SETA: A SEcure sharing of TAsks in clustered wireless sensor networks

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

***Abstract*—Secure data aggregation still represents a very challenging topic in wireless sensor networks' research. In fact, only few solutions exists to face, simultaneously, confidentiality, integrity, adaptive aggregation, and privacy issues. Furthermore, proposals available in literature mainly assume flat network architectures, without leveraging the peculiarities of clustered wireless sensor networks, which are very common in real life deployments. This work tries to bridge the gap by proposing a solution, namely SETA, tailored to a hybrid architecture composed of wireless sensor nodes and wireless mesh routers as cluster heads. In SETA, to lower the processing workload of sensor nodes, only cluster heads are allowed to perform integrity verification checks and message merging operations to face possible network congestions. To prove its effectiveness, it has been compared, using simulations, with respect to DyDAP in several realistic settings. Results have shown that it can provide a slight improvement of the robustness to malicious nodes and of the sensing accuracy, while increasing the overall energy efficiency and decreasing the signalling overhead in the network.**

I. Introduction

Wireless Sensor Networks (WSNs) represent one of the most successful technology of the ICT domain in the last decade [1]. They are composed by a set of sensor nodes that: sense the environment, perform basic processing functions, establish network paths, and report collected data to one or more sinks by means of multi-hop wireless communications.

In this context, the wireless nature of the communication channel and the remote access increase the risk of attacks that can lead to the violations of privacy, integrity, and confidentiality of exchanged data. In particular, two common threats may occur [2], [3]:

- *Eavesdropping*, that is, a malicious user could easily discover the communication content by listening to the messages exchanged among nodes.
- *Masking*, that is, a malicious node may mask its real nature behind the identity of a node (that is authorized to take part to the communication), trying to misroute messages.

At the present, to preserve privacy in a WSN, the available solutions may be classified into two main groups: (i) anonymity mechanisms based on data cloaking [2], [4]; (ii) approaches based on privacy policy [5]. The former schemes are based on perturbing data following some kind of criterion [2], [4][6]-[8]. Instead, the latter approaches state who can use individuals data, which data can be collected, for what purpose they can be used, and how they can be distributed [5], [9], [10].

Moreover, security solutions in this field should consider that WSN applications require the collection of a large amount of data; thus, due to the limited power resources of sensor nodes, it is necessary to aggregate such data in order to reduce the amount of transmitted information [11].

Secure data aggregation in WSN is a very mature research field and the literature reports many solutions addressing aggregation issues and security aspects (an exhaustive and very comprehensive view of this topic can be found in [12]). The approaches proposed so far can be classified into two big families depending on whether a hop-by-hop or an end-to-end data cryptography is used. Hop-by-hop encryption is usually based on symmetric key schemes, which demand less computing resources than the ones using asymmetric keys. These algorithms (e.g., see [13]-[21]), require that each aggregator node should decrypt every message it receives in order to allow in-network processing; obviously, this causes a confidentiality breach. Furthermore, the application of several consecutive encryption/decryption operations, along the path from source to sink, can negatively impair latencies and energy consumption. Finally, hop-by-hop aggregation requires that each node shares secret keys with all its neighbors. To face these problems, aggregation algorithms able to work on ciphered data, using either asymmetric or symmetric keys [22]-[28] have been proposed. But, such solutions have the main limitation that they allow the use of only very simple aggregation functions, like the sum and the average [12].

Despite this very broad variety of proposals [29], only in [11] a solution (namely DyDAP) has been conceived to address, at the same time: confidentiality, integrity, adaptive aggregation, and privacy issues. Unfortunately, DyDAP has been mainly conceived for flat network architectures made by nodes with homogeneous capabilities. In reality, in WSNs nodes have different capabilities (e.g., memory, processor, power consumption, transmission range, and so on) and many of these networks are build upon a hierarchical clustered infrastructure where, at the the lowest layer, each cluster of nodes is grounded to a Cluster Head (CH), whereas at the highest one, the CHs form a high speed wireless mesh backbone to convey messages to the sink [30]. To overcome this limitation, we propose in

this work an approach based on a new algorithm, i.e., the SEcure sharing of TAsks (SETA), which leverages the different energy and computational capabilities of sensors and CHs. In SETA, sensor nodes will only perform the sensing and the encryption of the data; while CHs will verify the integrity of the received data and, in case of no secure violation, aggregate the data according to the congestion level of the network. In this manner, more complex operations are accomplished only by nodes with more capabilities extending, at the same time, the lifetime of the whole network.

The main features that characterize and distinguish SETA with respect to DyDAP, highlighting the novelty of the present proposal, are:

1) the adoption of a hybrid architecture, more suitable for real environments and WSN applications where nodes have different capabilities. Sensor nodes are members of clusters communicating to CHs, whereas mesh routers are CHs communicating with the sink. The defined architecture allows to make a strict task sharing among network nodes (sensors and mesh routers) in order to overcome the limited power resources of sensor nodes.
2) The definition of a new protocol for the data integrity verification, based on hash functions. It allows to reduce DyDAP overhead due to integrity check messages. In fact, in SETA the use of both hybrid architecture and hashing technique allow the assignment for the integrity Controller role [31] only to mesh routers. Notice that in DyDAP all sensor nodes perform the Controller role and then many messages are exchanged, with an overhead and a related power consumption.

