Host Revocation Authority: A Way of Protecting Mobile Agents from Malicious Hosts

Oscar Esparza, Miguel Soriano, Jose L. Muñoz, and Jordi Forné

Technical University of Catalonia C/ Jordi Girona 1 i 3, Campus Nord, Mod C3, UPC 08034, Barcelona (Spain)
{oscar.esparza, soriano, jose.munoz, jforne}@entel.upc.es

Abstract. Mobile agents are software entities that consist of code, data and state, and that can migrate autonomously from host to host executing their code. Despite its benefits, security issues restrict the use of code mobility. The approach that is presented here aids to solve the problem of malicious hosts by using a Trusted Third Party, the Host Revocation Authority. The HoRA controls which are the hosts that acted maliciously in the past. The agent sender must consult the HoRA before sending an agent in order to remove from the agent's itinerary all the revoked hosts. The HoRA can also revoke a malicious host if the agent sender detects and proves that this malicious host did not act honestly.

1 Introduction

Mobile agents can migrate from host to host performing actions autonomously on behalf of a user. Despite their benefits, massive use of mobile agents is restricted by security issues. This paper introduces a new approach that aids to solve the problem of malicious hosts by using a Host Revocation Authority. The HoRA must be considered an independent Trusted Third Party (TTP) in a mobile agent system. The HoRA stores the revoked host identifiers, i.e. the identifiers of those hosts that have proven to be malicious. Before sending an agent, origin hosts must ask to the HoRA if the hosts in the agent's itinerary have been revoked. All those revoked hosts must be deleted from the agent's itinerary. An origin host can also try to revoke a host by demonstrating to the HoRA that it acted maliciously. In this sense, detection and proving of attacks are needed [5,1].

The rest of the paper is organized as follows: Section 2 presents some published approaches to solve the problem of the malicious hosts; Section 3 details how the HoRA works and finally, some conclusions can be found in Section 4.

2 Malicious Hosts

The problem of malicious hosts is by far the most difficult to solve regarding mobile agent security. Malicious hosts could try to get some profit of the agent modifying the code, the data, the communications or even the results due to their complete control on the execution. Current approaches can be divided in two categories: attack detection and attack avoidance.

J.M. Cueva Lovelle et al. (Eds.): ICWE 2003, LNCS 2722, pp. 289–292, 2003.

2.1 Attack Detection Approaches

This approaches need a TTP to punish malicious hosts in case of detection. This paper tries to solve this lack by adding the HoRA to the mobile agent system.

In [5], Vigna introduces the idea of cryptographic traces. The running agent takes traces of instructions that alter the agent's state due to external variables. If the agent owner wants to verify execution, it asks for the traces and executes the agent again. If new execution does not agree with the traces, the host is cheating. This approach has various drawbacks: (1) Verification is only performed in case of suspicion, but the way in which a host becomes suspicious is not explained; (2) Each host must store the traces for an indefinite period of time because the origin host can ask for them.

In [1] the authors present a protocol for detecting suspicious hosts by limiting the execution time of the agent. Each host saves the agent's arrival and leaving time. When the agent reaches the origin host, a set of time checks verifies if each host in the agent's itinerary spent more time than expected executing the agent. If so, the host is considered suspicious of malicious behavior.

2.2 Attack Avoidance Approaches

Detection techniques are not useful for services where benefits for tampering a mobile agent are greater than the possible punishment in case of detection, so attack avoidance techniques might be used. Unfortunately, there are no approaches that avoid attacks completely.

The environmental key generation [3] makes the agent's code impossible to decipher until the proper conditions happen on the environment, so previous analysis from hosts is avoided. The main drawback of this proposal is that hosts can perform a dictionary attack lying about the environment.

The Time Limited Blackbox [2] uses obfuscation as a way to hide the mobile agent's code and data to a malicious host. Protection is only assured for a period of time that depends on the computation capacity of the host. The scheme also needs a great amount of resources as the obfuscated code length is substantially greater than plain code.

The use of encrypted programs [4] is proposed as the only way to give privacy and integrity to mobile code. Hosts execute the encrypted code directly. Results are decrypted when the agent reaches the origin host. The difficulty here is to find functions that can be executed in an encrypted way.

3 Host Revocation Authority

This paper introduces a new entity in the mobile agent system, the Host Revocation Authority. The HoRA controls which hosts have been revoked in the mobile agent system, i.e. host that have proven to be malicious. The HoRA must be considered an independent TTP like the Certification Authority is considered in the PKI. The approach that is presented here can be considered neither a detection approach nor an avoidance approach, but a combination of them. First

attack performed by a host cannot be avoided, but if the agent sender can prove that the host acted maliciously, this host can be revoked. Therefore, any other attack from this malicious host can be avoided.

