A Secure Lightweight Approach of Node Membership Verification in Dense HDSN

Al-Sakib Khan Pathan, *Student Member, IEEE*, Gihyuk Heo, and Choong Seon Hong, *Member, IEEE*

***Abstract*—In this paper, we consider a particular type of deployment scenario of a distributed sensor network (DSN), where sensors of different types and categories are densely deployed in the same target area. In this network, the sensors are associated with different groups, based on their functional types and after deployment they collaborate with one another in the same group for doing any assigned task for that particular group. We term this sort of DSN as a heterogeneous distributed sensor network (HDSN). Considering this scenario, we propose a secure membership verification mechanism using one-way accumulator (OWA) which ensures that, before collaborating for a particular task, any pair of nodes in the same deployment group can verify each other's legitimacy of membership. Our scheme also supports addition and deletion of members (nodes) in a particular group in the HDSN. Our analysis shows that, the proposed scheme could work well in conjunction with other security mechanisms for sensor networks and is very effective to resist any adversary's attempt to be included in a legitimate group in the network.**

***Index Terms*—Group, One Way Accumulator, Security, Sensor**

I. Introduction

WIRELESS Sensor Networks (WSNs), composed of hundreds or thousands of inexpensive, low-powered sensing devices with limited computational and communication resources, provide a lucrative interface to the real world for acquiring specific types of data from specific areas of interest. A Distributed Sensor Network (DSN) is a variant of wireless sensor network. It differs from the traditional WSN in the sense that, it contains considerably huge number of sensors which are intended to be deployed over a large coverage area, where the communications among the sensors could be monitored. In such a network, the sensors are under constant threat of being captured by the enemy or of being manipulated by the adversaries. DSN is dynamic in nature in the sense that, new sensors could be added or deleted whenever necessary [1]. These sorts of networks are suitable especially for covering large areas for monitoring, target tracking, surveillance, moving object detection etc. which are very crucial tasks in many military and public-oriented applications.

It is anticipated that in most application domains, DSNs constitute an information source that is a mission critical system component and thus, require commensurate security protection. If an adversary can thwart the work of the network by perturbing the information produced, stopping production, or pilfering information, then the usefulness of sensor networks is drastically curtailed. So, it should be made sure that, the sensors that are participating in the data acquisition and supplying process are authentic, and are included as legitimate entities in the network.

In this paper, we focus on the secure management of group membership of the nodes in a heterogeneous distributed sensor network. We consider a network where different types of sensors take part as the contributors of a large network. In our scheme, we adopt One-Way Accumulator (OWA) [2] for group membership verification. Here, we restrict our attention only on the secure membership management and addressing other sorts of security (for example; routing security, data aggregation security, preventing DoS/DDoS attack, jamming etc.) for DSN is out of the scope of this paper.

The rest of the paper is organized as follows: following the Section I, Section II states our motivation and related works, Section III mentions our network model, assumptions and preliminaries, Section IV describes our approach of secure membership management, Section V contains the performance analysis, and Section VI concludes the paper with future research directions.

II. Motivation and Related Works

Benaloh and Mare [2] proposed a one-way hash function which satisfies a *quasi-commutative* property that allows it to be an accumulator. This property could be used for time stamping, building trust relationship between entities in many systems, and for solving variety of problems. We use the *quasi-commutative* property of one-way accumulator to serve our purpose of secure group management in heterogeneous distributed sensor network.

Prigent et. al. [3] presented a user-friendly distributed

Manuscript received May 22, 2007. This work was supported by the MIC and ITRC projects.

Al-Sakib Khan Pathan is with the Networking Lab, Department of Computer Engineering, Kyung Hee University, South Korea (phone: +82 31 201-2493; fax: +82 31 204-9082; e-mail: spathan@networking.khu.ac.kr).

Gihyuk Heo is with the Networking Lab, Department of Computer Engineering, Kyung Hee University, South Korea (phone: +82 31 201-2987; e-mail: jhheo@networking.khu.ac.kr).

Dr. Choong Seon Hong is with the Department of Computer Engineering, Kyung Hee University, South Korea. (phone: +82 31 201-2532; fax: +82 31 204-9082; e-mail: cshong@khu.ac.kr).

