Journal of Ambient Intelligence and Smart Environments 6 (2014) 157–177
DOI 10.3233/AIS-140250
IOS Press

Securing access to next generation IP-enabled pacemakers and ICDs using Ladon

Jasone Astorga [a,*], Juan Carlos Astorga [b], Eduardo Jacob [a], Nerea Toledo [a] and Marivi Higuero [a]
[a] *Department of Communications Engineering, University of the Basque Country UPV/EHU, Alameda de Urquijo s/n, 48013, Bilbao, Spain*
E-mail: {jasone.astorga,eduardo.jacob,nerea.toledo,marivi.hiquero}@ehu.es
[b] *Cardiology Service, Cruces Hospital, 48903, Barakaldo, Spain*
E-mail: juancarlos.astorgaburgo@osakidetza.net

Abstract. The upcoming development of the Internet of Things (IoT) envisions IP-enabled pacemakers and ICDs, giving place to a completely new scenario in the field of remote monitoring of patients implanted with these devices. Apart from the costs saved thanks to the reduction of in-clinic visits, this new approach will help improving the quality of life of chronic patients that depend on such devices. However, this scenario cannot be conceived without an effective mechanism to protect the privacy of the health information collected by implanted sensors, understanding privacy as the capacity to determine when, how and to what extent information is communicated to others. In this paper, we show how the Ladon authentication, authorization and key establishment protocol can be successfully applied to achieve this purpose. The Ladon protocol is based on Kerberos, but appropriately modified and extended to support independence of clock synchronization and authorization functionalities. In order to demonstrate the feasibility of introducing Ladon in the targeted scenarios, a prototype implementation based on general purpose sensors has been developed. The obtained results show that the performance penalty introduced by the protocol in terms of energy and time consumption is negligible.

Keywords: Authorization, ICD, Kerberos, pacemaker, privacy

1. Introduction

Driven by an ageing society, new and cost-effective solutions are necessary to improve the quality of life of patients with a chronic affliction and at the same time reduce the burden that regular in-clinic visits place on our welfare system. In this sense, an interesting and yet under development research field is the unobtrusive monitoring of implanted wireless medical sensors.

Given that cardiovascular diseases are the first cause of death in modern western societies, the implantation of pacemakers and ICDs (Implantable Cardioverters-Defibrillators), which are complex medical devices proven to reduce mortality in specific high-risk patient populations, are rapidly growing. In this work, both types of devices are considered together because they share similar operating environments and functionalities, although ICDs have the capacity to deliver more complex therapies, such as defibrillation.

Pacemakers were first implanted in humans in the fifties and ICDs about three decades later. However, they have since rapidly and significantly evolved. According to the European Society of Cardiology [12] in 2005 the number of patients with pacemakers implanted was greater than 500 per million of inhabitants in most European countries and the number of patients with ICDs implanted greater than 100 per million of inhabitants; and these figures have since increased.

A pacemaker controls the heart's electric impulses and sends electric pulses to make it beat at a more appropriate rhythm when necessary. ICDs are also responsible for controlling cardiac rhythms and when they detect dangerous paces they send electric shocks.

*Corresponding author. Tel.: +34 946 017 395; fax: +34 946 014 259.

To achieve this operation, they rely on microcontrollers running highly complex algorithms to detect the characteristics of certain sensed parameters and determine the suitable response. Additionally, these devices make a register of the heart's electric activity patterns whenever they detect an abnormal cardiac pace. This information is periodically checked by a doctor to plan future treatment. To that end, the information must be wirelessly transmitted to an external device usually placed within a few centimetres from the patient's skin. Currently this communication is carried out by means of proprietary protocols and technologies.

Pacemakers and ICDs are placed underneath the patient's skin and connected to his heart through special wires. Current common devices weigh a little less than 30 grams (an ounce) and consist of a hermetically sealed pulse generator and isolated conductor cables of electrodes. The pulse generator contains a battery and an electronic circuit responsible for receiving the cardiac activity and generating the pulses. The batteries have an expected useful life of about 6–9 years and their replacement implies opening a wound in the patient's chest. Therefore, power consumption is a key design parameter for protocols and algorithms to be implemented in these devices.

Implantable health sensors are in constant evolution. Future generations are expected to include more sophisticated detection algorithms and longer battery lives. Additionally, in a society where information and communication technologies are revolutionizing every sector of everyday life, it is difficult to believe that the healthcare area will escape this revolution, resulting in a higher quality of care, thanks to for example real-time and ubiquitous monitoring of patients. With the latest advances in microelectronics and communication technologies leading to the commercial availability of IPv6-enabled sensors, it is easy to envision a future with health monitoring sensors connected to the IP world. Despite the huge potential of this approach, a key factor to its success are not only the obvious security-related issues, but the protection of the privacy of the collected data [28], understanding privacy as the capacity to determine for oneself when, how and to what extent information is disclosed to others. In fact, health-related data are a clear example of sensitive data subject to strict privacy regulations in all developed countries.

In this paper, it is shown how a novel authentication and authorization protocol called Ladon [2] can be efficiently used to protect the privacy of the information collected by health-sensing resource-deprived devices, such as pacemakers and ICDs. Finally, a prototype implementation based on general-purpose sensors is presented to demonstrate the feasibility of using the proposed protocol in a real-world deployment.

This article is structured as follows. First, in Section 2, we introduce some background information related to current trends in remote monitoring of pacemakers and ICDs, while in Section 3, we present the advantages of an IOT-based remote monitoring approach. Section 4 deals with existing approaches to protect the privacy of the information retrieved by wireless sensor networks, specifying why they are not suitable for the considered scenario. Then, we present the specific goals pursued by the protocol proposed to protect access to implantable health sensors in Section 5 and we provide a brief description of its architecture and operation in Section 6. Section 7 discusses security and safety considerations regarding the presented protocol, while Section 8 presents a prototype implementation based on general purpose sensors and its performance evaluation. Finally, Section 9 highlights the most remarkable conclusions of our work.

2. Current trends in remote monitoring of pacemakers and ICDs

Today, most patients implanted with pacemakers or ICDs have to take regular check-ups, which are commonly performed by specially trained medical staff at a hospital. More specifically, according to the ACC/AHA/HRS guidelines [14], patients with pacemakers should be followed-up every 3–12 months, and patients with ICDs every 3–6 months. This implies an inconvenience for patients who have to regularly travel to their health institution, especially in the case of elderly or disabled people who suffer from reduced mobility. Additionally, given the broadening indications for implantation of such devices, the management of these patients and their sensors places a heavy burden on outpatient clinics and hospitals, and takes an increasing share of the time of highly qualified medical staff. Moreover, the value added by the specialists is very little, as there is no medical value in automatically retrieving the data stored in a remote device, only to find out that it is operating flawlessly.

In recent years, most major vendors of pacemakers and ICDs have started to commercialize remote administration devices, known as transmitters or patient devices, which wirelessly query the implanted

sensors and retrieve diagnosis data, either with the active participation of the patient (using a wand) or automatically at prearranged time intervals. Automatic wireless monitoring is preferable as it does not rely on the patient's compliance and allows for more frequent communications, which is essential for a high quality of care. Then, these data are transmitted encrypted, and usually through an analogue telephone line, to a central location, where they are processed and provided to the corresponding physicians in a more friendly format, frequently through a web page. Additionally, physicians are alerted by other communication channels (telephone, email, etc.) whenever critical data have been received. Although the devices offered by different vendors are very similar in functionalities and characteristics, they are all based on proprietary protocols, which avoids interoperability and limits competition.

The use of proprietary software and communication protocols has collateral consequences, such as the fact that currently it is difficult for physicians to keep updated in the analysis of the information provided by devices manufactured by different vendors as well as in their configuration mechanisms. This fact results in hospitals standardizing one or two vendors and having a few cardiologists specialized in the use of their devices. The move towards the use of widely deployed standard communication technologies would provide interoperability among wireless medical sensors and promote vendor competition, eventually resulting in more affordable systems.

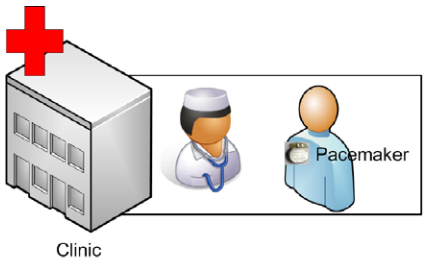
The remote follow-up of implanted patients implies clear economic and medical benefits both for individual patients and for the welfare system as a whole, basically in terms of reducing the number of regular in-clinic visits. Regarding economic benefits, Elsner et al. carried out a thorough study of the data acquired during the "Remote follow-up for ICD-therapy in patients meeting MADIT II criteria" (REFORM) trial [13] and they concluded that the use of remote monitoring saved 71,200 € and 81 hours of highly qualified specialists for 100 patient-year. Similarly, Fauchier et al. [15] studied the cases of 502 patients of six French university hospitals to determine the cost saving that could be achieved by the deployment of remote ICD monitoring. They calculated an average saving of $ 215 per visit. This study took into account physicians' fees and the cost of patients' transportation, weighted according to the distance between the patients' residence and the medical institution. From the medical point of view, remote monitoring allows improving the safety of patients, for example, by the unscheduled transmission of predefined alerts to the physician if some malfunction or medical issue is detected [11]. In addition, it implies clear comfort-related benefits for chronic patients, who can minimize trips to their health institution reducing the impact of their chronic affliction in their way of life.

3. Advantages of an IOT-based remote monitoring approach

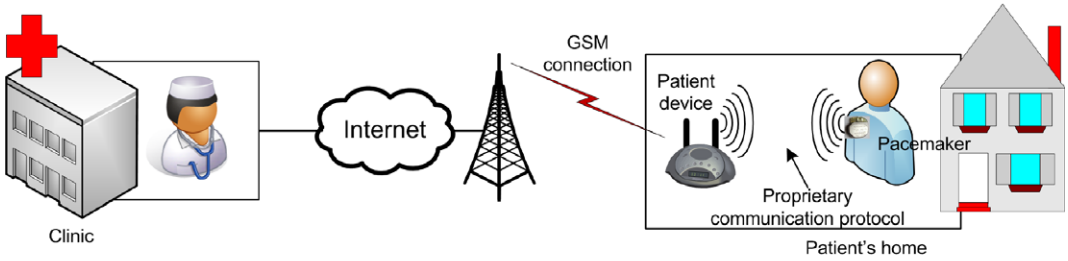
Current remote monitoring systems, based on the use of an intermediate storage element known as patient device, have clear limitations regarding patient mobility. Patients have to be located at most a few metres away from the patient device so that collected data can be directly transferred through a low power wireless communication technology from the health sensor to the intermediate storage device. Then, these data are usually transmitted through an analogue telephone line to a centralized system. To enhance patient mobility, work is being done in improving the portability of patient devices. Therefore, most modern remote monitoring systems consider the use of GSM or other cellular networks for the transmission of patients' data to the centralized processing service. However, patient mobility is still limited because they have to constantly take the patient device with them.

