7

TRUST BUILDING IN THE CREATION OF VIRTUAL ENTERPRISES IN MOBILE AGENT-BASED ARCHITECTURES

R. J. Rabelo; M. S. Wangham+; R. Schmidt; J. S. Fraga
Federal University of Santa Catarina – Department of Automation and Systems
PO Box 476-CEP 88040-900- Florianópolis(SC) - BRAZIL
{rabelo, wangham, rschmidt, fraga}@das.ufsc.br

This paper presents an approach for a partners search and selection system to support the creation of Virtual Enterprises from a cluster of existing companies. This process is carried out in an agile, smart *and* secure *manner, three basic enabling elements to enhance trust building inside the cluster. This approach exploits the use of mobile agents in an open and large-scale system, as well as introduces some security mechanisms in the agents systems.*

1. INTRODUCTION

Techmoldes[1] is a cluster of mould makers SMEs placed in the South of Brazil whose members have been collaborating to enhance their global competitiveness. It was created as a strategic solution to face the market needs that have demanded shorter delivery times and lower prices. The main idea of Techmoldes is to act as a "single" / larger productive entity in the market, combining the individual skills and resources of each member, but "transparent" to the final customer. On the other hand, each member remains independent and autonomous, even to make business out of the cluster. Techmoldes is currently composed of thirteen companies, including competing companies with equivalent resources. It is not a static cluster. New companies can enter and old members can leave the cluster.

In this work a cluster is basically seen as a group of enterprises strategically grouped by regional or technological similarities to better exploit the market, having the will to collaborate with the others to accomplish a given business opportunity. A Virtual Enterprise (VE) is seen as a temporary alliance of enterprises created around a given business opportunity whose collaborative execution is supported by computer networks. Therefore, in this case, a VE is created from the cluster's enterprises.

This work presents an approach for a partners search and selection (PSS) system. This system, called MobiC-II, aims to assign a given set of moulds to the cluster members and to select afterwards the most suitable subset of members – a VE – to carry it out. The matter is not (only) to support the VE creation, but on how to do this in an *agile, smart* and *secure* manners, three important enabling elements towards enhancing the trust building process inside Techmoldes. The mobile agents approach

+ Supported by CNPq – The Brazilian Council for Research and Scientific Development.

L. M. Camarinha-Matos et al. (eds.), *Processes and Foundations for Virtual Organizations*

is used for dealing with the two first elements, and introduces some security mechanisms in the agents system to partially cope the third one.

This paper is organized as follows. Section 2 addresses the PSS and the trust building problems. Section 3 introduces the PSS' functional model. Section 4 stresses the proposed security mechanisms. Section 5 describes the implementation prototype. Section 6 provides some preliminary conclusions.

This work has been developed within IST MyFashion.eu and IFM projects [2].

2. TRUST BUILDING IN THE PARTNERS SEARCH AND SELECTION

The PSS process comprised in this work is illustrated in the Figure 1. The business opportunity (BO) is received from a client and it is coordinated by a given member of Techmoldes. A BO is at last represented by a set of moulds with their respective specifications. As the Techmoldes' clients usually forbid subcontracting, a member should manufacture one entire mould. It means that the VE to be formed is a relation enterprise:mould, although one enterprise can get responsible for more than one mould. A BO can also specify limits for price and delivery time, and the names of the enterprises disregarded to participate. The coordinator enterprise acts as an *independent broker* (like the enterprises E1 and E12), centralizing the interaction with the client (X or Y), and being responsible for the whole PSS process[3]. It means that several BOs can be running simultaneously inside the cluster, that a member can be involved in several BOs simultaneously (like E2 and E12), and that several possibilities of VEs can be found out for a given BO (such as VE1 and VE3). After an evaluation process lead by the current broker, a set of enterprises is elected/identified and then the VE is created (like the one illustrated in the VE1 by E2, E8, E3 and E12).

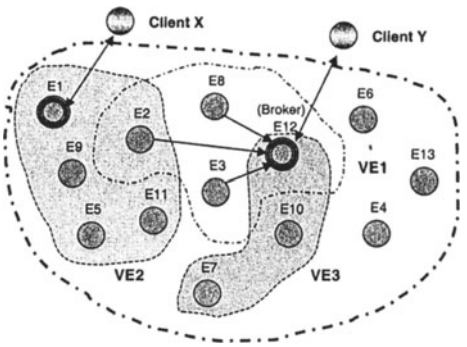

Figure 1 – The Techmoldes cluster scenario

The PSS system being developed (stressed in the section 3) deals with only a partial set of functionalities of a "complete" brokerage system, as specified in (Avila et al., 02). Anyway, the fundamental aspect is the fact that the enterprises should exchange information between each other in order to offer proposals of execution and to negotiate to each other until the VE is defined.

