SOS: Secure Overlay Sensornets

Leonardo B. Oliveira[1,3], **A. A. F. Loureiro**[2], **Ricardo Dahab**[1]

[1]UNICAMP - Brazil, [2]UFMG - Brazil, [3]Supported by FAPESP under grant 2005/00557-9

leob@ic.unicamp.br,loureiro@dcc.ufmg.br, rdahab@ic.unicamp.br

***Abstract*— Overlay Networks (ONs) are logical networks built on top of a physical network with the aim of moving part of the routing complexity to the application layer. At the same time, sensornets are ad hoc networks comprised mainly of small sensor nodes with extremely limited resources which can be used to monitor areas of interest. In this paper, we present Secure Overlay Sensornets (SOS). SOS builds an ON over a sensornet, and it establishes and monitors alternative overlay routes. By doing so, SOS is able to find out routes more secure than routes provided by the default routing protocol. Our results indicate that SOS improves the delivery ratio in scenarios under DoS attacks and that it is efficient in terms of energy consumption. To our knowledge, SOS is the first security mechanism based on ONs for sensornets.**

I. Introduction

Overlay Networks (ONs) are logical networks built on top of a physical network with the aim of moving part of the routing complexity to the application layer [1]. Based on a given criteria, ON are able to provide alternative routes to users. Such routes are built by means of overlay nodes that act as intermediaries in the transmission of data. Today, ONs have been used to simplify network management, e.g., in security and fault management.

On the other hand, sensornets – or wireless sensor networks (WSNs) – are ad hoc networks comprised mainly of small sensor nodes with limited resources and one or more base stations (BSs) [2]. WSNs are used for monitoring purposes in different application areas, ranging from battlefield reconnaissance and emergency rescue operations to surveillance and environmental protection.

Like any wireless ad hoc network, WSNs are vulnerable to attacks [3], [4]. Besides the well-known vulnerabilities due to wireless communication and "ad hocness", WSNs face additional problems. For instance, sensor nodes are small, cheap devices that are unlikely to be made tamper-resistant or tamper-proof. Also, they are often deployed in unprotected, or even hostile areas, which makes them more vulnerable to attacks. It is therefore crucial to add security to these networks, specially those that are part of mission-critical applications.

Our goal is to evaluate ONs for mitigating effects caused by Denial of Service (DoS) attacks in hierarchical WSNs. Instead of just using multiple routes, ONs monitor alternative routes and generate statistics of, e.g., route reliability. These statistics are then used by the current application to on-demand choose "best" routes, i.e., routes that best matches the requirements of the application. By employing the information provided by ONs in WSNs that are under DoS attacks, we expect to bypass compromised routes and achieve a higher delivery ratio at the BS. To do that, we propose **S**ecure **O**verlay **S**ensornets (SOS), a secure routing mechanism for WSNs. SOS builds an ON over a WSN, establishes and monitors alternative overlay routes, and looks for routes more *secure*[1] than those provided by the default routing protocol. These routes, in turn, are used by nodes to send messages that carry sensitive information. To our knowledge, SOS is the first security mechanism based on ON for WSNs. In this work, we focus on hierarchical and heterogeneous WSNs because, when compared to flat WSNs, they present advantages as increased system throughput and energy savings [5].

This paper is organized as follows. In Section II, we briefly discuss organization and security of hierarchical WSNs. In Section III, we present SOS. We describe how SOS was evaluated in Section IV and present results in Section V. Finally, we discuss related work in Section VI, and conclude in Section VII.

II. Hierarchical WSNs: Organization and Security

WSNs may be organized in different ways. In *flat* WSNs [6], all nodes play similar roles in sensing, data processing, and routing. In *hierarchical* WSNs [2], on the other hand, the network is typically organized into clusters, with ordinary cluster members and the cluster heads (CHs) playing different roles. While ordinary cluster members are responsible for sensing, the CHs are responsible for additional tasks such as collecting and processing the sensing data from their cluster members, and forwarding the results towards the BS.

