Dealing with Anonymity in Wireless Sensor Networks

Alberto Coen-Porisini Pietro Colombo Sabrina Sicari
Dipartimento di Informatica e Comunicazione
Università dell'Insubria
Via Mazzini, 5 - 21100 Varese (Italy)
{alberto.coenporisini, pietro.colombo, sabrina.sicari}@uninsubria.it

ABSTRACT

Nowadays Wireless Sensor Networks (WSN) are used in many application contexts. Data handled by WSN are required to be protected for privacy reasons since they can be directly or indirectly related to individuals. The problem of preventing the identification of individuals starting from their data, known as anonymity, is a fundamental requirement for privacy aware systems.

This paper proposes a solution to guarantee anonymity for a wide spread type of WSN by means of privacy policies. The solution is based on a UML model that introduces the conceptual elements and guidelines that are needed to build privacy policies for WSN.

1. INTRODUCTION

Wireless Sensor Networks (WSN) [2] technologies support data collection and distributed data processing by means of very small sensing devices. Nowadays, sensors are used in many contexts such as surveillance systems, systems supporting traffic monitoring and control in urban/suburban areas, military and/or anti-terrorism operations, telemedicine, assistance to disabled and elderly people, environmental monitoring, localization of services and users, industrial process control.

Privacy aware mechanisms are required for several WSN applications such as localization and telemedicine systems. However, it is necessary to take into account privacy also in those application contexts in which the data of individuals are not directly handled by WSN. In fact sensor nodes continuously store and elaborate a large amount of information, and although the managed information usually consists of raw scalar data (e.g., the current temperature, pressure, and so on) not directly related to people, an in-depth analysis of those data may reveal information on individuals. However, the low power resources, the poor computational and storage capabilities of sensor nodes impose severe constraints on how these requirements can be satisfied.

Among the different aspects of privacy, anonymity is an important requirement for a privacy aware system that aims at protecting the identity of the individuals whose data are handled by the system.

In order to achieve such a goal we propose a solution based on privacy policies that besides constraining the actions that can be executed on the sensed data hide the identity of the nodes. In fact, in different application scenario starting from the identity of the sensor nodes it is possible to retrieve directly or indirectly the identity and the behaviour of an individual. For example, in home networks in order to provide advanced services to improve the quality of human life and to guarantee energy saving, sensor nodes collect a large amount of data such as humidity and temperature that may reveal individuals' habits breaking their privacy.

This paper proposes a conceptual model that provides a sound foundation for the definition of privacy policies in the context of Wireless Sensor Networks. The model, which extends the work presented in [5], is defined in UML[14, 15] and represents a general schema that can be easily adopted in different contexts.

The model introduces concepts, such as nodes, data, actions, that are needed in order to define a privacy policy along with the existing relationships among them.

The paper also illustrates the definition of a privacy policy that guarantees anonymity for a wide spread type of WSN. The policy is built starting from the concepts and the guidelines imposed by the conceptual model, and consists in the definition of exchanged messages and actions (executable by nodes) organized by means of communication protocols.

The rest of the paper is organized as follow: Section 2 introduces the foundations for modeling privacy in the context of WSN and proposes the conceptual model; Section 3 illustrates the anonymity problem by proposing a solution built on the conceptual model; Section 4 presents some related works; finally, Section 5 draws some conclusions and provides hints for future works.

2. MODELING PRIVACY POLICIES FOR WIRELESS SENSOR NETWORKS

A privacy policy defines the way in which data referring to individuals can be collected, processed and diffused according to the rights that individuals are entitled to.

The rest of the paper adopts the terminology introduced by the EU directive [6]. Since the proposed terms are general, i.e., they are not dedicated to a specific type of network, it is required that they are refined to provide the concepts that are necessary to support the definition of privacy mechanisms concerning the communication in WSN.

SAC'10 March 22-26, 2010, Sierre, Switzerland.

- *personal data* in general means any information related to an identified or identifiable natural person (referred to as *data subject* or *subject*). In the context of WSN they represent the data that are sensed by the nodes of the network, in other words, the role of subject is played by the nodes since they receive information from the environment where they are located.

- *processing of personal data* (*processing*) means any operation or set of operations which is performed upon personal data, whether or not by automatic means, such as collection, recording, organization, storage, adaptation or alteration, retrieval, consultation, use, disclosure by transmission, dissemination or otherwise making available, alignment or combination, blocking, erasure or destruction. Besides the above operations, in the WSN context further activities can be defined such as sensing data (of different nature), transmitting messages to other nodes, receiving/retransmitting messages. Moreover, a node is also capable to perform operation on data and messages such as data aggregation, data encryption/decryption and data integrity verification.

