Information Leaks Out: Attacks and Countermeasures on Compressive Data Gathering in Wireless Sensor Networks

Pengfei Hu*, Kai Xing*, Xiuzhen Cheng† Hao Wei*, Haojin Zhu ‡,
*University of Science and Technology of China, Anhui, China 230027
Email: {kxing}@ustc.edu.cn, {pfhu, weihao}@mail.ustc.edu.cn
†George Washington University, USA
Email: cheng@gwu.edu
‡Shanghai Jiao Tong University, Shanghai, China
Email: zhu-hj@sjtu.edu.cn

***Abstract*—Compressive sensing (CS) has been viewed as a promising technology to greatly improve the communication efficiency of data gathering in wireless sensor networks. However, this new data collection paradigm may bring in new threats but few study has paid attention to prevent information leakage during compressive data gathering. In this paper, we identify two statistical inference attacks and demonstrate that traditional compressive data gathering may suffer from serious information leakage under these attacks. In our theoretical analysis, we quantitatively analyze the estimation error of compressive data gathering through extensive statistical analysis, based on which we propose a new secure compressive data aggregation scheme by adaptively changing the measurement coefficients at each sensor and correspondingly at the sink without the need of time synchronization. In our analysis, we show that the proposed scheme could significantly improve data confidentiality at light computational and communication overhead.**

I. Introduction

Wireless sensor networks (WSNs) have been widely deployed for various applications such as environment monitoring [1], event detection [2]–[4], target counting and tracking [5]–[8], just to name a few. A major task of the sensors for such an application is to report the relevant data to a sink node. Due to the strict energy limitation and the common vulnerability of WSNs, secure in-network data aggregation has been proposed as an essential approach to addressing the efficiency and security issues of data gathering in WSNs.

Various secure data aggregation techniques have been investigated to collect the real sensor signals [9] [10]. These approaches can be classified into three categories, namely hop-by-hop encrypted data aggregation [11], end-to-end encrypted data aggregation [12], [13], and secure hierarchical data aggregation [14]. A common weakness of these approaches is the significant communication overhead brought by in-network data processing, which brings significant challenges for the sensor network to improve both communication and data aggregation efficiency in practice.

Recent advances in Compressive Sensing (CS) provide a great opportunity to improve the communication and data aggregation efficiency in wireless sensor networks [15] [16] [17] [18] [19]. The basic idea is to multiply each raw sensor reading with a random measurement vector and then simply sum the partial projected results at each non-leaf node along the routing paths (tree) to the sink. Such a process is termed *compressive data gathering*, or *compressive sensing based data aggregation*. By exploiting the data redundancy in the spacial domain, compressive data gathering can accurately reconstruct the original sensor readings by a relatively small number of samples at the data sink, and the operations at each node are much simpler compared to existing non-CS based approaches. Thus the communication and data aggregation efficiency in compressive data gathering can be significantly improved. Nevertheless, the security of compressive data gathering is generally overlooked.

In this paper, we address the security issues of compressive data gathering in WSNs and identify the possibility of information leakage during data aggregation via statistical inference. We show that if there exists a node compromised by an attacker, the data of the subnetwork controlled by the compromised node could be released to the attacker via the following two statistical attacks: *controllable event triggering attack* and *random event triggering attack*. Comparing with the related literature summarized in Section II, we have identified the following unique contributions of this paper.

- We address the security issues of compressive data gathering in WSNs and demonstrate the possibility of serious information leakage when a regular sensor node is compromised. To our best knowledge, this is the first work to investigate the security of compressive data gathering.
- We propose two statistical inference attacks that could cause information leaks via a compromised node in compressive data gathering. An attacker only needs to compromise one sensor in the network, which implies the easiness and high possibility of information leakage. In our evaluation study, we demonstrate that the proposed attacks could effectively recover the sensor readings along the aggregation tree with an acceptable fidelity.

978-1-4799-3360-0/14/$31.00 ©2014 IEEE

- We design a new Secure Compressive Data Gathering Scheme (SCDG) to prevent possible statistical inference based attacks during the data gathering process under the existence of compromised nodes. The evaluation results indicate that the proposed scheme could well protect the network under an affordable SNR loss of 3db on average.

The rest of the paper is organized as follows: Section II presents the most related work. The preliminaries, models, and assumptions are introduced in Section III. Section IV details the two statistical inference attacks and Section V provides our theoretical analysis and evaluation results. According to our analysis on these two attacks, we propose a new secure compressive data gathering scheme (SCDG) in Section VI, and conclude the paper in Section VII.

II. Related Work

In this section, we briefly overview the data aggregation techniques in WSNs, summarize the approaches for secure data aggregation, and then outline the most related work of compressive data gathering.

Distributed source coding [20] [21] [22]. [23] is a typical data gathering approach that explores the correlation among multiple correlated sensor readings for in-network data compression. Considering that sensor readings in a WSN have temporal and spatial correlations, clustered data aggregation [24] [25] [26] suggests separating the entire network into several clusters and each cluster is represented by one single node [27]. Another promising data gathering technique is the collaborative wavelet transform [28] [29], which compresses the sensor data by wavelet transforms. Due to in-network processing, all these methods incur significant communication and computational overhead, which brings difficulties for sensors to improve their data aggregation efficiency in practice.

Recently, the Compressive Sensing (CS) theory motivates compressive data gathering to reduce the energy consumption and greatly improve the communication and data aggregation efficiency (e.g., [15] [16] [17] [18] [19]). Based on the CS theory [30] [31], when a signal can be sparsely decomposed in some domain, it can be accurately reconstructed by a relatively small number of samples based on sparse recovery techques such as ℓ_1-minimization [30], Orthogonal Matching Pursuit (OMP) [32], Greedy Matching Pursuit (GMP) [5], and Regularized Orthogonal Matching Pursuit (ROMP) [33]. In compressive data gathering, each sensor reading is multiplied with a measurement vector and all the results are combined at the non-leaf nodes along the routing tree to reach the sink. The component that consumes the most energy is the signal reconstruction, which is processed at the sink; while the data aggregation at the sensors is relatively lightly-weighted. Since compressive data gathering does not cause unbalanced energy consumption in the network, it performs well on load-balancing and can help to extend the network lifetime [34].