Like any WSN, hierarchical WSNs are vulnerable to a number of attacks [3], [4] including jamming, spoofing, and replay. In these networks, attacks involving CHs are particularly damaging, because CHs are responsible for critical functions such as data aggregation and routing. If an adversary manages to become a CH, it can stage attacks such as blackhole [4] and selective forwarding [3], thus disrupting potentially large fractions of the network. Adversaries may also leave the routing alone, and try to inject bogus data into the network; or they may choose to simply eavesdrop on communication between legitimate nodes, obtaining information that is being gathered by the BSs.

At a high level, the main security goals in a hierarchical WSN are: 1) access control, i.e., allow only legitimate nodes to take part in the network (e.g., become CHs and join a cluster); 2) guarantee the authenticity, confidentiality, integrity and freshness of data being passed from one member of the

[1]by secure we mean routes that are not under DoS attacks

network to another; and 3) guarantee availability (minimize the impact of attempts of DoS attacks). In this work, we design our solution to mitigate effects caused by DoS attacks and therefore it meets the third goal.

III. SOS

In this section we present SOS. Our goal is to route important messages through more secure routes thus increasing their delivery ratio[2].

1) Assumptions: 1) We consider hierarchical and heterogeneous networks in which nodes are static and the BS is trusted. 2) Communication is *single-hop* within the clusters (i.e., from children to CH) and *multi-hop* among CHs. The BS, in contrast, can communicate directly with nodes. 3) There are two classes of messages: *priority messages* (PMs) and *nonpriority messages* (NMs). Whereas NMs are messages that carry information not sensitive (*nonpriority data*) and, if lost, do not cause great damage to the network functioning, PMs are important messages that carry sensitive information (*priority data*). The classification is carried out between collection and transmission of data. It is performed by the same node that collected the data based on their relevance. 4) *Attackers* are interested in preventing collected data from reaching the BS. They are able to compromise network nodes and thus launch DoS attacks. 5) Nodes have been deployed, granularity of data report has been specified, and the default routing protocol has already been established.

2) Overview: SOS builds an ON over a WSN as follows. Some or all nodes belonging to the underlying WSN are chosen to make part of the ON. To each of these nodes, called *overlay nodes*, is assigned a list of logical neighbors. The neighbors, called *overlay neighbors*, are used as intermediaries for establishing routes alternative to that provided by the routing protocol (*default route*). Overlay nodes then alternate the transmission of messages via the routes (including the default one) equally. The BS thus computes the routes delivery ratio based on the last t time units. The routes with the highest rates are, obviously, considered the *best routes* (the most secure ones) and the BS informs the overlay nodes which routes these are. Fig. 1 presents a diagram of an ON over a WSN. Note that the source (SRC) node relies on the overlay routes in addition to the default route.

A. Protocol Description

1) Setup: To build the ON, the BS first establishes the connectivity graph of the network. To do that, nodes may, e.g., send a broadcast searching for neighbors [7]. As soon as nodes receive neighbors answers, they forward this information to the BS. The BS can thus build the graph.

Next, the BS determines which nodes will take part in the ON and assigns each a list of overlay neighboring nodes chosen based on connectivity graph information. These neighbors are used as intermediaries in data transmission over alternative routes – also called *overlay routes*. In other words, each overlay neighbor corresponds to the first hop of an alternative route. In Fig. 1, overlay neighbors are those connected by logical links. Note that the criteria for choosing overlay neighbors may vary depending on the application. However, as most of the time WSNs focus on minimizing energy consumption – which is a quadratic function of communication range –, we believe that overlay neighbors of a given node should be chosen among nodes in its vicinity.

