Secure Interaction Models for the HealthAgents System

Liang Xiao, Paul Lewis, and Srinandan Dasmahapatra

University of Southampton, UK
{lx,phl,sd}@ecs.soton.ac.uk

Abstract. Distributed decision support systems designed for healthcare use can benefit from services and information available across a decentralised environment. The sophisticated nature of collaboration among involved partners who contribute services or sensitive data in this paradigm, however, demands careful attention from the beginning of designing such systems. Apart from the traditional need of secure data transmission across clinical centres, a more important issue arises from the need of consensus for access to system-wide resources by separately managed user groups from each centre. A primary concern is the determination of interactive tasks that should be made available to authorised users, and further the clinical resources that can be populated into interactions in compliance with user clinical roles and policies. To this end, explicit interaction modelling is put forward along with the contextual constraints within interactions that together enforce secure access, the interaction participation being governed by system-wide policies and local resource access being governed by node-wide policies. Clinical security requirements are comprehensively analysed, prior to the design and building of our security model. The application of the approach results in a Multi-Agent System driven by secure interaction models. This is illustrated using a prototype of the HealthAgents system.

Keywords: Clinical Information System, Multi-Agent System, Security Model.

1 Introduction and Motivation

In a distributed collaborative healthcare environment, multiple clinical organisations from geographically different sites may be involved together in the delivery of healthcare services, each having its own users, resources, and access policies. Clinical users, residing in their own sites and doing their specific jobs, often need to access globally available resources and services under locally set constraints.

Such an environment brings challenges to distributed healthcare system infrastructures, especially when security is a concern. Security concerns, either to a conventional system or a distributed system, spread all over the system and differ from one system to another. If they are not taken into account, as early as a system begins to be built, the integrity and usability of the system may be critically compromised. Security challenges for a distributed healthcare system are notable in several aspects. Firstly, no global user repository will be available for distributed authorisation. Clinical centres may join or leave independently. The management and administration of resource access will have to be de-centralised in the network, where each site maintains their own users and resources to be accessed. Secondly, although access control

M.D. Harrison and M.-A. Sujan (Eds.): SAFECOMP 2008, LNCS 5219, pp. 167–180, 2008.
© Springer-Verlag Berlin Heidelberg 2008

becomes complicated in a distributed environment, we shall bear in mind that unless some degree of open access is promoted where hospitals and users are able to join in freely, the system will not be able to improve clinical decision making by using the knowledge they share. Thirdly, in such an open access condition, healthcare records which contain sensitive private information shall by no means be disclosed, even to collaborative centres and friendly clinicians, except for healthcare purposes. Lastly and more complicatedly, we shall consider the access constraints not only on individual cases, but also on what each of them consists of. Can doctors have access to all patient records in connected hospitals? Can a pathologist have access to complete records or even alter irrelevant reports?

Generally, securing healthcare information systems should authorise users with genuine needs to have access to the services and resource items, in order to perform their job responsibilities. Clinical requirements must be carefully studied in order to understand the constituents of job responsibilities and build the security model.

The rest of the paper is structured as follows. Section 2 analyses clinical security requirements including the principles that need to be supported by the security architecture under development. Section 3 discusses existing security approaches and describes their weaknesses in handling the requirements identified. Section 4 gives an overview of a layered security model and Section 5 illustrates this and the process of building it in detail, using the HealthAgents system. Section 6 concludes the paper.

2 Security Requirements of Healthcare Information Systems

We shall, in the beginning, draw distinctions between the types of threats imposed to healthcare systems and their likelihood. Though eavesdropping or hacking is a major concern to computer network security, it is so expensive that dedicated and capable intruders may consider using a more convenient way. Actually, 10% of GPs (general practitioner) in the UK have experienced their computers being physically stolen [8]. More likely, improper use of the system may lead to privacy leaks, by careless (or malicious) users, extra privileges given by the system incorrectly. A well-designed system should not only protect the communication sites and end users, but also carefully authorise users with genuine needs to have access to selective sharing of information without exposing additional information under protection. This security need has currently not been well addressed in healthcare information systems [4]. In this section, we outline the challenges and common security requirements of healthcare systems in a distributed environment, where preserving privacy and maintaining openness are crucial and information access decisions depend upon role and context.

2.1 The Distributed Environment of Healthcare Information Systems

Aggregating dispersed data into large databases is expensive and practically unfeasible, since geographically different healthcare centres have to have control over their datasets and at the same time maintain a globally consistent data schema. A more important reason to oppose data consolidation is concerned with healthcare data confidentiality. In the UK, the National Health Service (NHS), driven by the motives of easier central administration and better information availability, attempted to build a unified electronic patient record system and give access to extended NHS community. This has been opposed [7] [22] for the reason that such a system, collecting data from existing GP systems but out of their control, is in conflict with the ethical principle that no patient should be identifiable other than to the GP without patient consent [5] and the result from a survey that most patients are unwilling to share their information with NHS [6]. Another objection arises from the overwhelming workload such a centralised system could possibly put upon a security officer responsible for managing the data sharing [4].

