

An Adaptive Security Model for Multi-agent Systems and Application to a Clinical Trials Environment

Liang Xiao¹, Andrew Peet², Paul Lewis¹, Srinandan Dashmapatra¹, Carlos Sáez³, Madalina Croitoru¹,
Javier Vicente³, Horacio Gonzalez-Velez⁴, and Magí Lluch i Ariet⁵

¹University of Southampton, UK

²University of Birmingham, UK

³ITACA, Spain

⁴University of Edinburgh, UK

⁵MicroArt, Spain

Abstract

We present in this paper an adaptive security model for Multi-agent systems. A security meta-model has been developed in which the traditional role concept has been extended. The new concept incorporates the need of both security management as used by role-based access control (RBAC) and agent functional behaviour in agent-oriented Software Engineering (AOSE). Our approach avoids weaknesses of traditional RBAC approaches and provides a practically usable security model for Multi-agent Systems (MAS). A unified role interaction model framework has been put forward that incorporates not only functional requirements but also security constraints in MAS. A security policy rule scheme has been used to express security requirements in relation to affective roles. The major contribution of the work is that little redevelopment effort will be required when security is to be engineered into the overall MAS architecture, hence minimising the impact of the security requirements changes to the MAS architecture. We illustrate the approach through its potential application in a clinical trial setting involving a prototype medical decision support system, HealthAgents.

1. Introduction

Distributed decision making systems are becoming increasingly useful and important for involving collaborative partners in efficient service sharing amongst them. Security is a growing concern in designing such systems that organisations can trust and use. The internet infrastructure through which a distributed system openly transfers data is not, of itself, a safe environment. Well-studied data encryption algorithms and publicly available libraries based on them can alleviate this problem when incorporated into

the system messaging network. Yet more complex considerations are related with the management of the different levels of access rights to multiple types of resources by users distributed among and managed by multiple organisations. These organisations need to use resources from others and also need to prevent their own resources from unauthorised use. On one hand, if a system is over restrictive in resource access control then the system cannot be made full use of. On the other hand, if a system is not sufficiently restrictive then the organisations' private data is in danger of being exposed. These constraints entail flexible security policy management and organisations need to be able to configure policies themselves to reflect their actual (changing) needs. Some systems embed security policy modules within the application code. The tight coupling of software architecture with policies that spread all over the application, but which intend to change, makes such systems hard to maintain. An adaptive security model that is configurable and which is reusable across applications would represent a significant advance. A system is not safe if a model is developed but never managed afterwards. Policies handled by such a model need continuous maintenance to ensure the security model remains useful - security is a process, not a product [2]. Bearing this in mind, this paper extends the role concept, incorporating both role-based agent behaviour and role-based access control in a single role interaction model. The easily re-configurable model maintains not only functional requirements but also security constraints in MAS. We apply this method to the clinical trials domain.

2. HealthAgents Overview

The HealthAgents project [4], a Specific Targeted Research or Innovation Project (STREP) plans to create a multi-agent distributed Decision Support System (d-

DSS) based on novel medical imaging and laboratory tests to help determine the diagnosis and prognosis of brain tumours. Brain tumours are an important cause of morbidity and mortality [3] and there is a need to improve their classification, and management. Novel medical imaging techniques such as magnetic resonance spectroscopy (MRS) and laboratory techniques such as gene expression arrays promise to deliver these advances but suffer from a complexity of interpretation which has hindered their incorporation into routine clinical practice. These new techniques provide an excellent test bed for the development of a computer aided decision support system. Furthermore the rarity of many brain tumour types requires that information must be sought from many hospitals. The use of a distributed system for data collection and management is therefore a necessity.

Prior to incorporation into clinical practice new methods must be fully tested within a clinical trials setting. Such trials are subject not only to data protection laws but also regulations governing clinical trials including ethical approval and informed consent of the participants. For multinational projects, ethical approval is devolved to regional bodies without any coordinated or uniform decision making and so data gathered from different centres may be subject to different restrictions. Allowing for flexibility within the data security model is therefore essential.