Due to the typically remote and hostile deployment environments, it is essential to enforce secure data aggregation for high data fidelity [9] [10]. Secure data aggregation could be considered from three categories: hop-by-hop encrypted data aggregation, end-to-end encrypted data aggregation, and secure hierarchical data aggregation. Hop-by-hop encrypted data aggregation is proposed in [11], which dynamically partitions the nodes within a tree topology into multiple groups. End-to-end encrypted data aggregation [13] makes use of digital signatures and homomorphic encryption to achieve confidentiality. The detection of an intruder's manipulation over compromised nodes during data aggregation is studied in [14]. SIA [35] requires a Merkle-hash-tree based commitment to the data, through which users can ask the aggregator for authentication later. The integrity of the aggregation's result can be verified with the help of multiple witness nodes [36]. It can also be protected by cluster-based privacy preserving data aggregation [37]. In [38], the VMAT protocol realizes security through revoking malicious sensors in a timely manner and thus the capability of attackers can be reduced.

Existing works such as [39], [40] have made effort to protect the secrecy of compressive sensing. However, they mainly focus on outsider attacks, e.g., eavesdroppers. In this paper, we design two statistical inference attacks on compressive data gathering, which can recover a general measurement matrix. We also propose corresponding countermeasures against these attacks. The most related work of measurement matrix recovery is [41]; but it could only recover certain structured measurement matrices. In our study, via the help of a compromised node, an attacker can recover a general measurement matrix.

III. Preliminaries, Models, and Assumptions

A. Compressive Sensing based Data Aggregation in WSNs

Suppose a signal $\boldsymbol{x}$, denoted by a $N \times 1$ vector, can be sparsely decomposed in some domain $\boldsymbol{\Psi}$:

$$\boldsymbol{x} = \boldsymbol{\Psi s} \tag{1}$$

where $\boldsymbol{\Psi}$ is an $N \times N$ basis matrix and $\boldsymbol{s}$ is the representation of $\boldsymbol{x}$ in $\boldsymbol{\Psi}$. We say $\boldsymbol{x}$ is K-sparse in $\boldsymbol{\Psi}$ when only K elements of $\boldsymbol{s}[j]$ are nonzero and the other $N-K$ elements are zero.

In this paper, we consider a static network of N sensors, in which a sink node collects data from various sensors along aggregation paths forming a tree topology. Let $\boldsymbol{x}_i$ denote the readings of the ith-round sampling of a sensor network, with element x_{ij} corresponding to sensor j's reading, and $\boldsymbol{\Phi}$ denote an $M \times N$ measurement matrix, with the column vector $\boldsymbol{\phi_j}$ assigned to a sensor P_j, where the elements of $\boldsymbol{\Phi}$ can be chosen based on Gaussian distributions.

Here we use an example shown in Fig.1 to illustrate the compressive data gathering process. Taking the subtree rooted at P_0 as an example. After all nodes obtain their readings at the ith-round sampling, each node P_j multiplies its reading x_{ij} by its coefficient vector $\boldsymbol{\phi_j}$. Then P_3 and P_4 send their results $\boldsymbol{\phi_3} x_{i3}$ and $\boldsymbol{\phi_4} x_{i4}$ to P_2. After receiving the results from P_3 and P_4, P_2 adds them with $\boldsymbol{\phi_2} x_{i2}$ and sends the summation $\sum_{j=2}^{4} \boldsymbol{\phi_j} x_{ij}$ to P_1. Finally, P_0 gets the aggregation result $\sum_{j=1}^{4} \boldsymbol{\phi_j} x_{ij}$ of this subtree. With the same approach, the sink can obtain the aggregation result of the ith round

sampling of the whole network, which is,

$$\boldsymbol{y}_i = \boldsymbol{\Phi}\boldsymbol{x}_i \tag{2}$$

In traditional compressive data gathering, each sensor P_j is preinstalled a random coefficient vector $\boldsymbol{\phi_j}$, or computes its random coefficient vector based on a seed known to the sink only, in order to avoid sending the measurement matrix $\boldsymbol{\Phi}$ from the sensors to the sink.

The CS theory demonstrates that when M satisfies

$$M \geq cK \log \frac{N}{K} \tag{3}$$

where c is a positive constant, and the product $\boldsymbol{\Phi} \cdot \boldsymbol{\Psi}$ satisfies the Restricted Isometry Property (RIP) [30], the sparse signal $\boldsymbol{s}$ can be precisely recovered with a very high probability via the ℓ_1-minimization:

$$\boldsymbol{s} = \min \|\boldsymbol{s}\|_1, s.t. \boldsymbol{y} = \boldsymbol{\Theta}\boldsymbol{s} \tag{4}$$

where $\boldsymbol{\Theta} = \boldsymbol{\Phi}\boldsymbol{\Psi}$. The CS theory also states that if $\boldsymbol{\Phi}$ is random, $\boldsymbol{\Psi}$ is universal, which means that most random matrix $\boldsymbol{\Phi}$ can preserve RIP with a high probability; but the reconstruction can be affected by $\boldsymbol{\Phi}$ to certain degree.

In existing compressive data gathering approaches, the summation procedure at each sensor p_j needs to add $\boldsymbol{\phi_j} x_{ij}$ to the summation results it receives. In the following, we show that this aggregation procedure may cause potential information leakage in the network because the coefficient matrix $\boldsymbol{\Phi}$ can be estimated by an adversary through statistical inference. With a good estimation of the coefficient matrix, the attacker can easily recover the signal $\boldsymbol{x}$.

B. Security Model

Because of the common vulnerabilities of WSNs, an attacker can capture and compromise a sensor easily. Thus the attacker can get the summation of the subtree rooted at the compromised node. For simplicity, we assume that there exists one compromised node $Node_c$ in the network.

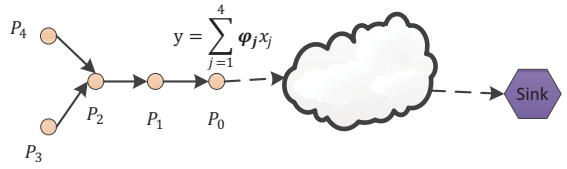


Fig. 1. The basic diagram of data aggregation in a network of N sensors

We further assume that the attacker has the ability to trigger events at arbitrary places in the network. Besides, we assume that the attacker has the topology information of the aggregation subtree rooted at the compromised node $Node_c$. We also assume that the attacker has the domain knowledge of $\boldsymbol{\Psi}$. Specifically, $\boldsymbol{\Psi}$ could be learned from a training data set collected by the compromised nodes or the attacker. We adopt the following notations:

- x_{ij}: The reading of sensor P_j at the ith-round sampling.
- u_{ij}: The value of $x_{(i+1)j} - x_{ij}$.
- y_i: The aggregated result of the ith-round sampling at the compromised node $Node_c$ (excluding the reading of $Node_c$). For example in Fig. 1, $y_i = \sum_{j=1}^{4} \boldsymbol{\phi_j} x_{ij}$ if P_0 is the compromised node.
- v_i: The result of $y_{i+1} - y_i$.