1-4244-1513-06/07/$25.00 ©2007 IEEE

approach to set up and maintain a secure long term community over a home ad hoc network. In their scheme, there is no central point to the community because; each device of the community considers itself as the central point that is, any device can introduce any other in its community provided that, they can communicate, even over insecure links.

Singh [4] did a study on the membership management protocols for groups in wireless sensor networks. The author investigated various sorts of applications, different geographic distributions and membership models relevant to sensor networks. In [5], the authors proposed a practical model of deploying the sensors in groups. Here, the authors considered deployment of sensor groups in such a way that the same group members stay close to each other after the deployment in the network. Based on the deployment model, the authors developed a novel group-based key pre-distribution framework, which can be combined with any of the existing key pre-distribution techniques. Zhou et. al. [6] proposed a group-based key predistribution scheme, GKE, which ensures secure node-to-node communication between any pair of sensors. According to [6], GKE provides a number of advantages like, accommodating different deployment models, establishing unique pairwise key regardless of sensor density or distribution, nearly resilient feature against node capture attacks and low communications overhead.

Motivated by the above mentioned works, in this paper, we propose a secure group membership management scheme. We adopt one-way accumulator (OWA) for testing the legitimacy of the group members (sensors) in a particular group in the network. Other security schemes could be employed along with this scheme to ensure robust security for the network.

III. Our Network Model and Preliminaries

A. Network Model

We consider a densely deployed Heterogeneous Distributed Sensor Network (HDSN), where sensors of various functional capabilities form different functional groups. For example, in a particular HDSN, let us suppose that, the temperature sensing sensors form a portion of the total network whose task is to sense and report the average temperature of a certain area of interest. Side by side, there are other types of sensor groups (or deployed sensor groups, *DSG*s) in the same area for other tasks like monitoring the seismic signals in that area, a *DSG* for sensing the acoustic signals, a *DSG* for tracking the moving objects in that area etc. Figure 1 shows an example scenario. Here, we have shown three types of deployed sensor groups (*DSG*s) in the same HDSN. The black, gray and white nodes are associated with three different *DSG*s. The sizes of these *DSG*s are dependant on the application at hand or are determined based on the requirements. These *DSG*s could have their separate sinks (or base stations) which could be interconnected with each other or could have a single sink for the network where all of them send their calculated reports. Also it is possible to have inter-group communications using efficient security mechanisms and a *DSG* for a particular function might also be divided into several sub-groups (sub-*DSG*s). Addressing the method of communications among sub-groups or among different types of *DSG*s is beyond the scope of this paper and will be noted in our future works.

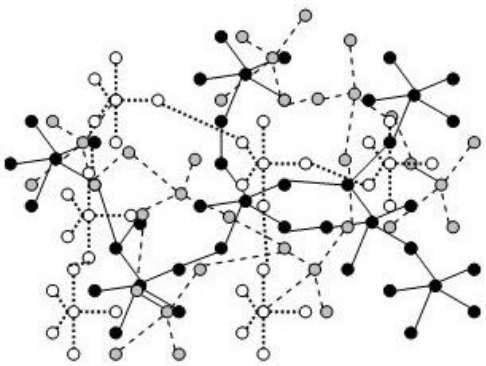

Fig. 1. An Example of Heterogeneous Distributed Sensor Network. Here, there are three types of deployed sensor groups that are participating at the same time for a particular target region. The black, gray and white nodes are the members of separate *DSG*s.

We assume that, in each *DSG*, for each participating node there is an end-to-end path from the corresponding sink, which is created via some other efficient mechanism. That is in a particular *DSG*, G_i, i=1, 2, 3….ξ, (where ξ is the maximum number of groups in the HDSN), for each node n_{G_i} there is a path from the corresponding sink, $S_{G_i} \rightarrow n_{m_i} \rightarrow n_{m_i\text{-}1} \rightarrow \ldots \rightarrow n_{G_i}$. There is also a secure node-to-node communication mechanism and other associated security protections in the network like an intrusion detection system which detects the abnormal behavior of nodes in the network.

The sink has enough processing power to do the initial calculations to initiate the groups of sensors. The sensors deployed in the network have the computational, memory, communication and power resources like the current generation of sensor nodes (e.g., MICA2 motes [7]). Once the sensors are deployed over the target area, they remain relatively static in their respective positions.