Clinical trials commonly use data from which personal information (e.g. name, address, date of birth) is removed but to which a unique patient identifier is added, often termed link-anonymised data. Such a scheme has the advantage of having a high chance of preserving patient anonymity whilst allowing data from the same patient to be added at a later date. This scheme also allows a specific patient's data to be located and removed from the project at any time they request, a condition usually imposed by ethics committees. Full patient records are kept for clinical purposes within the treating hospital and with the patient's permission may be used to generate and periodically update the clinical trials data.

Clinical trials are usually supported by a centralised database where the link-anonymised data is stored. This allows the patients to be reassured that their data will be afforded a high level of security and allows regulatory bodies ease of access to inspect the processes in place. For a distributed system, similarly robust arrangements must be designed to reassure ethics committees and patients that the data is secure. However, achieving this is a significant challenge and here we discuss a potential model for achieving this together with the necessary technical requirements and their proposed solutions. Each data collecting centre could have an associated link-anonymised database as approved by their appropriate ethics committee. Patient identifiers could

then be kept along with the clinical patient record in the treating hospital. These databases need be the only databases kept within the system giving a truly distributed data-warehouse. The limited data required for analysis could then be subject to stringent anonymisation processes and sent to a small number of specific sites for processing, for example the production of classifiers. In this way, the distributed nature of the system could be preserved whilst allowing appropriate regulatory access to data repositories. Security systems will need to be in place which can allow each centre to potentially limit the type of data transmitted and the locations it is transmitted to.

3. Existing Approaches and Related Work

Access control is central to the security of software systems, including authentication, authorisation, audit, as well as measures such as digital signatures and encryption. Authentication determines who can log on to a system and use it, authorization determines what a user can do, and audit/accountability identifies what a user did. Two earlier access control models are discretionary access control (DAC) and mandatory access control (MAC). DAC is an access policy determined by the owner of an object. MAC is an access policy determined by the system, not the owner. An access control list (ACL), a list of permissions attached to an object, can be used by both models and applied in operating systems such as Windows.

A newer access control model that supports efficient management is the widely accepted US National Institute of Standards and Technology model of role-based access control (RBAC) [1]. In RBAC as illustrated in Figure 1, roles represent job functions in an organisation. They bring together users and permissions. 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. For example, a clinician role can be created in a hospital and permission giving access to patient data can be associated with this role. When a new clinician joins the hospital, he/she can be assigned the clinician role and so have the permission to access patient data. A major benefit of using this type of model is that the reconfiguration of user-role, role-permission, and role-role relationships, directed by administrators, can reflect changing organisational policies. The maintenance of such a sub-system that is independent from the core application minimises the impact on the overall system of requirements changes with regard to security. RBAC is widely accepted as a best practice and implemented in one form or another in systems including Microsoft Active Directory, SELinux, FreeBSD, Solaris, and

Oracle DBMS. However, several weaknesses have been identified.

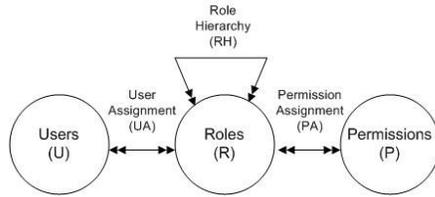


Figure 1. The RBAC model

In a hospital, different users with the same clinician role may have different permissions to particular resources. For example, one clinician that created a patient case in a hospital might have more rights than other clinicians in the same hospital. Clinicians in one hospital could have more rights to data in that hospital than clinicians from another hospital. Since permissions are not directly assignable to individual users, it is impossible to use RBAC to differentiate users with practically different capabilities in the system. Another insufficiency in the RBAC model is the lack of access context modelling. Access context can constrain specific conditions that must be met before the access. In the above example of clinicians accessing patient data, access permission is different depending on the different context (a clinician created the patient case or not). Finally, no explicit concept of organisation and negative permission makes it inconvenient to grant permissions to a group of users except particular individuals from the group.

The DAFMAT approach [5] is based on the RBAC model and applied to healthcare applications. Concepts of user, role, subject and domain are used and their mappings in pairs are defined to formulate access modes. Authorisation requests are validated using the access modes. However, their subjects represent executable domain functions and other resource types, such as data resources, are not protected. Moreover, the presentation of this model is only for human comprehension. A mechanism of forming security policies in an executable manner has not been considered.

