Energy Conserving Security Mechanism for Wireless Sensor Network*

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

Department of Computer Engineering, Kyung Hee University,
1 Seocheon, Giheung, Yongin, Gyeonggi, 449-701, Korea
{hamid, mustafiz}@networking.khu.ac.kr, cshong@khu.ac.kr

Abstract. This paper describes Wireless Sensor Network (WSN) security to conserve wasteful energy. Sensor networks are emerging fast and will be the next wave towards new network appliances. Security must be justified and ensured before the large scale deployment of sensors as individual sensors are prone to security compromise. In the sensor field, an adversary can compromise sensor nodes that can be used to generate random false sensing data. As these generated packets propagate through the network towards final data acquisition point, it will result in the energy consumption in a constrained low powered network. As WSN is multi-hop communication in nature, node-to-node authentication using shared secret is important for legitimate data packets to be forwarded. In this paper, we develop a security mechanism to detect energy-consuming useless data flows that propagate through network. Assuming that a sensor node can sense an event and generates multiple Message Authentication Code (MAC) using secret keys and these MACs are appended to the sensed data. The forwarding nodes along the path towards the data acquisition point verify the validity of the sensed data by checking the authenticity of the MACs attached to the original sensed data. Intuitively, early detection of the false data will make the entire network energy conserving which is one of the primary goals in the design of sensor networks. We have quantified the security strength through analysis and simulation.

1 Introduction

To integrate general purpose computing with multiple sensing and wireless communication capabilities, modern advanced nano-technology makes it technologically feasible and economically viable to develop low-power, battery operated devices. This tiny device is known as sensor node. It is envisioned, for most of the application, that a massive random deployment of sensor nodes, numbering in thousands or tens of thousands (Fig. 1). Harmonizing sensor nodes into sophisticated computation and communication infrastructures, called sensor network, will have strong impact on a wide variety of sensitive applications including military, scientific, industrial health and home network. The expected achievement of such a wireless sensor network is to produce, over an extended period of time, global information from local data sensed by individual sensor nodes.

* This work was supported by MIC and ITRC Project.
** Corresponding author.

M. Gavrilova et al. (Eds.): ICCSA 2006, LNCS 3981, pp. 866 – 875, 2006.

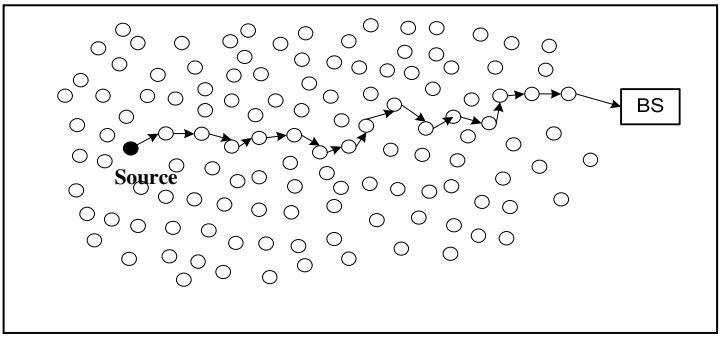

Fig. 1. Large-scale Wireless Sensor Network, source senses the events and forwards the data packet towards the Base Station (BS)

The characteristics of sensor network differ from traditional wireless sensor networks in a way where energy conservation and self-configuration are primary goals, while per-node fairness and latency are less important.

Misbehavior (by an adversary or a compromised node) that threats the work of the network by perturbing the information produced, stopping production, or proliferating information, then the perceived usefulness of sensor network will be dangerously curtailed. Implementing security mechanism to protect mass flow of bogus information can increase the life time of the entire network thereby conserving the energy. Note that, in sensor networks, in-network processing makes end-to-end security mechanisms harder to deploy because intermediate nodes need direct access to the content of the messages [3]. We address the following issues that lead us to developing a security protocol to deal with sensor network:

- Distribution of secret keys to the sensor nodes capable of checking the validity of the data by intermediate forwarding nodes.
- Engineer a security mechanism to detect and prevent the forged data packets to flow in the network, hence saving wasteful energy.
- Analysis and simulate the proposed protocol to justify the practicability.