A distributed healthcare service infrastructure, however, implies the capability that is required to cope with the administrative burden and the continuous maintenance needs arising from fully functional and networked clinical centres, each of which has its own users, data, access policies, and which assumes that cross-centre access is the norm. A distributed environment and its associated dynamics bring other concerns, such as patient privacy preserving, to the information-sharing healthcare network.

2.2 Preserving Privacy and Confidentiality in Shared Access

The privacy of patient information is an important issue and failure to recognise this will lead to risk of patient safety, loss of public confidence in clinical organisations, and so on [23]. A fundamental ethical principle stated by both the EU and the General Medical Council in the UK is that, patients must consent to data sharing. The British Medical Association [10] advises that clinical professionals, who have access to patient confidential information in order to perform their duties, are responsible for the information they hold under ethical or professional obligations of confidentiality and shall not use or disclose such information for any purpose other than the clinical care of the patient to whom it relates. This means patients shall be assured that they can trust the access of their information, by a care team within their treating hospitals or experts involved from collaborative centres, if any, is safe and accords with their agreement. The moving from a traditional patient-doctor relationship towards a modern patient-healthcare service relationship implies trust to clinical systems must be maintained rather than reliance on doctor responsibilities. The absence of a mechanism or policy framework in the interest of information governance and confidentiality protection, hence, may damage the healthcare services aimed to be delivered, since private information of any individual patient may be made available by systems to people not directly related with the care of that patient. This will give opportunities to potential threats, possibly coming from inside workers, as well as outside hackers. Such threats include ungraceful private information disclosure and abuse or even more risky, incorrect clinical decisions made for vulnerable patients due to clinical data being wrongly altered, accidentally or deliberately. It is worth noting that threats from outside intruding into the network are much rarer than from inside. The security risks tend to increase dramatically, therefore, when an interconnected clinical system network is in place which makes separately stored patient records and clinical information easily accessible and lets a wider range of people have access to them. Appropriate access control to patient records is the fundamental need for patient privacy and information security [23].

2.3 Maintaining an Open Access

Two aspects of openness must be maintained: 1) open for joining the system and not preventing any friendly but previously unknown clinical centre (bringing in its previously unrecognised users) from accessing information available across organisational boundaries; 2) open for information sharing to the network. Conducting healthcare research with more open use of information (identifiable data, etc.) under legitimate constraints and user acceptance, though not related with the clinical care directly, advances medical knowledge and promotes higher quality of healthcare service in the long run and is welcomed by the society. A clinical system can benefit most from clinical data as well as patient-specific data if such information can be machine-analysed and digested. The knowledge accumulated can be useful for later decision makings, particularly for rare but similar cases encountered in the future, confidential information contained in cases not being revealed.

2.4 The Different Access Needs to Data Subsets Due to Distinct Job Nature

The need of distinguishing only the relevant data for sharing among clinical professionals rather than the whole records arises from preserving privacy while maintaining open access. Even if name, address and other privacy information is removed to produce a seemingly anonymised record, a NHS clinician can easily identify a patient by the NHS number and they must be able to do so to perform their jobs. Therefore, it is sensible to grant access permission to particular record parts on the basis of users' expertise. This expertise determines their actual needs of access, to the data parts they routinely work with and by doing so, healthcare roles fulfilled. For example, pathology medical records or reports may be sent to a pathologist involved in a patient's care; prescription sent to a pharmacist; and sensitive parts not sent out at all. A specialist may have more control over their own partitions, e.g. write their reports or order certain tests, but limited permissions to other specialists' partitions or even not at all, e.g. to very sensitive medical test results.

2.5 The Access Policies and Principles Pertinent to Patients as Individuals

It is not rational to allow a professional to have access to all patient records, even if limited to the data subset fitting his/her expertise. Only relevant clinicians who have real life relationships with patients in clinical centres should access their records. This is documented in British Medical Association's security policy principles for clinical information systems [7], and the feasibility of adopting it has been evidenced in [23]. Two major principles are as follows.

Principle of Access: "Each identifiable clinical record shall be marked with an access control list naming the people or groups of people who may read it and append data to it. The system shall prevent anyone not on the access control list from accessing the record in any way."

Principle of Control: "One of the clinicians on the access control list must be marked as being responsible. Only she may alter the access control list, and she may only add other health care professionals to it."

