Adaptive Solutions for Access Control within Pervasive Healthcare Systems

Dana Al Kukhun, Florence Sèdes

Adaptive Solutions for Access Control within Pervasive Healthcare Systems

Dana Al Kukhun and Florence Sedes

IRIT, Paul Sabatier University
118 Route de Narbonne, 31062 Toulouse Cedex 9, France
{kukhun,sedes}@irit.fr

Abstract. In the age of mobile computing and distributed systems, healthcare systems are employing service-oriented computing to provide users with transparent accessibility to reach their distributed resources at anytime, anywhere and anyhow. Meanwhile, these systems tend to strengthen their security shields to ensure the limitation of access to authorized entities. In this paper, we examine mobile querying of distributed XML databases within a pervasive healthcare system. In such contexts, policies - as XACML - are needed to enforce access control. We study the reactivity of this policy in the case of a user demanding access to unauthorized data sources showing that the policy will respond negatively to user demands. Thus, we propose to employ an adaptive mechanism that would provide users with reactive and proactive solutions. Our proposal is accomplished by using the RBAC scheme, the user profile and some predefined semantics in order to provide users with alternative and relevant solutions without affecting the system's integrity.

Keywords: Access control, pervasive computing, adaptation, XACML.

1 Introduction

As new technologies direct toward user satisfaction and quality assurance, service-oriented computing is being more and more adopted by enterprises in order to ensure better functionality and easier service delivery to their clients. Pervasive computing has added a new dimension to quality assurance when it promised to users a transparent access to systems at anytime, anywhere and anyhow.

Providing an interoperable interaction within dynamic environments and distributed information sources is highly advantageous and that's why enterprises are turning their classical ***Enterprise Information Systems*** EIS into ***Pervasive Enterprise Information Systems*** PEIS as a fulfillment of their promises in improving the performance of their services and maintaining transparent interaction with users [2] this would increase the accessibility and usability of these dynamic environments.

In such open environments, systems tend to secure their resources against any attack and restrain access in order to provide information only to authorized users.

Security was usually disconnected from the business domain. Nowadays, with the deployment of ***Service-Oriented Architectures***, services are seamlessly interacting to

exchange information flows. Thus, a great need rises to secure these transactions through different layers using organizational policies and practices (e.g. fine grained access control) in order to govern the loosely coupled interactions that take place.

Consequently, the vision of interoperability had two perspectives: the first is the user's perspective who demands for maximal accessibility considering it as the objective of using a service, while the second is the system's perspective which should ensure secure access to business resources and clients information.

The principal objective of ubiquity and pervasiveness is to ensure a transparent access to information sources that are distributed in different physical locations. So as users tend to obtain access from different locations and using different personal gadgets, PEIS guarantee their availability, portability and security by following interoperable service-oriented architectures and by enforcing fine-grained access control security policies like XACML ***eXtensible Access Control Markup Language*** which is an XML-based policy that manages decentralized control on distributed resources and can thus, centralize decision making within PEIS [16].

As interoperability can be a double-faced coin between accessibility and security [2], we highlight the importance of applying an adaptive layer within PEIS in order to guarantee system integrity and user satisfaction. Thus, we argue that in some situations PEIS should be adaptive and consider providing users demanding access to unauthorized elements with suggestions to other authorized pieces of information that are relevant to their demands and are accessible according to their privileges profile.

In this paper, we have chosen a scenario within healthcare applications where a user is in an urgent situation and trying to gain access to a part of a document that is located in the distributed healthcare resources. Knowing that the documents are semi-structured (represented in XML) and that the access control policy used is XACML with the RBAC scheme (Role Based Access Control), the expected response of the conflict resolution mechanism that XACML adopts will be "access denied".

Therefore, we propose to extend the XACML policy by including a service based query rewriter that would execute a failure recovery mechanism that would either respond in a reactive manner by giving the user the option to correct his query by showing him the specific parts that he has access to or in a proactive manner where the system exploits the user profile and some predefined semantics to reform and adapt the issued query and provide users with alternative authorized access paths to access requested information.