The rest the paper is organized as follows. In section 2, we briefly explain some previous works and we define the problem and make some necessary assumptions in section 3. In section 4, initial key assignment is discussed. Then, we start presenting our security mechanism in section 5. We precisely state the data generation by source node and how to forward the data towards Base Station. Section 6 outlines the detection method of false data packet. We present an analytical description in section 7. We verify our analysis through simulation results in section 8. Finally, we make our short discussion and conclusion in section 9 and 10 respectively.

2 Related Work

Sensor network security has been studied in recent years in a number of proposals. Kulkarni et al. [2] analyzes on the problem of assigning initial secrets to users in

ad-hoc sensor networks to ensure authentication and privacy during their communication and points out possible ways of sharing the secrets.

In [3] Karlof et al. thoroughly discussed the problem of secure data transmission for different routing protocols and they conclude that many sensor network routing protocols have been proposed, but none of them have been designed with security as a goal. They suggested the security goals required for routing in sensor networks.

Passive attacks such as cipher text attack and chosen cipher text attacks, a security protocol has been proposed in [4] that ensures forward and backward secrecy of the session key, so that if any set of the session key is compromised, these compromised keys do not undermine neither the security of future session keys, nor the security of past session keys. Their works requires synchronization initiated by base station and also by sensor networks. SPINS [5] implements symmetric key cryptographic algorithms with delayed key disclosure on motes to establish secure communication channels between a base station and sensors within its range. Reference [6] implements ticket certification services through multiple-node consensus and fully localized instantiation, and uses tickets to identify and grant network access to well-behaving nodes. In URSA, no single node monopolizes the access decision or is completely trusted, and multiple nodes jointly monitor a local node and certify/revoke its ticket.

Sybil and Rushing attacks are well discussed in [8, 9]. Sybil attack is a threat to WSN where a node legitimately claims multiple identities. Random pairwise key distributions are discussed in [10] and [11] to make the sensor networks resilient to security threats. Wie Ye et al. in [1] proposed an energy efficient medium-access control protocol by keeping the sensor nodes periodically listen and sleep mode.

Our initial key assignment is a probabilistic key distribution presented in [2]. Our work focuses on the use of security mechanism to protect the unauthorized traffic to flow in the network and thereby saving network energy and increasing network resiliency.

3 Problem Definition and Network Assumption

Many sensor network routing protocols have been proposed, but none of them have been designed with security as a goal [3]. So, the network is susceptible to various kinds of security threats such as sybil attacks, wormholes, selective forwarding, acknowledgement spoofing, sinkhole and so on so forth. We assume that the adversary does not have the capability to attack the base station (i.e. sink) because the powerful base station can well protect any kind of malicious efforts. However, our assumption on network is that the attacker may know the basic approaches of the security mechanism and is able to either physically capture a node to obtain the security information installed in the sensor node or compromise through radio communication channel. Once captured, a node can be used to propagate sensed data that are false. Besides, it can launch various other attacks such as blocking the sensed data to be forwarded, record and replay old data thereby consuming network's overall energy. We focus on protecting the false data that are forwarded by intermediate forwarding nodes.

4 Initial Secret Key Assignment

In this section, we present the probabilistic protocol, the complementary tree protocol, for assigning the initial secrets. We will describe the single complementary tree protocol and then compute the multiple trees based key assignment. We organize the (Fig. 2) secrets in the tree of degree *d*. In this protocol, we require that *d>=3*. All nodes in the tree except the root are associated with a secret. Each leaf of the tree is associated with a sensor node. (Note that a leaf is associated with a sensor as well as a secret.) The secret distribution is as follows. For each level (except level 1), the node gets secrets associated with the siblings of its ancestors (including itself). Thus, node s_1 gets secrets k_2, k_3 (level 2), k_5, k_6 (level 3), k_{14} and k_{15} (level 4). A node does not get the secrets associated with its ancestors.