The trust building problem within the Techmoldes started here. Even the members knowing to each other and being aware about they should do that in order to be

candidate for a BO, they get reluctant to share some kinds of information, such as prices, delivery dates and capacities. The trust building process is indeed one of the most difficult issues to be overcome by the developers of VE solutions. Cultural, ethical, managerial, besides others "pure" IT-related problems, have been pointed out as obstacles for a wider adoption of the VE paradigm by the companies (Camarinha et al., 02). For instance, when Techmoldes was originally created there was only one broker, a business person in charge of performing the PSS process (the definitive VE composition was decided with the involved enterprises' representatives, helped by a PSS system prototype described in (Rabelo et al., 01)). However, this model had to be abandoned afterwards when some enterprises that lost businesses started to question how impartial the broker person had been in some business and how (s)he had handled private information from the candidates. This is the main reason why many independent and autonomous brokers can now coexist in the system.

3. THE PSS SYSTEM MODEL

The MobiC-II system being developed extends the work presented in (Costa et al., 02) and fits the new operation requirements depicted in the Section 2.

Several approaches have been applied in the development of PSS systems, with several degrees of computer assistance (Camarinha et al., 99)(Mejia et al., 02) (Siqueira et al, 01). Regarding the above mentioned system requirements and the envisaged scenario (figure 1), the multi-agent systems is seen as a suitable approach. An agent can be defined as a distributed software module launched in some node of the network that makes use of its skills, resources and communication with the other agents to perform some specific tasks. This definition reveals the notion of *stationary* agents, the ones that once launched in the network have no abilities to move through it. Despite the its potentialities, MobiC-II system exploits the use of *mobile* agents. It is seen as an agent that travels through a heterogeneous network, crossing various security domains and executing autonomously in its destination (Wangham et al., 03). The mobile agents paradigm has a number of advantages, such as lower dependence on the network for information exchange among agents, and a more flexible reconfiguration of the agents' missions and in the control architecture.

Figure 2 illustrates the envisaged scenario with the MobiC-II system. A broker receives a given BO and identifies (only) the potential enterprises that can accomplish each mould [1]. A summary of the mould specification is immediately sent out to them [2]. The enterprises receive it, evaluate its preliminary interest and capability, and send back to the broker either an answer "no" or "yes" [3]. The broker receives the answers and sends a mobile agent to the enterprises that said "yes", provided with the full BO's specification and the list of enterprises to visit [4]. The mobile agent arrives at the first enterprise and interacts with the local stationary agent, asking for the *delivery time* and the *capacity* [5]. The local agent, acting as the enterprise's representative, accesses these data from the legacy system / local database. After this the mobile agent asks the local supervisor about the *price*, as it is a very much critic information in the moulds sector. A negotiation process may be carried out locally [6]. The mobile agent moves to the next enterprise of the list with these information [7]. This process repeats until the end of the list, when the mobile agent returns to the broker agent with the proposals [8]. The agent broker generates the set of possible VEs, evaluates every VE composition and the human broker elects the most suitable

one, sending a "win" or "lose" message to the enterprises afterwards [9]. The election criteria applied on this case are the global *lowest cost* and *shortest delivery time*.

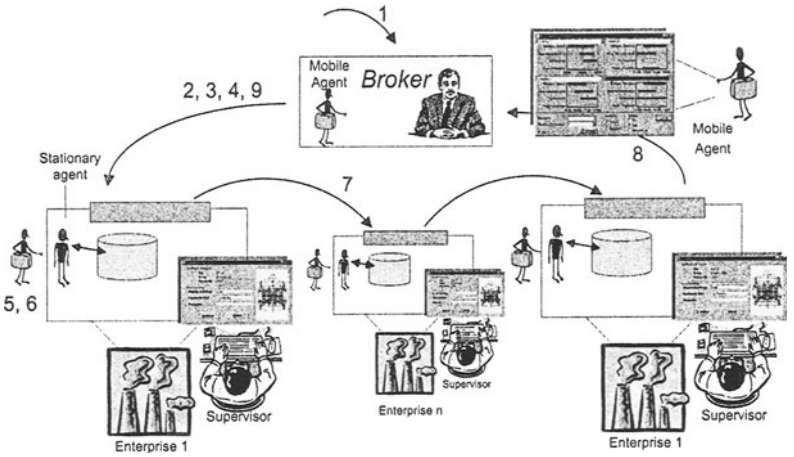

Figure 2 – The scenario for PSS for the VE creation

4. SECURITY ISSUES

In spite of its advantages, widespread adoption of the mobile agents technology is being delayed due to security concerns. Current available mechanisms to reduce the security risks do not efficiently cover all the existing threats, besides introducing performance restrictions that frequently outweigh its benefits (Jansen et al, 1999). Due to the characteristics of the mobile agent paradigm and the threats to which it is exposed, the security mechanisms have to be designed to protect: i) the communication channel; ii) the agents platforms; and iii) the agents themselves. In this paper, we deal with these issues in large-scale distributed infrastructure.