It is worth to note that our work provides a contribution in the field of privacy by defining a role-based context-independent solution that guarantees anonymity of nodes before data are collected into a database. Thus, the solution we are going to propose may be combined with both data cloaking mechanisms and some other privacy policy based approaches.

The effectiveness of SETA has been compared with respect to DyDAP algorithm by using the Omnet simulator[32] in several realistic settings. In particular, the evaluation of the power consumption in transmission and reception has been considered. Results have shown that SETA is able to reduce the communication overhead of DyDAP while, at the same time, it provides a slightly better sensing accuracy and robustness to malicious nodes.

The rest of the paper is organized as follows.

Section II describes the reference scenario and the foundations on which SETA is based. Section III presents SETA in details, while Section IV reports the simulation results that demonstrate the effectiveness of SETA. Finally, Section V draws some conclusions and provides hints for future works.

II. Reference Scenario

A. Network Architecture

The SETA algorithm entails a clustered WSN (see Fig. 1) where every cluster is made of several sensor nodes grounded at a CH. The CHs are connected to each other forming a wireless multi-hop mesh backbone and deliver data to the sink. The key aspect of SETA is to perform a sharing of tasks, based on the different energy and computational capabilities of sensors and CHs. In fact, the former are very constrained in terms of energy and processing resources. The latter, instead, can be assumed to be: (i) grid powered (or with a huge energy availability) and (ii) able to run more complex algorithms [33]. For this reason, in SETA sensor nodes will only perform the sensing and the data encryption; whereas CHs will verify the integrity of the received data and, in case of no secure violation, aggregate the data according to the congestion level of the network. The CHs implement the secure aggregation scheme also for data coming from different clusters. In this way, it will be possible to face: sensor energy consumptions, network congestions, privacy, and security issues.

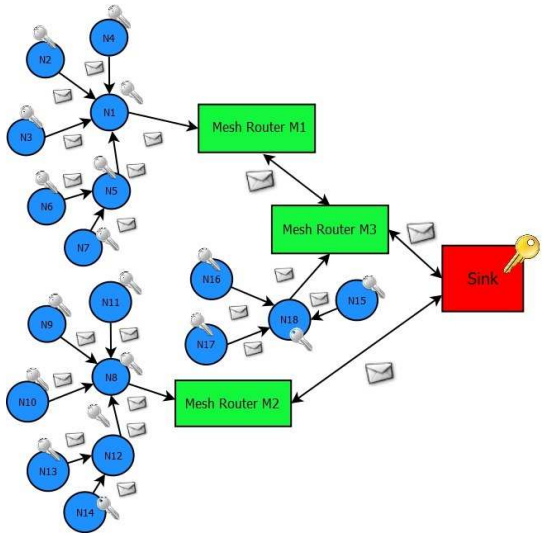

Fig. 1. Reference network model in SETA.

Concerning secure aggregation, sensor nodes adopt homomorphic stream ciphers, such as the one presented by Castelluccia et al. [24]. They allow the CH to aggregate data without deciphering them. After the aggregation process, the CH generates the Aggregated Message (AMEX), which properly contains the aggregated data.

We assume that nodes are aware of their own geographical position and that they access the wireless channel by using a CSMA-like MAC protocol [34]. The broadcast nature of wireless channel allows a node to determine, by overhearing the channel, if its messages are received and forwarded by its neighbors [35].

Moreover, we assume that in nodes and CHs, messages, before their transmission, may be queued in a buffer. Whenever the message input rate at a node (obtained as the sum of the local and transit traffic rates) exceeds the message forwarding rate at MAC layer (which depends on the MAC protocol, the radio channel, and the network topology), messages are queued in such a transmission buffer. Therefore, if no countermeasure is taken upon the saturation of the message forwarding rate, the transmission queue increases up till its maximum limit and, as a consequence, there are message losses due to traffic congestion. These losses have two negative effects: (i) they waste energy resources; (ii) they can severely impair the estimation accuracy of the WSN.

The approach presented in this paper exploits 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. In order to identify the list of the nodes whose encrypted sensed data belong to, the AMEX message has in its header a list of the identifiers (ids) of the nodes involved in the monitoring process. By means of the node ids, the sink is able to associate the right node key which must be used for the decryption process.

The entire procedure, exploited to securely aggregate and convey the samples collected by the sensor nodes to the sink, can be decomposed into the following steps:

1) Sensor nodes generate their encrypted data headed for the sink, sending them to a CH.
2) Each CH receives packets coming from sensor nodes in its cluster and from other CHs. Received packets are stored in a local queue and then the integrity of the data contained in the packets is verified. In case of privacy violation, a notification error is sent to the sink. Otherwise the data are aggregated according to the congestion level of the networks and the CH generates a AMEX packet sent to the sink.
3) The sink receives all the AMEX packets, deciphers ciphered data, and computes the more relevant statistical information (e.g., average and/or variance), with a high assurance level about the accuracy of the received information, thanks to the exploitation of the integrity violation notification.

B. Keys, functions, and roles

Encryption and decryption operations follows from the privacy model presented in [31]. Each node in the network (sensor or CH) represents a member of the network either interested in processing data or involved by this process. In particular, nodes are characterized by *functions* and *roles*. The former represents the *task* performed by a node in the network where it operates (e.g., data sensing, message transmission, message forwarding, data aggregation, and so on); whereas the latter is a key concept [36] to characterize sensor nodes and CHs with respect to the privacy feature.

