Author(s): Melomey, Divina; Mouratidis, Haralambos.
Article title: Evaluating the Security of Mobile Agent Platforms
Year of publication: 2006
Citation: Melomey, D. and Mouratidis, H. (2006) 'Evaluating the Security of Mobile Agent Platforms' Proceedings of the AC&T, pp.48-54.
Link to published version:
http://www.uel.ac.uk/act/proceedings/documents/ACT06Proceeding.pdf

EVALUATING THE SECURITY OF MOBILE AGENT PLATFORMS

Divina Melomey, Haralambos Mouratidis
Innovative Informatics Research Group
dmelomey@yahoo.com, haris@uel.ac.uk

Abstract: Mobile agents are software entities that can migrate autonomously throughout a network from host to host. This means they are not bounded to the platform they begin execution. This feature of agents makes them a very attractive technology, and in fact it has been argued many times in the literature that mobile agents help to reduce network traffic and perform tasks more efficient. However, security issues have not yet been fully investigated and in fact, mobile agent platforms sometimes they neglect the security issues involved with agent mobility. This paper presents a security related evaluation of 8 main mobile agent platforms.

1. Introduction

Developing complex computerised systems has proved to be a difficult task. Actually, it has been argued that developing software for domains like telecommunications represents one of the most complex tasks humans undertake.

Agent technology introduces an alternative approach in developing complex computerised systems. According to this, a complex computerised system is viewed as a multi-agent system in which autonomous software agents (subsystems) interact with each other in order to satisfy their design objectives. Such approach provides designers with more flexibility in their development. The actual design of the system takes place by specifying a multi-agent system as a society, similar to a human society, consisting of entities that possess characteristics similar to humans such as mobility, and intelligence with the capability of communicating.

The concept of a software agent, however, is not uniquely defined. Researchers have given definitions of the concept according to some typical characteristics, some operational characteristics or some cognitive functions that agents should implement.

One of the most promising features of software agents is mobility. Mobile agents are software entities that can migrate autonomously throughout a network from host to host. This means they are not bounded to the platform they begin execution. However, this feature of agents although makes them a very attractive technology, it also makes the development of platforms (known also as frameworks and environments) that will support mobile agent systems very challenging. One of the main challenges is to develop platforms which will allow a secure migration of mobile agents. Many issues are involved, with respect to security, such as securing the mobile agent from a malicious platform, security the platform from malicious agents and so on.

Although, many different platforms have been proposed by researchers, we believe that security, unlike some other non functional requirements such as performance, has not really thought of during the development of these platforms.

This paper indicates the results of an evaluation, with respect to security, of 8 major agent platforms. Our findings justify the above claim regarding the lack of adequate security mechanisms of these platforms. Section 2 presents a brief introduction to mobile agent migration, whereas Section 3 discusses the security implication of mobile agent systems. Section 4 discusses the evaluation and section 5 concludes the paper and presents ideas for future work.

2. Mobile Agent Migration

A mobile agent is made up of code and state information, which is needed to perform some form of computation (Jansen and Karrygianis, 1999). Therefore, for a mobile agent to execute, an agent platform is required, which is made up of the computational environment.
A mobile agent is characterized by its ability to migrate, during execution, from one host to another as well as between different platforms; even these are running in the same host (see Figure 1 for a partial graphical representation).

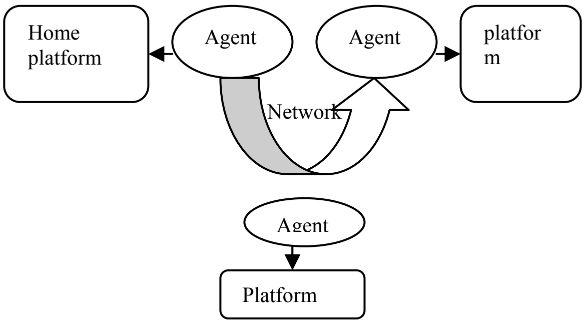

Figure 1. A mobile agent system (Jansen and Karrygianis, 1999)

A mobile agent either performs a hop or a multi-hop. A hop is defined as the movement of an agent from its home platform to another platform. Similarly, a mobile agent is said to multi-hop when it hops through various platforms.

3. Security Implications in Mobile Agent Systems

Security threats in mobile agent systems can be categorised into four main categories (Jansen and Karrygianis, 1999): (a) Agent to agent attack, when a malicious agent attacks another agent; (b) Agent to platform, when an agent attacks a platform; (c) Platform to an agent, when a platform launches an attack on an agent; (d) External to an agent, when other (non agent) entities attack an agent.