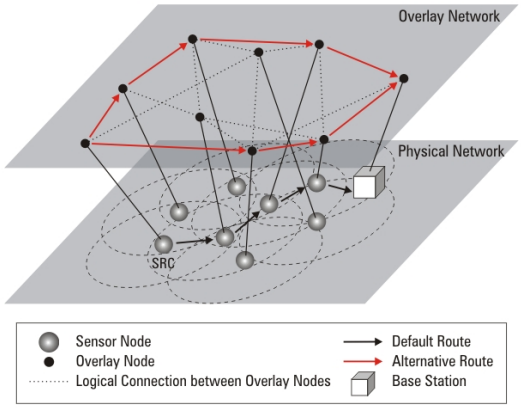


Fig. 1. SOS example diagram: an ON over a WSN

2) Collecting and Announcing Statistics: The network's sensing phase is then initiated and sensor nodes start sending PMs and NMs according to the relevance of the collected sensor data. A naive strategy for collecting statistics about routes is overlay nodes to generate monitoring messages and transmmit them to the BS through the different routes (i.e., default and overlay routes) from time to time. Note, however, that it would result in a high communication overhead. Our strategy, therefore, is to overload the function of NMs and use them not only to send nonpriority data, but also to monitor routes. So, every overlay node alternates the transmission of these messages among its default route and its overlay routes in a circular fashion (Table I). The PMs, on the other hand, are sent and forwarded via the default route while the best routes are still unknown, i.e., while the first monitoring results are not divulged by BS (Table I).

Two flags are added to each message by the source node to identify: 1) the class of the message, i.e., whether it is a PM or a NM; 2) the route (default, overlay #1, overlay #2,...) used to send the message. The flags are latter used by the BS to calculate the delivery ratio.

As soon as the BS receives the first messages it creates a register. The register contains NM delivery ratio for all the nodes and it is organized by node and route. The delivery ratio is calculated by taking into account the number of messages sent and received in the last t time units. The BS infers the number of NMs sent based on both the granularity of the data report and the fact that transmissions of NMs are evenly distributed among the routes. The calculation of the number of messages received is clearly easier, i.e., the BS just needs to identify the node source address of the message and the route

[2]Here, we consider message delivery ratio as the ratio of the number of messages delivered to the BS, to the number of messages originally sent to it

by which the message was sent used (by using the second flag). Note that PMs are not taken into consideration, since they are not sent alternately among the routes. Table II shows a register for a fictitious network comprising 3 nodes and 2 overlay routes per node. For node 1, e.g., the NM delivery ratios of its default, overlay #1, and overlay #2 routes are 76%, 54%, and 65%, respectively. The highlighted routes are the best.

The BS then informs the overlay nodes which the best routes are. These routes are used by nodes to send and forward future PMs, with the aim of increasing the delivery ratio of these messages. Concerning the NMs, however, the procedure is not altered (Table I), as routes are required to be monitored constantly. Note that the best route information may vary with time. In this case, the BS sends a message to the nodes affected containing updated information about their respective best routes. Also note that no message replication occurs, i.e., SOS does not employ redundancy to monitor routes.

msg class	best routes unknown		best routes known	
	send	forward	send	forward
PM	default	default	best route	best route
NM	alternately	default	alternately	default

TABLE I

SUMMARY OF ROUTING SOS POLICIES

node	route delivery ratio		
	default	overlay #1	overlay #2
1	76%	54%	65%
2	70%	78%	73%
3	31%	50%	39%

TABLE II

3-NODE, 2-OVERLAY-ROUTES REGISTER

3) Discussion: As most of the solutions devised to WSNs, SOS targets a specific problem, which here is to increase delivery ratio in scenarios under DoS attacks. And, as opposed to a number of cryptographic proposals (e.g., [8]–[12]), SOS does not address confidentiality and authentication among nodes. So, if SOS is used as-is, it is possible to adversaries: 1) take advantage of the best route information to infer the routes by which subsequent PMs will travel and then compromise these messages; and 2) use the flag that identifies the message class to target PMs, specifically. We note, however, that SOS must be used in conjunction with a cryptographic proposals (e.g., [8]–[11], [13]) to prevent these types of attacks.

IV. SIMULATION

To evaluate our proposal, we compared a sensing application running alone to the same application running together with SOS. We will refer to these instances as Plain and SOS networks, respectively, or *Plain* and SOS for short. The comparison was done using the *Network Simulator 2* (ns-2). Below, we describe how the simulation was conducted and present parameters and metrics considered.