Based on this network model, our goal here is to propose a mechanism by which the nodes in a certain *DSG* (or in a sub-*DSG*) might securely recognize one another so that any adverse entity cannot in any way be included in the network throughout its lifetime. This sort of membership verification could especially be required for performing some collaborative tasks in the network. For example, when the sink wants to know the average temperature of a certain region, the nodes in that region might have to work together to measure the average of the temperature readings over that particular area. There could also be some sort of data aggregator or pre-processor which would be responsible for doing the primary calculations. In fact, our secure membership verification mechanism could well be used with other clustering mechanisms where there are some cluster heads present to collect these sorts of readings from a certain set of sensors.

B. One Way Accumulator (OWA)

A one-way function F is a function with the property that,

for a given x, it is easy to compute $y = F(x)$. However, given F, y, it is computationally infeasible to determine x, such as, $x = F^{-1}(y)$. Generally, one-way functions take a single argument. However, Benaloh and Mare [2] considered hash functions which take two arguments from comparably sized domains and produce a result of similar size. In other words, according to [2], a hash function is a function F with the property that, $F : A \times B \rightarrow C$ where, $|A| \approx |B| \approx |C|$. This view introduces the one-way hash function with a special *quasi-commutative* property which is termed as one-way accumulator (OWA). According to the definition, OWA is a one-way function, $f : X \times Y \rightarrow X$ with the *quasi-commutative* property such that, for all, $x \in X$ and for all, $y_1, y_2 \in Y$,

$$f(f(x, y_1), y_2) = f(f(x, y_2), y_1)$$

A family of one-way accumulators is a family of one-way hash functions each of which is *quasi-commutative*.

This property is not unusual. In fact, addition and multiplication modulo *n* both have this property as does exponentiation modulo *n* when written as, $e_n(x, y) = x^y \bmod n$. Modular exponentiation also satisfies the *quasi-commutative* property of one-way accumulator:

$$f(f(x, y_1), y_2) = f(f(x, y_2), y_1) = x^{y_1 y_2} \bmod n$$

This could be extended for a long sequence of y_j values (where, $j = 1, \cdots, m$).

The *quasi-commutative* property of one-way accumulators *f* ensures that if one starts with an initial value, $x \in X$, and a set of values $y_1, y_2, \cdots, y_m \in Y$, then the accumulated hash,

$$z = f(f(f(\cdots f(f(f(x, y_1), y_2), y_3), \cdots, y_{m-2}), y_{m-1}), y_m)$$

would be unchanged if the order of the y_js were permuted. This feature could be used for membership verification in a large set of entities. We adopt this feature of OWA for secure membership management in HDSN.

IV. Secure Membership Management with OWA

A. Membership Management in DSN

As the concomitant fields of applications of sensor networks are increasing and gradually getting complex with the advancements of technologies, sometimes it is necessary in some applications to ensure secure membership of nodes in a distributed sensor network. The simplest way to maintain the membership information of a particular group of sensors could be storing the member ids (the ids of each participating sensor in that group) in each sensor node. However, the storage requirement for such member id list linearly increases with the increase of the number of sensors in that particular group. For the sensors with limited storage capabilities, this is not a good solution. Hence, we employ an efficient OWA-based scheme for managing the membership information of a group in such a way that, it could well be supported by the storage and computation power of the modern-era sensors.

B. Calculating Partial Accumulated Hash Value (PHV)

In case of one-way accumulator, if the values, $y_1, y_2, \cdots, y_m$ are associated with the users of any cryptosystem, the accumulated hash z of all of the y_js can be computed. A user holding a particular y_j can compute a partial accumulated hash z_j of all y_i with $i \neq j$. The holder y_j can then demonstrate that y_j was a part of the original hash by presenting z_j and y_j such that, $z = f(z_j, y_j)$. We use this partial accumulated hash values in the membership verification process. The following sub-sections present our scheme in details.

