Improving resilience and revocation by mitigating bad mouthing attacks in Wireless Sensor Networks

K.Vijaya, Dr.M.Selvam

Abstract— Sensor networks can be deployed in hostile environments where adversaries may be present. Since wireless sensor networks usually need to be controlled remotely by the network operator, they are often deployed in an unattended manner. The unattended nature of wireless sensor networks can be exploited by attackers. Specifically, an attacker can capture and compromise sensor nodes and launch a variety of attacks by leveraging compromised nodes. There are so many existing schemes to identify the malicious node and they could not provide the way to revoke them. The proposed scheme solves how to revoke the compromised nodes by using reputation based Credence management scheme, which incorporates a new scheme called SPRT. By using this Credence based management scheme a secure network can be established for secure communication among wireless nodes.

Index Terms— Biased SPRT, Compromised Node,Reputation based Credence management scheme , Zone

1 INTRODUCTION

A more aggressive attacker could undermine common sensor network protocols, including cluster formation, routing, and data aggregation, thereby causing continual disruption to the network operations. Therefore, an adversary with compromised nodes can paralyze the deployed mission of sensor networks. It has been demonstrated that rating Credence and reputation of individual nodes is an effective approach in distributed environments in order to improve security, support decision-making and promote node collaboration. Nevertheless, these systems are vulnerable to deliberate false or unfair testimonies. In one scenario the attackers collude to give negative feedback on the victim in order to lower or destroy its reputation. This attack is known as bad mouthing attack, and it can significantly deteriorate the performances of the network. The existing solutions for coping with bad mouthing are mainly concentrated on prevention techniques In this sense, it is very important to detect [1], [6], [23] and revoke compromised nodes. To mitigate the node compromise attack, several researchers have proposed various node compromise detection schemes in the context of wireless networks.

Specifically, the network is first divided into a set of regions, establish Credence levels for each region, and detect untrustworthy regions by using the Sequential Probability Ratio Test (SPRT). The SPRT [22] decides a region to be untrustworthy if the region's Credence is continuously maintained at low level or is quite often changed from high level to low level.

- *K.Vijaya is currently pursuing Ph.D in Anna University of Technology,Coimbatore, Tamil nadu, India. E-mail:vijaya.krishnamoorthy@gmail.com*
- *Dr.M.Selvam is currentlyworking as a Principal in Vel Tech High Tech Dr.Rangarajan Dr.Sakunthala Engineering College. Chennai, Tamil nadu, India.*

Once a region is determined to be untrustworthy, the base station or the network operator performs software attestation [1], [23] against all nodes in the ununtrustworthy region, detects compromised nodes with subverted software modules, and physically revokes them.

2 MODEL AND SYSTEM REQIREMENTS

2.1 Network Assumption

We assume a static sensor network in which the sensor nodes do not change their locations after deployment. We also assume that all direct communication links between the sensor nodes are bidirectional. We also assume that the base station is a trusted entity. This is a standard assumption; if the base station is compromised, the entire mission of the sensor network can be easily undermined [2], [4]. We assume that every sensor node is able to obtain its location information and identify its placement region by using an existing secure localization scheme such as. Finally, we assume time synchronization [9], [18] such that the clocks of all nodes in a region are loosely synchronized. This can be achieved by using one of the existing secure time synchronization methods.

2.2 Attacker Model

We assume that the attacker attempts to maximize the impact of node compromise attacks by compromising a subset of the nodes in each target region. In particular, we assume that the attacker could gain more by compromising at least a few nodes in one region, rather than compromising nodes that are isolated from each other. This is because a subset of compromised nodes in a region can launch much more effective attacks through collaboration than an isolated and compromised node can. For instance, false data injection attacks can make the reported observations in a region appear quite different from reality.

