GoNe: dealing with node behavior

Sabrina Sicari*, Alessandra Rizzardi*, Luigi Alfredo Grieco§, Alberto Coen-Porisini*
*Dipartimento di Scienze Teoriche e Applicate, Università degli Studi dell'Insubria, v. Mazzini 5 - 21100 Varese (Italy)
§Department of Electrical and Information Engineering, Politecnico di Bari, v. Orabona 4 - 70125 Bari (Italy)
Email: {sabrina.sicari; alessandra.rizzardi; alberto.coenporisini}@uninsubria.it, a.grieco@poliba.it

***Abstract*—The detection of malicious nodes still represents a challenging task in wireless sensor networks. This issue is particularly relevant in data sensitive services. In this work a novel scheme, namely GoNe, is proposed, able to enforce data security and privacy leveraging a machine learning technique based on self organizing maps. GoNe provides an assessment of node reputation scores on a dynamic basis and in presence of multiple kinds of malicious attacks. Its performance has been extensively analized through simulations, which demonstrate its effectiveness in terms of node behavior classification, attack identification, data accuracy, energy efficiency and signalling overhead.**

***Index Terms*—Wireless Sensor Network, Security, Reputation**

I. Introduction

A Wireless Sensor Network (WSN) is a distributed system which acquires, stores, and processes information by leveraging multihop wireless connectivity. Due to the nature of the radio channel and the remote deployment of the nodes, a WSN is exposed to different kinds of attacks, including the violation of data, and the injection of network failures (denial of service, clone, sybil). Once an attack is detected, its action has to be mitigated and any ill-behaved node has to be isolated. Many proposed solutions perform an assessment of the nodes, based on trust and reputation systems [1] [2] [3] [4].

According to [2] and [4] as well as authors knowledge, no existing work addresses the detection of multiple kinds of attacks, through a clever analysis of node behaviors, while guaranteeing the security and privacy of data. Besides security, WSN are highly constrained in terms of power resources; regarding this topic, many solutions [5] [6] adopt data aggregation for reducing the amount of transmitted information and avoiding network congestion. Good Network (GoNe) is proposed hereby in order to deal with the presented issues. It detectes and isolates malicious nodes on a dynamic basis, along and guarantees confidentiality, integrity and anonymity of the information. Moreover, GoNe adopts the privacy-aware data aggregation scheme presented in [7] in order to address congestion control. The reputation technique proposed with GoNe is based on Self Organizing Maps (SOM) [8], also known as Kohonen network, which is an architecture conceived for unsupervised neural networks. It presents many advantages since it requires no supervision, it is relatively fast and not expensive, also with a high dimensionality of data.

To demonstrate the effectiveness of GoNe, an extensive simulation campaign has been carried out, revealing that GoNe exhibits encouraging key performance indexes, such as false positive/negative rate, data accuracy, energy consumption, and packet delay. The rest of the paper is organized as follows: Sec. II describes the target scenario; Sec. III introduces the GoNe protocol; Sec. IV presents simulation scenarios and results, while Sec. V ends the paper and draws future research.

II. Target Scenario

The reference scenario is a clustered WSN, based on a wireless multihop mesh backbone. Each cluster is made of a variable number of sensors and one mesh router acting as cluster head (CH), directly or indirectly connected to the sink through the backbone. Sensor nodes are constrained in terms of energy and processing resources, while CHs can be assumed to be grid powered (or with a huge energy availability) and able to run more complex algorithms. For this reason, sensor nodes will only perform the sensing and the data encryption activities; whereas CHs verify the integrity of the received data and, in case of no violation, aggregate the data according to the congestion level of the network, as proposed in SETA [7]. Note that the CHs implement the secure aggregation scheme also for data coming from different clusters. Finally, the sink assesses the reputation score of WSN nodes and detects misbehaving ones. From such distinctions, it emerges that nodes are characterized by different functions and roles, following the model presented in [9]. Each node (sensor or CH) has a set of keys used according to the current function-role couple [7] (the key distribution scheme is out of the scope of this paper).