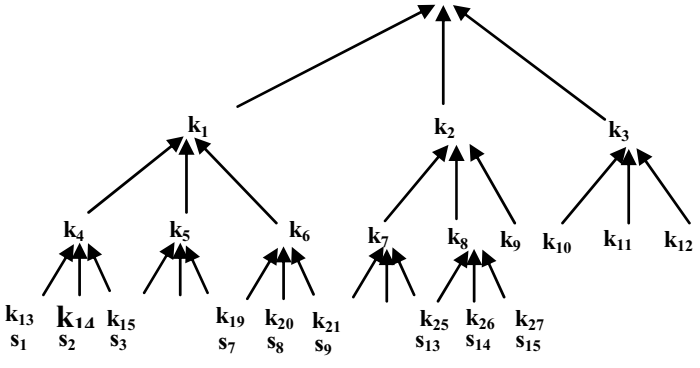

Fig. 2. Single Complementary Tree Key Assignment

When two nodes, say *j* and *k*, want to communicate, they first identify their least common ancestor. Let *z* be the least common ancestor of *j* and *k*. Let *x* denote the child of *z* that is an ancestor of *j*. Likewise, let *y* denote the child of *z* that is an ancestor of *k*. Now, to communicate, *j* and *k* use the secrets associated with all children of *z* except *x* and *y*. For example, if s_1 and s_2 want to communicate, they use the secret k_{15}. If nodes s_1 and s_9 want to communicate then they will use the secret k_5. And, if s_1 and s_{15} want to communicate then they will use the secret k_3.

It is possible to reduce the probability of compromise in the complementary tree protocol even further if we maintain multiple trees. More specifically, if we maintain *k* trees where there is no correlation between node locations in different trees, the probability of security compromise will be $((2/(d+1))^K)$. For detailed calculation authors request to see reference [2]. For *10* secret trees with degree $d = 3$, the probability of compromise is $(1/2)^{10} = 0.09\ \%$.

5 Data Generation and Forwarding

With the initial secrets, the sensors are deployed in the sensor field. From the key assignment protocol described in previous section, we suppose that there is pregenerated

total number of N keys and each sensor node has k number of keys. When an event occurs, the node that senses the signal will prepare an event's data as message to be sent to the base station through intermediate forwarding nodes. The message format is in the form of *[t, E]*, where t is the event detection time, and E is type of event.

Prior to forward the message to its neighbor, it randomly selects f number of keys from its k keys and generates f number of Message Authentication Codes (MACs) and attaches it with the event with the format: $[t, E, i1, M_{i1}, i2, M_{i2}, i3, M_{i3}, \ldots\ldots if, M_{if}]$, where the report contains f number of key indices and MACs. We set the constraint that a report with less than f MACs or key indices or one key used more than once to generate MACs, will not be forwarded. Intuitively, a larger value makes the injected false report to flow more difficult at the cost of increased overhead.

6 False Data Detection

Probabilistic key assignment ensures that each forwarding node has certain probability to possess at least one of the keys that are used to generate the MACs for a sensed data. So, each forwarding node is able to verify the correctness of the MACs attached with the packet. If a malicious (compromised) node has only one key, it can generate one correct MAC. Since there are f distinct MACs (and f distinct key indices) that must be present in a legitimate data packet, the attacker needs to forge $f-1$ key indices (i.e. needs to know valid keys) and corresponding MACs. This is a difficult task for a compromised node (attacker) as the pre-distribution of keys is in such a way where finding the exact key that is shared between any pair of sensor nodes is difficult as described in key assignment section. In case, each sensor node, carried keys randomly chosen from the total key pool, any attacker node can use f of its keys to generate multiple MACs, which would have been indistinguishable from those generated by f keys in the sending node.

At the time, the forwarding node receives the packet; it looks at the key indices and number of MACs. If it is less than f or one key index used more than once, the packet is detected as forged and thus dropped. Then if the node has any of the key indices common, it calculates the MAC using its own key and compares the result with the received MAC attached in the packet. The packet is dropped in case the attached one differs from the reproduced one. If it matches exactly or this forwarding node does not have any of the f keys in common, the node passes the packet to the next hop towards base station (sink).

7 Performance: Analytical Description

We consider two performance issues in this section. We first analyze the efficiency of illegitimate packet detection and secondly, we analyze the energy conservation through dropping of forged data packets.

