UML System-Level Analysis and Design of Secure Communication Schemes for Embedded Systems

G. Piscopo[1], M. Prevostini[2], I. Stefanini[3]
Advanced Learning and Research Institute (ALaRI)
University of Lugano
Via G. Buffi 13, CH-6900 Lugano, Switzerland

Abstract

In this work we develop a secure communication protocol in the context of a Remote Meter Reading (RMR) System. We first analyze existing standards in secure communication (e.g. IPsec, SSL/TSL) and existing implementations aimed at embedded systems with low-power constraints in general (e.g. lwIP, lwBT, ZigBee). Then, starting from a Platform Independent Modeling (PIM), we develop a protocol concept to address authentication, integrity and confidentiality, also covering battery lifetime checking and theft monitoring. Finally the protocol itself is described by means of UML. Limited resource and low-power constraints are taken into account when examining secure-transmission features. RMR is thus an example of an application requiring a light-weight protocol combined with security features. One of the future objectives is to switch from the PIM description to PSM implementation.

1 Introduction

As the complexity of systems increases, so does the importance of good specification and modeling techniques. Many factors contribute to the success of a project, and certainly one we cannot do without is a rigorous modeling language standard (see, e.g., [Har87, NVG92]). Introduced in recent years, the Unified Modeling Language (UML) is now widely used, basically for requirements specification and for the design of complex software systems and also embedded systems. A few months ago, a new community was created, called UML-SoC [SOC]. In the present paper we describe the use of UML for specification and modeling of the security and communication aspects of an embedded system, designed to work in a low-power mode. The case study chosen is a Remote Meter Reader (RMR) able to read energy consumption as displayed on a meter (e.g. electricity, gas, or water meter). Through UML we describe security issues such as data integrity, data confidentiality, authentication and theft monitoring. In the paper we show the security details of a research project of Advanced Learning and Research Institute. In Section 2 we outline the state of the art of low-power communication protocols and related security issues. Section 3 briefly describes the RMR case study that we have chosen, while in Section 4 we analyze advantages and disadvantages of a few communication protocol scenarios that could be applied in our case study. Section 4 explains the security flaws of the solutions presented in our case study and how we would like to deal with them. Using UML, we model the security solutions in terms of PIM as depicted in Section 6. Section 7 specifies our conclusions and suggests future research work.

2 State of the art

Data integrity, authentication and confidentiality The concept of *security* in networking may have different meanings, all related to the privacy of information exchange. Among others, special

[1] piscopo@alari.ch
[2] mauro.prevostini@unisi.ch
[3] ivan.stefanini@bluewin.ch

attention is paid to authentication (assurance about the identity of sender and receiver), confidentiality (that data are not read by others during transmission) and integrity (that data are not altered during transmission). Oddly enough, these functionalities can be carried out in many different ways depending on which layer of the communication stack they work at. Consequently, quite a lot of solutions have been designed in order to create mechanisms for secure communication.

IPsec may be better described as a suite of protocols which together give rise to a security framework at the network or packet processing layer. It provides a choice between two security services: an AH (Authentication Header) [KA98a], which essentially allows authentication of the sender of data, and an ESP (Encapsulating Security Payload) [KA98b], which supports both authentication of the sender and encryption of data. In this way it offers three of the most required services in secure networking: authentication, data integrity and data confidentiality. For key management the IKE (Internet Key Exchange) [HC98] protocol is available, even if separate key protocols can be selected. IKE is part of IPv6 [DH98] standard but it can also work with IPv4. Being at the network layer, it can operate in a very generic fashion over all IP traffic and so it shows more flexibility in configuring security when compared to other protocols. In fact it can protect data exchange between two hosts (end-to-end), over a set of links (route-to-route) or between two trusted networks (edge-to-edge). Its adoption is gaining ground, its use is expected to become increasingly widespread in the near future. IPsec answers many security questions and offers many advantages compared to other network security mechanisms (e.g. it is transparent to application layer). However, key management, configuration tools and performance still require careful assessment. So far its main application has been VPNs (Virtual Private Networks).

LwIP (LightWeight Internet Protocol) is an open source implementation of the TCP/IP protocol [Dun01]. It provides almost all functionalities of the standard TCP/IP protocol but has a smaller code and uses less memory than standard implementation. Also, it can work either with an underlying OS or without it. These features make it eminently suitable for use in embedded systems and generally in resource constrained systems, e.g. with tens of kilobytes for RAM data and for ROM code. Still, it features: IP (Internet Protocol), ICMP (Internet Control Message Protocol), UDP (User Datagram Protocol), TCP (Transmission Control Protocol) with its most common functionalities and optionally the Berkeley Socket API. It is worth bearing in mind that lwIP is available for free in C source code format.