2.3 System Requirement

Given the attacker model, we now focus on the requirements

for an effective defense against node compromise attacks in a wireless sensor network. A defense mechanism should have several key attributes: 1) fast and accurate detection capability; 2) efficiency; and 3) low cost for false positives. Fast detection of malicious nodes, without missing the attackers due to false negatives, is obviously a critical attribute for a defense against node compromise. In our scheme, we aim to save the costs of frequent attestation [1], [23], by using a low-cost reputation scheme to decide when to apply attestation.

3 Detection and Revocation of Bad Mouthing Attacks in Static Wireless Sensor Network

This section presents the details of the proposed scheme to detect node compromises and misbehavior on the basis of regions.

In SPRT [22] scheme, we divide the network into a set of regions, establish Credence values for each region, and detect untrustworthy regions in accordance with the region Credence values. Once a region is determined to be untrustworthy, the network operator attests the software modules of all sensors in the untrustworthy region, and detects and revokes compromised nodes in that region. Since benign nodes are attested only in untrustworthy regions, our scheme reduces the overhead incurred from attesting all benign nodes, as in the existing schemes based on software attestation. However, unlike other reputation-based schemes, our scheme benefits from the lack of false positives and ability to fully revoke compromised nodes that software attestation provides. The proposed protocol proceeds in three phases.

3.1 Discovering Region and Credence Aggregator Selection

After deployment, every sensor node u finds out its location and determines the region to which it belongs. We call this region the home region. From u's point of view, we call other regions the foreign regions. Node u discovers the IDs of all other nodes in its home region and establishes pair wise secret keys with them. After the region discovery process, the Credence Aggregator is selected in a round robin manner as follows: The time domain of a region is divided into a series of time slots. Each node pseudorandomly decides its duty time slots, during which it acts as a Credence aggregator, before each round starts, where a round consists of S time slots such that S is the number of nodes residing in the region. The reason why we adopt the pseudorandom settings of duty time slots is because the pseudorandom order of the duty time slots is beneficial for the secrecy of our scheme.

We now describe the selection of time duty slots in more detail. All nodes in the network share a pseudorandom number generator (PRNG) in which the starting time of each round is used as the seed value. According to the clock synchronization assumption in the starting time of each round (i.e., seed value) will be the same to all nodes in region. Each node thus will generate the same sequence of random values uniformly distributed between 0 and 1. Each node sets a duty time slot to a random number that corresponds to its order when the nodes in the region are sorted in ascending order. It repeats this duty time slot selection mechanism for each round before a round starts. Through this pseudorandom duty time slot mechanism, it is guaranteed that duty time slots are pseudo randomly determined per round and only one node is assigned to CA per time slot.

3.2 Credence Formation and Forwarding

For each time slot T_i, each node u in region Z computes neighborhood-Credence that is defined in accordance with the difference between the probability distributions of the information generated by u and the information sent to u by u's neighboring nodes in region Z. Neighborhood-Credence acts as an indicator of how much information is shared by two neighboring nodes. The more information u shares with its neighbors, the more belief u has in its neighbors. To compute the difference between two probability distributions, we use the information-theoretic metric Kullback-Leibler (KL)-divergence that is known to be suited for this computation [3] . Specifically, let us assume that u's generated information follows a probability distribution with mean and standard deviation and falls within the range $[\mu-c\sigma, \mu+c\sigma]$ with probability p, where c is a constant value. Moreover, we assume that the information sent and processed to u by u's neighbors falls within the range $[\mu-c\sigma, \mu+c\sigma]$ with probability q. Node u computes KL-divergence D as follows:

$$D=p*ln(p/q)+(1-p)*ln((1-p)/(1-q))$$

Such that $p>1/2$ and $p>=q$.