A named responsible clinician, possibly a patient GP, as in the UK or a primary care physician (PCP), as in the US, may set up a workgroup including the specialists who together deliver healthcare to the patient. According to the Principle of Access, it is the members of this group who will be in the patient access control list, as used by

RBAC for files [16], have access to a subset of data they are responsible for, reflecting their job nature. The one who sets up the workgroup will let the system know the group members and their roles in the group, in accord with the Principle of Control. This implies a data ownership. Such a scheme decentralises management burden and increases scalability. The distributed environment and open access requirements suggest that a named doctor may involve specialists from other sites (remote consultants, temporary attending physicians, etc.) into healthcare procedures. For example, a medical opinion requested on a surgical patient may require a medical registrar, from other directorates, to exercise override access to that patient's notes [23]. This is related with delegation [4]. Essentially, a responsible doctor grants access to local or remote users from trusted sites and occasionally, someone acts on their behalf, implying ownership transfer. A triangle relationship is described in [15]: a patient is associated with a workgroup, of which a user is a member, so that a user is permitted access via the workgroup to patient ("self-claimed" or "colleague-granted"/delegation).

3 Existing Security Solutions: Role-Based Access Control and Role Mapping in a Distributed Environment

In Role-Based Access Control (RBAC) [16], permissions that describe operations upon resources are associated with roles. Users are assigned to roles to gain permissions that allow them to perform particular job functions. Privileges may be calculated as follows [2]:

*Privileges = User-Role * Role-Definition + Rules-Function (User-Attributes)*

In addition to the static collection of rights accumulated by roles, a user can dynamically achieve extra rights if they expose certain attributes as defined by rules. This model is efficient when many users require the same set of rights in an organisation but otherwise unmanageable or even useless when roles vary in different conditions under which users act. In a hospital, roles can be defined for a number of classified groups to aggregate permissions, e.g. consultant, radiologist, nurse, who have static job functions. However, dynamic contexts exist in role playing, e.g. patients may be additionally assigned to or removed from a list for which a named doctor is responsible and this influences this doctor's role in caring these patients. RBAC has difficulties to capture such security-relevant contexts as patient, location, and time in healthcare environment [4]. Patient-doctor relationship is identified as a critical clinical security constraint to record access, described in Section 2.

The Community Authorisation Service (CAS) [1] provides a solution to the management of user access control within Virtual Organisations (VOs) spanning over multiple sites in the Grid environment. It breaks the tradition of requiring each resource provider to maintain the mapping of individual users (across VOs) to its local database roles in order to authorise access to its resources. Using CAS, user memberships are instead based on VO roles and local resource providers only need to map these to local database roles. This dramatically reduces the number of mapping entries across resource providers and the duplicated maintenance burden put on them once a new user joins or a current user privilege changes.

Such an approach requires no global user repository. However, a presumption of using the approach, as it is in RBAC, is that a large number of users can be grouped into several role groups requiring certain access levels in involved organisations.

For the same reason that RBAC is infeasible to address the clinical requirement that information access or travelling may alter from patient to patient and user led as stated in the Principle of Access, the CAS is encountered with similar difficulties. Suppose clinicians A and B with the same speciality are from hospitals P and Q respectively. They will be categorised into the same VO role and the same access rights to data in P and Q. But in reality A shall have more privileges than B to certain data, e.g. of patients in P under A's care, and vice versa for B's privileges in Q.

Managing a resource access model is complex where there is a large number and various types of users, resource items, and access policies, user responsibilities being dynamic and ownership being distributed. The common practice of simply defining roles that aggregate all permissions required for the collection of resources to complete tasks is not realistic due to the diversity of individual needs which literally entails each individual a distinct role. Even the burden of defining and maintaining a proper set of access control policies based on roles for automating authorisation could be considerable. A security solution must be able to cope with the complexity.

4 Overview of a Layered Security Model

It has been pointed out that healthcare systems should be designed with multilateral security rather than multilevel security [9]. Unlike some military systems prevent information flow "down" from top secret to secret then to confidential, healthcare systems usually prevent information flow "across" from one clinician to another or from one hospital to another. This is evidenced by the requirements outlined in Section 2.4 and Section 2.5 where different access needs to cases and case partitions are distinguished due to distinct job responsibilities.

However, we argue a multilevel security model is more manageable, task availability being in the top level control and resource availability to tasks in lower level control. A multilateral security model resides in the lower level and complements the multilevel security model. The assignment of tasks to users is a business decision to be made by stakeholders, possibly explicitly in rules. It is sensible to regard the accessibility to tasks the organisational privileges with which organisation seniority is related and access to business functions restricted. Since tasks already exist in organisations and are routinely performed by specific user groups, they help to functionally decompose the system and ease security management. If a user can perform a specific type of task, then there must be certain resource items available to him/her to load into the task, if not all. Without the context of accomplishing one or more tasks in different privilege levels, information access makes no sense. The rational of using a combined multilevel and multilateral model is further supported by the fact that a job responsibility is determined by the level of authority and the division of work [14]. The former prevents information flow downwards and the latter prevents information flow across, being concerned about workgroup membership and job speciality under our further refinement. This forms a layered security architecture that addresses the healthcare security requirements.