Our proposition aims to provide a level of adaptive security that would meet the user needs by taking into consideration the user profile (interests, preferences, location, device, etc.), the context in which he's using the service and the privileges granted to him (e.g. RBAC). Finally, we intend to show that adaptation isn't only beneficial from a user's perspective or a system's perspective but can also take an intermediate position between the both of them.

In this paper, we'll start by presenting pervasive healthcare systems and the importance of balancing access in these systems. Then, we continue with a state of the art about access control in pervasive systems. Next, we'll present a security challenging scenario that shows the struggle between accessibility and access control within a pervasive healthcare system. Finally, we'll justify our proposition by highlighting the importance of adapting query results not only to user preferences and needs but also to his access rights.

2 Pervasive Healthcare Systems

The vision of healthcare systems as Service-Oriented Applications came from the idea of viewing hospitals as enterprises that interact with different internal and external systems in order to provide adequate services to different clients connected by different means.

As information systems are becoming pervasive systems and aiming to be more responsive and adaptive to user mobility, new medical and healthcare systems have promoted a collaborative usage of patient's medical information through a distributed network; where the patient record has become a ***virtual record*** (digital record) that can be treated by different users in different physical locations.

The evolution of a central ***Electronic Health Record*** that can be accessed from different systems and connection lines has helped in reducing the volume of patient records that are archived redundantly and has helped to acquire online, up-to-date patient information whenever needed. This has highly affected the quality of patient treatment and the time consumed to retrieve patient record.

Integrating ubiquity in healthcare systems is promoting the emergence of ***Pervasive Healthcare Systems***. In such systems, the user will be able to access the system from anyplace at anytime and using different machines and connection technologies. As healthcare systems tend to Integrate highly developed technologies (mobile devices, RFID technologies, eTokens, Smart cards, wireless networks and adaptive middleware) in order to satisfy user needs, these systems aim to become pervasive.

Employing such technologies will support different applications and services like telemedicine, patient monitoring, location-based medical services, emergency response and management, pervasive access to medical data, personalized monitoring using health-aware mobile devices and would finally ensure lifestyle incentive management [18]. Such services promote the quality of healthcare systems and would have great effect on reducing medical coasts on the long term.

As shown in fig 1, pervasive healthcare systems aim to perform a real time transparent and interoperable access to online information sources that might be decentralized in different physical places; this accessibility might be ***local and direct*** such as the case of a doctor consulting the hospital's central database in order to retrieve the patient's record. Another case might involve an indirect access to external resources as the case of a home nurse consulting the patient's record by performing ***indirect access*** to the system's database using a web service for example. This indirect access might be also performed by an employee accessing the insurance company system in order to see the patient's status.

A third case might be the one of a doctor requesting access to a data stream of patient information existing ***on an external database*** residing on mobile devices held by a home nurse or by the patient at home (in order to monitor the patient status and progression).

Healthcare data should be available and accessible at anytime, anywhere but their access should be restricted to authorized people. Pervasive healthcare applications will help health providers, such as major hospitals, to achieve the level of data quality required without spending high investments in information infrastructures.

