R. Roman, and J. Lopez, "Integrating Wireless Sensor Networks and the Internet: A Security Analysis", *Internet Research*, vol. 19, pp. 246-259, 2009.
http://doi.org/10.1108/10662240910952373
NICS Lab. Publications: https://www.nics.uma.es/publications

Integrating Wireless Sensor Networks and the Internet: A Security Analysis

Rodrigo Roman and Javier Lopez

Department of Computer Science, University of Malaga, Spain
{roman,jlm}@lcc.uma.es

Abstract

Purpose – To analyze the security issues that arise when integrating Wireless Sensor Networks (WSN) and the Internet. Also, to review whether existing technology mechanisms are suitable and can be applied to this context.

Design / Methodology / Approach – Consider all the possible approaches that can be used to connect a WSN with the Internet, and analyze the security of their interactions.

Findings – By providing the services of the network through a front-end proxy, a sensor network and the Internet can interact securely. There are other challenges to be solved if the sensor nodes are integrated into the Internet infrastructure, although there exists interesting advances on his matter.

Research Limitations / Implications – The complete integration of sensor networks and the Internet still remains as an open issue.

Practical Implications – With the current state of the art, it is possible to develop a secure sensor network that can provide its services to Internet hosts with certain security properties.

Originality / Value – This paper studies the interactions between sensor networks and the Internet from the point of view of security. It identifies both solutions and research challenges.

Keywords – Wireless Sensor Networks, Security, Integration, Internet.

Paper type – Research paper.

1. Introduction

If a computer system needs to obtain data from a certain environment, one of the tools that may be used is wireless sensor networks, also known as sensor networks or WSN. The elements of these networks, the sensor nodes, can measure various physical properties like temperature and radiation, and produce data streams that are sent to a powerful device known as base station. These networks have certain features, such as self-configurability, autonomy, and easiness of deployment, that make them extremely useful for a variety of applications: environmental monitoring, home automation, medical applications, and many others. In fact, WSN are considered as one of the fundamental parts of the "pervasive computing" paradigm, where ubiquitous computers help us in our daily lives.

The issue of accessing to the data streams produced by sensor networks has not been profoundly considered. It is assumed that, once the data is retrieved from the sensors, the user of the network will be able to read it directly through the base station. However, this assumption must be challenged. All data streams produced by any sensor network should be available to any authorized user, being human or machine, anywhere in the world, by using standard mechanisms. By integrating the sensor networks into the

Internet, this vision can become a reality. There are many ways to accomplish this integration: from considering the base station as an Internet host that interfaces with the network services, to including the sensor nodes into the IP infrastructure.

Whatever the case may be, security becomes a really important factor. Sensor networks must be secure by themselves, and the interactions between the sensor network and the Internet must also comply with certain security properties. Sensor network security has received a lot of attention from the research community, but the secure integration of sensor networks and the Internet is an underdeveloped research subject. The purpose of this paper is to analyze which are the major security issues that need to be solved in order to successfully integrate the sensor networks and the Internet. Also, once those security issues are identified, this paper will review if the existing technology, protocols, and implementations are suitable for this purpose.

The structure of the paper is as follows. In section 2 we will review the sensor network technology, and how it could be integrated as part of the Internet. Section 3 will introduce the major security challenges that may hinder this integration process, showing both the sensor network security issues and the integration-specific security issues. Section 4 will analyze whether these security challenges remain to be open problems or can be solved using existing technologies and mechanisms. Finally, section 5 concludes the paper.

