Towards A New Security Architecture of Mobile Agents

Hassan Razouki, Abdellatif Hair

Abstract— Security presents a crucial point in mobile agent systems and may hinder the expansion and use of this paradigm. The protection of mobile agents is considered as one of the greatest challenges of security, because the platform of execution has access to all the components of the mobile agent. In this paper, we present a new architecture paradigm of mobile agents, which allows the separation of the implementation tasks of the agent and its security mechanisms.

Our approach is based on using two strategies of adaptation to adapt the mobile agent security at runtime, depending on the sensitivity of the services required to perform the duties of the agent and the degree of confidence of the visited platforms. The first is a static adaptation performed by the MSAS (the Management System of Agents Security). The second is a reflexive structural dynamic adaptation performed by the mobile agent itself. These two adaptations take into account the dynamic security requirements in systems based on mobile agent.

Index Terms—Mobile agent, Software components, Static adaptation, Dynamic adaptation, Security, Trusted platform, Cryptography.

I. INTRODUCTION

Nowadays, mobile agents [1] [2] [3] gain more and more important because of their ability to migrate autonomously through a network in order to have access to resources and remote services. Indeed, a mobile agent is particular software with capabilities of adaptability, autonomy and mobility. This agent can be the target of several types of attacks [4] [5] [6] [7] as it moves through platforms that can analyze or alter its content and through a network that is not always safe, in fact, a malicious host may try to attack a mobile agent to obtain a service freely or access to the memory of the agent in order to view or manipulate private information. Other examples of such attacks are malicious alteration of the code of the mobile agent and the monitoring of its execution. The agent is vulnerable while running on the platform of the host. Therefore, the owner requires safeguards for the protection of the agent against threats from malicious hosts. Thus, the mobile agent must protect itself against any act aimed at the damage, destruction or manipulation of its code, state or data.

Protection of the mobile agent against malicious host remains an open and difficult problem because the runtime has full control over the mobile agent.

The analysis at the end of this paper show that the mobile agent should use different security mechanisms for each platform, these mechanisms are determined by the sensitivity of the services requested by the mobile agent and the credibility of each platform.

The Proposed approach is to find a new mobile agent paradigm architecture, which can protect the mobile agent via two strategies of adaptation. It takes into account the dynamic aspects of the security needs of a mobile agent in each runtime environment.

- The first is a static adaptation performed by the MSAS (Management System of Agents Security) based on the sensitivity of the services requested by the agent, the MSAS adds additional security components and determines the policy of the dynamic adaptation to be followed by the mobile agent during its execution.
- The second is a reflexive dynamic structural adaptation performed by the mobile agent itself. According to the degree of confidence on the platform visited, the mobile agent selects and adapts security components to the tasks to be performed by this platform.

This paper is organized as follows: in section 2 we begin by identifying the state of art and present the different approaches based on adaptability and reflexivity to protect mobile agents. Section 3 is devoted to presenting some concepts used in our proposal. Section 4 presents the new architecture of our system and identifies the functions of the various components. Finally, a conclusion is presented in section 5.

II. THE SECURITY OF MOBILE AGENTS

2.1. Security issues related to mobile agents

Three types of problems arise with regard to security in the concept of mobile agent: the security of the agent migration, the protection of the platform against agents and malicious platforms, and the protection of the agent against other agents and malicious platforms [7].

An agent's attacks against platform are varied. A hostile agent may attempt to access resources without authorization of the platform, consume too many resources or try to impersonate another agent. The protection of the visited hosts against attacks carried by malicious mobile agents is a problem that is now fairly well controlled.

A platform's attacks are even more important, because the platform has access to all the components of the agent and may change them, if there is no technology to detect or better, to prevent these attacks. There are four major attack categories:

- The inspection is to examine the contents of the agent, or the execution flow to retrieve sensitive information carried by the mobile agent.
- The modification is realized by replacing certain elements (data or a portion of the code) of the mobile agent in order to lead an attack. The replacement of the code will prompt the agent to perform malicious operations on future guests to visit. While the replacement of data allows the malicious host to manipulate the mobile agent

Manuscript received January 15, 2014.

