An XACML-based Security Pattern to achieve Socio-Technical Confidentiality in Smart Homes

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

Abstract

In this paper we discuss and address multifold security challenges involved in the implementation of remote healthcare in smart homes. These security challenges are derived from real-world, industrially relevant scenarios. Validated security techniques and mechanisms providing certain security properties can be captured and implemented in security patterns, which can be applied in order to satisfy security requirements in the smart home healthcare scenarios. The presented results are parts of our ongoing research effort aiming at the development of an integrated security framework for remote healthcare and ambient intelligence systems.

1. Introduction

Occidental countries are facing great challenges from a population of elders increasing rapidly while the birth rate cannot sustain it. 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 [1]. Thus it is easy to forecast an increase of health injuries related to normal and pathological aging which will lead toward loss of autonomy and greater fragility, then reducing their quality of life. When injured, sick or cognitively impaired, aged and fragile, people will need continuous supervision. If resources are not adapted at home, this will mean more often a transfer to a hospital setting. Thanks to ubiquitous and pervasive computing, Smart Homes (SH) can interact with the resident to foster its autonomy and to provide for health monitoring [2].

Nonetheless security is a crucial if one want to build a SH for the real. They must be equipped it with privileged security setup adequate and adapted to Ambient Intelligent (AmI) security specific requirements. Indeed in addition to traditional home security requirements, SH adoption requires to solve brand-new security vulnerabilities deriving from the automated facets of SH. Unfortunately application developers in SH environments are usually not security experts. Security patterns can help overcome this and provide SH with the required security solutions. 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 project (“System Engineering for Security and Dependability”) address exactly this kind of situation [3]. One of its essential proposals is to provide novice users with the SERENITY Security & Dependability (S&D) patterns package. This package comprises of the expert-validated security solutions and tested plug-n-play deployable implementations.

In this paper, we illustrate how SERENITY can be applied to SH thanks to a simple case study (§**Ошибка! Источник ссылки не найден.**). First the case study is presented: a patient's health status is continuously monitored remotely through a SH. Of course, typical security problems such as confidentiality and privacy of the patient's medical data will arise. Several security requirements from such AmI environment are presented, and an AmI confidentiality requirement is fulfilled by applying security patterns. Section 0 presents an overview on security patterns. Next, we present the architecture of the authorization pattern and present an overview of the proposed security solution (§0). Finally in section 0 we conclude and present future work.

2. Case study

SH in a medical context raises difficult security issue, the average programmer is not able to cope with. Security pattern can then become an invaluable help. To illustrate security requirements and present our solution made by security patterns, we first introduce our case study (§0), and describe a couple of tightly related scenes (§**Ошибка! Источник ссылки не найден.**). Finally, we highlight security requirements of business applications closely related to confidentiality and privacy of the patient's medical data (§0).

2.1. Actors

The case study involves the following actors: the patient Bob, his daughter Rachel, the physician Andrew, and the *Monitoring and Emergency Response Center* systems (MERC).

Bob is a 70-year-old widowed man. Six months ago, he had a Cerebral Vascular Accident (CVA). Bob spent 4 months at the hospital after his accident. Since he still suffers from various troubles his health status needs to be monitored daily. Before leaving hospital, he subscribed to a Smart Home (SH) program to get assistance in his daily activities and to make his heart rate monitored continuously.

Bob's health status is electronically captured 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 from medical sensors. The EHR is used by the *Monitoring and Emergency Response Center* systems (MERC) to coordinate the medical team participating in Bob's medical aid.

The *MERC* receives and handles emergency requests arisen by patients. It also coordinates the activities of many other actors including doctors and social workers.

Dr. *Andrew* is a physician working at the MERC. In our case study, Andrew is Bob's personal doctor, who is in charge of Bob's case.

The *Smart Home* is coordinated to other participating actors through the MERC. The Smart Home is a conventional apartment equipped with various types of sensors to monitor and assist the patient in his Activities of Daily Living (ADL) [4, 5, 6]. 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. Patients interact with their environment using *touchable 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* are available at the Smart Home door for authenticating the access requesters among the medical staff, doctors, family and others during home visit.

Rachel is Bob’s daughter. Since her father CVA, she often runs his errands and visits him twice a week. Rachel in contrary to other visitors is a privileged user and can enter the SH using her RFID tag and password as approved by her father Bob.

The *SH’s terminal* combines an interface to interact with the SH’s server and the MERC’s server. It displays a calendar accessible through the MERC for adding medical or maintenance visits. It also contains an ADL assistant [5] and a communication interface with the MERC for emergency request or doctor assistance request. A medical interface is also included for periodically uploading medical data from the patient’s medical sensors to the MERC. Doctors may also use this interface to access patients EHR when visiting patients at home.