With the upcoming deployment of the Internet of Things (IoT) computation and communication capabilities will be embedded in all kinds of objects and living things. Then, these objects will be linked through low bit-rate and low power multi-hop wireless networks, implemented by means of technologies such as IEEE 802.15.4 [18] radio links. Additionally, these low power networks will also be connected to the Internet using IP, the quintessential protocol in the Internet today, thanks to approaches like 6LoWPAN [19]. Therefore, sensors will be directly addressable by any entity in the Internet, to which they will be connected through a 6LoWPAN bridge, a concept similar to a current WiFi AP. In fact, it is envisioned that in a few years' time IPv6-based sensor networks will proliferate in a way similar to current WiFi networks, providing nearly anytime-anywhere Internet access to any IPv6-enabled sensor.

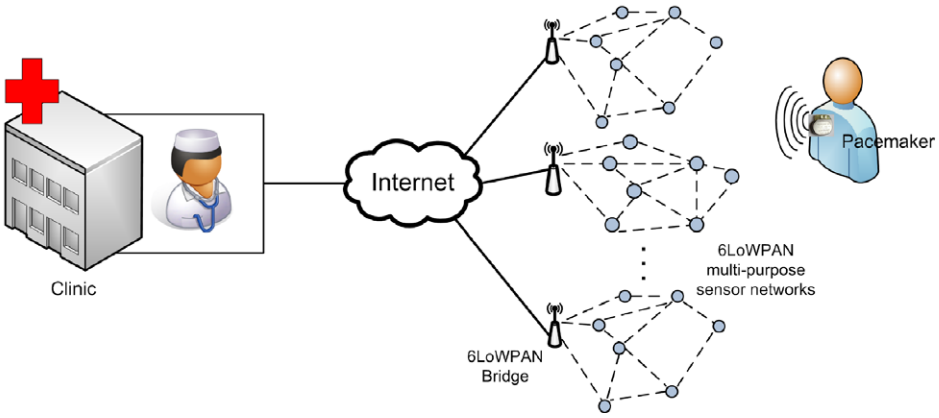
The implementation of technologies like 6LoWPAN and IEEE 802.15.4 in pacemakers and ICDs will imply a further step in the evolution of the monitoring techniques of health sensing devices. Figure 1



(a) In-clinic monitoring scenario.



(b) Classic remote monitoring scenario.



(c) 6LoWPAN-based remote monitoring scenario.

Fig. 1. Evolution of the implanted sensors monitoring scenario.

shows how the scenario for monitoring implanted sensors has evolved from being initially restricted to the physician's surgery, to include also the patient's home, thanks to current remote monitoring systems; and finally, to an ubiquitous scenario where a doctor can have nearly anytime-anywhere access to the data collected by implanted devices thanks to heterogeneous multi-hop sensor networks.

The scenario represented in Fig. 1c eliminates most of the limitations of current remote monitoring systems regarding patient mobility, as it is not necessary to have an external device connected to an analogue telephone line or a cellular network to transmit the data collected by the implanted devices. Instead, these devices will be able to have ubiquitous access to the Internet thanks to the widely available wireless sensors networks (at home, at office, in the car, etc.) that are one of the basic pillars of the Internet of Things. In fact, this scenario change implies moving from a closed, proprietary and expensive solution to an alternative based on open standards and cheap hardware devices. Therefore, patients may have multi-purpose sensor networks deployed anywhere where they spend a substantial amount of time: in their house, at work, in

their car, in their parents' house, in their holiday house, etc. Additionally, these multi-purpose sensor networks may also be deployed by the corresponding authorities in common spaces such as universities, libraries, leisure centres, stations, etc., as it happens today with WiFi networks.

In fact, Internet-connected pacemakers are not a futuristic approach any more, with the first IoT pacemaker implanted in an American woman in 2009 [8]. As explained, thanks to the use of 6LoWPAN, implanted health sensors can be directly connected to current IP infrastructures, reinforcing the end-to-end paradigm of the Internet. Therefore, IP-aware pacemakers and ICDs will be able to directly communicate with the information processing centre, without the need of an intermediary data storage device or patient device, as addressed by existing remote monitoring systems. Thus, it is avoided patients' private information being temporarily stored in a third entity outside the health institution's network, which can be subject to security attacks and vulnerabilities.

Additionally, doctors can have real time access to the data collected by the implanted health sensors, resulting in a better quality medical care. For example, a patient suffering from an abnormal or worrying symptom can telephone the doctor, who will connect to the sensor and have a first diagnosis immediately, assessing the seriousness of the issue and activating the appropriate first-aid procedures accordingly. Another advantage of the presented scenario is that the retrieval of data from the pacemakers or ICDs can be performed automatically, without the cooperation of the patient. This fact avoids the possibility of operational errors due to human factors, especially frequent when elderly people or patients not familiarized with cutting-edge technological developments are involved.

On the other hand, having health sensors based on widely used standard technologies will promote competition, both in hardware design as well as in software development, resulting in more affordable devices and applications. Additionally, making pacemakers and ICDs IP-aware will avoid interoperability problems among different vendors, giving place to a new market for the development of health monitoring applications and allowing for the integration of cutting-edge developments at all times. In fact, given the advantages of using standard communication technologies, in other less critical health-related applications work is already being done towards the use of standard technologies, such as Bluetooth, for the communication of personal health devices [6,20,34].

Nevertheless, having a heart directly connected to the Internet can be scary. Apart from the obvious consequences if an attacker is able to modify some sensor data, patient's privacy is also at stake if the attacker is able to access any information transmitted by the sensor regarding the patient's condition. For this reason, effective mechanisms to guarantee the patient's privacy are essential. The implementation of private communications invariably results in the enforcement of security mechanisms to restrict the access to private data to legitimate entities only. Although the implementation of secure communications is a long researched issue, current security protocols and mechanisms designed for powerful workstations, are not suitable for tiny devices which must operate on small batteries for years. On the other hand, current security mechanisms for sensor networks are also unsuitable for the considered target applications, as they do not addresses the enforcement of end-to-end security services between an entity within a LoWPAN and any other entity in the Internet. For this reason, the Ladon protocol has been developed, a novel privacy-enhancing authentication and authorization protocol, specifically tailored to the characteristics of low capacity devices.

Apart from providing trustworthy real-time information about the legitimacy of every attempt to retrieve data from a sensor, the Ladon protocol implies clear benefits to enhance patients' privacy. First, this protocol allows a real-time and easy management and modification of access rights. Therefore, a doctor can provide temporal access to a patient's health sensor to a colleague in order to exchange points of view or discuss a therapy. In the same way, patients' periodical monitoring can be easily transferred from their regular physician to a different one while the former is on holiday or out of work for some reason. In addition, every time a doctor attempts to obtain the necessary credentials to access the information collected by a given health sensor, the operation is exhaustively logged by a centralized server. In this way, it is possible to have a detailed register of who queried which sensors, when and if the access attempt was successful or not.

4. Review of current approaches to protect privacy of information retrieved by wireless sensors

The problem to solve is how to remotely provide a doctor with the medical data gathered by a health sensing device, while guaranteeing patient's privacy

and conforming to the performance requirements imposed by the hardware and energy limitations of the implanted devices. Among the basic features that must be implemented by any appropriate protocol, the following can be highlighted: authentication and authorization, real-time modification and revocation of permissions, register of whom and when accessed what information, etc.

First, a thorough study of lightweight protocols specifically designed to operate on IEEE 802.15.4 sensor networks is carried out. The goal of these protocols is basically to provide link-layer security in IEEE 802.15.4 radio links. Protocols such as [23,26,27,33, 36] focus on protecting the integrity and confidentiality of the data transmitted between neighbours of a wireless sensor network on a hop-by-hop basis and on guaranteeing that malicious nodes not belonging to the given wireless sensor network are not able to gain access to it. This last goal is generally achieved by means of distributing a group key, useful to distinguish between lawful members of the sensor network and attackers, but it does not allow the establishment of security policies so that the access to specific services provided by some members of the sensor network is restricted to a limited set of some other lawful participants of the network. Additionally, most of these protocols do not define the mechanisms so that legitimate participants can obtain or calculate the necessary keys to afterwards protect information through encryption or calculation of MAC codes.

On the other hand, there are also protocols oriented to the secure and efficient establishment of symmetric cryptographic keys between pairs of nodes within the same sensor network. They generally make use of certain information configured in the nodes prior to their deployment that identifies them as legitimate members of the network and of cryptographic material shared with the base station that acts as border router of the wireless sensor network. Among these types of protocols, two large groups can be distinguished: those that are exclusively based on symmetric key cryptography [10,32,44] and those that make use of public key cryptography [38,42]. Regarding the latter, it must be noted that although in general, public key cryptography is not considered a suitable alternative to operate on sensor devices due to the high computational and storage requirements that it entails [29], recently several public key-based approaches are emerging. The basis of such approaches is to make use of a reduced set of functionalities of this type of technique, and mainly of Elliptic Curve Cryptography (ECC), to efficiently implement authentication and key establishment mechanisms in sensor networks. However, these alternatives still present some of the drawbacks inherent to public key cryptosystems, such as large ciphertext expansion.

Additionally, most of the key establishment protocols specifically tailored to sensor networks rely for their operation on a fixed and predefined structure of the sensor network. That is, for their successful operation they need the sensor network to be organized in a given predefined way, commonly a certain centralized architecture: i.e. low capacity sensors organized into clusters so that each sensor communicates directly just with a higher-level preset entity known as a *clusterhead*. Then, all the cluster-heads communicate among them and with the gateway forming a higher level network [21,32]. Therefore, the applicability of such protocols is limited to sensor networks that conform to the structure defined by the given security protocol.

Therefore, all of these protocols specifically developed for sensor networks focus on protecting the communications at the link-layer level between sensors and the base station or between pairs of nodes within the same network, but none of them considers the possibility that an external entity connected to the Internet would directly query a node within the sensor network. For this reason, it has also been carried out a study of traditional security mechanisms existing in literature for end-to-end information protection between two communicating endpoints.

Among these mechanisms, the implementation of traditional public key infrastructures, meant for powerful workstations, presents big challenges due to the complexity associated to the acquisition of keys and certificates, the verification of revocation lists, etc. With respect to mechanisms that solely make use of symmetric key cryptography, an interesting approach is the well known Kerberos [31] protocol. Although not specifically developed for resource-deprived environments, this widely-used and long-tested protocol is very well suited to the requirements that these scenarios pose, mainly due to its centralized user account management. In fact, there have already been some initiatives to use Kerberos as an authentication mechanism for sensor networks. The results in [1,17,29] present Kerberos as a suitable solution to enable the establishment of pair-wise keys within two sensors of the same sensor network. Thus, they prove the viability of implementing Kerberos in sensor devices.

