CSI: Compressed Sensing-Based Clone Identification in Sensor Networks

Chia-Mu Yu
National Taiwan University
Academia Sinica, Taiwan
r91045@csie.ntu.edu.tw

Chun-Shien Lu
Institute of Information Science
Academia Sinica, Taiwan
lcs@iis.sinica.edu.tw

Sy-Yen Kuo
Department of Electrical Engineering
National Taiwan University, Taiwan
sykuo@cc.ee.ntu.edu.tw

***Abstract*—The clone detection, aiming to detect the illegal copies with all of the credentials of legitimate sensor nodes, is of great importance for sensor networks because of the substantial impact of clones on network operations like routing, data collection, and key distribution, *etc*. Based on a state-of-the-art signal processing technique, *compressed sensing*, we propose a novel clone detection method, called *CSI*. Not only is the design philosophy fundamentally different from the existing methods, but also it has the lowest communication cost among all detection methods. The performance and security of CSI will be demonstrated by numerical simulations and analyses.**

***Keywords*-Clone; Node replication attack; Sensor networks; Network security**

I. Introduction

Sensor nodes, which are the primary components of sensor networks, are usually resource-constrained and are lack of tamper-resistant hardware. Nevertheless, as deploying sensor networks for monitoring purpose in the hostile but critical environments where the adversary plays sensor compromises as the major means to subvert the network functionality is possible, equipping the capability of counteracting sensor compromises to the sensor network is necessary. In particular, this paper considers the *node replication attack*, by which the adversary can compromise one innocent sensor node, fabricate many clones having the same identity (ID) from the compromised node, and place these clones back into strategic positions in the network. Since all of the credentials are from the compromised node and this means that they are all legitimate, the clones are considered as legitimate members of the network, making detection difficult. From the security point of view, the node replication attack has its severe impact on networks, because clones can be harmful for network operations like routing, data collection, and key distribution, *etc*, and at the same time can easily help launch insider attacks.

Many research efforts have been conducted to develop countermeasures to node replication attacks. Almost all of them rely on a so-called *witness-finding* strategy, in which one or a few of the other nodes, which play as the *witnesses*, are responsible for identifying the clones. Unfortunately, the communication cost of these methods will be at least $O(n\sqrt{n})$, where n is the total number of sensor nodes. This results from three simple facts that each node needs to find its witness somewhere in the network, there are totally n nodes, and the average hop distance between two arbitrary nodes in a flat network with uniformly distributed nodes is approximately $0.521\sqrt{n}$. It is seemingly impossible to break this lower bound without incurring unrealistic assumptions. Nevertheless, by exploiting the theory of compressed sensing, which is a state-of-the-art technique for signal processing, our proposed CSI scheme can identify the clones with only $O(n \log n)$ communication cost.

A. Related Work

The underlying assumption behind the witness-finding strategy is that a sensor node has to broadcast a signed *location claim* to its neighbors with the purpose of either joining the network or claiming the existence of itself. Although the implementations are different in different proposals, the general procedure of applying witness-finding to detect the clones can be summarized as follows. After collecting the signed location claims $\langle v, L(v), sig(L(v))\rangle$ for each one hop neighbor v, where $L(v)$ and $sig(\cdot)$ denote the location of v and the digital signature function [9], respectively, node u sends the collected signed location claims to a subset W of nodes, where W serves as the set of witnesses. If clones exist in the network, some of the sensor nodes will receive the location claims, among which two with distinct locations have the same ID. This ID must be clones' ID. These nodes are called *critical witness nodes* and they can report or broadcast clones' ID. Afterward, the detected clones can be excluded using, for example, network-wide revocation.

A straightforward realization of witness-finding is Broadcasting [10], in which each node floods its collected location claims. In this case, each node in the network acts as a witness. Broadcasting incurs tremendous $O(n^2)$ communication cost and is, thus, undesirable. An alternative method, called DM [10], is that the witnesses can be chosen deterministically for each node via a publicly known hash function; *i.e.*, a subset W of nodes associated with each node has been set in advance such that the critical witness nodes can be easily found. Nevertheless, DM has been proven insecure [10]. Due to the security weakness of DM, two algorithms, RM and LSM, were also proposed in [10] to determine the witnesses in a random fashion. The difference between RM and LSM is that W in RM is constructed by selecting a

978-1-4244-9528-3/12/$26.00 ©2012 IEEE

random subset of nodes while W in LSM consists of not only the nodes in W in RM but also all of the intermediate nodes on the paths to the nodes in RM. It has been proven that by the optimized parameter setting, RM is the same as Broadcasting asymptotically while LSM is superior to RM in that it incurs only $O(n\sqrt{n})$ communication overhead.