In the beginning, we consider that nodes were randomly deployed in an area of interest and that mechanisms for key exchange among neighbors (e.g., LEAP [9]) and clustering around near CHs (e.g., LEACH [14]) were performed. To each sensor node, the default route towards the BS was computed through a protocol similar to the TinyOS beaconing routing protocol [15], which creates a minimum distance tree rooted at the BS.

In SOS, apart from the CHs that reach the BS directly, all other CHs take part in the ON. Common nodes are not included in the ON due to efficiency, i.e., to prevent the BS from the burden of disseminating route states to the entire network and to prevent ordinary nodes from receiving messages about route states. Two overlay neighbors were assigned to each overlay node X. These neighbors were chosen among the physical neighbors of X that did not belong to its default route. To prevent loops, the chosen overlay neighbor cannot have as its default first hop the node X, and nodes record the IDs of recent handled messages and do not forward those IDs that have already been handled.

The radio transmission range of all nodes is 100m and each ordinary node originates one message per period of 10s. This message is sent to the nearest CH, that in turn aggregates its children´s reports into a single message and forwards the message to the BS via its neighboring CHs. We assume 36-byte packets (the default packet size of TinyOS [15]) in SOS and 30-byte packets in Plain. This difference of 6 bytes is relative to the introduction of a Message Authentication Code (MAC). Actually, we consider a 8-byte MAC [16], but its introduction makes unnecessary a 2-byte Cycle Redundancy Check (CRC).

To estimate the energy consumption, we assume the same radio energy model used in LEACH [14]. In this model, a radio dissipates ϵ_r = 50 $\mathrm{nJ/bit}$ to run the transmitter or receiver circuitry, and ϵ_a = 100 $\mathrm{pJ/bit/m^2}$ for the transmitter amplifier. Also, the radios expend the minimum required energy to reach the recipients and are turned off to avoid receiving unintended transmissions. An δ^2 energy loss due to channel transmission is assumed as well. Under this model, the costs to transmit ($\mathcal{E}_T$) and receive ($\mathcal{E}_R$) a β-bit message at distance δ are given respectively by:

$$\mathcal{E}_T(\beta, \delta) = \beta\,\epsilon_r + \beta\,\delta^2\,\epsilon_a$$
$$\mathcal{E}_R(\beta) = \beta\,\epsilon_r$$

We consider two types of DoS attacks, namely, *blackhole* (BH) and *selective forwarding* (SF). In the former attack, a compromised node ceases to forward any messages [3]; and, in the latter attack, a node only forwards some messages and discards the others [3]. During the simulation, nodes under attacks also cease to collect and send data. Attacks were launched as soon as nodes were deployed and organized themselves into clusters. Compromised nodes were randomly picked out among the CHs since – as discussed in Section II – attacks staged to this class of nodes may disrupt larger fractions of the network. That is true that attacks targeted to nodes in the BS's vicinity could be more disruptive and then could have a higher chance to occur. However, exactly because these attacks focus on nodes near the BS – region which can be seen with naked eyes by users at the BS –, we believe

they are also more difficult to be launched. Concerning SF, in particular, the discard ratio was 50%, i.e., compromised nodes forwarded 50% of the received messages and discarded the remaining ones.

To assess how different parameters impact SOS in terms of security and energy efficiency, we varied the following parameters in our simulation: percentage of PMs, percentage of compromised CHs, and network size (in terms of nodes and maintaining the network density and percentage of CHs constant). While we varied a parameter, the others were held constant. Metrics were also chosen, namely, PM and NM delivery ratios (gain), energy consumption (cost), and energy consumption per PM delivered (efficiency relative to PM).

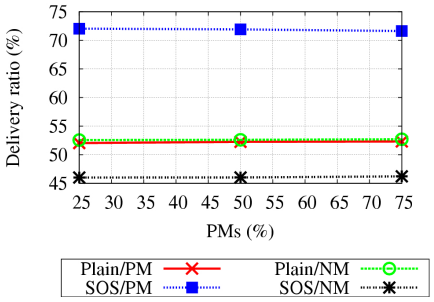
Finally, simulation time was 1000s and nodes have been endowed with sufficient energy to be operational during the entire simulation.