2.2. Scenario

In the general scenario, 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 patients’ calendars. Bob is then aware that Dr. Andrew will be the one assigned to visit him this week. Bob confirms his acknowledgement of the visit. Then the scenario is divided in the two scenes below to put in evidence some security issues. In scene 1, Andrew is alone with Bob. In scene2, Rachel joins them.

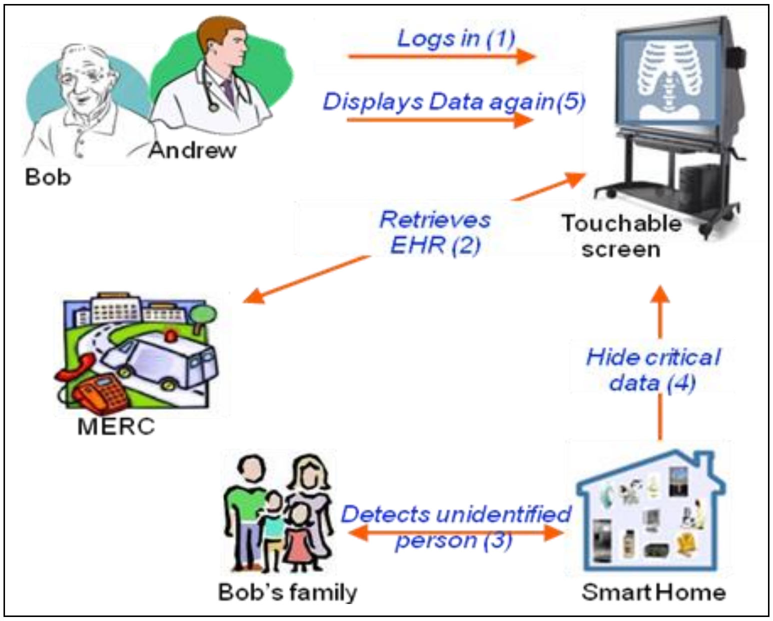


Figure 1 – Home visit case study

2.2.1. Scene 1: When Dr. Andrew arrives at the SH’s door, the door bell rings as for the RFID tag carried on his badge gets scanned and analyzed by the SH. 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, health status monitored daily and previous prescriptions (2). Such information must be kept confidential between doctors and patient, cameras and microphones are turned off inside the Smart Home when medical information is being displayed and discussed.

2.2.2. Scene 2: The home visit goes on, Bob's daughter comes on her way back from running errands to visit her father at the SH. The sensor network detects (3) her RFID tag, signals Bob her approach, and allows her to enter. Yet Andrew is still examining Bob, and most data displayed on the smart home terminal are delicate and strictly personal. Upon Rachel's presence in the entrance, and since no explicit approval was provided by Bob, the SH terminal automatically hides (4) the delicate data on the screen. Dr. Andrew recognizes Bob's daughter and with Bob's approval displays the medical data again (5).

In addition to the traditional security and dependability requirements, the two presented scenes highlight some AmI security requirements discussed in next section.

2.3. Security requirements

This case study is typical of e-Health services. Table 1 highlights a few of critical security requirements: non repudiation, service availability, access control, integrity, confidentiality, privacy, and reliability.

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

1.	The MERC shall be *available* and *reachable* 24 hours a day, 7 days a week.
2.	Each communication between the SH terminals and the MERC *shall guarantee messages delivery, integrity and confidentiality of the data exchanged, and mutual authentication*.
3.	The SH terminals shall always *be available* and *connected to the communication network*.
4.	The MERC shall continuously ensure the *reliable network connection* with the SH.
5.	Patient's data available at Smart Terminal shall be *kept confidential* and *accessible to authorized* requesters.
6.	Doctors shall *not repudiate scheduled home visit* after previous confirmation.
7.	The sensor network detection of unidentified persons approaching the examination area *shall be reliable*.
8.	To *ensure the patient's privacy*, cameras and microphones of the examination room shall be turned off during medical home visit.
9.	Data requesters have *to justify their need* to access patient's EHR, and *strictly least privileged access* will be granted, part of legal need-to-know principle [8].

The SERENITY security pattern approach can fulfill most of these identified security requirements. Indeed the security patterns described in this paper represent an excerpt of the library we are populating in the context of the SERENITY project. The security pattern library could serve as a reference in the design and deployment of systems having security requirements. Therefore it becomes clear that the fundamental plus value of the security pattern approach is providing security solutions to non-security experts [7].

