International Journal of Smart Home
Vol.3, No.2, April, 2009

Enforcing Security in Smart Homes using Security Patterns

Paul El Khoury[1,3], Pierre Busnel[2], Sylvain Giroux[2], and Keqin Li[1]
[1]*SAP Research, 805 avenue du Dr. Maurice Donat, 06250 Mougins, France*
[2]*University of Sherbrooke, DOMUS lab, Sherbrooke J1K 2R1, QC, Canada*
[3]*University of Lyon I, LIRIS CNRS UMR 5205, 8, Bd Niels Bohr, 69622 Villeurbanne Cedex, France,*
[1]*{paul.el.khoury, Keqin.li}@sap.com,*
[2]*{pierre.busnel, sylvain.giroux}@usherbrooke.ca,*
[3]*paul.el-khoury@liris.cnrs.fr*

Abstract

Providing context-dependent security services is an important challenge in ambient intelligence. The complexity and the unbounded nature of such systems make it difficult for software developers to integrate security solutions. In order to solve this problem, in this paper we discuss and address multifold security challenges involved in the implementation of remote healthcare in smart homes using the security patterns approach. First the security challenges are derived from a real-world, industrially relevant scenario. Then it is shown how validated security techniques and mechanisms providing certain security properties can be captured and implemented in security patterns. Next security patterns are applied to satisfy security requirements in the smart home healthcare scenario. The process is exemplified thanks to a running prototype implementing the scenario.

Keywords: *Smart Home, Security Patterns, Remote Healthcare Assistance.*

1. Introduction

Today in western societies ageing and chronic diseases already incur great human, social, and economic costs. But the current demographic trends lead to expect an even greater burden on healthcare systems. The European Union (EU) health sector employs almost 10% of the total European workforce that is almost 9% of the EU Gross Domestic Product (GDP) [1]. According to the EU eHealth Taskforce 2007 report, the health costs are rising faster than the GDP and are estimated to reach 16% of GDP by 2020 in countries members of the Organization for Economic Co-operation and Development (OECD) [2, 3]. Clearly EU has to face a huge challenge for sustaining its current health and social care systems. Across the Atlantic Ocean, USA is facing a similar situation. According to U.S. Census Bureau estimations, the population aged 65 and over was 36 million persons in 2003 and is projected to increase to 72 million in 2030 [4]. In the USA, 85% of all hospital costs and 69% of all physician costs are spent on treating chronic diseases. In Europe, chronic diseases are estimated to amount to over 70% of healthcare costs [5]. Thus it is obvious to forecast a significant increase of health injuries related to normal and pathological ageing which will lead to loss of autonomy and greater fragility, then reducing the quality of life of elders. This is a true challenge for healthcare providers especially in times where the number of caregivers is tremendously decreasing.

When injured, sick or cognitively impaired, people are more fragile and need continuous supervision. If their environment is not well adapted, this will mean more often a transfer to a hospital setting. Telemedicine and homecare are the segments with the greatest potential for financial and clinical impact, and are expected to expand rapidly [6]. Thanks to ubiquitous and pervasive computing, Smart Homes (SH) can interact with a patient to foster his autonomy and to provide health monitoring [7].

In the traditional Healthcare model, patient's access to the healthcare delivery system has been limited to predetermined points of entry, such as through a primary care physician. In contrast, electronic healthcare, eHealth, proposes a fundamentally different unconstrained structure in delivering health information or care. Therefore in eHealth, more specifically in its applications on Telemedicine, new regulations are now in place to preserve the rights of the patients for privacy and confidentiality. Applying these regulations at the security level is crucial for the adoption of SH for remote healthcare. SH must be equipped with specific security setup adequate and adapted to Ambient Intelligence (AmI) requirements. In addition to traditional home security requirements, SH adoption requires to solve brand-new security vulnerabilities deriving from the automated facets of SH, for instance the sensor data exchanged within the SH over network contains sensitive information that could be stolen, modified, deleted if not well protected. However, application developers in SH environments are usually not security experts. Security patterns can help overcome this issue. A security pattern describes a particular recurring security issue that arises in specific contexts, and presents a well-proven generic solution for it. The SERENITY EU project on "System Engineering for Security and Dependability" [8] reshapes and extends the traditional informal representation of security patterns to bridge the gap between security experts and software developers. One of its essential proposals is to provide non security experts with the SERENITY Security & Dependability (S&D) patterns package. This package couples expert-validated security solutions with tested plug-and-play deployable implementations.