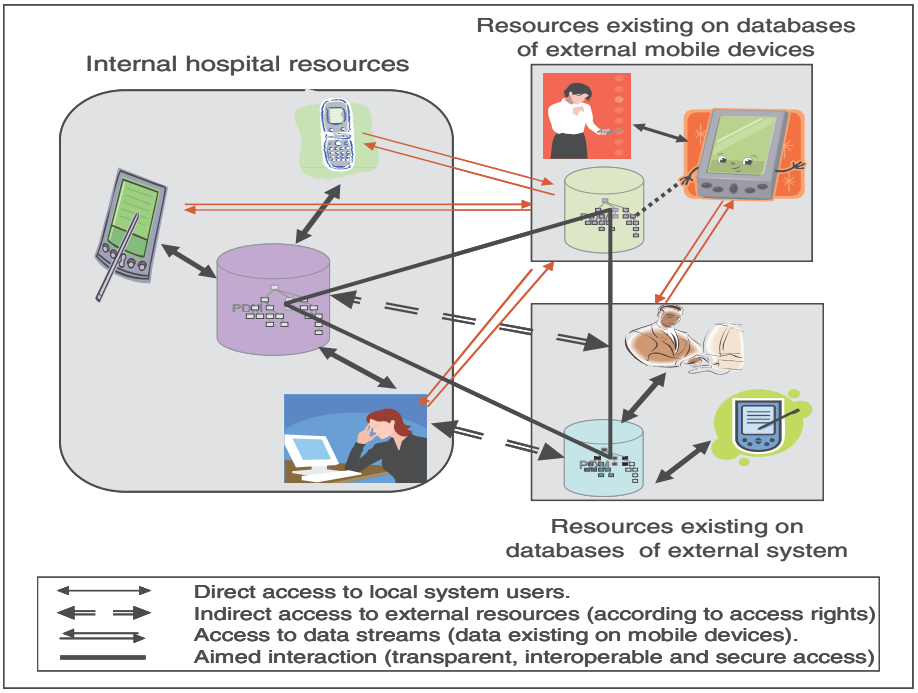

Fig. 1. Access modalities within Pervasive Healthcare Systems

In order to share patient information, medical centers tend to perform record transactions or to access medical records of external resources. Ensuring the patient privacy in such operations is very important and therefore any record transfer should be justified and any record access should only be allowed in particular purposes.

Finally, as we have illustrated the various interaction modalities that can take place, we highlight that the objective of a pervasive system is to provide transparent transactions within the system and to ensure seamless interaction between the user and the system in order to provide him with maximum accessibility to data sources. Next, we'll expose the different components of pervasive healthcare systems in order to find a way to secure access within these systems.

3 Access Control in Pervasive Environments

In pervasive environments, mobile users tend to request ubiquitous access to data from different terminals and under variable connection qualities. Being in the age of multimedia, data is heterogeneous in kind (comes in different forms and formats) and also in source (it could be located in different places within decentralized systems or coming from different sources).

In order to achieve transparent access and facilitate data distribution, the W3C introduced XML eXtensible Markeup Language as a standard for data representation and exchange [6].

Accessibility within pervasive environments is attained using unreliable connectivity channels. That's why access control is highly needed and is considered as an efficient way to restrict access to unauthorized users. In the context of controlling access to XML documents and databases, pervasive services should employ secure and efficient mechanisms to protect sensitive data against exposure.

Different authorization mechanisms were proposed to perform centralized access control to XML documents [4, 7, 9, 10, 13]. As we are moving towards mobile and decentralized access to data, [5] has proposed to move access control to the client side justifying that in the past, client devices weren't trustworthy so all client-based access control solutions relied on data encryption where the data are kept encrypted at the server and a client is granted access to subparts of them according to the decryption keys in its possession. Moreover, as centralized access controllers, these models had to minimize the trust required on the client's device by providing a static way of sharing data. The dynamic client-based evaluator of access control rules regulates access to XML documents and takes benefit from a dedicated index to quickly converge towards the authorized parts of a streaming document. The introduction of this method was justified by the emergence of hardware and software security.

These research works are efficient in the case of client-based access and when guaranteeing secure connection paths but with the emergence of pervasive computing, system are becoming more and more dependant on service-oriented computing and applications. That's why our works are interested in presenting a service oriented approach to ensure secure access to decentralized information sources containing semi-structured documents

In order to ensure secure transactions within PEIS, XACML ***eXtensible Access Control Markup Language*** was proposed by OASIS [14]. XACML provides an expressive security policy for data exchange within dynamic environments which enables a flexible way to express and enforce access control policies while exchanging data.