2. WSN and the Internet

2.1. WSN Overview

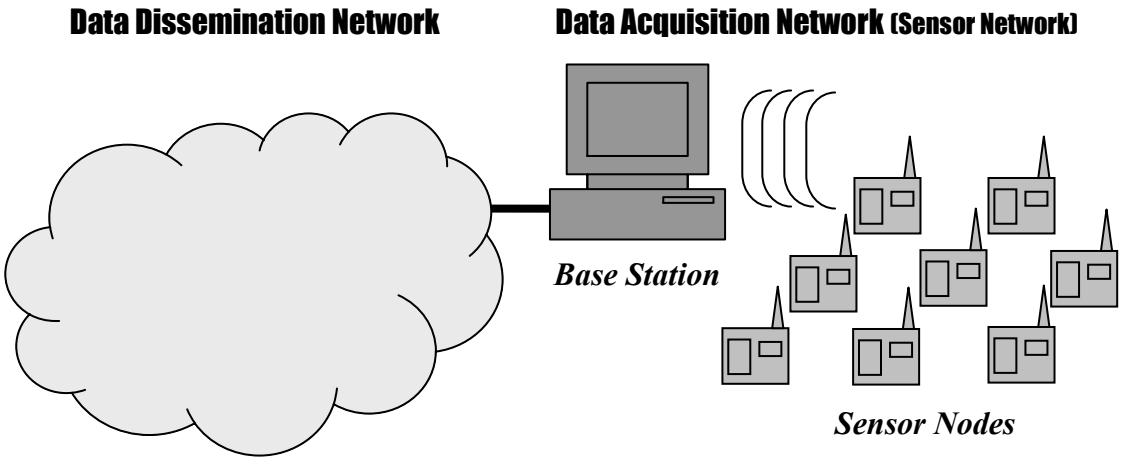

Fig. 1 – Structure of a Wireless Sensor Network

A WSN is an application-centric network, whose main purpose is to obtain physical information from the environment where it is deployed. These types of networks can be abstracted as the "skin" of a computer system, where a high number of "cells", specialized devices known as sensor nodes or motes, perceive physical properties such as temperature, light, radiation, and others. This "sensory data" is usually forwarded to a central "brain" system, known as base station. It will behave as an interface between a sensor network (data acquisition network) and other networks (data dissemination

networks), providing the sensory data to any potential users and allowing such users to modify the state of the network. An overview of the structure of a sensor network can be seen in Fig. 1.

Actually, sensor nodes are small and constrained devices equipped with sensor boards. All sensor nodes are battery-powered, thus they can be able to operate autonomously, if required. Also, they communicate with others nodes using a wireless channel, usually the IEEE 802.15.4 standard. Moreover, since every node has computational capabilities, they can process the sensory data and collaborate with other nodes in pursuing a common goal. At present, there are three classes of sensor nodes, as seen in Table 1. Of these three classes, class II is popularly considered as the example of a typical sensor node. Regarding the base station, is usually a powerful device with PC-like capabilities, although there can exist mobile base stations with the computational capabilities of class III nodes.

	Speed	RAM	ROM	Energy
Class I	4 Mhz	1 kB	4~16 kB	1.5 mA
Class II	4~8 Mhz	4~10 kB	48~128 kB	2~8 mA
Class III	13~180 Mhz	256~512 kB	4~32 MB	~40 mA

Table 1 – Sensor Node Capabilities

The services offered by a WSN can be classified into four major categories: monitoring, alerting, providing information “on-demand”, and actuating. As for the first case, sensor nodes can continuously monitor certain features of their surroundings (e.g. measuring the ambient noise level) and timely send such information to the base station. Secondly, sensors can check whether certain physical circumstances (e.g. a fire) are occurring, alerting the users of the system when an alarm is triggered. In the third case, the network can be queried about the actual levels of a certain feature, providing information “on-demand”. Finally, the network can modify the state of an external system (e.g. an irrigation system) according to the data, effectively going beyond its sensing capabilities.

The structure of a sensor network can be truly decentralized, where all the nodes participate in both the decision-making processes and the internal protocols, like routing. Such structure is called flat configuration. On the other hand, the network can be divided into clusters, or group of nodes, where all the organizational decisions, like data aggregation, are made by a single entity called “cluster head”. This structure is called hierarchical configuration. Notice that the structure of a sensor network can be hybrid with both configurations working at the same time, for example to avoid situations where the “spinal cord” of the network - the cluster heads - fails to work and the information must be routed to the base station.

Besides its sensing capabilities, wireless sensor network have other benefits that make them especially suitable for remote monitoring. A sensor network can be easily set up in any physical context where it is needed. Its sensor nodes are completely autonomous, and can operate without the intervention of a human user or in situations where there is no central management available. Also, they can self-configure themselves in order to react to any changes in the environment. As a result, the network is quite robust against the failure of its components. Concerning network lifetime, an optimized sensor

network can function for long periods of time, ranging from several days to one or two years. Finally, due to the computational capabilities of the nodes, it is possible to reprogram and reconfigure the network during its lifetime.

2.2. Integrating WSN and the Internet

The sensory cells of a living body need a brain that can react to the information coming from the physical world. This simile can also be applied to a sensor network: it needs to interact with an external system in order to be useful. The simplest sensor network deployment consists on a collection of sensor nodes and a base station. Either a computer system or a human being will be connected to that base station, and will make use of the information coming from the sensor nodes. For example, a farmer can inspect the moisture of his crops just by looking at the data collected by the base station.

In order to improve the accessibility and usability of the services provided by a sensor network, those services should become easily accessible from external networks, either through the base station or directly accessing to the sensor nodes. Consequently, the flow of information produced by the network could be accessed and analyzed by different applications situated in diverse geographical locations. Furthermore, an operator could control the sensor network remotely, without physically accessing the deployment field.

However, if those services were to be published by using standard interfaces in a public network infrastructure, their potential would be fully unleashed. Any interaction with a sensor network would go beyond the notion of "remote access", into the realm of "collaboration". Just as a living being's brain pictures the state of its surroundings by receiving information from many different types of sensory cells, the flow of information provided by a single sensor network could be further combined with other data sources to create increasingly complex services. Actual examples may include coordinating patient's health data with the bed occupancy at hospitals in case of disaster (Kambourakis et. al., 2007), and triggering imaging via satellite based on the input of different sensor networks measuring earthquake data (Lees et. al., 2007).