This paper illustrates how the SERENITY security pattern approach can be applied to SH. First, security issues are derived from a real-world, industrially relevant scenario related to healthcare in SH (§2). A SH is assisting a patient and monitoring its health status. When his doctor and his daughter are paying him a visit, the SH has to protect the confidentiality and privacy of his medical data, which are indeed typical security problems. The scenario is used to elicit security and dependability issues in AmI environments. Next an overview of security patterns explains how validated security techniques and mechanisms ensuring given security properties can be captured and implemented through security patterns (§3). Then security patterns, e.g. an authorization pattern, are applied to satisfy security requirements in the SH healthcare scenario (§4). The architecture and the security process are exemplified thanks to a running prototype implementing the scenario. Finally a short conclusion is drawn and future works are sketched.

2. Case study: healthcare services in a SH

It is indeed difficult for the average programmer to cope with challenges raised by security issues. Hence what about securing SH in the context of remote healthcare where rigid and new regulations [9] are setup? Therefore security patterns become an invaluable help as shown in [10, 11].

This section aims to illustrate security requirements in healthcare service in SH thanks to a simple yet complete scenario. First it presents the general context of the scenario: who the actors are, what are the information devices available in the SH, etc. (§2.1). Then the scenario per se is presented (§2.2). The patient was receiving a medical visit of his doctor when his daughter enters his home. Finally the scenario is analyzed from a security and dependability perspective, focusing on confidentiality and privacy of medical data (§2.3).

2.1. Context: a patient, his doctor, his daughter and his SH.

Bob, the patient, is a 70-year-old widowed man. Six months earlier, he had a Cerebral Vascular Accident (CVA). He spent 4 months at the hospital after his accident. Since he is back home, he still suffers from various troubles[1] and his health status needs to be monitored on a daily basis. Before leaving hospital, he subscribed to Smart Home (SH) services to get assistance in his activities of daily living (ADL). These services also monitor his heart rate continuously. Bob's health status is thus electronically captured and stored in an Electronic Health Record (EHR). The *EHR* refers to an individual patient's medical record in digital format which is composed of various pieces of information about the patient such as medicines prescribed, notes left by physicians and data recorded by medical sensors.

The *Monitoring and Emergency Response Center* systems (MERC) are coordinating the activities of the medical team (doctors, social workers, occupational therapists…) involved in Bob's medical aid. The MERC holds the HER. It also receives and handles information and emergency requests related to patients.

Dr. *Andrew* is a physician working at the MERC. He is Bob's family doctor.

The *Smart Home* (SH) is a conventional apartment equipped with various types of sensors to monitor and assist the patient in his Activities of Daily Living (ADL) [7, 12, 13]. Sensitive rugs, electro-magnetic sensors, infra-red and flow meters set all over the apartment, are used to recognize activities performed by the patient and prompt him with advices when necessary. Bob interacts with the environment using *touch screens* available in most of the rooms. *Microphones*, *speakers* and *cameras* are available to facilitate communications between the patient, the medical staff and his family. *RFID tag readers* located at the Smart Home front door are used to authenticate the medical staff, doctors and family members and to provide them access inside. The SH actions are coordinated with other participating actors through the MERC.

Rachel is Bob's daughter. Since her father CVA, she often runs his errands and visits him twice a week. Contrary to other visitors, Rachel has a privileged user status and can enter the SH using her RFID tag and password. Her father Bob has approved this.

The *SH terminal* combines an interface to interact with the SH server and the MERC server. It displays a calendar accessible through the MERC for adding medical or maintenance visits. It also contains an ADL assistant [12] and a communication interface with the MERC for emergency request or assistance request for a doctor. A medical interface is also included for periodically uploading medical data from the patient's medical sensors to the MERC. Doctors also use this interface to access patients EHR when visiting patients at home.

2.2. Scenario

[1] Cognitive deficits resulting from CVA include perceptual disorders, speech problems, dementia, and problems with attention and memory.

This section presents the scenario that will be used in remaining sections to illustrate the specification, the use, and the implementation of security patterns. This scenario is divided in three scenes putting in evidence security issues. In the first scene, the SH and the MERC coordinates their works to inform Bob of the venue of his doctor. In the second scene, Dr Andrew is at home with Bob (§2.2.2). In the third scene, Rachel joins them raising confidentiality issues related to his father medical data (§2.2.3).

2.2.1. Scene 1: The MERC has scheduled weekly medical visits for Bob's check-up. Each week, medical visits are assigned to available doctors, and events detailing the arrival time of doctors and their identities are added to Bob calendar. At one point in time, the calendar prompts Bob and then Bob confirms that he accepts the medical visit. Bob is then aware that Dr. Andrew is assigned to visit him this week.