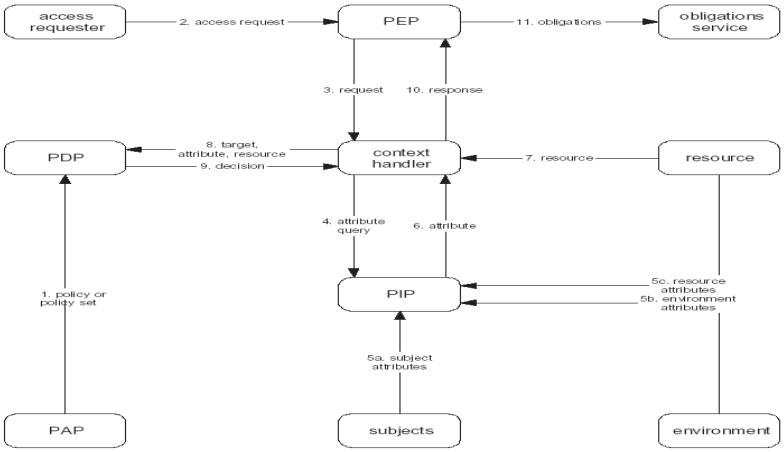

Fig. 2. OASIS XACML dataflow

As shown in fig 2, as a client makes a resource request upon a server; a PEP ***Policy Enforcement Point*** interferes to ensure a secure and authorized access. In order to enforce a security policy, PEP will formalize attributes describing the requester (these attributes can be extracted from the user profile) to the PIP ***Policy Information Point*** and delegate the authorization decision to the PDP ***Policy Decision Point***. Applicable policies are located in a policy store and evaluated at the PDP, which then returns the authorization decision. Using this information, the PEP can deliver the appropriate response to the client and ensures that only authorized resources are accessed [3].

From a business perspective, XACML is convenient to PEIS and serves as a decentralized service-oriented architecture that enables distributed settings of critical business rules and security policies. This is done by providing fine-grained access control that would take the business relationship contracts into account. In addition, the XACML authorization logic enables the abstraction of central applications so that an enterprise would be able to manage authorizations from central locations [16].

From a usability perspective, XACML allows a seamless decentralization but from a managerial decision making perspective, XACML ensures an interoperable interaction of different business policies and centralized decision making.

Access control has a significant importance in guaranteeing that employees are obtaining access according to their position in the organizational chart which is expressed as employee's role. Therefore, OASIS has defined a profile for the use of XACML in expressing policies for Role Based Access Control RBAC [15].

After mentioning the benefits of XACML as an interoperable security protocol for service oriented architecture, we'll present an introduction to healthcare systems as an example of a PEIS and we'll show the different challenges that face the interaction of their subcomponents.

4 Inspiring Scenario

The particularity of pervasive information systems lies in their composition of highly interactive subcomponents that cooperate seamlessly in order to satisfy user needs which is our essential motivation.

Our scenario takes place in a pervasive healthcare system and more precisely in an emergent case where a patient is being transported in an ambulance. The user of the system in this scenario is the treating nurse that is handling the patient in a critical status and trying to control his situation. In such context, the system would enable the nurse to gain access to authorized parts of the patient record which exists in the system's database. This access is usually controlled to maintain the system's integrity.

Our problem appears when the nurse connects to the system - using a mobile device - and executes a query to access a **non-authorized** attribute in the patient's record. A typical system response would be that access is **denied**. There, the nurse will try to make several requests for relevant elements that might or might not be authorized due to security constraints (attached to her role as a nurse). In such urgent cases, time is critical and there might be some authorized elements that could help the nurse in finding achieve her mission without breaking the rules.

Our proposition highlights the importance of having an adaptive service-oriented mechanism that would employ some predefined semantics in order to connect

unauthorized attributes with others that are authorized and relative. This way, the nurse would have a reactive and proactive system providing alternative solutions that might answer demands in such cases.

5 Introducing Adaptation to Access Control

Our system's security policy will be based on XACML (eXtensible Access Control Markup Language) [14] following the RBAC model (XACML RBAC profile [15]).