IV. Statistical Inference Attacks

As suggested in (2), sensor readings are multiplied with the measurement matrix $\boldsymbol{\Phi}$ along the data aggregation paths in compressive data gathering. If a sensor is compromised, the aggregated results become available to the attacker. Observing that the attacker could probably recover sensors' readings given the measurement matrix $\boldsymbol{\Phi}$ and the domain knowledge $\boldsymbol{\Psi}$, we propose the following two attacks aiming to obtain the measurement matrix $\boldsymbol{\Phi}$: *controllable event triggering attack (CETA)*, and *random event triggering attack (RETA)*. These two attacks are mainly based on the Least Square Estimation, and the attacker can launch one that meets its ability.

A. Controllable Event Triggering Attack ($CETA$)

According to the security model proposed in Section III-B, the attacker has the ability to trigger events at arbitrary places in the network. Let P_j be the target sensor of the attacker to trigger an event. Note that the attacker could trigger an event around P_j but the impact of the event is limited to P_j only; i.e., only P_j is affected by the event. This attack is termed as controllable event triggering attack (CETA). For example, the attacker can set a fire at P_j to increase the temperature readings of P_j.

During each round of sampling, the attacker launches a CETA attack at P_j once and gets a summation result y_i at the compromised node $Node_c$. After the attacker triggers c rounds of the event, it can get c tuples of the data $< x_{1j}, y_1 >, \ldots, < x_{cj}, y_c >$. We have

$$v_i = y_{i+1} - y_i = \sum_{k \in S} \phi_k u_{ik} + \phi_j u_{ij} \approx \phi_j u_{ij} \tag{5}$$

where S is the set of sensors on the aggregation subtree (excluding the target node). Note that it is reasonable to assume that the readings of the nodes in S remain almost unchanged between two successive rounds of samplings; thus $\sum_{k \in S} \phi_k u_{ik} \approx 0$.

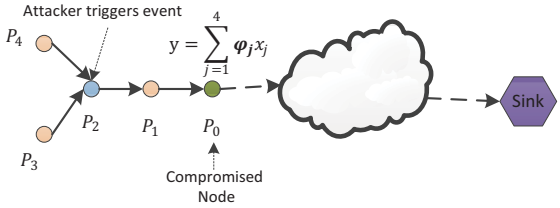


Fig. 2. Controllable Event Triggering Attack. The attacker triggers an event at the target sensor P_2 in order to gain its measurement coefficients. Meanwhile, it also compromises the sensor P_0.

For example in Fig.2, the attacker triggers an event around target node P_2 at each sampling round. This event can significantly affect the reading of P_2, but has no influence on other sensors. The attacker also compromises P_0 and thus it can get the summation $\sum_{j=1}^{4} \phi_j x_{ij}$ of each sampling round. P_1, P_3, P_4 are in S and their readings stay almost unchanged between two continuous sampling rounds. Thus the difference of two continuous aggregation results is mainly caused by the change of P_2's readings, which implies that $v_i = \phi_2 u_{i2}$.

Note that some of the sensor readings may vary significantly between two consecutive rounds of samplings. In order to provide a good estimate of $\hat{\phi}_j$, we propose the following data filtering strategy: When triggering an event, the attacker controls the influence of the event on P_j to ensure that $u_{ij} = C$, where C is a constant. As a result, $\phi_j u_{ij}$ should be a constant $\phi_j C$ too. We further define δ as the deviation of the summations caused by all other sensor readings. Thus if there is no other event occurring in the network during the interval of collecting the summation y_i and $y_i + 1$, v_i should satisfy a Gaussian distribution $\mathcal{N}(\phi_j C, \delta^2)$, where δ could be learned from the history data of the compromised node. Let $Med(V)$ denote the median of all v_is and Δ denote the estimated value of 2δ. The attacker can simply maintain a data set $\{v_i \mid v_i \in (Med(V) - \Delta, Med(V) + \Delta)$. Therefore the data not in this set are very likely influenced by other unknown events and thus should not be used to estimate ϕ_j.

Let $\boldsymbol{R}$ denote the data set and K denote the number of elements in $\boldsymbol{R}$. We have

$$v_i = \phi_j u_{ij} = \phi_j C \tag{6}$$

$$\hat{\phi}_j = \frac{\sum_{i \in R} v_i}{KC} \tag{7}$$

When launching $CETA$ at the target sensor P_j, the coefficient vector can be recovered through (7) accurately with the help of the data filtering strategy. Furthermore, from (7), one can see that $CETA$ does not incur accumulative errors.

B. Random Event Triggering Attack ($RETA$)

According to the security model proposed in Section III-B, the attacker can trigger events at a randomly chosen area in the network. Such an attack is termed as Random Event Triggering Attack (RETA). Specifically, the influence region of the triggered event in RETA could be an area rather than just one sensor. Let H denote the set of nodes in the influence region of the event, and S denote the set of sensors ahead of P_j on the aggregation path that participate in data aggregation (the nodes in H do not belong to S).

The process of $RETA$ is similar to $CETA$. During each round of sampling, the attacker launches RETA once at a randomly chosen area and gets a summation result y_i at the compromised node $Node_c$. Similarly, we assume that the readings of the sensors in S vary in a limited range between two successive rounds; thus the difference of the successive summations is mainly caused by the reading changes of the sensors in H.

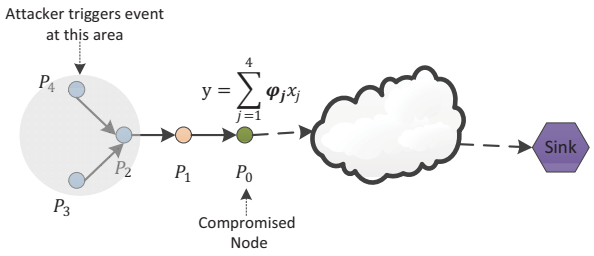


Fig. 3. Random Event Triggering Attack. The influence region of the triggered event includes P_2,P_3 and P_4.