It then computes neighborhood-Credence n_u= min(1,1/(1+D)) ;This definition of neighborhood-Credence is reasonable in the sense that neighborhood-Credence is inversely proportional to D and thus it will be one if two probability distributions exactly match with each other. The main rationale behind the restriction of $p>1/2$ is to make the majority of the information generated by u fall within the expected range, leading to better accuracy in neighborhood-Credence measurement than the case of $p<=1/2$. Also, n_u is a neighborhood-Credence calculated from the perspective of u and accordingly it has its maximum value of one when p=q. Thus, it is reasonable to set $p>=q$. Since $p>=q$ should hold, in case that $q > p$, node u calculates D after resetting q =p.

D-Credence report from u, node v verifies the authenticity of u's neighbor-hood-Credence report with K_{uv} and discards the report if it is not authentic. v collects the neighborhood-Credence reports that were measured during T_i from all nodes in region Z and aggregates the received neighborhood-Credences by using the mean function. We call the aggregated version of neighbor-hood-Credences the region-Credence. v sends the base station Z's region- Credence report, defined as $\{v||s_v||Z||t||MAC_{Kv}(v||s_v||Z||t)$, where s_v is a time stamp indicating the generation time of report, t is Z's region-Credence, and K_v is the shared secret key between v and the base station.

3.3 Detection and Revocation

Upon receiving a region-Credence report from a CA in region Z, the base station verifies the authenticity of the CA's report with the secret shared key between CA and itself and the freshness of the time stamp and the base station discards the report if it is not authentic or contains a stale time stamp. The base station also maintains a record per CA associating each CA's ID with its home region and time stamp. This prevents the compromised CAs from claiming multiple home regions and from launching replay attacks with benign region-Credence reports. We denote
the authentic reports from the CAs in region Z by R_1; R_2; . . . . The base station extracts the region Credence information t_i from report R_i. Let τ be a Credence threshold and B_i denote a Bernoulli random variable defined as

$$B_i = \{1 \text{ if } t_i < \tau \text{ , } 0 \text{ if } t_i >= \tau\}$$

If p is smaller than or equal to a Credence threshold ϱ', it is likely that the region Z is trustworthy. On the contrary, if $\varrho > \varrho'$, it is likely that the region Z is untrustworthy. The problem of deciding whether Z is trustworthy or not can be formulated as a hypothesis testing problem with null and alternate hypotheses of $\varrho <= \varrho_0$ and $\varrho >= \varrho_1$, respectively, such that $\varrho_0 < \varrho_1$. In this problem, the acceptance of he alternate hypothesis is considered to be a false positive error when $\varrho <= \varrho_0$, and the acceptance of the null hypothesis is considered to be a false negative error when $\varrho >= \varrho_1$. We define user-configured false positive rate ^{0} and false negative rate β^1 in order to provide upper bounds on the false positives and false negatives in the hypothesis testing problem. These upper bounds will be presented in the security analysis in the next section.

We now describe how the SPRT [22] is used to make a decision about region Z from the n observed samples, where Credence information t_i is treated as a sample. Let us define H_0 as the null hypothesis that region Z is trustworthy and H_1 as the alternate hypothesis that region Z is untrustworthy. We then define L_n as the log-probability ratio on n samples, given as

$$L_n = ln(Pr(B_1,\ldots,B_n | H_1))/ Pr(B_1,\ldots,B_n | H_0))$$

If a region Z is judged as trustworthy, the base station restarts the SPRT with newly arrived region-Credence reports. If, however, Z is determined to be untrustworthy, the base station terminates the SPRT on Z, and the network operator detects and revokes the compromised nodes by having nodes in other regions perform software attestation against sensor nodes in region Z.

However, we note that the attacker can aim to block honest reports from reaching the base station. To address this, we have the base station track the reporting from each CA [21] in the region; this is also necessary for ensuring equal reporting frequency. If the multiple nodes in a region are having their reports blocked, the base station will treat the region as untrustworthy and initiate software attestation on the region. If, during software attestation, the non reporting CAs are found to be malfunctioning or out of power, the base station can remove them from the list of CAs to prevent future false positives.