However, raw Kerberos does not address all the security requirements presented by IP-enabled sen-

sors, basically due to two reasons. First, Kerberos uses timestamps to determine the freshness of tickets and protocol messages. Having the clocks of all possible communicating entities synchronized can be a feasible solution when the operational environment is a controlled scenario. However, it is completely unrealistic that all entities in the Internet will permanently maintain synchronized clocks. Second, Kerberos does not support authorization functionalities. When sensors can be queried by any entity in the Internet, it is crucial to implement reliable and flexible authorization mechanisms, which allow an easy management of rights and permissions. Although since its development, multiple alternatives have been presented to add authorization support to Kerberos [22,30,40,43], all of them are aimed at high-performance machines and they are not suitable to be implemented in resource-deprived devices, such as sensors.

Taking all these reasons into account, a novel strong and clockless security protocol has been developed, based on the Kerberos architecture, but tailored to the specific characteristics of the resource-deprived devices considered in this work: the Ladon protocol [2]. To the best of our knowledge, this is the first protocol specifically designed to protect the data collected by sensors from illegitimate accesses, when these accesses can be originated outside the sensor network.

5. Protocol design goals

In order to guarantee the privacy of the communications between implanted sensors and a third party in the Internet, the use of an end-to-end authentication and authorization protocol is proposed, which restricts the access to the data collected and stored by the health sensors to legitimate authorized entities only. Apart from this main objective, several requirements specific to the characteristics of the targeted environments have been defined. The protocol proposed to secure access to implantable health sensors addresses all of them.

- Energy efficiency: as the targeted devices operate on batteries, the implemented protocol must be energetically efficient. Power consumption is mainly caused by two factors: use of the processor and especially, transmission of data over the air. Thus, to keep energy consumption low, it is essential to minimize the communication overhead.
- Independence of clock synchronization: commonly, sensors do not have a permanently and accurately synchronized time source and thus, the most common way to maintain the clock error within a limited clock-skew window is to periodically query time servers. Additionally, when querying time servers it is also necessary to authenticate their response, using an appropriate security protocol. Therefore, accurate clock synchronization is an undesired requirement for resource-deprived devices.
- Centralized management of roles and permissions: most common authorization models implement the mechanisms to provide the protected systems with the information required to take the authorization decision, but in the end, the actual authorization function must be performed by the end systems, usually by querying local access control lists (ACLs) or remote databases. Maintaining ACLs in resource-deprived devices presents several drawbacks: it implies consuming a part of the scarce ROM memory available, and this memory consumption grows with the length of the list; searching in the list implies a number of CPU operations, that is, power consumption; and whenever a new entity must be given access rights or the permissions of an existing entity modified, it is necessary to interact with the sensor. Querying remote databases is also inadvisable, as it implies increasing the number of messages sent/received by the sensor, what entails higher energy consumption. In addition, these queries should be protected by means of the appropriate security mechanisms, resulting in an even greater communication overhead and energy consumption. Therefore, a centralized and integrated management of authentication and authorization processes is preferable, even when it implies communicating with the centralized system for each access request. One of the main advantages of a centralized authorization management is that it allows to create and enforce dynamic access policies without having to load them individually in each protected pacemaker or ICD. For example, when a different physician must get access to certain patient's sensor or when a doctor leaves the hospital and her credentials must be removed.
- Support for multi-level access policies: the implemented access control mechanism should allow the definition and enforcement of different access levels to the information collected by health sen-

sors. For example, while the regular physician in charge of a patient should have access to all the data collected by the implanted sensor, a nurse in the night shift should only be able to see the values recorded by the device during her shift that exceed a pre-defined alarm threshold.

– Resistance to message losses: due to the intrinsic unreliability of wireless communication links, the implemented mechanisms should be robust against the loss of any protocol message and able to recover from such failure in a graceful way.

6. Protocol description

Being Ladon a protocol based on Kerberos, it is worth reminding some of its basic terminology and features, which will be used afterwards in this paper.

Each client or service is called a *principal* in Kerberos, and each *principal* is characterized by owning a secret key known only by the principal itself and the Kerberos Key Distribution Centre (KDC). The Kerberos authentication mechanism is based on the use of *tickets*. A *ticket* is a capability distributed by the Kerberos KDC that contains a proof of the identity of the principal that requested it. The tickets are encrypted so that only the entities for which they are intended are able to decrypt them. Therefore, each client that wants to authenticate to a server presents a ticket issued by the Kerberos KDC for that service. Specifically, a given client first authenticates against the Kerberos Authentication Server (AS) and obtains a kind of long-term master ticket known as Ticket Granting Ticket (TGT). This ticket allows the client to securely communicate with the Kerberos Ticket Granting Server (TGS), which is in charge of issuing the actual Service Tickets.

As already mentioned, the main advantages of Ladon with respect to Kerberos are support for authorization functionalities and independence of clock synchronization. To implement these new features, the design of Ladon implies the modification of the Kerberos KDC to include two new information stores: (1) an *Active Connections Information Base*, used to assess the freshness of tickets and protocol messages; and (2) an *Authorization Information Base*, used to store the authorization related policies. In addition, three new messages (LDN_AP_IND, LDN_AP_IND_REQ and LDN_AP_IND_REP) have been defined and the original meaning of some Kerberos message fields has been altered. Basically, special *nonces* (unpredictable bit strings) have been introduced to avoid the necessity for synchronized clocks. Although the replacement of timestamps with nonces is a classic technique to avoid time synchronization, the difficulty lies in doing it without increasing the number of messages exchanged by the protocol. In Ladon, this is achieved by the use of one-way key chains. To summarize the presented features, Table 1 provides a concise comparison between the Ladon and Kerberos protocols.

Table 1
Comparison between Kerberos and Ladon

	Kerberos	Ladon
Targeted protected devices	Powerful workstations	Severely resource-deprived devices
Authentication and key establishment functionalities	✓	✓
Authorization functionalities	×	✓
Independence of clock synchronization	×	✓

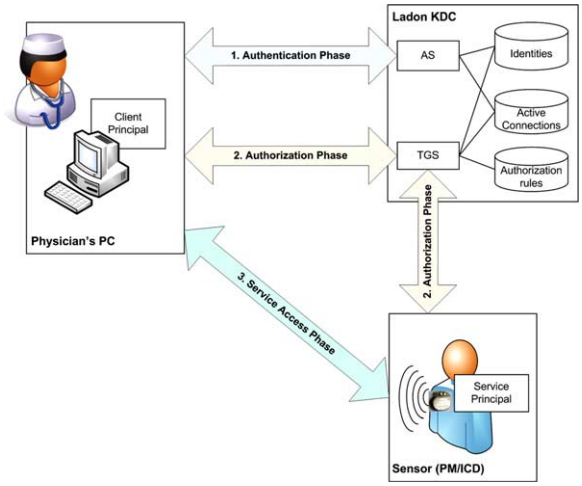


Fig. 2. Basic architecture and operation of the Ladon protocol.

The operation of the Ladon protocol is organized in three different phases: the authentication phase, the authorization phase and the service access phase. Figure 2 graphically depicts the basic architecture of Ladon and its operational phases. In the authentication phase, the Ladon AS verifies the identity claimed by the requesting client principal (the doctor). As a result of a successful authentication, the doctor obtains a TGT which allows him to prove his identity to the Ladon TGS in order to obtain as many Service Tickets as he may need during the validity period of the TGT. During the authorization phase, the Ladon

TGS checks if a legitimately authenticated physician is entitled to access some specific data stored within a given health sensor. The targeted implanted sensor and data are defined as a service principal. If the verification is successful, both the targeted service principal and the requesting physician are provided with the necessary information so that the communication can be successful. Finally, during the service access phase, the doctor presents the credentials provided by the Ladon TGS to the desired service principal, who checks them against the information provided by the Ladon TGS. If they can be positively validated, the health sensor responds with the requested information.

It must be noted that the designed security protocol effectively covers the key establishment and trust setup aspects. Once the legitimate communicating parties own the required secret keys, they can be used to protect subsequent information exchanges between them. In the considered scenarios, the established keys should be used to provide integrity protection and source authentication of all the messages exchanged between client physicians and protected health sensors, as well as to encrypt sensitive information.

In [2], a detailed description of the Ladon protocol is provided, along with a formal security and analytical performance evaluation. Regarding the analytical performance evaluation, it is focused on the analysis of the proposed protocol in terms of storage, communication and computational overhead. The current paper consists of a further research step with respect to the applicability and performance evaluation of Ladon, since it presents a prototype implementation of the Ladon protocol, which gives place to the evaluation of two of the most critical performance parameters for the considered medical applications: end-to-end delay and energy consumption. This evaluation provides a functional validation of the Ladon protocol, proving its feasibility to be introduced in a real scenario of remote monitoring of health sensors.

In the next three subsections a more detailed description of each of the Ladon protocol phases is provided. Table 2 gathers the notation used to describe the protocol, while Fig. 3 depicts the interactions defined by Ladon and Table 3 details the contents of the exchanged messages. Message names have been designed following the nomenclature used in the Kerberos v5 RFC [31], just replacing the KRB prefix with LDN. Note that LDN_AP_IND_REQ and LDN_AP_IND_REP messages have been represented in Fig. 3 with a special format: they have been drawn with a dashed line and assigned numbers 4.1 and 4.2. The reason for this is that these two messages are not sent during the normal operation of the protocol. As it will be explained later in this paper, these two messages are only rarely sent to face high packet error situations.

Table 2
Terminology Summary

Expression	Description
$\mathbf{C}$	Client Principal
$\mathbf{AS}$	Authentication Server
$\mathbf{TGS}$	Ticket Granting Server
$\mathbf{S}$	Service Principal
$\mathbf{K_{X,Y}}$	secret key shared between entities X and Y
$\mathbf{K_X}$	secret key of entity X, shared with the AS
$\mathbf{K^i_{X,Y}}$	i-th value of a one-way key chain used to provide freshness in the communications between entities X and Y
$\mathbf{Ticket_X}$	concatenated information encrypted with K_X that allows a client to authenticate to entity X
$\mathbf{Nonce_i}$	unpredictable bit string used to match a request with its corresponding response
$\mathbf{Nonce_{X,Y}}$	secret data used to prove that the credentials provided by X to Y have not expired
$\{\mathbf{M}\}\mathbf{K_X}$	encryption of message M with secret key K_X
$\mathbf{A}\|\|\mathbf{B}$	concatenation of data field A and data field B
$\mathbf{MAC}(\mathbf{K},\mathbf{M})$	message authentication code (MAC) of message M computed with key K

6.1. Authentication phase

The authentication phase consists of the exchange of the first two protocol messages and its objective is for the client physician to obtain a valid TGT ($Ticket_{TGS}$) and a session key to be shared with the TGS ($K_{C,TGS}$). The TGT is a credential that will allow the physician to communicate with the TGS in an authenticated way afterwards.