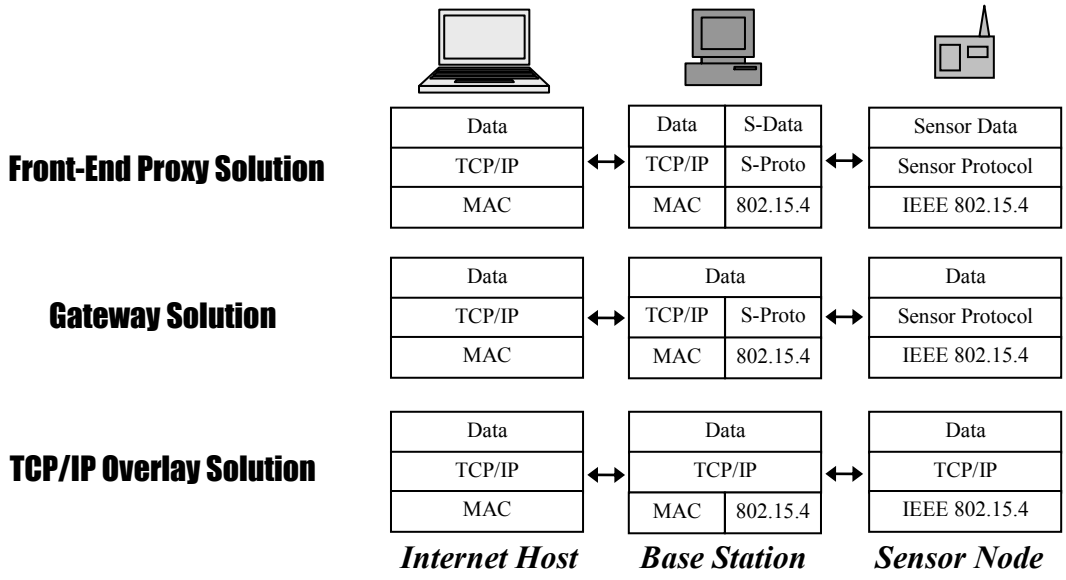

Fig. 2 – Integration strategies

All these visions can become a reality if sensor networks are connected to the Internet. The operational challenges of a completely successful integration are enormous. For example, the vast number of data sources should be harnessed in order to undertake broad scientific studies. Also, there should be mechanisms to discover and tap into real-time data sources. In addition, query processing should also be distributed across the Internet, pushing the query logic into the infrastructure (Welsh, 2005). However, the immediate benefits of a basic integration between sensors and Internet hosts are sufficient enough: any Internet user or device from any geographical location in the world will be able to access to the sensor network services.

There are many approaches that can be used to open the services of a sensor network to the worldwide community. Sensor networks can behave as adjacent "capillary networks" (Privat, 2006) that are not fully integrated in the Internet but provide its services through a standard interface, or can behave as any other IP-based network whose nodes are able to establish direct connections with any other Internet host. Besides, there is a third option, where the sensor network infrastructure retains its independency but allows the establishment of direct connections. These three approaches are shown in Fig. 2, and are discussed below.

In a *front-end proxy solution*, the base station serves as an interface between the data acquisition network (sensor network) and the data dissemination network (the Internet). The base station collects and stores all the information coming from the sensor network, and also sends any control information to the sensor nodes. There is no direct connection between the Internet and a sensor node: all incoming and outgoing information will be parsed by the base station. As the sensor network is completely independent from the Internet, it can implement its own protocols and algorithms. From a functional point of view, the base station can offer the services of its nodes (e.g. data streams) through standard mechanisms such as web services (Kansal et. al., 2007) or web streams (Dickerson et. al., 2008).

In the *gateway solution*, the base station acts as an application layer gateway, in charge of translating the lower layer protocols from both networks (e.g. TCP/IP and proprietary). As a result, the sensor nodes and the Internet hosts can exchange information directly. In this approach, the sensor network can still maintain some of its infrastructural independence, although it is necessary to create a translation table that maps the sensor node addresses to IP addresses. Some web services solutions such as TinyREST (Luckenbach et. al., 2005) take advantage of this approach. There is a small variation of this solution, the *Delay-Tolerant Network (DTN) gateway solution*, where the base station can be able to store and forward packets between the networks. In this case, if the link between the base station and the sensor node is broken, the packet is not transmitted and is stored for future forwarding (Kosanovic and Stojcev, 2007).

Finally, in the *TCP/IP overlay solution*, sensor nodes do communicate with other nodes using TCP/IP. Therefore, the main function of the base station is to behave as a router, forwarding the packets from and to the sensor nodes. These nodes must implement the protocols and standards used on the Internet, such as the TCP/IP stack and web services interfaces. Currently, there exist implementations and specifications for IPv4 (Braun et. al., 2003) and IPv6 (Montenegro et. al., 2007) in sensor nodes, although it is widely assumed that future deployments should be based on IPv6 (Domingues et. al., 2007).

Motes can also provide web services, by reporting its interface using the web service description language (WSDL) and connecting to other hosts using HTTP (Priyantha et. al., 2008).