SDC and P-MPC [13] can be, respectively, treated as the variants of RM and LSM with the consideration of cells and with the purpose of guaranteeing the existence of critical witness nodes. In particular, before sensor deployment, the sensing region is divided into cells. Compared to RM and LSM, which forward location claims node by node, SDC and P-MPC forward location claims cell by cell. Here the term "cell by cell" means that when entering a cell, the location claim is flooded within the cell. Based on the assumption that the special centralized broadcasting devices, such as satellites and Unmanned Aerial Vehicles (UAVs), help broadcast a pseudorandom seed to all of the sensor nodes periodically, the W in RED [2] is determined with a publicly known hash function and the seed distributed by satellites or UAVs. RED has extremely low communication overhead. Unfortunately, the use of satellites or UAVs in networks is not practical. Based on the double ruling [11] and Bloom filter [6], a suite of memory-efficient detection algorithms is introduced in [15]. The idea is to guarantee the intersection of traces in LSM via double ruling and to reduce the memory usage of intermediate nodes in LSM via the Bloom filter. The performance of the algorithms in [15] is asymptotically the same as LSM as well. In addition, the random walk technique is utilized in LSM in [14] to improve the fairness of the selection of critical witness nodes at the expense of communication overhead increased to $O(n\sqrt{n}\log n)$.

There are a few algorithms that do not adopt the witness-finding strategy. For example, the algorithm proposed in [12] detects clones by examining each node's social fingerprint, constructed according to the fact that the set of neighbors of each node should be fixed. The algorithm works only when a period of secure time, during which the adversary cannot attack the network, is available. Additionally, if certain type of key pre-distribution protocols [3] is used, the increased usage of certain keys in key pre-distribution protocols can also be utilized to identify the clones [1]. In SET [5], the non-emptiness of two exclusive subsets of sensor nodes' IDs also indicate that clones are involved in the networks. SET also needs $O(n\sqrt{n})$ communication overhead. All of the above works have communication cost at least $O(n\sqrt{n})$ in the absence of unrealistic assumptions. Thus, developing a detection method with communication cost lower than $O(n\sqrt{n})$ could be of both theoretical and practical interests.

B. Contribution

We propose a Compressed Sensing-based clone Identification (CSI) scheme for static sensor networks, which possesses the following advantages:

1) CSI has the lowest $O(n\log n)$ communication requirement while the other methods need $\Omega(n\sqrt{n})$ communication burden.
2) To the best of our knowledge, for the first time, the theory of compressed sensing (CS) is applied to sensor network security issues despite an extensive amount of researches that apply CS to (multimedia) signal processing and data collection in sensor networks.
3) Numerical simulations and analyses are conducted to confirm our performance and security results.

II. System Model

A. Network Model

There are n sensor nodes with IDs $\{1, 2, \ldots, n\}$ and a Base Station (BS) in the network. The usual operation pattern is that sensor nodes sense the environmental data and then forward the data to the BS for further analysis. The communication channel is assumed to be noiseless and symmetric.

To execute its monitoring mission and at the same time save the energy expenditure, an aggregation tree rooted at the BS is constructed among the sensor nodes [8]. Specifically, the sensed data are forwarded to the BS along the tree with the proper aggregation functions involved if necessary. The aggregation function could be value concatenation or value summation, *etc*. The former function behaves like each node sends its raw data to the BS separatively while the BS in the latter function obtains the summation of all of the sensed data[1]. Depending on the environmental factors, battery power of nodes, or any other possible issues, the aggregation tree needs to be recalculated and will possibly not be the same as the previous ones. After the tree is constructed, it is usual to assume that each node knows which node acts as its parent node and which nodes act as its children nodes because of the formation of tree construction.