3. An overview of security patterns

The pattern approach has been adopted into software engineering as a method for object-based reuse [9]. Following this particular path, Schumacher [7] applied the pattern approach to security problems by proposing a set of security patterns for the development process. Yoder and Barcalow [10] proposed architectural patterns that can be applied when introducing security into an application. Fernandez and Pan [11] described patterns for the most common security models such as Authorization, Role-Based Access Control, and Multilevel Security. One of the main problems of these proposals is the lack of tools that validate patterns with respect to expert knowledge. The usual natural language descriptions for security patterns open room for different interpretations of solutions provided and problems described by these patterns, as shown recently in [12].

SERENITY enables to capture security techniques and mechanisms into security artifacts. SERENITY's description for these artefacts enables the selection, adaptation, usage and monitoring at runtime by automated means of security techniques and mechanisms. There are three kinds of security artifacts, Security Classes, Security Patterns, and Security Implementations. Although this paper emphasizes the usage of security pattern artifact [13] and presents an intuitive and extensive description of all of them. 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 IS, the relations among the participating security patterns are described in addition to other information.

In [14] two possible ways of capturing the security mechanism, i.e. authorization using XACML, were introduced, one as a security pattern, and the other as an integration scheme. In the next section we exploit this work to answer the confidentiality requirements identified in our AmI case study.

4. Socio-Technical Security Solution

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 ensures them the most. In remote healthcare assistance Bob and the MERC, an external organization compliant to the authorities' regulations remotely assisting the patient, has the full control of the SH entry points. As mentioned in §2, Bob's explicit approval for opening the SH entrance door to Rachel, his daughter, overrides the SH control and grants her immediate entrance.

The presented prototype is fully operational through Service Oriented Architecture (SOA) using Web Services (WS). Two alternatives of deploying authorization mechanism are considered, i.e., using SAML (Security Assertion Markup Language) or using XACML (eXtensible Access Control Markup Language). Ensuring confidentiality of SH resources as referred in Table 1, particularly in *Req 5, 8 and 9,* requires evaluating requests on fine-grained resources such as location and time. With this aim, authorization using XACML is recommended over authorization using SAML for fine-grained access control [15].

Table 2 – Summary of XACML as S&D pattern

S&D Pattern	S&D Requirement	S&D Solution
SP1	Confidentiality for fine-grained resources in SOA using WS.	Authorization using XACML, e.g., Sun's implementation.
SP2	Single Enforcement Point for extensive logging capabilities to facilitate audits.	Policy Enforcement Point, e.g., proxy web server
IS1	Confidentiality for fine-grained and distributed resources in SOA using WS.	Combination of SP1 and SP2**.** Authorization using XACML, e.g., our proposal in this paper with distributed PEPs with single PDP.

4.1. Authorization using XACML as Security Patterns

Our previous work [14] on capturing the access control solution using XACML in security patterns is applicable to satisfy the presented confidentiality requirements. In that work we focused on the XACML authorization model more than the language, specifically on the Policy Enforcement Point (PEP) and Policy Decision Point (PDP). The results indicated that XACML authorization model can be captured in one security pattern, or a combination of security patterns (i.e., integration scheme). A summary of both results is illustrated in

Table 2. On the first hand, in local deployment of an XACML authorization solution, one S&D pattern is capable of capturing the validated conceptual model and providing a plug-n-play implementation. On the other hand a distributed deployment of the authorization using XACML as it is in our case requires one host for the evaluation engine (i.e., the MERC) and several hosts for the enforcement points (i.e., the MERC repository and the SH). This is captured by means of an integration scheme, where communications between the enforcement points and the evaluation points have to be secured.

A brief summary of the XACML model is depicted in Figure 2 – **XACML authorization model** and summarized hereafter. 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, 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 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. Actually, 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.