LwBT (LightWeight Bluetooth) is an open source implementation of the upper layers of Bluetooth stack [Ohu03]. It is devoted to systems with resource constraints but it is tightly coupled to lwIP: as a matter of fact it has been created to transport IP packets from lwIP over Bluetooth and it shares the same coding technique. LwBT features: L2CAP (Logical Link Control and Adaptation Protocol), SDP (Service Discovery Protocol), RFCOMM (Serial port emulation protocol), PPP (Point to Point Protocol), HCI (Host Controller Interface), LAP (LAN Access Point) and DUN (Dial-Up Networking) profiles. Just like lwIP, lwBT can work either with or without an underlying OS.

SSL/TLS SSL (Secure Socket Layer) is an open encryption protocol, which was initially developed by Netscape but has never become a standard. TLS (Transport Layer Security) has been derived as SSL next version but became an IETF standard [IETF]. It works at the session layer, between the application and transport layers, and is able to create a secure, encrypted channel between two applications, a client and a server. Authentication, integrity and confidentiality functionalities are provided for this data flow, without even knowing the types of data that the upper application layer is exchanging. Several different encryption methods are provided. As in other higher-level security protocols, before a session can be established there is a negotiation phase for key management and exchange: usually a public-and-private key encryption scheme is applied. TLS is a fairly general protocol, which works seamlessly with different applications at the upper layers, but relies solely on TCP for lower layers.

ZigBee is a new emerging protocol [Kin03] that provides an open standard for low-power wireless networking, especially targeted to devices exchanging data at low rates, such as sensors and controllers. It achieves this by relying on the IEEE 802.15.4 standard, which defines the low-level layers (PHY and MAC) of a new wireless PAN (Personal Area Network) standard protocol. On the

other hand, ZigBee is responsible for the upper layers of the stack – from network to application layer – and relative services, such as routing and security. The latter is managed through authentication and key management services, and at lower levels by AES encryption and authentication given by the IEEE standard. This protocol is naturally implemented on 8-bit microcontrollers, requiring tens of kilobytes for the whole stack and some extra RAM for coordinator network nodes. It communicates over the license-free ISM bands, which provide unrestricted geographical use (USA, Europe and Asia). The first products shipped with ZigBee standard are expected in the first half of 2005.

Blowfish is a symmetric block cipher encryption algorithm [Sch93] designed as a fast and free alternative to Data Encryption Standard (DES) [DES77] and IDEA. It works with a variable-length key, up to 56 bytes, and it is best suited to operate in contexts where the key itself doesn't change so often during the same communication. As on top of that it is quite light, ideal for use in embedded systems and in performance-constrained environments in general. Blowfish is license-free and available for all uses. It is slowly gaining acceptance as a strong encryption algorithm. Being a symmetric encryption algorithm, it denotes a lightweight approach compared to RSA [RSA78] algorithm, which is computationally more expensive, yet ensures a higher level of security and functionality thanks to the asymmetric key scheme. As a proof for its flexibility, Blowfish can be included as encryption tool in different protocols and layers (e.g. TLS at session layer), but it can also be directly implemented in hardware.

3 RMR case study presentation

RMR is a reading system to help utility providers to obtain data on consumption of gas, electricity and water via wireless technology. The objective of this research project is to design a device able to perform real-time determination of energy consumption (where, when, what energy and by whom it is consumed), using wireless technology in a low-power and low-cost environment. As described in Figure 1, the Meter Reader (MR) is the device we are analyzing, whereas traditionally meters are devices used to measure energy consumption. The MR is the low-power device that aims to solve this problem by reading and storing at regular intervals (e.g. every 15 minutes) data measured by meter and sending them (e.g. once a day) to a Data Collector (DC) through a connection (e.g. Bluetooth). The DC collects data and stores them in a local memory, then the DC sends these data (e.g. once a month) to the Service Provider (SP). The utility provider should also be able to read at any time data stored in DC and, if necessary, determine the real-time consumption stored in MR. 1For more information about the RMR project please refer to [MMM+03].

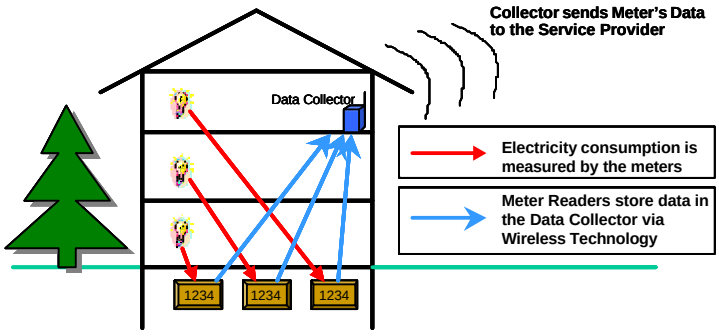

Figure 1: Problem description (model is valid also for water and gas meters).