We also assume that sensor nodes have been scheduled properly such that the data can be aggregated along the tree level by level.

B. Security Model

The adversary can compromise the legitimate nodes and then fabricate the compromised nodes. The node replication attack is to place the clones, the nodes with IDs the same as the compromised node and under the control of the adversary, in the network. To ease the presentation, we simply assume that there are only two nodes with the same ID; *i.e.*, these two nodes are clones. Throughout the paper, *clone ID* refers to these two clones' ID. Our method also applies to the case that multiple *clone groups* exist in the network. Here, a clone group is a group of clones with the same ID and the IDs in different clone groups are different.

[1] Assume that the sensed data are all numeric values.

The clones can arbitrarily choose to follow or not to follow the instructions from the network owner. The clones aim to subvert the network functionality, but they are stealthy; *i.e.*, they try their best to escape the clone detection. As in the literature, the digital signature function $sig(\cdot)$ is available for each node.

III. Proposed Methods

As compressed sensing plays an important role in our proposed CSI method, we first describe some background of compressed sensing and compressed sensing-based data collection in Sec. III-A and Sec. III-B, respectively. After that, the details of our proposed CSI method will be described in Sec. III-C.

A. Background on Compressed Sensing

Assume that x is a s-sparse vector of length n, which means that there are only s nonzero entries in x. The theory of compressed sensing (CS) states that if the measurement vector

$$y = \Phi x, \tag{1}$$

where $y \in \mathbb{R}^m$, $\Phi \in \mathbb{R}^{m \times n}$, $m \ll n$, is the measurement matrix, is obtained in the way that Φ is selected to satisfy restricted isometry property and $m = O(\log n)$ [4], then x can be perfectly reconstructed. Note that the restricted isometry property can be fulfilled by simply choosing a Bernoulli or Gaussian distributed matrix. The vector x can be reconstructed by, for example, ℓ_1-minimization as follows:

$$x^* = \underset{y=\Phi x}{\operatorname{argmin}} ||x||_{\ell_1}. \tag{2}$$

One unique characteristic of CS is the capability of simultaneous data acquisition and compressionin that, as shown in above, the length m of a measurement vector is significantly smaller than that of its corresponding original vector. Due to this property, CS has been vastly applied to data collection in sensor networks in order to reduce the amount of bits to be sent.

B. Background on CS-Based Data Collection

As stated in Sec. III-A, CS can be applied to reduce the communication overhead in data collection. In spite of many proposals, we select one of them, CDG [7], as the representative and use it in accordance with the design of our proposed CSI scheme.

Consider $x = [x_1 \cdots x_n]^T$ as a vector of sensory readings; *i.e.*, x_i is the reading provided by node i. Let $\Phi = [\Phi_{-,1} \cdots \Phi_{-,n}] \in R^{m \times n}$, where $\Phi_{-,i}$ denotes the i-th column vector of Φ. Eq. (1) can be rewritten as

$$y = \Phi x = x_1 \Phi_{-,1} + \cdots + \Phi_{-,n}. \tag{3}$$

Assume that the seed used to generate $\Phi_{-,i}$ is stored in node i in advance. Each node u, after receiving the measurement vectors $x_{v'}\Phi_{-,v'}$, $v' \in C(u)$, where $C(u)$ denotes the set of children nodes of u, computes and sends $\sum_{v' \in C(u)} x_{v'}\Phi_{-,v'} + x_u \Phi_{-,u}$ to its parent node. Afterward, BS can recover the vector x. In this case, each node sends a vector of length m, which is $O(\log n)$ as stated in Sec. III-A. Thus, the communication overhead is $O(n \log n)$.

C. CSI

A key insight in designing CSI is that the number of clones in a network is usually very limited. Consider a network in which there are a large number of clones. It is very uneconomical for the adversary to take the control of or subvert the functionality of the network by fabricating a large number of clones with the corresponding considerable fabrication cost. Thus, a rational way for the adversary is to deploy as few as possible clones to execute the stealthy tasks as many as possible. Since the clones are “sparse” in the sense of its number of appearances, the theory of compressed sensing (CS) can, thus, be readily utilized to identify the clones.

