A Group-based Network Architecture to Defend Node Capture Attacks in Wireless Sensor Networks

Md. Abdul Hamid, Md. Mustafizur Rahman and Choong Seon Hong

Department of Computer Engineering, Kyung Hee University
1, Seocheon, Giheung, Yongin, Gyeonggi, Korea, 449-701

hamid@networking.khu.ac.kr, mustafiz@networking.khu.ac.kr and cshong@khu.ac.kr

Abstract

This paper presents a group-based security scheme for distributed wireless sensor networks. The scheme is devised for dealing with sensory data aggregated by groups of collocated sensors. We model the network for secure routing with 3-tier sensor network composing three types of nodes: Base station, group dominator and ordinary sensor nodes. It is shown that more than 85% network connectivity can be achieved with the proposed model. Key pre-distribution is performed with the concept of star key graph where one superior sensor node dominates other ordinary nodes within the group. We design a group-based network that combines the star graph to form a secure weakly connected dominating set where group dominators form the backbone of the entire network. Analysis and simulation results on network robustness against node capture attacks are presented. The scheme is shown to be simple and scalable and it offers a stronger defense against node capture attacks.

1. Introduction

At present wireless sensor networks (WSNs) are fueling the interest in their wide variety of applications such as forest fire detection, battlefield surveillance, disaster management, homeland security operations, border control, and infrastructure protection and so forth [6, 7]. Generally, each sensor node in the network acts as an information source, sensing and reporting data from the environment for a given task. The low-cost sensor nodes forward the relevant data to a querying sink/base station. However, reporting sensing data is often unnecessary as in many cases sensor nodes in an area detect the common phenomena. So there is high redundancy in sensed data. Nevertheless, it is very inefficient for every single sensor to report their data back because every data packet traverses many hops to reach base station and nodes are stringent in resources. Recently many data aggregation protocols [1, 2, 3, 4, 5] have been proposed to eliminate the data redundancy in sensor data of the network, hence reducing the communication cost and energy expenditure in data collection. During a typical data aggregation process, sensor nodes are organized into a tree hierarchy rooted at a BS. The non-leaf nodes act as aggregators, fusing the data collected from their child nodes before forwarding the results towards the BS. In this way, data are processed and fused at each hop on the way to the BS, and communication overhead can be largely reduced.

This work was supported by MIC and University ITRC Project.

Unlike the tree-based topology, group-based deployment may also result in efficient and secure data aggregation. For example, to detect forest fire, sensor nodes may be strategically, randomly, and densely deployed in a forest [7]. Nodes can relay the origin of the fire to the end users before the fire is uncontrollable. However, as sensor nodes are deployed in open and unattended environments, so they suffer from severe security threats. An attacker can obtain the confidential information (e.g., shared keys) from a compromised sensor and use it with malicious code. This compromised node may produce a false fusion data and sends it to the base station causing the final aggregation report to deviate from the true and/or expected measurement.

Public-key cryptography is a possible solution for key establishment in wireless networks. Though recent work [8] demonstrates cases in which public-key cryptography can be implemented on some resource-constrained devices, it is not yet feasible for all multi-hop networks. In the absence of public-key protocols, key establishment must be performed using symmetric (shared) keys which are assigned to nodes prior to network deployment, a solution known as key pre-distribution. Another extreme solution is the assignment of a global key to every node in the network. However, if an adversary is able to obtain the global key, the security of the entire network is compromised.

A promising approach to symmetric key assignment which attempts to balance the trade-offs between complexity, storage and security is the star key graph based [11] key assignment. Through star key assignment, the complexity of public key protocols, the storage overhead

of pairwise key protocols and the easy compromise of global key protocols can be mitigated.

This paper presents a group-based approach to model and manage the sensor network topology and defines the connectivity in terms of small groups and pre-shared secrets. Topology consists of heterogeneous entities: Base Station (BS), Group Dominator (GD), and ordinary Sensor Node (SN). To preserve the efficiency, our scheme performs data aggregation in each logical group and generates one aggregate from each group. After the BS has collected all the group aggregates, it then checks the validity of the carried MACs (Message Authentication Codes) in the reports and discards the forged report. Major contributions that bring into focus of our work are:

- It is shown that more than *85%* network connectivity can be achieved with the proposed deployment model.
- An individual sensor node needs to maintain fewer shared keys than other existing probabilistic key sharing protocols such as [11], [12], [13] and hence, the protocol is simple and light weight in terms of storage overhead for the stringent sensor node resources.
- Results show that the scheme is robust against increasing number of compromised nodes and the robustness capability of the group-based network manifests itself not only in the number of sensor nodes, but also in the number of superior group dominating nodes.

The rest of the paper is organized as follows. In Section 2 we model the distributed sensor network topology and describe our security aware data aggregation scheme in details. Security analysis and simulation results are presented in Section 3. Section 4 presents existing works and comparisons on this area. We conclude our work in Section 5.

2. Secure group-based network architecture

We model the wireless sensor network in this section. Prior to network deployment, initial secret keys are assigned which is termed as key predistribution. Then the sensor nodes are deployed into the area of interest and the secure group-based network formation is described. Post-deployment rekeying is presented at the end of this section.