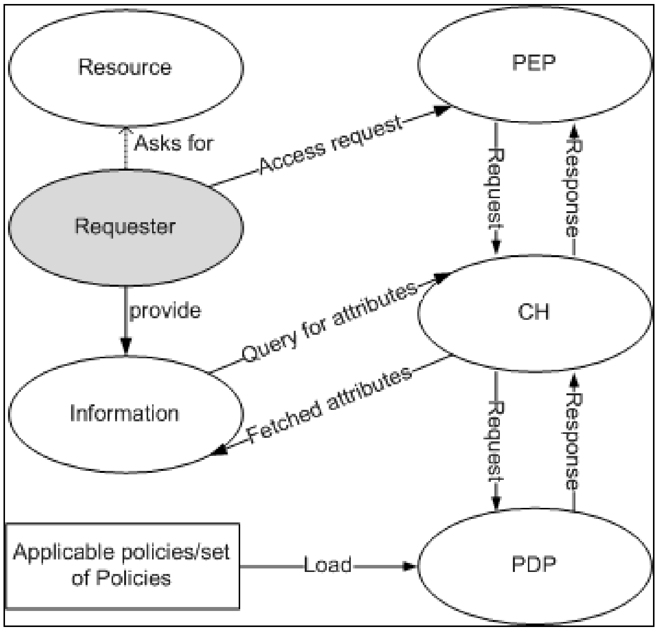


Figure 2 – XACML authorization model

4.2. Authorization using XACML as Integration Scheme for AmI confidentiality requirement

Clearly *Req 5, 8 and 9* of Table 1 matches better the *Confidentiality for fine-grained and distributed resources in SOA using WS* requirement of

Table 2. In order to show the deployment of *IS1*integration scheme and the two *SP2* security patterns we identified the environments of our case study that has to be mapped to the security patterns. An illustration is depicted in Figure 3, where *conceptually* we show a deployment of the *IS1* made by two *SP2* patterns, namely PEP-1 and PEP-2, and one *SP1* that contains the decision point. The medical data are periodically sent to the SH; hence PEP-2 enforces access control to these data too. The main difference between PEP-1 and PEP-2 resides in their physical distribution. In our case study, the PEP-1 is hosted on the MERC server in SAP Labs France, while the PEP-2 is hosted at the DOMUS Laboratory in University of Sherbrooke. Each of these implementations has to be connected to the requesters' terminals, the information retrieval service(s), the resources and the evaluator.

In *Scene 1* of section 2.2, Dr. Andrew gets into Bob's SH in order to assist him during weekly visits. Technically speaking, the deployment of *IS1* does not reveal the interesting challenges as for the steps in setting up the policies for satisfying *Req. 5 and 9* from *Scene 1*. Dr. Andrew, using his RFID tag, authenticates himself to the SH. The access request sent by the RFID reader is sent to PEP-2 of the SH.

The PEP-2 creates the following token <*Dr. Andrew RFID tag identifier, his password and onetime generated passcode, an open action*, on the *resource door*> and passes it to the CH as the XACML illustration in Figure 2. The CH creates an XACML request and sends it to the PDP of the IS1. After succeeding the strong authentication, the first applicable XACML policy authorizes Dr. Andrew entrance. In fact, this policy satisfies *Req 9, by* checking the validity of Dr. Andrew’s request with the Bob’s scheduled medical visit. The second applicable policy requires Bob's decision on opening the door. As part of the obligation for this policy, the door bell rings. Bob looks to his e-Health terminal, sees Dr Andrew’s request for access, compares the photos, and responds positively. As a result the access to the SH is granted to Dr. Andrew.

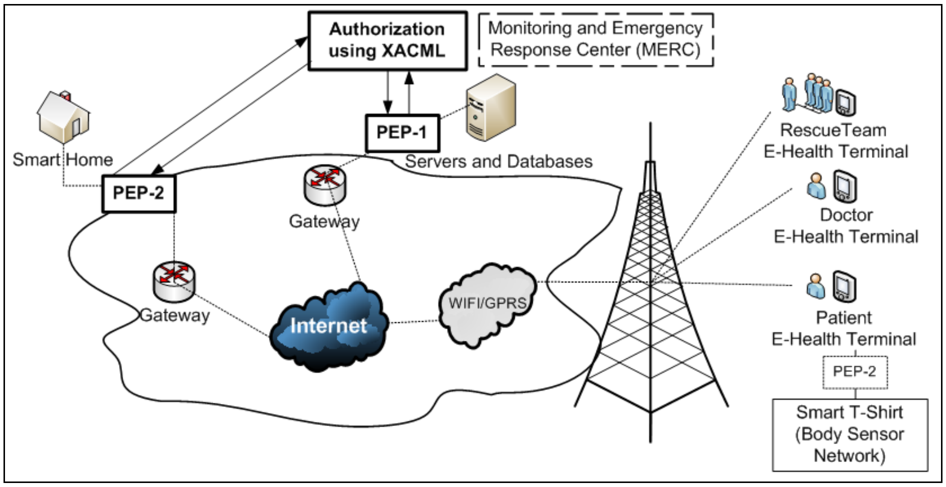

Authorization using XACML
Monitoring and Emergency Response Center (MERC)
Smart Home
PEP-1
Servers and Databases
PEP-2
Gateway
Gateway
Internet
WIFI/GPRS
RescueTeam E-Health Terminal
Doctor E-Health Terminal
Patient E-Health Terminal
PEP-2
Smart T-Shirt (Body Sensor Network)