After collecting enough data, the coefficient vectors of the sensors ahead of the compromised node $Node_c$ in the routing tree could be recovered based on the multi-coefficient least square theory. Specifically,

$$v_i = y_{i+1} - y_i = \sum_{j \in S,H} \phi_j u_{ij} = \sum_{j \in H} \phi_j u_{ij} \tag{8}$$

After c rounds, we have

$$\boldsymbol{\Phi}_{\boldsymbol{M \times |H|}}(\boldsymbol{H}) \;\; \boldsymbol{U}_{\boldsymbol{c \times |H|}}(\boldsymbol{H})^T = \boldsymbol{V}_{\boldsymbol{M \times c}}(\boldsymbol{H}) \tag{9}$$

Based on the multi-coefficient least square estimate, $\boldsymbol{\Phi}$ can be estimated given that the number of sampling rounds c is large enough

$$\boldsymbol{\Phi}(\boldsymbol{H})\boldsymbol{U}(\boldsymbol{H})^T\boldsymbol{U}(\boldsymbol{H}) = \boldsymbol{V}(\boldsymbol{H})\boldsymbol{U}(\boldsymbol{H}) \tag{10}$$

$$\boldsymbol{\Phi}(\boldsymbol{H}) = \boldsymbol{V}(\boldsymbol{H})\boldsymbol{U}(\boldsymbol{H})(\boldsymbol{U}(\boldsymbol{H})^T\boldsymbol{U}(\boldsymbol{H}))^{-1} \tag{11}$$

According to the derivations above, $RETA$ does not possess accumulative errors because all coefficients are calculated simultaneously without making use of the prior results. Therefore $RETA$ is also suitable for attacking large-scale sensor networks.

V. Theoretical Analysis

In this section, we analyzes the two attacks by conducting error analysis based on the error transfer formula.

Given $y = f(x_1, \ldots, x_n)$, according to the total derivative formula, the variance of y is:

$$\sigma_y^2 = \sum_{i=1}^{n} (\frac{\partial f}{\partial x_i})^2 \sigma_{x_i}^2 \tag{12}$$

In the following we use this formula to analyze the accuracy of the estimated measurement matrix of the two attacks.

A. Theoretical Analysis of $CETA$

Based on error transfer formula (12), the variance of ϕ_j is:

$$\sigma_{\hat{\phi}_j}^2 = \sum_{i=1}^{c} (\frac{\partial f}{\partial v_i})^2 \sigma_{v_i}^2 = \frac{\sigma_{v_i}^2}{KC^2} \tag{13}$$

Considering that

$$v_i = \sum_{k \in S} \phi_k u_{ik} \tag{14}$$

Let σ_u represent the standard deviation of the sensor readings in the node set S, we have

$$\sigma_{v_i}^2 = \sum_{k\in S}(\frac{\partial v_i}{\partial u_{ik}})^2\sigma_u^2 = \sum_{k\in S}\phi_k^2\sigma_u^2 \tag{15}$$

Through (13) and (15), we can obtain the variance of ϕ_j, $\sigma_{\hat{\phi}_j}^2$. According to (13), the attacker could increase the constant C to reduce its estimation error $\sigma_{\hat{\phi}_j}^2$. Furthermore, the standard deviation of each node's estimated coefficient vector should remain almost the same if the attacker keeps C unchanged during the attack process in the network.

B. Theoretical Analysis of $RETA$

The error analysis of $RETA$, which takes advantage of the multi-coefficient least square estimation, is mainly based on the following theorem:

Theorem 5.1: Given $n\times 1$ vectors $l_1 = (1,0,\ldots,0)^T$, $l_2 = (0,1,\ldots,0)^T$, ……, $l_n = (0,0,\ldots,1)^T$, and $n\times 1$ vectors $d_1 = (d_{11},d_{12},\ldots,d_{1n})^T$, $d_2 = (d_{21},d_{22},\ldots,d_{2n})^T$, ……, $d_n = (d_{n1},d_{n2},\ldots,d_{nn})^T$, by solving the equations below:

$$\begin{cases} U^TUd_1 = & l_1 \\ U^TUd_2 = & l_2 \\ & \vdots \\ U^TUd_n = & l_n \end{cases} \tag{16}$$

we can obtain d_{11}, d_{22}, …, d_{nn}. Then the coefficients, calculated based on the multi-coefficient least square theory, have the following standard deviations:

$$\begin{cases} \sigma_{\phi_1} = & \sigma_v\sqrt{d_{11}} \\ \sigma_{\phi_2} = & \sigma_v\sqrt{d_{22}} \\ & \vdots \\ \sigma_{\phi_n} = & \sigma_v\sqrt{d_{nn}} \end{cases} \tag{17}$$

Proof: The matrix U can be written as $(U_1,U_2,\ldots,U_n)$, where U_i is a $n\times 1$ column vector; and (10) can be written as:

$$\begin{cases} \phi_1[U_1^TU_1]+\ldots+\phi_n[U_n^TU_1] = & VU_1 \\ \phi_1[U_1^TU_2]+\ldots+\phi_n[U_n^TU_2] = & VU_2 \\ & \vdots \\ \phi_1[U_1^TU_n]+\ldots+\phi_n[U_n^TU_n] = & VU_n \end{cases} \tag{18}$$

Multiplying the ith equation with d_{1i}, we obtain

$$\begin{cases} \phi_1[U_1^TU_1]d_{11}+\ldots+\phi_n[U_n^TU_1]d_{11} = & VU_1d_{11} \\ \phi_1[U_1^TU_2]d_{12}+\ldots+\phi_n[U_n^TU_2]d_{12} = & VU_2d_{12} \\ & \vdots \\ \phi_1[U_1^TU_n]d_{1n}+\ldots+\phi_n[U_n^TU_n]d_{1n} = & VU_nd_{1n} \end{cases} \tag{19}$$

Adding up these equations, we have

$$\sum_{r=1}^{n}\phi_1[U_1^TU_r]d_{1r}+\ldots+\sum_{r=1}^{n}\phi_n[U_n^TU_r]d_{1r} = \sum_{r=1}^{n}VU_rd_{1r} \tag{20}$$

Then we select $d_{11},d_{12},\ldots,d_{1n}$ satisfying:

$$\begin{cases} \sum_{r=1}^{n}[U_1^TU_r]d_{1r} = & 1 \\ \sum_{r=1}^{n}[U_2^TU_r]d_{1r} = & 0 \\ & \vdots \\ \sum_{r=1}^{n}[U_n^TU_r]d_{1r} = & 0 \end{cases} \tag{21}$$