4 Communication protocols scenarios

The purpose of this section is to analyze advantages and disadvantages of a few communication protocols that could be applied to our Case Study briefly presented in Section 3. Table 1 describes the analysis we have carried out for IPSec, LwIP, LwBT, SSL/TSL and ZigBee.

Scenarios	↑ Advantages ↑	↓ Disadvantages ↓
IPSec	• General security features available at IP level. • Can authenticate packets coming from an IP sender sharing a commonly known key. • It works and can be changed transparently with respect to application.	• Complexity, as a side effect deriving from flexibility (many options, several ways of doing similar tasks [FS99]). • Simplification required. • Cannot authenticate a sender at higher layers (e.g. application layer). • Data confidentiality and integrity are ensured during transmission, not storage.
LwIP + LwBT	• Power-saving combination when dealing with Bluetooth-enabled devices. • LwIP by itself optimal for embedded systems or similar devices. • Already existing implementations for several platforms and boards. • API fully available. • Light framework to work with.	• A security protocol working on it must be implemented (e.g. simplified IPsec). • Required effort to implement security over set of IP functionalities provided by LwIP. • Data confidentiality and integrity are ensured during transmission, not storage.
SSL/TSL	• Commonly used protocol for managing secure transmission over the Internet.	• Authentication on a per-session basis: every new session needs a new authentication handshaking[4]. • Data confidentiality and integrity are ensured during transmission, not storage.
ZigBee	• Already provides security at MAC and Network layer. • Specially designed for low data-rates and small data packets. • Specially designed for systems with low-power constraints and long battery life requirements.	• Keys and policy setup is left to upper layers of the stack. • Strongly related to specific Physical, MAC and Data-link layers (IEEE 802.15.4).

Table 1: Advantages and disadvantages of communication scenarios applied to our Case Study

ZigBee stack seems to be the fittest for application fields like sensors, controlling, monitoring and in general for low data-rates applications, but it has two disadvantages with respect to this case study. First, it supports security only at lower levels of OSI stack, and this could also not be a real problem in the final implementation if there are no *hops* between MR and DC. Second, it is strongly bounded to a specific communication channel, thus it does not really represent a good starting point for PIM.

LwIP constitutes a valid alternative, thanks to its growing diffusion and proven portability over different platforms. All the security features have to be implemented as new functionalities between application and LwIP itself, but this would also happen in the case of ZigBee if we are not interested in having security only at lower stack layers.

5 RMR case study: security issues

In order to secure messages and content exchanged and stored by the modules composing the RMR System, security has to be applied. Secure functionalities have to be used to provide:

- Authentication (paragraph 5.1)

[4] Usually having a new authentication handshaking for every new session is an advantage. In this analysis it is considered a disadvantage because network participants are always the same and handshaking impacts on power consumption.

- Confidentiality (paragraph 5.2)
- Data integrity (paragraph 5.3)
- Key management (paragraph 5.4)

The issue of Theft monitoring (paragraph 5.4) must also be addressed.

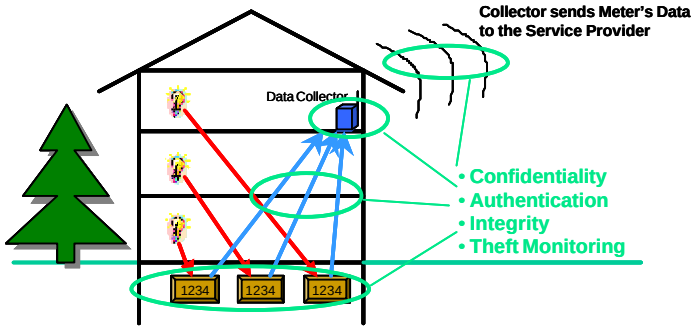

Figure 2: Ellipses describe where security features have to be applied.

5.1 Authentication

Authentication must be implemented into the system, in order to guarantee that every system module sending and receiving information is really what it claims to be. Data Collector has to be sure that all received data have really been generated and transmitted by an authorized Meter Reader. Service Provider has to be sure that all received data have really been generated and transmitted by an authorized Data Collector. Digital signatures techniques provide proof of authenticity of data and entities. To this purpose, we have chosen an authenticated Diffie-Hellman key agreement protocol, making use of a pre-shared secret key installed into the Meter Reader at construction time and also known to the Data Collector. This is to authenticate parties at the beginning of communication. Then, in order to authenticate each message exchanged, a pseudo-random function acting as a MAC (Message Authentication Code) has been chosen.