1) Privilege of performing various types/levels of tasks and executing associated interaction models is determined by job title or grade/level. Users may upgrade their job titles occasionally and this is managed locally. Semantics of job titles and task collections must be globally defined and agreed among organisations.

2) Privilege of loading case instances for performing tasks (or enactment of interaction models) is determined by real life workgroup memberships or job boundary. This is managed by the locally named doctors, who shall be flagged as owners in case records' access control lists.
3) Privilege of accessing case record partitions (e.g. patient data, biopsy data, microarray data, MRI and MRS data, diagnosis data, therapy data, surgery data, etc.) is determined by job nature or specialist one takes on in hospitals (e.g. oncologist, pathologist, radiologist, surgeon, etc.). This is managed by system administrators when the account is setup and is maintained at a high level of stability.

Thus, a user's overall privileges will be the sum of the user's access privileges in all tasks that user is involved in (being a policy), each of which is decided by the particular cases he/she can operate as a workgroup member to deliver healthcare service (being a fact upon interaction instantiation) at the time of performing tasks, which in turn will be constrained by the accessible case partitions as determined by user professional roles (being a fact).

*User Privileges = ∑(Interaction Model Set as determined by job level * Interaction Model's Operational Cases as determined by job boundary * Case Subset as determined by job nature)*

Alternatively, the following meta-rule determines the prerequisite a user exercises privileges: a user has a title above the one required for running an interaction model can load a case, that is under the care of a workgroup which the user is a member of, and perform operations on the case parts the user's specialists allow.

user_privilege (user, im, case, part, operation) ←
job_title(user, title1) & executable(title2, im) & above(title1, title2) &
member(user, workgroup) & responsible(workgroup, patient) & own(patient, case) &
job_specialist(user, specialist) & rights(specialist, part, op)

5 Secure Interaction Models for Healthagents: A Comprehensive Case Study

In this section, we describe our HealthAgents system, the elicitation of interaction models, and their secure running in our layered security model for HealthAgents.

5.1 The HealthAgents Architecture

The HealthAgents [18] system is a distributed decision support system that facilitates diagnosis and prognosis, employs a set of distributed nodes that either store patient case data, build classifiers that are trained upon case data and capable of classifying tumour types, or use classifiers for the diagnosis and prognosis of brain tumours. The magnetic resonance spectroscopy (MRS) data used by the system is built up using anonymous information from child and adult cases. Producer nodes receive requests from clinicians and generate classifiers for particular tumours. Clinicians with cases will employ classifiers (instead of the actual cases) to assist in the diagnosis of patients for particular tumours. Knowledge extracted from cases is implicitly involved for decision making and patient privacy not compromised, private case information not being revealed in the process. The HealthAgents system consists of a variety of agents each charged with a different task. A more detailed description of the HealthAgents components and architecture can be found in [19].

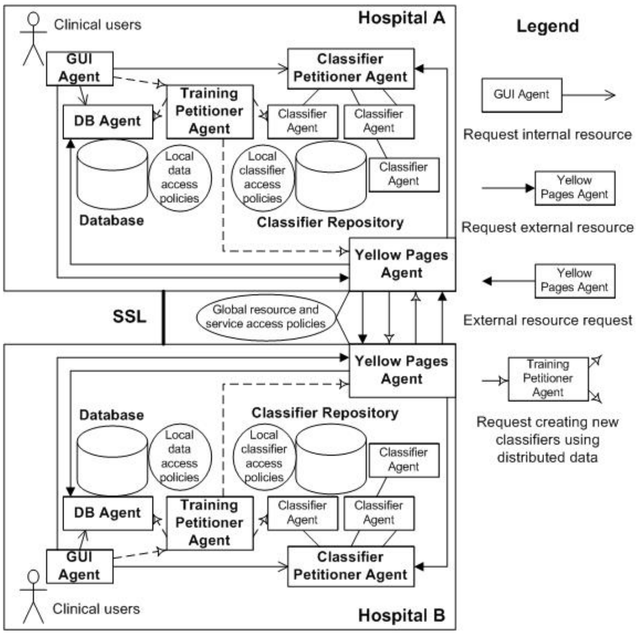
Clinical users
Hospital A
GUI Agent
Classifier Petitioner Agent
DB Agent
Training Petitioner Agent
Classifier Agent
Classifier Agent
Classifier Agent
Local data access policies
Local classifier access policies
Database
Classifier Repository
Yellow Pages Agent
SSL
Global resource and service access policies
Yellow Pages Agent
Database
Classifier Repository
Local data access policies
Local classifier access policies
Classifier Agent
DB Agent
Training Petitioner Agent
Classifier Agent
Classifier Agent
GUI Agent
Classifier Petitioner Agent
Clinical users
Hospital B
Legend
GUI Agent
Request internal resource
Yellow Pages Agent
Request external resource
Yellow Pages Agent
External resource request
Training Petitioner Agent
Request creating new classifiers using distributed data