7.1 Detection Efficiency

As our protocol deals with f MACs to send the packet through forwarding node, an adversary that has compromised keys in f or more, can successfully forge packets. In

this case, our proposed method can not detect or drop such forged packet. We compute the efficiency when an adversary has g number of compromised keys $(0 \leq g \leq f-1)$. So, if the attacker wants to forward forged data packet, he has to forge $f-g$ keys and MACs. Now, if attacker randomly chooses $f-g$ keys, we compute the probability that a forwarding node has one of the $f-g$ keys, and thus being able to detect an incorrect MAC and drop the forged packets. In this case, the probability that a sensor node has one of the $f-g$ keys, defined as p is: $p = (f-g) \times k / N$, where k is the number of keys each node possesses, N is the total number of keys. So, the per hop forged packet detection probability is $p_{per\text{-}hop} = p(1-p_{compromised})$. As the probability of compromise, $P_{compromised}$ is very small and thus it is negligible (key assignment section) and we take $p = p_{per\text{-}hop}$. So, we can compute the expected fraction of forge data being detected and dropped within h hop is given by $p_h = 1 - (1-p)^h$. The forged data packet traverses the average number of hops that is given by $\sum i(1-p)^{i-1}p = 1/p$, for $i = 0$ to ∞.

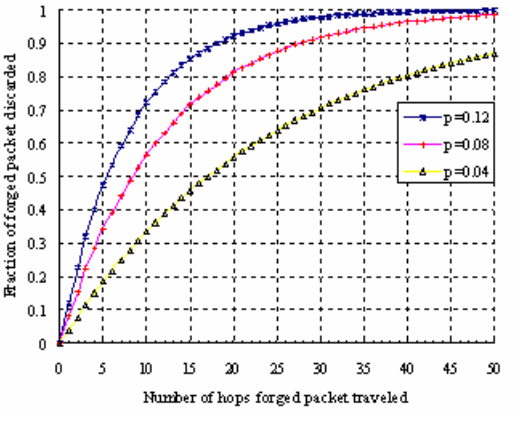

Fig. 3. Fraction of false data dropped as a function of number of hops traveled

The efficiency is shown in fig. 3, percentage of dropped packets increases as the number of hops grows. Here we consider, for example, each node maintains *20* keys, total number of keys *500* and each packet carries $f = 5$ MACs. We have $p = 0.12, 0.08\ 0.04$ when number of compromised keys $g = 2, 3$ and *4* respectively. Approximately *70%* false packets are dropped within *10* hops if the adversary has 2 compromised keys. In worst case, *70%* forged packets are dropped in *30* hops, when only one MAC is incorrect and they travel 25 hops on an average.

7.2 Energy Conservation

Energy consumption in sensor network is comprised of energy consumed in transmission, reception and computation. Extra parameters in our work are f key indices and f MACs. Let the length of the MACs and the key index be L_{MAC} and $L_{key\text{-}index}$, respectively. The length of original data is denoted as L_d. So, the total length of the data packet becomes, $L_{packet} = f \times L_{key\text{-}index} + L_{MAC} + L_d$.

Let the number of hops be h a data packet travels, amount of forged packets is Q_{forged} and legitimate data packet is *1*. The traffic travels all the h hops when security

mechanism is not incorporated in the network. But, with security, the false data traffic will travel exactly h hops with the probability $(1-p)^{h-1}p$. So, energy consumed to forward all the traffic without security, denoted by E and with security denoted by E_{sec}, will be:

$$E = L_d (E_t + E_r)(1+Q_{forged})h$$

$$E_{sec} = L_d (E_t + E_r)(1 + L_{MAC}/ L_d + fL_{key\text{-}index}/ L_d) (h + Q_{forged} (1- (1-p)^h)/ p)$$

where, the consumed energy in transmitting and receiving one byte are denoted by E_t, E_r respectively. And energy consumption for the computation of security parameter denoted by E_{comp} can be approximated by:

$$E_{comp} = f\times E_{MAC} + f\times h\times E_{MAC} + f\times E_{MAC}\times Q_{forged}(1- (1-p)^h)/ p$$

where E_{MAC} is the MAC computation energy. So, total energy consumption with security mechanism is $E_{sec\text{-}total} = E_{sec} + E_{comp}$.