2.2.1. Scene 2: When Dr. Andrew arrives at the SH's door, the RFID tag carried on his badge gets scanned and analyzed by the RFID tag reader which automatically rings the bell of the apartment. At the same time, the outdoor webcam takes a picture of Dr. Andrew. Bob, notified by the door bell, sees on his screen both the picture taken outside and the one corresponding to Dr. Andrew's identification badge, and unlocks the door from his terminal.
As shown in Figure 1, once inside Dr. Andrew logs onto the SH's terminal (1) to access Bob's EHR. The interface gives him access to notes left previously for instance by the occupational therapist, health status as monitored daily by the SH sensors and previous medication prescriptions (2). Since sensitive medical information must be kept confidential, thus SH cameras and microphones are turned off when such medical information is displayed and discussed between a doctor and his patient.

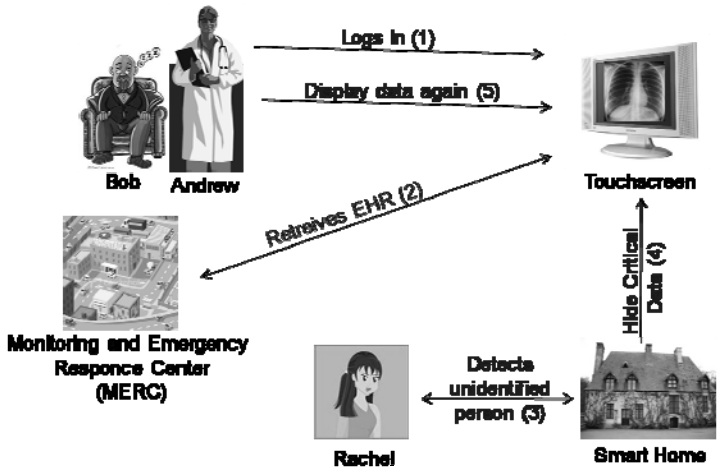

Figure 1. Home visit case study

2.2.2. Scene 3: While the medical home visit is going on, Rachel, Bob's daughter, comes on her way back to the SH from running her father's errands. The sensor network detects (3) her RFID tag, signals to Bob that she will enter, and allows her to enter. Yet Dr. Andrew is still examining Bob, and most data displayed on the smart home terminal are sensitive and strictly personal. Upon Rachel's presence in the hall, the SH terminal automatically hides (4) them on the screen. Dr. Andrew recognizes Bob's daughter and with Bob approval displays the medical data again (5).

In addition to the traditional security and dependability requirements, these small scenes highlight some AmI security requirements next section will address.

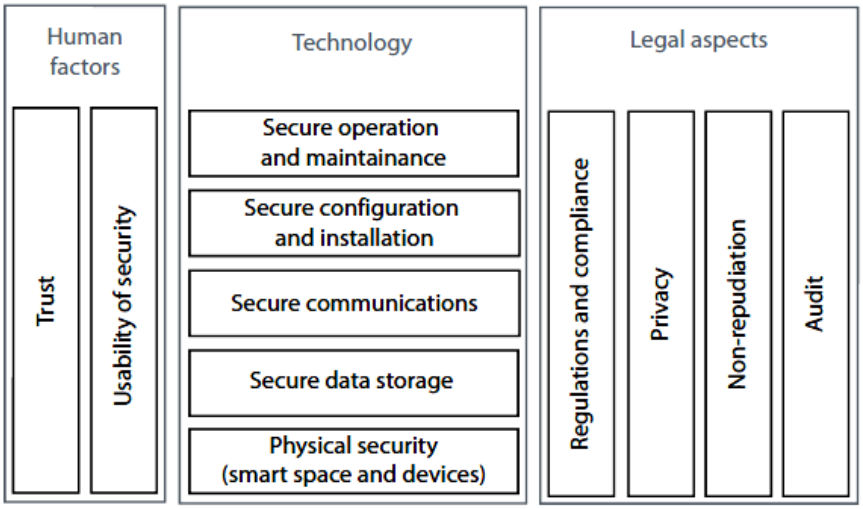

Figure 2. Security Requirements

2.3. Security requirements

Security engineering has to involve all parts of a system, from physical spaces, devices, and communication capabilities, to legal aspects, usability and social acceptance. A security study requires a detailed analysis for all the security requirements classified in Figure 2. Table 1 portrays an excerpt of high level security requirements related to the current case study which is typical of eHealth services. These requirements focus on non repudiation, service availability, integrity, confidentiality, privacy, and reliability for the identified actors and actions involved in the scenario.

Table 1. Sample S&D requirements for the home visit case study

	Reliability	*Confiden-tiality*	*Integrity*	*Non repudiation*	*Privacy*	*Availability*
Req 1- All MERC services	X					X
Req 2- EHRs provided by MERC	X	X	X			X
Req 3- Bob's EHR			X		X	X
Req 4- Dr. Andrew's visit				X		
Req 5- SH terminals	X					X
Req 6- SH services (Camera, bell microphones…)	X		X		X	X
Req 7- Smart Home communication with MERC	X	X	X	X		X

The SERENITY security pattern approach can fulfill most of these identified security requirements. The security patterns described in this paper represent an excerpt of the library populated in SERENITY. The security pattern library stands as a reference for the design and the deployment of systems having security requirements. Therefore the fundamental added value of the security pattern approach is to provide security solutions to non-security experts [14].