Hassan RAZOUKI, Laboratory of Modeling and Computation (LMC), Faculty of Science and Technology, University Sultan Moulay Slimane, Beni Mellal, Morocco.

Abdellatif HAIR, Laboratory of Modeling and Computation (LMC), Faculty of Science and Technology, University Sultan Moulay Slimane, Beni Mellal, Morocco.

to its advantage.

- The replay meanwhile is obtained by cloning the agent then running the clone in several configurations to find out the competencies of the agent.
- Denial of service: as a malicious host can ignore service requests, introduce unacceptable delays for critical tasks, do not execute the code of the agent or end it without warning.

Protecting a mobile agent against malicious hosts is to protect mainly its execution, integrity and confidentiality [8] [9] [10].

2.2. Approaches to the protection of the mobile agents

Several approaches for the protection of mobile agent have been proposed. They try to ensure the access of the mobile agent to hosts in which it may have confidence or detect those that are malicious. They are primarily intended to detect attacks or render them ineffective, our proposal is based on protection.

In the following section, we are interested in approaches based on adaptability and reflexivity.

The approach to the application of the code [11] protects the integrity of the mobile agent using an agent's code dynamically extensible. In this approach, new modules of the function of the agent can be added and redundant modules can be removed during the execution of the agent.

Ledoux and Bouraqadi-Saâdani [12] studied the adaptation of the mobile agent systems during the execution. The main idea is that the introspection of the network can be used to dynamically select the best execution policy.

Zhi Zhogwen [13] propose to highlight a framework in which mobile agents can dynamically select an implementation of security among others. The solution of a dynamic adaptation of the mobile agent security that changes the security implementation according to the change of the status of the environment in which it runs.

Amara Hachmi and El Fallah-Seghrouchni [14] propose a component-based architecture for a mobile agent in order to increase the modularity, reusability, extensibility, and self-adapting. When the mobile agent moves from one host to another, the context changes. The result of dynamic adaptation is the selection of suitable components and the linking of selected components for the new context.

Hacini [15] proposed an adaptive mobile agent to protect itself against malicious websites. It is based on an architecture based on behavioral adaptability that allows the mobile agent to change its behavior according to the perception of the different variations of the environment. The idea is to calculate a degree of confidence according to which the mobile agent chooses the appropriate behavior.

Leriche and Arcangeli [16] proposed a model of the mobile agent that is self-adaptive, by assembling reusable components. The agents are configured and may be reconfigured at runtime so that they are able to respond to changes in the execution environment.

III. BASIC CONCEPTS

Our proposal combines three concepts: adaptability, reflexivity and software components. In what follows, we present an overview of each.

3.1. Adaptability

Adaptation refers to the act of adjusting and reacting to changes in the environment constraints and the needs of users. The term "adaptability" refers to the capacity or the degree of adaptation. There are two types of adaptation:

- Static adaptation: this case corresponds to an adjustment made before the execution based on the knowledge held in the deployment environment.
- Dynamic adaptation: mobile agents foresee their execution environment and react autonomously to changes.

The concept of static and dynamic adaptability is used in our case for the prevention of attacks by the malicious host.

3.2. Reflexivity

Reflexivity built an attractive approach for the development of adaptable applications. It is considered as a way of internal organization of a system to facilitate its development, adaptation, and reuse. The great advantage of this approach is to express treatment in extremely broad terms, using only the constituent concepts of the system.

Malenfant [17] defines it as: "the ability of a program to manipulate data as something representing the state of the program during its execution". In other words, it is the ability for a program to reason and act upon it.

In our approach, we rely on the reflective and the structural adaptability that allows the mobile agent to inspect and modify its own structure, to make choices and adapt to the execution context, our mobile agent must know not only the state of its environment, but also its own state and its structure.

