

Security in Multi-Agent Systems

Niklas Borselius
 Information Security Group
 Royal Holloway, University of London
 Egham, Surrey, TW20 0EX, UK
 Niklas.Borselius@rhul.ac.uk

Abstract— The agent paradigm appears promising and much research has been devoted to it during the last decade. In this paper we consider security issues that need to be addressed before multi-agent systems can be a viable solution for a broad range of commercial applications. We do this through considering the implications of the characteristics given to agents and general properties of open multi-agent systems. We then look at some past and present work that addresses security issues of multi-agent systems. Finally, we consider how existing security technology can be used to address the security issues, and where the gaps are most likely to appear.

Keywords— security, multi-agent system, mobile-agent.

I. INTRODUCTION

The concept of an agent originates from the area of Artificial Intelligence (AI) but has now gained more widespread acceptance in mainstream computer science. The agent paradigm has in many ways become a buzzword, and a more mature technology than currently available is often implied. This is in particular true for security in multi-agent systems. Over-simplified assumptions and non-applicable references to security solutions are not uncommon in the literature. Naturally, security is not a driving force for research and development of multi-agent systems, and therefore has not received much attention from the agent community. Nevertheless, in order for agent technology to gain widespread use and provide viable solutions on a wider scale for commercial applications, security issues need to be properly addressed.

Autonomous agents and multi-agent systems represent a relatively new way of analysing, designing, and implementing complex software systems. In this paper we are only concerned with the security of the system and its components (leaving design methodologies to others). Several multi-agent systems are available as commercial products and many more have been implemented in various research projects, with varying success. Recent standardisation efforts [1], [2] have proven rather successful. Today there is a growing interest and research in implementing and rolling out (open) multi-agent systems on a wider scale¹.

The remainder of this paper is organised as follows. Section 2 briefly describes characteristics of agents and multi-agent systems of relevance to security. In section 3 we will discuss some security implications of these characteristics. Section 4 considers the current status of security in multi-

agent systems including standardisation efforts in the area. In our conclusions we consider how current security technology can be used to address the security issues and identify gaps where further research will be required. Finally, we describe planned future work based on the analysis presented in this paper.

II. AGENTS AND MULTI-AGENT SYSTEMS

In this section we will briefly describe some properties of agents and multi-agent systems (MAS). This is not intended to be a complete description of agents or MAS (we are, for example, not concerned with AI properties for agents here). We try to focus on issues with possible security implications.

Agents are software entities that have some kind of autonomy and certain ‘intelligence’. An agent is often assumed to represent another entity, such as a human or an organisation on whose behalf it is acting. No single universal definition of agents exists, but there are certain widely agreed universal characteristics of agents, include situatedness, autonomy, and flexibility [3].

- **Situatedness** means that the agent receives sensory input from its environment and that it can perform actions which change the environment in some way.
- **Autonomy** means that agents are able to act without the direct intervention of humans (or other agents), and that it has control over its own actions and internal state.
- **Flexibility** can be defined to include the following properties:
 - **responsive**: agents’ ability to perceive their environment and respond in a timely fashion to changes that occur in it;
 - **pro-active**: agents are able to exhibit opportunistic, goal-driven behaviour and take the initiative where appropriate;
 - **social**: agents should be able to interact, when appropriate, with other agents and humans in order to complete their own problem solving and to help others with their activities.

A number of other attributes are sometimes discussed in the context of agency. These include but are not limited to [4]:

- **Mobility**: the ability for an agent to move across networks and between different hosts to fulfil its goals [5].
- **Rationality**: the assumption that an agent will not act in a manner that prevents it from achieving its goals and will always attempt to fulfil those goals [6].

¹Examples of ongoing efforts to implement large scale multi-agent systems can be found at <http://www.agentcities.org> and <http://www.mobilevce.com>.