The selection of the security mechanisms that are being applied has been carefully evaluated in the *design phase* of the proposed multi-agent system. This enhances the quality of the system in the sense that the most suitable mechanisms can be conceived without losing their potentialities, which usually happens when they are implemented afterwards. Security policies and mechanisms determine which agents will be mobile and which will stay stationary, the scope of the agents' functionalities, and others.

4.1- Security in Mobile Agent Systems

Mobile agent platforms face several threats, such as (Jansen et al., 1999): *masquerade*, when an agent poses as an authorized agent in an effort to gain access to services and resources to which it is not entitled; *denial of service*, when agents launch attacks to consume an excessive amount of computational resources from an agents' platform; and *unauthorized access*, for example when an agent obtains read or write access to data for which it has no authorization.

The establishment of isolated execution domains (protection domains) for each incoming mobile agent and control of system domains entrances is an approach that has been commonly adopted with the purpose of offering protection to agent platforms. However, other issues need to be considered when distributed large-scale systems are the focus. In addition to this approach, other techniques have been

proposed based on conventional security techniques, such as: *secure code interpretation*, *digital signature*, *path histories*, and *Proof-Carrying Code (PCC)*.

The dangerous attacks of agents platforms against mobile agents are critical security problems to solve. The set of threats includes (Jansen et al., 1999): *masquerade*, when a platform poses as another platform in an effort to deceive a mobile agent as to its true destination; *denial of service*, when a malicious platform ignores agent service requests, introduces unacceptable delays for critical tasks, or simply does not execute the agent's code; *eavesdropping*, when a platform monitors every instruction executed by the agent, all public data, and all the subsequent data generated on the platform; and *unauthorized access*, when a malicious platform modifies a mobile agent by changing its code, its state, or both. Some mechanisms for agent protection include *Secure Hardware*, *Partial Result Encapsulation*, *Computing with Encrypted Functions*, and *Time Limited Blackbox*. However, these techniques cannot be considered suitable and flexible when a mobile agent needs to travel through several sites in a large-scale system. This occurs because mobile agents run under control of a platform and it is very difficult to prevent attacks against them.

4.2- Security Scheme Based on SPKI/SDSI Chain of Trust

The security mechanisms proposed are based on an agents model that assumes free itineraries and multi-hop. The Mobile Agent Facility (MAF) specification (OMG, 2000) is used as a guideline to achieve interoperability between agent systems. The proposed scheme has a completely decentralized authentication and authorization control based on certificate SPKI/SDSI[4] delegation mechanisms, that makes it adequate to distributed large-scale systems such as the Internet.

Figure 3 shows the procedures defined in the security scheme, which are composed by prevention and detection techniques. Below we analyze some aspects of to the mechanisms in the proposed scheme.

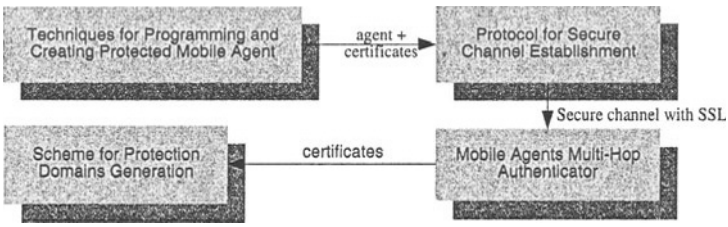

Figure 3 – Security Scheme for Agent System Protection

- *Techniques for Programming and Creating Protected Mobile Agent*

During the mobile agent creation process, the owner, being the authority that an agent represents provides a set of SPKI/SDSI authorization certificates defining agent's privilege attributes (its credentials). Also, agents can have attached platform lists that indicate which platforms are authorized to execute the agent.

The agent programmer can protect items in the agent's state so they are only accessible to certain platforms. This allows for selective disclosure of agent state (Karnik, 1998). In addition, the programmer can use empiric methods – as in (Hohl, 1998) - to prevent disclosure of agents' sensitive data by malicious platforms. Finally, the agent's owner must first sign the agent's code and the data defined by the programmer as read-only, and then create the agent in its home platform.