We take the energy required [14] (values $E_t = 17$, $E_r = 13$ and $E_{MAC} = 16$ micro-Joules) to transmit, receive one byte of data and RC5 computation. Fig. 4 shows the performance with different number of false packet Q_{forged}, when $h = 25$, $L_d = 32$ bytes, $L_{key\text{-}index} = 10$ bits, $L_{MAC} = 64$ bits and attacker has *1, 2* and *3* compromised keys. We observe that, E increases much faster than $E_{sec\text{-}total}$ and thus conserves overall network energy. If Q_{forged} increases, the amount of energy that is saved gets higher and higher. More than *60 %* energy is saved when *8* false packets are present and an attacker has only one compromised key.

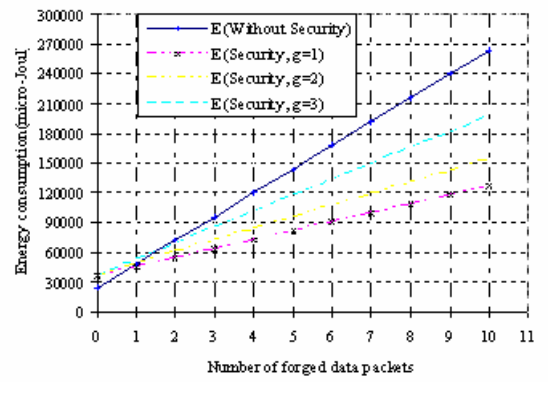

Fig. 4. Comparison of energy consumption with and without security as a function of forged data traffic

8 Simulation

Our proposed energy conserving security mechanism is simulated to further justify the analytical results. We present the packet discard efficiency and energy consumption in cases, when an adversary has $g = 0, 1,$ and *2* compromised keys. We simulate on the area of *60 X 60* Square meters, where *400* nodes are uniformly distributed. We place

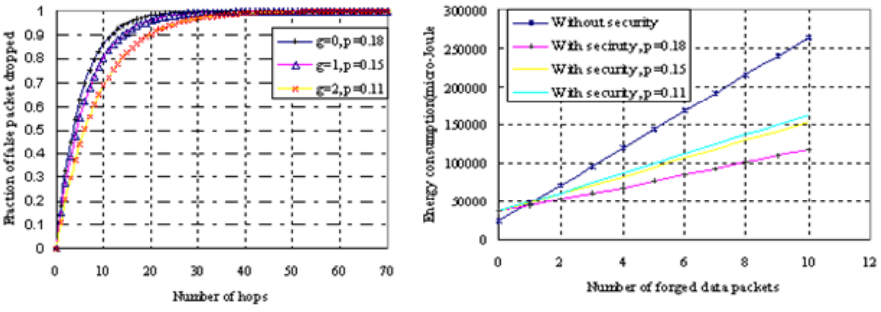

Fig. 5. Fraction of discarded false data packet (Left) and Comparison of energy consumption with and without security mechanism. The adversary has *0, 1* and *2* compromised keys.

the data acquisition point (Base station) and a source in opposite ends having about *80* hops in between. We consider stationary network for the simulation purpose. The packet transmission time is set to *15* ms. The source node generates one even in every 3 s. Each sensor node has $k = 30$ keys, and total $N = 800$ keys.

The efficiency of our protocol is justified as can be observed form the simulated results. Fig. 5 (Left) demonstrates the efficiency of discarding the bogus packets injected by the adversary as function of the number of hops traveled in case where $g = 0, 1,$ and 2 compromised keys respectively. In case, when no key is compromised, more than *80* % forged packets is dropped within *10* hops and *60* % with *2* compromised keys. More than 62 % energy is saved when an attacker has no compromised key and the number of forged packet is *10* (Graph in the right in fig.5). In case when attacker has 2 compromised keys, about *40* % energy is saved. Performance is even better when the number of false packets is higher.

9 Discussions

As the individual sensor nodes is subject to three fundamental constraints in storage, computation and power, asymmetric cryptographic operations is not feasible to apply. We used RC5 [16] to calculate the MACs. RC5 is a symmetric block cipher designed to be suitable for both software and hardware implementation. It is parameterized algorithm, with a variable block size, a variable number of rounds and a variable length of key. This provides the opportunity for greater flexibility in both performance characteristics and the level of security.