5.2 Confidentiality

All the sensitive information stored and/or transmitted by the system has to be protected against unintended or unauthorized access. Confidentiality is achieved using data encryption techniques, transforming the sensitive information into a ciphered text in such a way that an adversary who manages to have access to the ciphered text cannot understand this sensitive information. The entity having the rights to access the sensitive information possesses a secret decryption key that allows them to reverse the encryption transformation and retrieve data. Symmetric encryption schemes (DES, Blowfish) or Public/Private key schemes (RSA) can be used to achieve data confidentiality. The system must assure confidentiality on the information transmitted through the following links:

- Meter Reader → Data Collector
- Data Collector → Service Provider

Moreover, all the sensitive data stored in Meter Reader and in Data Collector could also be encrypted, in order to ensure confidentiality during storage. This could be especially true for Data Collector (typically a PC) with respect to Meter Reader (typically a custom board).

The encryption algorithm we choose is Blowfish, with a short-term session key 56 bytes long: even if it's quite short, it is feasible for the very short communication involved in this case study.

5.3 Data integrity

Data integrity guarantees that transmitted or stored data has not been accidentally or maliciously altered. In circumstances where integrity of data is important, hash functions and integrity controls (MACs) should be considered. The RMR System has to guarantee the integrity of all the sensitive

data exchanged. Thanks to the use of pseudo-random function as a MAC in order to authenticate messages, also integrity is ensured at the same time.

5.4 Key management

The main objective of key management is the secure administration and management of cryptographic keys and related information. Key management includes the generation, storage, derivation and destruction of keys. Any compromise or loss of cryptographic keys could compromise the authenticity, confidentiality or integrity of information. The key exchange protocol we choose is the *Aggressive Mode* from the standard IKE Version 1, Phase 1.

5.5 Theft monitoring

The purpose of theft monitoring mechanism is to alert the service provider about unauthorised manipulation of the remote system. The role of this functionality is to monitor the battery and sensors status as described in Figure 5. As soon as someone tries to vandalize a meter or a data collector, the system notifies the problem to the service provider through an alarm message. As a first approach to theft monitoring, the RMR System is designed in such a way that an error is reported to Data Collector if it's not possible to clearly read from the meter.

5.6 Security features overview

Table 2 summarizes the RMR System security issues described in the previous paragraphs (5.2–5.4). Here we show in which sensitive system points the security features should be provided. Regarding security during storage, this can be an issue or not depending on the actual realization of the system with its own physical parts.

	Authentication	Confidentiality	Integrity	Theft Monitoring	Key Management
Storage – Meter Reader		×		×	×
Storage – Data Collector		×		×	×
Link – Meter Reader → Data Collector	×	×	×		×
Link – Data Collector → Service Provider	×	×	×		×

Table 2: RMR System security features overview

6 UML Modeling

The RMR System modeled in this section will show the following functionalities to the actors wanting to interact with it:

- Regular transmission of meter data (Push Mode)
- Retrieval of stored meter data at any time (Pull Mode)
- Temporary storage of meter data locally to Meter Reader and Data Collector
- Theft monitoring activities to avoid unauthorised manipulation on batteries and sensors.

Authenticity, confidentiality and integrity of data must be provided along these functionalities. At this stage we specify a PIM of RMR where HW and SW components have not yet been defined.

