and receiving a bit back. The bounding using RF (radio frequency) signal requires dedicated hardware [25] (because V needs to measure time with nanosecond precision).

In [4] Sastry et al. proposed the Echo protocol, in which each verifier is in charge of the verification of a small circular region. To verify a prover P that is inside its circular region, the verifier V sends a nonce to P *using RF* and starts the timer, and the prover P immediately echoes the nonce back *using ultrasound*. Then V can use the elapsed time to compute the distance between them. The Echo protocol is similar to the distance bounding protocol [28] but it does not require sophisticate hardware (need no precise clock).

In [42] Vora et al. proposed a new method not based on distance bounding. They divided the verifiers into acceptors and rejectors. The acceptors and rejectors are deployed inside and at the boundary of protected region respectively. The verification process is the prover step by step increases its signal strength and broadcasts a signal, until a verifier hears the signal and responds. The verifiers accept the prover if none of the rejectors heard the prover during the process.

5.2. Single-position Verification

Base on the number of nodes verified at a time, we can further classify the verification algorithms into two types: *batch-verification* [47], [48] and *single-node-verification* [16], [17], [43]–[46]. The former verifies a batch of nodes at a time, and the latter verifies nodes one by one.

Batch-verification: In [48] Wei et al. proposed two algorithms, GFM and TI, to detect abnormal sensor locations. The GFM detects abnormal locations based four matrices which represent the neighborhood observed and neighborhood computed by estimated locations. In Ti, each node observing a node i continues to give a indicator value in respect of i (represent whether the node i has abnormal location). TI accepts a node's location when the node's final indicator value is greater than a threshold.

In [47] Hwang et al. proposed an algorithm for each node to detect the phantom nodes in its neighborhood. The node first creates a local map randomly using two other neighbors. Then in each such map, we try to find the largest consistent subset by checking each node that whether its measured ranges are consistent with its ranges in the map. The above process is repeated for given times and the largest subset in all the runs is selected, which contains all the normal nodes.

Single-node-verification: In [43] Du et al. proposed LAD, which is to use deployment information to detect localization anomaly. When sensors are deployed in groups, each node follows two-dimensional Gaussian distribution, which is centered at the deployment point of the node's group. Then the authors proposed three metrics to detect anomaly. Take the Diff metric for example, it represents the difference between the actual observation and the expected observation (an observation is a vector, in which the i value represents the number of neighbors in the i group).

In [16], [17] Capkun et al. proposed to use covert base stations (CBS) and mobile base station (MBS) to verify nodes' locations. In the CBS case, the node first broadcasts a RF signal and a sound signal. Then each CBS can calculate the distance between the CBS and the node. The CBS compares the calculated distance with the distance computed using node's reported location and CBS' location, and rejects the location if the difference is beyond a threshold. In the MBS case, the MBS first requires the node to broadcast RF and sound signals after given time. After that time, it has moved to a different location not known by the node, and can check the location similarly as a CBS.

In [45], [46] Ekici et al. proposed to verify a node's location with trusted verifiers. The node first floods its location in the network, with a hop count field. Each verifier can get the distance and hop count between the node and the verifier (a value pair). Then each verifier computes two probabilities: one represents the probability such value pair occurs with, and another represents the verifier's confidence. Finally a central node collects the information from all verifiers and decides to accept or reject the location.

Table 2. Location verification systems comparison.

	Batch-verification	**Single-node-verification**	**Additional hardware**
In-region		[4], [28], [42]	[4], [28], [42]
Single-location	[47], [48]	[16], [17], [43]–[46]	[16], [17], [45], [46]

5.3. Comparison of Location Verification Solutions

We list the classification of existing solutions in Table 2. Some single-position verification algorithms do not need any additional hardware [43], [47], [48]; however, in-region verification algorithms usually need additional hardware to represent the region to be protected or verified.

Single-node-verification systems usually are more efficient than batch-verification systems when we want to verify some critical nodes, e.g., the nodes which just reported events. However batch-verification systems are more appropriate when we want to verify all the nodes at one time.

6. Conclusion and Open Research Problems

In this paper we described the problems that secure localization and location verification try to solve. We also

discussed the known attacks in localization and location verification. Finally we described and classified existing solutions in both secure localization and location verification.

A number of research problems remain in the area of secure localization and location verification. First, no secure solution exists for localization in multihop & range-based systems (e.g., RobustQuad [12] and Sweep [5]). Collecting information through multipath may be a plausible way. Second, very few works exist for secure localization in some special WSNs, e.g., sparse WSNs [5] and mobile WSNs [49]. Third, no solution exists for location verification for one node at a time (i.e., single-node-verification) without any additional infrastructure and any deployment information. Possible solutions may utilize within-n-hop neighbors.