3.3. Software components

A software component is a unit of composition with contractually specified interfaces and explicit context dependencies only. A software component can be deployed independently and is subject to composition by third parties.

IV. A NEW PERSPECTIVE OF SECURITY

The study carried out in the previous section was used to evaluate the importance of the use of mobile agents in distributed applications and to measure the interest of solving the security problem. Also, it showed the lack of universal solution to the problem of malicious host.

The analysis at the end of this study showed that all approaches use the same security mechanism to protect mobile agents against different malicious platforms, without taking into account either the security requirements of each agent (the sensitivity of the information contained in the mobile agent) and the credibility of platforms. Indeed, each mobile agent requires different security mechanisms based on their services and based on the credibility of each visited platform. For example, as shown in Figure 1, an agent implements security mechanisms against different types of attacks based on the credibility of each host:

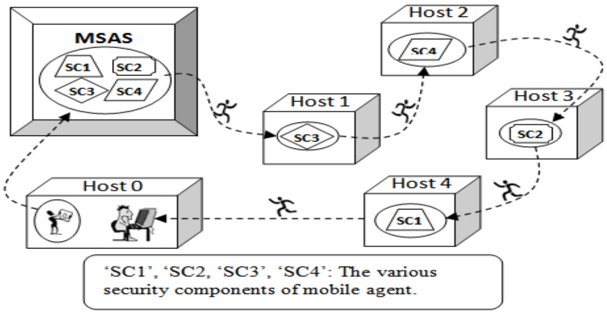
MSAS
SC1
SC2
SC3
SC4
Host 2
SC4
Host 1
SC3
Host 3
SC2
Host 0
Host 4
SC1
'SC1', 'SC2, 'SC3', 'SC4': The various security components of mobile agent.

Figure 1: The mobile agent uses different security mechanisms for each host.

The agent uses security mechanisms against replay attacks [18] in the host 1, mechanisms of protection of the itinerary [19] in the host 2, protection mechanisms of the calculation result [20] in the host 3, and the protective mechanisms of the results obtained during the execution of the agent [21] in the host 4. These security mechanisms are stored in the MSAS as security components. The MSAS must support and update all existing security mechanisms.

Our approach is based on two strategies of adaptation:

- The first is a static adaptation that adds security components according to the sensitivity of the services required by the agent, and determine the adaptation policy to be followed by the mobile agent during its execution. This adaptation is performed by the cooperation of the four components of MSAS (SSA, MAS, MACD and MAC) (Figure 2).
- The second is a reflexive, structural and dynamic adaptation which allows the selection and adaptation the security components of tasks to run on this platform. This adaptation is performed by the mobile itself according to the degree of confidence of the visited platform.

In order to increase the level of security offered and provide a satisfactory protection model, we also use traditional protection mechanisms such as password, cryptography and digital signature. These mechanisms help to preserve the integrity of the data of our mobile agent, and to control access to its resources.

4.1. General architecture of the system

A system based on mobile agents is a computer network consisting of a set of interconnected machines called hosts. A host is a machine identified by a name and a set of hardware resources (CPU, memory, network resources, etc...) and software (operating system, database, etc..) that can host one or more execution systems of mobile agents (Figure 2).

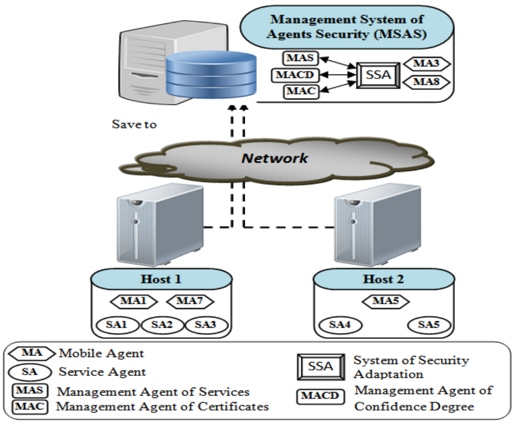
Management System of Agents Security (MSAS)
MAS
MACD
MAC
SSA
MA3
MA8
Save to
Network
Host 1
MA1
MA7
SA1
SA2
SA3
Host 2
MA5
SA4
SA5
MA Mobile Agent
SA Service Agent
MAS Management Agent of Services
MAC Management Agent of Certificates
SSA System of Security Adaptation
MACD Management Agent of Confidence Degree

Figure 2: General architecture of the system.