Note that

$$\begin{cases} h_{11} = & d_{11}U_{11}+d_{12}U_{12}+\ldots+d_{1n}U_{1n} \\ h_{12} = & d_{11}U_{21}+d_{12}U_{22}+\ldots+d_{1n}U_{2n} \\ & \vdots \\ h_{1n} = & d_{11}U_{n1}+d_{12}U_{n2}+\ldots+d_{1n}U_{nn} \end{cases} \tag{22}$$

then (20) becomes:

$$\begin{aligned} \phi_1 = & \sum_{r=1}^{n}VU_rd_{1r} \\ = & v_1(d_{11}U_{11}+d_{12}U_{12}+\ldots+d_{1n}U_{1n})+ \\ & \ldots+v_n(d_{11}U_{n1}+d_{12}U_{n2}+\ldots+d_{1n}U_{nn}) \\ = & v_1h_{11}+v_2h_{12}+\ldots+v_nh_{1n} \end{aligned} \tag{23}$$

where v_i is the ith value of the vector V. Because $v_1,v_2,\ldots,v_n$ are independent and should have similar standard deviation, which is denoted by σ_v, we have

$$\sigma_{\phi_1}^2 = (h_{11}^2+h_{12}^2+\ldots+h_{1n}^2)\sigma_v^2 \tag{24}$$

Based on (21) and (22), we get:

$$\sigma_{\phi_1}^2 = d_{11}\sigma_v^2 \tag{25}$$

The proof of $\sigma_{\phi_2},\sigma_{\phi_3},\ldots,\sigma_{\phi_n}$ is similar. ■

Let σ_{v_i} denote the standard deviation caused by the variation of u_{ik}, $\forall k\notin H, k\in S$, we have

$$\sigma_{v_i}^2 = \sum_{k\notin H,k\in S}(\frac{\partial v_i}{\partial u_{ik}})^2\sigma_u^2 = \sum_{k\notin H,k\in S}\phi_k^2\sigma_u^2 \tag{26}$$

Based on Theorem 5.1, we can solve (16) and calculate the standard deviations through (17) and (26).

C. Evaluation of the Two Attacks

This section conducts both numerical study based on the analysis in Section V and simulation study of the impact of the two attack methods on compressive data gathering. Specifically, we use a Gaussian random matrix as the measurement matrix, and Orthogonal Matching Pursuit (OMP) [32] as the reconstruction method, both of which are well accepted in compressive sensing based data aggregation for wireless sensor networks [32] [33] [42] [34].

In our simulation, we deploy $10k$ sensors in a line network – actually, our method can be applied to any kind of tree topology for data aggregation. The aggregation path we use

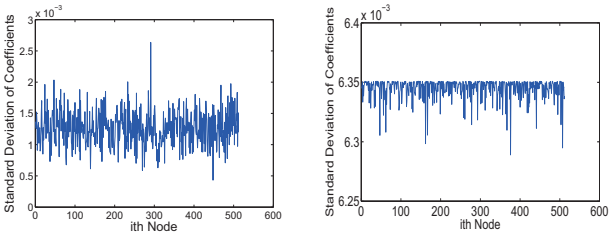
in our simulation contains the first 512 nodes. The data of sensor P_j at each sampling round is generated by $data_j = \frac{c}{(2+0.05j)^2} + \epsilon$, where c is a random number drawn from a continuous uniform distribution on the interval $(20, 25)$, and ϵ is the stochastic error that is normally distributed with a mean 0 and a standard deviation 0.1. All the results are averaged over 10 runs (at each run, we re-generate the constant c). When the attacker launches a CETA attack, it first compromises the sensor P_{512}, then triggers events around each sensor from P_0 to P_{511}. By this way the attacker can successively obtain the coefficient vector of each sensor. The process of RETA attack is similar to CETA, except that the attacker triggers an event in a certain area affecting multiple sensors.

We apply SNR [42] to represent the reconstruction effect:

$$SNR = 10log_{10}\frac{\sum_{j=1}^{N} x_j^2}{\sum_{j=1}^{N}(x_j - \hat{x_j})^2} \quad (27)$$

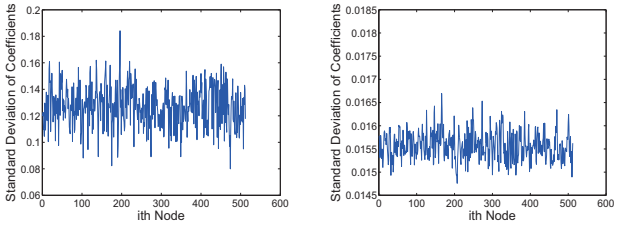
where $\hat{x_i}$ is the reconstructed data of sensor P_j.

Figs. 4(a) and 5(a) report the standard deviations of the estimated coefficients based on our simulation study. One can see that the simulated standard deviations of the estimated coefficients of $CETA$ and $RETA$ are relatively stable. This is because both attacks do not possess accumulative errors. We also observe that the standard deviations of the estimated coefficients of CETA are smaller according to Fig. 4(a), compared with those in Fig. 5(a). This is because the attacker filters the abnormal readings through an appropriate threshold δ (we set $\delta = 0.1$ in our simulation).



(a) Simulation Results (b) Numerical Results

Fig. 4. The coefficient standard deviations of $CETA$

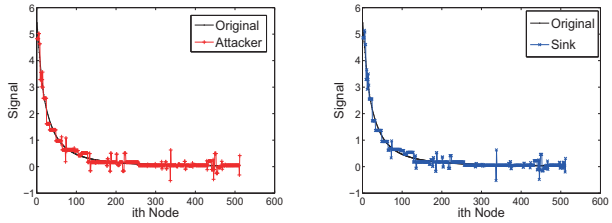


(a) Simulation Results (b) Numerical Results

Fig. 5. The coefficient standard deviations of $RETA$

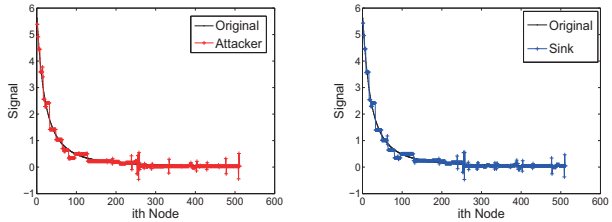
Figs. 4(b) and 5(b) report the numerical results of the standard deviations of the coefficients based on the analysis in Section V. One can see that the trends of the standard deviations of CETA and RETA obtained from simulation study agree with those of the numerical results in both Figs. 4 and 5. The numerical results of the standard deviations of $CETA$ and $RETA$ are relatively stable as shown in Figs. 4(b) and Fig. 5(b). These two figures indicate that the scale of wireless sensor networks has little impact on $CETA$ and $RETA$, because they do not have accumulative errors.