The basic concept of the RBAC model [9] is to assign different actions to different groups of users. This is accomplished by giving groups of users certain roles, then assigning permissions to these roles and finally users would acquire permissions by being members of certain roles.

The definition of an RBAC model for an enterprise hierarchy offers many benefits such as the ease of administration of security policies, scalability and having a model that follows the organization structure and allows fine-grained access control. Thus in our example, a nurse would be a member of the group of nurses of a certain hospital and would thus have access to certain resources assigned to the role "Nurse".

The exchange of medical information is traditionally ruled by strict sharing policies to protect the patient's privacy but these rules may face exceptions in particular situations (e.g. in an emergency case) [5], evolve over time (e.g. depending on the patient's treatment) and be subject to provisional authorizations [12]. Our proposition aims to provide a balance between strict security procedures and other procedures that break the rules ensuring that there are each case would have a justified context and that ours is where alternative solutions exist and could help in preserving the patient's privacy and the system's integrity.

In fig 3, we present an XML schema of a patient's medical record. Following the XACML RBAC model, if two users (a doctor and a nurse) demand access to this record, the schema will be viewed differently according to the user's role. This is due to the variation of access privileges to the system between a doctor who's allowed to access the whole schema (fig 3) and a nurse who can only view a part of it (fig 4).

As we have discussed, our case concerns a request for consulting a distributed database using a service-oriented architecture and that's where the XACML policy is usually employed for guaranteeing secure and interoperable access. Following the policy's dataflow, the system would evaluate a user's request according to his/her access rights and that's where the access request would be judged to be a **permit**, **deny** or **intermediate**.

Returning to our scenario, supposing that the nurse might request access to the patient's ***Clinical Exam*** in order to evaluate the patient's condition, this request will be verified at a central level of the XACML structure. As we can see in fig 4, the nurse is not authorized to obtain access to the patient's ***Clinical Exam***. Eventually, in this case, the system's ***PDP*** would reply Context Handler's request and would judge the request with a ***Deny***.

The model we're introducing in detail in the next section would provide XACML with a transparent interactive and proactive mechanism. The proposition would enable administrators to anticipate user needs even when they're requesting access to non-authorized parts of a document and would propose some adaptive and alternative solutions to recover the access failure that might occur.

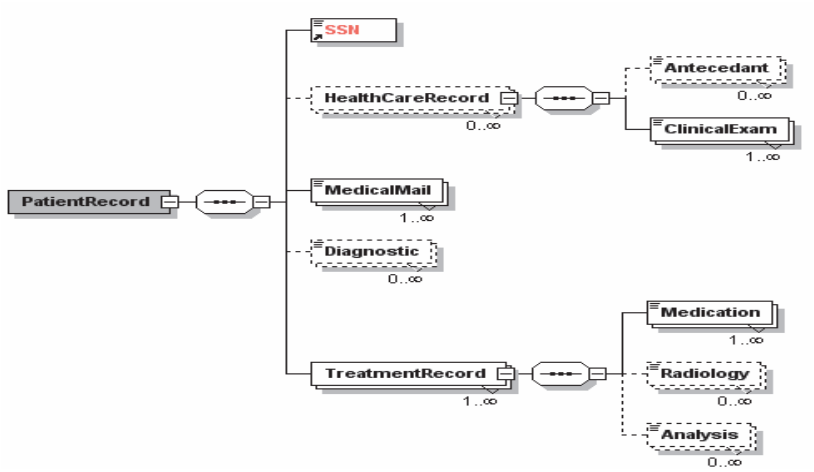

Fig. 3. A simplified XML Schema for a medical patient record (a Doctor's view)