In our approach, there are two types of host:

- MSAS (Management System of Agents Security): main host has information about the credibility, the hardware / software configuration and the list of services available to each host. The main functions of MSAS are:
- Record information about hosts in its database. This information is used for host authentication (Id-host, password, certificate, operating system, processor architecture ...).
- Record the list of available services and to estimate the degree of confidence for each host.
- Analyze and determine the sensitivity of the services requested by the mobile agents.
- Create intelligent itinerary (the list of hosts to visit with their confidence degree) and add the most suitable mobile agents security components.
- Host: This is the entity responsible for the creation and initial emission of mobile agents to the MSAS, in order to determine their intelligent itinerary based on the services requested by the mobile agent. The host is subscribed to one or more services that mobile agent will perform at the level of the host.

4.2. Components of the MSAS

4.2.1. SSA (System of Security Adaptation)

Systems based on mobile agents are characterized by a very dynamic aspect. This is due mainly to the migration of agents to multiple systems with different behaviors and security policies. Indeed, while visiting a new system, the agent must adapt dynamically to the security requirement.

The definition of an adaptation policy and security components of the mobile agent is a crucial step for the effective implementation of security in a mobile agent system. The goal of the SSA is to protect the mobile agent via a static adaptation (Figure 3). This adaptation is to transform the mobile agent to a secure mobile agent. The SSA adds additional security behaviors (security components) and determines its dynamic adaptation policy during execution.

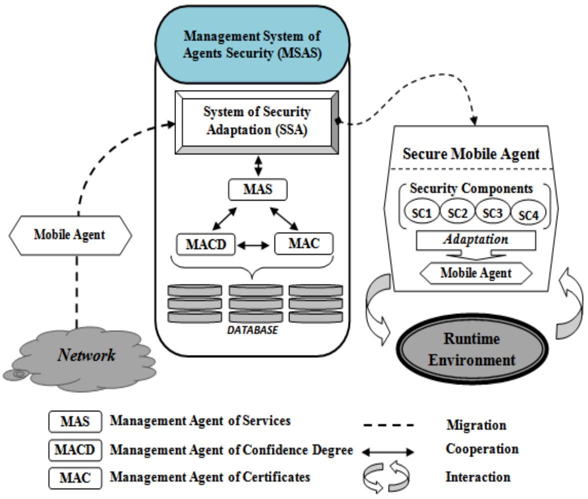
Management System of Agents Security (MSAS)
System of Security Adaptation (SSA)
MAS
MACD
MAC
DATABASE
Mobile Agent
Network
Secure Mobile Agent
Security Components
SC1
SC2
SC3
SC4
Adaptation
Mobile Agent
Runtime Environment
MAS Management Agent of Services
MACD Management Agent of Confidence Degree
MAC Management Agent of Certificates
Migration
Cooperation
Interaction

Figure 3: The adaptation of a mobile agent to the SSA.

4.2.2. MACD (Management Agent of Confidence Degree)

Security agent requires dynamic assessment of the

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

credibility of the host to visit, which can increase or decrease the security of the mobile agent. The mobile agent trusts the host when considering its degree of confidence as very favorable, or increase the level of security in the case of an unfavorable confidence degree. Hence, it is important that mobile agents can authenticate and identify the list of hosts in their itinerary.

MACD is used to evaluate the credibility and estimate the confidence of each host visited by the mobile agent. This estimation is based on information stored in a database in MSAS. The MACD updates such information following an inspection / observation by the mobile agent during its execution on the host visited.

4.2.3. MAS (Management Agent of Services)

The MAS is used to analyze and determine the sensitivity of the services requested by mobile agents, then look for and filter hosts that provide these services. The MAS can also record and update the list of services made available by a service agent.

