INTERNATIONAL JOURNAL OF INTELLIGENT CONTROL AND SYSTEMS
VOL. 12, NO. 4, DECEMBER 2007, 317-324

ROSS: Resource Oriented Security Solution for Heterogeneous Clustered Sensor Networks

Xiaomei CAO and Guihai CHEN

***Abstract*— Heterogeneous clustered sensor networks (HCSNs) help meet the cost, lifetime, and scalability requirements of real applications. However, the security solution should be reconsidered for their unique properties, such as uneven resources and in-network processing. In this paper, we provide a resource oriented security solution (ROSS) to protect the network connectivity of HCSNs. ROSS is a robust security solution that is embedded into network layer operations, resulting in in-depth defense against attacks from both external attackers and compromised nodes. Through security analysis and performance simulation, we demonstrate that ROSS not only achieves the predefined security goals, but also allows a tradeoff between security and performance cost.**

***Index Terms*— Heterogeneous sensor network, tiered architecture, secure data dissemination, multifence defence.**

1. INTRODUCTION

In recent years the successful deployment experience is pushing sensor networks beyond simple "sample and return" applications. Because constructing heterogeneous sensor networks allows the right components to be brought to bear on the individual application tasks, many applications will require a mixture of heterogeneous nodes. Generally, heterogeneous sensor networks contain two or more different types of nodes with varying levels of battery energy, computational and communication capabilities. These nodes are often organized into clusters to allow for scalability of MAC and routing. Large nodes usually act as cluster heads (CHs) and form a multihop communication backbone to carry aggregated traffic, while small nodes act as cluster members (CMs) which transmit sensing data to CH directly [1].

Wireless sensor networks are vulnerable to a wide range of security attacks [2]. For many mission-critical applications, security support should be an indispensable part of these networks. This paper aims to protect network connectivity of HCSNs from malicious attacks, which is the basis to support any network security services.

Up to now, many network layer security protocols have been proposed for homogeneous sensor networks and mobile ad hoc networks(*e.g.*, [3]- [10]). However, compared with these networks, HCSNs have some unique properties, such as uneven resources and in-network processing, which cause them vulnerable to many new security threats. For example, because CHs are responsible for critical functions such as data aggregation and routing in HCSNs, attacks involving CHs are particularly damaging. Ferreira et al. [6] and Su et al. [7] devise uniform security protocols for homogeneous cluster sensor networks which rotate the role of a CH over all the nodes. However, for HCSNs, it is convenient to exploit various security mechanisms to achieve security requirements of different nodes.

Several existing works(*e.g.*, [11]- [13]) have considered security issues in heterogeneous sensor network. Traynor et al. [11] propose an unbalanced hybrid pair-wise key establishment scheme for heterogeneous sensor networks. Oliveira et al. [12] propose solution that is suitable for networks with an arbitrary number of levels. However, this solution protect the network from attacks by outsiders only. In [13], Bohge and Trappe provide an authentication framework for hierarchical ad hoc sensor networks. This framework is based on Perrig's μTESLA [4] which to some extent also supports our contribution.

In this paper, we present the representative attacks against HCSNs and then propose ROSS to protect the network layer from these malicious attacks. The solution is designed to efficiently adapt to the underlying uneven resources, functionality and security requirements of different nodes. The design principle of ROSS is to perform heavy-duty computations at the base station (BS) and CHs, and minimize the role of CMs in dealing with security issue. More precisely, ROSS is a robust security solution which encompasses prevention, detection, and reaction mechanisms to defense against both external attackers and compromised nodes. We demonstrate the effectiveness of ROSS through both analysis and simulation results.

The rest of this paper is organized as follows: in Section II we give the statement of ROSS; in Section III we provide a detailed description of ROSS; in Section IV we analyze and evaluate the effectiveness of ROSS; in Section V, we discuss related work. Finally, conclusion is given and the future work is also pointed out in Section VI.

2. ROSS STATEMENT

2.1. Network Model

There are three key points that differentiate HCSNs from conventional sensor networks:

1) Different types of nodes have varying levels of battery energy, computational and communication capabilities.
2) Sensors do not communicate with each other. They transmit sensing data directly to their CH.
3) Large nodes aggregate sensing data and relay them via a separate, higher-capacity multi-hop network.

Manuscript received March 20, 2007; revised October 19, 2007. This work is partly supported by China NSF grants (60573131, 60673154, 60721002), Jiangsu Provincial NSF grants (BK2005208, BG2007039), and China 973 project (2006CB303000).

X. Cao and G. Chen are with the State Key Laboratory of Novel Software Technology, Nanjing University, Nanjing 210093, China (email: xmcao@dislab.nju.edu.cn; gchen@nju.edu.cn).