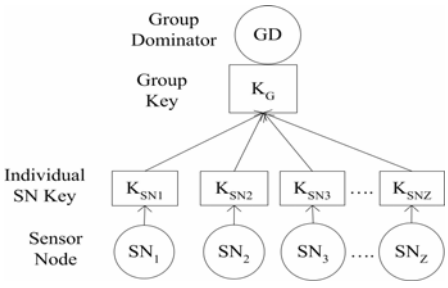


Fig. 1 Pre-distribution of initial secrets using star key graph

2.1 Key Pre-distribution

In the offline key pre-distribution phase, we assign the group keys and individual keys to a group of nodes. For this, the key assignment is accomplished according to Fig. 1. Each GD holds the group key K_G and all the individual keys of SNs, $K_{SN1}, \ldots, K_{SNZ}$, where Z is the number of ordinary sensor nodes in a particular group. Each SN holds group key K_G and its individual key K_{SN} shared with GD. There is also a shared key between BS and each SN and a sheared key between BS and each dominating node (not shown in fig. 1).

2.2 Network Model

We devise a *3*-tier sensor network composing three types of nodes: Base station, group dominator and ordinary sensor nodes. The base station is considered as resourceful enough in terms of residual energy, computation power and speed, and communication. We assume that SN is simple, inexpensive and stringent in resources, while GD is rich in resources and more compromise-tolerant and having transmission range more than $2*R_{SN}$, where R_{SN} is the transmission range of an ordinary sensor node. We also assume that one GD can communicate with its neighbor GD to forward aggregated messages towards the base station. The connectivity (and routing) from the sensor to the base station is dependent on the pre-distributed secret keys. BS stores all the keys of SNs and GDs. We assume that all sensor nodes and GDs are static after they are deployed in the deployment field (e.g., disaster management, forest fire detection). The final locations of the nodes differ from each other after the deployment. However, we assume that the final locations of a GD and its children sensor nodes follow the same probability distribution function. The detailed deployment model is described below.

The GDs and sensor nodes that are to be deployed are divided into n groups $\{G \mid i = 1,2,\ldots,n\}$ according to the group and sensor-GD pairwise keys. A Single GD and Z sensor nodes form a group. We assume that the groups are evenly and independently deployed on the targeted field. The nodes in the same group G_i are deployed from the same place at the same time with the deployment index i. During the deployment, the resident point of any node in group G_i follows a probability distribution function $f(x, y)$, which we call the deployment distribution of group G_i. The actual deployment distribution is affected by many factors. For simplicity, we model the deployment distribution as a Gaussian distribution or Normal distribution since it is widely studied and proved to be useful in practice [10].

Let us assume that V_{BS}, V_{GD} and V_{SN} are the sets of BS, GD and SN nodes respectively and $G(V, E)$ represents entire network in the graph model where, $V = \{V_{BS} \cup V_{GD} \cup V_{SN}\}$. The network backbone $G_B(V_B, E_B) \subseteq G(V, E)$ is formed by the GDs and base stations, provided that, all GDs present in the backbone have at least one path between each-other; i.e., $\{\forall x \in V_B, \exists y \in V_B \mid d(x, y) \le R_{GD}\}$, and, at least one path exists between the backbone and a Base Station BS; i.e.,

$$\{\exists x \in V_B \exists y \in V_{BS} \mid d(x,y) \le R_{GD}\}$$

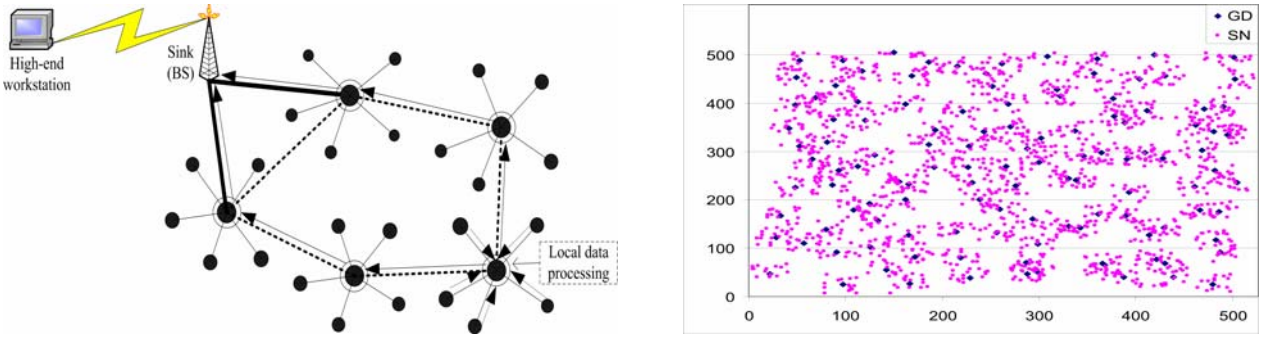
where, $d(x,y)$ is the distance between two nodes x and y, and R_{GD} is the transmission range of GD.

The sensor nodes form GD rooted local STARs (Fig. 2b) by pairwise and group keys. Hence the connectivity between a sensor node and its parent GD depends on two basic conditions:

i) $d(x,y) \le R_{SN}$, where *x* is an SN and *y* is a GD node,

ii) $Groupkey_{GD} = GroupKey_{SN}$.