The messages containing the sensed data are denoted by $m_{n,q}$ where n indicates the node that generated and transmitted the message; whereas q uniquely identifies the message among those generated by n. $L_n$denotes the list of the nodes which forward the data towards the sink. The integrity of transmitted data, also encrypted, is object of a malicious attack, which should modify the value of the sensed or aggregated data. A countermeasure is represented by the adoption of a hashing procedure. The hash of the encrypted sensed or aggregated data is calculated by the sensor nodes or the CHs and, then, also the hash is encrypted, to add another security level and to avoid attacks which can modify both the hash and the related data [7]. The CHs use such a hash to perform the data integrity verification. Concerning secure aggregation, sensor nodes adopt homomorphic stream ciphers [10] which allow the CH to aggregate data without deciphering them. The approach presented in this paper adopts data aggregation at CH level to avoid traffic congestion: when the transmission queue builds up, data therein are aggregated to keep the queue length under its maximum limit. Note that violated messages are not considered in the aggregation process. More details about sensing, integrity verification and data aggregation are available in [7].

III. GoNe

A. Inputs

GoNe aims not only at identifying data violations, but also at detecting the malicious nodes. To this end, a machine learning engine is used, which allows to isolate misbehaving nodes by evaluating their reputation. Such a reputation is a value in the range [0,1], where 0 is the lowest value (no confidence is associated to the node), while 1 is the highest value (there is a complete confidence in the node behavior). To evalutate the reputation of a node, besides the case of violation of the content of the packets, its behavior is analyzed, in relation to network and computing resource usage. As regards network usage, the inspected features are: number of message received/ generated/ forwarded/ dropped by a node; average packet arrival time; number of messages received by specific neighbor nodes. They allow to monitor the unusual traffic in a given neighborhood. As regards computing resource, the monitored features are memory and CPU utilization. Such information are gathered by the nodes themselves, which periodically send to their CH a packet with the following fields:

$$p_{s,q} = (n_{s,q},\ P_r,\ P_g,\ P_f,\ P_d,\ Mem,\ Cpu), \text{ where:}$$

$n_{s,q}$ is the couple (n_s, q_s), in which n_s identifies the sensor node that generated the packet, while q_s identifies such a message among those transmitted by n_s (this field is kept unchanged among transmissions); P_r is the number of packets received by n_s until the instant t_n, in which such a message was generated; P_g, P_f, P_d are the number of packets generated, forwarded, dropped, respectively, by n_s until t_n; Mem is the percentage of filling of the node buffer at t_n; Cpu is the CPU utilization until t_n, measured in MIPS (Million Instructions Per Second).

In order to guarantee the node anonymity and the confidentiality and integrity of the transmitted information, all the fields contained in $p_{s,q}$ are encrypted with a group signature [11] shared only with the sink; such a scheme allows the group (the nodes which belong to the network) to sign the messages on behalf of the group without revealing node identity; only the group manager (the sink) can open the signature and trace the original signer. Another parameter considered by the sink is the average packets arrival time of the nodes, indicated as $P_{avg}[k]$, where *k* represents the number of nodes in the considered cluster. All these information, along with the acquired data, are continuosly received by the sink through the CHs and sent to the *Machine Learning Engine*, which is the responsible of the classification and reputation score evaluation of the nodes (Fig. 1).