3. Primer on security patterns

The work of Yoder and Barcalow is the first known contribution to security patterns [15]. They propose to adapt object-oriented solutions to recurring problems of information security. Seven patterns were presented to be used when dealing with application security. As a natural evolution of Yoder and Barcalow work, Romanosky takes into consideration new questions that arise when securing a networked application [16]. Still following this path o security and networks, Schumacher et al. presented a set of security patterns for the development process [10]. Adopting a more general perspective, Fernandez and Pan have described patterns for the most common security models such as Authorization, Role-Based Access Control, and Multilevel Security [17]. Recently Fernandez et al. has highlighted the need to develop additional security patterns for database systems to integrate them into a secure software development methodology [18]. However all these security patterns had met little adoption in the security field. Their description in natural language limits indeed their applicability and forbids any reasoning mechanism.

The SERENITY EU project through a list of narrow yet complex studies [8, 11, 14] has tackled the security patterns objectives. In particular, SERENITY partners have built a model for secure and dependable applications [19]. Moreover, using security patterns they showed how SERENITY addresses, along with the tools provided, the challenge of developing, integrating and dynamically maintaining security mechanisms in open, dynamic, distributed and heterogeneous computing systems and particularly in Ambient Intelligence scenarios. One of the essential proposals from SERENITY is to provide novice users with the SERENITY Security & Dependability pattern package. This package comprises the expert-proofed security solutions and tested plug-and-play deployable implementations. SERENITY defines security patterns as detailed descriptions of abstract security solutions that contain all the information necessary for the selection, instantiation and adaptation of them. Such descriptions provide a precise foundation for the informed usage of the solution. An Integration Scheme (IS) is a special kind of security pattern defining the combination of security patterns. Complex security solutions relying on the usage and interactions of several patterns could be defined as integration schemes. In an information system, the relations among the participating security patterns are described in addition to other information related to the new integrated security solution. For instance, [20] shows how to integrate the XACML[2]-based security pattern to provide for socio-technical confidentiality in SH. The key of this security pattern is the access control policies ruling the access to the SH and protecting the privacy of Bob's her by enabling and disabling the display of sensitive information.

4. SERENITY Socio-Technical security pattern

[2] eXtensible Access Control Markup Language.

In *traditional* and *smart* homes, top priority for people is the feeling of living safely and securely. In general, a full control over their homes' entry points is what comforts them the most. In remote healthcare assistance Bob and the MERC, an external organization compliant to the authorities' regulations remotely assisting the patient, have the full control on the SH entry points. As mentioned in Section 2, Bob's explicit approval for opening the SH entrance door to Rachel, his daughter, overrides the SH control and grants her immediate access.

Requirement 3 from Table 1 specifies the right of Bob for privacy and asks for maintaining the confidentiality of his EHR data. While a traditional XACML implementation is able to provide partially this requirement, it was extended with a pattern that aims at enforcing this solution in particular for the SH.

4.1. Previous work: authorization using XACML for socio-technical confidentiality

The scenario is implemented in a fully operational prototype based on Service Oriented Architecture (SOA) and Web Services (WS) [20]. Within the constraints and limitation of this technical context, the XACML authorization mechanism was deployed addressing the provision of the EHR by the MERC, then satisfying Requirement 2 in Table 1 thanks to the Fine-Grained Authorization pattern.

In a nutshell, the XACML model comprises one or more Policy Enforcement Points (PEP), one or more Policy Decision Point (PDP), a Context Handler (CH), a Policy Information Point (PIP) and Policy Administration Point (PAP). The PEP is the XACML's front-end that receives a subject's request, initializes its evaluation process, and sends back the answer. The PDP selects the applicable policies and computes the authorization response by evaluating the requests with respect to these policies. In order to provide access control decision, as illustrated in Figure 3, the PEP intercepts access requests, passes them to the Context Handler (CH) that queries them in XACML language to the PDP. The PDP loads the applicable policy (or set of policies) based on the resource targeted by the request, and then asks for the credentials required for the policy evaluation. Once all applicable policies are evaluated, the pre-selected policy combination algorithm decides of the overriding evaluation. XACML defines several combination algorithms such as Deny-override and Permit-override. These combination algorithms are applied when combining access control rules to form a policy or when combining a set of policies. The access evaluation is passed back to the PEP for enforcement. Obligations are part of XACML language. Obligations are enforced by the PEP after a Permit decision. In the current implementation, the PDP sends the authorization Permit back to the PEP with a list of obligations that the PEP has to fulfill as part of the authorization request. If the PEP is unable to fulfill an obligation, this does not affect the access control decision.