- **Veracity:** the concept that an agent will not knowingly communicate false information [6].
- **Benevolence:** an agent cannot have conflicting goals that either force it to transmit false information or to effect actions that cause its goals to be unfulfilled or impeded [7]. A multi-agent System (MAS) is a system composed of multiple autonomous agents showing the following characteristics [3]:
 - each agent cannot solve a problem unaided,
 - there is no global system control,
 - data is decentralised, and
 - computation is asynchronous.

In order for agents to be able to form a useful open multi-agent system where they can communicate and cooperate, certain functionality need to be provided to the agents. This includes functionality to find other agents or find particular services. This can be implemented as services offered by other agents or service more integrated with the MAS infrastructure itself. Examples of such services include facilitators [8], matchmakers [9], mediators [10], and blackboards [11].

Open multi-agent systems are usually envisioned as systems, communicating over the Internet, allowing anybody to connect a platform on which agents are running. This means that the MAS lacks a global system control and that information in general is highly decentralised.

III. SECURITY IMPLICATIONS

In this section we will discuss security implications based on the characteristics described in the previous section.

A. *Situatedness*

Naturally agents need to ‘exist’ somewhere. A computer host, the immediate environment of an agent, is ultimately responsible for the correct execution and protection of the agent. This leads us to the question of where access control decisions should be performed and enforced. Does the agent contain all necessary logic and information required to decide if an incoming request is authentic (originating from its claimant) and if so, is it authorised (has the right to access the requested information or service)? Or can the agent rely on the platform for access control services?

The meaning of the term ‘environment’ appears to be somewhat arbitrary in the agent literature; it could for example be the Internet or the host on which the agent is executing. An agent is assumed to be ‘aware’ of certain states or events in its environment. Depending on the nature and origin of this information, its authenticity and availability need to be considered; (confidentiality of such information might also be relevant). If an agent’s ‘environment’ is limited to the host on which it is executing, no specific security measures might be necessary (assuming the host environment cannot be spoofed). The situation is however likely to be different if the agent receives environment information from, or via, the Internet. (Security of communication is further explored below.)

The environment might also need certain protection from the agents that it hosts. An agent should, for example, be

prevented from consuming all resources on a host, thus preventing the host from carrying out other things (such as executing other agents). Security issues related to the executing host becomes even more apparent for agents that are mobile, further described in section III-D.

B. *Autonomy*

Autonomy, when combined with other features given to agents, can introduce serious security concerns. If an agent, for example, is given authority to buy or sell things, it should not be possible for another party to force the agent into committing to something it would not normally commit to. Neither should an agent be able to make commitments it cannot fulfil. Hence, issues related to delegation needs to be considered for agents.

The autonomy property does not necessarily introduce any ‘new’ security concerns; this property is held by many existing systems. It is worth mentioning that Internet worms (often referred to as viruses) also hold this property, which enables them to spread efficiently without requiring any (intentional or unintentional) human interaction. The lesson to learn from this is that powerful features can also be used for malicious purposes if not properly controlled.

C. *Communication*

Of the flexibility properties, the social behaviour is certainly interesting from a security point of view. This means that agents can communicate with other agents and humans. Just as an agent’s communication with its environment needs to be protected, so does its communication with other agents and humans. The following security properties should be provided:

- **confidentiality:** assurance that communicated information is not accessible to unauthorised parties;
- **data integrity:** assurance that communicated information cannot be manipulated by unauthorised parties without being detected;
- **authentication of origin:** assurance that communication originates from its claimant;
- **availability:** assurance that communication reaches its intended recipient in a timely fashion;
- **non-repudiation:** assurance that the originating entity can be held responsible for its communications.

Fundamental to the above mentioned communication security properties are issues relating to the identification and authentication of the sending and receiving parties. These issues are further discussed in section III-F.

It should be noted that security usually comes at a cost. Additional computing resources as well as communication resources are required by most solutions to the above mentioned security functionality. Therefore, security needs to be dynamic. Sometimes it makes sense to protect all communication within a system to the same degree, as the actual negotiation of security mechanisms then can be avoided. However, in a large scale open multi-agent system, security services and mechanisms need to be able to fit the purpose and nature of the communications of various applications with different security requirements.