B. Machine Learning Engine

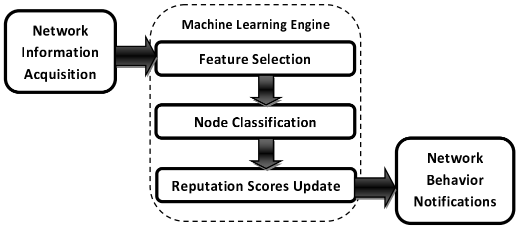

Fig. 1. Reputation Score Mechanism Scheme

The *Machine Learning Engine* is composed by: a mechanism of features selection able to process the parameters sent by the CH; a system for node behavior classification, which assesses for each node if it is normal, malicious or unknown; a module in charge of updating the scores associated to the nodes on the basis of the output of the classification phase. It is implemented using SOM algorithm and is able to organize various features in an internal representation; the input layer takes such features ($p_{s,q}$ vectors), as input signals for the neurons. In the input layer each neuron is directly connected to all the neurons in the output layer; the ouput layer represents the reputation score update (Fig. 1). At each algorithm iteration the weights among input and output neurons are updated; such an adjustement is a linear combination of input vector and current weight vector, as showed by the scoring function (Eq. 1):

$$W(t+1) = W(t) + R(t)(V(t) - W(t)), where: \quad (1)$$

W is the weight, *t* represent the instant time, *R* is a positive number less than 1, named *learning rate* (which decreases with time), and *V* is the current input vector. The weights are initialized with random values between 0 and 1. From the weights computation, a reputation score rep_s is derived for each sensor node. In the initial phase of the network, rep_s are set to 0.5, which is the average value between the two limits. Nodes are classified in three categories: *Normal* when rep_s is greater than 0.6, *Unknown* when rep_s is in the range [0.4, 0.6], *Malicious* when rep_s is less than 0.4. The classification ranges [0; 0.4) - [0.4; 0.6] - (0.6; 1] have been determined through simulations, which demonstrated that such ranges optimize the node classification in terms of false positive/negative rate (Sec. IV-A).

C. Outputs

At each iteration of the classification phase, once the sink notices relevant updates in the node reputation, it informs its cluster members about the changes of confidence towards the nodes, in order to isolate misbehaving sensors and preserve the network from data corruption and waste of resources. The sink adopts a proper type of message to notify the CHs and, consequently, the CHs have to inform the nodes of their cluster about the reputation score updates; the message sent by the CHs to the nodes has the following form:

$$sr = (r_{i,q},\ repList[n_s][rep_s]), \text{ where:}$$

$r_{i,q}$ is the couple (r_i, q_i), in which r_i identifies the CH which generated the packet, while q_i identifies such a message among those transmitted by r_i(this field is kept unchanged among transmissions); $repList[n_s][rep_s]$ is a list of couples node-reputation, in which n_s represents the identifier of a node belonging to the CH cluster; while rep_s is the reputation score associated with the node. Note that CHs are considered trusty nodes.

n_s has been just encrypted by the sink with the group signature and forwarded by the CH guaranteeing the end-to-end anonymity. Each sensor node stores the retrieved information in its own local table T, which aims at containing the couples node-reputation of all the nodes belonging to the same cluster. Moreover, sensor node verifies the reputation scores stored in T regarding the nodes which are in the field L_n of each received message. Also the nodes identifiers contained in L_n are encrypted with the group signature, therefore the nodes belonging to the network are able to establish the associated score without decrypting the identifiers. In case of nodes classified as *Normal* and/or *Unknown*, the node processes the packet in the standard way. Note that, in presence of nodes classified as *Unknown*, the CH does not aggregate the data, in order to preserve the data accuracy, and forwards it to the sink, which will decide whether to use it or not. Whereas, if almost a node in the fields L_n is identified as *Malicious*, then the packet is immediatly dropped, as well as all the other kinds of message sent by the malicious nodes, in order to isolate them from the network. The scope of the learning algorithm is to minimize, if not avoid, the nodes classified as *Unknown*.

A key characteristic of SOM is that the neighborhood nodes participate in the process of adaptation (learning). For this reason, SOM finds application in many contexts, such as recognition and identification (medical diagnosis, face recognition), data mining, monitoring and control (e-mail spam filtering, vehicle control), forecasting and prediction (financial applications). In this paper a hybrid architecture is adopted in order to exploit the advantages of SETA in terms of congestion control and end-to-end security; however this reputation mechanism is also suitable for a flat architecture.