The importance of security in the healthcare domain has also been recognised in [6], particularly for managing patient data and its communication in a distributed environment. Security tags are used to mark information with regard to privacy within the patient record structure so that access is restricted to trusted agents only. This approach is limited to secure patient data access and has paid no attention to the many security issues involved in the healthcare service provision process.

4. An Adaptive Security Model for Multi-Agent Systems

In this section we extend RBAC to avoid its weakness and to meet the unique characteristics of MAS.

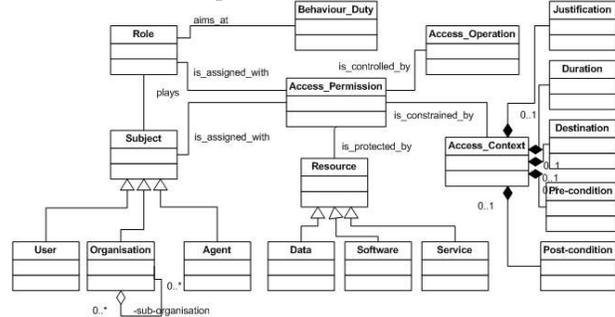


Figure 2. New security meta-model

The fundamental access permission policies take the following form:

{Subject (Id, Role, Organisation), Access Operation (Op), Access Context (Co), Resource (Id, Type)}

Policy rules externalise security requirements and are structured in this form for later continuous management. The meta-model has been motivated by the particular requirements of the HealthAgents project but it is generic so that other domains and applications may use it. The five key points below explain the major features of the security model illustrated in Figure 2.

1) Interacting with other agents is regarded as an additional type of resource (system service) to that of data and software (e.g. a classifier in HealthAgents).

Generally, a user agent requests a system resource from a resource provider agent through some intermediate agents. In the interaction process, one may enforce security policies just after the user agent delivers its request or before the resource is to be approved for use. However, we conceive all intermediate agents provide services in one form or another, so security constraints should be imposed in each agent interaction rather than in a single place as many other approaches do. Suppose an interaction is used by a house hunting user agent. It uses an estate agent finder agent and an estate agent to look for information of properties that match its preferences. An agent-finder service and a property preference matching service have been provided before the final property information is returned. Different levels of services might be provided in the process according to the user's service subscription and credit information. The permission of the contact of an estate agent does not necessarily mean it will provide all its information.

2) User agent and system agent are the two types of subjects, the former needs permissions to access resources and the latter provides services in MAS.

System agents have tightly coupled responsibilities decided by designers. Human users have loosely coupled permissions decided by system policies. In the HealthAgents d-DSS, a user who logs on to the system

will be associated with an agent with ID and roles. Permissions are gained to agents through those directly associated (via the subject ID), roles they are assigned to (via subject-role relationships), or organisations they belong to (via clinical organisation membership). Role definitions and user-role assignment are managed locally in individual hospitals. An administrative role can be assigned to a HealthAgents project manager to manage users and roles globally. On an individual basis, a clinician may have full access rights to his/her patient while other clinicians may not. A clinician role hierarchy may also be defined (manager, principle clinician, senior clinician, junior clinician, apprentice, etc.) so that some clinicians have more access to operations (who e.g. can add new cases to the system) than others (who e.g. can only run classifier) according to their experience.

3) RBAC is extended with permissions assignable to individuals as well as organisations.

It might be necessary for example to define that senior clinicians can access all instances of a particular type of resource, the classifiers. More likely, individual entities of a resource type are specified to be accessible by individual subjects. Permissions can be assigned upon a set (or type) of resources or for a group of subjects with exceptions. This can be configured by a positive permission policy for the whole collection and a negative permission for individual exceptions.

4) RBAC is extended with context to provide additional flexibility.

Access context might include descriptive justification of the access operation, where/when the requested data goes, the duration of the use of the data, the pre-condition & post-condition of the access operation. Agents play roles during their interaction (see point 5), context varies and agents behave differently while evaluating certain instance values populated at runtime. A clinician may have special control over data of a patient under the pre-condition (a type of context) that he/she is the principal doctor of the patient and this special identification must be checked against before a special operation is carried out. Context can also be used to enable access normally not seen through rights delegation, for example, when two hospitals (or clinicians) reach some agreements. A hospital can then delegate the use of its private classifiers to another hospital or delegate the access of its patient data to some particular external clinicians or bodies for classification,

given the appropriate ethical and patient permission has been obtained. Context specification is also useful to allow special access for appointed individuals, even being outside the HealthAgents network and having no user account or role assignment. By supplying a justification of how the required data will be used and the destination of the data transmission, the access may be granted if such information is approved under appropriate contracts and with specified permissions.