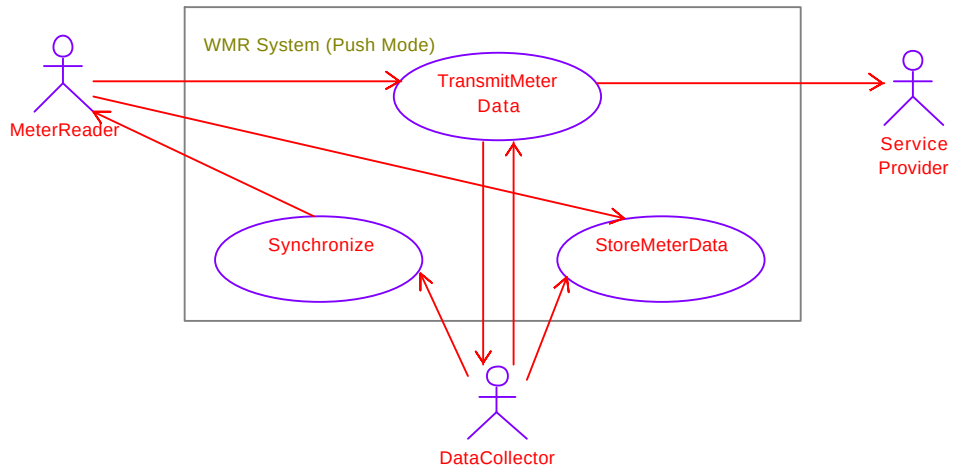

Figure 3: Use Case diagram of Push mode

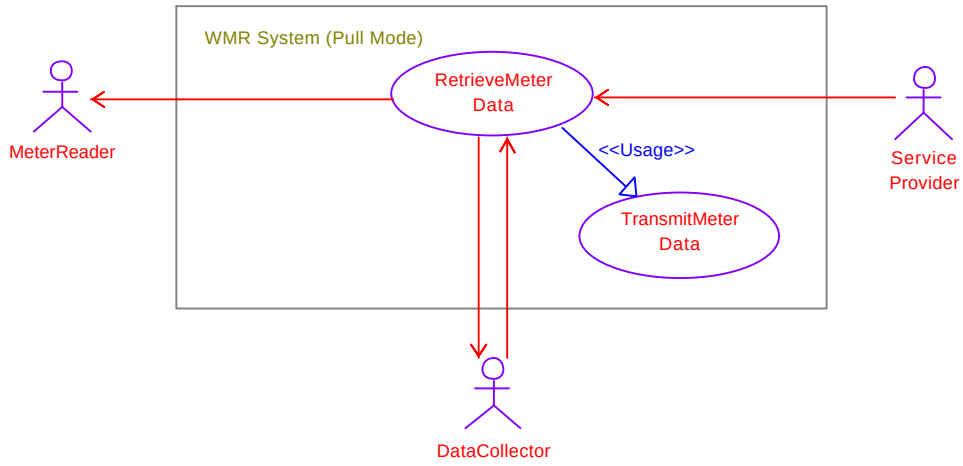

Figure 4: Use Case diagram of Pull mode

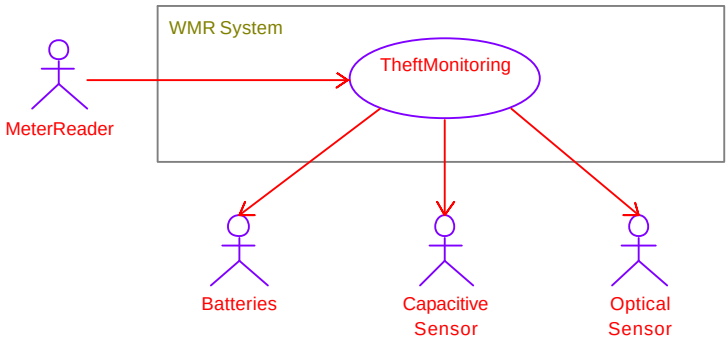

Figure 5: Use Case diagram of Theft monitoring activities

In the following sub-paragraphs, all the actors, the main use cases and the domain objects will be identified, analyzed and described using UML. The sequence diagram for the specific case of security in Push mode Use Case will also be analyzed and described.

6.1 Actors

Actor	Description
Meter Reader	The low-power device dedicated to energy consumption measurement. It reads and stores, at regular intervals, data measured by meter and sends them to a DC through a connection.
Data Collector	DC collects data and stores them in the local memory. Then it sends data to the SP. On request by SP, DC can ask the MR for a specific measure.
Service Provider	The utility (gas, water, electricity…) provider.
Batteries	MR batteries. Battery voltage is tested in order to send warning information to the DC before battery runs out. DC will in turn send the warning to SP.
Capacitive and Optical Sensors	Capacitive and Optical Sensors are able to read meter data and must be checked in order to avoid theft manipulations.

Table 3: RMR System actors

6.2 Use Cases

The main RMR System Use Cases are analyzed in this paragraph.

Transmit Meter Data Use Case describes the flow of data in the following scenarios:

- Data read by Meter Reader are transmitted to Data Collector
- Data collected by Data Collector are transmitted to Service Provider

Table 4 describes only the first scenario. However, if we change the active actor from Meter Reader to Data Collector and the passive actor from Data Collector to Service Provider, the same table will then describe the second scenario.