V. Results

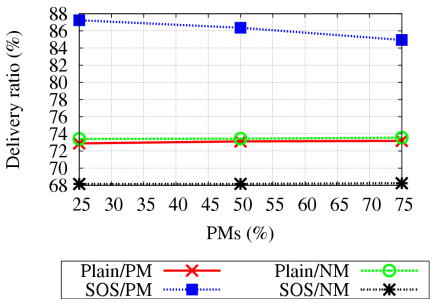
In this section we present the simulation results and the overhead incurred by SOS when compared to the plain network. As described in Section IV, three parameters were considered: percentage of PMs; percentage of compromised CHs; and network size. The energy overhead is presented in tables. For each type of attack, these values are discriminated by CH (*CH*), entire network (*overall* – including both CHs and ordinary nodes), and PM delivered (*per PM*). Note that the latter also measures the SOS efficiency. Since the CHs took part in the ON (as mentioned in Section IV), the energy overhead incurred by ordinary nodes was essentially due to the addition of MACs and, in turn, it was around 20% for all scenarios. For that reason, we decided to omit them in the tables. Finally, only noncompromised nodes were taken into account to calculate the energy overhead.

Fig. 2 presents, for Plain and SOS scenarios, the PM and NM delivery ratios when 25%, 50%, 75%, and 100% of the messages were PMs. Under BH attacks (Fig. 2(a)), the SOS's PM gain (difference between the PM delivery of SOS and the PM delivery of Plain) was stable and around 19.5% for the different PM percentages. The SOS's NM gain was also constant as the proportion of PMs increased, but conversely, it presents negative values (between -6.5% and -6.6%.), i.e., the NM delivery ratio of Plain was higher then that of SOS. The reasons for this result are twofold. Firstly, as in Plain, NMs in SOS are sent out without taking into account any statistics. Secondly, the average route length in SOS is higher than Plain average route length. (Note that to be assigned to a node, overlay neighbors are picked out among the node's physical neighbors. However, physical neighbors are not necessarily positioned toward the BS. They can also be in the same level of the routing tree of the node or even one hop backwards, which in turn increases the route length in one and two hops, respectively.) This in turn increases the chance of a message to pass through a compromised node.

Apart from SOS's PM delivery ratio, that decreases about 2.5% as the percentage of PMs increased, the delivery ratio curves under SF attacks (Fig. 2(b)) also presented stability. In general, the difference between the values for SOS and Plain



(a) **BH**



(b) **SF**

Fig. 2. PM and NM delivery ratios vs. percentage of PMs.

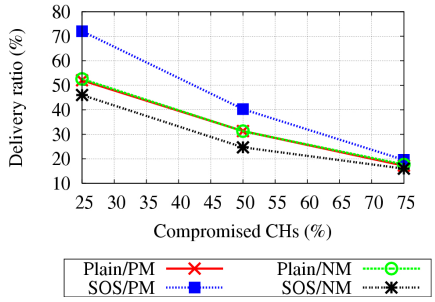
were smaller and the delivery ratios were higher compared to scenarios under BH attacks. This is because compromised nodes in these scenarios only discarded 50% of the messages that passed through them (as shown in Section IV). The decrease in the delivery ratio of SOS is because the higher the number of PMs, the lower the number of NMs. And this, in turn, decreases the frequency of route monitoring. As one of the central ideas of ON is route monitoring, this has influenced negatively the SOS performance.

priority messages	BH			SF		
	CH	overall	per PM	CH	overall	per PM
25%	46%	26%	-9%	55%	28%	7%
50%	50%	27%	-8%	63%	30%	1%
75%	52%	27%	-7%	63%	30%	12%

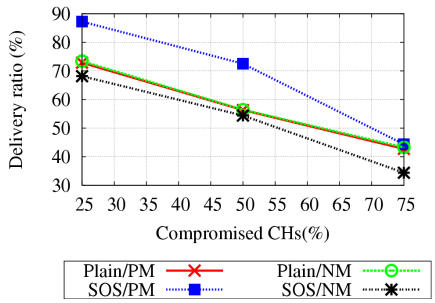
TABLE III
Energy overhead vs. percentage of PMs.

