Pollution Attack: A New Attack Against Localization in Wireless Sensor Networks

Yingpei Zeng[†‡] Jiannong Cao[‡] Shigeng Zhang[†‡] Shanqing Guo[⋆] Li Xie[†]
[†]State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, P.R. China
[‡]Department of Computing, Hong Kong Polytechnic University, Hong Kong
[⋆]School of Computer Science and Technology, Shandong University, Jinan, P.R. China
{csyzeng,csjcao,cssgzhang}@comp.polyu.edu.hk guoshanqing@sdu.edu.cn xieli@nju.edu.cn

***Abstract*—Many secure localization algorithms have been proposed. In these algorithms, *collusion attack* is usually considered as the strongest attack when evaluating their performance. Also, for ensuring correct localization under the collusion attack, a necessary number of normal beacons are needed and a lower bound on this number has been established (assuming the errors of distance measurements are ignorable). In this paper, we introduce *pollution attack*, a more powerful attack which can succeed even when the number of normal beacons is more than the lower bound. In this attack, victim node is misled to a special chosen location, which results in a confusion of compromised beacon with normal beacon. We propose a new metric to measure the vulnerability of a normal location reference set to pollution attack, and develop two algorithms to efficiently compute the value of the proposed metric. We also present a method to judge whether the output of the localization algorithm is credible under pollution attack. Simulation results show that the pollution attack can succeed with high probability.**

I. Introduction

It is important for sensors to get their correct locations in hostile environments (e.g., battlefield). Because applications (e.g., target tracking) and routing protocols (e.g., GPSR [1]) of sensor networks may depend on the locations of nodes. Fig.1 shows an attack scenario on the battlefield. Here sensors determine their locations with the help of beacons, and they will detect tanks passing by and report these events combining their locations to the base station. The attacker now has compromised several beacons, and he wants to spoof the three sensors s_1, s_2 and s_3 to false locations s'_1, s'_2 and s'_3, then the three sensors will report that they detect the tank on the false path.

Currently many secure localization algorithms have been proposed to defeat attackers [2]–[8]; usually these algorithms are evaluated with the assumption that the general *collusion attack* (i.e., all compromised beacons trying to mislead the victim node to the same false place) is the strongest attack [3], [4], [6]–[8]. In this paper, we introduce *pollution attack*, a more powerful attack to localization. In this attack, the *location references* (which is composed of the location of a beacon and corresponding distance to that beacon) given by attackers are consistent with some location references given by normal beacons, then these normal beacons cannot be used to defeat the attackers anymore ("polluted"). This attack is essentially a special *collusion attack*. Simulation results show that pollution attack breaks the ideal lower bound proposed in [8] with high probability (assuming relatively small measurement error).

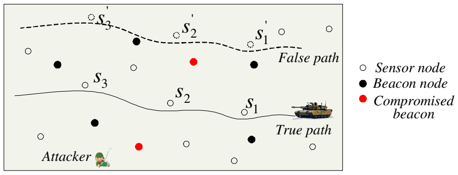


Fig. 1. An attack scenario on the battlefield.

Specifically, we make the following contributions in this paper:

1. We find a more powerful attack to localization algorithms. The attack can succeed even when the number of normal beacons is more than the lower bound in [8]. Our analysis focuses on a class of localization algorithms defined by us, but our simulations confirm that other kinds of secure localization algorithms are also vulnerable.

2. We show that the GDOP (geometric dilution of precision) metric cannot be used to measure the vulnerability of a location reference set to pollution attack, and we propose a new metric (ds_k), along with two algorithms for computing the value of the metric. We also present an examination method for the node to check the credibility of its location estimation.

The rest of the paper is organized as follows. Section II presents related work. Section III gives the network model and assumptions. Section IV presents the description of pollution attack as well as the algorithms for finding the metric ds_k and also the method for judging the credibility of an estimated location. Section V gives the simulation results of pollution attacks. Finally Section VI concludes this paper.