Parameter	Description
Name:	**Transmit Meter Data**
Active Actor:	Meter Reader
Passive Actors:	Data Collector
Brief Description:	The MR reads and stores, on a regular basis, data measured by meters and sends them to a DC through a connection. The DC receives the sent data and stores them in a local storage device.
Pre-condition:	Meter data have been read and stored by the MR and are ready to be transmitted to the DC.
Post-condition:	Data received by the DC have been stored
Normal Flow of Events:	1. MR reads meter data (and stores them) 2. MR sends data to DC through a (secure) channel 3. DC receives data

Table 4: Use Case, Transmit Meter Data

Store Meter Data Use Case describes the flow of data in the following scenarios:

- Data read by Meter Reader are encrypted (if needed) and locally stored
- Data collected by Data Collector are encrypted (if needed) and locally stored

Table 5 describes only the second scenario. However, if we change the active actor from Data Collector to Meter Reader, the same table will then describe the first scenario.

Parameter	Description
Name:	**Store Meter Data**
Active Actor:	Data Collector
Brief Description:	Upon reception of data, DC stores them in locally.
Pre-condition:	Data sent by the MR to the DC have been correctly received and interpreted, and are ready for storage.
Post-condition:	Meter data are into local storage.
Normal Flow of Events:	1. (If needed) DC applies confidentiality to data. 2. DC stores the data into locally.

Table 5: Use Case, Store Meter Data

Parameter	Description
Name:	**Synchronize**
Active Actor:	Data Collector
Passive Actors:	Meter Reader
Brief Description:	In order to have a common time-base between DC and MR for relating meter data, internal clock synchronization is needed. Common time could also be helpful in managing several MR by a single DC.
Pre-condition:	DC wants to ensure synchronization with MR.
Post-condition:	MR and DC are synchronized.
Normal Flow of Events:	1. At the end of each regular data transmission, DC sends a “Time Synch” signal with its own actual time to MR. 2. MR updates its own time with the one coming from DC.

Table 6: Use Case, Synchronize

Parameter	Description
Name:	**Retrieve Meter Data**
Active Actor:	Service Provider
Passive Actors:	Data Collector
Brief Description:	SP at any time must be able to read recent data still not received, either stored in DC or in MR.
Pre-condition:	SP wants to retrieve recent data still not received by DC.
Post-condition:	Requested data have been retrieved and can be consumed by SP.
Normal Flow of Events:	1. SP sends a “Retrieve Recent Data” request to DC, asking for data at some specific time. 2. If data is not already in DC, it sends in turn a “Retrieve Recent Data” request to MR, asking for data at some specific time. 3. MR sends (see “Transmit Meter Data, Push Mode” Use Case) requested meter data to DC. 4. DC sends (see “Transmit Meter Data, Push Mode” Use Case) to SP the data requested.

Table 7: Use Case, Retrieve Meter Data

6.3 Domain Object Model Diagram

Class SystemMgr represents the overall controller in Meter Reader, and it thus operate over the other classes, e.g. requiring to read, exchange or store some data. In order to do this, it uses Meter, CommunMgr and StorageMgr classes respectively.

Class CommunMgr implements the communication stack functionality allowing data to be sent and received on the channel, without exposing any internal detail about security or protocol. In order to add security to transmitted data, CommunMgr uses AuthentMgr class to setup communication and SecurityMgr class to exchange message securely.

Class SecurityMgr is used by CommunMgr and AuthentMgr classes in order to provide security functionalities. It implements all operations needed to add confidentiality, authenticity and integrity to data.

Class StorageMgr is used by SystemMgr class and provides storage functionalities. It implements the operations needed to store and retrieve data to/from a local storage device.

