Journal of Universal Computer Science, vol. 15, no. 12 (2009), 2429-2444
submitted: 15/12/08, accepted: 25/6/09, appeared: 28/6/09 © J.UCS

Modeling of an Intelligent e-Consent System in a Healthcare Domain

Chun Ruan
(University of Western Sydney, NSW, Australia
c.ruan@uws.edu.au)

Sang-Soo Yeo
(Division of Computer Engineering, Mokwon University, Daejeon, Korea
ssyeo@msn.com)

Abstract: Due to rapid advances of computing power and communications, healthcare services are increasingly rely on the electronic processing and transmission of confidential patient data to reduce the costs and improve the quality. It is becoming more and more important that accessing the health information should be both secure and privacy preserving. Therefore access control becomes an important integral part of any secure healthcare computer software systems. Specification of access control requirements at early steps of the software life cycle can provide stakeholders rapid feedback and protect the system in a best possible way. On the other hand, intelligent systems are widely used in various computing areas ranging from medicine to manufacturing industries to financial markets. This paper studies how to model an intelligent e-Consent system about the security requirements regarding healthcare information protection. In this paper, we use UML to specify and visualize the access control policies in a health application domain. These policies are represented in logic based e-Consent rules, and the patient's consents about their information access can be derived from these rules. We first identify various parts necessary to specify the e-Consent rules about patient record protection requirements, and then propose UML models to demonstrate these requirements.

Keywords: UML, access control, e-consent
Categories: H.4.3, J.7

1 Introduction

Intelligent computing has emerged as an exciting new paradigm that includes ubiquitous, peer to peer, and green computing to provide intelligent communications at anytime and anywhere. Intelligent environments are expanding into our real lives such as entertainment, health care, smart homes, security etc. The digital computer and information technology have changed our society, including medical society. More and more coordination of health care relies on the electronic transmission of confidential information about patients between different intelligent health care and community services. However, since the patient data is confidential, the need for electronic forms of patient consent, referred to as e-consent [Coier, 04], has to be considered. Patients should be able to delegate, give or withhold 'e-consent' to those who want to access their electronic health information. That is, the secure intelligent health information technology needs to support confidential patient and service provider interactions. The main application areas that need e-consent are those that

support coordinated health care. This is characterized by sharing patient data among multiple teams of health care professionals and institutions. Without the existence of some e-consent mechanism, such widespread information could be accessed by unauthorized individuals or used for purposes not originally consented to by the patient, which can lead to substantial breaches of personal privacy. By using the e-Consent, the patients are able to actively participate in the governance of the intelligent health services they need. To take advantage of strong expressive and reasoning power of logic programs, e-Consent can be expressed in logic rules enabling patients to express various conditions for the consents to be granted as well as new consents to be derived from the rules.

System analysis is an important phase in a software development lifecycle. It is important for the developers of software systems to fully understand the users' business requirements before going into the coding stage. They must understand what the system must do in order to service its purpose. A system description, or a model, is used to capture and precisely state requirements and domain knowledge so that all stakeholders may understand and agree on them. It is used to grasp conceptually what the components are and how they interact to carry out the system functions and objectives. Stakeholders include the end users, clients, architect, analysts, programmers, project manager, and funders. The model is also used to guide the developer to explore design solutions before writing code. A model of a software system is made in a modeling language, such as the Unified Modeling Language (UML) [Rumbaugh, 05]. UML is a widely accepted standard visual modeling language that is used to specify, visualize, construct, and document the artifacts of a software system. It captures decisions and understanding about systems that must be constructed.

Access control systems have become an important part in any secure computer software systems. Specification of these requirements at early steps of the software life cycle can provide stakeholders rapid feedback and protect the system in a best possible way. Although UML is widely used to model the software requirements, the work on UML specification for security purpose remains limited. In this paper, we propose a UML model to represent security requirements regarding an intelligent e-Consent system in a health care domain. We first identify various parts necessary to specify the e-Consent rules about patient record protection requirements, and then propose UML models to demonstrate these requirements. We utilize use case diagrams, class diagrams and activity diagrams etc to visualize and demonstrate the access control requirements for electronic patient records.

The rest of the paper is organized as follows. Section 2 gives a simple introduction of UML, while Section 3 discusses the role-based access control model and e-Consent. Section 4 presents various aspects that are taken into account for e-Consent. Section 5 describes the proposed UML models for e-Consent. Section 6 discusses related work. Finally section 7 concludes the paper.