Some implementations of MAS assume that security is provided by a lower layer. This approach might be sufficient in a closed system where the agents can trust each other and the only concern is external malicious parties. In an open system we believe that agents need to be ‘security aware’, i.e. they need to be able to make decisions based on where information is originating from and how well protected it is.

D. Mobility

The use of mobile agents raises a number of security concerns. Agents need protection from other agents and from the hosts on which they execute. Similarly, hosts need to be protected from agents and from other parties, that can communicate with the platform. The problems associated with the protection of hosts from malicious code are quite well understood.

The problem posed by malicious hosts to agents seems the hardest to solve. In fact some people hold the opinion that it is insoluble. The particular attacks that a malicious host can make have been described in [12] and [13], and can be summarised as follows.

- Observation of code, data and flow control,
- Manipulation of code, data and flow control – including manipulating the route of an agent,
- Incorrect execution of code – including re-execution,
- Denial of execution – either in part or whole,
- Masquerading as a different host,
- Eavesdropping on agent communications,
- Manipulation of agent communications,
- False system call return values.

E. Rationality, veracity, and benevolence

These properties could at a first glance appear to be very security relevant. However, on closer consideration they seem to be too abstract for us to consider as practical security concerns. The meaning (from a security point of view) of these properties seems to be: “Agents are well behaved and will never act in a malicious manner.” If we make this a genuine requirement, then the required redundancy for such a system is likely to make the system useless. It would, of course, be valuable to have a system where agents can be assumed to behave truthfully and honestly in every situation. However, this does not seem a likely scenario for a multi-agent system that is not under very strict control and under a single authority, which does not correspond to the assumed open system scenario.

F. Identification and authentication

Identification is not primarily a security issue in itself; however, the means by which an agent is identified are likely to affect the way an agent can be authenticated. For example, an agent could simply be identified by something like a serial number, or its identity could be associated with its origin, owner, capabilities, or privileges. As mentioned in section III-C, authentication is often fundamental to secure communication. FIPA (www.fipa.org) does not specify how agents are identified, allowing for ‘translation’

of agent identities when communication is taking place between different platforms. If identities are not permanent, security related decisions cannot be made on the basis of an agent’s identity.

Connected with identification and authentication is anonymity. While an entity’s identity is of major importance to certain applications and services, it is not needed in others. A certain degree of user anonymity is, for example, considered an important feature of GSM. A multi-agent system would probably require some sort of anonymity service to acquire great acceptance today.

G. Authorisation and delegation

Authorisation and delegation issues appear to be a major concern in multi-agent systems. Not only do agents need to be granted rights to access information and other resources in order to carry out its tasks, they will also be acting on humans’ behalf or on the behalf of other agents, requiring transfer of access rights between different entities.

IV. RELATED WORK

Many commercial and research MAS architectures have been implemented and many are still under development². Several of these recognise security as an issue to be taken care of in the future (e.g. [14]), while other imply that security is provided for. In this section we will briefly look at how security currently is being tackled for multi-agent systems.

It is common for MAS implementations to assume a VPN-like (Virtual Private Network) underlying network to provide security services. This approach usually does not provide for much flexibility, since secure communication between parties without pre-established relationships becomes cumbersome. Nevertheless, this solution can use well established security protocols and be adequate for applications where all communication is protected to the same degree. Such an approach usually leaves the agents completely unaware of security services as this is handled between agent platforms (or perhaps even on link level).

FIPA is a non-profit standards organisation that is developing standards for software agents to allow heterogeneous agent systems to interact. There are a growing number of agent projects, platforms and agent applications based on the FIPA standard (for example [15]). Earlier, today outdated, FIPA documents did consider some security issues. However, the current standards do not deal with security. FIPA has recognised this and recently issued a request for work in the area [16]. [17] includes a very brief attempt to add security to a FIPA agent system, where it is suggested that the agent platform implements both authentication of agents and facilitators and the use of encrypted channels. However, no details are included, key management and how authentication should be done is not specified.