Fig. 2 is a typical group-based deployment scenario according to our proposed model. In the figure, GD represents the local group dominator and SN represents the constrained sensor nodes. We term this network weakly connected since normal nodes are connected locally to its



(a) (b)

Fig. 2 Typical group-based network deployment scenario. (a) Desired group-based topology. (b) Observed topology from proposed deployment model.

GD and all the group dominators are connected network-wide (i.e. backbone network). The backbone network formed by GDs are connected to the base station and thus, the whole network is connected.

Table 1: Connectivity of group-based deployment

GD	SN	In Group	Out of Group	Out of Network	Connectivity (%)
200	3000	2382	189	429	85.70
300	3000	2371	197	432	85.60
300	4500	3571	286	643	85.70
300	6000	4719	421	860	85.60

Table 1 shows the observed connectivity of proposed group-based deployment model. We take different number of GDs and ordinary SNs with their group key and transmission range. More than *85%* sensor nodes fall within the overall network in which they are connected by their GD or other GD (out of group).

2.3 Secure Network Formation

After the deployment, each SN_i discovers its own GD_j. For this purpose, SN_i multicasts an encrypted SN_JOIN_REQ (using individual key K_{SNi}) message to all of the nodes within its transmission range. If the corresponding GD_j is within its transmission range (i.e., one hop distance), it gets the message and decrypts it as all the individual keys of the sub-ordinate nodes are known to the GD_j. Upon successful decryption of the message, the GD sends a JOIN_APPROVAL message to the SN_i encrypting it with the group key (Shared among SN_i in GD_j). Thus the SN_i becomes a dominated node of this corresponding GD_j. If, for any SN_i, the GD_j assigned during pre-distribution of keys, is not within one hop distance; the SN_i needs to inform the BS for resolving the issue. On discovering itself as out of its own GD, the SN_i sends a GD_ERROR message encrypting it with its individual key. This message is simply forwarded by other sensors and GDs in the network to reach to the BS. For resolving the unexpected issue of this kind, two special cases are considered.

Case I The SN has no dominator as its one-hop neighbor. In such a case, after getting the GD_ERRR message from the SN, the BS issues a command COMM_GD to assign the role of a GD to this SN. For sending the command, the BS uses the individual key of the SN. This newly formed GD does not have any other dominated SN; however, employing this method keeps the isolated node connected with the rest of the network.

Case II The SN does not have its own GD within its one-hop neighborhood but another GD is available in the vicinity. Failing to find out its own GD, SN sends the encrypted GD_ERROR message to the BS. Now, as the GD of another group is present within its one hop distance, it gets the encrypted GD_ERROR message (only detects the type of message and just notes this incident), informs this message encrypted with its group key to the BS. The BS checks the message and issues a command COMM_AGD to that neighbor GD to be its adopter and also sends the individual key for this SN. The GD in turn uses this key to send its K_G to the SN to welcome it in its own group. Thus, all the stars can form the overall network backbone where the GDs of the logical groups (stars) are the dominating nodes and all other nodes in the network are dominated. Fig.3 depicts the operation of SN, GD and BS while forming the group-based network.

2.4 Local Data Processing and Forwarding

Once the network is logically structured as a star-based weakly connected dominating set, the sensory data can be transmitted securely to the BS. GDs are responsible to aggregate data collected from different sensors. If there are *Z* number of ordinary sensors (SN) in a group, for fidelity

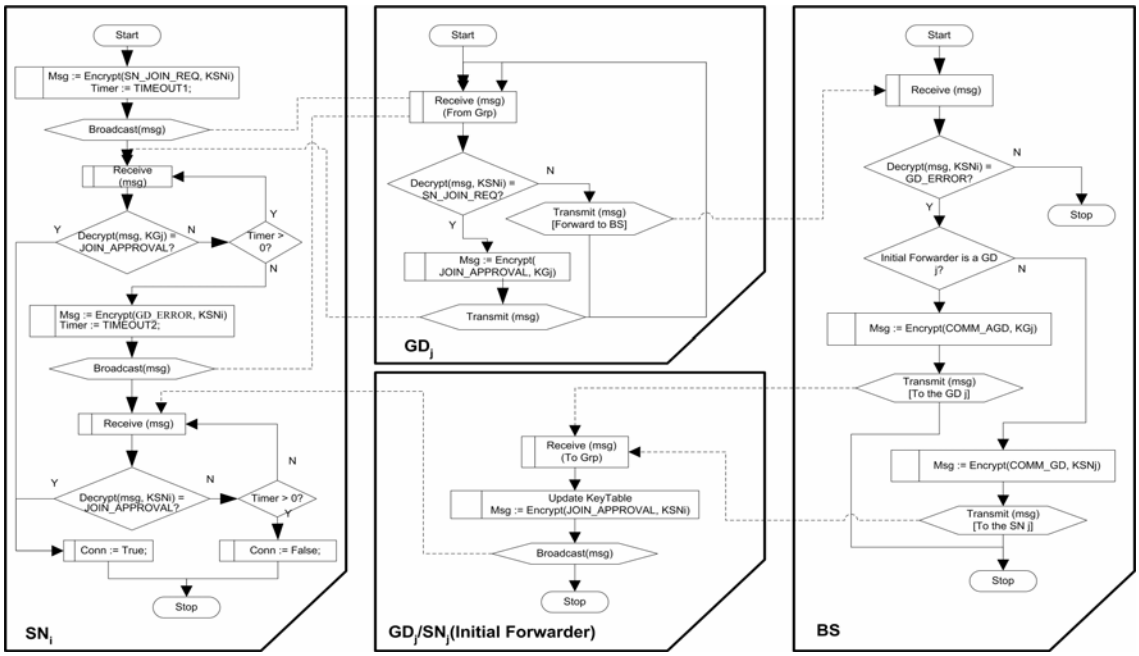


Fig. 3 Activity diagram of ordinary sensor node, group dominator, and base station for group-based network formation.