Probabilistic key assignment allows the neighbor nodes to share pairwise secrets to check the authenticity and thereby improving the resiliency of the network. The number of Message Authentication Codes is a parameter to be selected with the capability of the sensor node, else it will be heavy to compute and store the extra information. Also, it should not be too small because the forged packet will flow more hops.

The comparison of energy consumption with and without security protocol has been demonstrated (Fig. 5) and we have also shown the illegitimate packet detection efficiency. Our proposed security mechanism performs well in case where the data packet travels large number of hops.

We, at present, are exploring the feasibility of incorporating the security where the network can be managed using the idea of weakly connected dominating set. Each member of dominating set can be a zonal head to collect the data from a particular zone and forward through the shortest possible path and thereby consuming less overall network energy.

10 Conclusions

In this paper, we have put an effort to develop a security protocol that deals with the unauthorized data flow by an attacker. We have shown that our protocol can conserve significant amount of energy by validating the authenticity of the data packet. We have presented the complementary tree protocol for the distribution of secret keys to the sensor nodes capable of checking the validity of the data by intermediate forwarding nodes. We have presented the analytical results and compared the results with the simulated results. The protocol is efficient both in packet detection and energy conservation.

References

1. Wei Ye, J. Heidemann, D. Estrin: An Energy-Efficient MAC Protocol for WirelessSensor Networks. In: Proceedings of the IEEE Infocom, pp. 1567-1576. New York, NY, USA, USC/Information Sciences Institute, IEEE. June, 2002.
2. S. S. Kulkarni, M. G. Gouda, and A. Arora: Secret instantiation in ad-hoc networks. In: Special Issue of Elsevier Journal of Computer Communications on Dependable Wireless Sensor Networks, (2005) 1–15.
3. C. Karlof and D.Wagner: Secure routing in wireless sensor networks: Attacks and countermeasures. In: Elsevier's Ad Hoc Networks Journal, Special Issue on Sensor Network Applications and Protocols, 1(2–3)(2003) 293–315.
4. R. Di Pietro, L. V. Mancini, and S. Jajodia: Providing secrecy in key management protocols for large wireless sensors networks. In: Journal of AdHoc Networks, 1(4), (2003) 455-468.
5. V. Wen, A. Perrig, and R. Szewczyk: SPINS: Security suite for sensor networks. In: Proc. ACM MobiCom, (2001) 189–199.
6. H. Luo, J. Kong, P. Zerfos, S. Lu, and L. Zhang: URSA: Ubiquitous and robust access control for mobile ad hoc networks. In: Proc. IEEE/ACM Trans. Netw., Vol. 12, no. 6, (2004) 1049–1063.
7. A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, and et al.: A Line in the Sand: A Wireless Sensor Network for Target Detection, Classification, and Tracking. In: Computer Networks (Elsevier), Special Issue on Military Communications Systems and Technologies, 46(5) (2004) 605–634.
8. J.R. Douceur,: The Sybil attack. In: 1st International Workshop on Peer-to-Peer Systems (IPTPS _02) (2002).
9. Y. Hu, A. Perrig, and D. Johnson: Rushing attacks and defense in wireless ad hoc network routing protocols, In: Second ACM Workshop on Wireless Security (WiSe'03), San Diego, CA, USA (2003).

10. H. Chan, A. Perrig, D. Song: Random key predistribution schemes for sensor networks. In: IEEE Symposium on Security and Privacy (2003).
11. W. Du, J. Deng, Y. Han, P. Varshney: A pairwise key pre-distribution scheme for wireless sensor networks. In: ACM Conference on Computer and Communications Security (CCS), (2003) 42–51.
12. Y. Zhang, W. Lee: Intrusion detection in wireless ad hoc networks. In: Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking (2000) pp. 275–283.
13. S. Yi, P. Naldurg, R. Kravets: Security-aware ad-hoc routing for wireless networks. In: Proceedings of the 2001 ACM International Symposium on Mobile Ad Hoc Networking and Computing, ACM Press, New York(2001) 299–302.
14. Crossbow Technology Inc. [Online]. Available: http://www.xbow.com/
15. A. Manjeshwar, D. Agrawal: TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. In: 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing(2001).
16. A. Menezes, P. Oorschot, S. Vanstone, Handbook of Applied Cryptography, CRC Press, Boca Raton, 1996.