4. PROPOSED WORK

4.1 limitations of node compromise attacks

We first consider the false region-Credence report attack in which the compromised CAs report false region-Credence values to the base station. This attack can take two forms. First, the compromised CAs could send reports of low region-Credence values to the base station when the region-Credence is actually high. Since this attack leads to quicker detection of the compromised CAs, the attacker will not benefit from this approach. Second, the compromised CAs could send reports of high region-Credence values to the base station when the region-Credence is actually low. In this type of attack, the attacker will have all compromised CAs report a region-Credence of 1.0 to the base station in order to prevent the untrustworthy region from being detected. We investigate the impact of this attack on the detection capability of our scheme. For this investigation, we look into the impact of the fraction of compromised nodes in a region on the detection capability of our scheme through the following Lemma. Recall that B_i is a Bernoulli random variable indicating whether the region-Credence report is below ($B_i = 1$) or (equal to or above) ($B_i = 0$) the Credence threshold.

4.2. Computation and Storage Overhead

We define the computation and storage overhead as the average number of Message Authentication Codes (MACs) that are generated and verified by a node and the average number of region-Credence reports that need to be stored by a node, respectively. Assume that there are b nodes on an average within a region. In a region, every node acts as the CA in its designated time slot while acting as a region member in the other time slots. For each time slot, each region member generates a MAC of its neighborhood-Credence report, which is sent to a CA. Each CA in turn performs b -1 MAC verifications on the received neighborhood-Credence reports and generates a MAC of its region-Credence report. Thus, b nodes perform 2b - 1 MAC generations and verifications for every time slot. Accordingly, the computation overhead per node will be O(1) per time slot on an average. The base station will perform z MAC verifications for every time slot because z CAs reports their region-Credences for each time slot.

5 BIASED SPRT

we propose a SPRT-based node compromise detection and revocation scheme and analyze its security and performance. Although this scheme achieves fast and accurate node compromise detection and revocation, it will not work if more than 50 percent of the nodes in each region are compromised under reasonable configurations of the SPRT. To enhance the resilience of the SPRT-based scheme against the false region-Credence report attack with a large number of compromised nodes, we modify the sampling strategy in the SPRT in such a way that the SPRT takes the samples leading to acceptance of H0 (high-Credence samples) with less weight than the ones

leading to acceptance of H1 (low-Credence samples), while ensuring that the false positive rate remains below the desired rate.

We call this modification biased sampling and the corresponding scheme as the Biased-SPRT. Since a high-Credence sample is less likely to be accepted than a low-Credence sample, H1 will be more likely to be accepted when the region is untrustworthy. Biased sampling results in greater delay in accepting the null hypothesis and greater false positive rates, but these are not major costs in the system as designed. Note that benign regions are continually tested for Credence values, so there is no benefit to quickly detecting that the region is trustworthy. Also, a false positive only costs the additional overhead of a single software attestation against the nodes in the region. The benefit of biased sampling is that even a relatively small number of honest nodes can send region-Credence reports with low-Credence values, leading to detection. We will show that biased sampling improves the resilience of the proposed scheme against the false region-Credence report attack, even when the fraction of compromised nodes is more than 50 percent.

In the biased sampling, the samples with type of H0 are taken into the sequential test process with less weight than H1 in such a way as to replace ϱ_o, with $(\varrho_o)^{\epsilon}$ where $\epsilon > 1$ is a biased sampling factor. Thus, the log-probability on n samples Ln is changed to

$$Ln = \omega_n \ln(\rho_1/(\rho_o)^{\epsilon}) + (n - \omega_n) \ln((1 - \rho_1)/(1-(\rho_o)^{\epsilon}))$$

Accordingly, the SPRT for H0 against H1 is changed to:

- $\omega_n \le \lambda_0(n)$: *accept H0 and terminate the test.*
- $\omega_n \ge \lambda_1(n)$: *accept H1 and terminate the test.*
- $\lambda_0(n) < \omega_n < \lambda_1(n)$: *continue the test process with another observation.*