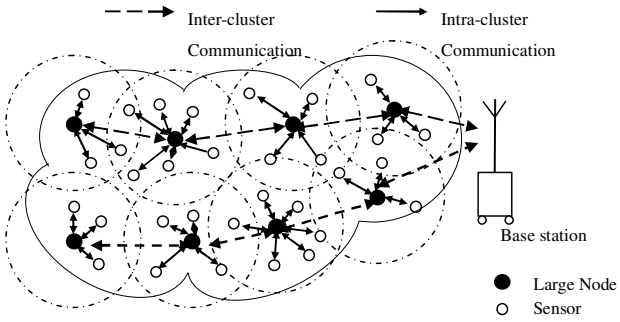

Fig. 1. Network and communication model of ROSS

Fig. 1 illustrates the network and communication model of ROSS. This architecture consists of three classes of wireless devices: high-power BS that routes received packets via radio links to the wired infrastructure, medium-powered large nodes that aggregate data and relay information from sensor nodes to BS, low-powered sensor nodes that have limited resources. We assume that large nodes are deployed uniformly in the whole region acting as CH of one sub-region. Each sensor node has a one-to-one relationship with its CH, and is part of exactly one cluster at a time. We also assume that all nodes are loosely time synchronized, and the network has sufficient redundancy, such that the attacker can not compromise all the nodes within a certain transmission range.

2.2. Attack Model and Security Goals

The malicious nodes can be either external nodes that do not know the cryptographic keys, or compromised nodes that possess the keys. A malicious node can perform attacks by itself or using arbitrary collusion with other nodes. Karlof et al. [3] give a general summary of attacks against homogeneous sensor networks. However, in HCSNs, several popular attacks could be enhanced to attack CHs, such as hello flood, sinkhole attack, black hole attack, select forward, and denial of service attacks. For example, the attacker broadcasts the bogus routing information and pretends to be a CH. Then, sensors send the sensing data to this attacker. The attacker can isolate BS by compromising the neighbor CHs of BS. Once all neighbor CHs of BS have been compromised, the sensing data could not be transmitted to BS and this make BS become a black hole. Moreover, attackers could send the redundant sensing data with large size to consume the energy of CHs.

There are three primary goals for ROSS: first, to ensure that the data received by BS is sent by an approved node; second, to verify that the data hasn't been modified on its way to the BS; third, to detect and isolate compromised nodes in time.

2.3. Assumption and Notations

In this work we largely neglect security threats in the physical layer and link layer. Such lower-layer attacks can be limited by mechanisms such as spread-spectrum technology, WEP protocol, and link layer misbehavior handling mechanisms [14]. We assume that multiple attackers may coexist in the network, and several attackers may even collude with each other. However, we assume that each group of colluding attackers has less than m nodes, where m is a design parameter (to be introduced in Section III).

While tamper resistance might be a viable defense for physical node compromise, extremely effective tamper resistance tends to add significant per-unit cost. Since sensor nodes are intended to be very inexpensive, we only assume BS has such ability. If an adversary compromises a sensor or a large node, all key material, data, and code stored on that node can be securely preserved from adversary only for a certain period of time. We also assume that the BS and large nodes have a certain degree of capacity for intrusion detection. They can monitor and perceive the invasions in the given sub-regions. Since intrusion detection may also incur extra computation overhead and energy consumption, we don't assume that sensor nodes have such capability. Finally, before all nodes are deployed into the network, the initial keys and certificates are already pre-distributed by the off-line trusted third party (TTP).

The notations that are frequently used in the paper are summarized in Table I.

TABLE I
NOTATION

Label	Meaning
A, B	Label of large nodes
a, b	Label of sensors
$M1 \| \| M2$	Message that combines M1 and M2
N_A	Nonce produced by A
K_A+, K_A-	A's public key and private key
$Cert_A$	A's public key certificate
K_{AB}	Authentication key shared by A and B
$\{M\}K_A+$	Encryption of M with key K_A+
$[M]K_A-$	Digital signature of M signed by K_A-
$F(M)$	Message digest of M computed by hash function
$HMAC(k, M)$	Message authentication code of M generated with authentication key k
QUERY	Query packet identifier
REPLY	Reply packet identifier
DATA	Data packet identifier

2.4. Overview of ROSS

According to nodes' functionalities and capacities, ROSS splits HCSNs into two layers: inter-cluster layer and intra-cluster layer. Different security mechanisms are provided to achieve security services at corresponding levels.

- Inter-cluster layer security mechanisms
  A mix-key cryptography mechanism is used to protect inter-cluster communication. That is, asymmetric encryption scheme is employed to achieve the entity authentication and session key establishment. Keyed-hash message authentication code(HMAC) is used to verify the integrity and authenticity of aggregated data. Besides, a collaborative reaction approach is used in this layer to

discover the occasional intrusions and take reactions to avoid persistent adverse effects.

- Intra-cluster security mechanisms
  In intra-cluster layer, ROSS adopts the concept of one-way hash chain proposed by the μTESLA protocol [4] to verify integrity and authenticity of sensing data. In addition, each CH maintains cluster member table (CMT) in a centralized way and isolate suspicious CM by deleting its item in this table.