and correctness of data, the GD waits for the same sensing event from at least q ($q \le Z$) number of the SNs, where q is the threshold value set for a particular group. We consider any one group with ordinary SNs and its corresponding GD. Once an event occurs, q out of Z ($0 \le q \le Z$) ordinary sensors (ID_1, ID_2, . . ., ID_q) within the sensing area detect the event and send information to the GD. The message aggregation is accomplished in the following manner:

$$SN_i \to GD_j : ID_i \,|\, E(M_i \,|\, MAC(M_i, K_{(SN_i, GD_j)}), K_{(SN_i, GD_j)}))$$ where,

i^{th} SN belongs to j^{th} GD, $K_{(SNi,\ GDj)}$ is the shared key between sensor node and corresponding group dominator, E(.) is the encryption, and | denotes bitwise XOR operation, and MAC is the message authentication code. Upon receiving the message sensed by SN_i (ordinary sensor), dominator GD_j verifies every single MAC and generates an aggregated report (and discards the false packet if any).

All SNs in a particular group j also creates a MAC using the shared key between SN and BS and this MAC is only to be verified by the BS but to be relayed by its GD. The message format is

$$SN_i \to GD_j : ID_i \,|\, MAC(K_{(SNi,B)})$$

where, $K_{(SNi,\ B)}$ is the shared key between SN and BS. Now, GD collects all the MACs from ordinary sensor nodes and sends q MACs, q IDs, ID_{GDj}, and M_{GDj} to the sink (BS) directly or via its neighboring GD (multi-hop path through consecutive GDs towards the Sink) as follows.

$$GD_j \to Sink : ID_{GDj}, E(K_{(GDj,B)}, ID_{GDj} \,|\, M_{GDj} \,|\, ID_1 \,|\, MAC_{(SN1,B)} \,|\, ...ID_q \,|\, MAC_{(SNq,B)})$$

The q MACs and an aggregated report M_{GDj} are sent securely to the base station. Upon receiving the aggregation report, the sink first decrypts the message using the corresponding key $K_{(GDj,B)}$, then it checks whether the report carries at least q distinct MACs from ordinary sensors and whether the carried MACs are the same as the MACs it computes via its locally stored keys. If no less than q MACs is correct, the event is accepted to be legitimate; otherwise it is discarded.

2.5 Rekeying

During the offline key pre-distribution, all the nodes are assigned the keys but not all the nodes are deployed. When any of those remaining nodes is deployed, it sends the JOIN_REQ_NEW message using its own individual key. If authorized by the access list of GD, it joins the group. Otherwise, GD forwards this to BS. BS informs GD about the individual key of that SN. If authenticated by BS, GD generates a new group key and encrypts the new group key with the newly added node's individual key and sends it to the SN. All other nodes in the group know about the change by a multicasting by the GD of that group. For leaving a star graph, the node simply leaves a message to inform the GD which in turn generates a new group key and unicasts it within the group members (Fig. 4).

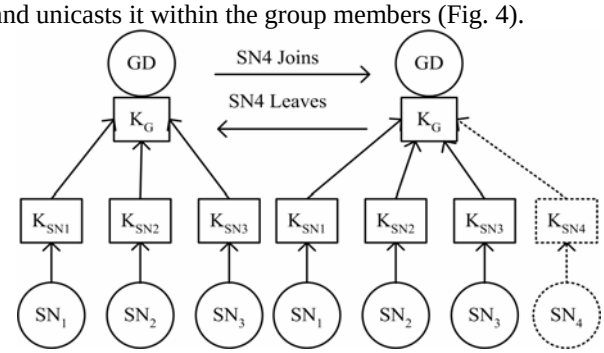


Fig. 4 Rekeying mechanism in a local group (node join/leave)

For example, let's say, SN_4 in Fig.4 wants to join the existing group in the figure shown above. GD changes the group key K_G to a new key K_{Gnew}, and sends the following rekeying messages:

GD Encrypts new group key with the old group key and sends: $GD_j \rightarrow allSN_i : E_{K_G}(K_{Gnew})$,and GD encrypts new group key with the joining SN's individual key and sends: $GD \rightarrow SN_4 : E_{K_{SN4}}(K_{Gnew})$. Similarly, when any SN wants to leave the group, it just sends a leave message: $SN_4 \rightarrow GD : E_{K_{SN4}}(leave)$. Upon receiving the leave message, GD deletes the leaving SN_4 and updates the K_G to a new K_{Gnew}, encrypted with remaining SN's individual key and unicasts the message: $GD \rightarrow SN_{i-1} : E_{K_{SNi-1}}(K_{Gnew})$. In this way, the key management for a new node join/leave can be handled.