4.2.4. MAC (Management Agent of Certificates)

The MAC uses symmetric and asymmetric cryptography to prevent the behavior analysis of the mobile agent. Symmetric cryptography is used to encrypt / decrypt sensitive tasks of the mobile agent. The list of secret keys is shared between the MSAS and the hosts. Asymmetric cryptography is used to ensure authentication and security of communication between the different entities of the system. For this, it uses reliable mechanisms such as encryption, hashing and digital signature. The MAC can also record and check the hosts' certificates (ID certificate, validate date ...).

4.3. The components of a host

4.3.1. Systems of execution agents

The host contains one or more systems of agent's execution that provide the basic functions for the execution of the mobile agents: the creation and initiation of the mobile agents, the communication (local and remote) between agents, the access to resources and the migration of the agent

4.3.2. Service Agent

Service Agent (SA) is an agent that cooperates with other agents to satisfy the request by a mobile agent (visitor). A service can be accomplished by one or by the cooperation of several service agents (Figure 4).

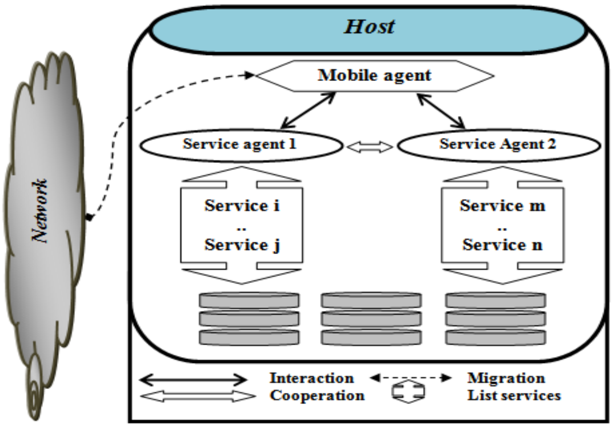

Figure 4: The interaction between a mobile agent and service agents.

The properties of a service agent can be distinguished, as follows:

- A service agent is used as an interface between the source of information on the host and the mobile agent visitor.
- A service agent has the competencies and provides a set of services.
- A service agent cooperates with other service agents and / or interacts with mobile agents in the same host.

4.4. Internal structure of a secure mobile agent

A secure mobile agent contains everything that is important: code (the library of tasks and security) and critical data (the agent and MSAS memories). In addition, it provides managing components (manager, controller, analyzer and interface) that are added by the SSA (Figure 5).

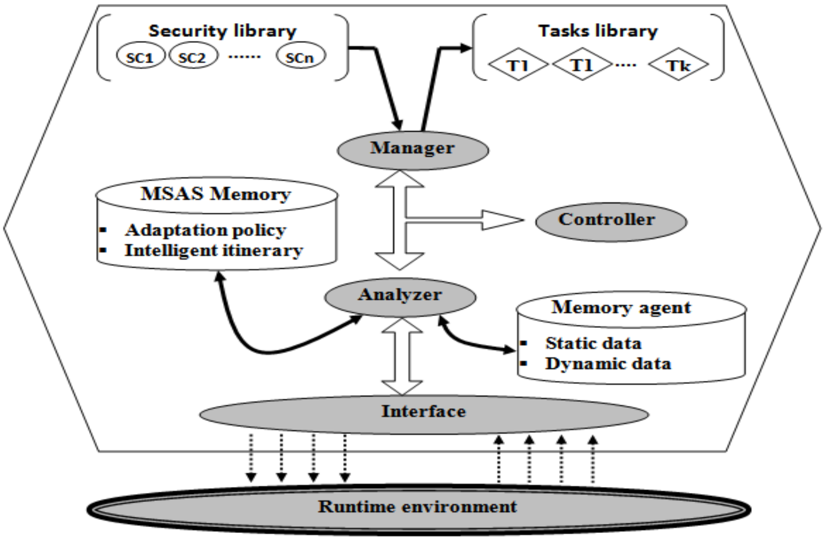

Figure 5: Internal structure of a secure mobile agent.