We depict in Figure 3 the final process after the integration of this pattern within the Serenity Framework. The SH and MERC concerned applications are represented by *application*, the PEP patterns discussed in [20] are deployed as *EHRProxy* and finally *PDP* represents the Authorization using XACML pattern is deployed with the XACML policies at the SERENITY Framework (at the MERC side). The *EHRdata* are the patient's medical data exposed by the EHR web service that need to be protected. The *EHRProxy, i.e. PEP pattern*, have the same interface as the web service exposing the real EHR data. It is the public interface for accessing to the EHR data and therefore it enforces all operations that need to be checked by the *PDP, i.e. Authorization using XACML pattern*. Even with this distributed architecture, the prototype is scalable and the response is in less than a second with the specification presented in Figure 4 and in Table 2. **Prototype Specification**.

This pattern is preserving the confidentiality of Bob's EHR but not enforcing fully the privacy requirement possible in scene 2. We need to make sure that when Rachel enters the SH while Dr. Andrew is examining Bob, all terminals showing the EHR will be blanked. In the next section we present the extension we provided to the Authorization using XACML to satisfy the Socio-Technical confidentiality requirement, Req. 3 in Table 1.

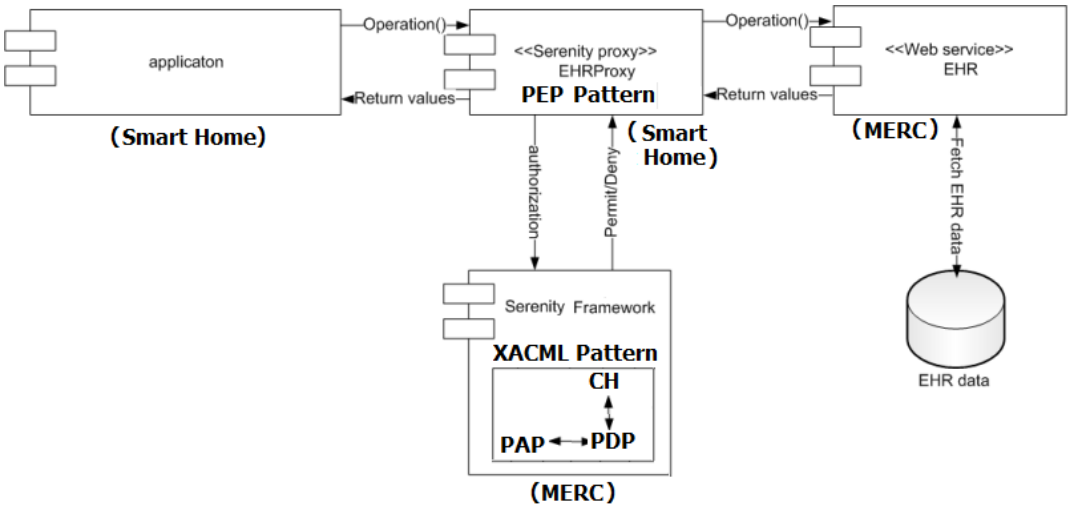

Figure 3. Fine-Grained Authorization Pattern Integration with SH applications

Table 2. Prototype Specification

Technologies	***Tools***
Web Server hosting Web Services and BPEL process	*Tomcat Apache as a web server installed and configured, to host (i) Web Services on top of Axis 2.0, and (ii) ActiveBPEL engine 4.1*
Database Management System	*HSQLDB as a portable relational database*
PC/Laptop with internet connection	*PC Pentium Centrino Duo with 2GB RAM with a Firefox 3.0 web browser to connect to the web server see Figure 4, C.*
Personal Digital Assistant (PDA) with WIFI connection	*HTC PDA (Diamond) with Windows Mobile 6.0 Operating System and Opera Web Browser.*
Smart Home or a Home enhanced with sensors and actuators	*Smart Kitchen, and sensitive rugs, see Figure 4, A*
A Bluetooth oxymeter that record the heart rate of the patient continuously and send the data to the MERC	*The oxymeter is connected to a computer at the patient's home, see Figure 4, B*

4.2. Extension of the socio-technical security solution

The security solution needs to satisfy the requirement of *scene 3* presented in Section 2.2. Dr. Andrew visits Bob at the SH, accesses his patient's EHR for examination. During the examination Rachel enters the SH. Since Bob didn't give his explicit consent to the SH to

authorize his daughter to see his EHR, the SH needs to *protect* the EHR as soon as she enters. So the information system has to perform the following steps:

- Monitor the house door to detect *someone*[3] has entered;
- Protect by hiding the EHR presented in the SH when the SH detects someone entering the house;
- Allow the doctor to show back the EHR through his credential once the EHR has been protected and the patient has provided his consent.