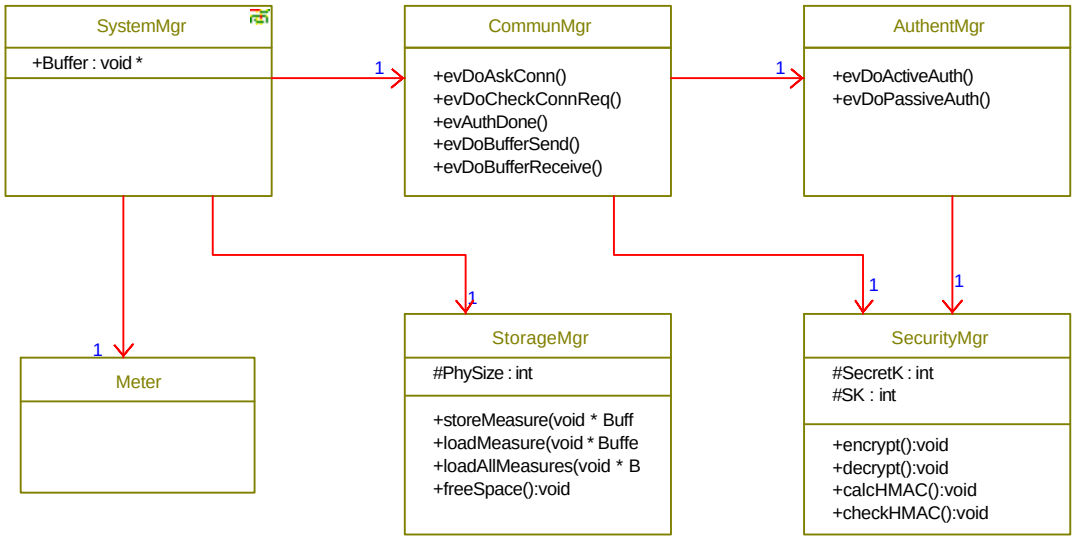

Figure 6: Domain Object Model diagram

6.4 Sequence diagram – Security in Push mode Use Case

This sequence diagram can be divided into two distinct sub-diagrams. One sub-diagram describes the flow during transmission phase (Meter Reader side) and the other describes the flow during reception phase (Data Collector side).

- SystemMgr asks CommunMgr for connection in order to do the regular data transmission
- CommunMgr connects and asks AuthentMgr for initial authentication
- AuthentMgr performs all needed steps for Diffie-Hellman exchange, also asking security services to SecurityMgr
- When authentication is confirmed, SystemMgr asks CommunMgr to send data
- CommunMgr gives data to SecurityMgr in order to secure them, that is to add confidentiality, authenticity and integrity
- SecurityMgr secures data
- CommunMgr sends data (to Data Collector)