The useful life of TGTs is limited by $nonce_{C,TGS}$ values. These nonces are included in the TGTs and stored in the *Active Connections Information Base*, along with the identity of the client principal that requested the TGT. Associated with each entry a counter is established, initially set to $Lifetime_1$, and when this counter expires, the entry is deleted. Thus, the TGS only considers fresh the received TGTs if the $nonce_{C,TGS}$ embedded in the ticket matches the corresponding value stored in the *Active Connections Information Base*.

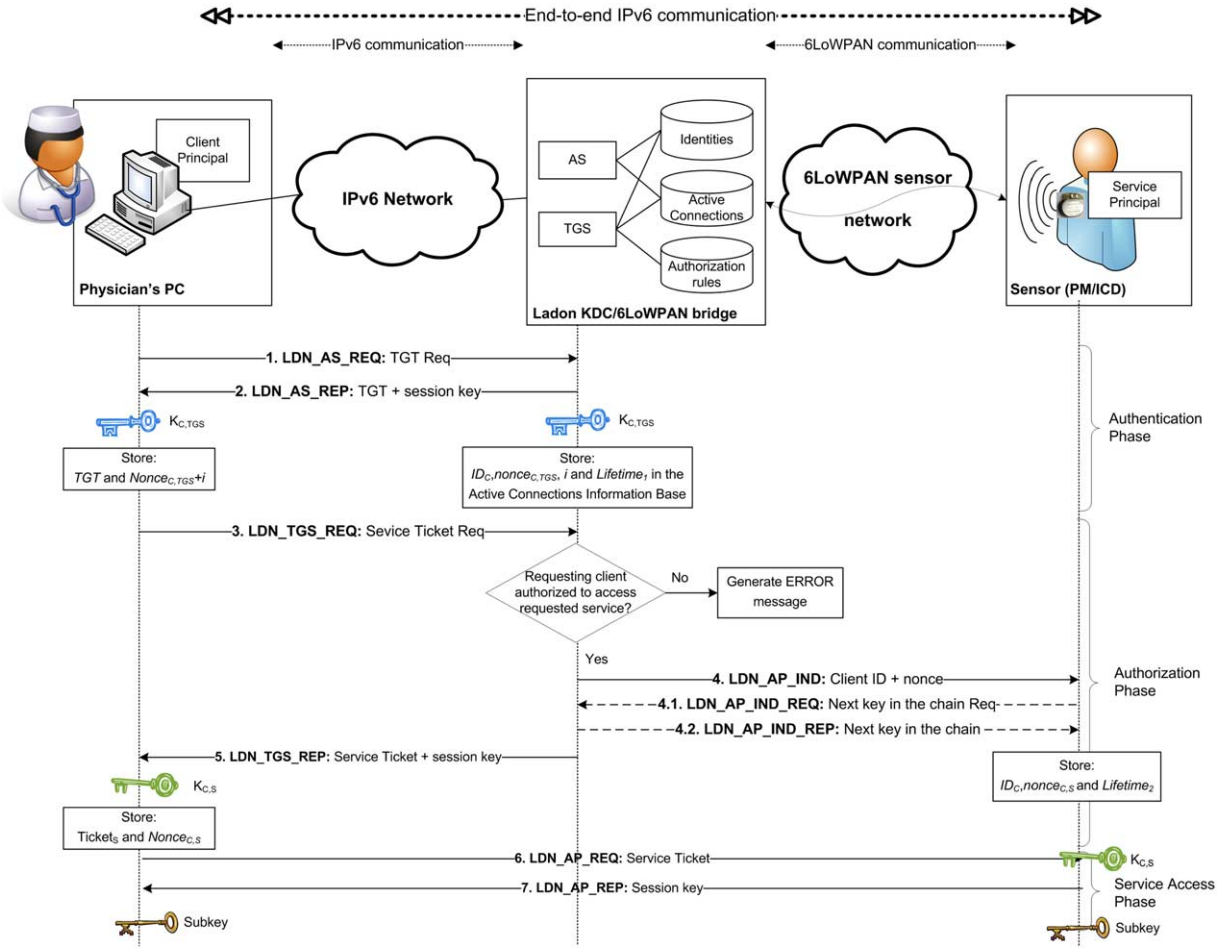


Fig. 3. Detailed operation and messages exchanged by the Ladon protocol.

6.2. *Authorization phase*

The authorization phase includes the exchange of messages 3, 4 and 5. When a doctor wishes to establish an authenticated and authorized session with a given health sensor, first he has to obtain a Service Ticket and the corresponding session key.

In the LDN_TGS_REQ message the doctor authenticates himself by including the TGT obtained in the authentication phase ($Ticket_{TGS}$) and a new *Authenticator*. After validating the LDN_TGS_REQ message, the TGS verifies if the client principal identity specified in this request is authorized to access the desired service principal, that is, to access a given set of information in a specific health sensor. This point constitutes a major difference with respect to Kerberos, where Service Tickets are created for every authenticated client, regardless of whether they have the right to access the requested service or not. Thus, while Kerberos Service Tickets convey authentication credentials, Ladon Service Tickets assert both, the veracity of the identity claimed by the client and his right to access the requested service.

Conveying authorization rights by the mere issuance of Service Tickets can be considered unsuitable, since it is not guaranteed that the authorization process actually took place. For this reason, Ladon Service Tickets have been defined to convey in their authorization payload field the identifier of the role undertaken by the client. This information is individually encrypted with the target service principal's secret key. This encryption is necessary to prevent third parties from generating their own authorization data and sending it to the TGS as encrypted authorization data to be included in the tickets, as defined by the Kerberos v5 RFC.

The proposed authorization model is based on a combined design of RBAC (Role-Based Access Con-

Table 3
Detail of the content of Ladon messages

MESSAGE	DIRECTION	CONTENT
LDN_AS_REQ	$C \rightarrow AS$:	$ID_C \| \| ID_{TGS} \| \| Lifetime_1 \| \| Nonce_1$
LDN_AS_REP	$AS \rightarrow C$:	$ID_C \| \| Ticket_{TGS} \| \| \{K_{C,TGS} \| \| Nonce_{C,TGS} \| \| Nonce_1 \| \| ID_{TGS}\} K_C \quad where,$ $Ticket_{TGS} = \{K_{C,TGS} \| \| ID_C \| \| Nonce_{C,TGS}\} K_{TGS}$
LDN_TGS_REQ	$C \rightarrow TGS$:	$ID_S \| \| Lifetime_2 \| \| Nonce_2 \| \| Ticket_{TGS} \| \| AuthN_{TGS} \quad where,$ $Ticket_{TGS} = \{K_{C,TGS} \| \| ID_C \| \| Nonce_{C,TGS}\} K_{TGS}$ $AuthN_{TGS} = \{ID_C \| \| Nonce_{C,TGS} + i\} K_{C,TGS}$
LDN_AP_IND	$TGS \rightarrow S$:	$ID_S \| \| ID_C \| \| Nonce_{C,S} \| \| Lifetime_2 \| \| K^i_{S,TGS} \| \|$ $MAC(K_S, ID_C \| \| K^i_{S,TGS} \| \| Nonce_{C,S} \| \| Lifetime_2)$
LDN_AP_IND_REQ	$S \rightarrow TGS$:	$ID_S \| \| Nonce_3 \| \| MAC(K_S, ID_S \| \| Nonce_3)$
LDN_AP_IND_REP	$TGS \rightarrow S$:	$ID_S \| \| K^{i+1}_{S,TGS} \| \| MAC(K_S, ID_S \| \| Nonce_3 \| \| K^{i+1}_{S,TGS})$
LDN_TGS_REP	$TGS \rightarrow C$:	$ID_C \| \| Ticket_S \| \| \{K_{C,S} \| \| Nonce_{C,S} \| \| Nonce_2 \| \| ID_S\} K_{C,TGS} \quad where,$ $Ticket_S = \{K_{C,S} \| \| ID_C \| \| Nonce_{C,S} \| \| AuthZ\} K_S$ $AuthZ = \{RoleID\} K_S$
LDN_AP_REQ	$C \rightarrow S$:	$Ticket_S \| \| AuthN_S \| \| Nonce_4 \quad where,$ $Ticket_S = \{K_{C,S} \| \| ID_C \| \| Nonce_{C,S} \| \| AuthZ\} K_S$ $AuthZ = \{RoleID\} K_S$ $AuthN_S = \{ID_C \| \| Nonce_{C,S} \| \| Subkey\} K_{C,S}$
LDN_AP_REP	$S \rightarrow C$:	$\{Nonce_{C,S} \| \| Subkey \| \| Nonce_4\} K_{C,S}$

trol) [16] and ABAC (Attribute-Based Access Control) [41]. It has been designed as an attribute-centric approach where a role is not a collection of permissions, but the name of an attribute called "role". Each service principal is pre-configured with the value of the role attribute that should be granted access to the given set of information provided by that service principal and only accept Service Tickets with that value embedded. Additionally, our model supports the dynamic assignment of roles to users based on contextual attributes. That is, a requesting subject is not directly assigned a role depending just on his identity. Instead, dynamic attributes, such as the time of day, are taken into account to render a final decision regarding the role assigned to the given subject. All the rules and constraints to assign roles to users are stored in the *Authorization Information Base*.

Whenever the TGS creates a new Service Ticket, before transferring this ticket to the requesting client, it sends a LDN_AP_IND message to the target service principal, specifying all the information required to validate the LDN_AP_REQ message that it will receive from the client afterwards. This message does not convey any secret value and thus, it is not encrypted. However, the information regarding the requesting client principal must be authenticated, for which the TGS appends a Message Authentication Code (MAC) calculated with the service principal's secret key.

After validating the received LND_AP_IND message, the service principal stores the ID_C and $nonce_{C,S}$ values embedded in it. However, in order to avoid overflowing the scarce storage capacity of the health sensor, if after a time $Lifetime_2$ no service request is received from the specified client principal, the service principal discards this information.

Finally, the LDN_TGS_REP message returns a Service Ticket ($Ticket_S$) and a block encrypted using the session key shared between the client physician and the TGS. These two encrypted portions convey the session key to be used for the communication between the requesting physician and the service principal at the targeted health sensor ($K_{C,S}$).