These specifications are generalized through a *NotificationConsumer* that needs to be notified when certain *Notification* is produced by a *NotificationProducer*.

This section presents this solution as a security pattern which contains *preconditions* (§4.2.1), a *description* (§4.2.2), and an *implementation* (§4.2.3).

Figure 4. Prototype Demonstration:
A) Smart Kitchen and sensitive rugs; B) oxymeter; C) client PCs

4.2.1. Pre-Conditions of the security pattern

The pre-conditions indicate assumptions and restrictions related to the deployment of the pattern. Before applying a pattern, users or applications in some cases should check the satisfiability of these pre-conditions. Obviously, pre-conditions are elements used during the selection of suitable patterns for a particular problem. For this pattern, there are the following assumptions:

- The *NotificationProducer* has an interface to provide the status of the *Notification*.
- The *NotificationConsumer* is able to ask the status of the *Notification* by accessing an interface of a web service.

4.2.2. Description of the security pattern

[3] By *someone* we designated authenticated and legitimate actors that have been already checked as shown in [20].

The solution is described in Figure 5, in which *NotificationProducer* and *Notification-Consumer* are existing entities in an application, and the others are introduced by the solution.

Three components are introduced: *ProducerBroker*, *Subscriber*, and *ConsumerBroker*. The *Subscriber* and the *ConsumerBroker* could be the same component in case of centralized architecture. The procedure is as follows:

1. The *Subscriber* subscribes for the *Notification* to the *ProducerBroker*, and asks to send the *Notification* to the *ConsumerBroker*;
2. The *ProducerBroker* keeps asking the *NotificationProducer* the status of the *Notification*;
3. The *NotificationConsumer* keeps asking the *ConsumerBroker* the status of the *Notification*;
4. When the *Notification* is available, the *NotificationProducer* sends the *Notification* to *ProducerBroker*
5. The *ProducerBroker* sends the *Notification* to *ConsumerBroker*;
6. The *NotificationConsumer* knows the *Notification* from *ConsumerBroker*.

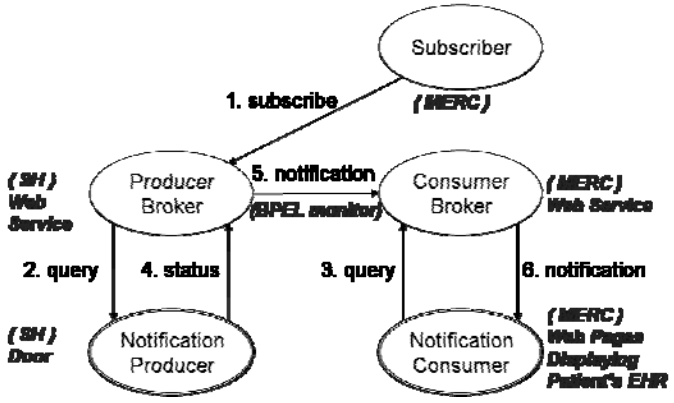

Figure 5. Socio-Technical Confidentiality

Considering the precondition validating that the *NotificationProducer* is able to be queried, and that the *NotificationConsumer* is able to query, it makes sense to connect them directly and make the *NotificationConsumer* keep asking the status of the *Notification*. However, as in distributed architecture, like ours, the *NotificationProducer* and *NotificationConsumer* are at different computers. Therefore we used the *ProducerBroker* to *push* the notification over the network (through *step 5*) and hence minimize traffic, then we configured the *NotificationConsumer* to *pull* from the *ConsumerBroker* periodically since they are on the same network and there is no risk of heavy network traffic being introduced.

We labeled with **bold** the concrete mapping of this pattern to our application in Figure 5. The MERC applications presenting the EHR to Dr. Andrew are web pages developed using Java Server Pages. These web pages are the *NotificationConsumer* that need to be partially hidden in case someone entered the SH. Hence *NotificationProducer* is the SH door. The pattern suggested to extend the application with *ProducerBroker* and *ConsumerBroker* that we developed them as two Web Services local to their network and interacting constantly (as described earlier) with the *NotificationProducer* and *NotificationConsumer*. Finally and in order to provide the monitoring system that pushes the notifications, we've developed a

process using BPEL to monitor the *ProducerBroker* which according to the status of the SH door decide to invoke the *ConsumerBroker* to change its status. The implementation of this monitor is depicted in Figure 6.

4.2.3. Implementation of the security pattern

WS-Notification is an emerging standard that allows creating a subscriber-publisher mechanism. The MERC would subscribe to a topic (representing the information he is interested in) and the door monitoring Web Service would publish updates on the door status. *WS-Notification* is subdivided in *WS-BaseNotification* and *WS-BrokeredNotification*. *WS-BaseNotification* can deal only with Web Services. Apache provides an implementation for *WS-BaseNotification* with the Apache Muse Project (http://ws.apache.org/muse/).