II. Related Work

Many algorithms have been proposed to address the attackers in the localization, and we roughly classify them into two approaches. The first approach is relatively aggressive; it tries to detect or prohibit malicious nodes with new hardware or protocols. Sastry et al. [9] proposed an echo protocol for verifiers to verify the location claims of nodes. Pires et al. [10] designed protocols to detect malicious message transmissions violating their positions. Lazos et al. [2] presented a new algorithm called SeRLoc using sector antennas. Liu et al. [11] proposed techniques to detect and revoke malicious beacons.

978-1-4244-2948-6/09/$25.00 ©2009 IEEE

Čapkun et al. [5] proposed protocols to bound the distance between nodes. Also in [12], Čapkun et al. proposed to use hidden and mobile stations to detect location spoofing. Zeng et al. proposed to use beacons to detect wormholes in [13].

Another approach is to filter or to tolerate the false information induced by attackers. This approach is more conservative since it usually only needs to run novel robust algorithms on the node; no new hardware or special nodes are needed. Our pollution attack is mainly against the algorithms in this approach, but the idea of pollution can definitely be used to attack algorithms in the previous approach. Li et al. [4] used Least Median of Squares (LMS) to tolerate the outliers in location references. Liu et al. [3] proposed another attack-resistant Minimum Mean Square Estimation(AR-MMSE) method. Misra et al. [6] generalized the secure localization problem to a second order cone problem (SOCP). Kiyavash et al. [7] proposed a fast algorithm to find consistent location references. Zhong et al. [8] also gave two algorithms to find the location of a node when there are less than $\frac{n-3}{2}$ malicious beacons (n is the total number of beacons). We note here that almost all these algorithms assume the general collusion attack is the strongest attack.

III. Network Model and Assumptions

We consider a simple model that only one sensor node M wants to obtain its position. The sensor node can hear n beacon nodes which know their locations. Both the sensor node and beacon nodes may be static or mobile. k beacon nodes are compromised (i.e., the number of normal beacon nodes is $g = n - k$). The information that M gets from beacons includes the locations of beacons and the distances from M to the beacons, which forms a location reference set: $\{<loc_i, d_i> | 1 \leq i \leq n\}$ (loc_i is the location of B_i beacon and d_i is the distance between M and B_i). Location references may be obtained from 1-hop beacons by receiving the location of beacon and measuring the distance between them (e.g., through measuring received signal strength indicator (RSSI) [14] or time difference of arrival (TDoA) [15]), also may be obtained from beacons multihop away (e.g., through DV-based methods in [16]).

We assume that the difference between the measured distance and the true distance (i.e., measurement error) is no more than ϵ, which is the same to the assumption in [3], [8] (In reality, the error may vary from 1%-5% of the real distance when using TDoA [17], and 10%-50% of the real distance when using RSSI [18]). We assume k beacons are compromised by one attacker for simplifying the discussion, since multiple colluding attackers have the same effect as a single attacker. Also we assume the attacker may change any fields of these k location references obtained by the sensor. This can be done by declaring false locations or distorting distance measurements [3]. In the following sections, we will first assume the attacker know the location references that the sensor get from normal beacons. Then we will show in Section V Fig.6(d) that in fact only the real locations of the sensor node and other normal beacons are needed.

IV. Pollution Attack to Localization

Although pollution attack can distort the result of different localization algorithms, different attack strategies are needed for efficient pollution. In particular, we first identify a class of algorithms for the description of our pollution attack strategy in this paper. Then we analyze how to judge the vulnerability of a location reference set when using this kind of algorithm, and how to examine the location estimation.

A. Class of Resilient Localization Algorithms

Before defining this algorithm class, we describe some terminology. The *cover ring* R_i of a location reference $< loc_i, d_i >$ (by B_i) is the area where each point X satisfies $d_i - \epsilon \leq dist(loc_i, X) \leq d_i + \epsilon$. We call a location reference (or the corresponding beacon) *covers* some location if the location is in the cover ring of the location reference.

Definition 1: We define the class of *resilient localization algorithms* as algorithms which always output a location that is maximally covered by location references.