2 UML

The UML is a de-facto standard modelling language for analysis and design of software systems issued by the Object Management Group (OMG) [Rumbaugh, 05]. The primary purpose of the UML is visualizing. Its notions and diagrams provide

industry standard mechanism to represent pictorially the requirements. In this paper, we use the following diagrams:

- Use case diagram: A use case diagram models the requirements of the system at a high level and facilitate understanding the business processes. It is used to visualize the use cases, actors and their interactions.
- Class diagram: A class diagram depicts the static structural aspects of an artifact being modelled. It represents properties through attributes, and behaviours through operations. Class diagrams can also show the relationships between classes, such as associations, aggregation and inheritance.
- Activity diagram: An activity diagram depicts the flow of activities in the system. It can model the dependency between the activities, the decision point enabling branching of the activities based on conditions specified, and the synchronization through multiple threads. It also helps in mapping the activities to corresponding actors.
- State machine diagram: A state machine diagram models the states of objects, and their transitions. It shows the states that an object could go through in its life cycle.
- Interaction overview diagram: An interaction overview diagram provides a high-level overview of interactions happening in the system. It also shows dependencies and flows between use cases.
- Sequence diagram: A sequence diagram models dynamic interactions between actors and objects.

3 Role-based access control and e-Consent

Classic access control is based on the individual subject accessing a resource (object).

subjects -> objects

Sometimes privileges are associated with roles other than individuals. Individuals get their privileges because their roles or positions in the organization. In other words, whoever gets the role would get the privileges of the role. When people leave the organization or change the positions, their privileges will be revoked or changed, too. This happens in many organizations from the viewpoint of organization administration. Fox example, a doctor in a hospital can access the patients' information in the hospital. If the doctor leaves the hospital, he/she usually lose the capability to access the patients' information, too. If the number of subjects and objects is large, individual access control becomes difficult. Each individual needs to be assigned each access right when they get a position in the organization and revoked each access right if the person changes the role or leaves the organization. When privileges are indeed assigned to roles other than individual subjects, role-based access control (RBAC) developed by Sandhu et al can greatly simplify the administration work [Sandhu, 96, Sandhu, 99].

In role-based access control, roles are placed between the user and the resource and subjects get their access rights indirectly by assigning access rights to roles and

roles to subjects. Roles describe rights, duties and tasks that people have to perform. When people leave or change roles, only the mapping from subjects to roles needs to be revoked or changed. On the other hand, if the duties of the roles change, only the mapping from roles to objects needs to be changed. Roles provide a more abstract viewpoint on access control.

subjects -> roles -> objects

The concept of role also applies to the provision of patient data in intelligent health care contexts. Some consents may be given by patients in relation to roles. For example, a patient may consent to have a pathology test done by the clinical lab staff. Multiple individuals may perform particular roles at different times, e.g. because of the need for shift-work in both intensive-care and extensive-care. Roles can be organised into hierarchies so that consents can be inherited, which could greatly reduce the amount of explicit consent specification.

4 Necessary parts for e-Consent rules

Consents may involve subjects to whom the consents are given, objects (data) to be protected, access rights allowed or prohibited on the information, and grantors who issue the consent. Consents may also be given based on purposes for the usage of data, or context of this consent. In addition, consents may be assigned for only a certain period of time.

Subjects: Roles, Individuals and Organizations

In the context of e-Consent for intelligent health care, the consent may be assigned on the basis of an individual's identity such as “Dr Smith”, or a clinical role within an organization such as Physician, or an organization such as Nepean Health Research Center. Roles can be organized into different hierarchies so that the consent can be inherited.

In this paper, we consider three types of subjects: role, individual person and organization. Organizations can be research centers, clinics or hospitals. Roles are classified into medical roles and management roles. The medical roles include doctors, nurses, and pathology collectors. The management roles include receptionists, system administrators, and practice manager.

Objects: Patient Data

In general, the data about a patient include: personal and contact details, clinic related details, and health details. To allow consent inheritance along the data dimension, data could be organized into hierarchies.

Access Rights

Usual access rights such as read, write, and update apply to the patient data. Access rights can also be organized into hierarchies to allow inheritance along this

dimension. For example, a consent to updating may also imply a consent to reading and writing.

Consent, Denial and Delegation

Both consents and denials are needed in a flexible e-consent system. Denials are useful when patients want to express explicitly that some disclosure is forbidden. In some circumstances, a patient may wish to delegate the capability to grant consent to nominated representatives or medical practitioners, who may further wish to delegate the power to consent to other health professionals. This is usually done for flexibility, cooperation, and convenience of the carer.

Conflict Resolution

Because of consent delegation, multiple grantors may exist for a specific consent and hence conflicts may arise. For example, a patient may wish to deny all information relating to HIV to be open to a research organization, but his/her family GP who has been delegated the privilege of consent granting may wish to do so. In this case, the organization may receive two conflicting authorizations, consent and denial. A proper conflict resolution policy is thus needed. For instance, predecessor-take-precedence method can be used when the grantors of the two authorizations have predecessor-successor relationship in terms of the delegation relationship. In this case, the authorization from the predecessor will override that from the successor. Otherwise, the more specific- take- precedence method can be used which favors the more specific authorizations. If neither applies, we can always use the authorization type to solve the conflicts by assigning an order to the types. For example, to achieve the maximum security, we can assign higher priority to denials than consents.

