Security Ontology for Mobile Agents Protection

S. Hacini and R. Lekhchine

Abstract—**Security is a very active field, were too much terminology is vaguely defined. This leads to difficulties for applications to evaluate their security level in order to communicate in safety manner. This problem increases when these applications use mobile agents. Indeed, mobile agents have to estimate the trust of environment where they will be executed.**

This issue is addressed, in this paper, by the use of security ontology. The development of this ontology must follow a process which consists on a set of phases in order to leads to a typical ontology.

Index Terms—**Mobile agent security, security ontology, security policy, Web Services security.**

I. Introduction

Mobile agents are becoming the paradigm of development of distributed and open applications like electronic commerce. The use of mobile agent paradigm provides several advantages to design and control distributed applications such as autonomy or dynamic adaptation. Unfortunately, it has introduced some problems. Security represents an important issue.

For some applications security is of critical importance, not only in terms of functionality, but also in terms of a trust environment with increased security and privacy features required for user confidence, as is the case of e-commerce applications. In the other hand, explicit differences in security policies can exist among organizations that transact over the Semantic Web [1].

Although there have been several approaches for generating security requirements specification and refinement, so far no approaches can provide unambiguous semantics and execution-environment-awareness simultaneously. This problem is increased when it is about mobile agents' communication.

Indeed, one crucial issue is the dynamic nature of many transactions, where agents (service requesters and service providers) interact without any prior direct trust relationship. In these situations, trust relationships must be established on the fly and for a limited purpose and time.

Ontology is a specification of a conceptualization [2]. It represents knowledge in a formal and structured form as well as provides a better communication, reusability and organization of knowledge and a better computational inference [3]. In this way, the main objective of ontologies is that of establishing ontological agreements not only to decrease language ambiguity but also to serve as a basis for secure communication between agents.

In this paper, the issue of accommodating security requirements in the communication is addressed by the use of security ontology. Thus, the goal of this paper consists of construction of a Mobile Agent Security Ontology in order to eliminate the semantic differences which exist in objects, attributes, and data of security policies.

This paper is organized as follows: the second section is dedicated to the presentation of the state of the art concerning existing security ontologies. Section3 and Section4 relate to detail our contribution. A conclusion achieves this paper.

II. Security Ontologies

Security ontologies are an important topic due to the increasing importance of security in Information Systems and the need of a common language for the Information Systems security area. We quote here some of the existing security ontologies.

To facilitate trust relationships establishment, Denker, et al. [4] proposed two security ontologies: Credential ontology, which summarizes various ways in which authentication using credentials take place, and Security ontology, which summarizes many of the commonly used security-related notations that can be used to describe user, agent or security service policies. This information can then be used during matchmaking processes to ensure that customers and service providers' security requirements meet each other.

In further work Kagal et al. [5] added security and privacy policies to the above mentioned proposal. They claim that policies should be part of the representation of the Web service, because they provide the specification of who can use a service under which conditions, how information should be provided to the service and how provided information will be used later.

Ashri, et al. [6] proposed a Semantic Firewall to reason about where the interacting entities are able to support the required security policies.

All the quoted ontologies tried to mask the heterogeneity existing among security policies.

III. Presentation of MASO

The construction of the ontology MASO, dedicated to the mobile agents security domain, is supported by METHONTOLOGY methodology [7]. The description logic formalism is adopted to express semi-formal ontology. OWL is selected as language for the ontology coding by using Protégé-OWL editor. Finally, the inference engine RACER (Renamed Abox and Concept Reasoner Expression), is

Manuscript received April, 15, 2012; revise May 15, 2012.

S. Hacini is with the Computer sciences department, Mentouri University of Constantine. She is member of LIRE Laboratory, Constantine, Algeria (e-mail: salimahacini@gmail.com).

R. Lekhchine, is with the Computer sciences department, Mentouri University, Constantine, Algeria (e-mail: lekhchineriad@gmail.com).

employed to test the consistency of the ontology throughout its construction process.

A. Security Context

Throughout its migration, the agent needs to interact with various environments. The environment, which is potentially heterogeneous and unpredictable, can influence the execution process. Consequently, mobile agent must be sensitive to its security context in order to check if the security requirements and the acquired capacities for its execution are satisfied.

With the ontology MASO, the mobile agent can reason on the security context of the entity in interaction. Among the various parameters of security, we quote security requirements, security threats and security weakness.

B. Communication between Mobile Agents and Hosts

During the interaction, MASO can mask the possible security heterogeneity, and provide an explicit semantics so that the security context becomes understandable.

The module of communication allows the agents to be able to question the ontology and to receive the results while coming. Jena Platform offers a certain number of OWL APIs being used to exploit the ontology.

IV. MASO Construction Process

The construction ontology process starts with vague knowledge and terminates with functional application ontology represented by OWL language. The great stages of this process are inspired by "METHONTOLOGY" methodology [7]. This process is based on HEMMAM work [8]. It is composed of five stages detailed in the following sub-sections.

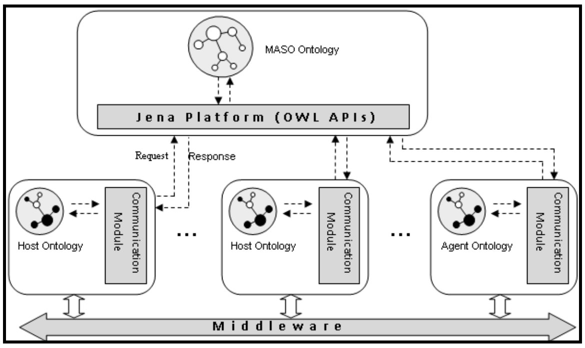

Fig. 1. Communication using ontology MASO in a dynamic environment

A. Needs Specifications

The specification stage consists in drawing up a document of needs specifications. Within this document, the ontology is described through five aspects: The knowledge domain (mobile agents security), the goal (to conceal the heterogeneity concerning security among the agents and the hosts in order to guarantee a better interworking), the users (mobile agents and hosts), sources of information (technical documents of the mobile agents security) [9], the technology of Web services, and ontologies for the computer security, the effect of the ontology (agent, host, algorithm, protocol, countermeasure, threat, etc). This stage is summarized in an RDF document.