The basic idea behind CSI is that each node broadcasts a fixed number α to its one hop neighbors. This fixed number α can be thought of as the sensory reading of each node. Therefore, in the following, the terms “the number α” and “the sensed data” are used exchangeably. Sensor nodes forward and aggregate the received numbers from descendant nodes along the aggregation tree via compressed sensing-based data gathering techniques (*e.g.* CDG in Sec. III-B) that can greatly reduce the communication overhead and at the same time can losslessly compress the numbers reported by nodes. BS, as the root of the aggregation tree, receives the aggregated result and recovers the sensed data of the network. According to the reconstructed result, the node with the sensory reading greater than α is the clone since a nonclone node can only report the number once. We should note that albeit the low communication overhead is attributed to the use of CS in CSI, none of existing methods can simply apply CS to their methods. CS works in identifying clones only because of the design of our CSI scheme, which contribute the novelty of this paper.

In the above basic idea, there are two issues to be solved. First, if clones collude with each other and broadcast α selectively, then CSI may not guarantee BS being able to detect the anomaly in the recovered sensed data. Second, clones, among which one is the descendant of another, can collude with each other such that the information of descendant clone will be erased. By doing so, the adversary also makes BS in CSI unaware of anomaly.

In fact, the first issue can be solved; *i.e.*, u knows its parent node (or children nodes) is a clone by using the following observation. When the case that the node u can hear the signals from its parent node (or children nodes) when the network is reporting data to the BS but u cannot hear the signals from its parent node (or children nodes) when the network is performing clone identification occurs, this means that u's parent node (or children nodes) is a clone. Nevertheless, the latter issue is more difficult to be solved and the corresponding countermeasure will be described below.

Specifically, assume that an aggregation tree has been constructed and a fixed signal α is predetermined by BS. The procedures of CSI are as follows. Assume that a hash function $h(\cdot)$ is stored in all sensor nodes before sensor deployment. Each node i generates its own $\Phi_{-,i}$ by using $h(i)$ as a seed. Such setting enables each node to calculate the column vectors of Φ corresponding to the other nodes. In addition, to ease the presentation, we simply assume that each transmitted message is accompanied with a signature without particular statements. Each non-leaf node u receives

$$\langle y_v, Y_{C(v)}, C(v)\rangle, \tag{4}$$

where y_v is the measurement vector calculated by v, $C(v)$ is the set of children nodes of v, and $Y_{C(v)}$ is the set of measurement vectors calculated by the nodes in $C(v)$, from each of its children node v. In CSI, leaf nodes do not need to send anything to their parent nodes since as we state in Sec. II-A, each node has knowledge of which are its children nodes and can generate the measurement vector of its children nodes by itself. This is also equivalent to leaf nodes sending out $\langle[], \emptyset, \emptyset\rangle$, where $[]$ and $\emptyset$ denote the empty vector and empty set, respectively. Thus, those that are parent nodes of leaf nodes can compute their measurement vectors without waiting for the measurement vectors from children nodes. This also avoids the spoofing from the clones that act as the leaf nodes since there is nothing they can do to disobey the CSI procedures. Note that, as stated above, the clones that act as leaf nodes can choose not to join the tree formation; however, the innocent nodes refuse to communicate with the nodes that does not take part in the tree formation.

After receiving $\langle y_v, Y_{C(v)}, C(v)\rangle$, u computes

$$y_u = \sum_{v' \in C(u)} y_{v'} + \alpha\Phi_{-,u}, \tag{5}$$

where $\Phi_{-,u}$ is the u-th column vector of Φ. Afterward, u, after waiting an acknowledgement from $C(C(u))$ for certain time, broadcasts $\langle y_u, Y_{C(u)}, C(u)\rangle$. The purpose of the acknowledgement waiting is to ensure that each node honestly broadcasts the measurement vectors calculated according to their received ones. Specifically, the procedures of acknowledgement waiting are that under the assumption that v is a children node of u and that symmetric communication channel is used, both u and $C(v)$ can hear the message $\langle y_v, Y_{C(v)}, C(v)\rangle$. Each children node $\tilde{v}$ in $C(v)$ of v checks if its information $\langle y_{\tilde{v}}, Y_{C(\tilde{v})}, C(\tilde{v})\rangle$ is included in $\langle y_v, Y_{C(v)}, C(v)\rangle$ by calculating y_v according to the received $Y_{C(v)}$ and $C(v)$ by itself. It floods a negative acknowledgement to announce the anomaly if the calculated y_v and the received y_v do not match and sends a positive acknowledgement otherwise. The sending of the positive acknowledgement can be replaced by the setting that if $\tilde{v}$ agrees with $\langle y_v, Y_{C(v)}, C(v)\rangle$, then it sends nothing. Such use of acknowledgment can prevent the aforementioned issue that the clones collude with each other such that some clones erase the information about the other clones.