Purposes and Contexts

Sometimes, a consent is assigned on the basis of specific use of information. Common purposes include treatment, cooperation, training, teaching, notification (requests by persons closely associated with the person concerned, such as guardians, partners and immediate family), research, and getting advice from specialists.

Sometimes, consent is assigned based on the current context. A doctor may not be allowed to read the patient's health data in a normal situation, but may be allowed to do so in an emergency situation.

Logic based e-Consent rules

To take advantage of strong expressive and reasoning power of logic programs, e-Consent rules can be specified in Extended Logic Programs (ELP) that supports both classical negation and negation as failure. ELP has strong expressive power in the sense that it can deal directly with incomplete information in representation and reasoning. As incomplete information is a common issue when it comes to patient data protection issues, security policies are easier to specify in extended logic programs. For example, we can specify a consent rule that medical staff in a patient's

hospital by default can access the patient data, while other people by default cannot access the patient data.

5 UML model for e-Consent

The access control requirements come from various sources such as domain experts and patients, and it is important to represent these requirements formally in UML models.

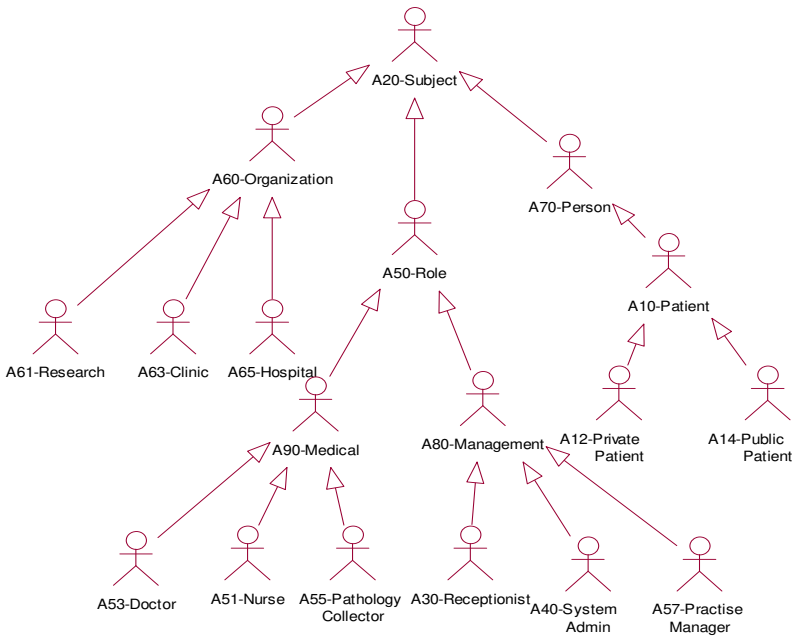

Figure 1: Role hierarchy

5.1 Use case diagram

We use Figure 1 to describe the Actor hierarchy, which represents users of the intelligent e-Consent system. The hierarchy reflects the inheritance relationship specified by arrows. For example, a Doctor inherits all the characteristics of a Medical Role which inherits Role which again inherits all the characteristics of a Subject. Inheritance provides opportunities to reduce the complexity on the e-Consent system. For example, if some patient wishes to give consent to all medical staff in an organization, instead of giving a consent to every individual medical staff, it is sufficient for him/her to give a consent to the medical role in the organization. The intelligent system will automatically propagate the consent along the hierarchy.