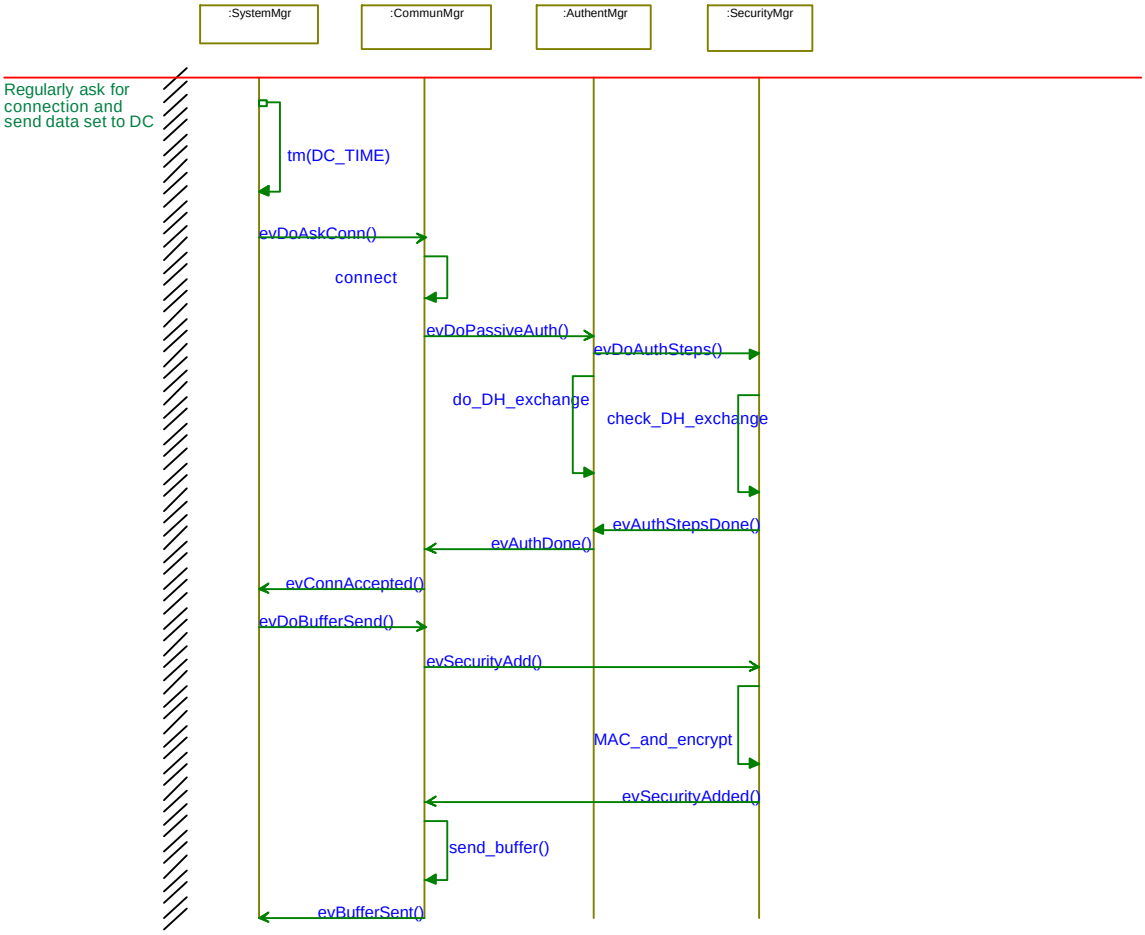

Figure 7: Sequence Diagram, Security at application layer

7 Conclusions and future research work

All sensitive information transmitted through the RMR System have to be secured. Authenticity, confidentiality and integrity features have to be applied to the system, in order to protect the system against security threats. There are several secure protocols based on several security schemes, which would satisfy the security requirements listed in this paper. Through UML we described the system-level design of the secure communication schemes of an embedded system. In particular, we used a PIM to analyze communication security aspects related to a RMR System. The choice of the most adequate protocol/scheme depends on the Platform Specific Modeling (PSM) required by the solution. In fact, the appropriate system and secure communication depends on:

- the choice of the platform on which the solution would be implemented,
- the wireless protocol through which data would be transmitted (e.g. Bluetooth, ZigBee)
- the specific service provider requirements.

ZigBee seemed very well tailored to this case study, but it is not ideal for PIM and is still not available on the market. For the moment, LwIP appeared as a good platform to work with: by the time we will move to PSM, ZigBee will be mature and eventually well established. Thanks to this PIM, it can still be taken into account without too many system modifications.

In our future research we will explore a more detailed PIM. As the PIM will not change, we will concentrate our efforts on the analysis of the PSM required for the implementation of specific solutions.

8 References

[Har87] D. Harel, B. Nando, "Statecharts: A Visual Formalism for Complex Systems", *Science of Computer Programming*, vol. 8, pp. 231–274, 1987.

[NVG92] S. Narayan, F. Vahid, D.D. Gajski, "System Specification with the SpecCharts language", *IEEE Design & Test of Computers*, pp. 6–12, December 1992.

[Dun01] A. Dunkels, "Design Implementation of the lwIP TCP/IP Stack", *Technical Report*, Swedish Institute of Computer Science, February 2001.

[Ohu03] C. Öhult, "lwBT – a lightweight Bluetooth Stack for lwIP", *Technical Report*, Luleå University of Technology, Computer Science and Electrical Engineering, 2003. [Online]. Available: www.sm.luth.se/~conny/lwbt.

[Kin03] P. Kinney, "ZigBee Technology: Wireless Control that Simply Works", *Communication Design Conference*, October 2003. [Online]. Available: www.zigbee.org.

[KA98a] S. Kent, R. Atkinson, "IP Authentication Header - RFC2402", *IETF RFC*, November 1998. [Online]. Available: www.faqs.org/rfcs/rfc2402.html.

[KA98b] S. Kent, R. Atkinson, "IP Encalpsulating Security Payload - RFC2406", *IETF RFC*, November 1998. [Online]. Available: www.faqs.org/rfcs/rfc2406.html.

[HC98] D. Harkins, D. Carrell, "The Internet Key Exchange - RFC2409", *IETF RFC*, November 1998. [Online]. Available: www.faqs.org/rfcs/rfc2409.html.

[DH98] S. Deering, R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification – RFC2460", *IETF RFC*, December 1998. [Online]. Available: www.faqs.org/rfcs/rfc2460.html.

[Sch93] B. Schneier, "Description of a New Variable-Length Key, 64-Bit Block Cipher (Blowfish)", *Fast Software Encryption, Cambridge Security Workshop Proceedings*, Cambridge, December 1993. [Online]. Available: www.faqs.org/rfcs/rfc2460.html.

[DES77] National Bureau of Standards, "Data Encryption Standard", U.S. Department of Commerce, *Federal Information Processing Standards*, Publication 46, January 1977.

[RSA78] R. Rivest, A. Shamir, L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems", *Communications of the ACM*, vol. 21 (2), pp. 120–126, February 1978.

[MMM+03] A. Minosi, A. Martinola, S. Mankan, F. Balzarini, A. Kostadinov, M. Prevostini, "Intelligent, Low-power and Low-cost Measurement System for Energy Consumption," *Proc. of VECIMS 2003*, pp. 125–130, July 2003.

[FS99] N. Ferguson, B. Schneier, "A Cryptographic Evaluation of IPsec", Unpublished manuscript, February 1999. [Online]. Available: www.schneier.com/paper-ipsec.html.

[SOC] UML-SoC: UML for SoC Design, www.c-lab.de/uml-soc

[IETF] IETF: Internet Engineering Task Force, www.ietf.org