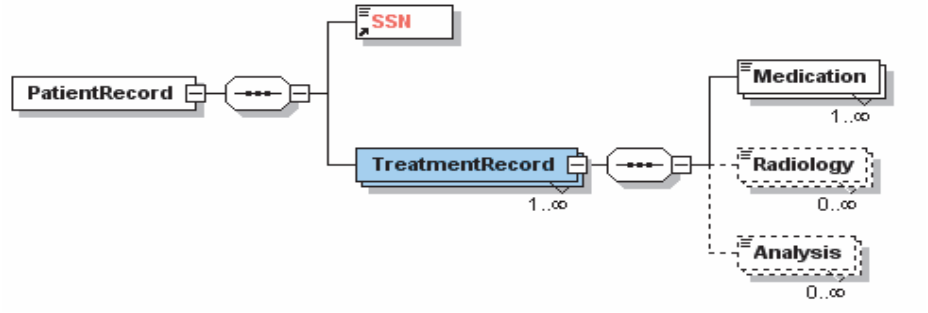

Fig. 4. The authorized part of the patient's record which the nurse can access

6 The MAAC Model: Mutually Adaptive Access Control

As pervasive and ubiquitous environments need special architectures and design [11], we present an adaptive system architecture called MAAC that aims to provide alternative solutions to unauthorized access requests. As we show in fig. 5, our model contains different components and the sequence of its functionality starts from the user, who enters the system by being authenticated (step 1) and then requests access to a certain element (step 2). This request will be interpreted by our ***Query Interpreter*** that will translate the request into an XACML request and would send it to the ***Query Analyzer*** (step3). The request will be analyzed in consideration with the user's profile - that would be automatically produced at the sign in process - and according to his context (XACML flow chart). As the analysis finishes, the Query Analyze would send the result directly to the user if it's a Permit (step 4a) or back to the ***Query Interpreter***, if it's a **deny** (step 4b).

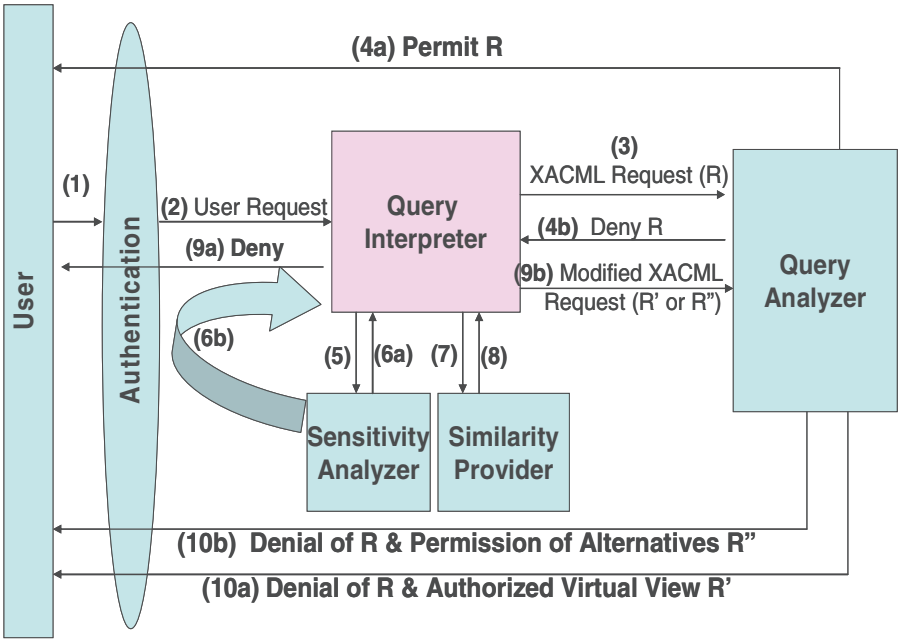

Fig. 5. The MAAC (Mutually Adaptive Access Control) Model

At this step, the adaptive procedure can take place in order to help the user in attaining information that are relevant to their requests instead of responding with an access denied result. Our model helps the user to obtain access to the system but meanwhile enforces some protective procedures in order to be sure that this adaptation wouldn't allow intruders to access undesirable elements. This protective procedure is required after checking the sensitivity of the resource from the ***Sensitivity Analyzer*** (Step 5) which could vary from 0 to 3 and is set explicitly by system administrators at the application side.