3. Description of ROSS

3.1. Inter-cluster Security Mechanisms

3.1.1) Certificate Management: In inter-cluster layer, each CH A is pre-distributed with K_A+, K_A-, $CERT_A$ and $K_{BS}+$ by an off-line TTP. The certificate of node A can be expressed as: $CERT_A = [A\|K_A + \|T_{sign}\|T_{expire}]K_{BS}-$. Each certificate has the same period of validity, i.e, $T_{expire} - T_{sign}$. The BS acts as certificate authority (CA) and is pre-distributed with its $K_{BS}+, K_{BS}-, CERT_{BS}$, as well as a public key binding table which consists of IDs and public key of all large nodes in the network.

During inter-cluster communication, each CH must possess a valid certificate in order to interact with other CHs and participate in communication. Before a normal communication between any pair of CHs, it is essential for both nodes to verify each other's certificate and obtain its public key. Then they will generate and distribute a secret session key for this communication.

The BS is responsible for certificate renewal. In order to avoid synchronized certificate renewal requests, we introduce randomization in the timers that they associate with such requests. Instead of requesting certificate renewal exactly before T_{expire}, the node randomly picks up a value $T_{expire'}$ with uniform distribution over $[0.2\times T_{sign}+0.8\times T_{expire}, T_{expire}]$.

In ROSS, the certificate of a convicted malicious large node will be revoked. The BS maintains a certificate revocation list (CRL) to record the revoked certificates. The basic idea is to evaluate the suspiciousness of each large node based on the alarms from neighbor nodes. The large nodes with high degree of suspiciousness will be considered as being compromised. We measure the suspiciousness of a large node with the number of alarms against it. Whenever a neighbor node determines that a particular large node is compromised, it reports an alarm to the BS. Since malicious nodes may report many alarms against benign large nodes, we limit the number of alarms each large node can report to mitigate this effect. The BS maintains an alarm counter and a report counter for each large node. The alarm counter records the suspiciousness of this large node, while the report counter records the number of alarms this node reported and accepted by the BS. For example, if node A perceives a wrong behavior taken by B, it will label B as a suspect node and then sent an alarm about B to BS as follows:

$$A : M = Alarm(B)\|N_A$$

$$A \rightarrow BS : A\|M\|[F(A\|M)]K_A-$$

$$BS : M = QUERY \| N_S$$

$$BS \rightarrow A : BS \| M \| [F(BS \| M)]K_{BS} -$$

$$A \rightarrow B : A \| M \| [F(A \| M)]K_A -$$

$$B \rightarrow T : B \| M \| [F(B \| M)]K_B -$$

$$T : M' = REPLY \| N_S \| \{N_T\}K_{BS} +$$

$$T \rightarrow B : T \| M' \| [F(T \| M')]K_T -$$

$$B \rightarrow A : B \| M' \| [F(B \| M')]K_B -$$

$$A \rightarrow BS : A \| M' \| [F(A \| M')]K_A -$$

Fig. 2. The sequence of authenticate routing packets exchange

Once the BS receives this alarm, it checks if the report counter of A has not exceed a threshold β and B is not revoked. If this is false, the BS ignores this alarm; otherwise, it increases both the alarm counter of B and the report counter of A by 1. Then the BS checks if the alarm counter of B exceeds another threshold m. If yes, the BS will put the certificate of B in its CRL. When one certificate reaches its expiration time, the BS will clear this certificate from its CRL. Note that the alarm from a revoked large node will still be accepted by the BS if its report counter does not exceed threshold β and the target node is not revoked. The purpose is to prevent malicious nodes from reporting a lot of alarms against benign nodes and having these benign nodes revoked before they can report any alarm.

The BS should send the latest CRL to benign large nodes periodically. In order to reduce communication overhead, we extend the aforementioned certificate renewal reply packet by CRL, so the certificate renewal reply packet that the BS generates and sends towards large node B is as follows:

$$BS \rightarrow B : B\|\{CERT_B^{new}\|CRL\}K_B+$$

3.1.2) Authenticated On-Demand Routing Setup: In ROSS, we consider on-demand routing protocols which are preferable to dynamic environment. The widely cited directed diffusion protocol [15] and geographic and energy aware routing protocol (GEAR) [16] are popular representatives of this category, where routing packets should be checked or changed in a hop-by-hop fashion. Digital signature is used to authenticate routing packets. Assuming that the route from the BS to the target CH T is $BS \rightarrow A \rightarrow B \rightarrow T$, Fig. 2 illustrates the format and sequence of authenticated routing packets exchange. The QUERY packet sent by the BS contains a nonce and a signed message digest. Upon receiving the QUERY, the node decrypts and verifies the freshness, origin and integrity of this packet. If all of these verifications are successful, the node will pick the next hop node among its neighbor nodes according to specific routing strategy, modify the packet according to the illustration, and send it to the selected neighbor. On receiving the QUERY packet, the destination T will broadcast it within its cluster, then collect and aggregate the feedback data. After that, T will generate and sign a RELAY packet and unicast it back to the BS by the reverse path. When the route discovery course has completed, the BS and node T will compute a pairwise session key K_{BsT} by executing one-way hash function

$F(N_S||N_T)$ respectively. Since N_T is encrypted with $K_{BS}+$, N_T is secret to all nodes along the route except the BS and T.