This definition is reasonable since we usually trust the one has the most supporters. In fact, existing algorithms that belong to resilient localization algorithm include grid-voting based localization in [3] and algorithms in [8]. Specially, when there are multiple locations (i.e., an area) that have the same supporter, a random location among them can be selected as in [8], but generally the centroid of all these locations is selected [2], [3], [6]. In following sections, we will describe our pollution attack to this class of algorithms, however we will show in Section V that the pollution attack is also able to subvert the results of other types of algorithms.

B. Pollution Attack Description

We here describe the *pollution attack* to the algorithm class defined in the previous subsection. Recall that the general *collusion attack* is that all the compromised beacons try to mislead the node to a location which is different from the node's real location [3], [4]. Now the *pollution attack* is different in that all the compromised beacons try to mislead the node to a *special* location. The location should be *not only different from the node's real location but also has enough normal beacons (denoted by n_c)* covering *it*. The value of n_c relies on the number of compromised beacons k: if the pollution attack wants to *definitely* (recall that in the previous subsection some algorithms may randomly select an candidate location) succeed, then $n_c \geq n - 2k + 1$ should be satisfied, for the number of beacons covering the real location (i.e., $n - k$) should be less than the number of beacons covering the false location (i.e., $n_c + k$).

Take Fig.2 for example. The real location of the node is M. If beacon B_5 doesn't exist yet, there are only 4 beacons. The compromised beacon B_4 can launch a *pollution attack* by covering a false location M' in the shadow area. The attack will *definitely* success since M' has more beacons covering it (4 vs. 3).

Pollution attack can succeed even the lower bound for the number of normal beacons in [8] is satisfied. The lower

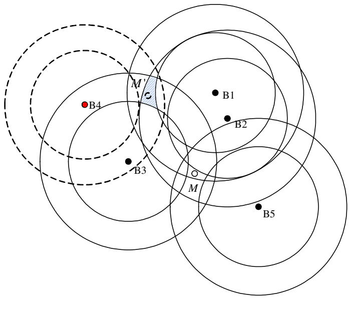


Fig. 2. Pollution attack example. Here B_1, B_2 B_3 and B_5 are normal beacons, while B_4 is compromised beacon.

bound can be simply stated as: the number of normal beacons should be more than $k+2$ (k is the number of compromised beacons)[1]. However in Fig.2, when another normal beacon B_5 exists, the lower bound is satisfied. But both M and M' have four beacons covering them, and if the resilient localization algorithm selects a random location as in [8], the attacker still may succeed. The lower bound failed because it is derived with the ideal assumption that measurement error is *ignorably* small; however, we will show by simulations in Section V that pollution sometimes may succeed with high probability even when measurement error is practically small value.

We next give a theorem on the maximum location error when the pollution attack succeeds:

Theorem 1: The maximum location error of the *resilient localization algorithm*: (i)when $k \geq g$, it's infinite, (ii) when $k < g$, it's $2(d_{max} + \epsilon)$, where $d_{max} = max\{d_1, d_2, ..., d_i, ..., d_g\}$.

Proof: (*sketch*) In the first case, the attacker can certainly mislead the node to any location, so the location error may be infinite. In the second case, the attacker must choose a false location covered by at least one normal location reference, then the false location must be within $2(d_{max} + \epsilon)$ distance to the true location of the node. ■

C. Why Not Use GDOP and the New Metric

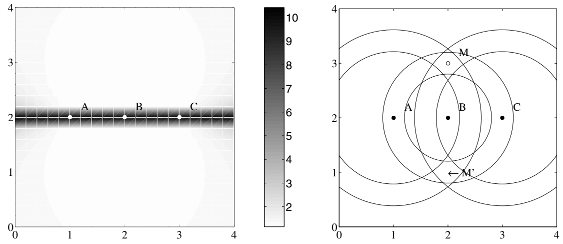
The theorem 1 in the previous section only gives the maximum error bounds in the general case, but the success of pollution attack depends on the layout of beacons and node (also the measurement error). So in this section, we want to find a metric to measure the vulnerability of a *given* location reference set to pollution attack. A related metric for measuring the (geometric) quality of received location information is GDOP in the GPS domain. It can be approximately computed by [19]:

$$GDOP = \sqrt{tr(\mathbf{H}^{[1]T}\mathbf{H}^{[1]})}, \mathbf{H}^{[1]} = \left.\frac{\partial \boldsymbol{\rho}}{\partial \mathbf{x}}\right|_{X_{\text{nom}}, Y_{\text{nom}}, Z_{\text{nom}}} \quad (1)$$

where (X_{nom},Y_{nom},Z_{nom}) is a nominal solution for the location (X,Y,Z), and $\boldsymbol{\rho}$ is a vector and is composed of:

$$\rho_i = \sqrt{(x_i - X)^2 + (y_i - Y)^2 + (z_i - Z)^2} \quad (2)$$

with the (x_i, y_i, z_i) is the location of the i satellite.



(a) A demo of GDOP in 2D.

(b) Pollution attack.

Fig. 3. Illustration of GDOP and pollution opportunity.

However we find that GDOP fails to meet our demand. We illustrate this by an example in 2D. Three satellites A, B and C are placed linearly, and their coordinates are $(1, 2)$, $(2, 2)$ and $(3, 2)$ respectively. The GDOP values of each location are shown in Fig.3(a). Then in Fig.3(b) we plot the three beacon nodes with the same coordinates, and the sensor M placed at $(2, 3)$ will get three location references so we plot corresponding cover rings. Now we can see that attacker controlling one beacon can cover a location in M' and then succeed in pollution, so the location M indicates high pollution threat. But the GDOP value of M in Fig.3(a) is very small. Also the induced location error by pollution attack will be bigger when M is more distant from the beacons; however this contradicts the GDOP distribution. Finally, GDOP doesn't use the measurement error ϵ, which in fact can impact the pollution opportunity (the area of M').

We hence define a new metric to characterize the vulnerability of a location reference set to pollution attacks:

Definition 2: We define ds_k as the maximum location error the attacker controlling k beacons can *definitely* achieve: $max\{dist(P', M)\} = ds_k$, where P' is any location successfully misled by the attacker and M is the real location. The location corresponding to ds_k is the *optimal pollution location*.

The definition is given from the aspect of the location distortion strength by the pollution attack. We note that some additional directional constraints such as "misleading the location of the node to the north" can be added to the metric. Also, a minimum distortion threshold ds_L usually is useful (similar to minimum strength of attack in [4]), because both the sensor and attacker may not care about pollution unless $ds_k > ds_L$. Next we present a theorem on the optimal pollution location.

Theorem 2: The *optimal pollution location* is on the edge of a *cover ring*, and has at least $g - k + 1$ location references *covering* it.

Proof: (*sketch*) Firstly, any non-boundary points within cover rings cannot be the *optimal pollution location*, otherwise we can certainly find another qualified point in its δ neighborhood (δ is an arbitrary small value) which has bigger location error. So the optimal pollution location must be on the edge of a cover ring. Secondly, as mentioned in Section IV-B, the number of needed normal beacons for covering the false location is at least $n_c = n - 2k + 1 = g - k + 1$. ■

[1] The authors are aware of another lower bound in [6] which is essentially proved in the similar way

D. Algorithms for Finding the Value of ds_k

We have defined a metric ds_k for the vulnerability of a given normal location reference, then how to compute the value of ds_k efficiently is a problem now. We present two algorithms here. The first algorithm is grid search algorithm, which is used as the baseline algorithm. It contains the following steps: (1) find a minimum rectangle covering all the g location references, (2) then divide the rectangle into small grids (cells), each grid maintains a counter initialized to 0, (3) all the cover rings of the g location references add the counter of a grid by 1 if they cover or intersect with the grid, (4) select cells satisfying: its counter is no less than $g - k + 1$ and it is intersecting or out of a circle centering at the real location of node (M) with radius ds_L, and then sort these cells in descending order by their distances to M, (5) check each cell sequentially that: if the cell size is more than a value (i.e., the threshold of accuracy), then recursively call step 2 - step 5 to divide the cell further; otherwise randomly sample some points, if find a point has more than $g - k + 1$ location references covering it, then output its distance to M as ds_k and stop the algorithm. The grid search method is also used in [3]; however the method here may have more iterations in step 5 because now it's more often that candidate cells don't contain a valid point eventually.