Figs. 6(a) and 7(a) demonstrate the recovered signals by the attacker with the estimated coefficients based on the two attacks. Figs. 6(b) and 7(b) illustrate the recovered signals by the sink node with the actual coefficients. We observe that the signal recovered by the attacker is similar to that of the sink, and the maximum reconstruction error is less than 0.5. This indicates that the attacker can recover the original signal with an acceptable fidelity.



(a) Attacker (b) Sink

Fig. 6. The recovered signals by the attacker and the sink with $CETA$.

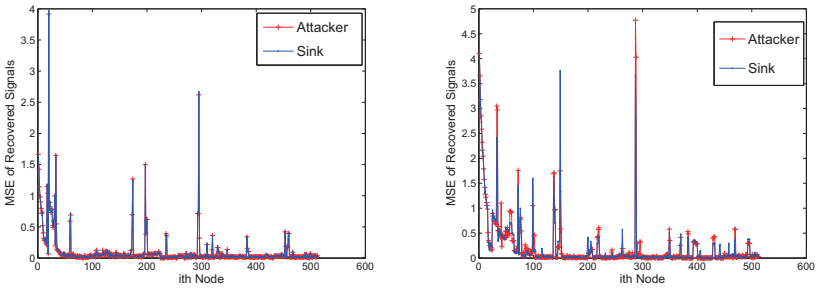


(a) Attacker (b) Sink

Fig. 7. The recovered signal at the attacker and the sink with $RETA$.

Fig. 8 compares the MSE of the signals (for the 512 nodes) recovered by the attacker and the sink. From these figures, one can observe that i) the MSE recovered by the attacker and the sink are very similar, which indicates that the attacker has a similar ability to recover the original signal as the sink, although the attacker only compromises one sensor in the network; and ii) the MSE of most sensors is relatively small (less than 0.05), which indicates that the attacker can recover the original signal with high fidelity.

Fig. 9 demonstrates the quality of the signals reconstructed by the attacker and the sink in terms of SNR. It is interesting to observe that the SNR values of the attacker are close to those of the sink (differ about $0.5dB$). This suggests that the two attacks are powerful enough to give the attacker sufficient information to recover the original signal, making the attack ability of the attacker comparable to that of the sink.



(a) CETA (b) RETA

Fig. 8. The MSE of the recovered signal.

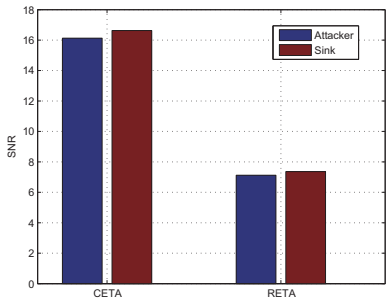


Fig. 9. SNR

VI. Countermeasure

A. Analysis of The Statistical Inference Attacks

Due to the typically remote and hostile deployment environment, it is difficult to provide an effective physical protection to sensors. In many cases, it is almost impossible to completely prevent attackers from triggering events in the sensor network or controlling the source of data aggregation. In order to achieve secure and efficient data collection, we should make compressive data gathering secure, rather than requireing more external protection.

According to the security model introduced in Section III, the confidentiality of the measurement matrix $\mathbf{\Phi}$ is the key to achieving secure compressive data gathering. Based on the observation that the measurement matrices employed by the existing approaches usually remain unchanged, an attacker can collect a large number of samples and then estimate the measurement matrix by statistical inference. Therefore if we can change the measurement matrix wisely during data aggregation, statistical inference based attacks could fail in compressive data gathering.

B. Secure Compressive Data Gathering (SCDG)

Based on the analysis mentioned above, we propose a new scheme called Secure Compressive Data Gathering (SCDG). In SCDG, the measurement matrix is changed at each round of data aggregation. SCDG contains two phases, which are detailed as follows.

1) Broadcast of the Sink: The sink first generates a one-way key chain $\{k_1, k_2, \ldots, k_n\}$ in the following way: it chooses k_n randomly, and applies a secure one-way hash function $k_i = F(k_{i+1})$ sequentially to generate the other keys. According to the property of one-way hash functions, no one can deduce k_{i+1} from k_i. We assume that the sink and the nodes in the network are loosely time synchronized. Time is divided into intervals with a duration of T_{int}. At a time instant t, the sink broadcasts an authenticated message containing certain information to inform the nodes in the network to collect data:

$$sink \longrightarrow nodes : message \mid t \mid T_{int} \mid \delta$$
$$MAC_{k_i}(message \mid t \mid T_{int} \mid \delta)$$

where MAC_{k_i} is the message authentication code based on k_i. After receiving the broadcast message, nodes cannot authenticate the message right away because they do not know the corresponding key k_i. After a delay of δ, the sink reveals the key k_i to all the nodes. Once a node receives the key k_i, it can verify k_i by applying $k_j = F^{i-j}(k_i)$, where k_j is a key received from the sink in the previous broadcast. Note that if the broadcast message or the new key k_i is received after a long delay, the node must drop it because an adversary might have altered it. After the key is verified, the node can then authenticate the message. If both checks are successful, the message is authentic and the node replaces k_j with k_i.

2) Data Aggregation at Each Node: We assume that each node P_i shares with the sink a unique one-way hash key chain $\{s_{i1}, s_{i1}, \ldots, s_{in}\}$, with $s_{ij} = H_i(s_{ij+1})$ (H_i is a one-way hash function shared with the sink. At the beginning, we assume each node P_i shares s_{in} with the sink), and a pseudo-random function F_i. Each key s_{ij} is used as a random seed to generate a random coefficient vector for P_i. In order to prevent the attacker from obtaining the random seed, the sink broadcasts a notification message to the whole network to change the seeds. Once the sensor receives this notification, it applies $s_{ij} = H_i(s_{ij+1})$ to compute the next random seed, which will be used by the sensor before the next notification. From the next round T_k, P_i uses this seed to generate a new random coefficient vector:

$$\phi_i = F_i(s_{ij}, T_k) \tag{28}$$

According to this equation, the coefficient changes based on the secret seed at each round.

Then P_i multiplies its reading x_{ki} with the coefficient vector ϕ_i at the kth round, and transfers the result to the subsequent node in the aggregation tree. At last, the sink can get the aggregation result. As the sink stores the one-way hash key chain, the pseudo-random functions at each node, the time of changing the random seed, and the round of data aggregation, it can easily re-generate the corresponding measurement matrix at each timer interval to recover the original sensor data.