6.3. Service access phase

After obtaining the corresponding Service Ticket, the doctor establishes a secure connection with the desired implanted sensor by sending a LDN_AP_REQ message. The validity of LDN_AP_REQ messages

is determined by the $nonce_{C,S}$ value. Basically, the service principal running in the health sensor compares the $nonce_{C,S}$ value included in the Service Ticket and in the *authenticator* with the corresponding value it owns for the requesting client principal identity and only considers the message fresh if they match. Information regarding client principal identities and $nonce_{C,S}$ values is provided by the KDC along with an expiration time ($Lifetime_2$) within LDN_AP_IND messages. To avoid Service Tickets being reused, health sensors delete this information either when $Lifetime_2$ expires or when they receive a Service Ticket that matches the stored information.

7. Security and safety considerations

For its operation in the considered scenarios, our approach makes some crucial considerations. First, the protocol assumes the existence of an underlying routing protocol that enables the successful delivery of packets to their destination. Additionally, it assumes that initially all pacemakers and ICDs share a secret key with the KDC. This key can be loaded into each node prior to their deployment as part of the initial configuration procedure. Finally, our protocol assumes perfect cryptography, which means that an attacker cannot solve encryption without knowing the whole key. As pacemakers and ICDs communicate using wireless links, our protocol does not rely on the underlying communication infrastructure and it assumes that an attacker can eavesdrop on all the transmitted traffic, insert new messages and replay old ones. Taking all these considerations into account, Ladon implements the necessary mechanisms to face all the possible attacks performed by an attacker with the previously defined features.

In [2], it has been already proved that the protocol design conforms to the specified security goals for the targeted scenarios, by means of the AVISPA Project formal validation tool [3]. Therefore, the goal of this paper is not to present a detailed formal description and evaluation of the security of the Ladon protocol. Instead, in the next subsections some insights into the key aspects that make Ladon robust and reliable are given.

7.1. Resilience against message fabrication and modification attacks

Origin authentication and integrity of protocol messages is guaranteed thanks to pair-wise shared session keys: for each message only the legitimate source owns the secret key needed to encrypt some set of information or calculate the corresponding MAC; likewise, for every message, only the intended destination owns the secret key necessary to access confidential information.

Specifically, in the case of LDN_TGS_REQ/_REP messages used by the client principal to obtain Service Tickets, origin authentication and integrity are implicitly guaranteed, as they are in Kerberos, through correct decryption with the appropriate keys. Namely, the keys used to protect these messages from fabrication and modification attacks are K_{TGS} and $K_{C,TGS}$. The TGS relies on the legitimacy of the $Ticket_{TGS}$ presented by the client principal because it is encrypted with K_{TGS}, a key only known by itself and an AS on which it relies. Then, the authenticity of the corresponding authenticator is assessed by decrypting it with $K_{C,TGS}$, a key extracted from the previously validated ticket. This key ($K_{C,TGS}$) is used afterwards to assert the veracity of the LDN_TGS_REP message sent by the TGS to the client principal conveying the requested Service Ticket.

The same reasoning can be applied to the protection of the LDN_AP_REQ/_REP messages that allow a physician to securely access the information provided by a health sensor, replacing K_{TGS} and $K_{C,TGS}$ with K_S and $K_{C,S}$ respectively.

The protection of LDN_AP_IND, LDN_AP_IND_REQ and LDN_AP_IND_REP messages used to securely provide the health sensor with the information needed to validate impending service requests is slightly different, since the legitimacy of these messages is not implicitly assumed by correct decryption with the corresponding key. Instead, each of these messages is accompanied by a suitable MAC which allows the intended recipient to assert the veracity of the received message.

7.2. Resilience against replay attacks

Avoiding replay attacks without using timestamps and without excessively incrementing the number of exchanged messages is not a straightforward issue.

In the case of the LDN_TGS_REQ messages used by the client to ask for a Service Ticket, the TGT by itself is insufficient to assert the freshness of the message, since TGTs are used to provide Single Sign-On functionalities and thus, they can be resent. Therefore, an authenticator is included to prevent invalid replay of tickets by proving to the TGS that the client currently

knows the $K_{C,TGS}$ session key and thus, he is entitled to use the ticket. The implemented authenticators are based on a shared counter kept independently by both endpoints of the communication ($nonce_{C,TGS} + i$). Therefore, each time a client principal needs to generate a new authenticator, it increments the counter by one and includes the updated value in the authenticator. The TGS, in turn, updates the same counter with every received message and rejects messages with an embedded $nonce_{C,TGS} + i$ value lower than the expected one.

The mechanism implemented to avoid the replay of old LDN_AP_REQ messages used to ask for access to a protected service principal is simpler, based on the fact that Service Tickets are not reusable credentials, but single-use. When a service principal receives a legitimate LDN_AP_REQ message, it deletes the information it owns to validate this message and thus, further instances of the same message will be considered unacceptable.

Regarding LDN_AP_IND messages sent by the TGS to provide the sensor with the necessary information to validate subsequent service requests, using a monotonically increasing counter, as when protecting LDN_TGS_REQ messages, is not suitable. The reason for this is that in the case of RAM-constrained sensor nodes, maintaining a table with the last counter value used by every client becomes problematic, even in modestly sized networks. This problem becomes even greater when the sensor cannot identify beforehand all potential clients. Therefore, to avoid replay of LDN_AP_IND messages, a mechanism based on a one-way key function is implemented. Remind that for a one-way key function F, it is relatively easy to compute forward (to obtain K^{L-1} given K^L), but it is computationally unfeasible to compute backward (to obtain K^{L+1} given K^L). Therefore, once the service principal owns a value $K^{i-1}_{S,TGS}$, it is enough to check that $F(K^i_{S,TGS}) = K^{i-1}_{S,TGS}$ to assert the freshness of the message conveying the $K^i_{S,TGS}$ value. The problem here lies in how to provide each service principal in an authenticated way with the first value $K^0_{S,TGS}$ necessary to validate the rest of the key chain. To this end, an initialization mechanism consisting of two messages (LDN_AP_IND_REQ/LDN_AP_IND_REP) has been designed. By means of this message exchange the protected device directly queries the KDC about the next value of the key chain to be used.

In fact, the loss of a message of this type is actually detected by the service principal when it receives a LDN_AP_IND with a $K^j_{S,TGS}$ value that does not match the condition $F(K^j_{S,TGS}) = K^i_{S,TGS}$. In such a case, the service principal repeatedly applies the one-way function to the received $K^j_{S,TGS}$ value up to a maximum number of attempts. If any of the obtained results matches the stored $K^i_{S,TGS}$ value, the service principal accepts the received LDN_AP_IND message as fresh and updates the stored $K^i_{S,TGS}$ value with the $K^j_{S,TGS}$ value embedded in this message. That is, suppose that a given service principal owns a $K^i_{S,TGS}$ value when it receives a LDN_AP_IND message with the $K^{i+2}_{S,TGS}$ value embedded, because the message conveying the $K^{i+1}_{S,TGS}$ value was lost during transmission. In such a case, $F(K^{i+2}_{S,TGS})$ will not match the stored value ($K^i_{S,TGS}$). However, the service principal will be able to validate the received message by checking that $F(F(K^{i+2}_{S,TGS})) = K^i_{S,TGS}$. This recovery mechanism is more efficient from the time and power consumption points of view, than any other procedure involving message retransmissions, as it minimizes the use of the antenna, which is one of the main causes of energy consumption.

Nevertheless, in the case of a network with a very high packet loss rate, it can happen that after computing the one-way function for the maximum allowed number of attempts, the service principal is still unable to validate the received $K^j_{S,TGS}$ value. In such a case, it will be necessary to start a more expensive recovery mechanism, which consists of the LDN_AP_IND_REQ/_REP message exchange.

7.3. Other security and safety considerations

Although sensors are subject to more intricate security attacks, such as resource consumption attacks or denial of service attacks, our protocol is not aimed at addressing any of these problems. All these attacks will be detected by the doctor or the system in charge of automatically retrieving the information collected by health sensors due to a communication failure or an abnormal battery consumption. Therefore, it is argued that these types of attacks could be best coped with at the application layer. That is, the distributed applications used to retrieve data from pacemakers and ICDs should be robust and designed in a way that enables them to detect such kinds of attacks and notify the doctor or the appropriate medical technician in charge of pacemakers' or ICDs' maintenance accordingly.

Other security considerations that affect patients' safety could be the case of an attacker that achieved

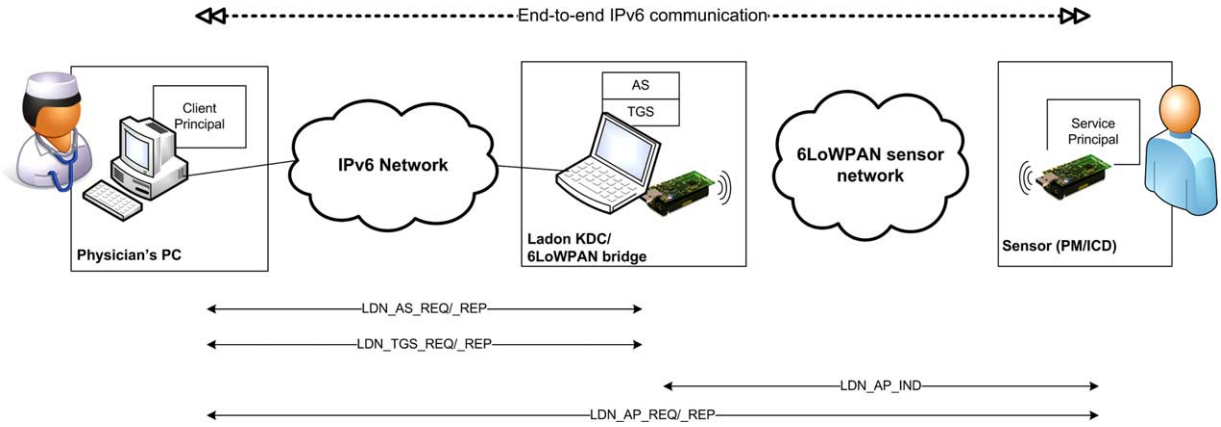


Fig. 4. Experimental setup.

to maliciously configure a pacemaker or ICD. Such an attack could harm a patient either by causing inaction (failing to deliver therapy when necessary) or excessive action (delivering therapy even in normal cardiac conditions). As in the previous case, such attacks could be best coped with at the application layer, developing robust applications that would not allow any remote configuration operation. That is, before effectively applying any change to the health sensor's settings, the application should ask for some kind of confirmation that would ensure that the doctor is physically close to the patient.

8. Implementation and performance evaluation

The objective of the experimental setup is to test the viability of the described privacy-enhancing authentication, authorization and key exchange protocol. To that end, the performance penalty introduced by the protocol has been evaluated in terms of memory footprint, and time and energy consumption.