- *controller* in general means the natural or legal person, public authority, agency or any other body which alone or jointly with others determines the purposes and means of the processing of personal data; in WSN the role of controller is played by the nodes of the network. A controller verifies the processing actions that handle sensed data.

- *processor* means a natural or legal person, public authority, agency or any other body which processes personal data on behalf of the controller; in WSN the role of processor is played by the nodes of the network.

- the data *subject's consent* (*consent*) means any freely given specific and informed indication of his/her wishes by which the data subject signifies his/her agreement to personal data relating to him/her being processed. The meaning is the same also for WSNs.

As a distinctive feature of a privacy policy, the processor is required to state for what *purpose* data are processed. A purpose can be defined either as a high-level activity (e.g., "monitoring", "tracking") or as a set of actions (e.g., "compute the average temperature", "evaluate the humidity").

The general goal of a WSN is to collect data sensed from the nodes that are distributed in the environment. Data once sensed are elaborated by nodes and transmitted, by means of messages, to other nodes that in turn receive, elaborate and retransmit the messages until reaching the sink. The actions of data elaboration, message transmission, reception and retransmission are processing actions executed with the purpose of communicating the results of the distributed computation to the sink. In other words, the purpose associated with the processing actions is the general functional goal of the network. A node that performs processing actions plays the role of processor. Notice that a node may also play the role of controller. For instance, a node can be required to verify the integrity of the data content of a retransmitted message with respect to the original message that was transmitted to the network.

Notice that the processing actions may be executed under specific obligations. *Obligations* are a set of actions that processor and controller guarantee to perform at the end of the processing activities. As an example, consider a node of a network that keeps track of the temperature of the ground. Whenever the temperature is less than 1° C, the node has to send an alert message to the sink stating that the ground is about to freeze.

In a privacy aware system, subjects have to grant their consent before any processing could be executed on their data. We assume that the consent is implicitly given by the nodes of the network. A node that belongs to a WSN accepts that its data can be the target of different processing activities. The processing activities consist in the set of processing actions (and obligations) that can be executed by the single node of the network, while the general purposes associated with such processing actions represent the computational goals of the whole network system. This type of implicit consent requires that the system modeler adopts adequate mechanisms to assure that a node trusts the network in which it wants to operate.

2.1 The UML Model

In the following we give a short overview of the conceptual model for privacy policies.

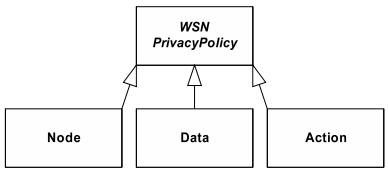

Figure 1: The Privacy Policy Class Diagram

The structural aspects are defined using UML classes and their relationships such as associations, dependencies and generalizations. Figure 1 depicts a class diagram that provides a high level view of the basic structural elements of the model.

A *WNS-PrivacyPolicy* is characterized by three types of classes: *Node*, *Data* and *Action*. Nodes interact among them inside the network in order to perform some kind of actions on data. Thus, an instance of *WSNPrivacyPolicy* is characterized by specific instances of *Node*, *Data* and *Action*, and by the relationships among such entities.

Let us focus on the classes introduced by the diagram:

- *Node* represents a member of the network either interested in processing data or involved by such a processing. Nodes are characterized by functions and roles (see Figure 2). More specifically:

  - *Role* [13] is a key concept of this approach; nodes are characterized depending on the role they play with respect to privacy. *Role* is extended by three distinct classes to represent the different roles: *Subject*, which is a node that senses the data, *Processor*, which is a node that processes data by performing some kind of action on them (e.g., transmission, retransmission, aggregation, etc.) and *Controller*, which is a node that verifies the actions executed by processor nodes.