Fig. 1. The HealthAgents system architecture and resource access flow control

5.2 Building an Interaction Model Hierarchy with a Goal-Decomposition Graph

Four major interaction models, as shown in Figure 2, are identified: create classifier, execute existing classifier, update classifier reputation value, and update case profile. They are elaborated as four sub-goals under the root goal of "tumour type diagnosis" via a goal decomposition graph, useful for requirements analysis and interaction model identification. A detailed goal decomposition procedure and underpinning process

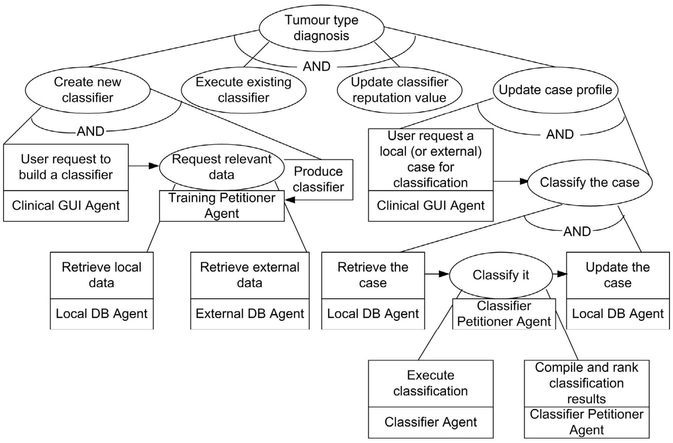

Fig. 2. The goal-decomposition graph for HealthAgents

Table 1. A high level view of selected interaction models

Goal	Sub-goals (Interaction Model)	Interaction Model privileges	Interaction Model participants	Interaction Model constraints
Tumour type diagnosis	Update case profile, etc.	N/A	All	N/A
Update case profile	Classify case	Principal clinicians or above	GUI Agent, DB Agent, Classifier Agent, and Classifier Petitioner Agent	The clinician can update the specialised data areas
Classify case	N/A	Junior clinicians or above	Classifier Agent, and Classifier Petitioner Agent	The clinician must be a workgroup member taking care of the case

elicitation can be found in [24]. Table 1 describes a specific branch of the graph ("Update case profile") for further discussion.

5.3 Secure Interaction Models and Lightweight Coordination Calculus (LCC)

Assume three job titles, senior clinician, principal clinician, and junior clinician, in that order, forms the existing clinical hierarchy, from top to bottom. Roles in a role hierarchy of RBAC have inheritance relationships. Likewise, a job title higher up in the hierarchy inherits task execution privileges from a job title further down in the hierarchy. Suppose the following rules in HealthAgents restrict task availability.

- Rule1: Senior clinicians can identify the need of new classifiers in the network and so are able to create classifiers, using all public cases and local private cases.
- Rule2: Principal clinicians have primary healthcare responsibilities and so are able to run classifiers, update case profiles and diagnosis results, as well as update classifier reputation values.
- Rule3: Junior clinicians assist in healthcare and can run classifiers and be advised of classification results.

Gaia [3] unifies responsibilities and permissions in a single role notion. It is also recognised in [21] that the coordination among agents/roles and resources must enable authorisation policy specification over interaction specification to achieve an expressive and safe interaction model. Thus, role, interaction, and constraint should be correlated. The descriptive interaction behaviour which consists of message passing and constraint solving have been defined in Lightweight Coordination Calculus (LCC) [12] that can be transmitted, interpreted, and executed by agents in the network. The LCC language has been developed in the OpenKnowledge project [13] and it uses logic expression to regulate the message exchange protocols among participant peers each of which plays a particular role. The LCC language combines role functions and constraints in a single framework and this gives us the opportunity to express permission enforcement prior to responsibility fulfilment within role playing behaviour, in the context of running interaction protocols. The following LCC clauses describe the fundamental interaction pattern for resource access control.

```
a(resource_request, RRID) ::
  request(Resource, Operation, Context) ⇒ a(resource_manager, RMID)
a(resource_manager, RMID) ::
```

request(Resource, Operation, Context) ⇐ a(resource_request, RRID) ← grantPermission(RRID, Resource, Operation, Context, Policies) then (
response(Grant_yes) ⇒ a(resource_request, RRID) or
response(Resource_result) ⇒ a(resource_request, RRID) ← getOperationResult(Resource, Operation, Access_result))