So according to the resource's sensitivity, the reaction of the system will change. Starting for example with the sensitivity value that equals to 0 (Sen = 0), the ***Query Interpreter*** would take this result (step 6a) and start a **Virtual View Adaptive Access Control (V-VAAC)** procedure where it would rewrite the user's query using the RBAC model and provide him with a virtual view containing all the elements that he could access in his current context, this modified request **R'** would be sent back to processing by the ***Query Analyzer*** (9b) which would return to the user the denial of the initial request R and the result of the request R' which is a virtual view containing all the authorized elements that the user can access that moment (Step 10a).

Taking the next choice where the sensitivity checked might be equal to 1 (Sen = 1). In this case, the system should be sure of the identity of the user so it will demand from the user to reauthorize himself using a more powerful authentication (Step 6b) and would then proceed making the **V-VAAC** procedure mentioned above.

Assuming that the sensitivity checked equals to 2 (Sen = 2), the ***Query Interpreter*** would take this result (step 6a) and start and would proceed with a **Similarity-based Adaptive Access Control SAAC** procedure. In this procedure, the ***Query Interpreter***

would search for Similarities with the help of the ***Similarity Provider*** (steps 7 and 8) and would use these values to reformulate the user's initial request R to a new request containing these similar elements R". This new request will be also analyzed at the ***Query Analyzer*** (9b) which would finally return to the user the denial of the initial request R and the proposition of providing some alternative solution of the request R" (Step 10b).

Finally, assuming that the sensitivity checked is equal to 3 (Sen = 3), the system will demand from the user to reauthorize himself (Step 6b) and would then proceed making the **SAAC** procedure mentioned above.

Working in the context of semi structured documents, similarity between elements can be added to the system in the form of an XML file created at the application side and would be then provided to the ***Similarity Provider***. In such operation, calculating similarity between elements would enrich the system's semantics.

For each role, we can provide a similarity coefficient between elements so if a system wants to allow strict adaptation, the querying process will check if the demanded resource is allowed and if not, it would try to check the elements having high similarity coefficient.

Adding semantics to the elements of data sources can be performed explicitly by system administrators, data owners or implicitly by using a parser that would compare the structure of the demanded XML file and the other XML documents that are authorized to the user.

For example, consider the case where a nurse has requested to obtain access to the patient's latest medical exam (which doesn't exist in her authorized view). Here, our system would use the nurse's profile containing his/her viewing preferences, current terminal, connection, location, situation and access privileges along with the similarity coefficient between the clinical exam and other relevant elements in order to evaluate his/her request in order to permit or deny her/his access request.

When demanding an unauthorized element, the system would notify the nurse of not having the right to access the demanded element and would respond in one of 2 ways: either a reactive way (**V-VAAC**) by providing the authorized scheme (in a tree format) where the nurse would check whether he/she can access another relevant element or in a proactive way **(SAAC)** by providing her with some elements that she might need by using predefined semantics. As we show in figure 4, our system would propose to the nurse to choose between the authorized elements or would proactively suggest the consultation of the patient's treatment record.

Finally, in order to choose alternative elements instead of the unauthorized requested element, the system would search for relevant element and would check their convenience according a precision degree précised by the administrator, for example it might follow this condition:

If Sim (Patient.Clinical_Exam, Patient.*) $\geq$ 70% then
    Reformulate **R** by **R"**

The calculation operation can be done at the **Query Interpreter** and thus, the similar elements that are retrieved using this operation will be embedded within the reformulated request R" instead of the originally demanded element.

7 Conclusion

In this paper, we have presented an adaptive solution for healthcare pervasive systems that offers flexible authorization and access in urgent situations. Our solution highlights the importance of using Role Based Access Control for easier distribution of access rights within distributed healthcare systems. In order to accomplish our adaptive vision, we have analyzed the functionality of XACML – a widely used access control policy within service oriented applications – showing that it provides Boolean solutions for access. Thus, our proposition aims to provide users of pervasive services with balanced solutions and adaptive accessibility - based on similarities and semantics - to meet their needs and satisfy the security requirements that the system imposes.