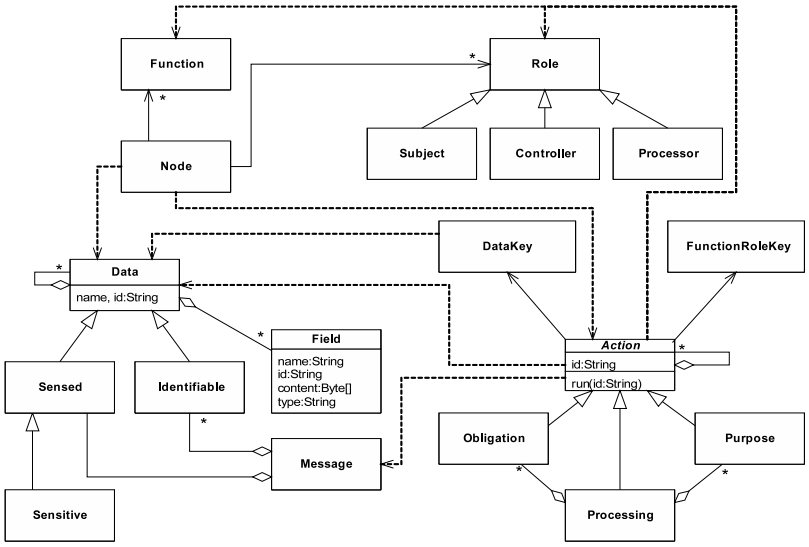

Figure 2: The WSN Privacy Class Diagram

– *Function* represents the task performed by a *Node* within the network in which it operates (e.g., data sensing, message transmission, message retransmission, data aggregation, etc.).

- *Data* represents the information referring to subjects that can be handled by processors. *Data* is extended by means of *Identifiable* data (e.g., node identifier), *Sensitive* data (e.g., health related data) and *Sensed* data (e.g., temperature, pressure). *Identifiable* data represent the information that can be used to uniquely identify nodes. *Sensed* data contain information that are sensed by the nodes of the network. Finally, *Sensitive* data represent information that deserve particular care and that should not be freely accessible.

  In the WSN context, sensitive data may be considered an extension of sensed data, i.e., they are sensed data related to individuals which require a particular care. For instance, in telemedicine applications a sensitive datum is the temperature which is sensed by nodes positioned on the body of patients.

  Notice that in the context of WSN also common sensed data deserve particular care. For example, consider a wireless meter reading system used to monitor the temperature and pressure of different rooms of the building where it is installed. Such a system comprises several sensor units which communicate information on the current temperature, barometric pressure, and humidity of the rooms where they are positioned. Although the data sensed by the nodes of the system cannot be classified as sensitive, they can be used to reveal information on personal habits of the people who live in the interested building. As an example, slight increments of temperature or of humidity may reveal the presence of one or more person in a room. By analysing such data it is possible to infer periods of the day or of the week during which the building is empty. Notice that such information might be exploited by potential burglars.

  Data is a complex structure composed of basic information units, named *Fields*. Each field represents a partial information related to the whole data structure.

- *Message* represents the basic communication unit exchanged by the nodes of the network. *Message* contains *Identifiable* information concerning the nodes involved in the communication and *Sensed* data.

- *Action* represents any operation performed by *Node*. *Action* is extended by *Obligation*, *Processing* and *Purpose*. Moreover, each action can be recursively composed of other actions. Since in a privacy aware scenario a processing is executed under a purpose and an obligation, *Processing* specifies an aggregation relationship with *Purpose* and *Obligation*.

  We propose the following actions as some of the most common operations performed in WSNs:

  – *sensing*, which represents the acquisition of data concerning a specific feature of the system where the node is located. For instance, the monitoring of temperature, pressure or humidity.

  – *transmission*, which consists in transmitting a message containing data that were sensed by the current node.

  – *reception*, which consists in receiving a message sent by another node of the network

  – *retransmission*, which consists in transmitting a message containing data that were sensed by other nodes of the network.

  – *verification*, which consists in checking the integrity of the data contained in a received message. In case any inconsistency is discovered, possible countermeasures are taken, such as sending a warning message to the sink.

  – *data aggregation*, which consists in aggregating data starting from the data contained in the re-

ceived messages, that will be sent to other nodes by means of the action *transmission*.

Notice that while in general for each function there may be defined several actions that can be performed, in the context of WSN usually, each function corresponds to one action.

In order to guarantee the confidentiality and integrity of data as well as to assure that only authorized nodes are allowed to access data and execute actions, our model introduces encryption mechanisms. More specifically, two classes representing encryption keys, named *DataKey* and *FunctionRoleKey*, are introduced. The former class is used for the definition of encryption mechanisms to protect the data content of messages; while the latter is used for defining mechanisms to assure that message communication and data handling are executed only by authorized nodes. Each node of the network owns a different *DataKey* used to encrypt the data content of the messages. Each node also owns multiple *FunctionRoleKey* that are used to constrain which nodes are allowed to execute specific actions on data. Actions are expressly built to be executed by nodes that belong to a given function-role combination. Since a node may play different functions and roles, it may own multiple function-role keys, one for each pair of function-role.