Since the concerned MERC's systems are not Web Service based we needed the brokered notification in order to allow the MERC to subscribe to the topic. Nevertheless the *WS-BrokeredNotification* is not implemented in the Muse Project. Therefore we used a similar approach to *WS-Notification*. We created a business process which gets a subscriber address and forwards the updates to this subscriber. Its actual specifications and development in ActiveBPEL are depicted in Figure 6. This business process resides in the SH network. The MERC plays the consumer role. This avoids unnecessary network traffic.

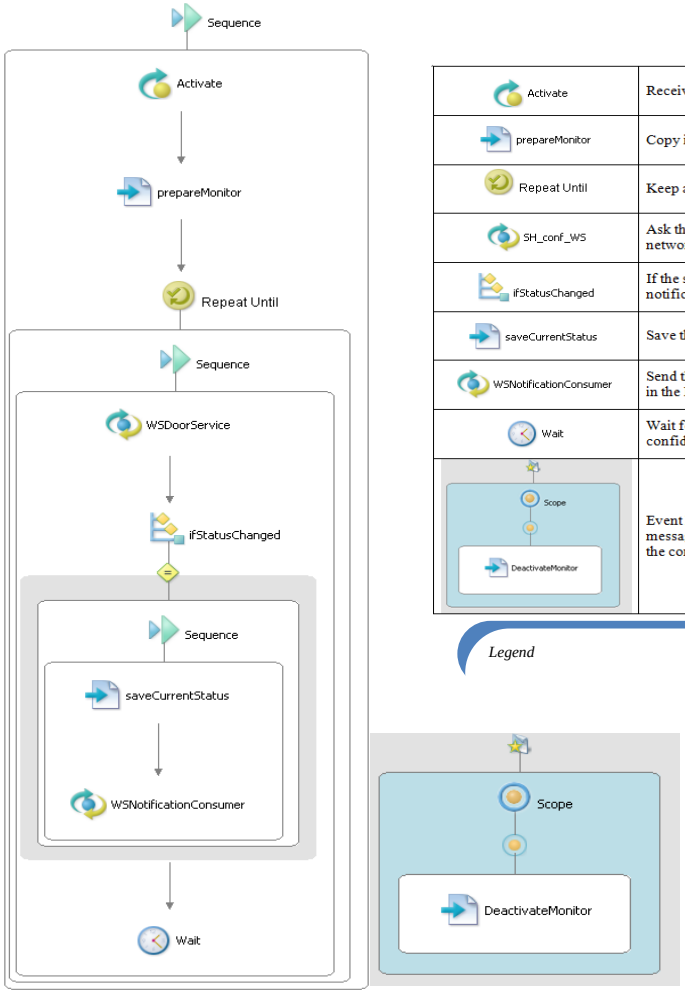

Figure 6. ActiveBPEL Model of the Socio-Technical Pattern

The MERC applications are not Web Service based, therefore we implemented a consumer Web Service, which resides in the MERC side (thus, the MERC pulls this Web Service to get the information). Since this Web Service is in the same machine as the MERC, network traffic is minimized. The resulting prototype was developed using the same configurations presented in Figure 4 and in Table 2. **Prototype Specification**. The sensitive information displayed on the screen was hidden in less than a second after the authentication of Rachel. In our SH settings there is around 3 seconds from the door at the entrance of the SH until the examination room which leaves enough time for the MERC to hide the sensitive data.

5. Conclusion and future work

This paper presented a remote healthcare assistance case study. We considered in detail the SH privacy requirements of the patient in a SH environment. We discussed extensively the privacy requirement prior to the authorization solution presented using the security pattern approach in [20]. We proposed security solutions satisfying the security requirements typically existing in such AmI environment. The presented prototype is fully implemented and operational (with additional scenes); the SH is hosted at the DOMUS laboratory in University of Sherbrooke and the MERC is hosted at SAP Labs France servers. The illustration and performance provided in this paper are the feedback of three days demonstration at the ICT 2008 Exhibit in Lyon, France. Future works will consider the combination of the presented security solutions with additional solutions operational at other layers, such as SSL at the network layer.

Acknowledgement

The authors would like to thank Vincenzo Manzoni and Michele Silva for their help in the prototype development. This work was partially funded by IST-FP6-IP-SERENITY.

References

[1] Accelerating the Development of the eHealth Market in Europe– eHealth Taskforce report 2007.

[2] Price, Waterhouse, Coopers study, HealthCast 2020: Creating a Sustainable Future, 2006.

[3] Health Information Network Europe (HINE) report 2006 – European eHealth forecast.

