A Secure and Efficient Way of Node Membership Verification in Wireless Sensor Networks*

Al-Sakib Khan Pathan and Choong Seon Hong
Department of Computer Engineering, Kyung Hee University
1 Seocheon, Giheung, Yongin, Gyeonggi 449701, Korea
spathan@networking.khu.ac.kr and cshong@khu.ac.kr

ABSTRACT

This paper proposes an efficient mechanism of node membership verification within the groups of sensors in a wireless sensor network (WSN). We utilize one-way accumulator to check the memberships of the legitimate nodes in a secure way. Our scheme also supports the addition and deletion of nodes in the groups in the network. Our analysis shows that, our scheme could be well-suited for the resource constrained sensors in a sensor network and it provides a lightweight mechanism for secure node membership verification in WSN.

1. INTRODUCTION

The sensors in a wireless sensor network (WSN) are often used for forming logical groups within the network for performing collaborative tasks. These sorts of collaborative tasks need to ensure that, the sensors participating in the groups are legitimate and securely introduced in the group. In this paper, we adopt the one-way accumulator for secure node membership verification in the groups formed in wireless sensor networks. Based on the one-way accumulator, we also propose efficient mechanisms to handle joining and leaving of nodes in the groups in the network.

The structure of this paper is as follows: following the Section 1, Section 2 mentions the preliminaries, Section 3 presents our mechanism, Section 4 contains the analysis and finally, Section 5 concludes the paper.

2. ONE-WAY ACCUMULATOR

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 [1] considered a hash function *F* with the property that, $F : A \times B \to C$ 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, $f : X \times Y \to X$ with the *quasi-commutative* property such that, for all, $x \in X$ and for all, $y_1, y_2 \in Y$,

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

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 OWA *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 WSN.

3. OUR PROPOSED SCHEME

3.1. Network Model

We consider a WSN with dense deployment of sensors. We assume that, for each node in a certain group, there is a secure end-to-end path from the corresponding base station. The base station could be a single entity for the whole network or separate sinks could be there in the network for each group. The sensors participating in the network has the similar memory, computation and energy resources like the modern-era sensor nodes like MICA2 motes [2].

3.2. Pre-processing Before Deployment of Sensors

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

* This paper was supported by ITRC and MIC

sensor participating in a particular deployment group.

2. Two safe 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. Partial accumulated hash value (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.

3.3. Secure Node 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 group 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, 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, $z = (z_q, y_q) = z_q^{y_q} \bmod n$ 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 group in the WSN. 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.

3.4. Addition and Deletion of Nodes in a Group

Suppose, a new sensor has the id y_{new}, which is assigned from the base station and wants to join a group in the network. Mathematically, the new OWA is, $z_{new} = f(z, y_{new})$. To inform all the sensors in that particular group 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 this value in it. To add a new set of sensors in a group, the base station securely sends all the ids of new sensors to the deployed sensors of that group.

For purging a node y_{adv} from a group, the delete command is issued by the base station and 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).

4. ANALYSIS

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_js) 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_js) of nodes, which is dependant on the decision of the central entity and based on the requirements. The memory requirement for our scheme could also be supported by today's sensors as for this scheme a node has to remember only four values.

5. CONCLUSION

In this paper, we have proposed an efficient, lightweight and secure node membership verification technique for the groups in a wireless sensor network. In future we would like to develop other supplementary security mechanisms to ensure robust security in wireless sensor network.

REFERENCES

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

[2] http://www.xbow.com/