5) A uniform role interaction model integrates the concept in agent paradigm and that in RBAC.

A role plays its behaviour duty if and only if its permission constraint is satisfied.

Role is an important concept in Agent-oriented Software Engineering (AOSE) and tightly associated with agent behaviour. However, the role concept in the AOSE research community and that in the Role-Based Access Control community are completely distinct and no research has ever been carried out to reconcile the two definitions of the concept for security control in MAS. In our security model, agent behaviour is specified in roles which not only realise functional requirements but also enforce security policy requirements. RBAC has no concept of duty and AOSE has no permission constraint for agents. The complementary nature directs straightforward integration of them in a role interaction model. Figure 3 shows a prototype interaction model for invoking a classification in the HealthAgents system. Table 1 provides an analysis of the security requirements for this prototype model. The application of the security model is discussed afterwards, sample security policies being used during the interaction.

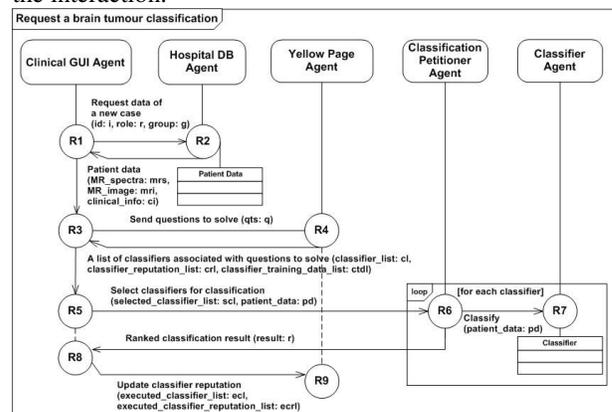


Figure 3. A possible role-based interaction model in HealthAgents

Table 1. Role behaviour duties and permissive constraints in interaction model

Interaction roles	Interaction description	Security requirements analysis & Resource access mode
R1 and R2	A clinician requests patient data for classification. The case might have been inserted by the same clinician or a different clinician previously.	This clinician must be authorised to have the access rights to the patient data before the data is returned. Direct system resource access: patient data

R3 and R4	The clinician sends questions (tumour or non-tumour, aggressive or non-aggressive, glioblastoma or metastasis, etc.) to solve. Classifiers that can discriminate among tumour classes (so answer various questions) have been previously registered on yellow pages, including description about their capabilities, reputation, and data they have been trained upon.	Queries about classifiers are answered only if the clinician has the access rights to the required classifiers in the classifier directory. Intermediate service access: yellow page directory service (query)
R5 and R6	The clinician requests a selection of classifiers to solve questions by supplying patient data that registered classifiers can operate upon.	The clinician must be authorised to perform the classification operation upon the current case. The passing of patient data must respect legal and ethical constraints regarding patient privacy. Sometimes only part of the data of a particular patient can be supplied externally (image, spectra, etc.). The communication parties might also maintain distinct security policies. Intermediate service access: petitioner broker service
R6 and R7	Classifiers will attempt to classify the patient data supplied to them. Only those classifiers which have the supplied information (MR image, MR signal, or clinical information such as age, sex, tumour location) that completely meets their input requirements will be executed eventually.	Classifiers located in one hospital that have been trained using data across many hospitals are not usable to all clinicians. Private classifiers are only usable to local clinicians. Appropriate contracts must be in place to allow clinicians to use external classifiers if they are not public ones. Direct system resource access: classifier
R6 and R8	Classification result is collected from the multiple classifiers. The statistics information from other clinicians or previous performance is used to rank the answers before they are returned to the clinician.	The clinician must be authorised to access the global statistics information before the classification answers are ranked. Intermediate service access: classification result compilation and ranking service
R8 and R9	The clinician evaluates the classification result produced by the selected classifiers when the diagnosis is finished. The reputation of these classifiers is updated on yellow pages so that other clinicians will have better knowledge about how good each of these classifiers is in later use.	The clinician must be authorised to update classifier reputation. Intermediate service access: yellow page directory service (updating)

In the above scenario, information of the operating clinician is passed on through the whole interaction process to gain data and service access (different agents maintain different policies for clinicians). Due to space limitations, only two sample security policies that must be enforced between the interaction of R1 and R2 are given in Figure 4.

