

A Key-Predistribution-Based Weakly Connected Dominating Set for Secure Clustering in DSN*

Al-Sakib Khan Pathan and Choong Seon Hong**

Networking Lab, Department of Computer Engineering, Kyung Hee University
Giheung, Yongin, Gyeonggi, 449-701 Korea
spathan@networking.khu.ac.kr, cshong@khu.ac.kr

Abstract. The intent of this paper is to propose an efficient approach of secure clustering in distributed sensor networks. The clusters or groups in the network are formed based on offline rank assignment and key-predistribution. Our approach uses the concept of weakly connected dominating set to reduce the number of cluster heads in the network. The formation of clusters in the network is secured as the secret keys are distributed and used in an efficient way to resist the inclusion of any hostile entity in the clusters. Along with the description of our mechanism, we present an analysis and comparison to justify the efficiency of our approach.

1 Introduction

A Distributed Sensor Network (DSN) is a wireless sensor network with a large number of sensors and large coverage area. It differs from the traditional wireless sensor network in the sense that, it contains considerably huge number of sensors which are intended to be deployed over hostile and hazardous areas where the communications among the sensors could be monitored, the sensors are under constant threat of being captured by the enemy or manipulated by the adversaries. DSN is dynamic in nature in the sense that, new sensors could be added or deleted whenever necessary [1]. DSNs are suitable for covering large areas for monitoring, target tracking, surveillance and moving object detection which are very crucial tasks in many military or public-oriented operations.

In this paper, we propose an efficient key-predistribution scheme which helps for offline rank assignments of the sensors and eventually plays the crucial role to form a network-wide weakly connected dominating set. Later analysis shows that, our approach could perform well to form secure clusters in a distributed sensor network.

This paper is organized as follows: Section 2 outlines the related works, Section 3 presents our model, Section 4 proposes our approach and the method for key-predistribution, Section 5 contains the performance analysis and comparison, and Section 6 concludes the paper.

* This work was supported by MIC and ITRC projects.

** Corresponding author.

2 Related Works

Grouping nodes into clusters is a good idea as it helps to divide the network into several separate but interrelated regions. It also helps for efficient routing within the network. Some of the previous works already have addressed clustering in sensor networks. Here we mention some of the works those dealt with clustering. [2] presents a distributed expectation-maximization (EM) algorithm suitable for clustering and density estimation in sensor networks. Energy-Aware clustering is addressed in [3], [4], [5], [6], [7] etc. In [17] the authors propose a load-balanced clustering scheme which increases the lifetime of the network. Other works on clustering in sensor networks are [8], [9], [10] etc. In fact, most of these works consider a secure environment while forming the clusters in the network which might not always be true for distributed sensor networks. For example, there might be a hidden and active enemy-sensor network operating in the area where the clusters are to be formed and while forming the clusters in the network the hostile-hidden nodes could actively try to participate in the formation process or hinder the formation of clusters in the friendly distributed sensor network. Hence, we focused on secure clustering from the very beginning of the network.

Our work differs from all of the mentioned works as we model our network to form clusters or groups based on offline rank assignments by pre-distribution of keys and using the notion of weakly connected dominating set considering the whole distributed sensor network as a graph.

3 Our Model

We consider the topology of the whole distributed sensor network as a unit-disk graph (UDG) [11], $G = (V, E)$, where V is the set of sensors (vertices) in the network and E is the set of direct communication links (edges) between any two sensors.

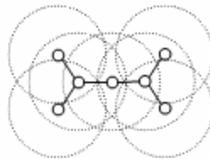


Fig. 1. Unit-disk Graph

Definition 1. A dominating set S is a subset of the vertex set V of a graph $G = (V, E)$ (i.e., $S \subseteq V$), so that all other vertices in the graph are adjacent to the vertices of S . For a dominating set S , $N_G[S] = V$, where $N_G[S]$ is the set of vertices including the vertices in S and the vertices adjacent to a vertex of S (see Figure 2). However, finding a minimum size dominating set in a general graph is NP-complete [12].

Definition 2. A connected dominating set (CDS), S_C is a dominating set of a given graph $G = (V, E)$ where the induced subgraph of S_C is connected. Figure 3(left) shows the connected dominating set for our graph model (i.e., all the black vertices).