While the advantages of connecting a sensor network with the Internet are obvious, it is not exactly clear whether sensor nodes should be completely integrated inside the Internet or should maintain its independency as part of a cloud of "adjacent" networks. Some of the infrastructural issues that have to be considered when deploying a sensor network connected to the Internet are discussed below:

- Addressing: In order to send a message to a specific sensor node in the front-end proxy solution, it is necessary to incorporate such functionality into the base station interface (e.g. as a web service). This is not the case of the other solutions, where a sensor node can be directly addressable using IP addresses. Nevertheless, it is not clear how IP addresses could be used in sensor networks that identify its nodes using location information or other data-centric protocols.
- Protocols: By achieving interoperability at the network and application layers (TCP/IP and web services, respectively), it is possible to integrate a sensor network inside the Internet. Besides, this is also beneficial for achieving interoperability between different vendors. On the other hand, by using specific protocols that are tailored to the requirements of the application and the scenario where the sensor network is deployed, such network may be able to react adequately and optimally to application-specific problems.
- Data Availability: When a sensor node is not available, it cannot be possible to obtain neither data streams from its sensors nor historic values stored inside its flash memory. However, if the data stream is accessed through a proxy or a gateway, there can be specific network mechanisms that allow the provisioning of data regardless of the state of the node (e.g. by measuring data from nodes that are in the same context than the broken node).
- Network-specific issues: A sensor network has very specific features that differentiate it from other network paradigms. Its nodes are battery-powered, and in most scenarios they should provide their services for as much time as possible. Besides, in order to save energy, the data rate of the wireless channel is not high (i.e. 250kbps in the IEEE 802.15.4 standard). Therefore, both the sensor network protocols and the applications that use its services should take into account these specific constraints.

3. Security Challenges

3.1. WSN-specific security challenges

Sensor networks are not inherently secure. Its nodes must be deployed near the source of the events, and they use wireless communication channels for exchanging messages. Therefore, any malicious adversary can manipulate the sensor nodes, the environment, or the communication channel on its own benefit. Besides, if that malicious outsider gains access to one or more sensor nodes, it may be possible to manipulate the information flow that traverses the nodes. Therefore, a sensor network must be prepared from the hardware of its nodes to their application layer to prevent or minimize the effect of such attacks (Walters et. al., 2006).

It is indispensable to provide basic security primitives to the sensor nodes in order to give a minimal protection to the information flow and a basis to create secure protocols. Those security primitives are symmetric key encryption schemes (SKE), public key cryptography (PKC), and hash functions. Since most sensor nodes are highly constrained in terms of resources, it is a challenge to implement them in an efficient way. These security primitives also need certain security credentials, i.e. secret keys, in order to work. The task of creating and providing these keys, hence constructing a secure key infrastructure, is done by the Key Management System (KMS). Constructing these KMS is not a trivial task, as they must comply with properties like scalability, communication overhead, connectivity, etc.

This underlying security infrastructure is essential for defending the network against attacks, but it is not enough to protect the entire infrastructure against attacks from the inside of the network. As a result, the most critical protocols of a sensor network must be prepared to deal with malicious activity and node failure as part of their core functionality. Routing algorithms must support full connectivity and network coverage while being fault tolerant. Data aggregation protocols must deal with false data received from faulty nodes or from nodes being controlled by an adversary. Time synchronization protocols must reduce errors must reduce errors related to malicious activity or accumulated propagation delays to the bare minimum. Finally, other protocols like clustering or location and positioning of nodes should be also as secure as possible.

There are other sensor network security aspects that should be also taken into account. For example, a sensor node must be able to discover any abnormal events that are occurring on its neighbourhood, detecting suspicious behaviour in other nodes of the network and protecting the network against any malfunctions. This self-awareness could be used as a foundation for complex security services, such as intrusion detection systems (IDS) and trust architectures. Moreover, other security aspects such as secure management of mobile nodes and base stations, delegation of tasks, data privacy, secure network agents, secure code updating, code attestation, secure random number generation, and many others, should be also considered. Note that the importance of all these security solutions must be consistent with the importance of the processed data and the security requirements of the scenario and the application.

3.2. Integration security challenges

As aforementioned, there are certain security challenges that must be tackled in order to create the foundations of a secure sensor network. However, once a secure sensor network is integrated with the Internet, there will appear new security challenges that need to be taken into account by both sides. As the users of the network will connect remotely to the services provided by the motes, there is the necessity of protecting the information flow. Besides, the aperture of the sensor network services to the whole world community will facilitate the existence of unauthorized users trying to access the network. Therefore, there must exist some mechanisms for authenticating both motes and users, and for checking if an user is authorized to access the mote. Other factors such as availability and accountability need to be considered, as well.