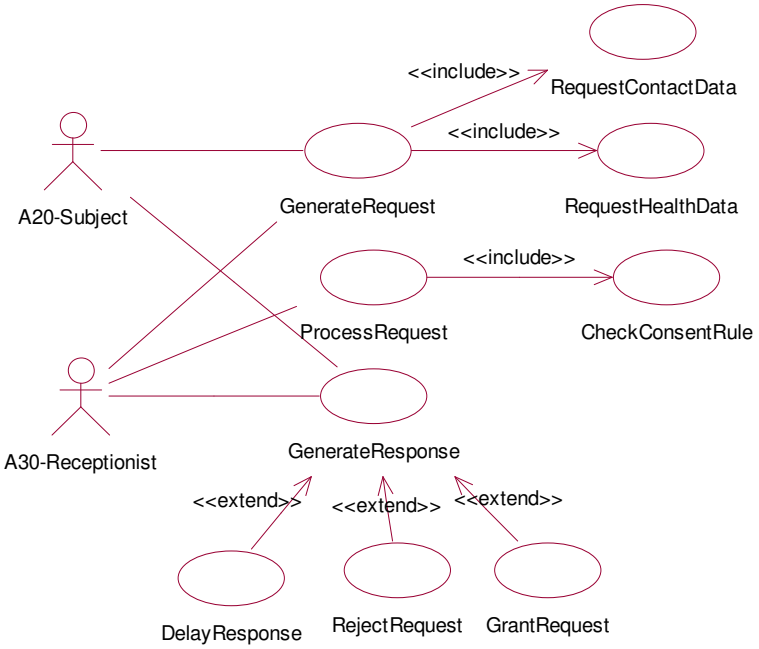

Figure 2: Use case diagram for request process

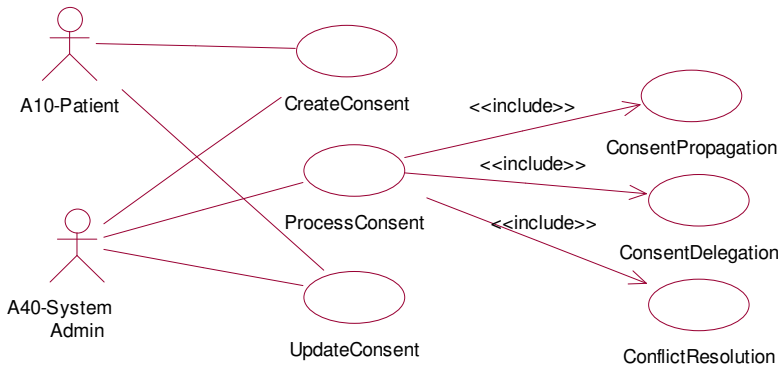

Figure 3: Use case diagram for consent maintenance

In the Request Process use case diagram as shown in Figure 2, the subject and receptionist are actors. The subject is an abstract actor that represents a patient, a role or an organization. The subject makes a request to access the patient records which include two use cases that request contact data and health data respectively. The

system processes the request based on the results of reasoning on the e-Consent logic programs, and generates the response to the subject. There are three possible responses to generate: grant, reject or delay (undecided) denoted by three extended use cases.

In the e-Consent Maintenance use case diagram shown in Figure 3, the patient and SystemAdmin are actors involved. The use cases about creating e-Consent rules, updating e-Consent rules, and reasoning on e-Consent rules reflect major e-Consent maintenance activities. In particular, reasoning on e-Consent will need to consider consent propagation along hierarchies of subjects, objects and access rights, conflict resolution, and consent delegation. The well-known answer set semantics for ELP will be used.

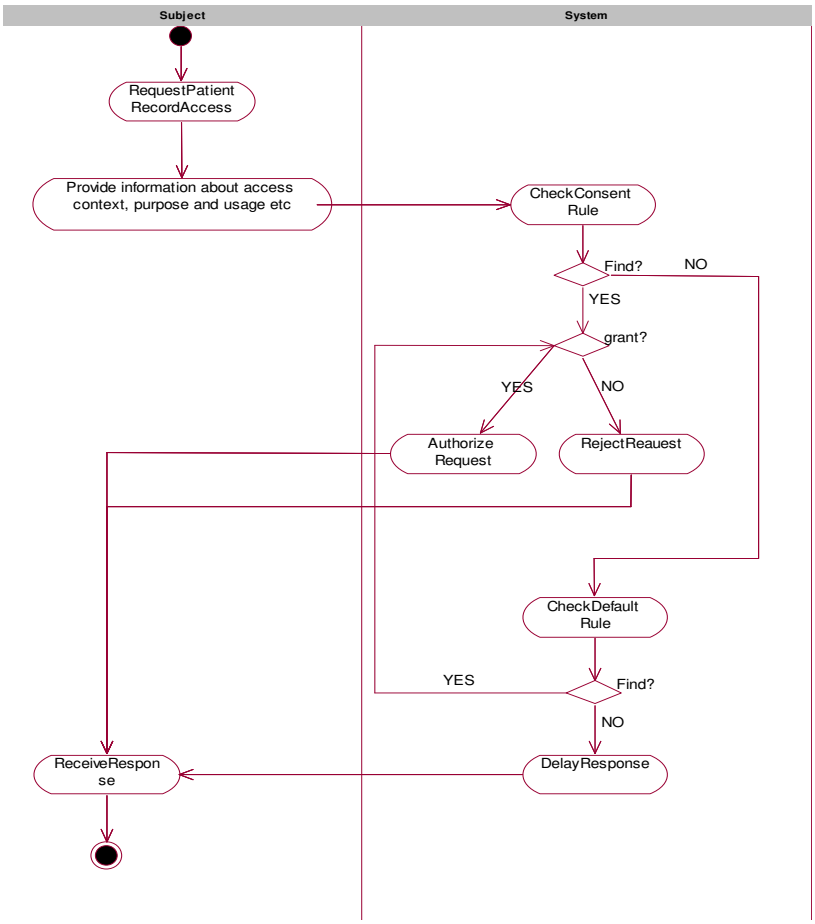

Figure 4: Activity diagram for access control

5.2 Activity diagram

Figure 4 demonstrates the access control activities on patient data based on the e-Consent rule system. For a request from a subject, the system generates a response based on the consent rule if it exists. Otherwise it generates a response based on the default rule (what to do by default) if it exists. Otherwise the response is undecided.