The connected dominating set for any type of ad hoc network could be used for efficient routing or message transmission throughout the network. However, for CDS, a large number of dominating nodes is needed to maintain the connectivity requirements of the network.

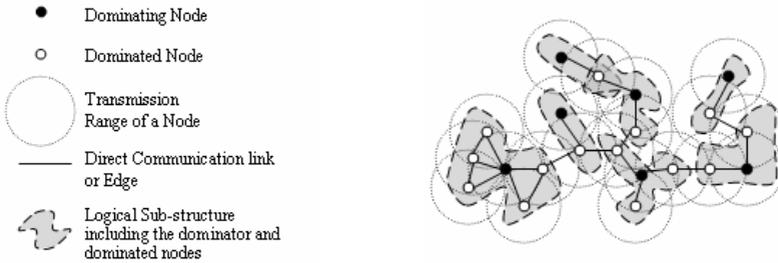


Fig. 2. (left) Legend used throughout the rest of the paper (right) Dominating Set consisting of black vertices

Definition 3. A weakly connected dominating set (WCDS), S_w is a dominating set where the graph induced by the stars of the vertices in S_w is connected. A star of a vertex comprised of the vertex itself and all the vertices adjacent to it (All the black nodes in Figure 3(right)). For any given graph,

$$|WCDS| \leq |CDS| \tag{1}$$

where, $| \cdot |$ denotes the size of the set. So, in case of WCDS, less number of dominating nodes is needed for establishing network-wide connectivity than that is required for CDS. For example, in Figure 3, $|WCDS|=8$ while $|CDS|=13$.

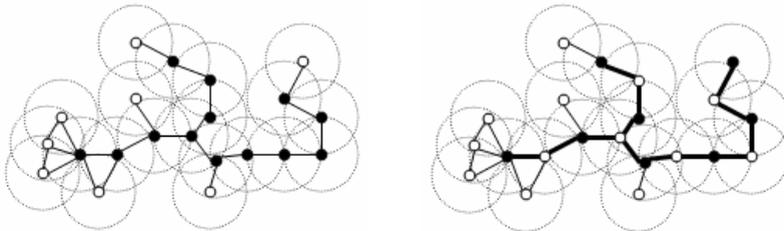


Fig. 3. (left) Connected Dominating Set (right) Weakly Connected Dominating Set

The weakly connected dominating set underpins our proposed scheme. In fact, it is easy to see that each dominating node (or vertex) in the weakly connected dominating set is at the center of a star (or, disk). Thus for each dominating node in a WCDS of the overall network, we have one star where all the other nodes in the star are just one hop apart (Figure 4(left)). Also it could be observed that, between two stars there is at least one common dominated node which could be used for the communication purpose between two separate stars. We term this common dominated node between two individual stars as ‘Mediator’ (Figure 4 (right)).

4 Our Approach

We apply two stage operations for secure formation of clusters in the network.

Assumption 1. Once the sensors are deployed they remain relatively static in their respective positions.

Assumption 2. In a unit disk or transmission range of a sensor, all the neighboring sensors do not necessarily have a direct communication link among themselves. If two nodes i and j have a direct communication link, it is bidirectional; $\forall_{i,j}, (i, j) \in E \Rightarrow (j, i) \in E$ and it exists if and only if i and j have common keys.

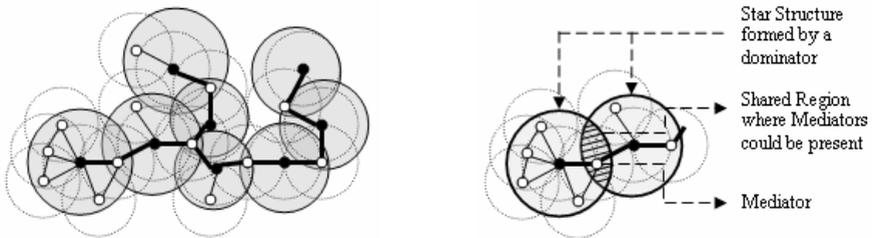


Fig. 4. (left) Dominators' coverage areas in WCDS (right) Mediator between two groups/stars