Where:

An increase in ϵ leads to faster acceptance of H1 but slower acceptance of H0. Intuitively, we can imagine the SPRT's one-dimensional random walk taking larger steps toward H1 and smaller steps toward H0. This modification has several consequences. First, untrustworthy regions will be more likely and quickly to be detected. Second, even a region with a majority of compromised nodes that send false region-Credence reports can be detected as untrustworthy. Despite a number of false high-Credence reports that cause the SPRT to take small steps toward H0, the regular presence of true low-Credence reports will take bigger steps toward H1.

6 SECURITY ANALYSIS

In this section, we first investigate how ϵ affects the false positive rate of the Biased-SPRT and then show how much the system's resilience against the false region-Credence report attack can be enhanced by the Biased-SPRT. Finally, we will demonstrate that the Biased-SPRT causes the attacker to have substantially lower gains than the ones in the SPRT in terms of our game-theoretic analysis.

6.1 False Positive Rate

To examine how the biased sampling factor ϵ affects the false positive rate α, we use the estimated value of α instead of the upper bound derived in Section 3.2, because the estimated value will contribute to more an accurate investigation than the upper bound. According to Wald, α in the SPRT is approximately estimated as follows:

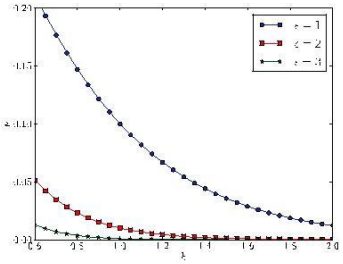


Fig:α versus h.

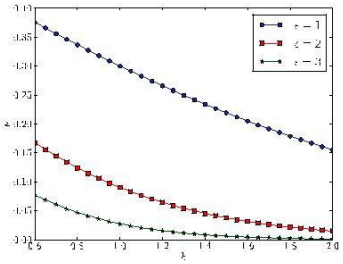


Fig:ρ versus h, when $\rho_0 = 0{:}1$ and $\rho_1 = 0{:}9$.

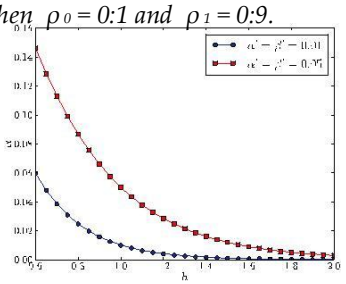
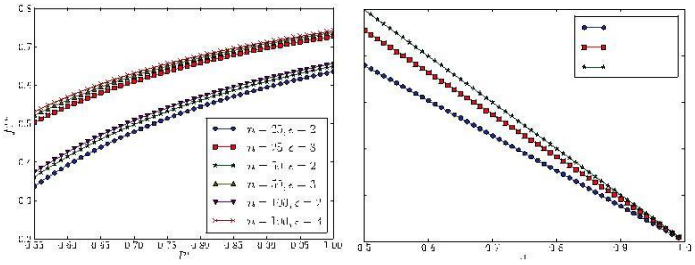


Fig:ρ versus h when $\rho_0 = 0{:}3$ and $\rho_1 = 0{:}7$.