In detail, each sensor node plays two different roles: *Subject*, to represent nodes that sense data, and *Processor*, to represent nodes that process data by performing some kind of action on them (e.g., transmission, forwarding.). Instead, each CH, a part from performing the role of *Subject* when generates/aggregates data and of *Processor* when performing actions like transmission and forwarding, could play also the role of *Controller* when it verifies the actions executed by processors. Notice that a sensor node and a CH can play more than one role.

In the considered scheme, each node(sensor or CH) owns a hashing key (HK) which allows it to verify the integrity of the received data. Moreover, each node has a set of keys that are used according also to the considered function-role couple. Summarizing, there are four type of keys according to the function-role couple or the hash function: Sensing-Subject (SS), Transmitter-Processor (TP), Notifier-Controller (NC), and a HK.

The sink owns the key sets of all the nodes of the whole network. It is important to remark that the key distribution method is out of the scope of this work.

III. The SETA Framework

Herein, we presents the integrated framework SETA whose goal is to provide an end-to-end secure data aggregation scheme with privacy capabilities. More specifically, its objectives are: (i) data integrity; (ii) anonymity; (iii) energy efficient usage of the WSN; and (iv) end-to-end secure data aggregation. Thus, in what follows we provide an thorough discussion about the structure of exchanged messages, the protocols, and the algorithms that SETA exploits to meet the previous mentioned goals.

A. Message structure

In SETA, a message is the basic application unit exchanged by the nodes of the network. In what follows, exchanged messages 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 the node n.

According to the model described in [31], messages may contain data that can be further classified as: *identifiable* data (they include the information useful for identifying a node) and *sensed* data (they include all information sensed by nodes).

A sensed data, before reaching the sink, passes through different nodes of the network (i.e., multi-hop communication) by means of different messages. The integrity of the transmitted data, also encrypted, should be the object of a malicious security attack, which should modify the value of the sensed or aggregated data. To overcome this issue, a countermeasure is represented by the adoption of a hashing procedure. In our solution the hash of the encrypted sensed or aggregated data is calculated by the sensor nodes or the CH and, then, also the hash is encrypted, to add another security level and to avoid attacks that modify both the hash and the related data. The encrypted hash information is transmitted in the header of the message, allowing the verification of the message integrity, as explained in details below.

A SETA message $m_{n,q}$ can be defined as a tuple in which all fields, unless otherwise specified, are ciphered.

$$m_{n,q}=\langle c_{nq}, s_{nq}, D, H_c, \sigma_D^2, p_{xy}, \sigma_P^2, \pi_{xy}, e, \chi_{nq}, e_{nq}, L_n, S_n\rangle$$

where

- c_{nq} is the couple (n, q) identifying the current message.
- s_{nq} is the couple (n_s, q_s) or (r_i, q_s) where n_s or r_i identifies the *Subject* sensor node or the CH, respectively, that either sensed or aggregated the data, and q_s identifies such a message among those transmitted by n_s or r_i. In case of error notification r_i identifies the CH that discovered the error.
- D is the data either sensed or aggregated by the node described by s_{nq}.
- H_c is the hash of the D used by a CH to verify the integrity of the received data. It is important to

highlight that the hash is calculated on the field D that is encrypted.

- σ_D^2 is the variance associated with D. In case of aggregated messages, this field expresses how much spread are sensed data, aggregated in the same message.
- p_{xy} is the geographical position (x, y) of the node described by s_{nq}. This field is in clear.
- σ_P^2 is the variance associated with p_{xy}. This field is in clear. It has the same functionality ad σ_D^2, but it refers to the geographical coordinates.
- π_{xy} is the ciphered version of p_{xy}. It is included in order to detect any possible modification of p_{xy} made by a malicious node.
- e is set to 1 if an error is detected, otherwise it is equal to 0.
- χ_{nq} is the couple (n_s, q_s) that in case of error notification identifies the node that either sensed or aggregated the correct data and the identifier of the message transmitted by such a node.
- e_{nq} is the couple (n_m, q_m) that identifies the node that generated the error message and the identifier of the message reporting the error transmitted by such a node.
- L_n is a list of the nodes which forward the data towards the sink. This field is important also for identifying malicious nodes. In fact, the CHs should make an analysis of the node routing paths, identifying malicious nodes which should adopt a flooding routing instead a multicast tree distribution. The monitoring of the route by means of the list L_n allows the block of this malicious behavior.
- S_n is a list of nodes that processed the original data. It is used by the sink to identify the keys required in the decryption process of received data. Depending on the Maximum Transfer Unit (MTU) of the underlying link layer, the size of this list is upper bounded. For now on, we will refer to id^M as the maximum number of ids that can be concatenated to form the L_n.

It is worthwhile to note that χ_{nq} and e_{nq} are used only in the case of error notification, i.e., when e is set, and that the encryption of the fields $c_{nq}$and s_{nq}, besides supporting integrity, guarantees anonymity.

We consider two possible different encryption (decryption) functions denoted by E_c and E_c^* (D_c and D_c^*). Functions E_c and D_c must provide an homomorphic encryption schema as introduced in [24]; instead, functions E_c^* and D_c^* may not. This means that E_c and E_c^* (D_c and D_c^*) may be or not be the same function.