3.1.3) Authenticated Data Forwarding: In the packet forwarding phase, T collects and generates DATA packet and then sends it to BS by the reverse path.The DATA packet is conducted as follows:

$$DATA\|N_T\|\ HMAC(K_{BsT}, DATA\|N_T)$$

3.2. Intra-cluster Security Mechanisms

We assume that all sensors and large nodes are predistributed with a network-wide secret key K_s. There is a time delay T_{resist} for capturing a sensor and obtaining the secret key from it. Therefore those nodes which are set up at the same time must be reliable in the time period of T_{resist}. In this initial stage, K_s is used for the establishment of authentication keys that are shared between each CM and its CH, while in the latter stage it is used for authentication local broadcast issued by CH. After the initial stage, each CH will obtain a CMT, which consists of IDs and pair-wise authentication key shared with all valid CMs in its cluster.

We also assume that T is the CH of one sub-region, t is a CM in that region, and the authentication key shared between t and T is K_{Tt}. When t sends the sensing data D to T, T needs to verify D to avoid forwarding the fake or redundant data sent by attackers. Because sensing data are usually periodically sent from t to T, the generation of HMAC for each sensing data is a heavy and time-consuming load for t and not practical for WSN. Therefore, we adopt the concept of one-way hash chain proposed by the μTESLA protocol [4] as session keys. First, t generates a sequence of numbers $n_0^t, n_1^t, n_2^t, ..., n_{k-1}^t, n_k^t$, such that $(n_{i-1}^t) = F(n_i^t)$, where F is a one-way function , $0 < i < k$, and n_k is chosen randomly. Then t sends n_0^t which is encrypted with the pairwise key K_{Tt} to T. When t sends its first D message, that message contains a one-way hash chain number n_1^t. When T receives this message, it verifies the one-way hash chain number in the message by checking if $n_0^t = F(n_1^t)$. If such a match is found, T assumes that the message has been generated from t. T then caches the one-way hash chain number it just received, and process the message; otherwise the message is dropped. When t sends ith D message, it attaches n_i^t. When T gets this message, it will use its cached one-way hash chain number to verify the message.

Furthermore, for those applications that concern about if certain event had occurred, our scheme requires that only when T collects at least $\tau + 1$ legal reports (including itself) which are agreed on the same report should the sensing data D be considered a valid report, otherwise the CH T will simply refuse to forward the report. Here τ is a security threshold based on the security requirements of the application under consideration and the network node density. For example, at least $\tau + 1$ neighbor nodes should agree that the local temperature is higher than $150F$ for a valid report to be sent to the BS. Thus, if $\tau > 0$, an adversary cannot cause a false fire alarm by compromising just one mote. However, this threshold mechanism does not adapt to those applications which are interested in what had happened and require the precise data, such as target localization and tracking.

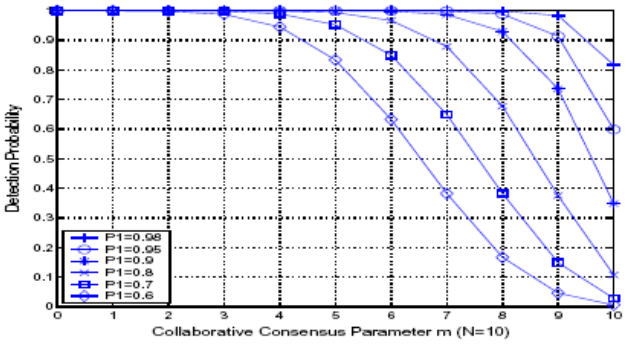

Fig. 3. Increasing detection probability by collaborative consensus

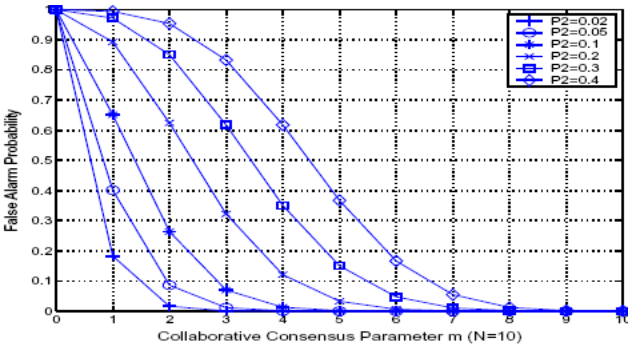

Fig. 4. Decreasing false alarm probability by collaborative consensus

3.3. Analysis of Alarm Threshold

In ROSS, we use threshold (i.e. "m out of N") strategy as the consensus criteria for misbehavior detection. Let P_1 and P_2 denote the detection and false alarm probability of individual monitoring results, respectively. With collaborative consensus, the detection probability is:

$$P_D = \sum_{k=m}^{N} (_k^N) P_1^k (1 - P_1)^{N-k}$$

Meanwhile, the false alarm probability is:

$$P_F = \sum_{k=m}^{N} (_k^N) (1 - P_2)^k P_2^{N-k}$$

The above equations are visualized in Fig. 3 and Fig. 4. By choosing an appropriate value for m, one can increase detection probability P_D and decrease false alarm probability P_F simultaneously. The selection of m represents a tradeoff between the prompt reaction to the attackers and the protection of legitimate nodes from false accusation. We will study the impact of different schemes in future work.

4. Security Analysis and Simulation Results

4.1. Security Analysis

4.1.1) External Attacks: In inter-cluster routing setup phase, one-way hash function and digital signature are exploited to provide authentication, message integrity, and nonrepudiation services. Particularly, the message digest of every

packet is signed and verified in a hop-by-hop fashion during its transmission, so an external attacker cannot fabricate, modify or inject routing information without being detected. Because each packet also contains a nonce, replay attacks can also be detected. In inter-cluster data packets delivery phase, on the other hand, HMAC is used to verify both the integrity and the authenticity of sensing data. In particular, the sender computes an HMAC over the packet with a pairwise session key and includes the HMAC with the packet. HMAC is hard to be forged without the secret key. This implies if an external attacker alters a valid message or injects a bogus message, he cannot compute the corresponding HMAC value, so authorized receivers will reject this message. Therefore, most external attacks against inter-cluster communication are effectively prevented.

During intra-cluster data collection phase, the light-weight one-way hash chain mechanism is adopted as session key. With this mechanism, CHs could check if sensing data packets are from the claimed source node, and if they are modified by other nodes. So external attacks against intra-cluster communication can be detected locally and timely.

4.1.2) Internal attacks: Internal attackers may control everything of the compromised nodes, such as the private keys or secrets shared with other nodes. Therefore, even with cryptographic primitives, the internal attackers can still pass authentication and generate correct digital signature or HMAC for modified or fabricated packets. The security solution should ensure that each node indeed forwards proper packets according to its routing table. In ROSS, this is achieved by the reactive mechanisms.

In the stage of inter-cluster packets delivery, internal attackers are either prevented by periodical certificate renewal, or detected and isolated by collaborative certificate revocation mechanism. On the other hand, in the intra-cluster data collection stage, sensor nodes don't have intrusion detection capability for their limited resources. However, CH should maintain cluster member table in a centralized way and isolate suspicious node by deleting its item in that table.

4.2. Energy Cost Evaluation

Energy consumption is one of the most important factors in sensor network. The two components that make an impact on energy cost are the CPU and the radio.

Asymmetric algorithms (e.g. RSA, ECC) require more computation than equivalently strong symmetric ones. Fortunately, in ROSS, asymmetric cryptography algorithm is used by BS and large nodes in the phases of certificate management and routing setup. Moreover, Wander et al. [17] quantify the energy cost of asymmetric cryptography on an 8-bit microcontroller platform. Their analysis suggests that the use of ECC can lead significant energy savings than RSA. Meanwhile, its smaller keys and certificates lead to significant savings in asymmetric-key communication costs. In general, hash functions (e.g. MD5, SHA-1) are the least complex of the cryptographic algorithms, and should intuitively incur the least energy cost. For example, when implemented on 8 bit ATmegal128L microprocessor, MD5 takes $4.5\mu J/B$ and SHA-1 takes $5.9\mu J/B$ [18]. Any iterative cryptographic hash function may be used in the calculation of an HMAC. However, HMAC is observed to be more compute-intensive than hash algorithms without secret key ($9\mu J/B$ on ATmegal128L microprocessor).

The communication overhead of ROSS comes from both cryptographic mechanisms and reaction mechanisms. In ROSS, the length of nonce, HMAC and signed message digest are 16b, 32b and 64b, thus the overhead of intra-cluster data packet, inter-cluster data packet and routing packet are 32b, 48b and 80b. Now let us consider the energy cost of intra-cluster communication. We assume that the payload of intra-cluster data packet is 29B, because the costs of receiving and transmitting one byte of Mica2 are $28.6\mu J$ and $59.2\mu J$ [18], then receiving one 33B packet (including 4B HMAC) costs $33 * 28.6\mu J = 0.94mJ$ and transmitting one 33B packet costs $33 * 59.2\mu J = 1.95mJ$.

4.3. Simulation Results

In this paper, we use the ns-2 simulator [19] to simulate and compare performance overhead among large nodes with or without security mechanisms. The MAC layer used for this simulation is the distributed coordination function (DCF) of IEEE 802.11, GEAR is adopted as the foundational routing protocol of ROSS, and HMAC is implemented with MD5 algorithm. The simulation parameters are given in Table II.