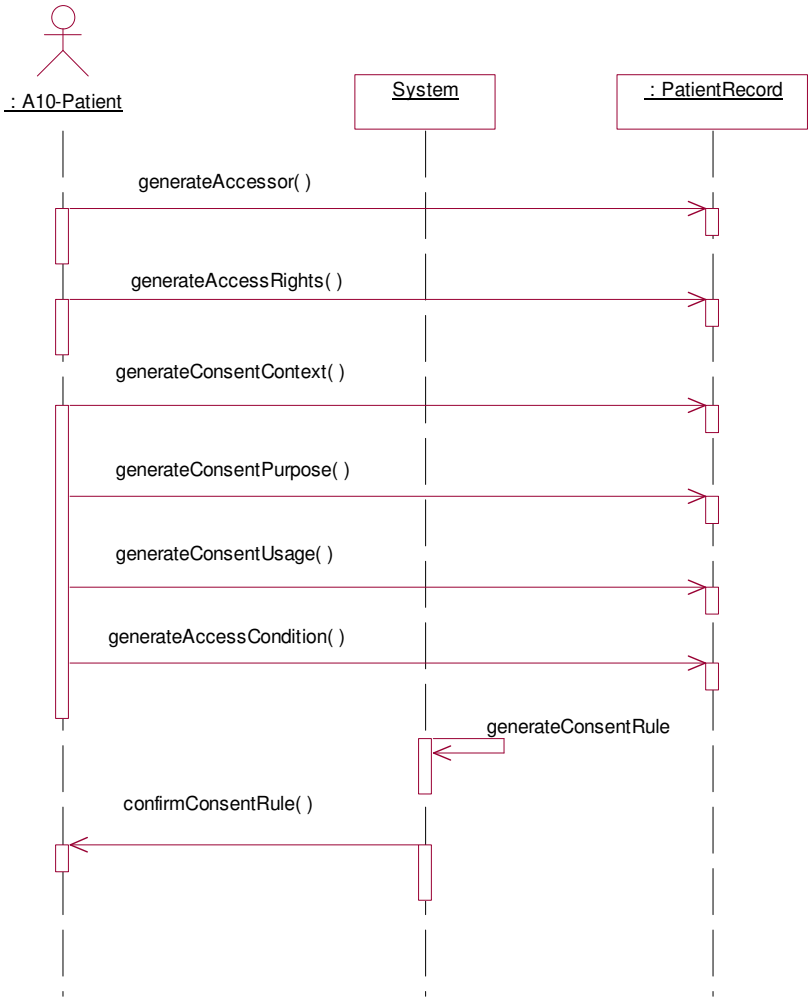

Figure 5: Sequence diagram for CreateConsent

5.3 Sequence diagram

Sequence diagrams can visualize the activities and message interactions within a use case. For example, we use Figure 5 to visualize the activities happening within the use

case CreateConsent. The patient informs the system who can access their records in what contexts, for which purposes, and in what way and conditions. The system then generates the corresponding consent rules and confirm with the patient.

5.4 Class diagram

Figure 6 is about e-Consent. The class Consent is associated to subjects, patient records, access rights, types, contexts, purposes and time period. In addition, consent propagation, delegation and conflict resolution are considered in its operations. Figure 7 is about patient record. The class PatientRecord inherits class Person, which has association relationship with class ContactData. The PatientRecord is an aggregation of classes Consultation and HealthData. Also, the PrivatePatient inherits the attributes and operations of the class Patient. Figure 8 is about subject. The class Subject is a generalization of classes Role, Person and Organization, which have association relationships with each other. Due to the space limit, we omit some attributes and operations in the class diagrams.