Table III shows the energy overhead incurred by SOS vs. percentage of PMs. In general, the overhead increased as the percentage of PMs increased and it was higher in scenarios under the SF attack. This is because the large the percentage of PMs, the larger the absolute number of messages that take advantage of SOS. This increases the network traffic and as consequence the energy consumption. Also, even though the *CH* overhead has achieved considerable values (e.g., 63%), the *overall* overhead was between 26% and 27% for scenarios under BH attacks, and 28% and 30% for scenarios under SF attacks. Finally, due to the higher PM delivery ratio of SOS, the overhead values *per PM* were the lowest, even achieving negative values under BH attacks.

Fig. 3 presents the PM and NM delivery ratios vs. the percentage of compromised CHs. As expected, the ratios were

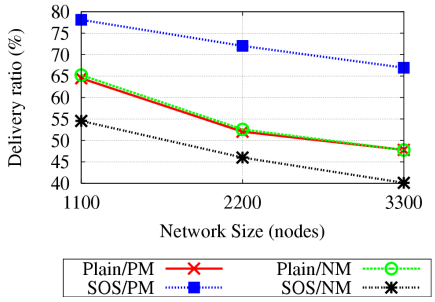


(a) **BH**

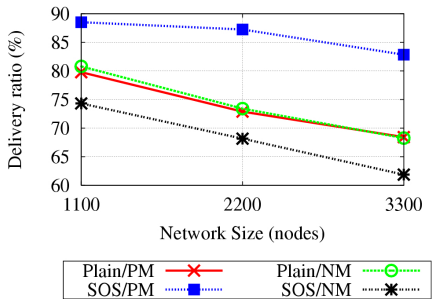


(b) **SF**

Fig. 3. PM and NM delivery ratios vs. percentage of compromised CHs.



(a) **BH**



(b) **SF**

Fig. 4. PM and NM delivery ratios vs. network size.

inversely proportional to the percentage of compromised CHs for both networks and both types of attack. Also, SOS's PM gain over Plain presented a similar behavior for the two attacks (Figs. 3(a) and 3(b)). It was considerable when 25% (20% and 14.4% for BH and SF, respectively) and 50% (9% and 16.2%, for BH and SF, respectively) of the CHs were compromised and negligible when 75% (9% and 1.6% for BH and SF, respectively) were compromised. This result indicates that if a great fraction of the network nodes is compromised, it is difficult even for SOS to discover noncompromised routes. Concerning the NMs as whole, the delivery ratio of SOS again turned out to be lower than that of Plain.

compromised nodes	**BH**			**SF**		
	CH	**overall**	**per PM**	**CH**	**overall**	**PM**
25%	46%	26%	-9%	55%	28%	7%
50%	28%	21%	-6%	49%	25%	-8%
75%	22%	20%	6%	43%	23%	12%

TABLE IV

ENERGY OVERHEAD VS. PERCENTAGE OF COMPROMISED CHS.

Table IV shows the energy overhead incurred by SOS vs. percentage of compromised CHs. As the percentage of compromised nodes increased, the *overall* overhead decreased in all scenarios. This is because the difference between the delivery ratios – and therefore the network traffic – of SOS and Plain has also decreased (Fig. 3). In general, the overheads present similar behavior as when varying the percentage of PM. Most of them were higher under SF attacks than under BH attacks and the *per PM* overhead sometimes achieved negative values.

Fig. 4 presents the PM and NM delivery ratios vs. network size. It shows that all the delivery ratios decreased as the network size increased. This is also because of the average route length, which becomes larger and increases the chance of the message to pass through a compromised node. Concerning SOS's PM gain over Plain, specifically, it was higher (about 20% under BH attacks and 14% under SF attacks, as shown in Figs. 4(a) and 4(b), respectively) under medium (2200 nodes) and large (3300 nodes) network sizes. This is because larger networks can offer more alternative routes. And, as a result, an ON can select the best routes from a wider range of options. SOS's NM gain over Plain, conversely – due to reasons already mentioned above –, once more presented negative values, namely -10.7%, -6.5%, and -7.6% under BH (Fig. 4(a)), and -6.5% -5.3% -6.4% under SF (Fig. 4(b)), for 1100, 2200, and 3300 nodes, respectively.