We propose another heuristic algorithm, which is to use points of intersection of cover rings as the pollution location candidates. Because we know that functions reach their extreme values in the stationary point, or point where the derivative is not defined. Also, if we assume $k \leq g - 1$, then $n_c \geq 2$, which means the pollution location must lie in an intersection area of cover rings. So points of intersection are good candidates for optimal pollution location. The details of our heuristic algorithm are shown in Algorithm 1. The worst-case time complexity of this algorithm is $O(g^3 - g^2)$, and it's very fast in practice. Also, the algorithm can be speeded up when the memory is not a critical resource: we can sort the points of intersection by their distances to M before line 3 of the algorithm, and then we can break the loop immediately after line 7.

Algorithm 1 Find the value of ds_k by points of intersection.

```
Input: location of node $M$, $g$ normal location references, compromised beacon number $k$, measurement error $\epsilon$ and the minimum distortion threshold $ds_L$.
 1: $ds_k = -1$.
 2: Compute points of intersection of all the cover rings $R_i$.
 3: for each point of intersection $I$ do
 4:   if $dist(I, M) > ds_L$ then
 5:     if the number of location reference covering $I$ is no less than $g - k + 1$ then
 6:       if $dist(I, M) > ds_k$ then
 7:         $ds_k = dist(I, M)$.
 8:       end if
 9:     end if
10:   end if
11: end for
```

Proposition 1: The error of the ds_k value computed by Algorithm 1 is no more than $(2 - \sqrt{2})(d + \epsilon)$ when all the g normal location references have equal $d_i = d$, and is no more than $2d_{max}$ when d_is are not equal.

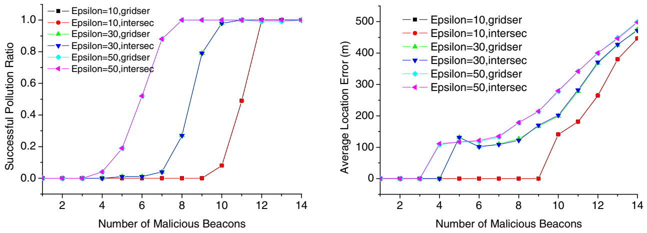


Fig. 4. Pollution results for uniform measurement error.

The proof is given in Appendix A. In our simulation the ds_k value computed by Algorithm 1 is very approximate to the real value.

E. Defending Against Pollution Attacks

In previous subsections we try to measure the vulnerability of g normal location references to the pollution attack. Here we consider the condition that the attack may has happened and the node wants to defend itself. Apparently the *resilient localization algorithms* cannot be used. Other secure localization algorithms may tolerate partial pollution attacks (we will show this in Section V), because the current pollution strategy (i.e., selecting *optimal pollution location* as the false location) doesn't consider other statistical properties such as the square errors. But if not follow the current strategy, pollution attack sometimes can be essentially hard to defend against, e.g., in Fig.2 the false locations M' has nearly the same properties with M. In fact, no matter what pollution strategy the attacker select, if the node is using a *resilient localization algorithm*, there is an examination method for the node to check the credibility of output location. The node can restrict itself to update its location only when the output location is definitely correct.

The problem is formally stated as follows: a node receives n location references, the output location of its resilient localization algorithm is P, and it knows that there are at most k compromised beacons, then, is the output location correct (or has less than ds_L deviation) under a pollution attack? The examination method is essentially according to the required n_c number in Section IV-B: (1) find the number of location references covering P, denoted by a, (2) execute Algorithm 2 (which is a similar algorithm to Algorithm 1) to find the minimum required number of compromised beacons (denoted by m) and corresponding pollution location (denoted by P'), (3) find the number of location references covering P and P' at the same time (denoted by c), (4) if $c + k < a$ holds, then the location P is definitely correct, else the location P may be polluted and incorrect.