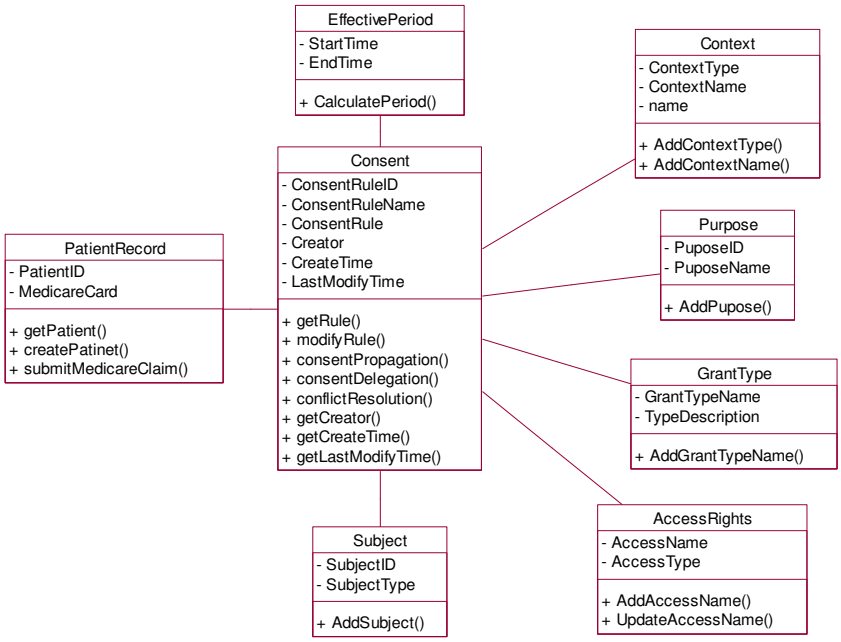

Figure 6: Class diagram for consent

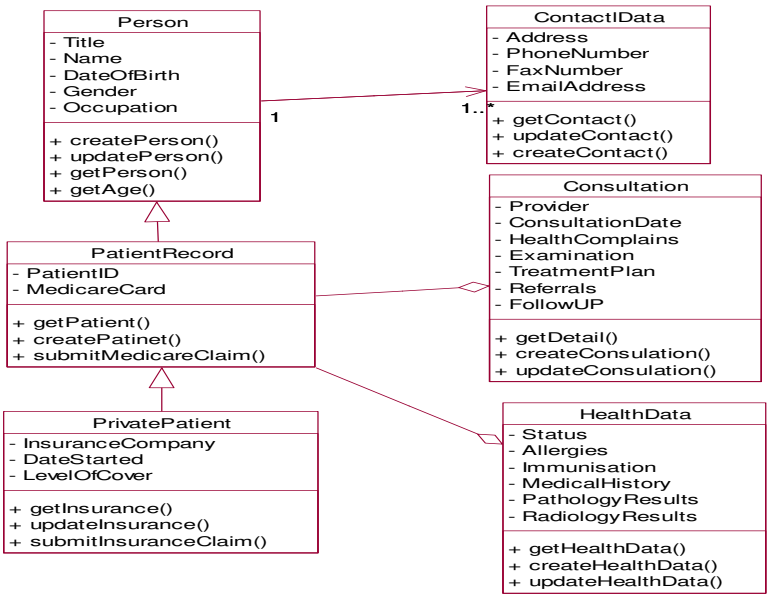

Figure 7: Class diagram for patient record

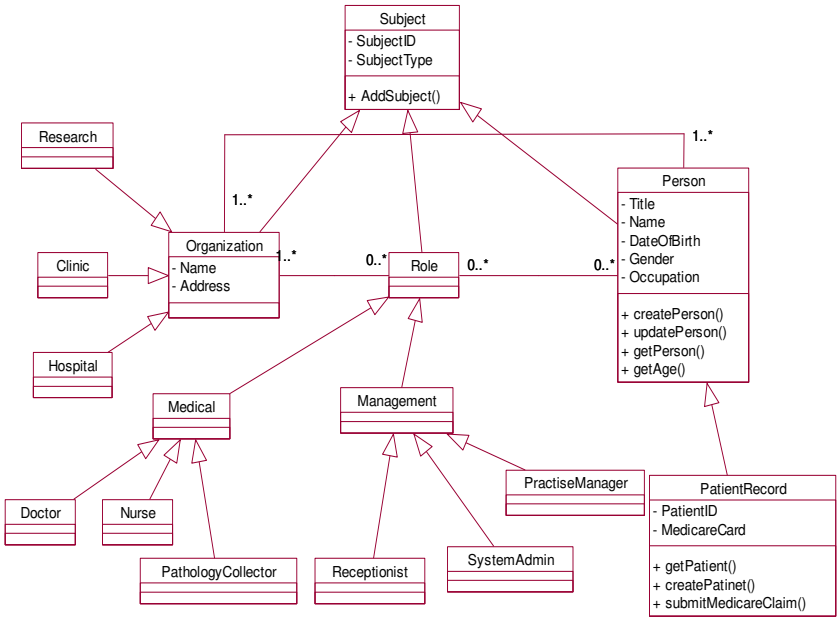

Figure 8: Class diagram for subject

5.5. State machine diagram

Figure 9 shows how to use the state machine diagram to describe the states and their transitions for an object. Here the object we choose is the e-Consent. The e-Consent will go through a number of states in its life cycle. It should be submitted first by the patient. The e-Consent is then repeated processed with delegation, propagation and conflict resolution until the conclusion is made. This ends the state transitions.

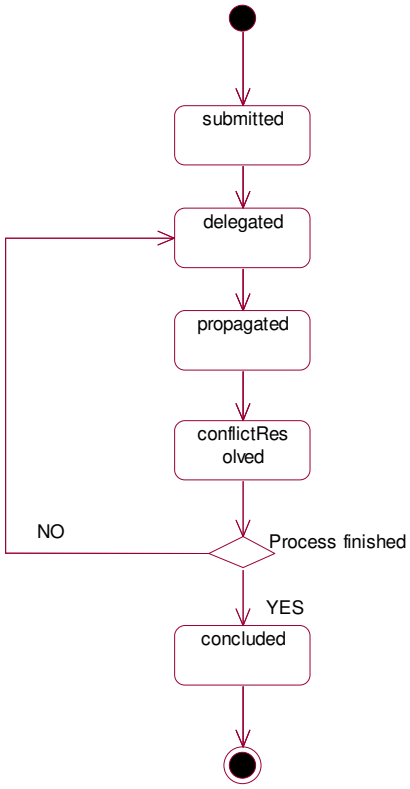

Figure 9: State machine diagram for e-Consent

5.6 Interaction overview diagram

As there are some sequence relationships between the use cases in Figure 2, we use the interaction overview diagram, as shown in Figure 10, to represent this high level flow and dependencies. Please note that the use case of ProcessRequest can be repeated several times before a conclusion is reached.