4.4.1. Library of tasks and security

The tasks library contains the functional code of the agent (behavior determined by the programmer of the agent application).

The security library contains security replaceable components that the mobile agent can use during adaptation. The components are added by the MSAS based on the security needs of the mobile agent. Each component is a piece of code selected by the analyzer According to the degree of confidence of each platform.

4.4.2. Agent memory and MSAS

Memory agent contains static and dynamic data transported from the original host or hosts visited by the mobile agent.

Static data: this is information that is transported from the original host. This information contains data that do not change such as the identity of the creator and the digital signature.

Dynamic data: this is information collected from the runtime environment after each migration. Such As the partial results of calculation of each platform, the data obtained during the execution of the agent...

The MSAS memory contains the data added by the MSAS and carried by the mobile agent during its execution.

The policy of adaptation: represents a very important factor to support the protection strategy we propose for our mobile agent. Indeed, the adaptation policy is a set of rules that defines the relationship between the security components of the agent and the degree of confidence associated with changes of the execution context.

The intelligent itinerary: consists of a set of nodes. A node is a connection point between the task to be executed by the agent, the identity and the confidence degree of the platform of execution. This section also contains the estimated time of the execution of each task in order to avoid replay attacks.

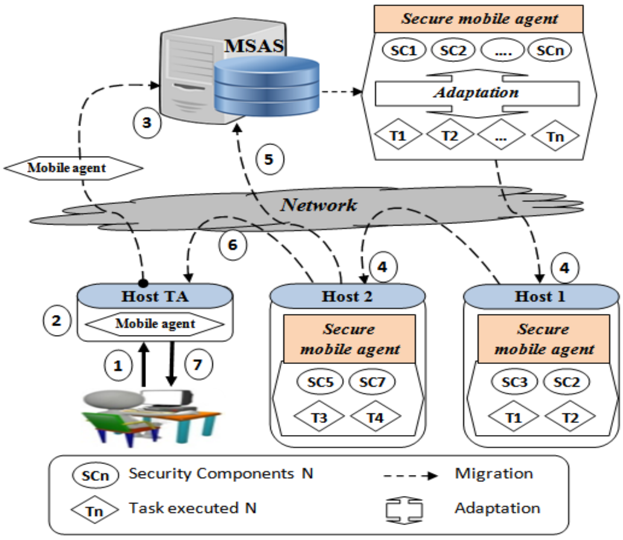

Figure 6: Execution simulation of mobile agent.

The traveler asks his travel agency to organize a trip from Casablanca to Paris for 7 nights.

1) The travel agency creates and initiates a mobile agent to perform the tasks requested by the traveler. The behavior of the mobile agent is encrypted using the public key of MSAS, which allows it to guarantee the confidentiality of migration

4.4.3. Managing components

Interface: this is the component through which the agent communicates and interacts with the runtime environment. This component allows authenticating platforms, detecting accesses of the external hosts to its resources, receiving requests and providing services in a suitable form.

Analyzer: After authenticating the interface, the analyzer determines the nodes associated with the execution platform, and then extract the tasks to be performed and the degree of confidence from this node. Depending on the degree of confidence and adaptation policy, the analyzer decides to add or replace a security component in the tasks to be executed by the platform.

Manager: allows changing the structure of the mobile agent, to fit with the new execution condition. It gels the selected components by the analyzer and places them in the code of the tasks associated with the execution platform. The manager serves only as a mechanical operator to achieve the dynamic change of components.

Controller: Provides the proper functioning of the whole agent system and can also detect any poor execution of the mobile agent in the platform in question. The controller compares the actual execution time of each task with the estimated time. If the controller discovers an exceeding in the prescribed time, it can assume this as a misuse of the mobile agent and therefore, the agent must stop its execution in the concerned host in order to migrate to the next host in its itinerary. The controller sends notifications to MSAS, following a detected attack.