When a sensor node and an Internet host communicate, it is important to set up a secure channel that supports end-to-end integrity and confidentiality services. If the integrity of

the sensory data is protected, attacks targeting the data stream will not be able to falsify any readings. In addition, once the confidentiality of the sensory data is assured, any adversary will not be able to read any sensitive information from the data stream. The creation of a secure communication channel requires of a common secret key between both peers, which should be negotiated using standard Internet security protocols such as SSL/TLS (Dierks and Rescorla, 2006). Note that, in most cases, using pre-shared keys (e.g. as in TLS-PSK (Eronen and Tschofenig, 2005)) is not a feasible solution: sensor nodes should be able to offer their services to any user with a certain degree of flexibility.

It is also important to provide support for device authentication and user authentication. An Internet host should have the certainty that he is reading sensory data from the right mote in the right network, while a sensor node should know that is providing its services to the right client. Again, standard security protocols such as SSL/TLS do provide authentication, although in most cases it requires the sensor nodes to manage digital certificates and to belong to a certain public key infrastructure (PKI). Nonetheless, authenticating the devices that are interacting may not be enough for specific scenarios, where the user that is trying to access the node should also present some credentials.

The authorization problem is closely related to the authentication problem. Once any user of the network (being a human user or a machine) proves its identity, it may be necessary to check whether that user have the rights to access the sensory data. For example, public data such as the temperature of an university building may be accessed by everyone, while other data streams such as the radiation level of a nuclear power plant should only be accessed by those who have enough authority. Not only should the access to the data be controlled, but also the granularity of the data, as well. Beyond data, it is also necessary to watch over control operations: sensors should be remotely reprogrammed and retasked, and only authorized personnel should be able to do so.

Other specific issues that may arise while integrating sensor networks in the Internet are accountability and availability. Since a heterogeneous set of users will be accessing the sensor network services, it would be important to record the interactions with those users for detecting or recreating security incidents. Unlike Internet servers, most sensor nodes have a limited amount of memory available to store transaction state information and descriptions of changes to data. Therefore, it is necessary to devise an optimized way to store this log data, either by storing the information inside a more powerful server or by choosing which kind of interactions should be logged.

Finally, as the main purpose of a sensor network is to provide sensory data to its users, the availability of this data is another sensible subject that must be analyzed in detail. If a sensor node is unavailable due to hardware error or any other malfunction, any external hosts will not be able to query it for data streams. Moreover, the node may be subject to external attacks by malicious hosts, and those attacks can affect the node in a variety of ways: from saturating its scarce resources to exhausting its battery. It is then vital to devise some mechanisms that can protect the nodes and assure the availability of the data streams and other network services.

While protecting the interactions between the sensor networks and the Internet, it is essential to have in mind the inherent limitations of the sensor node hardware. In most

cases, constrained sensor nodes may not have the resources to implement full-fledged Internet protocols. Even more, sensor nodes should spend as less energy as possible during their normal operation, if they are deployed for long periods of time. Therefore, the protocols should be adapted to function optimally in these limited devices. Even more, it should be considered if the sensor network deployment allows the security tasks to be delegated to more powerful devices such as the base station.

4. Security Achievements and Open Issues

4.1. Securing wireless sensor networks

The security of sensor networks has been extensively studied by the research community, and while there are some open problems that remain to be solved, it is now possible to create a sensor network that complies with a minimal set of security properties. On the other hand, the secure integration of sensor networks and the Internet is an underdeveloped research subject, although it is actually possible to set up a sufficiently strong infrastructure where sensor networks can interact securely with Internet hosts. It is the purpose of this section to show these findings, together with some open issues.

For complying with a minimal set of security properties on their internal operations, sensor networks need to use cryptographic primitives, support key management systems, and provide support for self-configurability. The actual sensor hardware is perfectly capable of using cryptographic primitives, such as symmetric key cryptography, public key cryptography, and hash functions. The IEEE 802.15.4 standard does provide HW support for AES-128, although this and other symmetric key algorithms can be perfectly implemented in SW. As shown by (Law et. al., 2006), the implementation of the AES-128 standard requires ~8kB of ROM and ~300 bytes of RAM. Note that there are other block ciphers and stream ciphers like Skipjack (Law, 2006) and RC4 (Choi and Song, 2006) that have smaller memory requirements: ~2600 and 428 bytes of ROM, respectively.

While sensor nodes have been usually considered as highly constrained devices that cannot implement any kind of PKC, new research findings have challenged this assumption. By using Elliptic Curve Cryptography (ECC), it is possible to have support for encryption and decryption (ECIES), signing and verifying (ECDSA), and key establishment (ECDH) in a sensor node. The memory and computational requirements of ECC are quite high, though: in a standard class II mote, the execution time of a single ECDSA signature is around 2 seconds, and the algorithm consumes 17kB of ROM and 1.5kB of RAM (Liu and Ning, 2007). Finally, sensor nodes can implement hash functions such as SHA-1 in only 3kB of ROM.

By implementing ECDH in sensor nodes, it is possible to solve the problem of distributing link-layer keys in a sensor network. Besides, ECC can be an extremely useful tool on the integration with the Internet. Nevertheless, there might be scenarios where the functionality of the application is so complex that sensor nodes do not have enough memory to implement this feature, or where the requirements of the application do not need of the complexity of ECDH. Key Management Systems is still a hot

research topic, although with the current state of the art it is possible to satisfy the requirements of small and simple networks (Alcaraz, 2008).