Normally, clinicians can be approved to have access control and diagnose permission to patients in their hospitals. This, however, may not necessarily constrain the system from perhaps flexibly assigning an external clinician with sufficient expertise and competence to diagnose a particular patient in an emergency situation (when his/her original clinician is away) under appropriate contracts. Justification and duration context is used. This second policy allows an external clinician to play his/her behaviour duty in R1 and R2 & R5 and R6 anytime between the specified duration if a proper contract allows external classification behaviour.

Roles that capture the function of agents during their interaction have been standardised in XML with a scheme of {event, processing, (condition, action)_n, belief} by our Adaptive Agent Model (AAM) approach [9][10]. The structure describes, on receiving event messages from other agents, how actions are taken under various conditions after message processing and decision making. Agents execute annotated roles dynamically. Non-security policy rules are evaluated and applied during agent interaction by using a Fact Manager Agent and a Policy Rule Manager Agent. Externally specified interaction roles and policy rules capture functional requirements and keep maintained via a set of configuration tools [11]. The security policy rules as specified in Figure 4 will be later integrated with the existing AAM framework so that the existing role/rule repository will incorporate those with

regard to security policies. Agents in the system will assume their role dynamically and will only perform the specified functions when they find, at runtime, all security constraints affective to their roles are met.

```

<Security_Policies>
  <Policy id="p_001">
    <Affection>
      <Role>clinician</Role>
    </Affection>
    <Permission description="A clinician can read and insert patient data only to the hospital he/she belongs to">
      <Subject id="clinician_10">
        <Role>clinician</Role>
        <Organisation>H1</Organisation>
      </Subject>
      <Access_Operations>
        <Access_Operation>read</Access_Operation>
        <Access_Operation>insert</Access_Operation>
      </Access_Operations>
      <Resource>
        <Type>patient_data</Type>
        <Location>hospital_H1</Location>
      </Resource>
    </Permission>
  </Policy>

  <Policy id="p_002">
    <Affection>
      <Role>clinician</Role>
    </Affection>
    <Permission description="p_001 is relaxed under certain context">
      <Subject id="clinician_10">
        <Role>clinician</Role>
        <Organisation>H1</Organisation>
      </Subject>
      <Access_Operations>
        <Access_Operation>read</Access_Operation>
        <Access_Operation>classify</Access_Operation>
      </Access_Operations>
      <Access_Context>
        <Justification>The doctor who created patient_00001 is away and clinician_10 has a contract contract_01 that allows the clinician to read and classify the patient data</Justification>
        <Duration>
          <Start_Time>time1</Start_Time>
          <End_Time>time2</End_Time>
        </Duration>
      </Access_Context>
      <Resource id="patient_00001">
        <Type>patient_data</Type>
        <Location>hospital_H2</Location>
      </Resource>
    </Permission>
  </Policy>
</Security_Policies>

```

Figure 4. Sample annotated security policy rules

In the implementation level, security policy rules can be externally linked to the existing AAM interactive role structure within which a new global condition element will be added to the scheme. The relevant rules from the linked rule set must be evaluated as satisfactory before agents can select specific actions to perform from the branches specified locally in (condition, action) couplets for their usual interaction. The integration of the security model with the AAM leads to an adaptive and secure MAS that can dynamically interpret its behaviour from integrated requirements and design models, always being under proper maintenance. This provides a practical usable scheme for Model Driven Architecture (MDA) [12] within the agent paradigm. Capturing not only behaviour duty but also permission constraints, the role interaction model is distinct from other security modelling approaches towards MDA such as SecureUML [13], which generates an architecture that is only related to access control.