C. Analysis on SCDG

According to the CS theory, the elements of the measurement matrix should satisfy certain distributions such as Gaussian distribution in order to keep RIP. We can easily find a pseudo-random function F_i to use the secret key s_{ij} and the timestamp T_k as seeds to generate random coefficients that meet Gaussian distribution. Thus in SCDG, the ϕ_j calculated from (28) can meet RIP actually.

In order to estimate the measurement matrix accurately, the attacker should collect a large number of samples before statistical inference. However, the measurement matrix is changed after each round of data aggregation in SCDG and thus collecting enough samples of one single measurement matrix becomes impossible. Therefore the sensor network can get rid of statistical inference attacks successfully.

Another important feature of SCDG is that the whole security mechanism depends mainly on the one-way hash function and the key chain. Without them, it is impossible for the attacker to reconstruct the data all the time even though the attacker gets the measurement matrix by some external help occasionally, because after each time interval, P_i changes the secret seed chosen from the one-way hash key chain once it receives the changing-key notification from the sink. Since each sensor just stores the function H_i, which is not shared with others, the attacker can never get all the one-way hash key chains and functions of all sensor nodes. On the other hand, with the guarantee of the one-way property, the attacker can't recover the former coefficient vectors even if he gets the one-way hash function. From this analysis, we claim that SCDG can achieve secure compressive data gathering effectively.

In the following, we make a comparison between SCDG and the original compressive data gathering scheme when the two statistical inference attacks proposed in this paper exist. We use the same network settings and data generation scheme as in Section V-C. All the results are averaged over 20 runs.

Fig. 10 reports the comparison results in terms of the standard deviations of the recovered signals between SCDG and the original compressive data gathering scheme. We can see that the new scheme recovers the signal with a little bit higher standard deviation but still keeps the same magnitude.

We also apply SNR [42] to represent the reconstruction

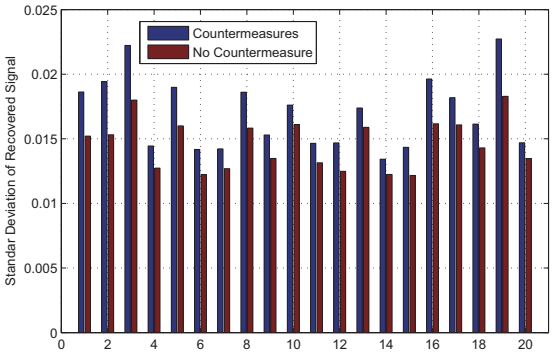


Fig. 10. Comparison of the standard deviation.

effect:

$$SNR = 10log_{10}\frac{\sum_{j=1}^{N} x_j^2}{\sum_{j=1}^{N}(x_j - \hat{x_j})^2} \tag{29}$$

where $\hat{x_i}$ is the reconstructed data at node P_j. From Fig. 10, we observe that the SNR of the signal recovered through SCDG is also just a little bit smaller than that of the original scheme. In fact, this slight difference is reasonable. In the original approach, we can choose an optimized measure matrix for better reconstruction because we keep use the same measurement matrix. In SCDG, the measurement matrix changes all the time and it is unavoidable to get a bad measurement matrix at certain rounds. As mentioned in Section III, the measurement matrix can affect the recovery precision.

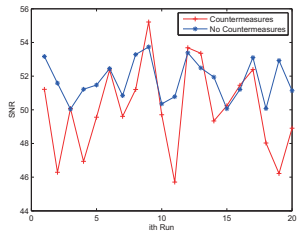


Fig. 11. Comparison of SNR.

VII. Conclusion

In this paper we have identified two statistical inference attacks against compressive data gathering and show that traditional approaches may suffer serious information leakage under these attacks. In particular, we quantitatively analyze the estimation error of compressive data gathering through extensive statistical analysis, based on which we propose a new compressive data aggregation scheme by adaptively changing the measurement coefficients at each sensor and correspondingly at the sink without the need of time synchronization. In our analysis, we show that the proposed scheme

could significantly improve the data confidentiality at a light computational and communication overhead.

ACKNOWLEDGMENT

This work is supported by US NSF grants CNS-0963957 and CNS-1017662, and China NSFC grants 61332004,61170267,61003218,61272444, Jiangsu NSFC BK2011358, RFDP 20113402120008.

REFERENCES

[1] K. Xing, X. Cheng, F. Liu, and S. Rotenstreich, "Location-centric storage for safety warning based on roadway sensor networks," *Journal of Parallel and Dist. Comp.*, vol. 67, no. 3, pp. 336–345, 2007.

[2] F. Liu, X. Cheng, and D. Chen, "Insider attacker detection in wireless sensor networks," in *IEEE INFOCOM*, 2007.

[3] W. Wu, X. Cheng, M. Ding, K. Xing, F. Liu, and P. Deng, "Localized outlying and boundary data detection in sensor networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, no. 8, pp. 1145–1157, August 2007.

[4] M. Ding and X. Cheng, "Robust event boundary detection in sensor networks - a mixture model based approach," in *IEEE INFOCOM 2009 Mini-Conference*, 2009, pp. 2991–2995.

[5] B. Zhang, X. Cheng, N. Zhang, Y. Cui, Y. Li, and Q. Liang, "Sparse target counting and localization in sensor networks based on compressive sensing," in *IEEE INFOCOM*, 2011, pp. 2255–2263.

[6] D. Wu, D. Chen, K. Xing, and X. Cheng, "A statistical approach for target counting in sensor-based surveillance systems," in *IEEE INFOCOM*, 2012, pp. 226–234.

[7] D. Wu, X. Cheng, D. Chen, W. Cheng, B. Chen, and W. Zhao, "A monte carlo method for mobile target counting," in *IEEE ICDCS*, 2011, pp. 750–759.

[8] M. Ding and X. Cheng, "Fault-tolerant target tracking in sensor networks," in *ACM Mobihoc'09*, 2009, pp. 125–134.

[9] S. Ozdemir and Y. Xiao, "Secure data aggregation in wireless sensor networks: A comprehensive overview," *Computer Networks*, vol. 53, no. 12, pp. 2022–2037, 2009.

[10] H. Chan and A. Perrig, "Security and privacy in sensor networks," *Computer*, vol. 36, no. 10, pp. 103–105, 2003.