3. Security Analysis

We contemplate on analyzing the threat posed by compromising the ordinary sensor nodes and group dominators. An adversary may jeopardize the group-based network in the following ways:

First, an adversary can simply deploy some malicious ordinary sensors into a group. A malicious sensor without having the secret key may try to produce a false sensing result with an identity of a legitimate sensor. Since, a malicious sensor does not have the capability of producing a valid MAC for the false sensing result, the group dominator will identify it. Second, an adversary can copy some ordinary SNs from some group and deploy them into another group. As each GD knows the set of ordinary SNs within its own group, a GD is able to identify an illegitimate ID of ordinary SN. Nevertheless, an ordinary compromised SN can not fool the GD of another group without knowing the valid key. Similarly, a compromised GD from one group does not fool the ordinary SNs of another group without knowing the valid keys. Third, given any group having a GD and Z ordinary SNs, an adversary may launch attack in three ways: (a) compromise some ordinary sensors only, (b) compromise the group dominator only, and (c) compromise the dominator and some sensors concurrently. We term this security threat as intra-group node capture. Fourth, an adversary may launch attack in a distributed manner: (a) randomly compromise ordinary SNs throughout the entire network and/or (b) intelligently compromise few GDs where maximum number of ordinary SNs has already been captured. We term this security threat as network-wide node capture. In the rest of this section, we analyze intra-group and network-wide node capture attacks. We also analyze the effect of deployment density on this kind of attacks posed by an adversary.

3.1 Intra-group node capture

3.1.1 Ordinary sensors nodes are captured

A compromised ordinary sensor in a particular group may produce an invalid MAC by providing wrong guarantee for an aggregated report. The group-based scheme is robust against this kind of attack as long as no more than *q* sensors within a local group are compromised. Since we devise our scheme where each aggregated report carries q number of MACs from ordinary sensors. Only the base station checks the correctness and no less than *q* correct MACs from ordinary sensors is accepted by the base station.

3.1.2 Only the group dominator is captured

When a GD is captured, it may fabricate a report. But to do that, at least *q* MACs need to be forged. The probability that at least *q* out of Z MACs is correct is given by

$$p_{GD} = \sum_{j=q}^{Z} \binom{Z}{q} p^q (1-p)^{Z-q}$$

where, $p = 1/2^L$ and L is the MAC size in bits. It can be seen that this probability p_{GD} is negligible for 4-byte CBC MAC [16]; moreover, only one group out of the entire network is in fact affected while other groups are not.

3.1.3 Group dominator and sensor nodes are captured

We consider the situation where an adversary has compromised a GD and some number $x\ (0 \le x \le q)$ ordinary sensor nodes concurrently. To inject a false report, a GD needs at least *q* valid MACs. Since a GD has to forge $(q - x)$ more MACs, the probability that $(q - x)$ out of $(Z - x)$ is valid, is given by

$$p_{GD}^{x} = \sum_{j=q-x}^{Z-x} \binom{Z-x}{j} p^j (1-p)^{Z-x-j}$$

Again, this probability is almost negligible [16]. If an individual key is compromised, the attacker at best could send false report to the GD but when any message from the GD comes encrypted with the group key, it cannot decrypt it. So, for successful attack, it needs both the individual and group key at the same time. Moreover, compromising one key doesn't affect the rest of the keys used among other nodes and links in the network.

3.2 Network-wide node capture

We analyze the robustness of our scheme when an adversary has randomly compromised $Q\ (0 \le Q \le N)$ ordinary SNs and $j\ (0 \le j \le Y)$ GDs from the entire network. Let p (j, q) be the probability that j^{th} GD (i.e., j^{th} group) having $q\ (0 \le q \le Z)$ SNs compromised, we get

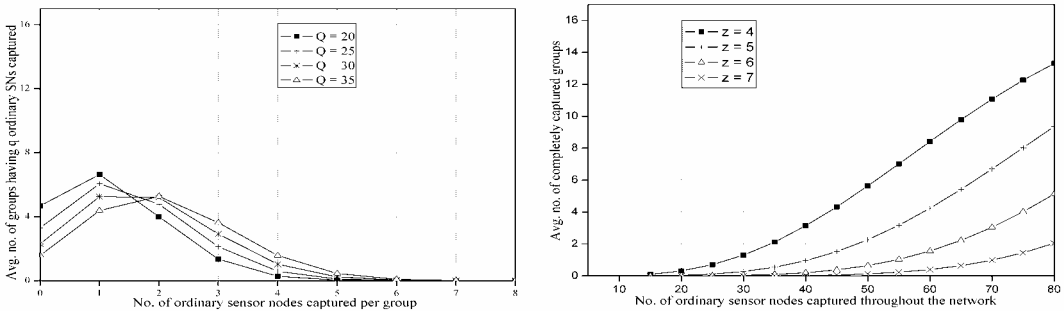
$$p(j,q) = \frac{\binom{Z}{q}\binom{N-Z}{Q-q}}{\binom{N}{Q}}$$

We define $g_{j,q}$ as:

$$g_{j,q} = \begin{cases} 1, \text{if ordinary senor nodes are captured in } j^{th} \text{ group} \\ 0, otherwise \end{cases}$$