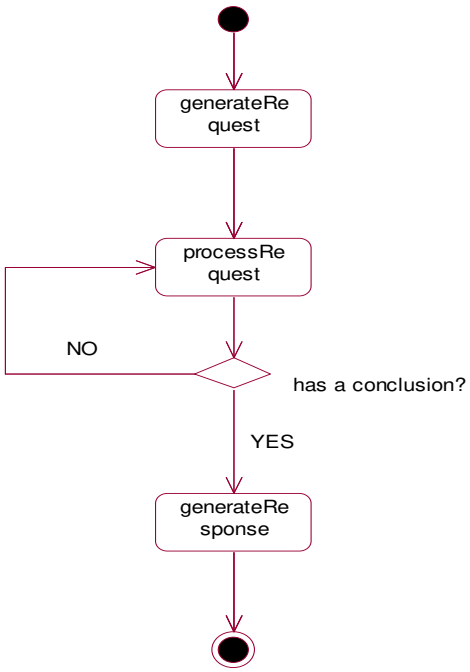

Figure 10: Interaction overview diagram for request process

6 Related work

Although a lot of work appears in the area of security policies for e-Healthcare services, UML modelling of e-Health security requirements have received relatively little attention. In [Raistrick, 05], Raistrick describes how MDA and UML were used to model the new access control capabilities, specify the capabilities of existing key components and facilitate system integration. However, the work is only focused on using class diagrams to model the access control requirements for the patients' data, while our work has utilised various UML diagrams to model different aspects of the security requirements. There has been some separate research work on UML modelling of the security requirements in a general domain [Basin 04, Juerjens, 02, Alghathbar, 03, Jurjens, 05, Lodderstedt 02]. The difficulties of modelling security requirements due to lack of systematic support for software engineers are discussed in [Devanbu 00, Nuseibeh, 00]. A way to consider several perspectives about security requirement modelling is presented in [G. Herrmann, 02]. Herrmann et al take into consideration of the static, functional and dynamic security requirements from the life cycle of the objects in a business process. In [Koch, 05], Koch et al present a way of specifying role based access control requirements by UML use case and sequence diagrams. In [M. Koch, 06], Koch further proposes a methodology to integrate the specification of access control policies into UML and provided a graph-based formal

semantics. Access control policies are specified by means of UML class and object diagrams. The diagrams are then translated into graphs and graph rules for the purpose of checking the coherence of an access control specification. In [Rodriguez, 06], Rodriguez et al presents a way to use UML 2.0 profile for security requirements modelling in a business process through activity diagrams. In [Rodriguez, 07], they further present a Business Process Modelling Notation metamodel with extension through artifacts that can incorporate security requirements into Business Process Diagrams. In [Ray, 04], Ray et al show how RBAC constraints can be specified by object diagrams representing forbidden object states. It also shows how to use class diagrams to represent RBAC features. Another work is presented in [Darimont, 07] about the security requirements specification with UML in the context of civil aviation. Differently from their work, in this paper, we will present the requirements engineering process for security purpose in the context of e-consent in the intelligent health care domain. We will investigate how various diagrams supported by the current UML system, such as use case diagrams, state machine diagrams, interaction overview diagrams, class diagrams, sequence diagrams, class diagrams etc, can be used to model the various aspects of e-Healthcare security requirements.

Access control in healthcare services is a better researched area. There has been considerable research work on protecting privacy [Senicar 03]. Over the years, different models for authorization and access control for electronic patient record (EPR) have been proposed to facilitate a wide scale use of EPR in large health organizations [Reid, 03, Louwerse, 98, Anderson, 00, Motta, 03, Varadharajan, 96]. The main objective is to support the patient privacy and the confidentiality of patient data, whereas being flexible enough to facilitate collaborations between medical practitioners and to deal with special cases such as emergency treatment.

7 Conclusions

In this paper, we have shown how patient requirements regarding health information protection using e-Consent rules can be specified in UML models. One of the main benefits of the approach has been to raise all access control issues regarding patients' records at the analysis stage of software development process. This enables better communication to stakeholders and reduces the risk of delivering a system that does not meet patients' security needs. For the future work, we intend to extend and enrich the security requirements specifications using UML. We also intend to investigate automated code generation for security requirements from the UML model.

Acknowledgements

This work was supported by the Research Grant Scheme of University of Western Sydney, Australia.

References

[Alghathbar, 03] Alghathbar, K., Wijesekera, D.: authUML: a three-phased framework to analyze access control specifications in use cases. *FMSE (2003)*, pp 77-86.