B. SETA protocols

In this section, we present the following protocols used by SETA:

- *Sensing*, which defines the actions that a node carries out when data are sensed.
- *Integrity Verification*, which defines the actions carried out in order to identify malicious behavior.
- *Data Aggregation*, which comprises the actions that a node carries out to aggregate received messages into a new message.

1) Sensing: Let n be a node sensing a data d from the environment where it is located. Moreover, let us assume that the position of n is represented by the coordinates (x_n, y_n). According to the function-role classification, when sensing d, the node acts as a SS and therefore the node encrypts d using the corresponding key, $k_{n,SS}$. Let q denote the number of messages that n already transmitted over the network. Then, the node calculates the hash of the encrypted data field and encrypts the obtained result with its own hash key, $k_{n,HK}$. Thus, the message $m_{n,q+1}$ is prepared according to the structure discussed in the previous section. Once ready it is queued in the transmission buffer. Note that, when preparing the message, the node acts as a TP and, therefore, all the cyphered fields (but D and H_c) are encrypted using the key $k_{n,TP}$. In particular, for each field of $m_{n,q+1}$, we have

$$
\begin{aligned}
c_{n,q+1} &= (E_c^*\{n, k_{n,TP}\}, E_c^*\{q+1, k_{n,TP}\}) \\
s_{n,q+1} &= (E_c^*\{n, k_{n,TP}\}, E_c^*\{q+1, k_{n,TP}\}) \\
D &= E_c\{d, k_{n,SS}\} \\
H_c &= E_c\{hash(D), k_{n,HK}\} \\
\sigma_D^2 &= 0; \qquad p_{xy} = (x_n, y_n) \\
\sigma_P^2 &= 0 \\
\pi_{xy} &= (E_c\{x_n, k_{n,TP}\}, E_c\{y_n, k_{n,TP}\}) \\
e &= 0; \qquad \chi_{nq} = \epsilon; \qquad e_{nq} = \epsilon \\
L_n &= (E_c^*\{n, k_{n,TP}\}, E_c^*\{q+1, k_{n,TP}\}) \\
S_n &= (E_c^*\{n, k_{n,TP}\}, E_c^*\{q+1, k_{n,TP}\})
\end{aligned}
$$

where ϵ represents an empty field.

Note that for the field π_{xy} since coordinates have a finite representation, we assume, for the sake of simplicity, that they are represented by integer numbers.

2) Integrity Verification: Let $m_{j,h}$ be a message received by a CH. The message is analyzed to find out whether an integrity violation has been performed. More in details, the CH calculates the hash of the field D of the received message and then encrypts the output with the hash key, $k_{n,HK}$. If the obtained result matches with the field H_c of the received message any security violation has been performed and the data are aggregated, according to congestion level of the network, following the same algorithm as in DyDAP [11]. Instead, if they do not match, then the received message should be considered as corrupted and therefore the CH transmits over the network an error notification message structured as follows.

Notice that in this last case the CH acts as NC.

$$
\begin{aligned}
c_{nq} &= (E_c^*\{r_i, k_{r_i,TP}\}, E_c^*\{q+1, k_{r_i,TP}\}) \\
s_{nq} &= (E_c^*\{r_i, k_{r_i,TP}\}, E_c^*\{q+1, k_{r_i,TP}\}) \\
D &= \epsilon\ ; \qquad H_c = 0; \qquad \sigma_D^2 = 0 \\
p_{xy} &= (x_{r_i}, y_{r_i}) \\
\sigma_P^2 &= 0; \qquad \pi_{xy} = \epsilon\ ; \qquad e = 1 \\
\chi_{nq} &= m_{j,h}.s_{nq} \\
e_{nq} &= m_{j,h}.c_{nq} \\
L_n &= m_{j,h}.L_n \cup E_c\{r_i, k_{r_i,TP}\} \\
S_n &= m_{j,h}.S_n
\end{aligned}
$$

where $m_{j,h}.f$ denotes the field f of the received message $m_{j,h}$ and ϵ represents an empty field..

Notice that field s_{nq} identifies the CH which found the error (i.e., node r_i); field χ_{nq} equals the content of field s_{nq} of the received message;field e_{nq} equals field c_{nq} of the received message to provide information on where the error occurred; e is set to 1 to indicate that the current message is an error message. The fields D and π_{xy} are left empty. Finally, the new message is queued in the transmission buffer. The use of the hash function allows us to assign the role of security controller to the CH, reducing the amount of the network traffic, and, in particular, of the overhead in comparison with DyDAP in which the role of controller is performed by all the nodes of the network.

3) Data Aggregation: Whenever the number of messages in the transmission buffer exceeds a given threshold, messages are aggregated to avoid buffer overflow, using the same algorithm as in DyDAP [11].

In particular, the aggregation strategy iteratively operates to arrange enqueued messages in a suitable number of aggregation groups. Then, the messages of each aggregation group are merged into a single message. As a result the number of messages in the transmission buffer decreases. Notice that error messages (i.e., messages having the field e set to 1) are not considered.

The only difference with respect to DyDAP, as regard to data aggregation process, is that, in SETA only CHs are allowed to perform data aggregation.