C. Pre-Processing and Pre-storing of PHV

Before deployment of a group of sensors, the following steps are performed:

1. A unique id, y_j, $j = 1, \cdots, m$ is assigned for each sensor participating in a particular deployment group.
2. Two safe relatively prime numbers, p and $q = 2p + 1$ are generated.
3. *n* and $\phi(n)$ are computed as, $n = pq$ and Euler's totient function, $\phi(n) = (p-1)(q-1)$.
4. A random number x (as a seed) is generated which is same for every node in the group.
5. *PHV* for each node y_j is computed using the formula,

$$z_j = x^{\prod_{i=1, i \neq j}^{m} y_i} \bmod n$$

6. Now the values of z_j, n, $\phi(n)$ and corresponding y_j are stored in each sensor in the deployment group.

D. Secure Membership Verification

After deployment of the sensors in the target area, if a node needs to verify the membership of another node (whether they are in the same *DSG* or not), the *PHV*s and the identities of the nodes are used. For example, let us suppose that, two nodes, n_p and n_q want to verify whether they are in the same group or not. For this membership verification, these two nodes exchange their pre-stored partial accumulated hash values z_p, z_q and their identities, y_p, y_q. Node, n_q calculates $z = f(z_p, y_p) = z_p^{y_p} \bmod n$, while the other node calculates, $z = f(z_q, y_q) = z_q^{y_q} \bmod n$ locally. If both of the locally computed one-way accumulator values match with each other, the nodes could be sure that, they are participating as the siblings in the same *DSG* in the HDSN. Once the

accumulator value is calculated and matched, it could be preserved in the node for successive node membership verification for a given collaborative task. As mentioned in the network assumption, there is a mechanism which ensures secure communications between any pair of nodes. As an example a scheme like [8] (or other node-to-node secure communication scheme like [6]) can be employed to set up the secure communication link between two nodes in the network. In fact, using the secure communication mechanism, any node can communicate with any other node in the HDSN but to work together for a particular task, they must be of the same categories and must be the members of the same *DSG*.

E. Addition of New Member in a DSG

Addition of new sensors in a particular *DSG* for a particular area could be handled in two ways. At the time of deploying the sensors in a group, all the sensor in that group might not be used, rather some of them could be kept for later use, which is actually dependant on the requirement or the application at hand. Say for example, total number of sensors in a group before deployment is λ. So, a certain portion, say η of these sensors could be deployed first for that particular group and the remaining, $(\lambda-\eta)$ sensors could be added later and in such a case, all the newly added sensors could still be able to prove their legitimate membership to other already deployed sensors in that group using our OWA-based verification scheme. Note that, the nodes in two different groups cannot pass the membership verification process but if required, they are allowed to communicate with each other using the assumed secure communication mechanism.

In the first method, we basically handle addition of new sensors just by employing a good deployment policy. However, one-way accumulators allow addition of completely new sensors for a certain *DSG* in the HDSN. For example, let us consider that, a new sensor has the id y_{new}, which is assigned from the base station. Mathematically, the new OWA is, $z_{new} = f(z, y_{new})$. To inform all the sensors in that particular *DSG* about the newly added sensor, the base station uses the dedicated end-to-end path (according to our assumption) of each sensor. In turn, each sensor updates its *PHV* using the formula,

$$z_{j\,new} = f(z_j, y_{new}) = z_j^{\,y_{new}} \bmod n$$

Before deployment of the new node, the base station calculates its PHV and stores in it. To add a new set of sensors in a *DSG*, the base station securely sends all the ids of the newly included sensors to the deployed sensors of that group.

F. Purging of Member from a DSG or from a sub-DSG

Any suspicious behavior detected by the intrusion detection system could convince the base station to purge any sensor from a particular group in the HDSN. To purge an adverse node, y_{adv}, the sink uses the secure end-to-end paths for the sensors in the *DSG* to send the id of the deleted node. Getting the delete command, each of the remaining nodes calculates the stored *PHV* using the equation,

$$z_{u\_j} = z_j^{\,y_{adv}^{-1} \bmod \phi(n)} \bmod n$$

Euler's totient function $\phi(n)$ is used here for the modular operation to ensure that underflow doesn't occur and the purged id could not be reused by any adversary. In case of a node failure due to any unwanted incident like power outage (or other), it should be made sure that the node's id could not be used by any other entity or any attacker. So, to handle this, the same procedure for node purging is employed. And the sink is responsible for taking decision of purging. In some applications, where the clustering techniques are employed, it is possible to assign the charge of taking group-related decisions to the cluster head of the particular cluster (or sub-group). In this case, our scheme offers a decentralized node membership verification mechanism and reduces the burden of tasks of the corresponding sink(s).