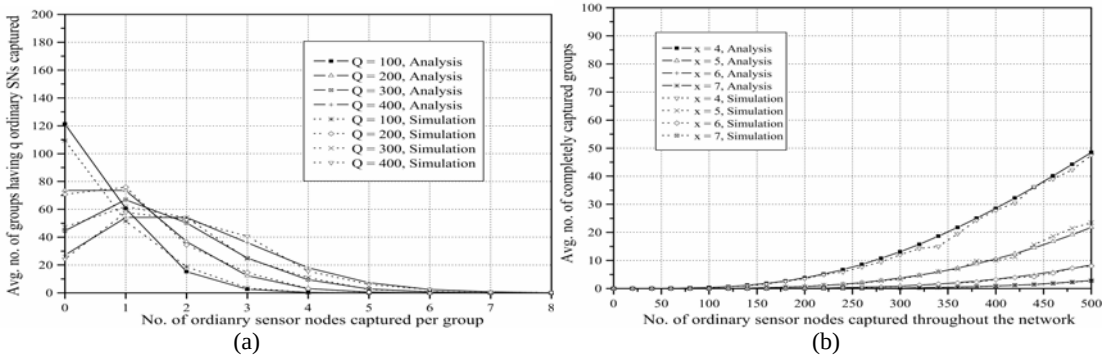
And let G_q denote the number of groups having *q* ordinary SNs captured from the entire network, we get $G_q = \sum_{j=1}^{Y} g_{j,q}$, and, the expected number of groups having *q* ordinary sensors captured can be calculated by the following equation:

$$E\left[\sum_{j=1}^{Y} g_{j,q}\right] = \sum_{j=1}^{Y} E\left[g_{j,q}\right] = Y.E\left[g_{j,q}\right]$$



(a) (b)

Fig. 5 Quantitative analysis: network-wide robustness against node capture attacks (*N = 170, Y = 17, Z = 10*).



(a) (b)

Fig. 6 Poisson analysis and simulation: network-wide robustness (*N = 3000, Y = 200, Z = 15*).

$$= Y \cdot p(j,q) = Y \cdot \frac{\binom{Z}{q}\binom{N-Z}{Q-q}}{\binom{N}{Q}}$$

Next, we assume that an adversary has captured some groups having the z $(z \geq q)$ SNs compromised and we call this situation as a complete group capture. Let X be the number of completely captured groups from the entire network. We can compute $E[X]$ by the following equation:

$$E[X] = \sum_{q=z}^{Z} G_q = \sum_{q=z}^{Z} Y \cdot \frac{\binom{Z}{q}\binom{N-Z}{Q-q}}{\binom{N}{Q}}$$

For demonstration purpose, we take a simple example where total number of SNs is $N=170$, with $Z=10$ SNs in each group (i.e., $Y=17$ groups). Fig.5a shows the expected number of compromised groups against the entire network's compromised ordinary SNs. When *20* SNs are captured, *5* groups having *0* SNs compromised and only *1* group having *3* SNs compromised. Fig.5b demonstrates the number of fully compromised groups that depends on the value z. Four cases are shown when z is *4*, *5*, *6* and *7*.

When z is *4*, *3* (*16.66%* of entire network) groups are fully compromised against *40* (*23.5%*) ordinary compromised SNs. But, as the value of z increases (e.g., *6* or *7*), number of fully compromised groups are much smaller. We observe that robustness can be improved significantly by increasing the value of z.

If the condition $Q \ll N$ holds, then, $C_1, C_2, \ldots, C_Z$ are said to be independent according to the properties of Poisson Process [15] and the number of captured sensors in a particular group follows the Poisson distribution with approximated mean value Q/Y. So, if q sensors are captured in a group, we get the probability as follows:

$$p[q] \approx e^{-Q/Y} \frac{(Q/Y)^q}{q!}$$

Now, if the total number of groups is Y, we calculate the expected value of the number of groups Y having q sensors captured as

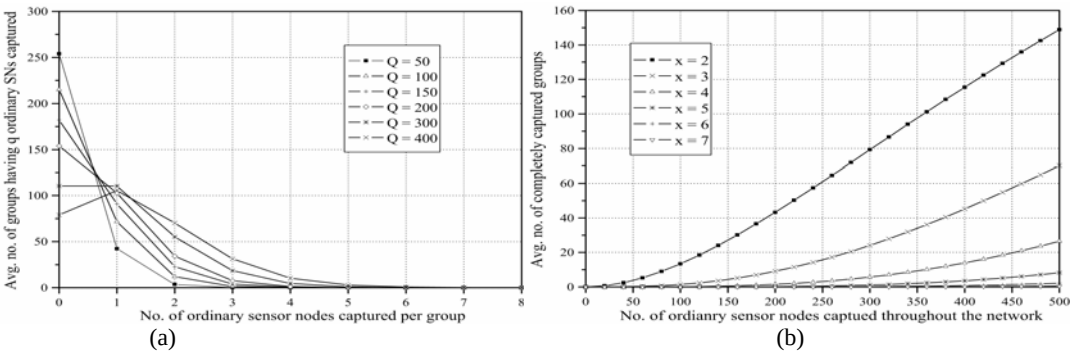
$$E[Y_q] \approx Y.e^{-Q/Y} \frac{(Q/Y)^q}{q!}$$

Next, considering the case where an adversary has captured some groups having the x $(x \geq q)$ SNs captured and we call this situation as complete group capture. Let X be the number of completely captured groups from the entire network. We can compute the expected value $E[X]$ by the following equation:

$$E[X] \approx \sum_{q=x}^{Z} Y_q \approx \sum_{q=x}^{Z} Y.e^{-Q/Y} \frac{(Q/Y)^q}{q!}$$