- *Protocol for Secure Channel Establishment*

In the proposed scheme, mutual authentication between the involved platforms must be established before agents can be transferred, which creates a secure channel in the communications infrastructure. This is performed via a Challenge/Response protocol based on SPKI/SDSI certificates of the owners of the platforms. The basis for authentication in SPKI/SDSI are chains of authorization certificates (Ellison et al, 1999). This process is concluded with the establishment of a secure channel that will remain valid in the subsequent interactions. For secure channel establishment, an underlying security technology (Secure Sockets Layer - SSL) is used for ensuring confidentiality and integrity of the communications between agent platforms.

- *Mobile Agents Authentication*

Before instantiating a thread to an agent, the destination platform must authenticate the received agent. We define a multi-hop authenticator that establishes trust on an agent, based on the authenticity of the owner of the agent, on the authenticity of the platforms visited by the agent.

Receiving a mobile agent, a platform must first check, through verification of the agent's signature, that this agent has not been corrupted and confirm its association to a *principal*, its owner. Thus, modifications introduced by malicious platforms can be detected by any platform visited by the agent.

In addition, for detecting others possible modifications and checking the multi-hop agent's traveling history, the destination agent platform must analyze the record of the agent's path. Moreover we propose that platform-generated sensitive data should be stored in a container to be carried by the agents. These sensitive data should be signed by the generating platform so that possible modifications can be detected. Visited platforms must be associated to agent's authority. They are defined in a list carried by the agent. If deemed necessary, it is possible to verify the authenticity of the visited platforms' signatures in each record entry as well as the container's integrity.

- *Procedure for Generation of Protection Domains*

Protection domains and the permissions assigned to them are defined after trust in an agent has been established. They are based on the agent's SPKI/SDSI authorization certificates. The authorization chains carried by an agent, representing its credentials, need to be verified by the platform guardian for the set of permissions to be defined and for the protection domains to be generated. This scheme decouples the privilege attributes granted to *principals* (agent's credentials) from the attributes required to access resources protected by the platform (control attributes or policies), offering a more flexible and dynamic access control for large-scale systems.

5. IMPLEMENTATION MODEL

Three classes of agents compose the MobiC-II architecture/system:

- Broker Agent: it is a stationary agent basically responsible for getting the business opportunity (BO), distributing it to the potential enterprises, sending an Agent Messenger to them, and collecting/electing the final VE composition.
- Mobile Agent: it is a mobile agent responsible for delivering the BO to the enterprises, negotiating locally with them, and keeping traveling through the net to the other enterprises and finally back to the broker.

- Enterprise Agent: it is a stationary agent responsible for receiving the BO, evaluating it, accessing the local database to get the required information, and answering the BO request to the mobile agent.

The use of mobile agents requires extra services from the machines where those codes will be executed. Each machine needs an *agent platform* to support the applications to dynamically relocate its software components in different sites.

The agents are placed in two heterogeneous platforms. The *Broker* and *Mobile* agents use *Aglets* (Java based) and the Enteprise agent uses *Massyve* (C++ based). Aglets is an open-source platform developed by IBM (1996). Its development kit (ASDK) provides a standard computing environment called *Tahiti* which supports creation, cloning, execution, dispatching, and retraction of agents. Java RMI is used as the supporting communication infrastructure for the Aglets-based agents. The *Enterprise* agents are based on the Massyve platform. The interoperability between the Java and C++ agents are supported via CORBA.

To meet the specific requirements of the applications, we are designing a flexible framework that implements the proposed security scheme and allows selecting a subset of mechanisms that could also be adequate to the applications' functionalities. For example, if the cluster decides that enterprise's answers is a sensitive data that could not be revealed to others enterprises, then the broker can use the framework to setup that this data will be encrypted and stored in an agent's container.

The protocol for secure channel establishment and the multi-hop authenticator (see section 4.2) were implemented with the SDSI 2.0 library and with Java 2 cryptographic tools. The SSL is used as an underlying security technology in our scheme and it was integrated to the Aglets platform.

As the agent platform chosen for the prototype is based on Java, the secure interpretation of the agents' code and the definition of the protections domains to mobile agents are provided, in part, by the Java 2 security model. The definition of protection domains has been fully designed but only partially implemented. The process for generating the set of permissions was defined to overcome the limitations related to the Java 2 access control model. Some extensions to the Java 2 security model are needed for generating the protection domain.