Eventually, BS is supposed to receive the measurement vector $\Phi[\underbrace{\alpha \cdots 2\alpha \cdots \alpha}_{n}]^T$ since two clones are assumed to be in the network. Currently, BS cannot identify the clones if only $\Phi[\alpha \cdots 2\alpha \cdots \alpha]^T$ is available. Nevertheless, we have the prior knowledge that the received measurement vector should be $\Phi[\alpha \cdots \alpha \cdots \alpha]^T$ if no clones are present. Based on exploiting this prior knowledge, BS can perform CS recovery on $\Phi[\alpha \cdots 2\alpha \cdots \alpha]^T - \Phi[\alpha \cdots \alpha \cdots \alpha]^T = \Phi[0 \cdots \alpha \cdots 0]^T$ to attain $[0 \cdots \alpha \cdots 0]^T$ for colne detection. Due to its remarkable sparsity, the clone ID, which is the position of recovered vector $[0 \cdots \alpha \cdots 0]^T$, can be easily identified.

IV. Performance and Security Evaluation

We are mainly concerned with two performance metrics: communication overhead and detection probability. The former measures the number of bits required for the entire network to identify the clones. The latter measures the capability of identifying the clones.

A. Communication Overhead

To detect the clones, the information about nodes' IDs or positions is usually required to report to the other nodes or BS in an uncompressed way because the lossy compression of such information, which disturbs the information about nodes' IDs or positions, compromises the detection capability. Although Bloom filter is utilized in [15] to reduce such burden, the cost remains the same asymptotically. Nevertheless, as a recently emerging signal processing technique, CS offers a great way for simultaneous acquisition and compression of sparse signals and at the same time does not compromise the (reconstructed) information used for clone identification in a manner of significant communication overhead reduction.

Assume that the average number of one-hop neighbors for each node is d, and each entry of measurement vectors is of length 64 bits. In our CSI scheme, each node u needs only to send out $\langle y_u, Y_{C(u)}, C(u)\rangle$, which implies that the communication overhead at a single node is $64((1+d)m + d) = O(\log n)$. Therefore, the communication overhead for a whole WSN is

$$64n((1+d)m + d) = O(n \log n), \tag{6}$$

because of $m = O(\log n)$.

We can observe from Eq. (6) that the communication overhead varies with different d's and n's. As Fig. 1 shows, our

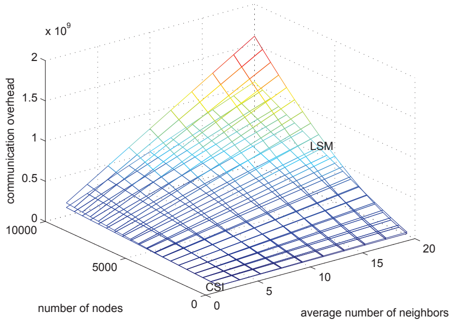

Figure 1: The communication overhead comparison between CSI and LSM.

CSI scheme outperforms LSM because of their difference in communication overhead. Nevertheless, in practice, the gap between the communication costs of LSM and CSI could be larger. This is due to the fact that from vector point of view, two factors, the number of clone groups and the number of clones, have influence on the measurement vectors. With such observation, one may build a database offline, in which different x's with varying number of clones and clone groups are used to calculate the corresponding y's. Specifically, the entries in the database are $\langle x, y\rangle$'s. Once BS receives y' from the network, it searches for the y' in the database. As a result, the corresponding x is the vector that can be used to identify clones this time.

