Establishing a Secure Open-Environment for Using Mobile Agents in Electronic Commerce

Paulo Jorge Marques*, Luís Moura Silva, João Gabriel Silva
Departamento de Engenharia Informática, Universidade de Coimbra, Portugal

{pmarques, luis, jgabriel}@dei.uc.pt

Abstract

Although mobile agents are a promising technology, the large-scale deployment of agents and the existence of hosts running agencies will not happen until proper security mechanisms are well understood and implemented. When considering global open environments as the Internet, mobile agents can be the victims of attacks by malicious hosts. In this paper, we present a security framework that protects agents from interference of untrusted and potentially malicious hosts. The framework can be used to enable technologies as electronic commerce, using the mobile agent paradigm in a secure and trustful way.

1 Introduction

When considering the deployment of applications based on mobile-agents for electronic-commerce two issues are of most relevance: protecting the host from attacks of malicious agents and protecting the agents from malicious hosts [2][3].

In this paper we will assume that agents will be deployed on a large open network – the Internet. We will present a security framework that can be effectively used to protect agents from attacks by malicious hosts. The framework is based on restricting the access level of the agent according to the trust level that is assigned to the current host. Certain methods can only be executed on certain hosts that are minimally trusted. Methods that cannot be executed on that host are kept encrypted. Data is also selectively accessible according to the trust placed on the host.

Our approach combines the use of encrypted methods and data with a scheme of surveillance agents. These agents assess levels of trust on network hosts, enable encrypted methods to be executed and data to be accessed and manipulated.

2 Encrypted methods and Surveillance Agents

When an agent starts to execute in a host it is possible that its data and actions are spied. Also it is possible that the host interferes and manipulates the agents flow of execution and data. Let us consider the case where an agent is shopping for a certain asset in the network. A malicious host may try to read and modify offers from other hosts or even change the flow of execution of the agent in order to force it to take an offer being proposed [2].

Because agents execute on a target platform and have to expose both their code and state, it does not seem possible to fully protect agents from malicious hosts. Even so, because of the characteristics of e-commerce systems, it is possible to guarantee some degree of safety for the agents.

An electronic commerce transaction may be viewed in terms of four different phases [1]:

(i) Product brokering
(ii) Merchant brokering
(iii) Negotiation
(iv) Payment and delivery

Product brokering consists in the gathering of information about the product that is going to be bought.

Merchant brokering involves the evaluation of a set of alternatives in order to make the purchase. Making the decision implies considering all the tradeoffs that the various products offer: price, warranties, delivery-time, and others.

During the negotiation phase the agent settles the final terms of the commercial transaction. The characteristics of the market directly influence the outline of this phase. In markets where prices and characteristics are fixed, negotiation may not even exist.

* This research was partially supported by the Portuguese Research Agency FCT, through the program PRAXIS XXI (scholarship number DB/18353/98).

Finally, in the purchase and delivery phase of the transaction, the agent actually makes the acquisition and delivers the money (or its electronic equivalent) against the goods.

When we consider the security requirements in terms of protecting agents from malicious hosts, different needs exist during the different stages of the electronic transaction.

In our security framework there are two kinds of agents: worker agents and surveillance agents. Worker agents are agents that migrate to hosts doing work in order to solve a problem. Some of their methods and data are encrypted and require keys to be accessed. Surveillance agents are light-weighted agents that have the keys needed by the worker agents. They are used to assess trust on hosts and distribute keys to the agents operating on trusted hosts. The code of the surveillance agents implements the trust policy that the user wants to use to allow the methods of the worker agents to execute on a given host.

Let us assume that the agent's itinerary in not known when it first migrates. The application at user's home deploys surveillance agents to known trusted hosts. These agents carry private symmetric encryption keys that grant access to the worker agent methods and data. These keys are only valid during one agent run (i.e., after the agent completes its task and returns home, they are no longer valid) and cannot be used again. Finally, the worker agent is informed of where the surveillance agents are located.

The worker agent now migrates to the first host where it has work to do. Since this is an untrusted host, the agent can only use non-sensitive methods and data. All the results produced at that host are securely stored in a folder of the agent's briefcase. The briefcase if made non-tamperable by using Karjoth's algorithms [4].

After a certain number of migrations, the agent might have to execute a task that requires the use of a sensitive method or information. At this point the trust manager of the agent contacts one of the surveillance agents for a key that can be used to access that data or method. The surveillance agent, which is stationary at a trusted host, decides if the host where the agent is can be trusted or not. If it can, then it sends the key that it is used to decrypt the methods or data needed by the agent. We assume that the surveillance agent can authenticate the host by using a public-key directory infrastructure.

In the simplest case, trusted hosts are given by a list of hosts that the owner of the agent trusts. If the agent is currently at one of those hosts, it has the permissions to access the keys. Nevertheless, trust assessment is still a difficult issue with no general solution known. History-based approaches are a promising way of assessing trust. Either way, the surveillance agents can transparently incorporate any trust assessment policy that the owner of the agent decides on.

Since trust assessment is done outside the untrusted host and surveillance agents can be used for accounting purposes, it is even possible to negotiate at run-time contractual settlements that allow agents on untrusted hosts to executed privileged operations.

One important issue is that the communication channels used between the trusted host and the host where the agent is must be secure. If not, third parties may intercept messages obtaining keys that are used by the agent.

At the end, after completing its job, the agent returns home. The surveillance agents are signaled to also return home. There, they can be checked for accounting purposes. It is possible to know exactly what data and methods hosts had access to. If sensitive information was disclosed at a host, worker agents had to communicate with surveillance agents. Thus, the surveillance agents also serve for auditing purposes. They can also be used for trust assessment by helping the user to maintain behavioral history of hosts [5].

3 Conclusion

The present framework allows the secure execution of agents that do not have to perform sequential computations on untrusted hosts. The major limitation of the approach is the assumption that the computations have to be independent of previous collected data if the host cannot be trusted. However, most open-network applications seem to fall under this paradigm.

References

[1] P. Maes, R. Guttman, and A. Moukas, "Agents that Buy and Sell," in *Communications of the ACM*, vol. 42, 1999.

[2] F. Hohl, "A Model of Attacks of Malicious Hosts Against Mobile Agents," presented at 4th Workshop on Mobile Object Systems: Secure Internet Mobile Computations, France, 1998.

[3] M. Greenberg, J. Byington, and D. Harper, "Mobile Agents and Security," *IEEE Communications Magazine*, 1998.

[4] G. Karjoth, N. Asokan, and C. Gülcü, "Protecting the Computation Results of Free-roaming Agents," presented at Second International Workshop on Mobile Agents (MA' 98), Stuttgart, Germany, 1998.

[5] G. Edjlali, A. Acharya, and V. Chaudhary, "History-based Access Control for Mobile Code," presented at Fifth ACM Conference on Computer and Communications Security, San Francisco, 1998.