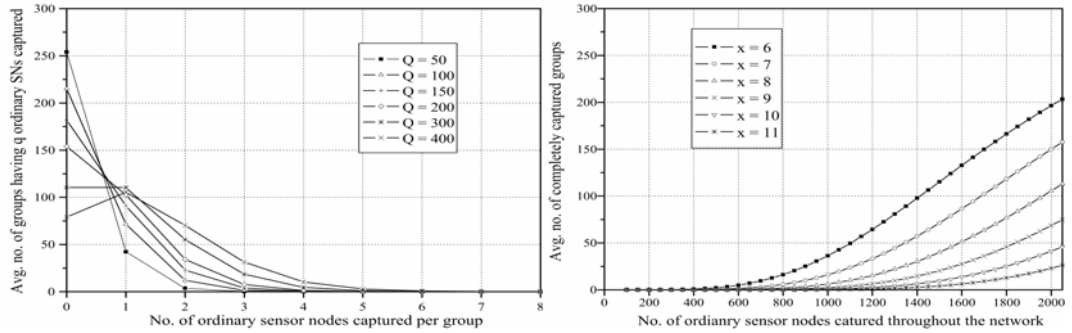
Robustness against node compromise has been plotted in Fig. 6 using analytical result calculated from Poisson Process as well as the results obtained by simulation. Analytical and simulated results are almost similar as can seen in Fig. 6. Fig. 6a shows the average number of groups where q $(0 \leq q \leq Z)$ ordinary SNs are compromised by the attacker in each individual group. *37* groups having *0* and *18* groups having *2* SNs compromised when $Q = 100$ SNs are captured from the entire network. Worst case scenario in our approach, when *400* SNs are capture network-wide, only *18* groups out of *100* having *4* SNs captured.

Fig. 6b shows the average number of groups that are fully captured when more than q $(q \leq x \leq Z)$ sensor nodes are captured.. For the values x equals *4*, *5*, *6* and *7*, we realize that for $x = 4$, *73%* groups are fully captured when *25%* SNs are captured throughout the network. But, as the value of x increases, its robustness gets stronger. For



(a) (b)

Fig. 7 Impact of deployment density (*N = 3000, Y = 300, Z = 10*).



(a) (b)

Fig. 8 Impact of deployment density (*N = 6000, Y = 300, Z = 20*).

example, *x = 7*, *23%* (*23* out of *100*) groups are captured for *25%* (*500* out of *2000*) network-wide captured SNs. So, significant improvement in network robustness can be achieved by increasing the value of *x*.

3.3 Robustness effect on the number of Group Dominators and ordinary Sensor Nodes

Different number of GDS and ordinary SNs are taken to observe the overall performance trend against node capture by the adversary. Fig. 7a plots the distribution of compromised nodes among groups. It can be seen from the figure that when *200* ordinary sensors are captured, *154* groups are having *0* ordinary SNs captured and *103* groups having only *1* SN captured. In case when *400* ordinary SNs are compromised, *80* groups remain with no SNs captured and *106* groups having only *1* SN captured. Fig. 7b plots number of completely captured groups when *x* is set to *2*, *3*, *4*, *5*, *6*, and *7* respectively. It is observed that *26.48* groups are completely captured when *x = 4* and *Q = 500* compromised ordinary SNs. Furthermore, when *x* is *5* or *6*, same amount of compromised ordinary nodes lead to only *8.26* (*2.75%* of all the groups) or *2.19* (*0.73%* of all the groups) groups completely captured.

Fig. 8 plots our analysis when the deployment density is increased in terms of ordinary SNs. Fig. 7a and 8a demonstrates that the effects of deployment density on the distribution of the ordinary SN compromise are the same. So, expected number of groups having *q* ordinary SNs captured in each group is independent on the low and high deployment density. However, the effect on the complete group compromise is significant. It can be seen from Fig. 8b that when *x = 10*, and *Q = 1000* (i.e. *16.66%*) ordinary captured nodes leads to only *0.70* (i.e. *0.23%* of entire groups) groups to be completely captured. Hence, robustness may significantly be enhanced by increasing the number of ordinary SNs as well. It is observed that network robustness is improved by increasing the number of GDs (i.e. dividing deployment area into more groups). This feature can also be seen as the number of ordinary SNs is increased (i.e. by dense deployment). To choose which type of nodes to be increased against security threats is an issue to be resolved and left as an optimization problem of determining the level of security and cost.