Briefly, a(resource_request, RRID) :: $\mathrm{Def}_{\mathrm{RRID}}$ and a(resource_manager, RMID) :: $\mathrm{Def}_{\mathrm{RMID}}$ denotes that agents RRID and RMID play the roles of resource_request and resource_manager respectively as defined in the definitions follow. $\mathrm{Def}_{\mathrm{RRID}}$ has a single and $\mathrm{Def}_{\mathrm{RMID}}$ has a composite message passing behaviour constructed using the following forms: Def_a *then* Def_b (Def_a satisfied before Def_b), Def_a *or* Def_b (either Def_a or Def_b satisfied), or Def_a *par* Def_b (both Def_a and Def_b satisfied). In the Def, $M_l \Rightarrow A_m$ denotes that a message M_l is sent to agent A_m while $M_l \Leftarrow A_m$ denotes that a message M_l is received from agent A_m. In the above role definitions, a message of resource access request is sent from the agent that plays the request role to the agent that plays the manager role. Upon receipt of this message, the resource manager agent applies appropriate security policies and responds by sending back a message either saying the request has been granted (or rejected) or by providing the actual resources (or the results of their usage) being requested. In the Def, $\leftarrow \mathrm{Cons}_n$ denotes that a constraint must be satisfied (as some running code) before the clause prior to it.

The notion *a(id, role)* defines the role a certain agent should play and its identity can be bound with executable tasks, workgroup memberships, and professional specialists at runtime. The role playing behaviour defines the common responsibilities an entitled user supposed to fulfil, being in a position with/above a given title as are in Gaia, the organisational roles in well-defined positions associated with expected behaviour. Then the memberships and professional specialists further constrain the concrete resource usage in the role's interaction model participation, being identity-specific and role-independent. This layered architecture is discussed as follows, illustrated by a principal clinician updating case profile after classification.

Level 1: Interaction Model constraints

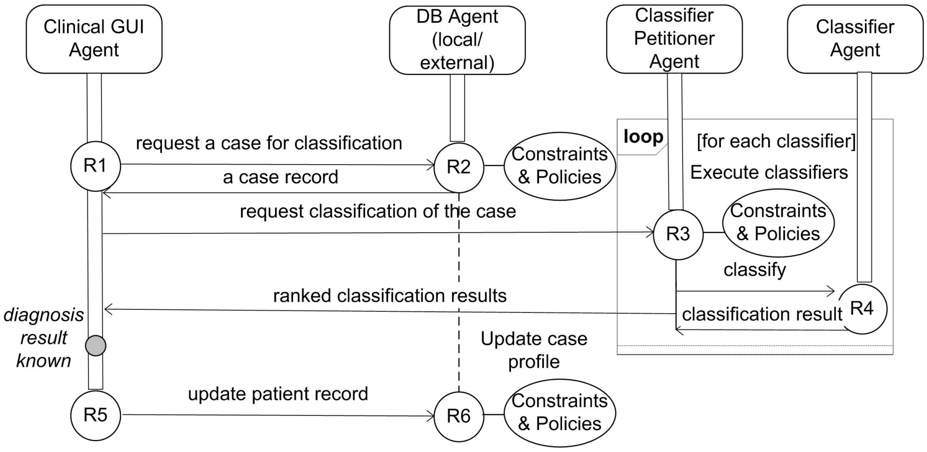

Fig. 3. Interaction Model: update case profile (including case classification)

The first layer filters interaction model availability. A principal clinician (possibly a GP) can load cases for which they have caring responsibilities and later update its profile (diagnosis result, etc.). A junior clinician can perform classification but cannot do the update. Figure 3 shows the interaction model and the following LCC clauses show its specification. The clinician plays a role of classification (R1) and updating case profile (R5). The role changes when an accurate diagnosis result is known.

```
/* R1: classify a case */
a(clinician_classify, CID) ::
  requestCaseRecordByID(I) ⇒ a(database, DBID) then
  caseRecord (R) ⇐ a(database, DBID) then
  requestClassification(R, C) ⇒ a(classifier_petitioner, CPID) then
  classificationResults(S) ⇐ a(classifier_petitioner, CPID) then
  a(clinician_followingdiagnosis, CID)
/* R5: update case record and classifier reputation following diagnosis */
a(clinician_followingdiagnosis, CID) ::
  ( updateCaseRecordByID(I) ⇒ a(database_update, DBID) then
    caseRecordUpdated(Y) ⇐ a (database_update, DBID) )
  par
  ( updateClassifier(I) ⇒ a(classifier_petitioner, CPID) then
    classifierUpdated(Y) ⇐ a (classifier_petitioner, CPID) )
```