3.1 Agent to agent

This is usually in the form of (i) *masquerade*, in which one agent assumes the identity of another to deceive an unsuspecting agent and gain access to sensitive information; (ii) *denial of services to another agent*, which is usually in the form of spam messages sent repeatedly to an agent in order to consume its resources; (iii) *unauthorized access*, where an agent interferes directly with another agent by the invocation of its public methods if the agent's home platform has no control mechanism in place; (iv) *repudiation,* which occurs when an agent denies participation on a transaction; (v) *eavesdropping,* where an agent can gain access to information about other agents' activities, by using services provided by the platform.

3.2 Agent to Platform

This is usually in the form of (i) masquerade where an agent tries to gain access on a platform by assuming the identity of another agent; (ii) Denial of Service, in which an agent disallows access to services on the agent's platform;(iii) unauthorized access, in which an agent gains unauthorised access to a platform and is capable of causing harm to that platform.

3.3 Platform to an agent

This is usually in the following forms: (i) masquerade, where a platform can assume the identity of another platform in an attempt to deceive another agent with regards to an intended destination as well as its security policy; (ii) denial of service, where a platform ignores service request or may terminate request without notification; (iii) eavesdropping, when confidential and sensitive information is monitored and interpreted by agent platform; (iv) alteration, when an agent arrives at the platform and exposes its code, state and data to the platform. A malicious platform will attempt to modify the code, state and data of the visiting agent unknowingly to the agent. This alters the integrity of the agent.

3.4 Other to agent

This occurs in the following ways: (i) masquerade, where an agent makes a request from a platform either remotely or locally. An agent or a remote platform can assume the identity of another to get unauthorized access to resources to which it is not entitled to; (ii) denial of Service, where an entity can access agent platforms server either remotely or locally where an agent with malicious intent can interfere with services that are offered by the platform and inter-platform communication; (iii) unauthorised access ; If remote access to the platform is not properly secured or protected, entities can get access easily and free through scripts available on the internet that can be used to subvert operating system in order to gain control of all systems resources; (iv) Copy and replay; when a mobile agent migrate from one host to the other, it exposes itself to security threat, the message it is migrating with can be intercepted and replay or clone for retransmission [8].

Figure 2 provides a summary of threat per each category.

Threats	Agent to Agent	Agent to Platform	Platform to Agent	Other Entities to Platform
Masquerade	X	X	X	X
Denial of Service	X	X	X	X
Unauthorized access	X	X		X
Eavesdropping	X		X	
Alteration			X	

Figure 2. Threats per category

3.5 Security requirements

In general, mobile agent systems have the same requirements as general computer systems. These requirements as suggested in (Jansen and Karrygianis, 1999) are:

1. Confidentiality; any data that is stored privately on a platform or carried by an agent should remain confidential. Intra platform and inter platform communication

must also remain confidential and must be ensured by agent framework,

2. Integrity; ensuring that there is no unauthorized modification of the agent framework.

3. Availability; Data and services to both local and remote agents must be made available by the agent platform. Data that is shared must be available in a form that can be used as well as capacity to handle availability of large volumes of request by visiting platform and remote agent.

4. Anonymity; that there should be a balance between the needs of an agent for privacy with the needs of an agent for platform to hold an agent accountable for their actions.

5. Accountability; .all actions must be accountable for by the agent i.e. all processes, operations, meetings of an agent on any given platform. Accountability is necessary for building trust among agent platforms and agent. Audit logs are invaluable source for platform recovery of security breach.

4. Platform Evaluation

Literature provides a wide range of available agent platforms (Melomey, 2005). For the purpose of our evaluation we have identified some of these platforms, which we think are the most appropriate for our research. The selection was based on the following criteria:

- It supports mobility. A basic requirement for any mobile agent infrastructure is its ability to migrate autonomously from one computer or host computer to the other. First, agent should be able to migrate with its entire codes as it goes along and be able to run on any server. Secondly, some servers only require a pre installation of agents' code; such servers do not need transfer of codes to resume execution. Lastly, with some servers, no code is carried by the agent but rather contains a reference to its code base.
- It should be free to use and active. All platforms chosen for this evaluation are available for free download. Moreover, the project is still active meaning the platform is supported either by the developers or from a user group.
- It is written in a language that is widely known with preference to java and scripting language. All the platforms for this evaluation are written in java except for Telescript which uses the scripting language but it compatible with java platforms and also widely known.