[4] He W. et al. 2005. “65+ in the U.S.: Current Population Reports” Washington, DC: U.S. Bureau of the Census, pp. 23-209.

[5] WHO report, 2006, Building foundations for eHealth.

[6] Gartner study: The potential of telemedicine applications, October 2006.

[7] H. Pigot, A. Mayers, and S. Giroux, “The intelligent habitat and everyday life activity support”, Proceedings of the 5th international conference on Simulations in Biomedicine, Slovenia, April 2003, pp. 507–516.

[8] A. Mana, C. Rodolph, G. Spanoudakis, V. Lotz, F. Massacci, M. Molideo, and J. S. Lopez-Cobo, Security Engineering for Ambient Intelligence: A Manifesto, IGI Publishing, 2007.

[9] Legally eHealth Putting eHealth in its European Legal Context– Legal and regulatory aspects of eHealth Study report March 2008.

[10] M. Schumacher, Security Engineering with Patterns: Origins, Theoretical Models, and New Applications, Lecture Notes in Computer Science, LNCS 2754, Springer Verlag, August 2003.

[11] L. Compagna, P. E. Khoury, F. Massacci, R. Thomas, and N. Zannone. “How to capture, model, and verify the knowledge of legal, security, and privacy experts: a pattern-based approach”, International Conference on Artificial Intelligence and Law, 2007, pp. 149–153.

[12] J. Bauchet, S. Giroux, H. Pigot, D. Lussier-Desrochers and Y. Lachapelle, "Pervasive Assistance in Smart Homes For People with Intellectual Disabilities : A Case Study on Meal Preparation," *IJARM*, vol. 9, pp. 53-65, December 2008.

[13] D. Vergnes, S. Giroux and D. Chamberland-Tremblay, "Interactive assistant for activities of daily living," in *From Smart Home to Smart Care*, July. 4-6, 2005, Sherbrooke, Canada.Proceedings of the 3rd International Conference on Smart Homes and Health Telematic (ICOST), Canada, 2005.

[14] A. Benameur, P. E. Khoury, M. Seguran, S. Kumar Sinha. "Serenity in e-Business and Smart Items Scenarios". Security and Dependability for Ambient Intelligence Series: Advances in Information Security , Vol. 55 Spanoudakis, George; Mana Gomez, Antonio; Spyros, Kokolakis (Eds.) 2009, Approx. 375 p. 20 illus., Hardcover ISBN: 978-0-387-88774-6 2009.

[15] J. Yoder and J. Barcalow. "Architectural Patterns for Enabling Application Security", Conference on Pattern Languages of Programs (PLoP), 1997.

[16] S. Romanosky, Ed., Security Design Patterns, 2001.

[17] E. Fernandez and R. Pan, "A Pattern Language for Security Models", Conference on Pattern Languages of Programs (PLoP), 2001.

[18] E. B. Fernandez, J. Jurjens, N. Yoshioka, and H. Washizaki, "Incorporating database systems into a secure software development methodology," 19th International Workshop on Database and Expert Systems Applications, pp. 310–314, 1-5 September 2008, Turin, Italy.

[19] G. Spanoudakis, A. Mana Gomez, and K. Spyros, Eds., Security and Dependability for Ambient Intelligence. Series: Advances in Information Security , Vol. 55, Springer, April 2009.

[20] P. Busnel, P.E. Khoury, S. Gyroux, K. Li, "Achieving Socio-Technical Confidentiality using Security Pattern in Smart Homes" The Third International Symposium on Smart Home, IEEE ed. 2008.

Authors

Paul El Khoury is currently a Research Associate in the Security & Trust program of SAP Research Center, France. He is doing his Ph.D. jointly between SAP Research and University Claude Bernard of Lyon on Security Engineering particularly on Security Patterns. He authored various international conference and journal publications in the area of Security Engineering in addition to several patents.

Pierre Busnel is currently doing his Ph.D. at the DOMUS laboratory of the University of Sherbrooke, QC, Canada.
He actively collaborated with SAP Research, France in the European project SERENITY and authored many international publications in conferences and journals related to smart homes, medical applications, and security.

Sylvain Giroux is a well established scientist currently working as professor at the Department of Computer Science at the University of Sherbrooke, QC, Canada.
He has co-founded the DOMUS laboratory of the University of Sherbrooke with the professor Hélène Pigot, in 2005.
He received a Ph.D. in computer science from the University of Montréal, QC, Canada, in 1993.
His main research interests are smart homes, mobile computing, pervasive computing, cognitive assistance, multi-agent systems, tangible user interfaces, and user modeling.

Keqin Li is a researcher in the Security & Trust Program of SAP Research, France. His current research interests include Security Engineering, Software Testing, Network Protocol Testing, etc. He obtained his PhD degree of Computer Science in 2000 in Peking University, Beijing, China.