Notice that the system modeler is allowed to use the key generation algorithm and the encryption algorithm that he/she considers the most suitable for the application domain.

Figure 2 depicts the aforementioned entities along with their relationships by means of a UML Class diagram. For instance, the dependency relationship between *Action* and *Data* means that data are processed by actions, while the association between *Subject* and *Data* expresses data ownership.

3. THE ANONYMITY PROBLEM

Anonymity is a fundamental requirement for privacy aware systems, which aims at preventing the identification of subjects starting from their data. In the context of WSN the aim is enforced by preventing the identification of nodes that sensed data starting from the messages that are exchanged within the network. A significant example that depicts the importance of the anonymity requirement is shown by WSN used in military applications. As an example consider a network composed of nodes that sense the temperature and that are located on ground behind the enemy lines. The sensed data generated by each node may be used to determine the presence of enemy troops in the region where the sensor is positioned. Hence, it is required that it cannot be possible to identify which node generated a sensed datum by sniffing the messages transmitted by the network.

3.1 Reference scenario

We consider a dense network composed of n nodes, with $n \in \aleph$, each of which senses a given type of data (e.g., temperature, pressure, brightness, position and so on). Each node directly communicates with its closer neighbours (at one hop distance). The broadcast nature of wireless channels enables a node to determine, by snooping the channel, whether its packets are received and forwarded by its neighbors, for any kind of MAC layer [21]. Each node waits a random time before re-transmitting the message in order to reduce the collision probability.

The aim of communication is to provide the sink with the data sensed by the nodes of the network.

Each node of the network is characterized by a label N_i, with $1 \leq i \leq n$, which unambiguously identifies the node in the network.

Each node owns different types of keys each of which corresponds to a given Function-Role pair. We identify the following Function-Role pairs: *Sensing-Subject*, *Authenticator-Processor*, *Transmitter-Processor* and *Notifier-Controller*. A key is described by means of the following notation $k(nd, fr)$, where nd and fr represent the node that owns the key and the Function-Role played by such a node, respectively. More specifically, $nd = N_1..N_n$ and $fr = \{ss, ap, tp, nc\}$ where ss represents the label associated with the Function-Role *Sensing-Subject*, ap with *Authenticator-Processor*, tp with *Transmitter-Processor* and nc with *Notifier-Controller*. For instance, the *Sensing-Subject* key of node N_i is represented by $k(N_i, ss)$.

Notice that in the proposed solution keys are pre-shared in the nodes.

We assume that each node is also equipped with a table where it stores the last sent messages. The usefulness of the table will be clarified in the following sections.

3.2 The proposed solution

The proposed solution starts from the classification of data, functions and roles played by the nodes described by the previously presented conceptual model.

3.2.1 Message structure

In order to define anonymity mechanisms in a WSN, the messages handled by the nodes of the network need to be suitably structured. More specifically, a message contains data that according to the conceptual model may be classified as *identifiable* and *sensed*. *Identifiable* data includes the information that can be used to identify a node. *Sensed* data includes all information sensed by the nodes, such as the environmental temperature, pressure and so on.

A message represents the object of a single step of the communication towards the sink, more specifically it refers to a single transmission hop between adjacent nodes. A message is identified by means of the notation $msg_{x,d}$ where x represents the node that generated and transmitted the message, while d identifies the message among those generated by node x. The pair x,d unambiguously identifies the message among those trasmitted within the network. A sensed data before reaching the sink passes through different nodes of the network (multi-hop communication) by means of different messages. In order to guarantee the integrity and confidentiality of the end-to-end communication, we propose a message structure that keeps track of the two last hops of the transmission. This condition will allow us to implement a basic enforcement schema that checks the integrity of the data content of the message.

A message $msg_{x,d}$, is a tuple
$msg_{x,d} = \langle current, previous, subject, data \rangle$ where:

- *current* is a tuple $current = \langle N_x, d \rangle$, which unambiguously identifies the current message among the ones transmitted within the network. It includes N_x, the identifier of the node that is going to transmit the message and d, an identifier of the message among those generated by node N_x.