V. Simulation Results

This section presents the simulation results of the pollution attack. In all simulations, 15 normal beacon nodes and k $(1 \leq k \leq 14)$ compromised beacon nodes are randomly deployed in a circular area with radius=250m. The non-beacon node is located at the center of the circle (i.e., M). We assume the transmission range is 250m then the non-beacon

```
Algorithm 2 Find the minimum $m$ and corresponding $P'$.
Input: output location $P$ of the resilient localization algorithm,
    the number of location references covering $P$ as $a$, $n$ location
    references, measurement error $\epsilon$ and the minimum distortion
    threshold $ds_L$.
$m = BIG\_INT$, $P' \leftarrow null$. // Initialize
Compute points of intersection of all the cover rings $R_i$.
for each point of intersection $I$ do
  if $dist(I, P) > ds_L$ then
    let $x$ be the number of location reference covering $I$.
    if $a - x + 1 < m$ then // $I$ needs less compromised beacons
      $m = a - x + 1$, $P' \leftarrow I$.
    else if $a - x + 1 = m$ then
      if $dist(I, P) > dist(P', P)$ then
        $P' \leftarrow I$.
      end if
    end if
  end if
end for
```

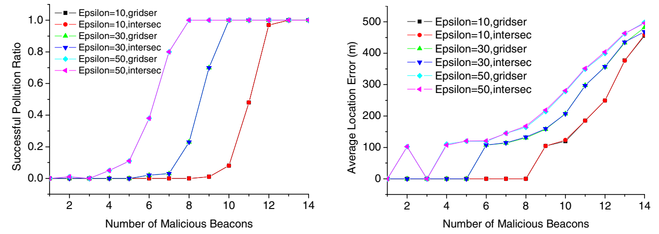
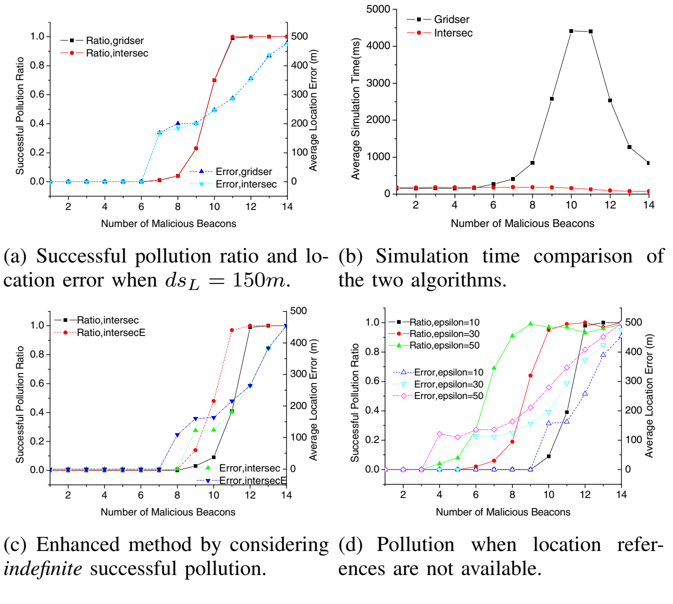


Fig. 5. Pollution results for normal measurement error.

node can always receive beacon signals from all the beacons. The minimum distortion threshold ds_L is set to 100m. The pollution attack process contains 3 steps: first we find ds_k and corresponding *optimal pollution location* P' by algorithms in Section IV-D, then we use a simple method to generate k malicious location references for the false location P': generate the locations of compromised beacons along the line $P'M$ (for avoiding pollution to other locations unconsciously), at last, we feed the $15 + k$ location references to the grid-voting based algorithm [3] (a *resilient localization algorithms*) to judge whether the pollution is successful. The performance metrics in our simulation are: (1) the *successful pollution ratio* (SPR), which is number of successful pollution rounds divided by number of total rounds, (2) the *average location error* (ALE), which is the average location error *when the pollution is successful*. For convenience, we call the grid search algorithm as *gridser* algorithm, and Algorithm 1 as *intersec* algorithm.