8.1. Testbed implementation

The evaluation setup, illustrated in Fig. 4, consists of a TelosB sensor node representing a protected pacemaker or ICD, which runs the service principal application. This device features an IEEE 802.15.4 compliant CC2420 transceiver and implements 6LoWPAN. A second TelosB sensor node is connected to a laptop running Ubuntu OS, implementing a 6LoWPAN software bridge. This same laptop runs also the two C programs implementing the AS and TGS processes. Finally, a remote computer connected to the IPv6-enabled research network of the University of the Basque Country executes the client principal application.

The client application first obtains the necessary keys and tickets to securely communicate with the targeted service principal at the remote sensor using Ladon and then queries the remote sensor device by sending a legitimate LDN_AP_REQ message to which it appends an application-specific query. The remote sensor node listens on a preset UDP port and only responds with a LDN_AP_REP message to successfully authenticated and authorized requests. Additionally, it appends to the reply message a preconfigured list of information, representing the data collected by the remote pacemaker or ICD. This information is encrypted using the *subkey* specified in the LDN_AP_REP message.

8.2. Protocol implementation decisions

An important design decision has been not to reuse any of the existing Kerberos implementations due to the high resource limitations of the targeted devices in comparison with the code size and computational requirements of existing implementations, designed to operate on multiple platforms and to support all the options considered in the standard. For the same reason, a binary codification of Ladon messages has been designed, instead of following the ASN.1 codification, which results in a significant reduction of the size of the messages sent and received by the sensors. As an specific example, a simple Kerberos KRB_AS_REQ message including the same fields as our LDN_AS_REQ message, but encoded following the ASN.1 codification and with client and prin-

Table 4
Lengths of Ladon protocol messages

Message type	Length (bytes)
LDN_AS_REQ	15
LDN_AS_REP	62
LDN_TGS_REQ	47
LDN_AP_IND	33
LDN_AP_IND_REQ	14
LDN_AP_IND_REP	22
LDN_TGS_REP	63
LDN_AP_REQ	61
LDN_AP_REP	32

cipal identities of a maximum length of 20 ASCII characters, would result in a 74-byte message, while a LDN_AS_REQ message is encoded in 15 bytes. Therefore, our protocol implementation sacrifices generality for performance, resulting in a tailored codification of the protocol, which optimizes efficiency. In fact, the used binary codification allows reducing the size of the exchanged messages in about an 80%. Table 4 summarizes the lengths of Ladon protocol messages.

As the implementation and execution of cryptographic algorithms entails by nature high resource consumption, another crucial implementation decision is the one regarding the encryption algorithm to be used. Cryptographic operations will demand extra processor and RAM use, which will result in an increased latency, and more importantly, an increased power consumption. Therefore, a thorough study has been carried out in order to select the most appropriate cryptographic algorithms and options for the targeted scenarios.

Although in general terms stream ciphers are faster than block ciphers [39], they present a devastating drawback: if for some reason the same IV (Initialization Vector) is ever used to encrypt two independent packets, then it is possible to recover both plaintexts. As in Ladon counters are used as IVs, this fact makes the use of stream ciphers unacceptable. Regarding key length, the focus has been to use secure enough keys, but without penalizing performance for free. According to Lenstra [24], a 128-bit symmetric cipher is supposed to remain secure against mathematics attacks until at least 2090.

Following these directives, a testbed study of three of the most popular block ciphers has been conducted, namely 3DES, RC5 and AES. To implement these cryptographic algorithms a reduced version of the open source C cryptographic libraries provided by the

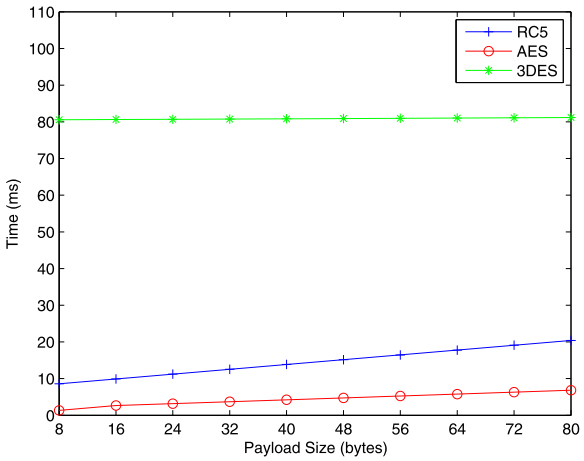


Fig. 5. Comparison of the time needed to encrypt different size payloads with 3DES, RC5 and AES block ciphers.

LibTom project [25] has been used. The study consists of the encryption of different size payloads with each of the evaluated block ciphers. The obtained results are summarized in Fig. 5, which represents the average execution times calculated after repeating each encryption test 500 times.

3DES has been directly discarded for being by far the slowest one. With respect to RC5 and AES, they do not show a significant difference regarding encryption time. Therefore, RC5 has been preferred to AES, as its minimum allowed block size (8 bytes) is smaller than the minimum block size allowed by AES (16 bytes). This fact implies that when encrypting short messages (8 bytes or less) the output provided by RC5 is half the size the output provided by AES. In this way, when encrypting individual fields or short parts of messages the number of bytes to be transmitted over the air is consequently reduced, and so it is the energy consumed by the transceiver.

For messages longer than one block size, the *ciphertext stealing* technique [37] is used to ensure that the ciphertext is the same length as the corresponding plaintext.

Regarding encryption modes, the choice has been to use CBC [4]. First, this mode of operation allows the implementation of the previously mentioned ciphertext stealing technique. Additionally, it allows for memory saving, as the same code can be reused for message authentication by using the CBC-MAC construction [5]. However, CBC was designed to be used with random IVs and it presents a leakage problem when used with counters as IVs. Fortunately, this problem can be easily

dealt with by pre-encrypting the counter value before using it as IV.

Regarding the overall implementation of the Ladon protocol, it has been developed as four independent processes: the client principal, the AS, the TGS and the service principal. The first three processes have been implemented in C, while the latter has been developed using NesC, a C dialect optimized for embedded systems, which is the default programming language of TinyOS, a widely used operating system for sensors.

8.3. Performance metrics

To evaluate the performance of the Ladon protocol it has been measured the additional time and energy consumption that its implementation implies for the protected sensors, which are the most resource-deprived participants of the protocol. Additionally, as sensors are characterized by having limited memories, it has also been measured the memory footprint of the resulting service principal application. Regarding the overall performance of the protocol, it has also been evaluated the delay introduced by the Ladon message exchanges when a given client attempts to access the information collected by a sensor. Therefore, the measured performance metrics can be summarized as follows:

- Energy consumption in the sensor (E_S): this is in fact the most important performance parameter. It represents the energy consumed by the sensor due to the implementation of the Ladon protocol, for each service request. It takes into account both the processor use as well as the energy consumed by the transceiver to transmit/receive bits over the air. Therefore, this parameter is calculated as a summation of the energy consumed by the transceiver during the reception of LDN_AP_IND and LDN_AP_REQ messages, the energy needed by the CPU to process these two messages, the energy consumption related to the generation of the LDN_AP_REP message and the energy needed to transmit this last message.

$$E_S = E_{Rx}^{AP\_IND} + E_{CPU}^{AP\_IND} + E_{Rx}^{AP\_REQ} + E_{CPU}^{AP\_REQ} + E_{CPU}^{AP\_REP} + E_{Tx}^{AP\_REP}$$

- Time consumption in the sensor (T_S): time spent by the sensor in performing Ladon related tasks, for each service request. It entails the time needed by the CPU to process the involved messages, as well as the time needed by the transceiver to receive/send those messages. More specifically, this parameter takes into account the time needed to receive and process the LDN_AP_IND and LDN_AP_REQ messages and to generate and transmit the LDN_AP_REP message.

$$T_S = T_{Rx}^{AP\_IND} + T_{CPU}^{AP\_IND} + T_{Rx}^{AP\_REQ} + T_{CPU}^{AP\_REQ} + T_{CPU}^{AP\_REP} + T_{Tx}^{AP\_REP}$$

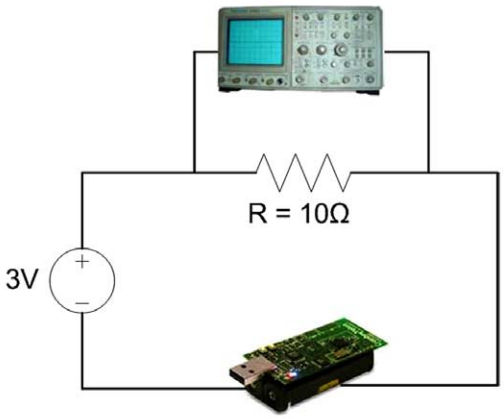
- Memory footprint: RAM and ROM capacity consumed by the Ladon service principal application. These data are directly provided by the application used to install the service principal application in the sensor to be protected.
- Delay for information retrieval operations (T_C): time needed each time a client attempts to access the information collected by a sensor due to Ladon-related security tasks.

8.4. Obtained results

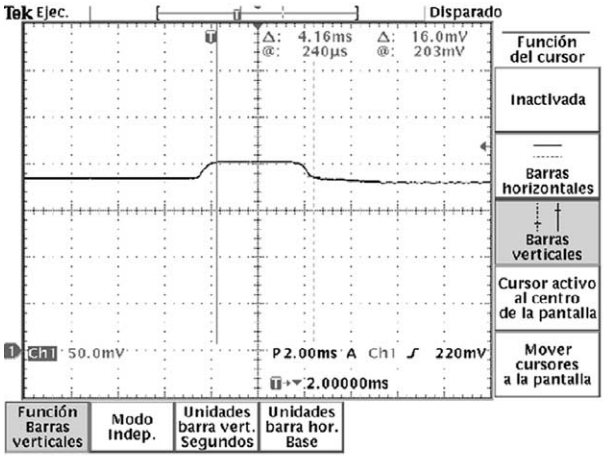
The implemented protocol results in a ROM and RAM footprint of 30.9 and 2.4 KB respectively.

As previously mentioned, to evaluate the added time and energy consumption in the sensor, both the use of the processor and the reception and transmission of bits over the air are taken into account. Regarding the processor use, timers are set within the code to count the CPU cycles corresponding to the execution of each of the security related tasks. On the other hand, to measure the energy consumed by the sensor when performing CPU operations, and the time and energy consumed while transmitting and receiving messages, an experimental measurement set-up has been used. It consists of an adjustable and stabilized power supply and an oscilloscope, connected to the sensor as shown in Fig. 6a. Figure 6b illustrates an example of the data provided by the oscilloscope while the sensor transmits a 40-byte message, while Fig. 6c presents the information provided by the oscilloscope while the sensor is performing the processing of different messages. As shown in Fig. 6c, the overall power consumption of the sensor while using the CPU remains constant independently of the specific microcode operation performed at each time.