Since the success probability ϱ in the Bernoulli distribution is a parameter required to estimate α , we take into account ϱ for the study of 's impact on . We consider two cases: $\alpha^1 = \beta^1 = 0{:}01$ and $\alpha^1 = \beta^1 = 0{:}05$. We also examine these two cases: $\varrho_0 = 0{:}1$ and $\varrho_1 = 0{:}9$, $= 0{:}3$ and $\varrho_1 = 0{:}7$. Moreover, we set to 1, 2, or 3 such that $\epsilon = 1$ and $\epsilon = 2$; 3 indicate the SPRT and Biased-SPRT, respectively. As shown in Fig. 3, decreases as h increases and as ^{0} and ^{0} decrease. As shown in Figs, diminishes as h and ϱ_1 increase and as ϱ_0 decreases. We also notice that a rise in contributes to a decrease in when h is fixed. Given that $h = 1{:}0$, $\alpha^1 = \beta^1 = 0{:}01$, and $\varrho_0 = 0{:}1$ and $\varrho_1 = 0{:}9$ (Case I), we have 0:01 and = 0:1, 0:01, 0:001 when $\epsilon = 1, 2, 3$, respectively. This means that the false positive rates are estimated as one percent as long as a region-Credence value is measured to be less than Credence threshold with a probability of 0.1 (respectively 0.01, 0.001) when the SPRT (respectively Biased-SPRT) is employed. Thus, the Biased-SPRT requires to be lower than the one in the SPRT in order to achieve high-detection accuracy. Given that $h = 0{:}6$, $\alpha^1 = \beta^1 = 0{:}01$, and $\varrho_0 = 0{:}1$ and $\varrho_1 = 0{:}9$ (Case II), we have $\alpha \sim 0{:}06$ and $\varrho = 0{:}05, 0{:}013$ when $\epsilon = 2; 3$, re-

spectively. This indicates that the Biased-SPRT with lower value of h can achieved



less but still reasonable false positive rates at higher value of . When $_0$ and $_1$ in Cases I and II are configured to a larger value (0.3) and a smaller value (0.7), respectively, we see that the Biased-SPRT fulfills the same false positive rates as Cases I and II while requiring higher value of than Cases I and II.

7 Conclusion

In this proposed scheme region-based node compromise detection and revocation scheme for static sensor networks using the SPRT is enhanced with the robustness of the SPRT [22] with biased sampling. We have shown that our scheme achieves robust untrustworthy region detection capability even if a majority of nodes in each region are compromised. Furthermore, we have proposed countermeasures against the attacks that might be launched to disrupt the proposed scheme. As a future enhancement a new scheme will be proposed with the better detection and revocation using SPRT for dynamic environment in WSN.

References

[1] T. Abuhmed, N. Nyamaa, and D. Nyang, "Software-Based Remote Code Attestation in Wireless Sensor Network," Proc. IEEE GLOBECOM, Dec. 2009.

[2] S. Capkun and J.P. Hubaux, "Secure Positioning in Wireless Networks," IEEE J. Selected Areas in Comm., vol. 24, no. 2, pp. 221-232, Feb. 2006.

[3] T.M. Cover and J.A. Thomas, Elements of Information Theory. Wiley-Interscience, 2006.

[4] F. Delgosha and F. Fekri, "Threshold Key-Establishment in Distributed Sensor Networks Using a Multivariate Scheme," Proc. IEEE INFOCOM, Apr. 2006.

[5] W. Du, J. Deng, Y.S. Han, S. Chen, and P. Varshney, "A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge," Proc. IEEE INFOCOM, pp. 586-597, Mar. 2004.

[6] S. Ganeriwal and M. Srivastava, "Reputation-Based Framework for High Integrity Sensor Networks," Proc. ACM Workshop Security of Ad Hoc and Sensor Networks (SASN), Oct. 2004.

[7] J. Ho, M. Wright, and S.K. Das, "Fast Detection of Replica Node At tacks in Sensor Networks Using Sequential Analysis," Proc. IEEE IN FOCOM, Apr. 2009.

[8] J. Ho, M. Wright, and S.K. Das, "RegionCredence: Fast Region-Based Node Compromise Detection and Revocation in Sensor Networks Using Sequential Analysis," Proc. IEEE Symp. Reliable Distributed Systems (SRDS), Sept. 2009.

[9] X. Hu, T. Park, and K.G. Shin, "Attack-Tolerant Time-Synchronization in Wireless Sensor Networks," Proc. IEEE INFOCOM, Apr. 2008.