The rest of the section explains the tasks that the HoRA must perform:

3.1 Keeping the Revocation Information

The aim of host revocation is to distinguish the malicious hosts from the honest ones. Unfortunately, it is not possible to know if a honest host can turn into malicious behavior just in the current transaction. However, it is possible to know if a host acted maliciously in the past. The HoRA knows which hosts have been revoked by saving their host identifiers in a list. Host identifiers must be unique in the mobile agent system, for instance IP addresses or DNS names can be used.

3.2 Revoking Malicious Hosts

Before an origin host starts a host revocation process, one of any existing detection and proving approaches must be used. For instance, the cryptographic traces approach [5] is widely known, but it has two major drawbacks: (1) How a host becomes suspicious and; (2) Each host must store the traces for an indefinite period of time. These drawbacks can be solved by using suspicious detection techniques [1]. Using jointly both mechanisms it is possible to detect suspicious hosts and to ask for the traces just when the agent returns to the origin host.

If there is a way to have proofs that demonstrate that a host did not execute an agent properly, the origin host can start the revocation process by sending to the HoRA this proofs. The HoRA receives the proofs and verifies the execution integrity. If finally the proofs are considered valid, the HoRA adds the malicious host's identifier in the list of revoked hosts.

The rest of the tasks depends on the revocation policy. Assuming that the HoRA works in a similar way as the Certification Authority regarding certificate revocation, two possible revocation policies can be followed.

3.3 Offline Revocation Policy: Generating the HRL

The off-line revocation policy is based on the distribution of revocation information using a Host Revocation List (HRL), i.e. a list of revoked host identifiers signed by the HoRA. Origin hosts must download a copy of the HRL in order to consult it before executing an agent. Origin hosts must also update the list periodically to take into account new malicious hosts. Generating the HRL is as easy as signing the list that the HoRA has internally. As it is a signed list, it can also be downloaded from non trusted repositories. In this sense, the HRL works in a similar way as the Certificate Revocation List in the PKI.

3.4 Online Revocation Policy: Receiving and Replying to Requests

The on-line revocation policy is based on asking the revocation information to the HoRA directly. Before sending a mobile agent, each origin host sends a request to the HoRA asking for the status of the hosts in the agent's itinerary. The HoRA consults if these hosts are included in its internal list, and it sends a signed response to the origin host pointing out which hosts in the agent's itinerary have been revoked. This mechanism works in a similar way as the Online Certificate Status Protocol used in the PKI.

3.5 Improvements

The following improvements can be achieved in the HoRA:

- The list that the HoRA has internally grows in an indefinite way. This problem can be solved by using an Agent Execution Certificate, i.e. a certificate issued by the HoRA that permits the hosts to execute agents during a validity period. In this case, the HoRA does not revoke the host identifier, but the certificate.
- The HoRA must be accessible for all hosts. An alternative topology based on repositories and a replication policy between entities must be thought.

4 Conclusions

The approach that is presented here aids to solve the problem of malicious hosts by using a TTP, the Host Revocation Authority. The HoRA controls which are the hosts that acted maliciously in the past. Each agent sender must consult the HoRA before sending a mobile agent in order to remove from the agent's itinerary all the revoked hosts. The HoRA can also revoke a malicious host if the agent sender proves that this malicious host did not act honestly.

Acknowledgments. This work is supported by the Spanish Research Council under the project DISQET CICYT TIC2002-00818.

References

1. O. Esparza, M. Soriano, J.L. Muñoz, and J. Forné. Limiting the execution time in a host: a way of protecting mobile agents. In *IEEE Sarnoff Symposium "Advances in Wired and Wireless Communications"*, 2003.
2. F. Hohl. Time Limited Blackbox Security: Protecting Mobile Agents From Malicious Hosts. In *Mobile Agents and Security*, volume 1419 of *LNCS*. Springer-Verlag, 1998.
3. J. Riordan and B. Schneier. Environmental Key Generation Towards Clueless Agents. In *Mobile Agents and Security*, volume 1419 of *LNCS*. Springer-Verlag, 1998.
4. T. Sander and C.F. Tschudin. Protecting mobile agents against malicious hosts. In *Mobile Agents and Security*, volume 1419 of *LNCS*. Springer-Verlag, 1998.
5. G. Vigna. Cryptographic traces for mobile agents. In *Mobile Agents and Security*, volume 1419 of *LNCS*. Springer-Verlag, 1998.