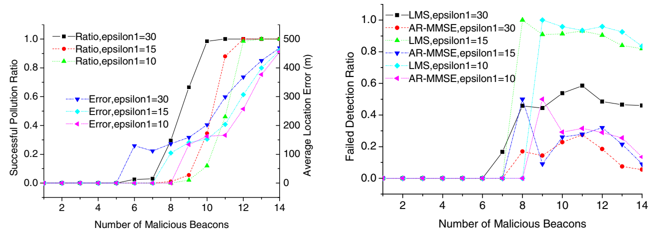
(a) Successful pollution ratio and location error when $ds_L = 150m$.

(b) Simulation time comparison of the two algorithms.

(c) Enhanced method by considering *indefinite* successful pollution.

(d) Pollution when location references are not available.

Fig. 6. four simulations still about *resilient localization algorithms*.



(a) Successful pollution ratio and location error with shrank ϵ.

(b) The ratio of pollution attacks cannot detected by LMS and AR-MMSE.

Fig. 7. Pollution attack to *non-resilient localization algorithms*.

First, we simulate the pollution attack with two different types of measurement errors: uniform distribution over $[-\epsilon, \epsilon]$ and normal distribution with mean 0 and standard deviation $\frac{\epsilon}{2}$ [2]. In both case, we run the two algorithms (*gridser* and *intersec*) with ϵ from 10m to 50m in steps of 20m, and k from 1 to 14 in steps of 1 (all with 100 runs). Fig.4 and Fig.5 show the results of them respectively. We can see that both the SPR and the ALE of *intersec* algorithm are quite similar to the results of *gridser* algorithm (lines are overlapped). The SPR can be as high as 100%, e.g., when measurement error is uniformly distributed, ϵ is a relatively small value 10m, $k = 12$. Corresponding ALE is up to 260m. Also, it isn't surprising to see that the bigger ϵ is, the more powerful pollution attack is. Finally the results show that the lower bound in [8] is not applicable, because according to the lower bound, there should be no successful attack when $k \leq 12$.

Then we give additional five simulations to study the pollution attack further. *In the first simulation*, ds_L is set to a bigger value 150m, with normally distributed measurement error and ϵ=30m. In Fig.6(a) we can see that both the SPR and ALE degrade a little comparing with Fig.5(a). *In the second simulation*, the execution time of *gridser* algorithm and *intersec* algorithm is compared. From Fig.6(b) we can see that the simulation time of *intersec* algorithm is very steady, however, the simulation time of *gridser* algorithm become much more than the *intersec* algorithm when there are 8-12 compromised beacons. The reason is that in step 5 (in Section IV-D) the algorithm spends much time on dividing and searching grids which eventual have no qualified point. Since *intersec* algorithm has the similar pollution ability but is more efficient than *gridser* algorithm, we will only use it in following simulations.

In the third simulation, we relax the lower bound of n_c by

[2] Similar to [8], we modify the distribution to make sure the probability density outside $[-\epsilon, \epsilon]$ becomes 0 (i.e., we discard the samples outside $[-\epsilon, \epsilon]$), then the values of measurement errors are always within $[-\epsilon, \epsilon]$.

1 to $n - 2k$. Fig.6(c) shows that as expected both the SPR and ALE tagged with *intersecE* are slightly higher. *In the fourth simulation*, we assume that the attacker doesn't know the normal location references, and it guesses that the distances in the location references are the same as the actual distances from the sensor to normal beacons. Results are shown in Fig.6(d) (uniform distribution). We can see that both SPR and ALE have only very little degradation comparing with results in Fig.4. *In the last simulation*, we study pollution attack to other kinds of secure localization algorithms, e.g., LMS [4] and AR-MMSE [3] (normal distribution and ϵ=30m). Since these algorithms are related to square errors so we slightly modify our pollution strategy: we shrink the ϵ in the *intersec* algorithm to be not equal to 30m (we use ϵ_1 to represent it). The results are shown in Fig.7. We can see that both LMS [4] and AR-MMSE [3] are vulnerable to the pollution attack, and a smaller ϵ_1 makes the pollution attack harder to detect.