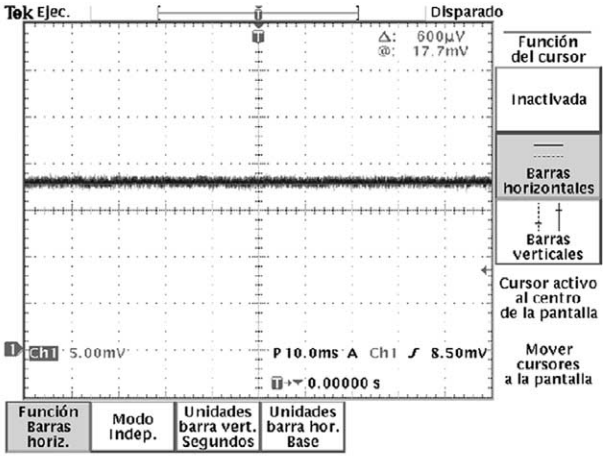
To evaluate the cost of Ladon regarding CPU related operations, CPU ticks are counted specifically from the reception of a LDN_AP_IND message until it is successfully validated and then, from the reception of a LDN_AP_REQ until the corresponding LDN_AP_REP is transmitted. Then, the corresponding energy consumption is calculated using the mean



(a) Set-up for the measurement of energy consumption during data transmission



(b) Information provided by the oscilloscope during the transmission of a 40-byte message



(c) Information provided by the oscilloscope during the processing of different Ladon protocol messages

Fig. 6. Measurement of energy consumed during transmission.

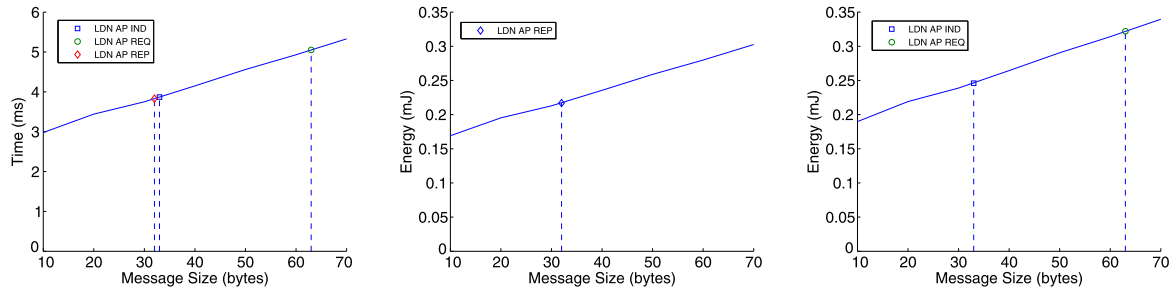
power consumption obtained from Fig. 6c. The results are shown in Table 5 and they correspond to the average of 20 runs of the protocol.

To evaluate the cost of Ladon with respect to communication operations, the set-up of Fig. 6a is used to measure the time and energy cost of transmitting and receiving the different protocol messages. Figures 7a, 7b and 7c represent respectively the average time and energy consumption for the transmission and reception of different size messages in the range of the messages exchanged by Ladon. As shown in these figures, the energy and time consumption has a fixed cost and grows linearly with the data size. Note that the specified message lengths correspond to application layer messages. Therefore, to calculate the total amount of bits transmitted to the air, the overhead introduced by the headers of lower layer protocols (UDP, 6LoWPAN and IEEE 802.15.4 MAC/PHY) must be added. In total, this gives an additional overhead of 42 bytes.

Table 5
Time and energy consumption of Ladon-related CPU operations

Processing phase	CPU ticks	Time (ms)	Energy (mJ)
LDN_AP_IND	652	19.89	0.105
LDN_AP_REQ/_REP	962	29.36	0.155

According to the results showed in Table 5 and Fig. 7, the performance penalty introduced by the Ladon protocol in the protected sensors for each remote monitoring operation is calculated. The specific experimentally measured values used for this calculation are shown in Table 6.



(a) Time consumption (b) Energy consumption (Tx) (c) Energy consumption (Rx)

Fig. 7. Time and energy consumption for the transmission of different size messages.

Table 6
Measured values of the sensor performance metrics

Operation	Message	Time (ms)	Energy (mJ)
R_x	LDN_AP_IND	3.8	0.25
	LDN_AP_REQ	5	0.32
T_x	LDN_AP_REP	3.8	0.22
CPU	LDN_AP_IND	19.89	0.10
	LDN_AP_REQ/_REP	29.36	0.15

Table 7
Measured delays for information retrieval operations

Interaction type	Time (ms)
LDN_AS_REQ – LDN_AP_REP	193.514
LDN_TGS_REQ – LDN_AP_REP	179.500
No security	83.875

Regarding the delay suffered by the client for each information retrieval operation, real-time timers have been set to measure the time elapsed since the client asks for a TGT by means of a LDN_AS_REQ, until he successfully receives the desired information appended to a LDN_AP_REP message. To account just for the time needed by the security tasks, it has also been measured the time needed by the same client to retrieve the same information from the sensor, but without the implementation of any security protocol. The difference between both values represents the time added by the security protocol. Additionally, for comparative purposes, it has also been measured the time elapsed when the client already owns the necessary TGT to directly ask for the desired Service Ticket. That is, the time elapsed since the client sends the LDN_TGS_REQ message until he receives the LDN_AP_REP message. The obtained results are summarized in Table 7.

Finally, Table 8 provides a summing up of the obtained results.

Table 8
Summary of measured performance values

Performance metric		Value
Memory footprint	ROM	30.9 KB
	RAM	2.4 KB
Information retrieval delay	without TGT	109.639 ms
	with TGT	95.625 ms
Time Consumption in the sensor (T_S)		61.85 ms
Energy Consumption in the sensor (E_S)		1.04 mJ

8.5. Discussion

First, it is worth comparing the measured overall energy consumption of the Ladon protocol with that of other security protocols aimed at similar purposes. More specifically, we compare Ladon with two Kerberos implementations in sensor networks. The authors in [17] present a reduced version of Kerberos and they calculate the total energy consumption that a Kerberos execution implies for the overall sensor network. That is, they consider that not only the protected service principal is implemented by a sensor, but that the client principal and the trusted third party are also sensors. The obtained power consumption values for the tasks that imply the participation of the protected service principal are higher than the ones measured for Ladon, although they remain within the same order of magnitude. This difference can be caused by the increased average power consumption of the sensor node used in [17] (WINS) with respect to the TelosB sensor node. Additionally, regarding the size of the Kerberos messages used in [17], specifically the KRB_AP_REP consists just of 8 bytes. The reduction of this message to 8 bytes entails also a reduction of the protocol functionalities, as it implies eliminating the *subkey* field. Therefore, the service principal is no longer allowed to propose a key to the client principal for the pro-

tection of further communications. Both Kerberos v5 and Ladon allow the service principal to send a key to the client principal within the corresponding AP_REP message.

In a similar study, Meulenaer et al. [29] measured the energy consumed by the execution of a reduced version of Kerberos in a TelosB sensor node. Regarding the energy cost of CPU related operations, they just took into account the encryption of 8 blocks of 128 bits using AES-128. This gave them an estimated energy consumption of 0.14 mJ. Regarding transmission and reception operations, they measured an energy consumption of 2.4 mJ, which is the result of a higher amount of transmitted and received protocol bits (1568 bits in total) in comparison with the 1024 bits transmitted/received in the Ladon protocol. Additionally, an important difference between the performance evaluations presented in [17] and [29], and the one carried out in this paper is that in [17] and [29] the energy consumed by the sensors is not actually measured, but estimated through energy models. Finally, it is also worth mentioning that in our work the performance penalty is only measured for the protected sensor, which represents the actual pacemaker or ICD to be protected. As critical medical sensors will never be used as intermediate routing entities of a sensor network, unlike in [17], our evaluation does not consider any energy consumption due to multi-hop routing.

Next, to evaluate the effect that the introduction of the Ladon protocol may have in the battery lives of health sensors, the consumption of the Ladon protocol is compared with the typical consumption range of a common pacemaker when releasing its therapy. Currently used pacemakers and ICDs are based on proprietary designs, being very difficult to obtain details about their hardware specifications. Additionally, the power consumed by these devices strongly depends on the configured lead impedances and thresholds and on the patient's dependence on the device. Typical lead impedances are of 250–750 Ω and thresholds are usually configured so that the device output is 1.5–5 volts with a pulse width of 0.1 to 0.5 ms, when it is delivering a common pacemaker therapy (Dr. Iñigo Sainz, electrophysiologist, personal communication, October 20, 2011). Following basic rules for electrical calculations, this gives a current draw range of 2–20 mA.

These devices are designed so that they can operate for years even in patients which are completely dependent on their pacemaker or ICD. Therefore, in the worst case, a patient will receive therapy a total of 1440 minutes a day, with the pacemaker generating a pulse every second. That would give a total pulse time of 8.64–43.2 s and consequently, an energy consumption of 0.26–4.32 J per day. Therefore, even if the data collected by the health sensors are daily downloaded by the doctor, the penalty introduced by our protocol regarding energy consumption is negligible compared to the energy consumed by a pacemaker when releasing its therapy.

Additionally, transceivers specifically designed for health-sensing devices are orders of magnitude energetically more efficient than transceivers used in general purpose sensors [7]. Consequently, the power consumption and thus, the battery life of these devices will be fully determined by the patient's pathology, which is a rational conclusion taking into account the final goal of the considered devices.

It is also worth noting that the data files generated by common devices (e.g., the .pdd files generated by Medtronic devices) have a typical size of 40–100 KB. Therefore, when transmitting such a file, the power consumed by the few bytes of the Ladon protocol messages can be considered negligible.

Finally, the low latency introduced by the Ladon protocol from the client or physician point of view (about 100 ms) makes it a suitable authentication, authorization and key exchange protocol to be used in time-sensitive critical applications. Although the health sensor monitoring applications considered in this work do not present stringent requirements regarding maximum end-to-end delay, there are other critical applications for human safety which are very sensitive to delay, for example, telesurgery. The studies carried out in [9,35] show that the maximum allowed end-to-end delay for telesurgery applications is about 600–700 ms. Therefore, in most scenarios, the smaller than 100 ms delay introduced by Ladon for the authentication and authorization of the remote surgeon prior to the communication establishment between both endpoints can be considered negligible.

9. Conclusions

IP-enabled heart-sensing devices seem to be the next natural step in the evolution of remote monitoring of patients with pacemakers and ICDs implanted. Thanks to this evolution, physicians will be able to query the information logged by heart monitoring sensors at any time and from any device connected to the Internet. However, such a scenario cannot be conceived without effective mechanisms to guarantee that medical infor-

mation is only provided to legitimate, authorized requests. In this context, the Ladon protocol presents a suitable approach to achieve this aim. Even when the design of Ladon is based on the Kerberos architecture, Ladon implies revolutionary features that constitute the basis over which privacy-aware applications for sensors can be implemented. To the best of our knowledge, this is the first approach to provide end-to-end authentication and authorization functionalities at the application level between a sensor and an Internet-connected device.