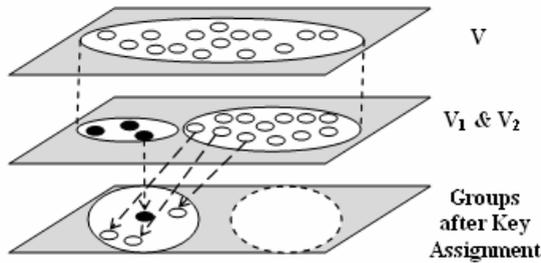


Fig. 5. Ranking of the sensors based on the key pre-distribution

4.1 Offline Rank Assignment

The sensors in the network are assigned their ranks based on the offline key-distribution. We divide the whole set of sensors V into two subsets, V_1 and V_2 , where V_1 contains the probable group dominators (GD or cluster heads) and V_2 consists of ordinary sensors (Os). The set V_2 is further divided into several subsets $w_i \subset V_2, i=1, 2, 3, \dots, N$ and N is the maximum number of possible proper subsets of V_2 . Each w_i is assigned to one element in the set V_1 . The sensors in the subset w_i ($Os_1, Os_2, \dots, Os_{\eta}$) and corresponding one sensor from V_1 (let, $GD_i, i=1$) are taken for group-wise key-predistribution (Figure 5). All the sensors in the set w_i are assigned two keys one of

which is the group key shared by only the particular sensor and the GD_i . The GD_i contains all the individual keys of the sensors in its w_i and its own group key.

Assumption 3. All the sensors have same transmission range. Each node transmits within the transmission range isotropically (in all directions) so that each message sent is a local broadcast.

Assumption 4. The Base Station (BS) contains all the individual keys and group keys of the network.

Assumption 5. The number of Oss (value of η) in each group is decided on demand. It could be group specific or set to a common value for all the groups. η is actually the maximum degree ($\Delta(GD_i)$) of a GD in a group.

4.2 Secure Cluster Formation

The groups of sensors are deployed over the target region one group at a time. After deployment, each O_s tries to find out its own GD by sending a join request packet encrypted with its individual key. The corresponding GD in turn sends the join approval message encrypted with the group key. In both cases, both the GD and the O_s can decrypt the messages and form the group. In some cases, the corresponding GD of an O_s might not be within one-hop transmission range (disk). In this case, the O_s detects the presence of other GDs of other groups in its surroundings, collects their ids and sends an error message to the base station (BS) with this information. The GDs within its one-hop transmission range also detects such erroneous O_s and reports to the BS. The BS in turn assigns one of the neighboring GD as the adopter of the orphan O_s . In the worst case, the O_s might not find any GD in its surroundings. In this case, The BS assigns the rank of a GD to that particular O_s though it does not contain any other sub-ordinate sensors. An O_s which gets its own GD and another GD of another group in its transmission range is the mediator in this case. As stated earlier, all the stars thus shaped could use mediators for the inter-group (inter-star or inter-cluster) communication (see Figure 4(right)). In this way, eventually the resultant logical model of the whole network contains a weakly connected dominating set where the GDs of the logical groups (stars) are the dominating nodes and all other nodes in the network are dominated. This logical model now could be used for secure message delivery within the network (using the secret keys).The pseudo code for secure cluster (group) formation algorithm is presented in Figure 6.

All the groups of sensors could be deployed at a time or more groups could be deployed later based on demand. If it is needed, some sensors in a group could be deployed later. During the offline key pre-distribution, all the nodes are assigned the keys but all the nodes might not be deployed. When any of those remaining nodes is newly deployed, it follows the procedure of joining a group. If authorized by the access list of GD, it joins the group. Otherwise, GD forwards the id of this sensor to BS. BS informs GD about the individual key of that O_s if it is a legitimate node. 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 that particular O_s . All other nodes in the group know about the change of group key by a local broadcast by the

```

Let,
enci(.) - message encrypted by individual key of i
enciNOT(.) - message encrypted by an unknown individual key
encg(.) - message encrypted by the group key
encgNOT(.) - message encrypted by an unknown group key
Osg - the set of Oss allowed under a group dominator g
locbr(.) - local broadcast within one hop transmission
range

for each s ∈ Vos
    locbr(enci(JOIN_REQ))
    if encg(JOIN_APRV) from any g ∈ Vgd and hop(s,g)=1
        edge(s,g)
        dominator(s) ← g
    else
        flood(enci(GD_ERR)) destined to BS
    end if
    if encgNOT(JOIN_APRV) from any g ∈ Vgd and hop(s,g)=1
        neighbor_dominator(s) ← g
    end if

for each g ∈ Vgd
    if enci(JOIN_REQ) from any s ∈ Vos and s ∈ Osg
        send encg(JOIN_APRV)
        edge(s,g)
        sub-ordinate(g) ← s
    end if
    if enciNOT(JOIN_REQ) from any s ∈ Vos
        mediator(g) ← s
    end if
    if enci(GD_ERR) from any s ∈ Vos and hop(s,g)=1
        report encg(ORP_ERR) to BS
    end if

#In case of the BS:
if enci(GD_ERR) from any s ∈ Vos and encg(ORP_ERR) from any
g ∈ Vgd
    if same id of s, issue command: Adopter_GD(s) ← g