IV. Performance evaluation

This section compares GoNe with SETA [7] and *Verifiable Multilateration* (VM) [12] schemes. It will be shown that GoNe, like SETA, guarantees: privacy, security, congestion control, data accuracy, as well as acceptable delay and power consumption. In addition, GoNe, being able to detect malicious nodes, enhances network performances in terms of resilience to malicious attacks and allows a better classification of sensor nodes with respect to VM. GoNe is analyzed in terms of:

- Data accuracy, estimated by means of a comparison between the environmental temperature estimated by the sink and the actual temperature (the temperature is a double in the range [20, 30] generated following the model adopted in SETA)
- Delay of packets arrival, which represents the time elapsed between the packet generation at a sensor node and its reception at the sink
- Node power consumption, estimated using Energino models [13]
- Overhead due to the reputation algorithm in terms of percentage of signalling messages with respect to the total messages transmitted by the network
- Lost messages: the aim is to estimate how GoNe responds to certain traffic network attacks, avoiding message losses in comparison to SETA, which does not cope with such malicious behaviors
- Number of nodes correctly classified as *Normal*, *Unknown* and *Malicious* with respect to the secure localization protocol VM; such an evaluation points out also the false positive/negative rate
- Detection time: the dynamics of node scores is analyzed too, in order to shed further light on the GoNe behavior
- Evaluation of the best intervals of scores to correctly classify nodes as *Normal*, *Unknown* and *Malicious*.

TABLE I
Simulation parameters

Parameter	*Description*	*Value*
N	Number of nodes	100, 200
C	Number of clusters	3
D_c	Depth of connections	5
M	Percentage of malicious nodes	up to 40%
P	Interval time of data generation	1s, 2s
P_{Max}	Max packet size	93 bytes
br	Bit rate	250 kbps
C_m	Cluster Head (CH) buffer size	20 kB
N_m	CH percentage of buffer size emptying	90%
Q_n	Node buffer size	10 KB
t_S	Duration of simulation	1800 s

To evaluate the performance, the Omnet++ simulator is used [14]. Parameters and simulated scenarios are summarized in Tab. I. In order to exploit the header compression gain due to 6LoWPAN standard [15], messages are encapsulated in a IPv6 over IEEE 802.15.4 stack [16]. Several models of attacks are inserted, such as attacks to data integrity, to resources (DoS attacks), to routing behavior (wormhole attacks) [17] [18]. The outcomes are presented for different percentages of malicious nodes (up to 40% of the total nodes, and, when not specified, with a percentage of 20%).