- *previous* is a tuple $previous = \langle N_y, e \rangle$, which includes N_y, the identifier of the node that operated the second last (re)transmission of the sensed data contained in the current message and e, the idenfier that node N_y associated with such a message.
- *subject* is a tuple $subject = \langle N_z, f \rangle$, which includes N_z, the identifier of the node subject, which originally sensed the data, and f, the identifier that such a node associated with the message that started the communication of the sensed data towards the sink.
- *data* is a tuple $data = \langle sd, er, si, mi \rangle$, which includes the data sensed by the subject node and additional fields used for error notification.
  - *sd* (sensed data) contains the data that were sensed by the subject
  - *er* (error) is a flag that indicates an anomaly was identified in the message content.
  - *si* (sensing identifier) is a tuple $si = \langle N_v, g \rangle$, which contains the identifier of the node that sensed the data and the identifier of the message transmitted by such a node.
  - *mi* (mistaking identifier) is a tuple $mi = \langle N_w, h \rangle$, which contains the identifier of the node that generated the error and the identifier of the message containing the error transmitted by such a node.

  Notice that *sensing identifier*, *mistaking identifier* are used only in case of error notification, i.e., when *error* is set to true, as it will be described in the *retransmission & verification protocol*.

We assume that $x, y, z, v, w \in (1..n]$ and d,e,f,g,$h \in \aleph$.

3.2.2 Behavioral aspects

The dynamics of the system are described by means of the following protocols:

- *sensing*, which defines the actions that a node of the network executes to sense data and to communicate such data to the other nodes of the network.
- *retransmission and verification*, which defines the actions that a node must perform to retransmit data received from other nodes, and specifies the actions that a node must execute in order to verify the integrity of the messages that are transmitted within the network.

Both protocols make use of cryptography in order to implement anonymity. Notice that at this level we do not need to consider any particular encryption technique. The proposed solution is independent from encryption algorithms. The system modeler is allowed to use the technique that he/she considers the most suitable for the application domain. The usage of encryption is denoted by the notation $en(pc, key)$, where en is an encryption function,

$en : String \times String \rightarrow String$ that taken a plaintext pc, and a key key, returns the cipher-text cc.

3.2.3 Sensing protocol

1. *Data sensing.* The node N_z senses a data sd from the environment where it is located. Notice that in this case the node plays the *Role* of *Subject* and the *Function* of *Sensing*.
2. *Data encryption.* The node encrypts the sensed data sd by using its *Sensing-Subject* key $k(N_z, ss)$[1]. The resulting output is denoted $en(sd, k(N_z, ss))$
3. *Message identifier generation.* The node generates an identifier for the message that has to transmit to the sink $\langle N_z, f \rangle$
4. *Identifiable data encryption.* The node encrypts the generated identifier by using its personal *Transmitter-Processor* key, $k(N_z, tp)$. As a result we have the content $en(\langle N_z, f \rangle, k(N_z, tp))$
5. *Message structuring.* A new message $msg_{z,f}$ is generated starting from the resulting outputs of steps 2 and 4. The resulting message is structured as follow:
   - *current* is set to $en(\langle N_z, f \rangle, k(N_z, tp))$ since the current transmitter is the subject itself.
   - *previous* is initialized to an empty string. This is the first transmission, no retransmission has been executed yet.
   - *subject* is set to $en(\langle N_z, f \rangle, k(N_z, tp))$.
   - *data* is set to $en(sd, k(N_z, ss))$.
6. *Message storing.* The node stores the content of the encrypted field *data* in its local table. It uses the content of the field *current* $en(\langle N_z, f \rangle, k(N_z, tp))$ of the message $msg_{z,f}$ as the hash key for the sensed data that have to be stored.
7. *Message transmission.* The node waits for a random time and transmits the message $msg_{z,f}$ to its closer neighbours (at one hop distance).

3.2.4 Retransmission & verification protocol

1. *Message reception.* The node N_h (with $0 < h < n$) receives the message $msg_{r,v} = \langle c_{r,v}, p_{r,v}, s_{r,v}, d_{r,v} \rangle$, where $c_{r,v}$, $p_{r,v}$, $s_{r,v}$, $d_{r,v}$ represent the fields *current*, *previous*, *subject* and *data*, respectively.
2. *Role check.* The node N_h analyses the message in order to understand what type of action it has to execute on the contained data. More specifically, it looks for the message among those stored in the local table by using the encrypted content of field *previous* as hash key.

   If the message is not found then this means that it was not previously transmitted by node N_h. In this case the nodes goes on playing the role of *Processor* and the function of *Transmitter* by executing the following steps required for the retransmission of the message.

   a) *Message identifier generation.* The node generates a new identifier for the message, $\langle N_h, t \rangle$, which will be retransmitted towards the sink

   b) *Identifiable data encryption.* The node encrypts the identifier by using its personal *Transmitter-Processor* key $k(N_h, tp)$.