IV. Performance evaluation

This section discusses the effectiveness of SETA's approach considering data accuracy and security and comparing its behavior with respect to the DyDAP protocol. More in details, it will be shown that SETA, like DyDAP, guarantees: privacy, security congestion control, a good level of delay, and data accuracy also after the aggregation process. In addition, SETA, thanks to the adoption of a mesh architecture that allows a strict sharing of tasks, is able to provide a saving of power consumption for each nodes and for the whole network. In fact, as regard sensor nodes, they perform less actions than in DyDAP;whereas as regard the whole netowrk, in SETA the controller role is performed only by the CHs, as just we said, and the definition of the integrity verification protocol (based on a hashing approach) allows the reduction of the power consumption in comparison with DyDAP. In order to show these aspects, in simulations we compare SETA and DyDAP in terms of: percentage of messages received by the sink, as an indication that the aggregation strategy is effective in avoiding congestion episodes; the percentange of error notification messages, advising malicious behavior received by the sink, as a measure of the transmission overhead; the percentage of the detected errors, as a measure of the robustness towards malicious behavior. Finally, we show the data accuracy, the delay, and the power consumption obtained by using, SETA and DyDAP. In particular, we estimate the data accuracy by means of a comparison between the environmental temperature estimated by the sink and the environmental temperature sensed by the sensor nodes. We quantify the power consumption using the energy consumption models presented in [37]. These models have been derived using an empirical evaluation of the power consumption of typical wireless devices using *Energino*, a real time energy consumption monitoring toolkit. The high performances of *Energino* in term of both sampling frequency and resolution allow us to precisely isolate the impact of specific traffic patterns on the overall energy consumption of wireless devices.

TABLE I. Simulation parameters: default values

Param.	*Description*	*Topology* 1	*Topology* 2	*Topology* 3
N	Number of nodes	50	100	200
C	Cluster Number	3	3	3
N_1	Nodes in cluster 1	20	30	70
N_2	Nodes in cluster 2	10	30	60
N_3	Nodes in cluster 3	20	40	70
M	Percentage of malicious nodes	10%	10%	10%
P	Interval time of data generation	2s	2s	2s
P_{cKmax}	Max Packet size	93 bytes	93 bytes	93 bytes
br_c	Cluster bit rate	250 kbps	250 kbps	250 kbps
br_b	Backbone bit rate	11 Mbps	11 Mbps	11 Mbps
Q_m	CH buffer size	200 msgs	200 msgs	200 msgs
S_m	CH percentage of buffer size emptying	70%-90%	70%-90%	70%-90%
Q_n	Node buffer size	100 msgs	100 msgs	100 msgs
S_n	Node percentage of buffer size emptying	70%	70%	70%
Tab	DyDAP table size	50 msgs	100 msgs	500 msgs
t_S	Duration of simulation	200 s	200s	200s

Notice that the features of SETA concerning privacy (e.g., anonymity and data integrity) are not the target of such simulations even though they are achieved by means of the protocols investigated herein. In fact, privacy management depends on the design choices made in order to develop SETA according to the privacy model described in [11].

In order to exploit the header compression gain due to 6LoWPAN standard [38], we have encapsulated SETA messages in a IPv6 over IEEE 802.15.4 stack as specified in [39]. The simulations have been realized by using the OMNET++ simulator [32]; three different topologies are investigated, characterized by different number of sensor nodes and CHs as shown in the Tab. I, which contains also all the parameters used for the simulations.

A. Simulation results

Herein, we discuss simulation results. Reported results refer only to the topology 1 of Tab. I, since similar outcomes have been obtained for the other topologies. As first, Figures 2 and 3 show a comparison between SETA and DyDAP in terms of amount of the messages received by the sink, with a percentage of malicious node equal to 10%, when the S_m parameter in SETA is set equal to 70% and 90%, respectively. This parameter represents the set point on the transmission buffers of the CHs that the aggregation algorithm seeks to reach to avoid packet losses. The higher S_m the higher the steady state queue level that SETA tries to settle. As just we said above, the percentage of aggregated messages is a measure of the effectiveness of the aggregation strategy, which, to avoid congestion episodes, merges multiple messages in one AMEX packet. It is worth to note that SETA and DyDAP differ in the quota of aggregated messages depending of the value of S_m. This means that such a parameter can be tuned to tailor SETA (or DyDAP) to the specific needs of the application domain, the WSN is adopted for. Furthermore, the lower percentage of error notification messages of SETA reduces the transmission overhead with respect to DyDAP. Such a behavior, as show in Figure 4, is emphasized as the percentage of malicious nodes increases. To shed a further light on this peculiarities of SETA, figure 5 reports the overall number of messages transmitted within the WSN in ideal conditions (i.e., no malicious nodes) and when the percentage of malicious nodes is 10%. From these results it is straightforward noticing that, when malicious nodes are present, SETA is able to reduce the network load due to a clever management of integrity checks with respcet to DyDAP. On the other side, in ideal conditions, DyDAP achieves a lower load because notification meassages are absent.