A. Simulation results

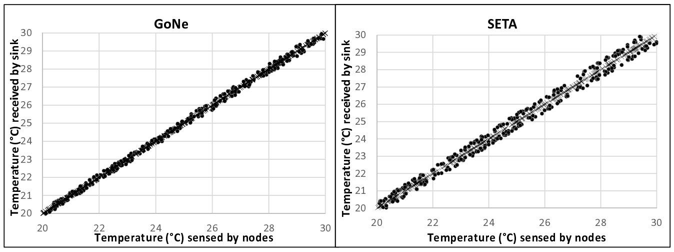

Fig. 2. Data accuracy

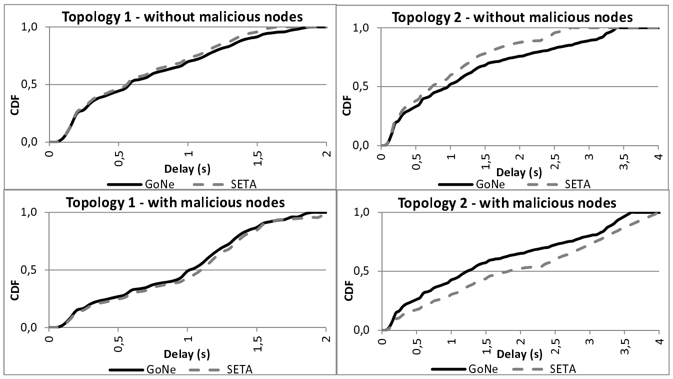

Fig. 3. CDF of packet delay

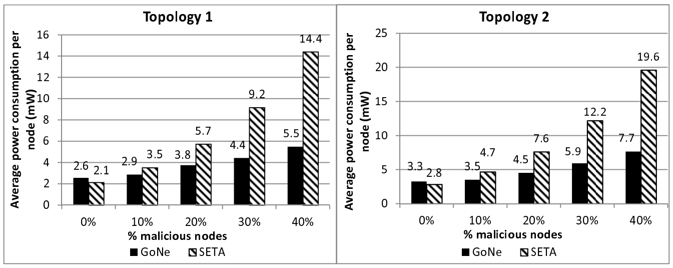

Fig. 4. Mean energy consumption of sensor nodes

As regards the accuracy of the data received by the sink (Fig. 2), for both topologies (100 and 200 nodes), varying the percentage of malicious nodes, GoNe and SETA grant a high level of data quality. This result was expected since GoNe extends SETA functionalities, which already provides accurate measurements. Without the presence of malicious nodes, GoNe introduces a slightly higher delay in comparison with SETA (Fig. 3) due to signalling messages. Whereas, in presence of malicious behavior, GoNe provides equal or smaller delays with respect to SETA, thanks to its ability to isolate malicious nodes. Analogous considerations hold for the mean energy consumption (Fig. 4).

Fig. 5 shows the effects of the reputation mechanism in terms of percentage of messages related to the score evaluation with respect to the total number of packets transmitted over the network. It emerges that (i) without malicious nodes the overhead is concentrated at the beginning of the simulation; (ii) with the increase of malicious nodes the peak overhead is lowers, but its long term value is higher than before, since the reputation algorithm needs a certain time to recognize the malicious behavior. Since SETA does not face attacks to traffic or resources, an higher percentage of lost packets with respect to GoNe is expected (Fig. 6). Figs. 7 and 8 show that GoNe remarkably reduces the number of nodes classified as *Unknown* with respect to VM. Moreover, GoNe is able to identify the kinds of attack (data integrity, network resources, routing protocols), whose percentages are shown in Fig. 10; they respect the percentage of malicious behaviors included in the simulation scenarios (approximatively, 40% integrity attacks, 30% attacks to resources, 30% attacks to routing protocols).