Cryptography can be used as a foundation for essential security services, such as confidentiality, integrity, and authentication. Still, these services are not enough for supporting one of the sensor network specific properties: self-configurability. To be fully autonomous and self-capable, it is essential for the nodes to be aware of their environment, that is, to recognize certain events that might affect the behaviour of the network. Currently, there exist lightweight situation awareness mechanisms that can detect abnormal events occurring inside the sensor network (Roman et. al., 2008). These mechanisms can also be used for accountability purposes, and as a support for the securing of the "core protocols" of a sensor network: routing, aggregation, and time synchronization.

4.2. Securing the integration strategies

- **Front-end proxy solution.** Although sensor networks must be sufficiently secure by themselves, it is necessary to protect the interactions between the networks and the Internet. In a front-end proxy solution, the base station acts as a representative of all the sensor nodes. It provides all the functionality of the network, behaving as an Internet host. Therefore, most protection mechanisms can be implemented and deployed in the base station. As the Internet and the sensor network are logically separated, it can be possible to protect the information exchange between an Internet host and the base station by using any of the existing security standards, while any interaction between the base station and the sensor nodes can make use of simpler security approaches. Besides, the sensor network can implement specific mechanisms (e.g. sensor redundancy) that make the most of the specific features of the network.

  Note that, in this case, the base station becomes a single point of failure: if the base station is successfully attacked, an adversary may have access to all the information flow. Moreover, if the base station malfunctions, the sensor network will be completely inaccessible. A possible solution is to use multiple base stations with the purpose of improving the availability of the network in case of base station failure and including new features such as load balancing. However, new challenges such as maintaining the information consistency between all the base stations will need to be solved.

  In this front-end proxy solution, Internet hosts can negotiate and establish secure end-to-end channels with the base station by using standard protocols such as TLS in the transport layer or WS-SecureConversation (OASIS, 2007) in the application layer, while the base station and the sensor nodes can make use of simpler mechanisms such as pre-negotiated shared keys, or even public key cryptography if it is available. Regarding authentication, as all nodes are considered to be under control of the base station, such base station can authenticate itself (e.g. present its own digital certificate) on behalf of its sensor nodes. User authentication is also managed by the base station, either by using public key certificates or other authentication mechanisms such as (user, password) pairs or complex protocols like Kerberos (Neuman et. al., 2005).

  About authorization, the base station can either analyze the credentials presented

by the users (e.g. attribute certificates) or check whether the user is authorized to perform certain operations (e.g. by using access control lists – ACL). Afterwards, the base station can change how the network services are provisioned: filtering the data provided by the sensor nodes, allowing the user to change only certain configuration values of the motes, and so on. About accountability, there can be a close collaboration between the sensor nodes and the base station to control and monitor the actual state of the network. The base station can store any interaction between the hosts and the motes, and can also retrieve and analyze any behavioural-related information from the nodes themselves. Finally, the front-end proxy solution alleviates certain problems such as the storage of historic data and the availability of malfunctioning nodes. The base station can behave as a "cache server" by storing the sensor nodes data, although it requires querying the sensor nodes even if there is no data requests from external hosts. Also, it can monitor whether a certain node is accessible or not, forwarding any control information if the node becomes available again.

- **TCP/IP overlay solution.** By considering the Internet and the sensor network as separate entities, it is not necessary to use the already limited resources of the sensor nodes to implement costly Internet standards. However, this situation changes in the TCP/IP overlay solution, where the sensor nodes become Internet hosts. As a result, the sensor network should be no longer treated as an independent entity, and both the protocols and the security mechanisms that are used in the Internet hosts should also be supported by the sensor nodes.

  As of 2008, it is possible to send IPv6 packets through a IEEE 802.15.4 network by using the 6lowpan specification (Montenegro, 2007). For link-layer security, 6lowpan recommends to make use of the security mechanisms defined at the link layer by the IEEE 802.15.4 standard. However, for network layer security, IPsec is currently not supported, and it is not clear whether sensor nodes will be able to support the use of these low-level security mechanisms that have end-to-end properties. In fact, the use of a full-fledged IPsec implementation for sensor nodes is discouraged, and even some incoming standards such as ISA100.11a (ISA, 2008) defines only a simple transport level security above UDP. 6lowpan advises to identify the relevant security model and a preferred set of cipher suites that are appropriate given the constraints (Kushalnagar et. al., 2007).

  Even if there is no support for an end-to-end secure channel at the network layer, most applications still may need to create such channel to protect the information flow. This issue can be partially solved by providing security at the transport layer, using the TLS/SSL standard. A prototype known as Sizzle, developed at Sun Microsystems in 2005 (Gupta et. al., 2005), serves as a proof-of-concept. In order to save enough resources, Sizzle implements only a relatively small set of cryptographic algorithms. In addition, there is no certificate parsing code, thus clients may need to authenticate themselves using other mechanisms such as passwords.