Following the above criteria, we have identified the following platforms for our evaluation: Ajanta, Aglet, Voyager Concordia, Telescript , Agent Tcl, Tacoma , and JADE

4.1 Criteria for assessment

Criteria for performing evaluation of the selected platforms have been developed based on the security countermeasures and requirement of mobile agent platforms. In total, forty-one criteria were identified. However, due to lack of space we focus on six of them1.

Criterion 1: Audit Log for the platform should trace agent falsely repudiating an action.

Criterion 2: Safe code interpreter should evaluate all codes.

Criterion 3: Agents should be held accountable for their action by using audit trails

Criterion 4: The agents function should be encrypted.

Criterion 5: support of fault tolerance mechanisms.

[1] Please refer to (Melomey,2005) for the complete list of criteria

Criterion 6: Support for authentication and access lists when authorised agents join a transaction

The following table indicates the evaluation of the platforms with relation to the above criteria.

1	2	3	4
NOT SUPPORTED	POORLY SUPPORTED	ADEQUATELY SUPPORTED	FULLY SUPPORTED

CRITERION	PLATFORMS							
	1	2	3	4	5	6	7	8
Criterion 1	1	1	1	1	1	1	1	1
Criterion 2	4	4	1	1	1	1	1	4
Criterion 3	3	1	3	3	1	3	3	3
Criterion 4	4	4	1	2	4	1	1	2
Criterion 5	4	1	4	4	1	4	1	4
Criterion 6	3	1	3	4	4	1	3	4

Key
Platform 1 AJANTA, Platform 2 AGENT TCL
Platform 3 VOYAGER, Platform 4 CONCORDIA
Platform 5 TELESCRIPT, Platform 6 TACOMA
Platform 7 AGLETS, Platform 8 JADE

Table 1. The evaluation Table

4.2 Discussion about the evaluation

Criterion 1: All platforms except Agent Tcl and Telescript provide adequate support. JADE provides full support, mainly because it is based on FIPA specification. Under FIPA 98 specification, an automated mechanism is used to record platform activities in an audit log which is protected. This takes place in order to maintain accountability at platform level, especially with regards to repudiation.

Criterion 2: Fully supported by Ajanta, Agent Tcl and JADE. Ajanta provides (or loads) code on demand from a specified agent server. Moreover, agents execute a protected domain that is isolated, in order to prevent agent interference. The function of the safe code interpreter is to execute commands requiring access to system resources. JADE, Aglet, Voyager and Concordia use byte code for verification (Kadhi and Boury, 2001), whereas Agent Tcl uses safe code and Tacoma uses firewalls.

Criterion 3: Adequately supported by Ajanta, Voyager, Concordia, Tacoma, Aglet and JADE. Ajanta's full support is based on the fact that the audit trail should indicate the host identity and that f the next (host) as well as its (agent) intended destination. Concordia, Aglets and JADE check if the previous host is a trusted one, whereas Ajanta poorly supports this (Karnik andTripathi, 1998).

Criterion 4: With encrypted functions, the host must have full control over the mobile code by encrypting it using some agreed conversation algorithms. Ajanta and Telescript fully support this, whereas Concordia provides adequate support.

Criterion 5: To avoid tampering and ensure that a code reaches its destination, a Fault Tolerance Mechanism is used. This mechanism when in place helps to achieve replication and voting. Voyager, Tacoma and Concordia fully support this feature, whereas JADE (Andrei, 2002) provides adequate support. If an exception is encountered that it cannot be handled, the system's server can take appropriate actions to assist that specific application to recover. Moreover, it should be able to determine the cause of the crash. For this reason, Ajanta supports itinerary abstraction.

Criterion 6: Fully supported by Ajanta and adequately supported by Agent Tcl, Concordia and JADE. JADE achieves this on its runtime environment by enforcing the use of authentication and access lists when joining a transaction. On the other hand, Agent Tcl uses safe Tcl in enforcing access restriction based on its authenticated identity.

The results of the evaluation were analysed graphically and tabulated. Although more than one platforms demonstrated adequate support for most of the evaluation criteria, our analysis of the evaluation demonstrated that JADE offers the best support for security amongst all the platforms, followed closely by Aglets and Agent Tcl (Melomey,2005). Figure 3 illustrates a comparison of different platforms against the full set of forty-one criteria