B. Detection Probability

The first thought about the detection probability of CSI is that it is dominated by the probability of perfect recovery in CS. Nevertheless, more precisely, the detection probability of CSI is actually dominated by the probability of identifying the nonzero positions from $\Phi[0 \cdots \alpha \cdots 0]^T$. Note that identifying the nonzero positions is easier than perfect recovery in that once the vector has been perfectly recovered, the nonzero terms are naturally identified but the reverse is not true. As $[0 \cdots \alpha \cdots 0]^T$ is a sparse vector, the number of measurements required for recovering its nonzero positions will be less than that required for the perfect recovery numerically provided that the sparsity is known in advance. For example, as shown in Fig. 2, where only one clone group with two clones exists in the network, even if the network size grows to 5000, the detection probability remains nearly one with approximately 20 measurements. Fig. 3, where k clone groups, each of which contains 3 clones, exist in the network, also shows the detection capability of CSI. As it shows, the number of measurements required for clone identification only slightly increases with the increase of the number of clone groups.

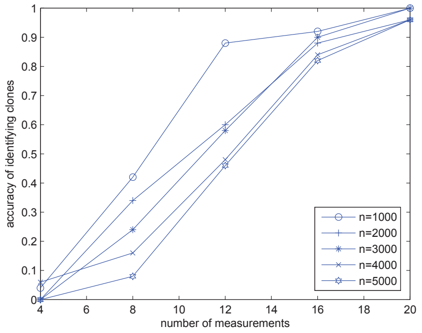

Figure 2: The detection probability with single one clone group ($\alpha = 1$).

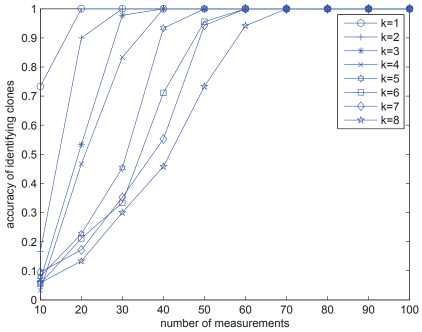

Figure 3: The detection probability with multiple clone groups ($\alpha = 1$).

C. Discussion

From the sparsity point of view, the setting of $\alpha = 1$ offers the sparsity and at the same time the lowest communication overhead. Nevertheless, from the CS recovery point of view, one may argue that larger α could make the CS recovery easier; *i.e.*, the number of measurements needed could be reduced. Fig. 4, whose setting is the same as Fig. 2 except that $\alpha = 1000$ is used, shows different detection probabilities in different scenarios. The comparison between Fig. 2 and Fig. 4 implies that the increase of α does not have substantial influence on the number of measurements needed. Thus, the setting of $\alpha = 1$ achieves the best balance between the detection probability and the communication overhead.

The CS recovery in CSI is performed by BS that is usually assumed to have plenty of computation power. However, since BS usually executes the other tasks such as data analysis and data forwarding, *etc*, the clone identification should increase the computation burden as few as possible. We argue that the computation burden introduced by CSI is quite low as Fig. 5 shows that approximately two seconds suffices to identify clones.

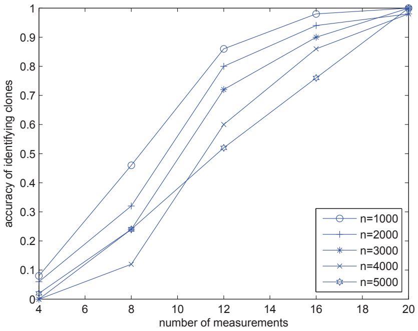

Figure 4: The detection probability with multiple clone groups and larger magnitudes of spikes ($\alpha = 1000$).

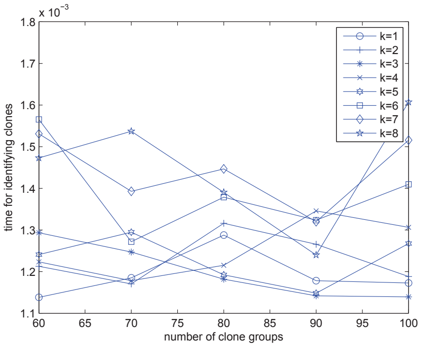

Figure 5: The time required in identifying clones.