Apart from implementing authentication and access control features, Ladon involves the establishment of a pair-wise end-to-end secret key between a legitimate, authorized entity in the Internet and an IPv6-based sensor. This key can then be used to implement further security mechanisms, such as IPSec, a security mechanism already considered within the 6LoWPAN Working Group for other applications or scenarios.

This paper shows how the performance of Ladon in terms of the time and energy consumption in the protected sensors, and the delay for the information retrieval requests sent by client applications is affordable for the targeted scenarios. More specifically, the obtained results show that the average energy consumed by the execution of the Ladon protocol in a sensor is comparable to that of other protocols which implement even less advanced features. In addition, it has also been proved how the power consumption caused by Ladon is negligible in comparison with the power consumed by a common medical sensor when applying its therapy. Finally, it is also worth noting that due to the scant delay added by the Ladon interactions to each information retrieval operation, this protocol is also suitable to protect critical delay-sensitive applications.

Therefore, the measured performance parameters prove that it is suitable to implement Ladon in the proposed scenarios to efficiently protect remote monitoring operations of critical health sensors, such as pacemakers and ICDs. This results in an improved quality of life of patients implanted with such devices and in significant economic benefits for the affected patients and for the whole welfare system.

Acknowledgements

The work described in this paper was produced within the Training and Research Unit UFI11/16 supported by the UPV/EHU. This work was also partially funded by the Department of Industry, Innovation, Tourism and Trade of Basque Government through the Future Internet II strategic research project.

References

[1] F. Amin and A.H. Jahangir, Time and energy cost analysis of Kerberos security protocol in wireless sensor networks, in: *7th International Conference on Information Assurance and Security (IAS)*, 2011, pp. 308–313.

[2] J. Astorga, E. Jacob, M. Huarte and M. Higuero, Ladon: End-to-end authorisation support for resource-deprived environments, *IET Information Security Journal* **6**(2) (2012), 93–101.

[3] AVISPA: Automated Validation of Internet Security Protocols and Applications, FET Open Project IST-2001-39252, http://www.avispa-project.org.

[4] M. Bellare, A. Desai, E. Jokipii and P. Rogaway, A concrete security treatment of symmetric encryption: Analysis of the DES modes of operation, in: *Proc. 38th Annual Symposium on Foundations of Computer Science (FOCS 97)*, 1997.

[5] M. Bellare, J. Kilian and P. Rogaway, The security of the cipher block chaining message authentication code, *Journal of Computer and System Sciences* **61**(3) (2000), 362–399.

[6] Bluetooth Health Device Profile (BT HDP) v1.0 rev00 Bluetooth Special interest Group (SIG), http:// www.bluetooth.com (2011).

[7] P.D. Bradley, An ultra low power, high performance medical implant communication system (MICS) transceiver for implantable devices, in: *Proc. IEEE Biomedical Circuits and Systems Conference*, 2006, pp. 158–161.

[8] J. Carusso, The internet of things now includes a human heart, *Network World* (2009).

[9] B. Challacombe, L. Kavoussi, A. Patriciu, D. Stoianovici and P. Dasgupta, Technology insight: Telementoring and telesurgery in urology, *Nature Clinical Practice Urology* **3**(11) (2006), 611–617.

[10] H. Chan and A. Perrig, PIKE: Peer intermediaries for key establishment in sensor networks, in: *Proc. 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2005)*, 2005, pp. 524–535.

[11] R.A. Clark, S.C. Inglis, F.A. McAlister, J.G. F. Cleland and S. Stewart, Telemonitoring or structured telephone support programmes for patients with chronic heart failure: Systematic review and meta-analysis, *BMJ: British Medical Journal* **334** (2007), 942–950.

[12] H. Ector and P. Vardas, Current use of pacemakers, implantable cardioverter defibrillators, and resynchronization devices: Data from the registry of the European Heart Rhythm Association, *European Heart Journal Supplements* **9** (2007), 144–149.

[13] C. Elsner et al., A prospective multicenter comparison trial of home monitoring against regular follow-up in MADIT II patients: Additional visits and cost impact, *Computers in Cardiology* **33** (2006), 241–244.

[14] A.E. Epstein et al., ACC/AHA/HRS 2008 guidelines for device-based therapy of cardiac rhythm abnormalities, *J. Am. Coll. Cardiol.* **51**(21) (2008), e1–62.

[15] L. Fauchier et al., Potential cost savings by telemedicine-assisted long-term care of implantable cardioverter defibrillator recipients, *Pacing and Clinical Electrophysiology* **28**, 255–259.

[16] D.F. Ferriarolo, R. Sandhu, S. Gavrila, D. Kuhn and R. Chandramouli, Proposed NIST standard for role-based access control, *ACM Transactions on Information and Systems Security* **4**(3) (2001), 224–274.

[17] J. Großschädl, A. Szekely and S. Tillich, The energy cost of cryptographic key establishment in wireless sensor networks, *Proc. 2nd ACM Symposium on Information, Computer and Communications Security (ASIACCS '07)*, 2007, pp. 380–382.

[18] IEEE 802.15.4 Standard: Wireless Medium Access (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal AreaNetworks (WPANs) (2006).

[19] IPv6 over Low power WPAN (6lowpan) Working Group, http://datatracker.ietf.org/wg/6lowpan/charter/.

[20] ISO/IEEE11073 – Personal Health Devices standard (X73PHD), Health Informatics. [P11073-00103. Technical report-Overview] [P11073-104zz. Device specializations] [P11073-20601. Application profile – Optimized exchange protocol], http://standards.ieee.org (2011).

[21] G. Jolly, M.C. Kusçu, P. Kokate and M. Younis, A Low-energy key management protocol for wireless sensor networks, in: *Proc. Eighth IEEE International Symposium on Computers and Communications (ISCC '03)*, 2003, pp. 335–340.

[22] P. Kaijser, T. Parker and D. Pinkas, SESAME: The solution to security for open distributed systems, *Computer Communications* **17**(7) (1994), 501–518.

[23] C. Karlof, N. Sastry and D. Wagner, TinySec: A link layer security architecture for wireless sensor networks, in: *Proc. 2nd Int. Conf. Embedded Networked Sensor Systems (SenSys2004)*, 2004, pp. 162–175.

[24] A.K. Lenstra, Key lengths, in: *Handbook of Information Security*, H. Bidgoli, ed., John Wiley & Sons, 2005, pp. 617–635.

[25] Libtom Project, available at: http://libtom.org/.

[26] L.E. Lighfoot, J. Ren and T. Li, An energy efficient link-layer security protocol for wireless sensor networks, in: *Proc. IEEE Int. Conf. Electro/Information Technology*, 2007, pp. 233–238.

[27] M. Luk, G. Mezzour, A. Perrig and V. Gligor, MiniSec: A secure sensor network communication architecture, in: *Proc. 6th Int. Conf. Information Processing in Sensor Networks (IPSN'07)*, 2007, pp. 479–488.

[28] K. Malasri and L. Wang, Securing wireless implantable devices for healthcare: Ideas and challenges, *IEEE Communications Magazine* **47**(7) (2009), 74–80.

[29] G. de Meulenaer, F. Gosset, F.-X. Standaert and O. Pereira, On the energy cost of communication and cryptography in wireless sensor networks, in: *Proc. IEEE International Conference on Wireless and Mobile Computing, Networking and Communication (WIMOB'08)*, 2008, pp. 580–585.

[30] C. Neuman, Proxy-based authorization and accounting for distributed systems, in: *Proc. 13th Int. Conf. Distributed Computing Systems*, 1993, pp. 283–291.

[31] C. Neuman, S. Hartman and K. Raeburn, The Kerberos network authentication service (v5), *RFC 4120* (2005).

[32] T. Park and K.G. Shin, LiSP: A lightweight security protocol for wireless sensor networks, *ACM Transactions on Embedded Computing Systems* **3**(3) (2004), 634–660.

[33] A. Perrig, R. Szewczyk, J.D. Tygar, V. Wen and D. E. Culler, SPINS: Security protocolos for sensor networks, *ACM Wireless Networks* **8**(5) (2002), 521–534.

[34] Personal Health Devices Working Group (PHDWG), IEEE Standards, http://standards.ieee.org/PHDworkgroup/ (2011).

[35] R. Rayman et al., Effects of latency on telesurgery: An experimental study, in: *Proc. 8th International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI'05)*, 2005, pp. 57–64.

[36] K. Ren, W. Lou and Y. Zhang, LEDS: Providing location-aware end-to-end data security in wireless sensor networks, *IEEE Transactions on Mobile Computing* **7**(5) (2008), 585–598.

[37] B. Schneier, *Applied Cryptography*, 2nd edn, John Wiley & Sons, 1996.

[38] R.A. Shaikh, S. Lee, M.A.U. Khan and Y.J. Song, LSec: Lightweight security protocol for distributed wireless sensor network, in: *Proc. 11th IFIP Int. Conf. Personal Wireless Communications (PWC'06)*, 2006, pp. 367–377.

[39] R. Venugopalan, P. Ganesan, P. Peddabachagari, A. Dean, F. Mueller and M. Sichitiu, Encryption overhead in embedded systems and sensor network nodes: Modeling and analysis, in: *2003 International Conference on Compilers, Architectures and Synthesis for Embedded Systems*, 2003, pp. 188–197.

[40] M. Walla, Kerberos explained, *Windows 2000 Advantage magazine* (2000), http://technet.microsoft.com/en-us/library/bb742516.aspx.

[41] L. Wang, D. Wijesekera and S. Jajodia, A logic-based framework for attribute based access control, in: *Proc. ACM Workshop on Formal Methods in Security Engineering (FMSE'04)*, 2004, pp. 45–55.

[42] R. Watro, D. Kong, S. Cuti, C. Gardiner, C. Lynn and P. Kruus, TinyPK: Securing sensor networks with public key technology, in: *Proc. 2nd ACM Workshop Security of Ad Hoc & Sensor Networks*, 2004, pp. 59–64.

[43] G.H. Wettstein and J. Grosen, IDfusion, an open-architecture for Kerberos based authorization, in: *Proc. AFS and Kerberos Best Practices Workshop*, 2006.

[44] S. Zhu, S. Setia and S. Jajodia, LEAP: Efficient security mechanisms for large-scale distributed sensor networks, in: *Proc. 10th ACM Conf. Computer and Communications Security (CCS'03)*, 2003, pp. 62–72.