Level 2: Case level constraints

An interaction model is uniquely defined and its running context varies, e.g. involved clinicians and cases. A resource manager must check the request (resource and operation) against the requester identity at runtime, in compliance with the access policies. Specifically, the clinician must be a member of the workgroup delivering care to the owner of the case before the case is allowed to be updated, being a meta-rule of healthcare access control. Additional local policy rule satisfaction must also be considered for extra constraints, e.g. a particular clinician can/cannot access particular resource items. A generic security policy schema for healthcare is described in [25] that can complement the meta-rule with any number of specific policies. The following shows the LCC constraints used by the database agent, being a resource manager, for permission checking before the actual role functions are carried out. The database agent issues a case record (R2) and updates the same record (R6), different levels of permissions being needed.

```
/* R2: send a case record for classification */
a(database_download, DBID) ::
  requestCaseRecordByID(I) ⇐ a(clinician_classify, CID) ← grantPermission(CID, I, Read, Nor-
mal_classify_from_local_site, Local_database_read_policy_set) then
  caseRecord(R) ⇒ a(clinician_classify, CID) ← getCaseRecordByID(I, R) then
  a(database_update, DBID)
/* R6: update a case record after classification */
a(database_update, DBID) ::
  updateCaseRecordByID(I) ⇐ a(clinician_followingdiagnosis, CID) ← grantPermission(CID, I, Update,
Normal_update_from_local_site, Local_database_update_policy_set) then
  caseRecordUpdated (Y) ⇒ a(clinician_followingdiagnosis, CID)
```

It is at the point of checking the LCC constraint of “grantPermission” that user workgroup and case will be related (clinician identity of CID and case identity of I), and other locally set read or update policies applied, prior to the required operation. A clinician not in the right workgroup may be able to download a case but cannot update it. The running and execution of LCC specification is supported by the OpenKnowledge kernel.

Level 3: Case partition constraints
Similarly with level 2, a user identity is bound with professional specialists at runtime and this will constrain further permission to case partitions, e.g. only the named clinicians may update or write major diagnosis results; certain specialists may write reports in their areas; others on the case care list may only read those areas. Thus, a three dimension resource request of (user, resource, operation) will be constrained in two dimensions: user-resource must match workgroup membership and user-operation match job specialist.

6 Conclusions and Discussion

In this paper, we have analysed the general security requirements for clinical information systems and developed a layered security model, illustrated by its application to the HealthAgents system but which is also applicable to other healthcare systems.

Organisational structure and context association are key assumptions to our privilege model. Organising authorisation at user level cannot realise cooperation and inter-organisational communication in extended health networks, as stated in [17]. The authors distinguish structural roles, describing prerequisites or competencies for actions and functional roles, being bound to the realisation of actions. Such a conjunctional perspective of role is in accordance with the privilege control in business processes and then their contextual constraint. The semantic similarity of clinical user group privileges and the business processes they can perform is described in [11]. In addition to that, access decisions need to be made on the exercise of privileges in business processes depending upon contextual information. Structuring business process (or task) context related constraints, e.g. attending relation between physician and patient as well as clinician speciality, as contextual parameters to task execution that affect access control decisions is expressed in [20].

Clinical task execution privileges, therefore, should be distinguished, and represented by the privileges of running interaction models in our approach. The layered security model authorises at a higher level, the users' task accessibility based on a static organisational structure and at a lower level, within task enactment, users' case and case partition accessibility based on dynamic functional needs in order to perform tasks. This inevitably avoids the occasion that a junior clinician creates a classifier of poor quality or updates a classifier reputation value improperly.

Next, higher level business function-based constraints are coupled with lower level data-based constraints. A limited set of data, determined by user workgroup memberships, will be allowed to be populated into the limited set of task functions. Finally, data-based constrains are additionally coupled with operation-based constraints. The available operations, determined by job nature and specialists, will be allowed, e.g. write (reports) or update (diagnosis results), upon particular data sections. These constraints, as well as individually defined local policies, must be satisfied prior to interaction model running. In sum, we constrain the availability of tasks to users, case availability to tasks, and further operations availability to cases, as the overall layered security architecture. The architecture is scalable since access rights are precisely controlled by the combination of these dimensions. For example, a senior pathologist doctor who is responsible for a patient can update the pathology part of this patient profile but someone who is a senior pathologist but not involved in caring for the patient cannot, or someone who is a junior doctor, or someone who is not specialised in pathology at all.

No global user account repository is required in our system. The necessary interaction models are globally agreed. The case to workgroup assignment is locally defined and user to workgroup possibly across organisations, for enabling interaction model running. When one user invokes an interaction model and this involves resources from other sites, the permission checking is determined by this user being involved in patient care or not, e.g. a remote clinician may perform a classification on behalf of a named doctor who is on holiday and delegates the responsibility to this clinician, in emergency situations, even the local hospital has not set up a local account for the clinician.

Interaction models can be publicly accessible since the descriptive interaction logic among peers reveals no secret information itself and so no issue exists such as alternative interaction model provision to certain users under certain conditions. Rather, alternative resource peers may be selected because the access to others is restrictive or, a subset or related/alternative resource items from query returned to the requester peer with a limited set of privileges. Such an autonomic query relaxation paradigm, as part of our future work, will avoid additional user interaction and frustrating experience. Another direction of future work is via monitoring unsuccessful resource access, an interaction model adjustment is advised if an access without satisfying constraints is encountered but considered necessary. It may be useful to let such requests be recorded and routed to responsible doctors or other delegated authorisers who may or may not approve the issuing of additional privileges, either permanently or temporarily. With better understanding of the necessity of such exceptional requests possibly after real life communication, critical and timely care aimed to patients will not be compromised.