6. CONCLUSIONS

This paper presented an approach on how trust building in mobile agent-based architectures can be reinforced by using some security mechanisms in the process of searching and selecting partners to create a Virtual Enterprise. In order to minimize the usual outweigh of these mechanisms, this work allows their configuration at the design phase of the application, using only the necessary mechanisms with their full features. The security mechanisms are used to protect the communication channel, the agents platforms, and the agents themselves, in open and large-scale systems.

Some preliminary implementation results are shown and discussed. The developed system combines stationary and mobile agents. The agents are placed in two heterogeneous platforms in order to support the communication between the system and the enterprises' systems. These platforms are Java-based and C++ based, and the interoperation between them is support by CORBA.

The system performs the partners' search and selection process in an agile way as the business opportunity is immediately spread out to the enterprises by the so-called broker enterprise and a set of possible VEs is rapidly generated. This is carried out in a smart way since only the potential enterprises are consulted as well as negotiation actions can occur locally, between the mobile and the stationary agents. It is accomplished in a more reliable manner as the mobile agents do not require a permanent connection with the broker enterprise, so network failures are overcome.

Next steps will be related to testing and assessing this approach in a more real scenario, considering its performance and the efficiency of the security mechanisms chosen. Once validated it will be tested in the Techmoldes environment. Further the integration of the PSS system into a wider business process management system under development that handles with the VE configuration and operation phases.

Acknowledgements
The authors would like to thanks the Brazilian funding agencies CNPq and Capes for the financial support, Prof. Rolando Vallejos for having provided information about the cluster, and to Mr. Carlos Gesser, Mr. Galeno Jung and Ms. Elizabeth Fernandes for the implementation of some parts of the work.

7. REFERENCES

Avila, P.; Putnik, G.; Cunha, M. (2002). Brokerage function in agile virtual enterprise integration, Proceedings PRO-VE'2002 Conf., Kluwer Academic Publishers, pp.65-72.

Camarinha-Matos, L.M.; Afsarmanesh, H. (2002). Dynamic Virtual Organizations, or not so Dynamic ?, Proceedings PRO-VE'2002 Conf., Kluwer Academic Publishers, pp.111-124.

Camarinha-Matos, L.M.; Cardoso, T. (1999), Selection of Partners for a Virtual Enterprise, Proceedings PRO-VE'99 Conf., Kluwer Academic Publishers, pp.259-278.

Costa, S.; Rabelo, R. (2002). Supporting the creation of virtual enterprises using mobile agents, Proceedings PRO-VE'2002 Conf., Kluwer Academic Publishers, pp.371-378.

Ellison, C. M. and et al (1999). *SPKI Requirements*, The Internet Engineering Task Force.

Hohl, F. (1998). Time limited blackbox security: Protecting mobile agents from malicious hosts, *In:* G. Vigna (ed.), *Mobile Agents and Security*, V. LNCS 1419, Springer, p. 92–113.

IBM (1996). Aglets software development kit. (http://www.trl.ibm.co.jp/aglets).

Jansen, W. and Karygiannis, T. (2000). *NIST Special Publication 800-19 - Mobile Agent Security*, National Institute of Standards and Technology, USA.

Karnik, N. (1998). Security in Mobile Agent System, PhD thesis, Univ. of Minnesota, USA.

MASSYVE, http://www.gsigma-grucon.ufsc.br/massyve.

Mejia, R.; Aca, J.; Garcia, E.; Molina, A. (2002). E-Services for Virtual Enterprise Brokerage, Proceedings BASYS'2002 Conference, Kluwer Academic Publishers, pp.133-140.

OMG (2000). Mobile Agent Facility specification, OMG Document 2000-01-02.

OMG (2001). The Common Object Request Broker Architecture v2.6, OMG Doc. 01-12-30.

Rabelo, R.; Vallejos, R. (2001). A semi-automated brokerage for a virtual organization of mould and die industries in Brazil, Proc. I3E'2001, Kluwer Academic Pub., pp. 193-208.

Siqueira, J.; Bremer, C. (2000). Action research: the formation of a manufacturing virtual industry cluster, Proceedings PRO-VE'2000 Conf., Kluwer Academic Pub., pp.261-268.

Wangham, M., Fraga, J., Obelheiro, R. (2003). Security Mechanisms for Mobile Agent Platforms Based on SPKI/SDSI Chains of Trust. To be presented in SELMAS'2003 Conf.

---

[1] *Techmoldes* is a fictitious name as it was not allowed to use the real name.

[2] www.myfashion.org, www.ifm.org.

[3] So far there is not a supporting system to help the broker in the coordination of the business execution, i.e. along the VE *operation* phase. It is made in the traditional way (phone, fax, etc.).

[4] Simple Public Key Infrastructure / Simple Distributed Security Infrastructure (Ellisson et al., 1999)