Acknowledgment

This work is supported in part by Hong Kong Research Grant Council under CERG grant PolyU 5102/07E, the Hong Kong Polytechnic University under the ICRG grant G-YF61, and Natural Science Foundation of China under Grant No.60673154, and Natural Science Foundation of Jiangsu Province under Grant "Research and Realization on ASLR in operating systems". A longer version of this paper is at [50].

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, p. 393C422, 2002.

[2] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of MobiCom*, 2000.

[3] D. Liu and P. Ning, "Location-based pairwise key establishments for static sensor networks," in *Proceedings of ACM SASN*, 2003.

[4] N. Sastry, U. Shankar, and D. Wagner, "Secure verification of location claims," in *Proceedings of WiSe*, September 19 2003.

[5] D. K. Goldenberg, P. Bihler, M. Cao, J. Fang, B. D. O. Anderson, A. S. Morse, and Y. R. Yang, "Localization in sparse networks using sweeps," in *Proceedings of MobiCom*, 2006, pp. 110–121.

[6] A. Boukerche, H. Oliveira, E. Nakamura, and A. Loureiro, "Secure localization algorithms for wireless sensor networks," *IEEE Communications Magazine*, vol. 46, no. 4, pp. 96–101, April 2008.

[7] A.-T. Ferreres, B. Alvarez, and A. Garnacho, "Guaranteeing the authenticity of location information," *IEEE Personal Commun. Mag.*, vol. 7, no. 3, pp. 72–80, July-Sept. 2008.

[8] A. Savvides, C.-C. Han, and M. Srivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," in *Proceedings of MobiCom*, Rome,Italy, 2001.

[9] D. Niculescu and B. Nath, "Ad hoc positioning system (APS)," in *Proceedings of IEEE GLOBECOM*, 2001.

[10] ——, "Ad hoc positioning system (aps) using aoa," in *Proceedings of INFOCOM*, San Francisco, CA, April 2003.

[11] Y. Shang, W. Ruml, and Y. Zhang, "Localization from mere connectivity," in *Proceedings of MobiHoc*, 2003.

[12] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *Proceedings of SenSys*, 2004.

[13] C. Savarese and J. Rabay, "Robust positioning algorithms for distributed ad-hoc wireless sensor networks," in *Proceedings of USENIX*, 2002.

[14] A. Savvides, H. Park, and M. B. Srivastava, "The bits and flops of the n-hop multilateration primitive for node localization problems," in *Proceedings of WSNA*, 2002, pp. 112–121.

[15] J. Liu, Y. Zhang, and F. Zhao, "Robust distributed node localization with error management," in *Proceedings of MobiHoc*, 2006, pp. 250–261.

[16] S. Čapkun, M. Cagalj, and M. Srivastava, "Securing localization with hidden and mobile base stations," in *Proceedings of INFOCOM*, 2006.

[17] S. Capkun, K. Rasmussen, M. Cagalj, and M. Srivastava, "Secure location verification with hidden and mobile base stations," *IEEE Transactions on Mobile Computing*, vol. 7, no. 4, pp. 470–483, 2008.

[18] R. Anderson and M. Kuhn, "Tamper resistance - a cautionary note," in *Proceedings of the Second Usenix Workshop on Electronic Commerce*, 1996.

[19] Y. Hu, A. Perrig, and D. Johnson, "Packet leashes: A defense against wormhole attacks in wireless ad hoc networks," in *Proceedings of INFOCOM*, 2003.

[20] D. Liu, P. Ning, and W. K. Du, "Attack-resistant location estimation in sensor networks," in *Proceedings of IPSN*, 2005.

[21] Z. Li, W. Trappe, Y. Zhang, and B. Nath, "Robust statistical methods for securing wireless localization in sensor networks," in *Proceedings of IPSN*, 2005.

[22] Y. Zeng, J. Cao, S. Zhang, S. Guo, and L. Xie, "Pollution attack: A new attack against localization in wireless sensor networks," in *Proceedings of WCNC*, 2009.

[23] L. Lazos and R. Poovendran, "SeRLoc: Secure range-independent localization for wireless sensor networks," in *Proceedings of ACM WiSe*, 2004.

[24] ——, "HiRLoc: high-resolution robust localization for wireless sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 233–246, Feb. 2006.

[25] S. Capkun and J.-P. Hubaux, "Secure positioning of wireless devices with application to sensor networks," in *Proceedings of INFOCOM*, 2005.

[26] S. Čapkun and J. P. Hubaux, "Secure positioning in wireless networks," *IEEE J. Sel. Areas Commun.*, 2006.