V. PERFORMANCE ANALYSIS

We analyzed our scheme in terms of security, complexity and storage requirements. To understand the performance of our scheme, in our system setting, we varied the number of nodes in the groups from 25 to 300. As we use similar type of method like RSA [9], like RSA cryptosystem, our scheme requires that, the size of *n* should be sufficiently large, typically 1024 bits or more. We kept the values of z_j, n, $\phi(n)$ of same length (1024 bits) but considered different lengths (in bits) for the ids of the sensors. Each sensor node in a group has to store a little information; only four values for the secure membership verification mechanism. Figure 2(a) shows the storage requirements for a single sensor when our scheme is employed and in the traditional membership list storing technique. Figure 2(b) shows the storage requirement for variable length of the values for y_j in OWA-based scheme. From the figures it is evident that, the storage requirement for employing our scheme could be well-met with current generation sensor nodes. From these figures it is clear that, our scheme is very efficient in terms of storage as it doesn't depend on the size of the group and doesn't increase with the increase of the size of the group. For larger lengths of n, the storage requirement increases. However, still it is affordable by today's sensors and comparatively much less than that of storing the whole membership list.

One-way accumulator uses one-way hash function which means that, given $x \in X$ and $y \in Y$, for a given $y' \in Y$, it is difficult to find an $x' \in X$ such that, $f(x,y) = f(x',y')$. So, an adversary that wants to forge a particular y' would face with the difficulty of constructing an x' with the property that, $z = f(x',y')$. Likewise, in our scheme, the use of arbitrary values for *PHV* and identity of node cannot pass the membership verification mechanism and the adversary cannot in any way be included in the group. A potential threat is that, if a dishonest member in the group tries

to construct a false pair (x', y') such that, $z = f(x', y')$ by combining various node identities (y_j s) in one way or another. However, as mentioned earlier, this is not practical as the adverse node faces the difficulty of finding such a pair. Other methods of generating the pair might be possible. However, this could be handled by restricting the choice of the identities (set of y_j s) of nodes, which is dependant on the decision of the central entity and based on the requirements.

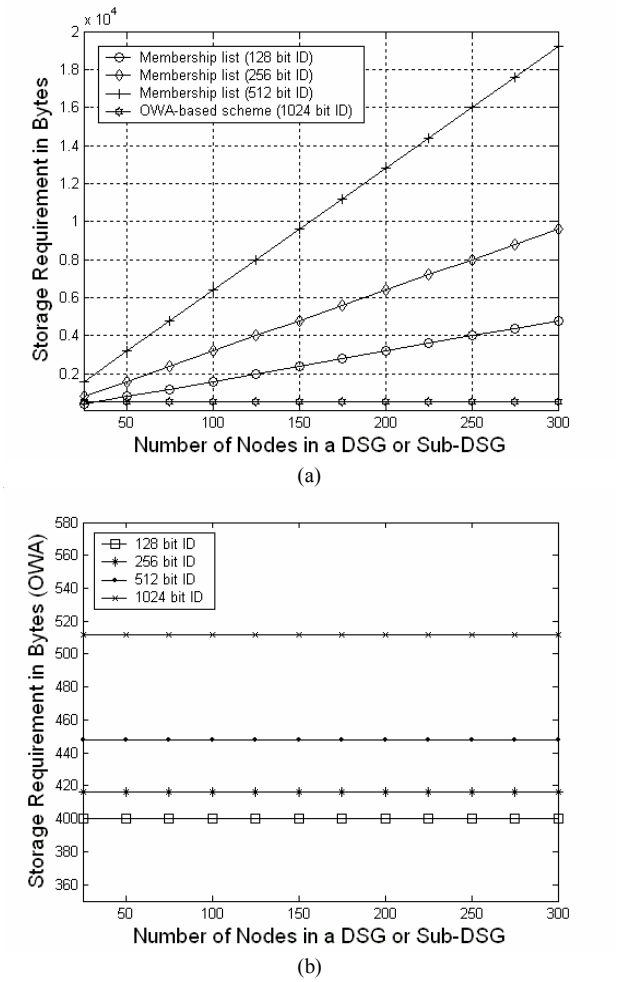

Fig. 2. Storage requirement (a) For a single node in membership list storing mechanism (for different lengths of IDs) and our scheme (considering ID and other values are of 1024 bits in length) (b) For a single node in OWA-based scheme for various lengths of IDs.