[11] Y. Yang, X. Wang, S. Zhu, and G. Cao, "Sdap: A secure hop-by-hop data aggregation protocol for sensor networks," *ACM Transactions on Information and System Security (TISSEC)*, vol. 11, no. 4, p. 18, 2008.

[12] Y. Sang, H. Shen, Y. Inoguchi, Y. Tan, and N. Xiong, "Secure data aggregation in wireless sensor networks: A survey," in *Parallel and Distributed Computing, Applications and Technologies, 2006. PDCAT'06. Seventh International Conference on*. IEEE, 2006, pp. 315–320.

[13] J. Albath and S. Madria, "Secure hierarchical data aggregation in wireless sensor networks," in *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*. IEEE, 2009, pp. 1–6.

[14] H. Chan, A. Perrig, and D. Song, "Secure hierarchical in-network aggregation in sensor networks," in *Proceedings of the 13th ACM conference on Computer and communications security*. ACM, 2006, pp. 278–287.

[15] Y. Tang, B. Zhang, T. Jing, D. Wu, and X. Cheng, "Robust compressive data gathering in wireless sensor networks," *IEEE Transactions on Wireless Communications*, 2013, accepted.

[16] J. Haupt, W. U. Bajwa, M. Rabbat, and R. Nowak, "Compressed sensing for networked data," *Signal Processing Magazine, IEEE*, vol. 25, no. 2, pp. 92–101, 2008.

[17] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak, "Compressive wireless sensing," in *Proceedings of the 5th international conference on Information processing in sensor networks*. ACM, 2006, pp. 134–142.

[18] J. Luo, L. Xiang, and C. Rosenberg, "Does compressed sensing improve the throughput of wireless sensor networks?" in *Communications (ICC), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1–6.

[19] L. Xiang, J. Luo, and A. Vasilakos, "Compressed data aggregation for energy efficient wireless sensor networks," in *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2011 8th Annual IEEE Communications Society Conference on*. IEEE, 2011, pp. 46–54.

[20] J. Chou, D. Petrovic, and K. Ramachandran, "A distributed and adaptive signal processing approach to reducing energy consumption in sensor networks," in *IEEE INFOCOM*, vol. 2, 2003, pp. 1054–1062.

[21] G. Hua and C. W. Chen, "Correlated data gathering in wireless sensor networks based on distributed source coding," *International Journal of Sensor Networks*, vol. 4, no. 1, pp. 13–22, 2008.

[22] K. Yuen, B. Liang, and L. Baochun, "A distributed framework for correlated data gathering in sensor networks," *Vehicular Technology, IEEE Transactions on*, vol. 57, no. 1, pp. 578–593, 2008.

[23] Z. Xiong, A. D. Liveris, and S. Cheng, "Distributed source coding for sensor networks," *Signal Processing Magazine, IEEE*, vol. 21, no. 5, pp. 80–94, 2004.

[24] S. Yoon and C. Shahabi, "The clustered aggregation (cag) technique leveraging spatial and temporal correlations in wireless sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 3, no. 1, p. 3, 2007.

[25] H. Gupta, V. Navda, S. Das, and V. Chowdhary, "Efficient gathering of correlated data in sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 4, no. 1, p. 4, 2008.

[26] X. Xu, X.-Y. Li, P.-J. Wan, and S. Tang, "Efficient scheduling for periodic aggregation queries in multihop sensor networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 20, no. 3, pp. 690–698, 2012.

[27] C. Liu, K. Wu, and J. Pei, "An energy-efficient data collection framework for wireless sensor networks by exploiting spatiotemporal correlation," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 18, no. 7, pp. 1010–1023, 2007.

[28] A. Ciancio, S. Pattem, A. Ortega, and B. Krishnamachari, "Energy-efficient data representation and routing for wireless sensor networks based on a distributed wavelet compression algorithm," in *Proceedings of the 5th international conference on Information processing in sensor networks*. ACM, 2006, pp. 309–316.

[29] M. Crovella and E. Kolaczyk, "Graph wavelets for spatial traffic analysis," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, vol. 3. IEEE, 2003, pp. 1848–1857.

[30] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *Information Theory, IEEE Transactions on*, vol. 52, no. 2, pp. 489–509, 2006.

[31] D. L. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, 2006.

[32] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *Information Theory, IEEE Transactions on*, vol. 53, no. 12, pp. 4655–4666, 2007.

[33] D. Needell and R. Vershynin, "Uniform uncertainty principle and signal recovery via regularized orthogonal matching pursuit," *Foundations of computational mathematics*, vol. 9, no. 3, pp. 317–334, 2009.

[34] C. Luo, F. Wu, J. Sun, and C. W. Chen, "Compressive data gathering for large-scale wireless sensor networks," in *Proceedings of the 15th annual international conference on Mobile computing and networking*. ACM, 2009, pp. 145–156.

[35] B. Przydatek, D. Song, and A. Perrig, "Sia: secure information aggregation in sensor networks," in *SenSys'03*, ser. SenSys '03, 2003, pp. 255–265.

[36] W. Du, J. Deng, Y. S. Han, and P. K. Varshney, "A witness-based approach for data fusion assurance in wireless sensor networks," in *Global Telecommunications Conference, 2003. GLOBECOM'03. IEEE*, vol. 3. IEEE, 2003, pp. 1435–1439.

[37] W. He, X. Liu, H. Nguyen, and K. Nahrstedt, "A cluster-based protocol to enforce integrity and preserve privacy in data aggregation," in *Distributed Computing Systems Workshops, 2009. ICDCS Workshops' 09. 29th IEEE International Conference on*. IEEE, 2009, pp. 14–19.

[38] B. Chen and H. Yu, "Secure aggregation with malicious node revocation in sensor networks," in *Distributed Computing Systems (ICDCS), 2011 31st International Conference on*. IEEE, 2011, pp. 581–592.

[39] M. R. Mayiami, B. Seyfe, and H. G. Bafghi, "Perfect secrecy using compressed sensing," *CoRR*, 2010.

[40] S. Agrawal and S. Vishwanath, "Secrecy using compressive sensing," in *Proceedings of Information Theory Workshop*, 2011.

[41] H. S. Anderson, "On discovering the compressive sensing matrix from few signal/meas," *Tech Report*, 2012.

[42] J. Wang, S. Tang, B. Yin, and X.-Y. Li, "Data gathering in wireless sensor networks through intelligent compressive sensing," in *INFOCOM, 2012 Proceedings IEEE*. IEEE, 2012, pp. 603–611.