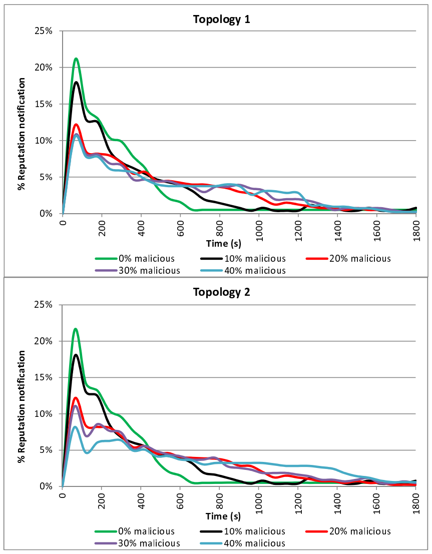

Fig. 5. Overhead of GoNe

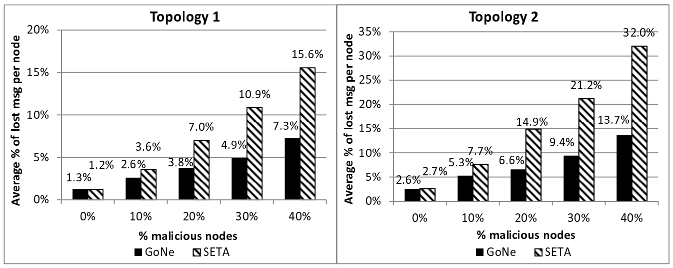

Fig. 6. Lost packets

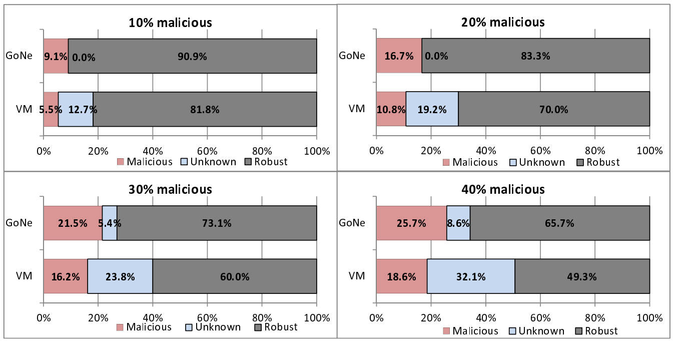

Fig. 7. Node classification - Topology 1

In Fig. 9 the transient score behavior associated to a normal and a malicious node is analyzed. The algorithm is able to quickly recognize if a node presents malicious behavior or not and to gradually stabilise the associated score. Finally, Fig. 11 analyzes the thresholds tested for correctly classify the nodes, showing that the best ones are those used for all the presented simulations: 0.4 and 0.6.

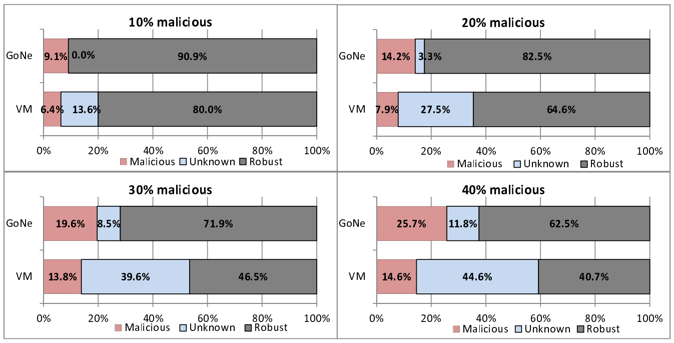

Fig. 8. Node classification - Topology 2

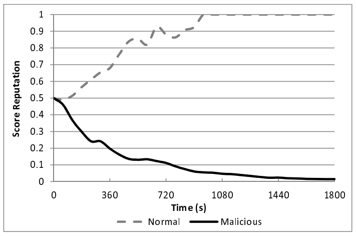

Fig. 9. Evaluation of node reputation

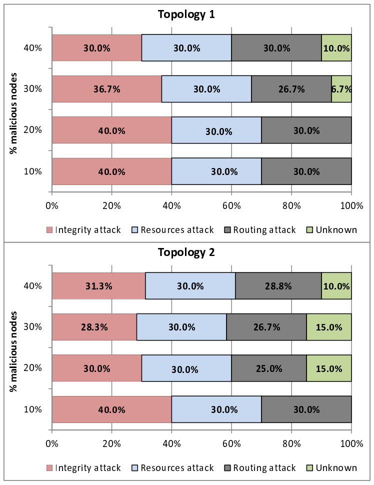

Fig. 10. Attacks evaluation

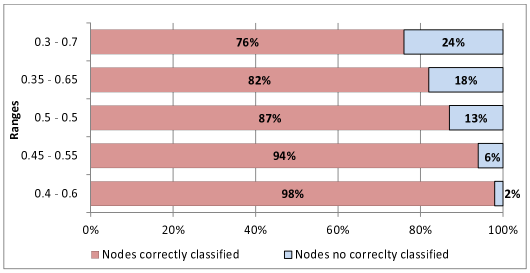

Fig. 11. Reputation range evaluation

V. Conclusion

GoNe protocol has been presented to identify malicious attacks towards data integrity, network resources and routing protocols. With respect to previous works, it addresses both privacy and security issues, adopting a node reputation assessment scheme. GoNe reduces the node energy consumption since the reputation algorithm does not generate a high overhead and, isolating malicious nodes, it limits the percentage of lost packets. For the future, we are planning the integration of GoNe in an Internet of Things framework.