[10] J. Jung, V. Paxon, A.W. Berger, and H. Balakrishnan, "Fast Port Scan Detection Using Sequential Hypothesis Testing," Proc. IEEE Symp. Security and Privacy (S&P), May 2004.

[11] C. Karlof and D. Wagner, "Secure Routing in Wireless Sensor Net works: Attacks and Countermeasures," Proc. IEEE Workshop Sensor Network Protocols and Applications, May 2003.

[12] D. Knuth, The Art of Computer Programming, third ed., vol. 2, pp. 145-146. Addison-Wesley, 1998.

[13] F. Li and J. Wu, "Mobility Reduces Uncertainty in {MANET}," Proc. IEEE INFOCOM, May 2007.

[14] Z. Li, W. Trappe, Y. Zhang, and B. Nath, "Robust Statistical Methods for Securing Wireless Localization in Sensor Networks," Proc. Fourth Int'l Symp. Information Processing in Sensor Networks (IPSN), Apr. 2005.

[15] T. Park and K.G. Shin, "Soft Tamper-Proofing via Program Integrity Verification in Wireless Sensor Networks," IEEE Trans. Mobile Computing, vol. 4, no. 3, pp. 297-309, May/June 2005.

[16] B. Parno, A. Perrig, and V.D. Gligor, "Distributed Detection of Node Replication Attacks in Sensor Networks," Proc. IEEE Symp. Security and Privacy (S&P), May 2005.

[17] A. Seshadri, A. Perrig, L. van Doorn, and P. Khosla, "SWATT: Soft Ware-Based Attestation for Embedded Devices," Proc. IEEE Symp. Security and Privacy (S&P), May 2004.

[18] K. Sun, P. Ning, C. Wang, A. Liu, and Y. Zhou, "TinySeRSync: Secure and Resilient Time Synchronization in Wireless Sensor Networks," Proc. ACM Conf. Computer and Comm. Security (CCS), Oct. 2006.

[19] Y. Sun, Z. Han, W. Yu, and K. Liu, "A Credence Evaluation Framework in Distributed Networks: Vulnerability Analysis and Defense against Attacks," Proc. IEEE INFOCOM, Apr. 2006.

[20] G. Theodorakopoulos and J.S. Baras, "Game Theoretic Modeling of Malicious Users in Collaborative Networks," IEEE J. Selected Areas in Comm., vol. 26, no. 7, pp. 1317-1326, Sept. 2008.

[21] D. Wagner, "Resilient Aggregation in Sensor Networks," Proc. ACM Second ACM Workshop Security of Ad Hoc and Sensor Networks (SASN '04), 2004.

[22] A. Wald, Sequential Analysis. Dover Publications, 2004.

[23] Y. Yang, X. Wang, S. Zhu, and G. Cao, "Distributed Software- Based Attestation for Node Compromise Detection in Sensor Networks," Proc. IEEE 26th Int'l Symp. Reliable Distributed Systems (SRDS), Oct. 2007.

[24] F. Ye, G. Zhong, S. Lu, and L. Zhang, "A Robust Data Delivery Proto col for Large Scale Sensor Networks," Proc. Second Int'l Conf. Infor mation Processing in Sensor Networks (IPSN), Apr. 2003.

[25] F. Ye, H. Yang, and Z. Liu, "Catching 'Moles' in Sensor Networks," Proc. 27th Int'l Conf. Distributed Computing Systems (ICDCS), June 2007.

[26] W. Zhang, M. Tran, S. Zhu, and G. Cao, "A Random Perturbation-Based Scheme for Pairwise Key Establishment in Sensor Networks," Proc. ACM Mobihoc, Sept. 2007.

[27] Y. Zhang, J. Yang, L. Jin, and W. Li, "Locating Compromised Sensor Nodes through Incremental Hashing Authentication," Proc. Int'l Conf. Distributed Computing in Sensor Systems (DCOSS), June 2006.