References

1. Pereira, A.L., Muppavarapu, V., Chung, S.M.: Role-based access control for grid database services using the community authorization service. In: Transactions on Dependable and Secure Computing, vol. 3(2), pp. 156–166. IEEE, Los Alamitos (2006)
2. M-Tech Information Technology, Inc.: Beyond Roles: A Practical Approach to Enterprise User Provisioning (2006)
3. Wooldridge, M., Jennings, N.R., Kinny, D.: The Gaia methodology for agent-oriented analysis and design. Journal of Autonomous Agents and Multi-Agent Systems 3(3), 285–312 (2000)
4. Zhang, L., Ahn, G., Chu, B.: A role-based delegation framework for healthcare information systems. In: 7^{th} ACM Symposium on Access Control Models and Technologies, pp. 125–134. ACM, New York (2002)
5. Joint Computer Group of the GMSC and RCGP: GMSC and RCGP guidelines for the extraction and use of data from general practitioner computer systems by organisations external to the practice. Appendix III In: Committee on Standards of Data Extraction from General Practice Guidelines (1988)
6. Hawker, A.: Confidentiality of personal information: a patient survey. Journal of Informatics in Primary Care, 16–19 (1995)

7. Anderson, R.J.: Clinical system security: interim guidelines. British Medical Journal 312, 109–111 (1996)
8. Pitchford, R.A., Kay, S.: GP Practice computer security survey. Journal of Informatics in Primary Care, 6–12 (1995)
9. Anderson, R.J.: Patient Confidentiality - At Risk from NHS Wide Networking. Proceedings of Healthcare 96 (1996)
10. BMA - British Medical Association, `http://www.bma.org.uk/`
11. Chandramouli, R.: Business Process Driven Framework for defining an Access Control Service based on Roles and Rules. In: 23rd National Information Systems Security Conference (2000)
12. Robertson, D.: A lightweight coordination calculus for agent systems. In: Leite, J.A., Omicini, A., Torroni, P., Yolum, p. (eds.) DALT 2004. LNCS (LNAI), vol. 3476, pp. 183–197. Springer, Heidelberg (2005)
13. Robertson, D., et al.: Open Knowledge: Semantic Webs Through Peer-to-Peer Interaction. OpenKnowledge Manifesto (2006), `http://www.openk.org/`
14. Crook, R., Ince, D., Nuseibeh, B.: Modelling Access Policies Using Roles in Requirements Engineering. Information and Software Technology 45(14), 979–991 (2003)
15. Calam, D.: Information Governance - Security, Confidentiality and Patient Identifiable Information, `http://etdevents.connectingforhealth.nhs.uk/eventmanager/uploads/ig.ppt`
16. Sandhu, R.S., Coyne, E.J., Feinstein, H.L., Youman, C.E.: Role-Based Access Control Models. Computer 29(2), 38–47 (1996)
17. Blobel, B.: Authorisation and access control for electronic health record systems. International Journal of Medical Informatics 73(3), 251–257 (2004)
18. HealthAgents, `http://www.healthagents.net/`
19. Xiao, L., Lewis, P., Gibb, A.: Developing a Security Protocol for a Distributed Decision Support System in a Healthcare Environment. In: 30th International Conference on Software Engineering, pp. 673–682. ACM, New York (2008)
20. Hu, J., Weaver, A.C.: Dynamic, Context-Aware Access Control for Distributed Healthcare Applications. In: 1st Workshop on Pervasive Security, Privacy and Trust (2004)
21. Omicini, A., Ricci, A., Viroli, M.: RBAC for organisation and security in an agent coordination infrastructure. Electronic Notes in Theoretical Computer Science 128(5), 65–85 (2005)
22. Anderson, R.: Undermining data privacy in health information. BMJ 322, 442–443 (2001)
23. Denley, I., Smith, S.W.: Privacy in clinical information systems in secondary care. BMJ 318, 1328–1331 (1999)
24. Xiao, L., Greer, D.: Adaptive Agent Model: Software Adaptivity using an Agent-oriented Model Driven Architecture. Information & Software Technology. Elsevier. In: Press (2008), `http://dx.doi.org/10.1016/j.infsof.2008.02.002`
25. Xiao, L., Peet, A., Lewis, P., Dashmapatra, S., Sáez, C., Croitoru, M., Vicente, J., Gonzalez-Velez, H., Lluchi Ariet, M.: An Adaptive Security Model for Multi-agent Systems and Application to a Clinical Trials Environment. In: 31st IEEE Annual International Computer Software and Applications Conference, pp. 261–266. IEEE, Los Alamitos (2007)