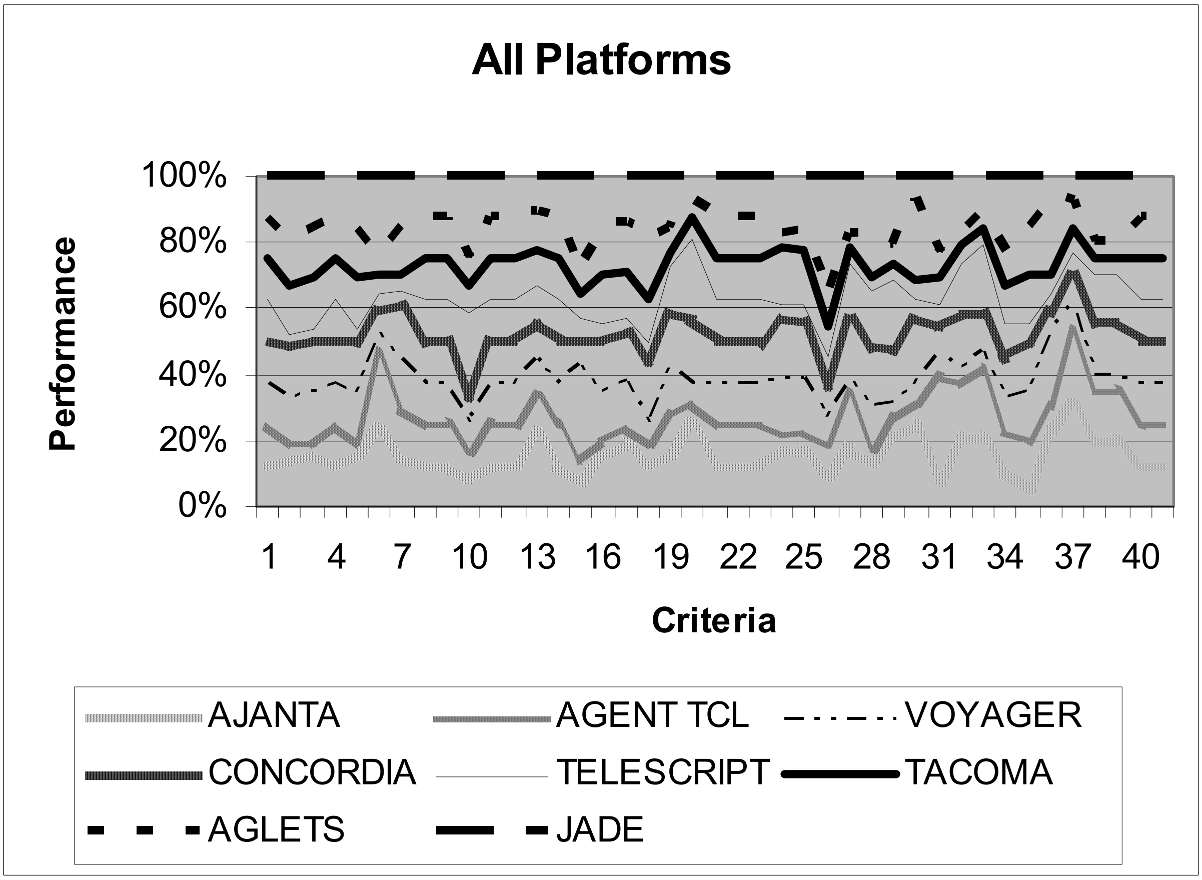

Figure 3. The comparison

5. CONCLUSIONS

Security threats to mobile agents have been explored in this paper. A list of evaluation criteria were illustrated together with an evaluation of 8 main mobile agent platforms against those criteria. The presented set of criteria was derived by considering general security requirement of networked systems as well as special implications of mobile agent systems. The chosen platforms went through the evaluation process and the values assigned were justified on the basis of their ability to meet the requirement in the following order; not supported, poorly supported, adequately supported and fully supported.

Our work is not complete. Future work involves expanding our evaluation criteria to include more specialised criteria, and the development of more experiments in order to validate from an implementation point of view our results.

5. References

Dancus Andrei, “JADE- A FIPA compliant Java Agent Development Framework” Spring 2002, Worcester Polytechnic Institute, Spring 2002

FIPA Agent Management Specification, 2004/18/03

Jansen,Wayne & Karygiannis,Tom(1999) “Mobile Agent Security”, National Institute of Standards and Technology (NIST) special publication 800-19, October, 1999

Kadhi N., Boury P., “Statistic Analysis of Java Cryptography Applets”. In proceeding of ECOOP2001 (Budapest) Workshop on Java Formal Verification, 2001

Karnik N. and Tripathi A., ”design Issues in Mobile Programming Systems2 IEEE Concurrency 1092-3063, 1998

Melomey, Divina MSc Thesis, “ Evaluating the Security of Agent Platforms” University of East London, 2005