[1]The *Sensing-Subject* key is equivalent to the *DataKey* defined in the conceptual model

c) *Message structuring.* A new message $msg_{h,t} = \langle c_{h,t}, p_{h,t}, s_{h,t}, d_{h,t}\rangle$is generated starting from the resulting output of step b) and the encrypted content of the field *current*, *subject* and *data* of the received message $msg_{r,v}$. As a consequence, the resulting message is structured as follows:

$c_{h,t} = \langle en(\langle N_h, t\rangle, k(N_h, tp))\rangle$, $p_{h,t} = c_{r,v}$, $s_{h,t} = s_{r,v}$, $d_{h,t} = d_{r,v}$

Notice that field *previous* of the new message $msg_{h,t}$ is equal to field *current* of the received message $msg_{r,v}$ since *current* and *previous* are updated at each retrasmission.

d) *Message storing.* The node stores the content of the encrypted field *data* in its local table. It uses the content of the field *current* of the message $msg_{h,t}$, i.e., $en(\langle N_h, t\rangle, k(N_h, tp))$, as the hash key for the sensed data that have to be stored.

e) *Message transmission.* After a random time the node transmits the message $msg_{h,t}$ to its closer neighbours (at one hop distance)

Otherwise, if the message is found then it means that it was originally transmitted by node N_h itself. In this case the node changes its current function and role, i.e., it has to play the role of *Controller* and the function of *Notifier* to verify the integrity of the previously transmitted message. Hence, the node compares the encrypted content of field *data* of the received message with the encrypted data extracted from its table.

If the data match, this means that the *Controller* is sure that the node from which it received the message preserved the integrity of the data content. In this case no additional action is performed by the node.

If the data do not match, the content of field *data* is different from the data extracted from the local table or no data entry corresponds to the search key. This means that something wrong happened. In this case, the node generates a new message to notify the sink that a corrupted message is spreading through the network.

a) *Message identifier generation.* The node generates a new identifier, $\langle N_h, t\rangle$, for the message $msg_{h,t}$ that will be retransmitted to the sink

b) *Identifiable data encryption.* The node encrypts the resulting identifier by using its personal *Transmitter-Processor* key $k(N_h, tp)$.

c) *Message structuring.* A new message is generated starting from the resulting output of steps b), the encrypted content of the field *current*, *subject* and *data* of the received message $msg_{r,v}$. The resulting message is structured as follows:

- *current* is set to $en(\langle N_h, t\rangle, k(N_h, tp))$.
- *previous* is a copy of the field *current*, $c_{r,v}$, of the received message
- *subject* is set to $en(\langle N_h, t\rangle, k(N_h, tp))$in order to specify identifiable information of the node that retrieved the error. Notice that since such a node is the current one, the content of *subject* is equal to *current*.
- *data* contains: 1) a code that specifies that the current message is an error message; 2) the field *subject* of the received message, $s_{r,v}$, which contains the identifier of the node that sensed the data and started the transmission and the identifier of the first message generated by it; 3) the field *current* of the received message, $c_{r,v}$, which contains the identifier of the node that made the mistake[2]; 4) the correct data, *sd*, that was stored in the local table $en(sd, k(N_z, ss))$.

  The whole content of field *data* is encrypted with the *Notifier-Controller* key of the current node $k(N_h, nc)$.

  $en(errorcode, s_{r,v}, c_{r,v}, en(sd, k(z, ss))\rangle, k(N_h, nc))$

d) *Message storing.* The node stores the content of the encrypted field *data* in its local table. It uses the content of the field *current* of message $msg_{h,t}$, $en(\langle N_h, t\rangle, k(N_h, tp))$, as hash key for the field *data* that have to be stored.

e) *Message transmission.* After a random time the node transmits the message to its closer neighbours (at one hop distance).

3.2.5 Towards trust

The previously described protocol supports both the anonymity and the confidentiality of the communication among the nodes of the network. A further requirement concerns the trust of the communication, in other words it is required that only authorized nodes are allowed to communicate within the system. A step towards the achievement of this requirement can be done by using an authenticator-processor key that is known by all the nodes of the network.

Notice that before each transmission and retransmission the node could encrypt the involved message with its authenticator key. The aim of this encryption phase is to assure that the transmitter is a trusted node of the network since it knows and uses the membership key of the network. It is also required that a node that receives an "authenticated" message decrypts it by using its authenticator-processor key before handling its content for elaboration or retransmission purposes. This mechanism prevents external untrusted nodes from spreading and accessing messages that are travelling across the network.