```

Fig. 6. Pseudo Code for Clustering Algorithm

GD of that group. In this case, the previous group key is used for encrypting the new group key. For leaving a group or cluster, the node simply leaves a message to inform the GD which in turn generates a new group key and multicasts it within the group members.

5 Performance Analysis and Comparison

We form a WCDS to cover almost all of the nodes in the network with minimum effort. The offline rank assignment reduces the burden of executing resource-hungry

operations to form clusters like other clustering mechanisms. As shown in equation (1), WCDS requires less number (or equal to) of dominating nodes to cover the whole network than that of a CDS requires. Depending on the requirements we can increase or decrease the value of η (the expected degree of a GD in a group). In ideal case, the size of the dominating set created in our approach could be obtained by,

$$\begin{aligned}
 \text{Size of Dominating Set} &= \frac{\text{Number of vertices in the Graph}}{\eta+1} \\
 &= \frac{\text{Number of vertices in the Graph}}{\Delta(GD)+1}
 \end{aligned}
 \tag{2}$$

In our experiment, we generate random graphs of 20-200 and 40-200 nodes with expected average degree 6 and 12 respectively. To simulate the structure of the sensor network, we place the vertices randomly over a 2-D rectangular plane. The network size and density is set by changing the number of vertices and transmission ranges of the nodes. Applying our approach and two algorithms (I and II) of [13] we find that our approach generates much smaller number of group dominators or cluster heads. For a large number of sensors it works effectively. Figure 7 shows the size of dominating sets in comparison with that of Algorithm I and Algorithm II of [13]. The major advantage of our approach is the flexibility to set the value of η (expected maximum degree of a GD) according to the requirements.

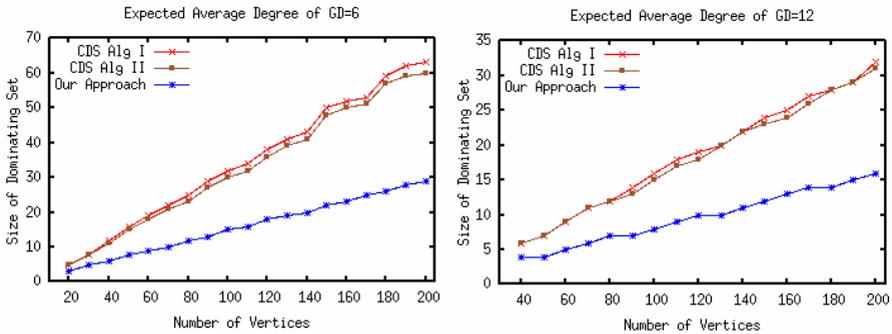


Fig. 7. (left) Size of the dominating set when expected average degree 6 (right) when expected average degree 12

We use the distinct group keys for each of the GDs and distinct individual keys for each Os. So, in general case, the number of distinct keys required for our network is equal to the number of sensors in the whole network.