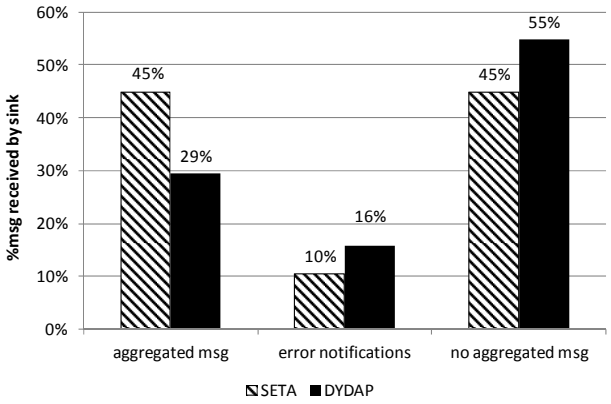

Fig. 2. Number of message received by the sink when in SETA S_m is 70%.

The decrease of the amount of the error notification messages is not associated to a smaller robustness towards the malicious behavior. This is shown in Figure 6, where it is evident that SETA detects a similar percentage of integrity violations in comparison with DyDAP.

The good performance of SETA is also shown in terms of data accuracy (see Figs. 7 and 8), although the percentage of aggregated data is higher. Hence, we evaluate the delay introduced by the adoption in SETA of a hierarchical architecture and we measure the corresponding CDF. The results, in Fig. 9, demonstrate that SETA and DyDAP have a very similar behavior, meaning that the timeliness of sensing operations is not compromised by the SETA approach, even if it requires a smaller processing power.

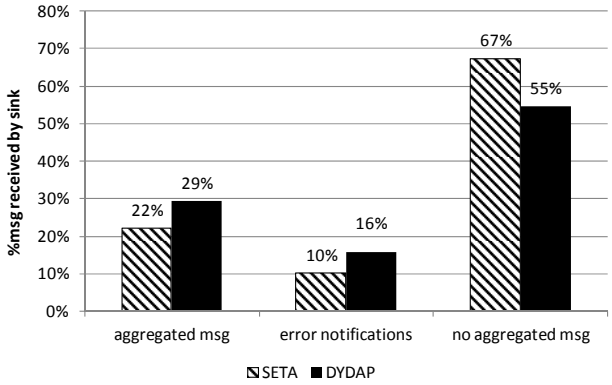

Fig. 3. Number of message received by the sink when in SETA S_m is 90%.

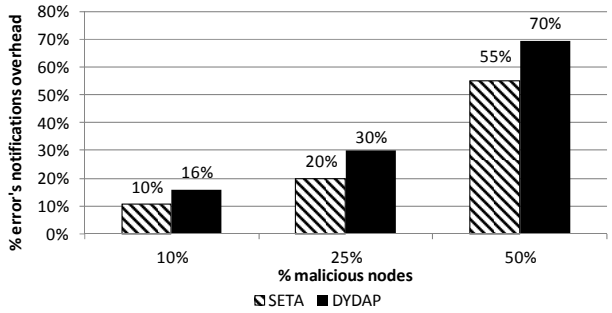

Fig. 4. Error notification overhead.

Finally, in order to provide a complete view of the SETA performance, the power consumption has been evaluated, as shown in Fig. 10. Notice that with DyDAP the energy consumption is slightly higher than using SETA because of the higher number of notifications that sensor nodes have to handle.

Summarizing, the robustness of SETA towards maliciousness and its data accuracy have been obtained without any increase of overhead, delay or energy consumption.

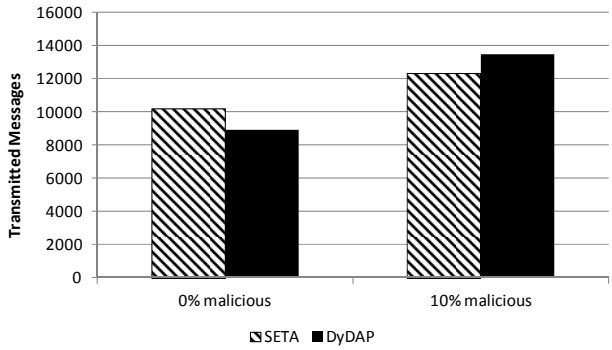

Fig. 5. Transmitted messages.

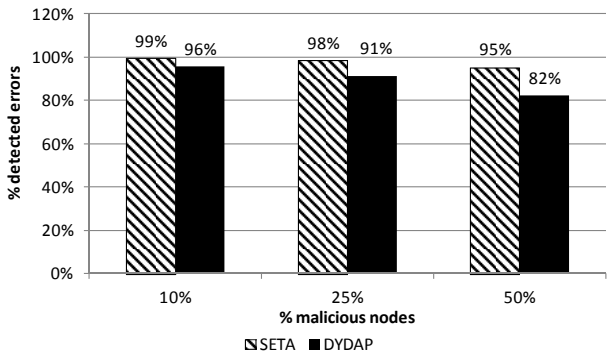

Fig. 6. Detected errors.

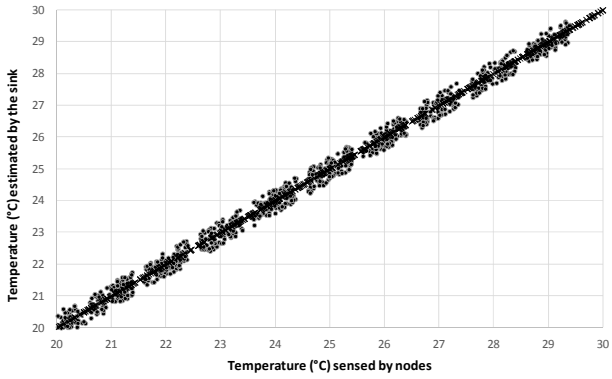

Fig. 7. SETA accuracy.