[Anderson 00] Anderson, J.: Security of the distributed electronic patient record: a case-based approach to identifying policy issues. *International Journal of Medical Informatics*, 60(2):111-118, 2000

[Basin, 06] Basin, D.A., Doser, J., Lodderstedt, T.: Model driven security: From UML models to access control infrastructures. *ACM Trans. Softw. Eng. Methodol*. 15(1): 39-91 (2006).

[Coier, 04] Coiera, E.: e-Consent: The Design and Implementation of Consumer Consent Mechanisms in an Electronic Environment. *Journal of the American Medical Informatics Association*, 11 (2004), pp. 129-140.

[Darimont, 07] Darimont, R., Lemoine, M.: Security requirements for civil aviation with UML and goal orientation. *LNCS 4542*, (2007), pp. 292-299.

[Devanbu, 00] Devanbu, P. T., Stubblebine, S.: Software engineering for security: a roadmap. *In Proc. of 22nd International Conference on Software Engineering, Future of Software Engineering Track*, (2000), pp 227-239.

[Herrmann, 98] Herrmann, G., Pernul, G.: Viewing business process security from different perspectives, *In Proc. of the 11th International Bled Electronic Commerce Conference*, (1998).

[Juerjens, 02] Juerjens, J.: UMLsec: Extending UML for secure systems development. *In Proc. of 5th Int. Conf. on the Unified Modeling Language*. LNCS 2460, (2002), pp 412-425.

[Jurjens, 05] Jurjens, J.: Secure systems development with UML. Springer, 2005.

[Louwerse 98] Louwerse, K.: The electronic patient record; the management of access-case study: Leiden University Hospital, International Journal of Medical Informatics, vol. 49, no. 1, pages 39-44, 1998.

[Koch, 05] Koch, M., Pauls, K., Parisi-Presicce, F.: Generation of Role-Based Access Control Requirements from UML Diagrams. In *Proc. Symp. on Requirements Engineering for Information Security*, (2005).

[Koch, 06] Koch, M., Parisi-Presicce, F.: UML specification of access control policies and their formal verification. *Software System Model,* 5 (2006), pp. 429-447.

[Lodderstedt, 02] Lodderstedt, T., Basin, D., Doser, J.: Secure UML: A UML based modelling language for model driven security. In *Proc. of 5th Int. Conf. on the Unified Modeling Language, LNCS* 2460 (2002), pp. 426-441.

[Motta 03] Motta, G.H.M.B., Furuie, S.S.: A contextual role-based access control authorization model for electronic patient record. *IEEE Transactions on Information Technology in Biomedicine*, 7(3): 202-207, 2003.

[Nuseibeh, 00] Nuseibeh, B., Easterbrook, S.: Requirements engineering: A roadmap. In *Proc. of 22nd International Conference on Software Engineering, Future of Software Engineering Track*, (2000). pp 35-46.

[Raistrick, 05] Raistrick, E.: Applying MDA and UML in the development of a healthcare system. *LNCS* 3297, (2005), pp. 203-218.

[Ray, 04] Ray I., et al.: Using UML to visualize role-based access control constraints. *SACMAT*, (2004), pp. 115-123.

[Reid 2003] Reid, J., Cheong, I., Henricksen, M., Smith, J.: A novel used of RBAC to protect privacy in distributed health care information systems. *Proceedings of the Eighth Australasian Conference on Information Security and Privacy* (2003), LNCS 2727, pp 403-415.

[Rodriguez, 06] Rodriguez, A., Fernandez-Medina, E., Piattini, M.: Security requirement with a UML 2.0 profile. In *Proc. of the First International Conference on Availability, Reliability and Security*, 2006, pp. 670-677.

[Rodriguez, 07] Rodriguez, A. et al, A BPMN extension for the modeling of security requirements in business processes. IEICE *Trans. Inf.& Syst*. 4 (2007), pp. 745-752.

[Rumbaugh, 05] Rumbaugh, J., Jacobson, I., Booch, G.: *The Unified Modeling Language Reference Manual*, Addison-Wesley Publishing Company, 2005.

[Sandu 96] R.S. Sandhu, E.J. Coyne, H.L. Feinstein and C.E. Youman, Role-based access control models. *IEEE Computer*, 29(2):38-47, 1996.

[Sandhu 99] Sandhu, R.S., Bhamidipati V., Munaswer, Q.: The ARBAC97 model for role based administration of roles. *ACM Transactions on Information and Systems Security* (TISSEC), 1(2): 105-135, 1999.

[Senicar 03] Senicar, V., Jerman-Blazic, B., T. Klobucar, T.: Private-enhancing technologies – approaches and development. Computer Standards & Interfaces, 25 (2003), pp.:147-158.

[Varadharajan, 96] Varadharajan V., Calvelli, C.: An Access Control Model and Its Use in Representing Mental Health Application Access Policy, *IEEE Trans. on Knowledge and Data Eng*., 8(1): 81-95, 1996.