4.5. Execution simulation of mobile agent

The objective of this simulation is mainly to show how to use a secure mobile agent to arrange a trip from Casablanca to Paris for 7 nights (Figure 6). The travel agency helps its clients to supervise the organization of the trip: it takes care of the trip’s arrangements by selecting the best hotels, route planning and guarantee the best tickets price …
between the host of the travel agency and MSAS. The host of the travel agency sends the mobile agent to the MSAS.

2) The MSAS decrypts the behavior of the mobile agent using its private key, and then analyses and determines the sensitivity of the services requested by the agent to adapt the agent’s security:

a) Adds security components that are best suited to the sensitivity of services, such as a security component to protect the results obtained during the execution of the agent (cheapest room, tickets), and ensure the confidentiality of sensitive tasks (online payment ...).

b) Creates the intelligent itinerary for the mobile agent (the host ID, password and tasks to be performed, the degree of confidence). This information is encrypted using the public key of MSAS. The comparison between the degree of confidence and sensitivity of services allows the MSAS to perform a first filtration of hosts.

c) Uses symmetric cryptography and hash functions to protect sensitive tasks against inspection and modification attacks.

3) Arriving on the execution host :

a) The interface of the mobile agent must authenticate the host in question, and verify some information necessary for the performance of its task. To do so, the agent must obtain certain information from the runtime environment (for *example, the identity of the host visited the password ...). Then, the interface encrypts the information collected using the public key of MSAS, to compare them with those it holds.*

b) After authentication of the host, the analyzer determines the degree of confidence and the tasks that need to be executed on this platform. Depending on the adaptation policy, the analyzer may decide to decrypt sensitive tasks and execute them on the host if the confidence is very favorable, add or replace security components with the help of the manager if the degree of confidence is not favorable, or, stop and leave the host to go to the next while notifying the MSAS about this failure.

4) The controller sends a notification/observation to MSAS, after a detected attack or an exceeding in the execution time, during the dynamic interaction between the mobile agent and the host, this notification reduces the credibility given to the host.

5) The mobile agent returns to the host of the travel agency with the results obtained from the different hosts (example: the name of the hotel, the reservation price, the departure date and the ticket price...).

6) The travel agent host sends the results to the traveler, using e-mail, HTTP request …

V. CONCLUSION

Our study showed that the security of the mobile agent requires the addition of a dynamic assessment of the credibility of the host visited and determination of the sensitivity of the services requested by the mobile agent. According to our knowledge, no approach has used the service sensitivity and adaptability to solve this problem. We have proposed a new architecture for mobile agents and identified the functions of different system components. This proposal is based on two strategies of adaptation. The first is a static adaptation performed by the MSAS. The second is a reflexive dynamic structural adaptation performed by the mobile agent itself.

In our future work, we will try to learn how to identify the sensitivity of the services requested by the mobile agent and the parameters defining the degree of confidence of each platform.