References

1. Al Kukhun, D., Sèdes, F.: A Taxonomy for Evaluating Pervasive Computing Environments. In: IEEE International Conference on Pervasive Systems, MAPS 2006 proceeding, Lyon, 26/06/06-29/06/06, pp. 29–34 (2006)
2. Al Kukhun, D., Sèdes, F.: Interoperability In Pervasive Enterprise Information Systems: A Double-Faced Coin Between Security And Accessibility. In: International Conference on Enterprise Information Systems (ICEIS 2007), Funchal, Madeira - Portugal, 12/06/07-16/06/07, pp. 237–243. INSTICC Press (2007)
3. Anderson, A.: A Comparison of Two Privacy Policy Languages: EPAL and XACML, consulted on 8/12/2007 (September 2005), `http://research.sun.com/techrep/2005/smli_tr-2005-147/abstract.html`
4. Bertino, E., Castano, S., Ferrari, E., Mesiti, M.: Specifying and Enforcing Access Control Policies for XML Document Sources. World Wide Web Journal 3(3), 139–151 (2000)
5. Bouganim, L., Dang Ngoc, F., Pucheral, P.: Client-Based Access Control Management for XML Documents. In: Proc. of the Very Large Data Bases Conference, Toronto, Canada (2004)
6. Bray, T., et al.: Extensible Markup Language (XML) 1.0. World Wide Web Consortium (W3C) (October 2000), `http://www.w3c.org/TR/REC-xml`
7. Damiani, E., Vimercati, S.D., Paraboschi, S., Samarati, P.: Securing XML Documents. In: Zaniolo, C., Grust, T., Scholl, M.H., Lockemann, P.C. (eds.) EDBT 2000. LNCS, vol. 1777, pp. 121–135. Springer, Heidelberg (2000)
8. Duan, Y., Canny, J.: Protecting User Data in UbiComp: Towards trustworthy environments. In: Martin, D., Serjantov, A. (eds.) PET 2004. LNCS, vol. 3424, pp. 167–185. Springer, Heidelberg (2005)
9. Ferraiolo, D.F., Kuhn, D.R.: Role Based Access Control. In: 15th National Computer Security Conference, October 1992, pp. 554–563 (1992)
10. Gabillon, A., Bruno, E.: Regulating Access to XML documents. In: Fifteenth Annual IFIP WG 11.3 Working Conference on Database Security, July 15-18, 2001. Niagara on the Lake, Ontario, Canada (2001)
11. Helal, A., Hammer, J.: UbiData: Requirements and Architecture for Ubiquitous Data Access. SIGMOD RECORD 33(4) (December 2004)

12. Kudo, M., Hada, S.: XML document security based on provisional authorization. In: Proceedings of the 7th ACM CCS 2000, pp. 87–96. ACM, New York (2000)
13. Munoz, J., Pelechano, V.: Building a Software Factory for Pervasive Systems Development. In: Pastor, Ó., Falcão e Cunha, J. (eds.) CAiSE 2005. LNCS, vol. 3520, pp. 342–356. Springer, Heidelberg (2005)
14. OASIS, A brief Introduction to XACML, consulted on: October 15, 2007 (March 14, 2003), http://www.oasis-open.org/committees/download.php/2713/Brief_Introduction_to_XACML.html
15. OASIS, XACML Profile for Role Based Access Control (RBAC), consulted on: 15/10/2007 (13/2/2004), http://docs.oasis-open.org/xacml/cd-xacml-rbac-profile-01.pdf
16. Seeley, R.: SOA governance, security concerns drive XACML interop (posted on 13/6/2007), http://searchwebservices.techtarget.com/originalContent/0,289142,sid26_gci1260713,00.html