TABLE II
SIMULATION PARAMETERS

Parameter	Value
Network coverage	(60,60)$\sim (200, 200)m^2$
Base station location	(0, 0)
Large nodes	30 $\sim$ 100
Radio propagation speed	3×10^8 (m/s)
Radio speed	1Mbps
Initial energy	2J
Routing packet size	36B
Data packet size	512B

In the first experiment, we measure average route setup latency and average end-to-end packets transmission delay of GEAR and ROSS. We assume the routing packet processing delay is 2ms. The computing delay of MD5 has been set at 74μs according to [18]. Each session generated 10 data packets of 512 bytes each at the rate of 2 packets per second. The simulation results are shown in Fig. 5 and Fig. 6.

Fig. 5 shows that the average route setup latency for ROSS is approximately double that for GEAR. In comparison with GEAR, while processing ROSS routing control packets, each node has to verify the signature of the previous node, and then generate and replace this with its own signature. Both signature generation and verification cause additional delays at each hop, which increase the latency of route setup.

Fig. 6 shows that the average end-to-end data packet transmission delays of GEAR and ROSS are almost identical. For the reason that the percentage of route control packets is small,

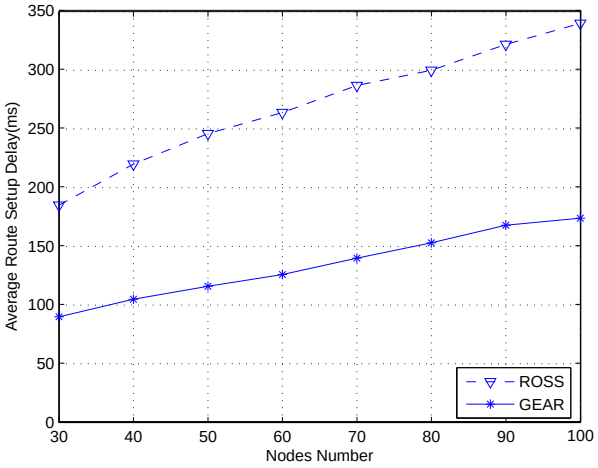

Fig. 5. Average Route Setup Latency

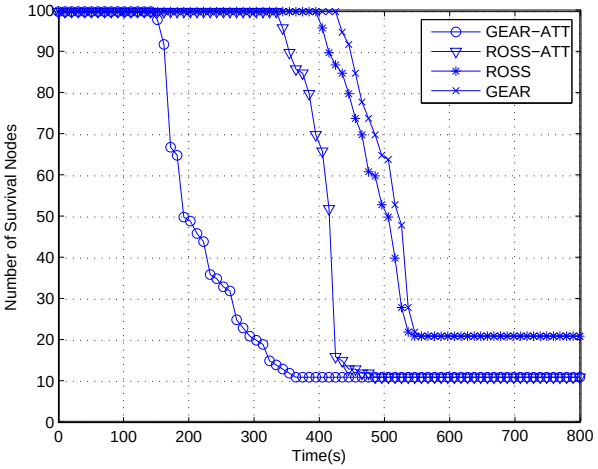

Fig. 7. The relationship between the number of alive nodes and time

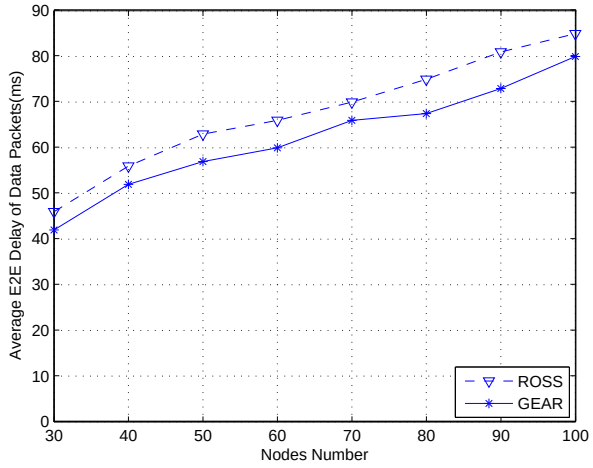

Fig. 6. Average End-to-End Delay of Data Packets

so route setup latency only has little influence on end-to-end data packets delay.

In the second experiment, the network coverage area is fixed as $200m$ by $200m$ square with 100 nodes. This experiment compares the impact of GEAR and ROSS on network lifetime with or without attack traffic. The network lifetime is simply defined as the time elapsed until the last node depletes its energy. The attack traffic is assumed as that there are ten percent of nodes compromised by the attacker, and these compromised nodes as well as three external attackers consume the energy of other nodes by randomly injecting spurious data packets. The relationships between the number of total alive nodes and simulation time are shown in Fig. 7. The result shows that GEAR is the best one with longest network lifetime, but the network lifetime is obviously decreasing when GEAR is under attack (GEAR-ATT). The cryptographic operations of ROSS, such as certificate and key management, encryption primitives, cause some energy consumption. However, the simulation result of ROSS-ATT is much better than GEAR-ATT because ROSS can detect and isolate compromised nodes, and drop abnormal packets as well.