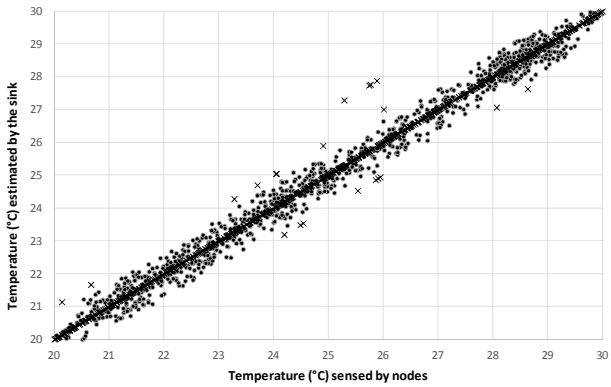

Fig. 8. DyDAP accuracy.

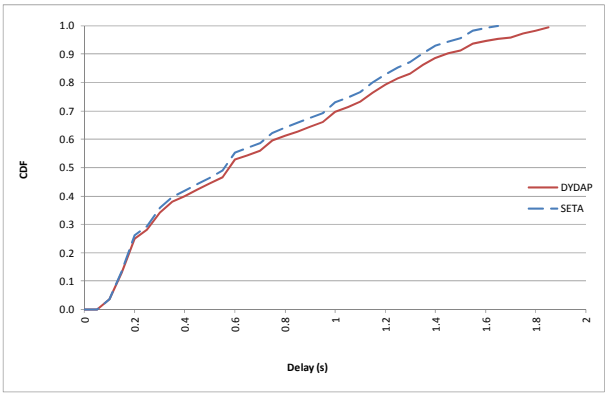

Fig. 9. CDF of packet delay.

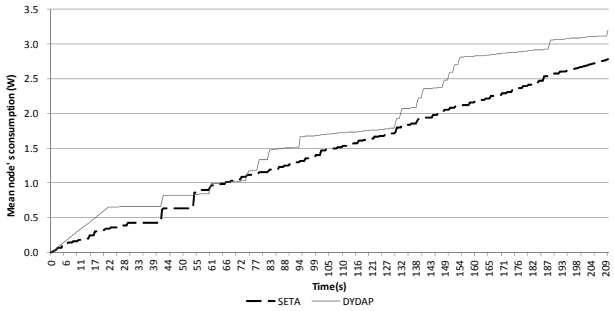

Fig. 10. Mean energy consumption of nodes.

V. Conclusion

In this paper the new SETA solution has been conceived in order to address, at the same time, in a WSN: confidentiality, integrity, adaptive aggregation, and privacy issues. With respect to previous approaches reported in literature, this new scheme allows to take into account the different capabilities of nodes in the WSN, thus reducing computational overhead and power consumption, while increasing the whole network lifetime. The effectiveness of such a new proposal has been proved by simulations in comparison with respect to the recent DyDAP solution. In particular, results have highlighted that SETA is able to fullfill the same design goals as DyDAP while improving robustness against attacks of malicious nodes, environmental data accuracy, and communication overhead. In the next future we are extending SETA fucntionalities for handling multimedia data. Hence, we are evaluating the adoption of SETA in a more complex Internet of Things framework.

References

[1] I.Akyildiz, W.Su, Y.Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, august 2002.

[2] M. Gruteser, G. Schelle, A. Jain, R. Han, and D. Grunwald, "Privacy-aware location sensor networks," in *In Proceedings of the 9th USENIX Workshop on Hot Topics in Operating Systems (HotOS IX)*, 2003.

[3] H. Chan and A. Perrig, "Security and privacy in sensor networks," *IEEE Computer Magazine*, pp. 103–105, March 2003.

[4] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The cricket location support system," in *In Proceedings of ACM Sixth Annual ACM International Conference on Mobile Computing and Networking (MOBICOM)*, 2000.

[5] M. G. S. Duri, P. M. X. Liu, R. Perez, M. Singh, and J. Tang, "Framework for security and privacy in automotive telematics," in *In Proceedings of 2nd ACM International Worksphop on Mobile Commerce*, 2000.

[6] M. Gruteser and D. Grunwald, "A methodological assessment of location privacy risks in wireless hotspot networks," in *In Proceedings of the first International Conference on Security in Pervasive Computing*, 2003.

[7] A. Smailagic, D. P. Siewiorek, J. Anhalt, and Y. W. D. Kogan, "Location sensing and privacy in a context aware computing environment," in *In Proceedings of Pervasive Computing*, 2001.

[8] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: k-anonymity and its enforcement through generalization and suppression," *Technical Report SRI-CSL-98-04, Computer Science Laboratory, SRI International*, 1998.

[9] M. Langheinrich, "A privacy awareness system for ubiquitous computing environments," in *In Proceedings of the 4th Int. Conf. on Ubiquitous Computing*, 2002.

[10] E. Snekkenes, "Concepts for personal location privacy policies," in *In Proceedings of 3rd ACM Conf. on Electronic Commerce*, 2001.

[11] 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.

[12] S. Ozdemir and Y. Xiao, "Secure data aggregation in wireless sensor networks: a comprehensive overview," *Computer Networks*, vol. 53, 2009.