Each group dominator (GD) in the network has to remember one group key and all the individual keys of the Oss of that particular group. So, the storage requirement for each GD in number of bits is,

$$\gamma_{GD} = (\eta + 1) \times k
 \tag{3}$$

and for each Os,

$$\gamma_{Os} = 2 \times k
 \tag{4}$$

where, η is the number of Oss in that particular group and k is the number of bits required for representing the key. As the value of η increases, the storage load for a GD increases. Hence, the value of η is set according to the requirements or a particular situation at hand. So, if initially we have α number of GDs and β number of Oss, the network wide storage usage for storing the keys is,

$$\begin{aligned} \Gamma_{network-wide} &= \alpha \times ((\eta + 1) \times k) + \beta \times (2 \times k) \\ &= k \times (\alpha \times (\eta + 1) + 2 \times \beta) \end{aligned} \tag{5}$$

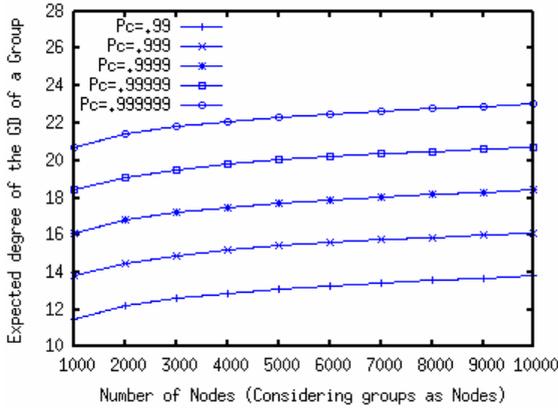


Fig. 8. Given a connectivity probability, expected degree of a GD from the high level view

After formation of clusters within the network, the mediators are used for communication among clusters. From the higher level view, we could consider the clusters (or groups) as nodes in a random graph $G=(n, p)$, where n is the number of nodes (i.e. clusters in our case) for which the probability that an edge (i. e. communication link via mediator) exists between two nodes is p . $p=0$ when there is no edge and $p=1$ when the graph is fully connected. According to Erdős and Rényi [14], for monotone properties, there exists a value of p such that the property moves from “nonexistent” to “certainly true” in a very large random graph. The function defining p is called the threshold function of a property. Given a desired probability P_c for graph connectivity, the threshold function p is defined by,

$$P_c = \lim_{n \rightarrow \infty} P_r[G(n,p) \text{ is connected}] = e^{e^{-p}}, \text{ Where, } p = \frac{\ln(n) - \ln(-\ln(P_c))}{n}$$

Let, p be the probability that an edge (communication link via mediator) exists between two GDs of two clusters, n be the number of nodes (i.e. clusters/groups in the entire network in this case), and d be the expected degree of each GD, then,

$$d = p \times (n-1) = \frac{(n-1)(\ln(n) - \ln(-\ln(P_c)))}{n} \tag{6}$$

In our approach, the sensors could be added later on rather deploying all of them at a time. Sometimes the entire terrain info and deployment diagram could be available (consider a battlefield scenario where the sensors are deployed prior to the enemy forces' invasion). In this case, the extra sensors could be deployed within the range of its appropriate group or cluster. If the sensors are deployed randomly, in the worst case, all the extra or newly added sensors will not be within the range of their intended group dominator and even no other GD could be available in their surroundings. Hence, in the worst case, all the newly added sensors would be included in the dominating set which would increase the size of the dominating set. Still it could be less than the number of dominators needed in case of a connected dominating set (CDS) when the network size is very large. Keeping the size of the dominating set to a minimum is helpful as less number of dominators means less number of entry paths for the false information injected in the network and also it is cost effective if the expected dominators are considered to have relatively higher resources than those of the ordinary sensors.

Our scheme ensures that, each of the GDs and the corresponding Oss could directly form the groups (i. e. clusters) maintaining the security of the network from the bootstrapping state. As encryption is used for message-transmission within the network from the very beginning of the network formation, our scheme could successfully defend Hello Flood Attack [15] and most of other attacks in wireless sensor networks [16]. Again, as each node carries distinct individual and group keys, compromising one node affects only one link in the network while other links remain safe from the attacks by the adversaries. If the group key of a particular group is compromised, still the adversary needs valid individual keys of the Oss for decrypting the information sent from an Os. In case of the compromise of a GD, the base station gets involved for revoking the keys and even in this case, only one group is affected while others could still operate. The re-keying feature ensures robust security as with each addition of a new sensor, the group key is renewed. If considered as resource-exhaustive, the key renewal mechanism could be omitted. However, for military networks as security is the major issue, we could consider a slight increase of the usage of the resources in the sensors.