V. Conclusion

Based on the sparsity of clone appearance in the network, we propose a Compressed Sensing-based clone Identification (CSI) scheme for static sensor networks. Due to our novel exploitation of the sparsity in this issue, CSI achieves the lowest communication overhead among all methods. The performance and security of CSI have been demonstrated by numerical simulations and analyses.

Acknowledgment

Chia-Mu Yu and Chun-Shien Lu were supported by the National Science Council, Taiwan, under Grants NSC 100-2628-E-001-005-MY2 and NSC 98-2221-E-001-004-MY3. Sy-Yen Kuo is also with School of Electronic and Information Engineering, Beijing Jiaotong University. Sy-Yen Kuo was supported by National Science Council, Taiwan, under Grant NSC 97-2221-E-002-216-MY3, National Natural Science Foundation of China (NSFC) under Grant 60833002, and “111” Project under Grant B08002.

References

[1] R. Brooks, P. Y. Govindaraju, M. Pirretti, N. Vijaykrishnan, and M. T. Kandemir. On the Detection of Clones in Sensor Networks Using Random Key Predistribution. *IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1246-1258, Nov. 2007.

[2] M. Conti, R. D. Pietro, L. V. Mancini, and A. Mei. A Randomized, Efficient, and Distributed Protocol for the Detection of Node Replication Attacks in Wireless Sensor Networks. *ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2007.

[3] H. Chen, A. Perrig, D. Song. Random Key Predistribution Schemes for Sensor Networks. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P)*, 2003.

[4] E. J. Candès, J. K. Romberg, and T. Tao. Robust Uncertainty Principles: Exact Signal Reconstruction from Highly Incomplete Frequency Information. *IEEE Transactions on Infomation Theory*, 52(2):489-509, 2006.

[5] H. Choi, S. Zhu, T. F. La Porta. SET: Detecting node clones in Sensor Networks. *International ICST Conference on Security and Privacy in Communication Networks (Securecomm)*, 2007.

[6] S. Lumetta and M. Mitzenmacher. Using the Power of Two Choics to Improve Bloom Filters. *Internet Mathematics*, vol. 4, no. 1, pp. 17-33, 2009.

[7] C. Luo, F. Wu, J. Sun, and C W Chen. Compressive Data Gathering for Large-Scale Wireless Sensor Networks. In *The Annual International Conference on Mobile Computing and Networking (MobiCom)*, 2009.

[8] S. Madden, M. Franklin, J. Hellerstein, and W. Hong. TAG: a Tiny AGgregation Service for Ad-Hoc Sensor Networks. In *Operating Systems Design and Implementation (OSDI)*, 2002.

[9] D. J. Malan, M. Welsh, and M. D. Smith. Implementing Public-Key Infrastructure for Sensor Networks. *ACM Transactions on Sensor Network*, vol. 4, no. 4, pp. 1-23, 2008.

[10] B. Parno, A. Perrig, and V. Gligor. Distributed Detection of Node Replication Attacks in Sensor Networks. *IEEE Symposium on Security and Privacy (S&P)*, 2005.

[11] R. Sarkar, X. Zhu, and J. Gao. Double Rulings for Information Brokerage in Sensor Networks. *IEEE/ACM Transactions on Networking (TON)*, vol. 17, no. 6, 2009.

[12] K. Xing, F. Liu, X. Cheng, and D. Du. Real Time Detection of Clone Attack in Wireless Sensor Networks, In *IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2008.

[13] B. Zhu, S. Setia, S. Jajodia, S. Roy, and L. Wang. Localized Multicast: Efficient and Distributed Replica Detection in Large-Scale Sensor Networks. *IEEE Transactions on Mobile Computing*, vol. 9, no. 7, 2010.

[14] Y. Zeng, J. Cao, S. Zhang, S. Guo, and L. Xie. Random-Walk Based Approach to Detect Clone Attacks in Wireless Sensor Networks. *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 5, 2010.

[15] M. Zhang, V. Khanapure, S. Chen, X. Xiao. Memory Efficient Protocols for Detecting Node Replication Attacks in Wireless Sensor Networks. *IEEE International Conference on Network Protocols (ICNP)*, 2009.