[27] L. Lazos, R. Poovendran, and S. Čapkun, "ROPE: robust position estimation in wireless sensor networks," in *Proceedings of IPSN*. Piscataway, NJ, USA: IEEE Press, 2005, p. 43.

[28] S. Brands and D. Chaum, "Distance-bounding protocols," in *Proceedings of EUROCRYPT*. Springer-Verlag, 1993, pp. 344–359.

[29] Y. Zeng, S. Zhang, S. Guo, and X. Li, "Secure hop-count based localization in wireless sensor networks," in *Proceedings of CIS*, 2007.

[30] D. Liu, P. Ning, and W. Du, "Detecting malicious beacon nodes for secure location discovery in wireless sensor networks," in *Proceedings of ICDCS*, 2005.

[31] A. Srinivasan, J. Wu, and J. Teitelbaum, "Distributed reputation-based secure localization in sensor networks," *Journal of Autonomic and Trusted Computing*, 2007.

[32] A. Srinivasan, J. Teitelbaum, and J. Wu, "Drbts: Distributed reputation-based beacon trust system," in *Proceedings of DASC*, Oct. 2006, pp. 277–283.

[33] D. Liu, P. Ning, A. Liu, C. Wang, and W. K. Du, "Attack-resistant location estimation in wireless sensor networks," *TISSEC*, vol. 11, no. 4, pp. 1–39, 2008.

[34] S. Misra, S. Bhardwaj, and G. Xue, "ROSETTA: Robust and secure mobile target tracking in a wireless ad hoc environment," in *Proceedings of MILCOM 2006*, 2006.

[35] S. Misra, G. Xue, and S. Bhardwaj, "Secure and robust localization in a wireless ad hoc environment," *IEEE Transactions on Vehicular Technology*, to appear.

[36] N. Kiyavash and F. Koushanfar, "Anti-collusion position estimation inwireless sensor networks," in *Proceedings of MASS*, 2007.

[37] S. Zhong, M. Jadliwala, S. Upadhyaya, and C. Qiao, "Towards a theory of robust localization against malicious beacon nodes," in *Proceedings of INFOCOM*, 2008.

[38] P. Rousseeuw and A. Leroy, *Robust regression and outlier detection*. Wiley-Interscience, 2003.

[39] Y. Zhang, W. Liu, Y. Fang, and D. Wu, "Secure localization and authentication in ultra-wideband sensor networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 4, pp. 829–835, April 2006.

[40] F. Anjum, S. Pandey, and P. Agrawal, "Secure localization in sensor networks using transmission range variation," in *Proceedings of MASS*, Nov. 2005, pp. 9 pp.–203.

[41] T. Kindberg, K. Zhang, and N. Shankar, "Context authentication using constrained channels," in *Proceedings of WMCSA*, 2002, p. 14.

[42] A. Vora and M. Nesterenko, "Secure location verification using radio broadcast," *IEEE Transactions on Dependable and Secure Computing*, vol. 3, no. 4, pp. 377–385, Oct.-Dec. 2006.

[43] W. Du, L. Fang, and P. Ning, "LAD: Localization anomaly detection forwireless sensor networks," in *Proceedings of IPDPS*, April 2005, pp. 41a–41a.

[44] T. Leinmuller, E. Schoch, and F. Kargl, "Position verification approaches for vehicular ad hoc networks," *IEEE Wireless Communications*, vol. 13, no. 5, pp. 16–21, 2006.

[45] E. Ekici, J. Mcnair, and D. Al-Abri, "A probabilistic approach to location verification in wireless sensor networks," in *Proceedings of ICC*, vol. 8, June 2006, pp. 3485–3490.

[46] E. Ekici, S. Vural, J. McNair, and D. Al-Abri, "Secure probabilistic location verification in randomly deployed wireless sensor networks," *Ad Hoc Networks*, vol. 6, no. 2, pp. 195 – 209, 2008.

[47] J. Hwang, T. He, and Y. Kim, "Detecting phantom nodes in wireless sensor networks," in *Proceedings of INFOCOM*, May 2007, pp. 2391–2395.

[48] Y. Wei, Z. Yu, and Y. Guan, "Location verification algorithms for wireless sensor networks," in *Proceedings of ICDCS*, June 2007, pp. 70–70.

[49] Y. Zeng, J. Cao, J. Hong, S. Zhang, and L. Xie, "SecMCL: A secure monte carlo localization algorithm for mobile sensor networks," in *accepted by WSNS*, 2009.

[50] "A longer version of this paper," in *http://zyingp.googlepages.com/SecLocSurveyLong.pdf*.