6 Conclusions and Future Works

This paper presents an efficient approach for secure clustering in distributed sensor networks based on key-predistribution and prior rank assignments. As the group dominators rule over all other sensors in the group for data transmission, the dominators could require more energy, processing and storage power. For this, a set of sensors with greater resources could be considered as dominators. As future works, we would like to deal with secure routing and an efficient method to prevent Denial-of-Service (DoS) and Distributed DoS (DDoS) attacks in distributed sensor networks using our approach. Due to the page limitations, we have shortened the details of some of the parts in this paper.

References

1. Carman, D. W., Kruss, P. S., and Matt, B. J., "Constraints and Approaches for Distributed Sensor Network Security", NAI Labs Technical Report # 00-010, dated 1 September, 2000.
2. Nowak, R. D., "Distributed EM Algorithms for Density Estimation and Clustering in Sensor Networks", IEEE Transactions on Signal Processing, Vol. 51, No. 8, August 2003, pp. 2245-2253.
3. Halgamuge, M.N., Guru, S.M., and Jennings, A., "Energy efficient cluster formation in wireless sensor networks", 10th International Conference on Telecommunications, 2003 (ICT 2003), Volume 2, 23 February-1 March 2003, pp. 1571-1576.
4. Lee, S., Yoo, J., and Chung, T., "Distance-based energy efficient clustering for wireless sensor networks", 29th Annual IEEE International Conference on Local Computer Networks, 2004, 16-18 Nov 2004, pp. 567-568.
5. Younis, O. and Fahmy, S., "Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach", IEEE Transactions on Mobile Computing, 3(4), Oct-Dec 2004, pp. 366-379.
6. Ye, M., Li, C., Chen, G., and Wu, J., "EECS: an energy efficient clustering scheme in wireless sensor networks", 24th IEEE International Performance, Computing, and Communications Conference, (PCCC 2005), 7-9 April 2005, pp. 535-540.
7. Liu, J.-S. and Lin, C.-H.R., "Power-efficiency clustering method with power-limit constraint for sensor networks", Proceedings of the 2003 IEEE International Performance, Computing, and Communications Conference, 2003, 9-11 April 2003, pp. 129-136.
8. Tzevelekas, L., Ziviani, A., Amorim, M. D. D., Todorova, P., and Stavrakakis, I., "Towards potential-based clustering for wireless sensor networks", Proc. of the 2005 ACM conference on Emerging network experiment and technology, Toulouse, France, 2005, pp. 292-293.
9. I. Wokoma, L. S. cks and I. Marshall, "Clustering n Sensor Networks using Quorum Sensing," in the London Communications Symposium, University College London, 8th-9th September, 2003.
10. Banerjee, S. and Khuller, S., "A clustering scheme for hierarchical control in multi-hop wireless networks", Proc. of the IEEE INFOCOM 2001, Volume 2, 22-26 April 2001, pp. 1028 - 1037.
11. Clark, B. N., Colbourn, C. J., and Johnson, D. S., "Unit Disk Graphs", Discrete Mathematics, 86: 165-177, 1990.
12. Garey, M. L. and Johnson, D. S., "Computers and Intractability: A Guide to the Theory of NP-Completeness", W. H. Freeman, San Francisco, 1979.
13. Das, B. and Bharghavan, V., "Routing in ad-hoc networks using minimum connected dominating sets", Proc. IEEE International Conference on Communications (ICC'97), June 1997, pp. 376-380.
14. Erdos and Renyi, "On Random Graphs", Publ. Math. Debrecen, Volume 6 (1959), pp. 290-297.
15. Karlof, C. and Wagner, D., "Secure routing in wireless sensor networks: Attacks and countermeasures", Elsevier's Ad Hoc Network Journal, Special Issue on Sensor Network Applications and Protocols, September 2003, pp. 293-315.
16. Pathan, A-S. K., Lee, H-W., and Hong, C. S., "Security in Wireless Sensor Networks: Issues and Challenges", Proc. of the 8th IEEE ICACT 2006, Volume II, 20-22 February, Phoenix Park, Korea, 2006, pp. 1043-1048.
17. Gupta, G. and Younis, M., "Load-balanced clustering of wireless sensor networks", IEEE International Conference on Communications (ICC'03), Volume 3, 11-15 May 2003, pp.1848-1852.