5. Conclusions

In this paper we have presented an adaptive security model for MAS and shown its potential application to a clinical trial developing a prototype tumour classification system, HealthAgents. A unified role interaction model framework has been developed that incorporates not only functional requirements but also security constraints using the extended role concept. The weaknesses of traditional role-based access control approaches have been avoided in the model we designed. The major advantage of using this approach is that, although the requirements of access control could change from time to time, the system can be dynamically adapted with up-to-date policies applied simply by the configuration of independent policies.

Eventually the idea would be to document all HealthAgents security policies in a rule repository and develop extended policy configuration tools based on the model structure. Moreover, we will improve the security model in the following directions: adding detection and reaction mechanisms for ongoing security processes; introducing extra protection to yellow pages from unauthorised read or change (so no agent can pretend to provide services but with malicious purposes). We will also add semantics-aware support to our security model, which is necessary for role-mapping and policy-mapping in an inter-operable multi-domain environment [7]. Hospitals may sign contracts among themselves and delegate roles to external users and apply their individual policy rules, given that these meet the required ethical requirements. Roles and policies, possibly produced with different schemes, role relationships and policy vocabularies need to share common understanding before integration and inter-operation in the distributed

environment. Ontology mapping is a useful technology which is already used in HealthAgents. Its use for solving semantic difference among security policies across clinical domains will be further investigated in our future work. We will examine issues arising from the added semantics exploring whether data or knowledge of the system can be inferred via knowledge-level conversation carried out among semantics-aware agents, or through ontology translation [8].

Acknowledgements

This work is supported under the OpenKnowledge and HealthAgents STREP projects funded by EU Framework 6 under Grants: IST-FP6-027253 and IST-FP6-027214.

References

- [1] Sandhu, R.S., Coyne, E.J., Feinstein, H.L. & Youman, C.E., "Role-Based Access Control Models", *Computer* 29(2): 38-47, IEEE Computer Society Press, 1996.
- [2] Schneier, B., *Secrets and Lies: Digital Security in a Networked World*, John Wiley & Sons, 2004.
- [3] Bray, F., Sankila, R., Ferlay, J. & Parkin, D.M., "Estimates of cancer incidence and mortality in Europe in 1995", *European Journal of Cancer*, 38(1):99-166, 2002.
- [4] HealthAgents: <http://www.healthagents.net/>.
- [5] Chandramouli, R., "A framework for multiple authorization types in a healthcare application system", *Proceedings of the 17th Annual Computer Security Applications Conference*, pp.137- 148, 2001.
- [6] Wimalasiri, J.S., Ray, P. & Wilson, C.S., "Maintaining security in an ontology driven multi-agent system for electronic health records", *Proceedings of the 6th International Workshop on Enterprise Networking and Computing in Healthcare Industry*, pp. 19-24, 2004.
- [7] Joshi, J.B.D., Bhatti, R., Bertino, E. & Ghafoor, A., "Access-Control Language for Multidomain Environments", *IEEE Internet Computing*, 8(6):40-50, 2004.
- [8] Farkas, C. & Huhns, M.N., "Making Agents Secure on the Semantic Web", *IEEE Internet Computing* 6(6):76-79, 2002.
- [9] Xiao, L. & Greer, D., "The Agent-Rule-Class Framework for Multi-Agent Systems", *International Journal of Multiagent and Grid Systems*, 2(4):325-351, IOS Press, 2006.
- [10] Xiao L. & Greer, D., "Externalisation and Adaptation of Multi-Agent System Behaviour", *Advanced Topics in Database Research*, Volume 5, pp 155-177, Idea Group, 2006.
- [11] Xiao, L. & Greer, D., "The Adaptive Agent Model: Software Adaptivity through Dynamic Agents and XML-based Business Rules", *Proceedings of the 17th International Conference on Software Engineering and Knowledge Engineering (SEKE'05)*, pp. 62-67, 2005.
- [12] Object Management Group, Inc., 250 First Ave. Suite 100, Needham, MA 02494, USA.
- [13] Lodderstedt, T., Basin, D. & Doser, J., "SecureUML: A UML-Based Modeling Language for Model-Driven Security", *Proceedings of the 5th international Conference on the Unified Modeling Language*, LNCS 2460, Springer, pp. 426-441, 2002.