References

[1] Z. Bankovic, D. Fraga, J. M. Moya, J. C. Vallejo, P. Malagón, Álvaro Araujo, J.-M. de Goyeneche, E. Romero, J. Blesa, D. Villanueva, and O. Nieto-Taladriz, "Improving security in wmns with reputation systems and self-organizing maps," *Network & Computer Applications*, vol. 34, no. 2, pp. 455–463, 2011.

[2] G. Han, J. Jiang, L. Shu, J. Niu, and H.-C. Chao, "Management and applications of trust in wireless sensor networks: A survey," *Computer and System Sciences*, vol. 80, no. 3, pp. 602–617, 2014.

[3] F. G. Mármol and G. M. Pérez, "Towards pre-standardization of trust and reputation models for distributed and heterogeneous systems," *Computer Standards & Interfaces*, vol. 32, no. 4, pp. 185–196, 2010.

[4] Y. Yu, K. Li, W. Zhou, and P. Li, "Trust mechanisms in wireless sensor networks: attack analysis and countermeasures," *Network & Computer Applications*, vol. 35, no. 3, pp. 867–880, 2012.

[5] X. Xu, R. Ansari, and A. Khokhar, "Power-efficient hierarchical data aggregation using compressive sensing in wsns," in *IEEE International Conference on Communications*, June 2013, pp. 1769–1773.

[6] C. Zhao, W.Zhang, X. Yang, Y. Yang, and Y. Song, "A novel compressive sensing based data aggregation scheme for wireless sensor networks," in *IEEE International Conference on Communications*, June 2014, pp. 18–23.

[7] S. Sicari, L. Grieco, A. Rizzardi, G. Boggia, and A. Coen-Porisini, "SETA: A secure sharing of tasks in clustered wireless sensor networks," in *IEEE WiMob*, Lyon, France, Oct. 2013.

[8] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, Sep. 1990.

[9] S. Sicari, L. A. Grieco, G. Boggia, and A. Coen-Porisini, "DyDAP: A dynamic data aggregation scheme for privacy aware wireless sensor networks," *Journal of Systems and Software*, vol. 85, no. 1, pp. 152–166, 2012.

[10] C. Castelluccia, E. Mykletun, and G. Tsudik, "Efficient aggregation of encrypted data in wireless sensor networks," in *Mobile and Ubiquitous Systems: Networking and Services*, 2005, pp. 109–117.

[11] D. Chaum and E. Van Heyst, "Group signatures," in *10th Annual International Conference on Theory and Application of Cryptographic Techniques*, Berlin, Heidelberg, 1991, pp. 257–265.

[12] A. Coen-Porisini and S. Sicari, "Improving data quality using a cross layer protocol in wireless sensor networks," *Computer Networks*, vol. 56, no. 17, pp. 3655–3665, 2012.

[13] K. Gomez, R. Riggio, T. Rasheed, D. Miorandi, and F. Granelli, "Energino: A hardware and software solution for energy consumption monitoring," in *Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks*, May 2012, pp. 311–317.

[14] "Omnet++ simulator," 2014, accessed: 2015-01. [Online]. Available: http://www.omnetpp.org/

[15] J. Hui and P. Thubert, "Compression format for ipv6 datagrams over ieee 802.15.4-based networks," IETF RFC 6282, Tech. Rep., Sep. 2011.

[16] M. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, L. Grieco, G. Boggia, and M. Dohler, "Standardized protocol stack for the internet of (important) things," *Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1389–1406, Mar. 2013.

[17] S. M. Zin, N. B. Anuar, M. L. M. Kiah, and A.-S. K. Pathan, "Routing protocol design for secure wsn: Review and open research issues," *Network and Computer Applications*, vol. 41, no. 0, pp. 517–530, 2014.

[18] Y. Yu, K. Li, W. Zhou, and P. Li, "Trust mechanisms in wireless sensor networks: Attack analysis and countermeasures," *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 867–880, 2012.