In the pre-processing stage (section IV(C)), we use a rigid value of *n*. According to Benaloh and Mare [2], the advantage of using a rigid integer, $n = pq$ is that the group of squares (quadratic residues) modulo *n* that are relatively prime to *n* has the property that it has size, $n' = \frac{(p-1)}{2} \cdot \frac{(q-1)}{2}$ and the function, $e_n(x, y) = x^y \bmod n$ is a permutation of this group whenever y and n' are relatively prime. Thus, if the factorization of n is hidden, "random" exponentiations of an element of this group are extremely unlikely to produce elements of any proper subgroup. This means that repeated applications of $e_n(x, y)$ are extremely unlikely to reduce the size of the domain or produce random collisions. Although constructing rigid integers is somewhat harder than constructing ordinary, '*difficult to factor*' integers, it is still quite feasible.

Another advantage of our scheme is that, as the entire pre-processing step is done by the base station, the sensors do not need to bother about the calculations and no sensor resources are used for initializing the deployable groups of sensors.

In this paper, we have basically focused on ensuring secure membership of the nodes in the deployed groups in HDSN. Other security mechanisms can run side-by-side our scheme. In our approach, several different groups of sensors could operate in the same heterogeneous distributed sensor network at the same time without hampering the operation of the nodes of other group. If required, there could be some other mechanisms of communications between the sinks of different groups or the nodes of different groups. This actually depends on the type of service required from the HDSN or the application at hand. At the time of deployment, if policy-based deployment is used, a certain portion of the sensors could be kept for future deployment. Yet, it would not affect anything in the deployed sensor group and all the other sensors in the DSG can verify each other's legitimacy of membership.

VI. Conclusions and Future Works

In this paper, we have proposed a secure membership verification mechanism and node management scheme for a heterogeneous distributed sensor network. Our scheme could be employed alongside other supplementary security mechanisms for DSN. Our analysis shows that our scheme requires considerably very small storage and processing power, and is efficient enough to ensure secure membership of nodes in the groups in HDSN. As our future work, we will develop an efficient secure communication mechanism between any two nodes in the HDSN which will work along with our membership verification scheme and will require considerably less storage, computation and energy resources than the other existing schemes.

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] Benaloh, J. and Mare, M. d., "One-Way Accumulators: A Decentralized Alternative to Digital Signatures," *LNCS 765*, Springer-Verlag, pp. 274-285, 1994.

[3] Prigent, N., Bidan, C., Andreaux, J.-P., and Heen, O, "Secure Long Term Communities in Ad Hoc Networks," *Proc. of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN'03)*, 2003, pp. 115-124.

[4] Singh, K. H., "A Study of Membership Management Protocols for Groups in Wireless Sensor Networks," M.S. thesis, Dept. Computer Science, University of Illinois at Urbana-Champaign, USA, 2004.

[5] Liu, D., Ning, P., and Du, W., "Group-Based Key Pre-Distribution in Wireless Sensor Networks," *Proc. of the 4th ACM WiSE'05*, Cologne, Germany, 2005, pp. 11-20.

[6] Zhou, L., Ni, J., and Ravishankar, C. V., "Efficient Key Establishment for Group-Based Wireless Sensor Deployments," *Proc. of the 4th ACM WiSE'05*, Cologne, Germany, 2005, pp. 1-10.
[7] Xbow Sensor Networks, Available at: http://www.xbow.com/
[8] Pathan, A.-S. K., Dai, T. T., and Hong, C. S., "A Key Management Scheme with Encoding and Improved Security for Wireless Sensor Networks," *LNCS 4317*, Springer-Verlag, pp. 102-115, 2006.
[9] Rhee, M. Y., *Internet Security Cryptographic principles, algorithms and protocols*, WILEY, ISBN 0-470-85285-2, 2003