  Precisely, regarding user authentication, it may not be feasible to store user credentials (i.e. user and password pairs) inside a sensor node. In this case, all the sensor nodes that belong to the same network should create a mechanism for storing and maintaining such credentials. The same problem applies to user

authorization: it would be necessary to maintain the access control model in a distributed form, which is an extremely difficult task. A possible solution is to use Kerberos. For authentication, the sensor nodes just need to check the ticket provided by the ticket granting server. Kerberos can also pass authorization information generated by other services. Of course, Kerberos requires continuous availability of a central server, and all devices must maintain their clocks loosely synchronized. Note that PKC solutions (identity certificates, attribute certificates) can also be applied if supported.

The low storage capacity of highly constrained nodes significantly hinders the accountability of the network. Sensor nodes should be intelligent enough to detect an abnormal situation caused by hosts accessing to its services, and only store information related to these incidents. In addition, storage capacity partially influences availability: a single node can only store its readings for a limited time, thus the historic data should be either stored elsewhere or summarized somehow.

- **Gateway solution.** Some of the challenges that are associated with the TCP/IP overlay solution can be partially solved using the gateway solution. The gateway (i.e. the base station) can take the role of storing the accountability information regarding the interactions between hosts and motes. It also can store the historic data of the nodes, if they have the risk of running out of storage space. In addition, it can improve the availability of the network acting as a "cache server" or as an intelligent forwarder, like in the front-end proxy solution. On the other hand, if end-to-end secure channels are negotiated, it should be necessary to implement non-trivial mechanisms in the gateway to parse the information between an Internet host and a sensor node.

It should be pointed out that, in the TCP/IP overlay solution, the router that connects the Internet and the sensor network must translate the IPv6 packets to 6lowpan packets. This translation can be done in the base station, thus the base station could perform the same tasks as the gateway (e.g. accountability, availability). Another aspect that needs to be highlighted is that some sensor networks are composed by different types of nodes, where both cluster heads and certain special nodes that control external systems have higher computational capabilities (i.e. similar to class III nodes). Consequently, it should be studied how to implement full-fledged Internet protocols in these cluster heads and special nodes, and how they could be accessed.

On a final note regarding the implementation of all these mechanisms and protocols, the computational capabilities of modern class II nodes may not be enough to support standard Internet protocols. Contemporary sensor nodes are either limited by low RAM (4kB) or low flash ROM (48kB). One of the 6lowpan stack implementations (Sensinode, 2008) needs ~28kB of ROM. Also, Sizzle (SSL over motes) requires ~16kB of ROM without including certificate parsing code (i.e. managing client-side certificates) and 6lowpan support. Besides, the RAM usage of a basic Sizzle program is very near to 4kB. As a result, it may be necessary to improve the minimal memory capabilities of future class II nodes.

5. Conclusions

The real potential of sensor networks can be fully unleashed if they are connected to the Internet. One of the existing challenges in this area is to assure the existence of certain security properties in the collaboration and integration of sensor networks and Internet hosts. Not only sensor networks should be secure by themselves, but also the interaction between these networks and the Internet should comply with security properties such as confidentiality, integrity, authentication, authorization, accountability, and availability.

This paper has reviewed how sensor networks can securely interact with the Internet, and whether existing technology mechanisms are suitable for this scenario. Currently, it is possible to use the base station as a secure front-end proxy that interfaces with the network services, although it is not possible to open a direct channel between hosts and motes, amongst other issues (single point of failure, information consistency …). Other security challenges appear if the sensor nodes are completely integrated into the Internet: creation of end-to-end secure channels, specific protection mechanisms (against denial of service attacks, battery exhaustion attacks, and many others), availability and accountability, sensor node capabilities, and many others. Still, the benefits of integration surely justify the effort of overcoming these challenges.

Acknowledgements

This work has been partially supported by the ARES CONSOLIDER project (CSD2007-00004) and the CRISIS project (TIN2006-09242).

References

Alcaraz, C. (2008). "KMS CRISIS Guidelines Web Application", available at: http://www.isac.uma.es/CRISIS/tools.html (accessed 23 June 2008).

Braun, T., Voigt, T. and Dunkels, A. (2007), "TCP support for sensor networks", *Proceedings of the Fourth Annual Conference on Wireless on Demand Network Systems and Services (WONS 2007)*, Obergurgl, Austria, pp. 162-169.

Choi, K.J. and Song, J.-I. (2006), "Investigation of feasible cryptographic algorithms for wireless sensor network", *Proceedings of the 8th International Conference on Advanced Communication Technology (ICACT 2006)*, Phoenix Park, Korea, pp. 1379-1381.

Dickerson, R., Lu, J., Lu, J. and Whitehouse, K. (2008), "Stream Feeds: an Abstraction for the World Wide Sensor Web", *Proceedings of the Conference on the Internet of Things (IOT 2008)*, Zurich, Switzerland, pp. 360-375.