VI. Conclusion

In this paper we have described the new attack called pollution attack. We mainly focus on pollution to the *resilient localization algorithms* defined by us. As the ability of attack is related to concrete location reference set, so we propose a metric ds_k to measure the vulnerability of a given normal location reference set. Then we give two algorithms for finding the value of ds_k. We also present a method to check the credibility of the output location. Finally our simulation shows that the pollution attack is very powerful and breaks an existing lower bound with high probability. Base on the results we argue that researchers need to pay attention to the pollution attack when designing and evaluating secure localization algorithms. In the future we plan to theoretically analyze the success probability of the attack.

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, China 973 Project under Grant No. 2006CB303000, and the Natural Science Foundation of Jiangsu Province under Grant No.BK2007136.

References

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

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

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

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

[5] S. Čapkun and J. P. Hubaux, "Secure positioning in wireless networks," *IEEE JSAC*, 2006.

[6] 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.

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

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

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

[10] W. R. P. Júnior, T. H. de Paula Figueiredo, and H. C. Wong, "Malicious node detection in wireless sensor networks," in *Proceedings of IPDPS*, 2004.

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

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

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

[14] 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.

[15] 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.

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

[17] "Mica-2 experimental data," in *http://osl.cs.uiuc.edu/nest/data/?M=D.*

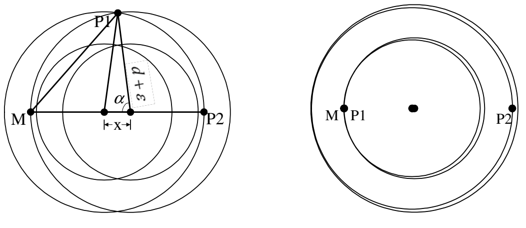
[18] K. Whitehouse, C. Karlof, A. Woo, F. Jiang, and D. Culler, "The effects of ranging noise on multihop localization: an empirical study," in *Proceedings of IPSN*, 2005.

[19] M. S. Grewal, L. R. Weill, and A. P. Andrews, *Global Positioning Systems, Inertial Navigation, and Integration*. John Wiley & Sons, Inc, 2001.

Appendix A

Proof of Proposition 1:

Since we assume that $n_c \geq 2$, so the *optimal pollution location* must lie in an intersection area (denoted by S) of at least 2 cover rings. Then it becomes a problem of finding the maximum distance between any points of intersection at the boundary of S and any points in S. Because more intersecting cover rings will make S smaller, we here consider the case of 2 intersecting cover rings to give an upper bound. We use E to denote the error of ds_k.



(a) Case of equal d_is.

(b) Case of unequal d_is.

Fig. 8. Demonstrations.

(1) When all the d_is are equal, Fig.8(a) shows the case that the intersection area of two cover rings is the biggest. We have $E = MP_2 - MP_1$. We need to find the maximum of E when $x \in [0, 2\epsilon]$. Following the law of cosines, we have $MP_1 = \sqrt{2(d+\epsilon)^2 - 2(d+\epsilon)cos(\alpha)} = \sqrt{2(d+\epsilon)^2 - (d+\epsilon)x}$. So $E = (2(d+\epsilon) - x) - MP_1 = 2d + 2\epsilon - x + \sqrt{2(d+\epsilon)^2 - (d+\epsilon)x}$. Then $\frac{\partial E}{\partial x} = -1 - \frac{1}{2\sqrt{2 - x/(d+\epsilon)}}$, which is away less than 0 when $x \in [0, 2\epsilon]$. So E reaches its maximum when $x = 0$, $E = (2 - \sqrt{2})(d+\epsilon)$.

(2) When d_is is not equal, E reaches its maximum when a cover ring (d_i=d_{max}) intersects with another a little smaller ring, as shown in Fig.8(b). Then $E = MP_2 - MP_1 = 2(d_{max} + \epsilon) - 2\epsilon = 2d_{max}$.

Thus the proposition is proved.