Although the usage of this encryption mechanism may improve the level of trust among the nodes of the network, it requires computational resources that are relevant for nodes of a WSN. In such a scenario the number of encryptions/decryptions is equal to the number of message transmissions and receptions.

This scenario can be effectively applied whenever the nodes are characterized by adequate computational capabilities and battery units. Hence, the proposed solution could be applied to the next generation of sensor technologies [1, 19].

3.2.6 Enforcement

Privacy policy enforcement consists in verifying the compliance of the actions performed by node with a given privacy policy. In general, there are two different ways in which

[2]Notice that such a content as well as the one of field *subject* are taken from the received message, and are encrypted with the key of the previous node and of the *Subject* node, respectively.

such a verification can be carried out: the first way consists in providing *ex-post* enforcement mechanisms that is, the controls are done after all the actions related to a policy are performed (e.g., audit-based mechanisms). The second one consists in having *run-time* enforcement mechanisms that is, the effect of every action is checked before actual execution. The conceptual model supports the definition of both types of enforcement mechanisms. The proposed solution, built on the conceptual model, provides an example of how it is possible to define both types of enforcement mechanisms. The *ex-post* mechanism is implemented in the *Retransmission& verification Protocol*, when *Controller* checks the integrity of the retrasmitted message under the obligation to notify an error message to the sink in case something is wrong. The *run-time* mechanism is implemented when the basic protocol is extended by means of trust management mechanisms. In such a case an authenticator-processor key is used, which assures that communication is effectively perfomed by nodes belonging to the network.

3.3 Final remarks

The proposed solution satisfies the anonymity requirements since it proposes mechanisms that mask the identity of subject nodes.

The separation of sensed data from identifiable data, and the adoption of encryption techniques make it more difficult to associate sensed data with the identity of the node that sensed them. The computation effort that it is required to support the anonymity is limited as well as the overhead and delay. The solution also supports the check of the integrity of the messages that are exchanged.

4. RELATED WORKS

WSN applications require to collect a huge amount of data that may be used to violate directly or indirectly the privacy of individuals. Notice that the risk of violation increases due to both the wireless nature of the communication channel and the remote access. Exploiting such vulnerabilities the following common threats against sensor privacy may occur[10, 4]:

- Eavesdropping: malicious users could easily discover the communication content listening to data.
- Masking: some malicious nodes may mask their real nature behind the identity of nodes that are authorized to take part to communication, and misroute the packets.

The available solutions defined to guarantee privacy in WSN starting from their vulnerabilities and related threats may be classified into two main groups: anonymity mechanisms based on data cloaking [10, 17] and privacy policy based approaches [7]. Data cloaking anonymity mechanisms perturbs data following some kind of criterium, for instance K-anonymity guarantees that every record is indistinguishable from at least k-1 other records [18].

In [10, 9, 17, 8] four main data cloaking anonymity approaches[3] are proposed:

- Decentralize Sensible Data: the basic idea of this approach is to distribute the sensed location data through a spanning tree, so that no single node holds the complete view of the original data.
- Secure Communication Channel: the use of a secure communication protocols, such as SPINS [16], reduces the eavesdropping and active attack risk by means of encryption techniques.
- Change Data Traffic: the traffic pattern is altered with some bogus data that obfuscate the real position of the nodes.
- Node Mobility: the basic idea is to move the sensor nodes in order to change dynamically the localization information, making it difficult to identify the node.

For instance, [10] proposes a solution that guarantees the anonymous usage of location based information. More specifically, such a solution consists of a cloaking algorithm which regulates the granularity of location information to meet the specified anonymity constraints. This work only focuses on localization services and therefore, constrains the middleware architecture required to support the proposed algorithm. Hence, such a solution cannot be considered a general context independent anonymity approach.

Privacy policy based approaches [11, 20, 7] state who can use individuals data, which data can be collected, for what purpose the data can be used, and how they can be distributed. A common policy based approach addresses privacy concerns at database layer after data have been collected [20]. Other works [12] address the access control and authentication issues, for instance Duri et al.[7] propose a policy-based framework for protecting sensor information. The Mist routing project for mobile users [3] combines location privacy with communication aspects. It faces the source location privacy problem by designing *ad hoc* routing protocol that keeps the location private from source to routers.

Our work provides a contribution in the field of privacy policy based approaches by defining a role-based context-independent solution that guarantees anonymity of the nodes before sensed data are collected into a database. Our solution may be combined with both data cloaking mechanisms and some other privacy policy based approaches.