[13] M.Bagaa, N.Lasla, A. Ouadjaout, and Y. Challal, "Sedan: Secure and efficient protocol for data aggregation in wireless sensor networks," in *In Proceedings of IEEE LCN*, Dublin, Ireland, 2007.

[14] A. Mahimkar and T. Rappaport, "Securedav: a secure data aggregation and verification protocol for wireless sensor networks," in *47th IEEE Global Telecommunications Conference (Globecom)*, 2004.

[15] B. Przydatek, D. Song, and A. Perrig, "Sia : secure information aggregation in sensor networks," in *SenSys03*, 2003.

[16] H. Cam, S. Ozdemir, P. Nair, D. Muthuavinashiappan, and H. Sanli, "Energy-efficient and secure pattern based data aggregation for wireless sensor networks," *Comput. Commun., Elsevier*, vol. 29, no. 4, pp. 446–455, 2006.

[17] W. Du, J. Deng, Y. Han, and P. Varshney, "A witness-based approach for data fusion assurance in wireless sensor networks," in *IEEE Global Telecommunications Conference (GLOBECOM 03)*, 2003.

[18] K. Wu, D. Dreef, B. Sun, and Y. Xiao, "Secure data aggregation without persistent cryptographic operations in wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 1, pp. 100–111, 2007.

[19] H. Sanli, S. Ozdemir, and H. Cam, "Srda: secure reference-based data aggregation protocol for wireless sensor networks," in *IEEE VTC Fall Conference*, Sep. 2004.

[20] Y. Yang, X. Wang, S. Zhu, and G. Cao, "Sdap: a secure hop-by-hop data aggregation protocol for sensor networks," in *ACM MOBIHOC06*, Sep. 2006.

[21] S. Ozdemir, *Secure and reliable data aggregation for wireless sensor networks*. LNCS 4836, 2007, pp. 102–109.

[22] ——, "Functional reputation based reliable data aggregation and transmission for wireless sensor networks," *Elsevier Comput. Commun.*, vol. 31, no. 17, pp. 3941–3953, 2005.

[23] D. Westhoff, J. Girao, and M. Acharya, "Concealed data aggregation for reverse multicast traffic in sensor networks: encryption key distribution and routing adaptation," *IEEE Trans. Mobile Comput.*, vol. 5, pp. 1417–1431, 2006.

[24] C. Castelluccia, E. Mykletun, and G. Tsudik, "Efficient aggregation of encrypted data in wireless sensor networks," in *Conference on Mobile and Ubiquitous Systems: Networking and Services*, 2005.

[25] I. Rodhea and C. Rohner, "n-lda: n-layers data aggregation in sensor networks," in *28th International Conference on Distributed Computing Systems Workshops*, 2008.

[26] W. Zhang, Y. Liu, S. Das, and P. De, "Secure data aggregation in wireless sensor networks: a watermark based authentication supportive approach," *Elsevier Pervasive Mobile Comput.*, vol. 4, pp. 658–680, 2008.

[27] J. Domingo-Ferrer, "A provably secure additive and multiplicative privacy homomorphism," in *Information Security Conference*, 2002.

[28] B. Sun, N. Chand, K. Wu, and Y. Xiao, "Change-point monitoring for secure in-network aggregation in wireless sensor networks," in *IEEE Global Telecommunications Conference (GLOBECOM 2007)*, 2007.

[29] 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.

[30] R. Riggio, T. Rasheed, and S.Sicari, "Performance evaluation of an hybrid mesh and sensor network," in *Proc. of Globecom 2011*, Miami, USA, December 2011.

[31] A. Coen-Porisini, P.Colombo, and S.Sicari, "Dealing with anonymity in wireless sensor networks," in *Proc. of 25th annual ACM symposium on Applied Computing (ACM SAC)*, Sierre, Switzerland, 2010.

[32] Http://www.omnetpp.org/.

[33] I. Akyildiz, X.Wang, and W.Wang, "Wireless mesh networks: a survey," *Elsevier Computer Networks*, vol. 47, no. 4, pp. 445 – 487, 2005.

[34] B. H. Walke, S. Mangold, and L. Berlemann, *IEEE 802 Wireless Systems*. NJ, USA: John Wiley & Sons, Ltd, 2006.

[35] H. Zhanga, A. Arorab, Y. Choic, and M. Goudac, "Reliable bursty convergecast in wireless sensor networks," *Elsevier Computer Communications*, vol. 30, no. 13, pp. 2560–2576, 2007.

[36] Q.Ni, A.Trombetta, E. Bertino, and J. Lobo, "Privacy-aware role based access control," in *Proceedings of the 12th ACM symposium on Access control models and technologies*. New York, NY, USA: ACM, 2007.

[37] K.Gomez, R.Riggio, T.Rasheed, D.Miorandi, and F.Granelli, "Energino: a Hardware and Software Solution for Energy Consumption Monitoring ," in *Proc. of IEEE WinMee*, Paderborn, Germany, 2012.

[38] J. Hui and P. Thubert, *Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks*, RFC 6282, IETF RFC 6282, September 2011.

[39] M. R. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, L. A. Grieco, G. Boggia, and M. Dohler, "Standardized Protocol Stack For The Internet Of (Important) Things," *IEEE Communications Surveys and Tutorials*, 2013, to be published.