B. Conceptualization

Once this knowledge is acquired, it must be organized and structured using semi-formal intermediate representations. This phase concerns the construction of a set of documents like concepts classification diagram, binary relations diagram, table of the logical axioms or Table of the instances.

The glossary contains the definition of all the terms relating to the field (concepts, instances, attributes, relations) which will be represented in final ontology. As an example, the terms Agent and Algorithm are concepts, fulfillObjective and useProtocol are relations, etc.

The concepts classification diagram shows the organization of the concepts of the ontology in a hierarchical order. It expresses the subclass relations (Cf. Fig.2).

C. Formalization

In this stage, we use the description logic formalism in order to formalize the conceptual model obtained during the conceptualization step.

We build the TBox and define concepts and roles using the constructors provided by description logics. Moreover, we build TBox by the specification of the relations of subsumption which exist between different concepts/roles. To build ABox, we describe facts using the assertional language:

A(C): To specify that A is an instance of the class C.

R(A1, A2) : To specify that the two individuals A1 and A2 are related by relation R.

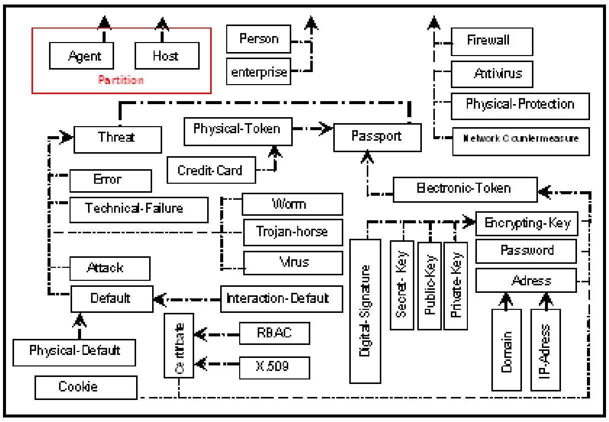

Fig. 2. Concepts classification diagram

D. Implementation and Test

OWL which represents a coding language is used to implement the ontology MASO for all its semantic functionalities that are more interesting than those of languages RDFS and DAML+OIL.

PROTEGE OWL is a modular interface. It enabled us to build the hierarchy of concepts, the classes, the properties, the attributes and the relations (name, type, and domain).

The Racer system was used to test MASO. During the tests applied to MASO ontology, no error has occurred.

V. Conclusion

We developed ontology of security domain while following the stages of the selected process which is inspired by METHONTOLOGY. We started with the specification stage. Then, we organized and structured the knowledge obtained by using semi intermediate formal representations independent of any coding language. Afterwards, we used the formalism of description logics to represent the application ontology in a formal language and finally we implemented and tested the ontology by using Prot ég é-OWL editor and RACER Reasoner.

The ontology of security domain was built, we must follow its evolution by adding the new concepts in its terminological part (TBOX). The result of this stage will be a new ontology with a new hierarchy of concepts. For that, we propose the use of the classification-based reasoning which is one of the basic mechanisms for description logics.

REFERENCES

[1] K. J. Lee, S. J. Upadhyaya, H. R. Rao, and R. Sharman, "Secure knowledge management and the semantic web," *Communications of the ACM*, vol. 48, no. 12, pp.48–54, 2005.

[2] T. Gruber, "Towards Principles for the Design of Ontologies used for Knowledge Sharing," *International Journal of Human-Computer Studies*, vol. 43, no. 5/6, pp. 907-928, 1995.

[3] M. Gruninger and J. Lee, "Ontology Applications and Design," *Communications of the ACM*, vol. 45, no. 2, pp.39-41, 2002.

[4] G. Denker, L. Kagal, and T. Finin, "Security in the Semantic Web Using Owl," *Information Security Technical Report*, vol. 10, no. 1, pp. 51-58, 2005.

[5] L. Kagal, M. Paolucci, N. Srinivasan, G. Denker, T. Finin, and K. Sycara, "Authorization and Privacy for Semantic Web Services," *IEEE Intelligent Systems* vol. 19, no. 4, pp. 50-56, 2004.

[6] R. Ashri, T. Payne, D. Marvin, M. Surridge, and S. Taylor, "Towards a Semantic Web Security Infrastructure," In Semantic Web Services 2004 Spring Symposium Series, Stanford University, Stanford California, 2004.

[7] M. Fernandez, A. Gomez-Perez, and N. Juristo, "METHONTOLOGY: from ontological art toward ontological engineering," *Spring Symposium Series on Ontological Engineering*. AAAI97, USA, 1996.

[8] M. Hemam and Z. Boufaida, "An Ontology Development Process for the Semantic Web," EKAW'04 Workshop on Knowledge Management and the Semantic Web, Whittlebury Hall, Northamptonshire, UK. 2004

[9] S. Hacini, "Sécurité des Systèmes d'Information : Mise en Œuvre de la confiance et de l'adaptabilité pour la protection de l'agent mobile," Ph.D. dissertation, LIRE Laboratory, Mentouri university, Constantine, 2008.