Dierks, T. and Rescorla, E. (2006), "RFC 4346: The Transport Layer Security (TLS) Protocol Version 1.1", *Request for Comments*, April 2006.

Domingues, M., Friaças, C. and Veiga, P. (2007), "Is global IPv6 deployment on track?", *Internet Research*, Vol. 17, No. 5, pp. 505-518.

Eronen, P. and Tschofenig, H. (2005), "RFC 4279: Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)", *Request for Comments*, December 2005.

Gupta, V., Wurm, M., Zhu, Y., Millard, M., Fung, S., Gura, N., Eberle, H. and Shantz, S.C. (2005), "Sizzle: A standards-based end-to-end security architecture for the embedded Internet", *Pervasive and Mobile Computing*, Vol. 1, No. 4, pp 425-445.

ISA (2008), "ISA100.11a, Release 1", available at: http://www.isa.org/isa100/ (accessed 23 June 2008).

Kambourakis, G., Klaoudatou, E. and Gritzalis, S. (2007), "Securing Medical Sensor Environments: The CodeBlue Framework Case", *Proceedings of the Second International Conference on Availability, Reliability and Security (ARES 2007)*, Vienna, Austria, pp. 637-643.

Kansal, A., Nath, S., Liu, J. and Zhao, F. (2007), "SenseWeb: An Infrastructure for Shared Sensing", *IEEE Multimedia*, Vol. 14, No. 4, pp. 8-13.

Kosanovic, M. R. and Stojcev, M. K. (2007), "Implementation of TCP/IP Protocols in Wireless Sensor Networks", *Proceedings of the XLII International Scientific Conference on Information, Communication and Energy Systems and Technologies, (ICEST 2007)*, Ohrid, Macedonia, pp. 143-146.

Kushalnagar, N., Montenegro, G. and Schumacher, C. (2007), "RFC 4919: IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals", *Request for Comments*, August 2007.

Law, Y.W., Doumen, J., Hartel, P. (2006), "Survey and Benchmark of Block Ciphers for Wireless Sensor Networks", *ACM Transactions on Sensor Networks*, Vol. 2, No. 1, pp 65-93.

Lees, J. M., Johnson, J. B., Ruiz, M., Troncoso, L. and Welsh, M. (2007), "Reventador Volcano 2005: Eruptive Activity Inferred from Seismo-Acoustic Observation", *Journal of Volcanology and Geothermal Research*, in press.

Liu, A. and Ning, P. (2007), "TinyECC: A Configurable Library for Elliptic Curve Cryptography in Wireless Sensor Networks", *Technical Report TR-2007-36*, North Carolina State University, Department of Computer Science, November 2007.

Luckenbach, T., Gober, P., Arbanowski, S., Kotsopoulos, A. and Kim, K. (2005), "TinyREST – a Protocol for Integrating Sensor Networks into the Internet", *Proceedings of the First Workshop on Real-World Wireless Sensor Networks (REALWSN 2005)*, Stockholm, Sweden.

Montenegro, G., Kushalnagar, N., Hui, J. and Culler, D. (2007), "RFC 4944: Transmission of IPv6 Packets over IEEE 802.15.4 Networks", *Request for Comments*, September 2007.

Neuman, C., Yu, T., Hartman, S. and Raeburn, K. (2005), "RFC 4129: The Kerberos Network Authentication Service (V5)", *Request for Comments*, July 2005.

OASIS Web Services Secure Exchange TC (2007), "OASIS Standard: WS-SecureConversation 1.3", available at: http://www.oasis-open.org/committees/ws-sx/ (accessed 23 June 2008).

Privat, G. (2006), "From Smart Devices to Ambient Communication", *Workshop 'From RFID to the Internet of Things'*, Brussels, Belgium, available at: http://cordis.europa.eu/ist/audiovisual/neweve/e/conf6-70306/conf6-70306.htm (accessed 23 June 2008).

Priyantha, B., Kansal, A., Goraczko, M. and Zhao, F. (2008), "Tiny Web Services for Sensor Device Interoperability", *Proceedings of the ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN 2008)*, St. Louis, USA, pp. 567-568.

Roman, R., Lopez, J. and Gritzalis, S. (2008), "Situation Awareness Mechanisms for Wireless Sensor Networks", *IEEE Communications*, Vol. 46, No. 4, pp. 102-107.

Sensinode Ltd. (2008), "Nanostack, a 6lowpan implementation", available at: http://www.sensinode.com (accessed 23 June 2008).

Walters, J. P., Liang, Z., Shi, W. and Chaudhary, V. (2006), "Wireless Sensor Network Security: A Survey", in Xiao, Y (Ed.), *Security in Distributed, Grid, and Pervasive Computing*, CRC Press, London, pp. 367-410.

Welsh, M. (2005), "Extending the Internet Architecture to Sensor Networks: Some Open Questions", *Microsoft Research Sensor Networks Workshop 2005*, available at: http://www.eecs.harvard.edu/~mdw/ (accessed 23 June 2008).