REFERENCES

1. J. Baumann, F. Hohl, K. Rothermel and M. Straber, “Mole– Concepts of a mobile agent system,” World Wide Web Journal, vol. 1, no. 6, 1998, pp. 123-137.
2. D. Kotz, R. Gray, “Mobile Agents and the Future of the Internet,” ACM Operating Systems Review, vol. 33, no. 3, 1999, pp. 7-13.
3. R.J. Thomas, T.D. Mount, "Using software agents to test electric markets and systems," Power Engineering Society General Meeting, IEEE, Vol. 3, 2005, pp. 2808 – 2812.
4. N. Karnik, “Security in mobile agents systems,” PhD thesis, Department of Computer Sciences and Engineering, University of Minnesota, Minneapolis, USA, 1998.
5. E. Bierman, E. Cloete, “Classification of Malicious Host Threats in Mobile Agent Computing,” Proceedings of SACICSIT2002, pp 141-148, 2002.
6. N. Borselius, “Mobile Agent Security,” Electronics Communication Engineering Journal, IEEE. London, vol. 14, no. 5, 2002, pp 211-218.
7. P. Bella vista, A. Corradi, C. Frederici, R. Montanari and D. Tibaldi, “Security for Mobile Agents: Issues and Challenges,” Invited Chapter in the Book Handbook of Mobile Computing, I. Mahgoub, M. Ilyas(eds.), CRC Press, 2004.
8. J. Mir, J. Borrell, “Protecting General Flexible Itineraries of Mobile Agents,” in K. Kim (Ed.): ICICS 2001, LNCS 2288. © Springer-Verlag Berlin Heidelberg 2002, pp 382-396.
9. H. Yunguick Lee, J. Alves, “The Use of encrypted functions for mobile agent security,” Proceedings of SAICSIT 2002, pp141-148.
10. A. Dadon-Elichai, “RDS: Remote Distributed Scheme for Protecting Mobile Agents,” In: The Third International Joint Conference on Autonomous Agents and Mutli-Agent Systems, 2004.
11. T. Wang, S. Guan and T. Khoon Chan, “Integrity Protection for Code-On-Demand Mobile Agents in E-Commerce,” The Journal of Systems and Software 60, pp 211-221, 2000.
12. T. Ledoux, N. M.N.Bouraqadi-Saâdani, “Adaptability in mobile agent systems using reflection,” in ECOOP 2000, Workshop on Reflection and Metalevel Architectures, Cannes, France, 2000.
13. W. Zhi, G. Zhogwen, “A Dynamic Security Adaptation Mechanism For Mobile Agents,” Proceedings of the International Computer Congress, China, 2004, pp 334-339.
14. N. Amara-hachmi, A. El Fallah-Seghrouchni, “Towards a generic architecture for self-adaptive,” Proceedings of 5th European Workshop on Adaptive Agents and MultiAgent Systems (AAMAS’05), Paris, 2005.
15. S. Hacini, C. Cheribi and Z. Boufaïda, "Dynamic Adaptability using Reflexivity for Mobile Agent Protection", in Transactions On Engineering, Computing And Technology Enformatika, vol. 17, 2006 ISSN 1305-5313, Cairo, Egypte, pp 222-227,December 2006.
16. S. Leriche, J. Arcangeli, “Flexible architectures of adaptive agents : the agentφ approach” International journal of grid computing and multi agent systems (IJGCMAS), vol. 1, no. 1, 2010, 55-75.
17. J. Malenfant, F. Peschanski, L. Seinturier et J-P. Briot, "ALADYN: Architectures logicielles pour l’auto-adaptabilité dynamique", Université Pierre et Marie Curie UFR d’informatique, 2004.
18. C. Garrigues, N. Migas, W. Buchanan, S. Robles and J. Borrell, “Protecting mobile agents from external replay attacks,” The Journal of Systems and Software, vol. 82, no. 2, 2009, 197–206.
19. C. Garrigues, S. Robles and J. Borrell, “Securing dynamic itineraries for mobile agent applications,”Journal of Network and Computer Applications, vol.31, no. 4, 2008b, 487–508.
20. P. Maggi, R. Sisto, “A configurable mobile agent data protection protocol,” In Proceedings of the 2nd International Conference on Autonomous Agents and Multiagent Systems (AAMAS ’03). ACM Press, 2003, pp. 851–858.
21. J. Zhou, J.A. Onieva and J. Lopez,”Analysis of a free roaming agent result-truncation defense scheme,” In Proceedings of the IEEE International Confernce on e-Commerce Technology (CEC ’04). IEEE Computer Society, 2004, pp. 221–226.

AUTHORS PROFILE

Hassan Razouki, is with the Laboratory of Modeling and Computation (LMC), Faculty of Science and Technology, University Sultan Moulay Slimane, Beni Mellal, Morocco.

Abdellatif Hair, is with the Laboratory of Modeling and Computation (LMC), Faculty of Science and Technology, University Sultan Moulay Slimane, Beni Mellal, Morocco.