4. Related Works

The Eschenauer-Gligor scheme proposed in [11] consists of three phases: key pre-distribution phase, shared-key discovery phase, and path-key establishment phase. In the key pre-distribution phase, a large key pool of *n* keys is generated. Then, *m* keys are randomly selected from the *n* keys (the key pool) and are distributed to each node. Therefore, any two nodes have one common key with a certain probability. For example, if *n = 10000* and *m = 83*, the probability that two nodes have at least one common key is *50 %*. Moreover, with an increase in the size of the key pool (*n*), the number of keys stored by each node (*m*) should be increased to maintain a certain connectivity. For example, if *n = 100000* and the required connectivity is *50%* (i.e., the probability that two nodes have at least one common key is *50 %*), *m* must be greater than *260*. This value is too large for resource-constrained sensor nodes. The Du et al. scheme proposed in [12] is one of the improvements of the Eschenauer-Gligor scheme. In this scheme, two nodes deployed to the same rectangle can share keys with a high probability because their subset key pools are the same. Two nodes deployed to neighboring rectangles share keys with a lower probability because

their subset key pools have fewer common keys. If two nodes are deployed to non-neighboring rectangles, the probability of key sharing is zero. Therefore, the damage due to the compromise of one node is limited to neighboring rectangles, and hence, the network is resilient against node capture. This scheme assumes the group-based deployment model, and the deployment points should be horizontal grid points. For other deployment models, the manner in which the target deployment area should be divided and the subset key pools should be generated are not clearly described and performing these tasks appears to be difficult. Moreover, if the pdf of node deployment is not a two-dimensional normal distribution or if it changes in real time (e.g., the wind direction changes during deployment), this scheme does not appear to work. In Statistical En-Route Filtering [13] scheme, each intermediate forwarding node verifies one MAC and five hash computations (for Bloom filters) probabilistically if it has one of the keys in common. In our scheme, only the GDs forward the aggregated report, but they don't perform this intermediate checking. SEF is constrained by sensor's storage since to increase one hop detection probability, the number of keys a sensor stores should be large. Each report is about *15* bytes long in SEF scheme and communication overhead is (15.h) bytes, where the number of hops, h, a report travels is very high (necessary for better performance).

Our approach, instead, each of the ordinary SNs needs to maintain only *3* keys. Moreover, our solution assigns different level of security loads according to resource capability and is applicable to members that are affiliated subgroups of nodes. Also, our scheme supports both forward and backward secrecy as nodes are added or removed.

5. Conclusions

We have proposed a group-based secure network in which sensed data is processed locally by the dominating node that prepares the authenticated report for the destination. We evaluate our scheme through analysis and simulation to show that group-based architecture is strongly resilient to node capture attacks. Analytically it is shown that robustness depends both on the number of GD and SN (e.g., low and high density deployment). For a complex and probabilistic system, we speak of the "infeasibility" of breaking the security system rather than the "impossibility" of breaking the same system. Some implications may be worth considering by the designers: The proposed model suggests that nodes can be grouped into small clusters or cells where in each cell one can designate a specific node (group dominator in this case) to carry all the burden of relaying multihop packets.

6. References

[1] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: Scalable coordination in sensor networks," in *Proc. of ACM Mobicom,* Seattle, Washington, USA, August 1999, pp. 263–270, ACM.

[2] C. Intanagonwiwat, D. Estrin, R. Govindan, and J. Heidemann, "Impact of network density on data aggregation in wireless sensor networks," in *ICDCS*, 2002, pp. 457–458.

[3] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *MOBICOM*, 2000, pp. 56–67.

[4] B. Krishnamachari, D. Estrin, and S. Wicker, "The impact of data aggregation in wireless sensor networks," in *International Workshop on Distributed Event-Based Systems, (DEBS '02)*, Vienna, Austria, July 2002.

[5] Samuel Madden, Michael J. Franklin, Joseph M. Hellerstein, and Wei Hong, "TAG: A tiny aggregation service for ad-hoc sensor networks," in *OSDI*, 2002.

[6] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, 2002.

[7] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Networks: A Survey," *Computer Networks*, vol. 38, no. 4, March 2002.

[8] N. Gura, A. Patel, A. Wander, H. Eberle, S. C. Shantz, "Comparing elliptic curve cryptography and RSA on 8-bit CPUs," in *Proc. of the Sixth Workshop on Cryptographic Hardware and Embedded Systems (CHES'04)*, Cambridge, MA, USA, 2004, pp. 119–132.

[9] C. K. Wong, M. Gouda and S. S. Lam, "Secure Group Communications Using Key Graphs", *IEEE/ACM Transactions on Networking*, Vol. 8, No. 1. February 2000.

[10] Donggang Liu, Peng Ning, and Wenliang Du., "Group-Based Key Pre-Distribution in Wireless Sensor Networks," in *Proc. ACM WiSE'05*, September 2, 2005.

[11] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proc. of the 9th ACM Conference on Computer and Communications Security*, pp. 41–47, November 2002.

[12] W. Du, J. Deng, Y. S. Han, S. Chen, and P. K. Varshney, "A key management scheme for wireless sensor networks using deployment knowledge," in *Proc. of the IEEE INFOCOM 2004*, pages 586–597, March 2004.

[13] Ye, F., Luo, H., Lu, S, and Zhang, L, "Statistical en-route filtering of injected false data in sensor networks," *IEEE Journal on Selected Areas* in *Communications*, Volume 23, Issue 4, April 2005, pp. 839 – 850.

[14] G.J. Pottie, and W.J. Kaiser, "Wireless integrated network sensors," *Communications of the ACM* 43 (5) (2000) 551–558.

[15] S. M. Ross, *Introduction to Probability Models*, Academic Press, 2003, 8th Edition.

[17] Y.W. Law, J. Doumen, and P. Hartel, "Survey and benchmark of block ciphers for wireless sensor networks," *ACM Transactions on Sensor Networks*, Vol. 2, No. 1, pp. 65–93, February 2006.