5. Related Work

The main network-layer operations in WSNs are ad hoc routing, data aggregation and data forwarding, which interact with each other and fulfill the functionality of delivering packets from the source to the destination. Up to now, many network-layer security protocols had been proposed for homogeneous sensor networks and mobile ad hoc networks. A thorough analysis of attacks against routing protocols in sensor networks and possible counter measures is given in [3].

Most secure routing protocols [4]- [6] attempt to prevent attackers from launching attacks in the first place, typically through various cryptographic techniques. SPINS [4] addresses secure communication in resource constrained sensor networks, introducing two low level secure building blocks, SNEP and μTESLA. We leverage some of these concepts to implement robust multi-hop routing for HCSNs. In particular, ROSS utilizes keyed message authentication codes (HMAC) similar to SNEP to verify the integrity of packets transmitting among BS and CHs. The concept of one-way hash chain seen in μTESLA is also exploited to provide loose authentication within clusters. Wang et al. [5] focus on secure routing and provide secure directed diffusion(SDD) to securely diffuse data, but SDD has not considered with the in-network aggregation as a goal. Ferreira et al. [6] devise symmetric key based security extensions for LEACH [20], a WSN routing protocol which dynamically and periodically rearranges its clustering.

Mechanisms to overhear neighbor communication in a wireless channel have been used to minimize the effect of misbehaving nodes in mobile ad hoc network [8]- [9]. In the watchdog scheme [8], the sender of a packet watches the behavior of the next hop node for that packet. If the next hop node drops or tampers with the packet, the sender announces it as malicious to the rest of the network. While Watchdog focuses on packet forwarding misbehavior, SCAN [9] aims at monitoring both routing and packet forwarding activities

of each node. Su et al. [7] propose intrusion prevention and intrusion detection approaches for homogeneous clustered sensor networks which rotate the role of cluster head over all the nodes. DICAS is proposed in [10] to mitigate attacks against control traffic by detection, diagnosis, and isolation of malicious nodes in homogeneous sensor network.

Several existing works [11]- [13] have considered security issues in heterogeneous sensor network. In Traynor et al. [11], an unbalanced hybrid pair-wise key establishment scheme is proposed for heterogeneous sensor networks, where security is provided through the leveraging of an distribution of symmetric keys. Oliveira et al. [12] propose solution that relies exclusively on symmetric key schemes and is suitable for networks with an arbitrary number of levels. However, this solution protects the network from external attacks only. Bohge and Trappe [13] propose an authentication framework for a concrete 3-tier network organization. In their solution, only the sensor nodes in the lowest tier do not perform public key operations.

In comparison with the pre-mentioned proposals, ROSS aims to provide robust and efficient security support for the network-layer security issues of HCSNs. It encompasses prevention, detection, and reaction mechanisms to protect control traffic and data traffic from both external attacks and compromised nodes. Security analysis and performance simulation results show that ROSS can effectively resist various attacks with flexible and efficient security mechanisms.

6. Conclusions and Future Work

Most existing security schemes of sensor networks are not suitable for HCSNs because of the unique properties of HCSNs. In this paper, we propose ROSS, a resource oriented network-layer security solution which is customized for HCSNs. ROSS contains proactive cryptography mechanisms and reactive mechanisms to provide a multiple defense solution for the network layer. Both security analysis and performance cost evaluation have confirmed that the effectiveness and efficiency of ROSS meet the predefined standard in protecting the network layer of HCSNs.

However, ROSS is designed for static sensor networks where nodes cannot move. The remaining work in the future is to develop a dynamic mechanism which can adapt to mobile situations.

Acknowledgement

The authors would like to thank the comments provided by the anonymous reviewers and editor, which help the authors improve this paper significantly.

References

[1] M. Batalin, G. Sukhatme, Y. Yu, M. Rahimi, G. Pottie, W. Kaiser, and D. Estrin, "Call and Response: Experiments in Sampling the Environment", in *Proc. of the 2nd ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2004, pp. 25-38.

[2] F. Hu and N. K. Sharma, "Security considerations in ad hoc sensor networks", *Ad Hoc Networks*, vol. 3, issues 1, 2005, pp. 69-89.

[3] C. Karlof and D. Wagner, "Secure Routing in Sensor Networks: Attacks and Countermeasures", *Ad Hoc Networks*, vol. 1, issues 2-3, 2003, pp. 293-315.

[4] A. Perrig, R. Szewczyk, V. D. Culler, and J. D. Tygar, "SPINS: Security Protocols for Sensor Networks", in *Proc. of the 7th Annual ACM International Conference on Mobile Computing and Networks (MobiCom)*, 2001, pp. 521-534.