5. CONCLUSIONS

The present work proposed a conceptual model for the definition of privacy policies in the context of Wireless Sensor Networks. The proposed model provides the basic concepts involved when dealing with the management of privacy-related information in a WSN. Basic elements such as the concepts of message, action, node, sensed data are represented by means of UML.

The choice of using UML is motivated by the fact that such a notation is well known by a wide range of analysts and modelers that operate both in software and system engineering fields. Moreover, UML can be used for representing concepts at different levels of abstraction. Even though the model is described at a very high level, it can be easily extended and adapted for specific application domains.

This paper also proposes one of these applications concerning the WSN anonymity problem. Such a problem states that it cannot be possible to identify the nodes which sensed data starting from the messages that are exchanged within a network.

[3]notice that [10, 9] are specific for cloaking localization information

The model provides the conceptual foundations that are required to build anonymity assurance mechanisms, such as the separation of sensed from identifiable data, and the classification of roles and functions. The proposed solution consists in protocols that state the structure of the messages that can be exchanged in the network, the keys and the encryption mechanisms that are required to protect the communication in the network and in all the activities that are required to support anonymity.

The proposed solution is general and can be easily adopted for different types of WSN, from simple networks where nodes sense data and transmit them without any further elaboration, to network supporting advanced form of data aggregation. The proposed solution is also independent from the types of data that are sensed and handled by the nodes, hence it can be applied to simple networks that sensed the temperature of the environment, as well as to multimedia sensor networks whose nodes may exchange audio and video signals.

At present we are experimenting the application of the proposed solution by using simulation tools with the aim to identify possible changes that could diminish the computational power and memory usage with the aim to improve the network performances.

6. REFERENCES

[1] I. F. Akyildiz, F. Brunetti, and C. Blazquez. NanoNetworking: A New Communication Paradigm. *Computer Networks Journal*, (Elsevier), June 2008.

[2] I.F. Akyildiz, T. Melodia, and K. Chowdhury. A Survey on Wireless Multimedia Sensor Networks. *Computer Networks Journal*, (Elsevier), March 2007.

[3] J. Al-Muhtadi, R. Campbell, A. Kapadia, M. D. Mickunas, and S. Yi. Routing through the mist: privacy preserving communication in ubiquitous computing environments. In Proc. of *IEEE Int. Conf. on Distributed Computing systems (ICDS)*, Vienna(Austria), 2002.

[4] H. Chan and A. Perrig. Security and privacy in sensor networks. *IEEE Computer Magazine*, pp 103-105, 2003.

[5] A. Coen-Porisini, P. Colombo, S. Sicari and A. Trombetta. A Conceptual Model for Privacy Policies. In Proc. of *SEA 2007*, Cambridge, Boston, 2007.

[6] Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. Official Journal of the European Communities of 23 November 1995 No L. 281 p. 31.

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

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

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

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

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

[12] D. Molnar and D. Wagner. Privacy and security in library rfid : Issues, practices, and architectures. In Proc. of *ACM CCS*, 2004

[13] Q. Ni, A. Trombetta, E. Bertino, and J. Lobo. Privacy-aware role-based access control. In Proc. of *ACM Symp. on Access Control Methods And Technologies (SACMAT07)*, 2007.

[14] OMG. Unified Modeling Language: Infrastructure, 2009. Ver. 2.2, formal/2009-02-04

[15] OMG. Unified Modeling Language: Superstructure, 2009. Ver. 2.2, formal/2009-02-02

[16] A. Perrig, R. Szewczyk, J. D. Tygar, V. Wen, and D. E. Culler. Spins: security protocols for sensor networks. *Wireless Networking*, 8(5):521-534, 2002.

[17] N. B. Priyantha, A. Chakraborty, and H. Balakrishnan. The cricket locationsupport system. In Proc. of the *Sixth Annual ACM International Conference on Mobile Computing and Networking (MOBICOM)*, August 2000.

[18] 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.

[19] J. P. M. She and J. T. W. Yeow. Nanotechnology-Enabled Wireless Sensor Networks: From a Device Perspective. *IEEE Sensors Journal*, Vol. 6 (5) Oct. 2006.

[20] E. Snekkenes. Concepts for personal location privacy policies. In Proc. of the *3rd ACM Conf. on Electronic Commerce*, 2001.

[21] H. Zhanga, A. Arorab, Y. Choic, and M.G. Goudac. Reliable bursty convergecast in wireless sensor networks. *Computer Communications*, (Elsevier), 30(13), Pages 2560-2576, 2007.