Figure 3 – Authorization using XACML Integration Scheme in the case study

Once inside, Dr. Andrew accesses Bob's EHR through the SH terminals. In the first phase, Dr. Andrew uses his username, password and the onetime generated *passcode* to be authenticated to the e-Health application provided by the MERC. In the second phase, Dr. Andrew requests to view and modify Bob's EHR (ref. Table 1, *Req. 5*). The terminal client page creates a message request to the PEP-1 of the MERC. The PEP-1 creates the following request <*Dr Andrew's identifier* and the *SH terminal*, *view and modify* as actions, and *Bob's EHR* as resource> to the CH. The applicable policy for fetching Bob’s EHR checks whether the request was initiated from Bob’s SH terminal (even if it is created by Dr. Andrew) or through his Laptop or his e-Health terminal. If Dr. Andrew's request was initiated from a terminal different from the ones previously mentioned, then *Req. 9* of Table 1 wouldn’t have been satisfied. However, since Dr. Andrew’s request is triggered from the SH’s terminal, a Permit access is returned as it justifies his need to access Bob’s EHR.

In *Scene 2* of section 2.2, Rachel's visit requires yet its share of the access control policies. Similarly to Dr. Andrew's case, Rachel's visit shall be checked from the SH's door until her entrance. Once at the door, within the authentication step, Rachel as already having Bob’s consent for direct access does not require additional confirmation from her farther for entering. Nevertheless, the applicable policy takes into consideration Bob’s privacy (*Req.* 8). It has to ensure that Bob's EHR is not displayed on any screen even when family members are within the SH without Bob's approval. This is interpreted in our policy as an *obligation* added to the grant permission. The obligation locks down the visualization of all connected e-Health terminals. Dr. Andrew after having Bob’s acceptance has to re-authenticate himself to the SH terminal and unlock the screen.

5. Conclusion and future work

This paper presented a remote healthcare assistance case study. Most of the smart home security requirements are discussed extensively: non repudiation, service availability, access control, integrity, confidentiality, privacy, and reliability. Then an authorization solution is applied using the security pattern approach to satisfy 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 XACML implementation has been implemented and demonstrated 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.

6. References

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

[2] J. Jorge, “Adaptive tools for the elderly new devices to cope with age induced cognitive disabilities” in Proceedings of the 2001 WUAUC, 2001, pp. 66–70.

[3] 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.

[4] 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.

[5] J. Bauchet, D. Vergnes, S. Giroux, and H. Pigot, “A pervasive cognitive assistant for smart homes”, Proceedings of the International Conference on Aging, Disability and Independence (ICADI), USA, 2006, pp. 228.

[6] J. Bauchet and A. Mayers, “A modelisation of adls in its environment for cognitive assistance”, Proceedings of the 3rd International Conference on Smart Homes and Health Telematic (ICOST), Canada, 2005, pp. 221–228.

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

[8] 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.

[9] E. Gamma, R. Helm, R. Johnson, and J. Vlissides. Design patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley Professional, 1994.

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

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

[12] A. Armando, R. Carbone, L. Compagna, J. Cuellar, L. T. Abad, “Formal Analysis of a SAML Web Browser Single Sign-On Protocol”, to appear in the Formal Methods in Security Engineering, 2008.

[13] F. Sanchez-Cid and A. Mana. “Patterns for automated management of security and dependability solutions”, 1st International Workshop on Secure systems methodologies using patterns (SPattern), 2007.

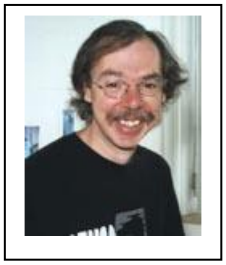
[14] F. Sanchez-Cid, A. Munoz, P. El Khoury, and L. Compagna, “XACML as a Security and Dependability (S&D) pattern for Access Control in AmI environments,” Ambient Intelligence Developments (AmI.d), Springer, September 2007.

[15] OASIS XACML specification, http://www.oasis-open.org.

Authors

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 various international conference publications in the area of Smart Home, medical applications and security.

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.



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 mobile computing, pervasive computing, distributed artificial intelligence, multi-agent systems, user modelling, intelligent tutoring systems.

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.