[5] K. Ren, W. J. Lou, and Y. C. Zhang, "Multi-user broadcast authentication in wireless sensor networks", in *Proc. of the 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, 2007.

[6] A. C. Ferreira, M. A. Vilaca, L. B. Oliveira, E. Habib, H. C. Wong, and A. A. Loureiro, "On the security of cluster-based communication protocols for wireless sensor networks", in *Proc. of the 4th International Conference on Networking(ICN)*, 2005.

[7] C. C. Su, K. M. Chang, Y. H. Kuo, and M. Horng, "The New Intrusion Prevention and Detection Approaches for Clustering-Based Sensor Networks", in *Proc. of the IEEE Wireless Communications and Networking Conference(WCNC)*, 2005, pp. 1927-1932.

[8] S. Marti, T.Giuli, K. Lai, and M. Baker, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks", in *Proc. of the 6th annual ACM/IEEE International Conference on Mobile Computing and Networking(MOBICOM)*, 2000, pp. 255-265.

[9] H. Yang, J. Shu, X. Meng, and S. W. Lu, "SCAN: Self-organized Network Layer Security in Mobile Ad Hoc Networks", *IEEE Journal on Selected Areas in Communications (JSAC)*, Special Issue on Security in Wireless Ad Hoc Networks, vol. 24, issues 2, 2006, pp. 261-273.

[10] I. Khalil, S. Bagchi, and C. N. Rotaru, "DICAS: Detection, Diagnosis and Isolation of Control Attacks in Sensor Networks", in *Proc. of the 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks(SecureComm)*, 2005, pp. 89-100.

[11] P. Traynor, R. Kumar, H. Choi, G. H. Cao, S. Zhu, and P. Thomas, "Efficient Hybrid Security Mechanisms for Heterogeneous Sensor Networkss",*IEEE Transaction on Mobile Computeing*, vol. 6, issues 6, 2007, pp. 663-677.

[12] L. B. Oliveira, H.C. Wong, A.A.F. Loureiro, "LHA-SP: Secure protocols for hierarchical wireless sensor networks", in *Proc. of the 9th IFIP/IEEE International Symposium on Integrated Network Management (IM)*, 2005, pp. 31-44.

[13] M. Bohge and W. Trappe, "An authentication framework for hierarchical ad hoc sensor networks", in *Proc. of the 2003 ACM workshop on Wireless security(Wise)*, 2003, pp. 79-87.

[14] P. Kyasanur and N. Vaidya, "Detection and Handling of MAC Layer Misbehavior in Wireless Networks", in *Proc. of IEEE DSN*, 2003, pp. 173-182.

[15] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks", in *Proc. of the ACM/IEEE International Conference on Mobile Computing and Networking(MOBICOM)*, 2000, pp. 56-67.

[16] Y. Yu, R. Govindan, and D. Estrin, "Geographical and Energy Aware Routing: a Recursive Data Dissemination Protocol for Wireless Sensor Networks", Tech. Rep. UCLA/CSD-TR-01-0023, Computer Science Department, University of California at Los Angeles, May 2001.

[17] A. Wander, N. Gura, H. Eberle, V. Gupta, and S. Shantz, "Energy Analysis of Public-Key Cryptography for Wireless Sensor Networks", in *Proc. of 3rd IEEE International Conference on Pervasive Computing and Communication*, 2005, pp. 324-328.

[18] P. Ganesan, R. Venugopalan, P. Peddabachagari, A. Dean, F. Mueller, and M. Sichitiu, "Analyzing and Modeling Encryption Overhead for Sensor Network Nodes", in *Proc. of ACM International Workshop on Wireless Sensor Networks and Applications(WSNA)*, 2003, pp. 151-159.

[19] The Network Simulator ns-2, At: www.isi.edu/nsnam/ns/.

[20] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks", in *Proc. of IEEE Hawaii Int. Conf. on System Sciences*, 2000, pp. 4-7.

Xiaomei Cao received the B.E. degree in computer science from Henan University, Kaifeng, China, in 1996, and the M.S. degree in computer science from East China Normal University, Shanghai, China, in 2000.

She is currently a Ph.D. candidate in Nanjing University, Nanjing, China. Her research interests are in wireless networks, sensor networks, and security.

Guihai Chen obtained his BS degree from Nanjing University, M. Engineering from Southeast University, and PhD from University of Hong Kong. He visited Kyushu Institute of Technology, Japan in 1998 as a research fellow, and University of Queensland, Australia in 2000 as a visiting professor. During September 2001 to August 2003, he was a visiting professor in Wayne State University. He is now a full professor and deputy chair of Department of Computer Science, Nanjing University. Prof. Chen has published more than 100 papers in peer-reviewed journals and refereed conference proceedings in the areas of wireless sensor networks, high-performance computer architecture, peer-to-peer computing and performance evaluation. He has also served on technical program committees of numerous international conferences. He is a member of the IEEE Computer Society.