Table V shows the energy overhead incurred by SOS vs. network size. In fact, there are many variables that govern the behavior of overhead as the network size increases (e.g., the average route length and traffic increases, and the percentage of nodes that are neighbor of the BS decreases) and it is difficult to determine precisely which were predominant in each scenario; but, as a rule, the overhead was dictated by the amount of traffic, i.e., the difference between the delivery ratio of SOS and Plain (Fig. 4). Note that the higher this difference, the smaller the *overall* overhead. The overhead *per PM*, as expected, presented an opposite behavior, and decreased as the difference increased.

VI. RELATED WORK

WSNs are a subclass of MANETS and much work (e.g., [17]) has been proposed for securing routing in MANETs as a whole. These studies are not applicable to WSNs because they either assume laptop- or palmtop-level resources – which

network size	BH			SF		
	CH	overall	per PM	CH	overall	per PM
1100	38%	23%	6%	46%	25%	15%
2200	46%	26%	-11%	55%	28%	7%
3300	42%	25%	-13%	53%	29%	5%

TABLE V
ENERGY OVERHEAD VS. NETWORK SIZE.

are orders of magnitude larger than those available in WSNs –, or do not take into account the asymmetric many-to-one routing of WSNs. This motivated the advent of a number of studies specifically targeted to security of resource-constrained WSNs (e.g., [8]–[13]). For instance, Karlof and Wagner [4] and Wood and Stankovic [3] have both focused on attacks and vulnerabilities. Stadon *et al.* [7] proposed an efficient algorithm for detecting failed nodes and bypass information through alternative routes. Perrig *et al.* [16] offered a solution for flat and homogeneous networks based on pairwise key sharing between each of the nodes and the BS. SPINS, as their solution is called, is a suite of symmetric key based protocols for providing baseline security (confidentiality, authentication, integrity, freshness) and authenticated broadcast.

A subset of these proposals offers security and/or reliability through multiple routes. Ganesan *et al.* [18] proposed a multi-route version of the Directed Diffusion protocol [19]. Regularly, redundant messages are sent via alternative routes to check whether they are still operational and, whenever a fault in the default route occurs, these routes are used. The goal of the work is to improve the fault tolerance of Directed Diffusion and it does not address the issue of compromised nodes. Deng *et al.* [20] proposed INSENS, in which a node always sends a message through more than one route. Lou *et al.* [21] split a message in multiple shares through secret sharing schemes. Each share is sent via an independent route, so that even if a small number of message were compromised, the secret as a whole is kept confidential. Note that all these works employ some degree of redundancy to deliver messages. And, although this increases the chance of a given message reach its destination, this also incurs overhead in energy consumption. Still, none of the works effectively monitor alternative routes, as an ON does. Finally, message classification is not used, and a message containing priority data has the same chance to be compromised as any other message.

VII. CONCLUSION

In this paper we present a secure routing mechanism for WSNs called **S**ecure **O**verlay **S**ensornets, or SOS. SOS builds an ON over a WSN and establishes and monitors alternative overlay routes to identify routes which are more secure than those provided by the default routing protocol. These routes in turn are employed by CHs to send messages that carry sensitive information. We also evaluated the costs, benefits, and scalability of our solution under a variety of scenarios. For each scenario, we considered the BH and SF attacks. The results showed that SOS not only improves the delivery ratio in scenarios under DoS attacks, but also that it is efficient in terms of energy consumption.

REFERENCES

[1] David Andersen, Hari Balakrishnan, Frans Kaashoek, and Robert Morris. Resilient Overlay Networks. In *The 8th ACM Symposium on Operating Systems Principles (SOSP'01)*, pages 131–145, Banff, CA, Oct. 2001.

[2] Deborah Estrin, Ramesh Govindan, John S. Heidemann, and Satish Kumar. Next century challenges: Scalable coordination in sensor networks. In *Mobile Computing and Networking*, pages 263–270, Seattle, WA USA, 1999.

[3] Anthony D. Wood and John A. Stankovic. Denial of service in sensor networks. *IEEE Computer*, 35(10):54–62, October 2002.

[4] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: Attacks and countermeasures. *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, 1(2–3):293–315, 2003. Also appeared in 1st IEEE International Workshop on Sensor Network Protocols and Applications.

[5] E. Melo and M. Liu. The effect of organization on energy consumption in wireless sensor networks. In *IEEE Globecom*, 2002.

[6] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114, August 2002.

[7] J. Staddon, D. Balfanz, and G. Durfee. Efficient tracing of failed nodes in sensor networks. In *the 1st ACM international workshop on Wireless sensor networks and applications*, pages 122–130. ACM Press, 2002.

[8] Laurent Eschenauer and Virgil D. Gligor. A key management scheme for distributed sensor networks. In *9th ACM conference on Computer and communications security (CCS'03)*, pages 41–47. ACM Press, 2002.

[9] Sencun Zhu, Sanjeev Setia, and Sushil Jajodia. LEAP: efficient security mechanisms for large-scale distributed sensor networks. In *10th ACM conference on Computer and communication security*, pages 62–72. ACM Press, 2003.

[10] Donggang Liu, Peng Ning, and Rongfang Li. Establishing pairwise keys in distributed sensor networks. *ACM Transactions on Information and System Security (TISSEC)*, 8(1):41–77, 2005. Also appeared in 10th ACM CCS '03.

[11] W. Du, J. Deng, Y. S. Han, P. K. Varshney, J. Katz, and A. Khalili. A pairwise key pre-distribution scheme for wireless sensor networks. *ACM Transactions on Information and System Security*, 2005. Also appeared in 10th ACM CCS '03.

[12] Leonardo B. Oliveira, Hao Chi Wong, M. Bern, R. Dahab, and A. A. F. Loureiro. SecLEACH – a random key distribution solution for securing clustered sensor networks. In *5th IEEE International Symposium on Network Computing and Applications*, 2006. 145-154.

[13] Leonardo B. Oliveira, Hao Chi Wong, Ricardo Dahab, and Antonio A. F. Loureiro. On the design of secure protocols for hierarchical sensor networks. *International Journal of Networks and Security (IJSN)*, 1(2):–, 2006. Special Issue on Cryptography in Networks, to appear.

[14] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *IEEE Hawaii Int. Conf. on System Sciences*, pages 4–7, january 2000.

[15] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, Alec Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler. TinyOS: An operating system for wireless sensor networks. In W. Weber, J. Rabaey, and E. Aarts, editors, *Ambient Intelligence*. Springer-Verlag, 2004.

[16] Adrian Perrig, Robert Szewczyk, Victor Wen, David Culler, and J. D. Tygar. SPINS: Security protocols for sensor networks. *Wireless Networks*, 8(5):521–534, September 2002. Also appeared in MobiCom'01.

[17] Lidong Zhou and Zygmunt J. Haas. Securing ad hoc networks. *IEEE Network*, 13(6):24–30, 1999.

[18] Deepak Ganesan, Ramesh Govindan, Scott Shenker, and Deborah Estrin. Highly-resilient, energy-efficient multipath routing in wireless sensor networks. In *2nd ACM international symposium on Mobile ad hoc networking & computing (MobiHoc'01)*, pages 251–254, New York, NY, USA, 2001. ACM Press.

[19] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In *6th annual international conference on Mobile computing and networking*, pages 56–67, Boston, MA, 2000.

[20] Jing Deng, Richard Han, and Shivakant Mishra. A performance evaluation of intrusion-tolerant routing in wireless sensor networks. In *IPSN*, volume 2634 of *Lecture Notes in Computer Science*, pages 349–364, Palo Alto, CA, 2003. Springer.

[21] Wenjing Lou, Wei Liu, and Yuguang Fang. SPREAD: Enhancing data confidentiality in mobile ad hoc networks. In *INFOCOM*, 2004.