KQML (Knowledge Query and Manipulation Language) [2] is a message protocol for software agents to communicate with each other. The protocol has been developed as

²See <http://www.agentbuilder.com/AgentTools> for a list of available systems.

part of an ARPA project. KQML does not deal with security issues but depends on security being provided by lower layers. [18] proposes a security architecture for KQML. Symmetric or asymmetric cryptography is supported and keys are assumed to be agreed beforehand. The proposed extension provides for confidentiality, authentication, and (limited) data integrity protection. However, it does not protect against message replay attacks. A solution using mediating agents to enable communication with crypto unaware agents is also proposed. Another suggestion for enhancing KQML with security is proposed in [19]. Parameters for certificate management are defined leaving the format of the certificate undefined.

The security issues posed to mobile agents are rather well understood by the security community and hence much research is being devoted to the area. There have been many attempts to address the threats posed to mobile agents, either completely or in part. Most of these attempts fall into one of the following broad categories.

- The first category comprises approaches that do not allow an agent to leave a trusted environment. Solutions to this include using a host infrastructure that is operated by a single party, allowing agents to migrate only to trusted hosts [20], or possibly hosts with a good reputation [21].
- The second category is pragmatic; it consists of solutions to a single part of the malicious host problem. These consist of agents detecting when they have been modified [22], and proof verification techniques [23].
- The third class consists of assuming that there is special, tamperproof hardware available, see for example [24] or [23].
- In the fourth category are attempts to split the task between agents residing on different platforms and thereby not having to depend on a single host (e.g. [25] and [26]).
- The final category uses software methods to obscure the code from the host. Approaches include obfuscation [27], mobile cryptography [28], [29] and using environmental conditions to hide parts of the code [30].

A few of the mentioned security mechanisms do not require infrastructure support, while others do. Most of these mechanisms are not implemented in available MAS implementations. The area of mobile agents is currently attracting much research effort.

V. CONCLUSIONS

In this section we will consider how well existing security technology fits the requirements for security in multi-agent systems.

A. Mobile agents

We have chosen to separate the discussion of solutions for mobile agents from agents in general. While we believe that the security issues for non-mobile agents can be tackled to a great extent through existing technology (described below), this does not appear to be the case for mobile agents.

There does not seem to be a single solution to the security problems introduced by mobile agents unless trusted

hardware is introduced, which is likely to prove too expensive for most applications. The way forward lies probably in a range of mechanisms aimed at solving particular (smaller) problems. This could, for example, include mechanisms that depend on agents executing on several hosts rather than on only one host, mechanisms and protocols binding agent actions to hosts, generation of various audit information that can be used in case of disputes, and so on. Solutions to certain problems do exist but for mobile agents to be more widely adopted this is an area that requires further research.

B. Host security

Computer hosts providing execution platforms for agents need to make use of what are usually referred to as computer security techniques. This would for example include local access control mechanisms to prevent agents from reading and manipulating information (including the agent itself) belonging to other agents. It would also include resource management, assuring fair allocation of resources to agents executing on the host, thereby preventing a single agent (or a group of agents) from consuming too many resources and thereby preventing other agents from proper execution. As for any service connected to an open network, such as the Internet, controls to prevent unauthorised remote access would also be required.

Even if agents are not mobile, the sandbox concept is likely to be useful. One can imagine a model where a user develops and/or configures an agent that would be executed on a host owned by a service provider. A sandbox limits the environment and actions available for an agent to prevent it from carrying out malicious actions.

The security measurements mentioned in this section are all rather well known and used in today's computer systems. It is also likely that some of these measures have been implemented on existing agent platforms. If built upon an existing operating system (as most agent platforms are likely to be) some of these measures would be provided by the operating system.

C. Agent communication

For agent communication we believe it is important to implement security at a layer such that agents are aware of security properties, as opposed to relying on security being provided by lower layers. As suggested elsewhere, public key cryptography and a supporting public key infrastructure can be used as important parts of inter-agent communication.

With a public key infrastructure in place, security protocols and mechanisms already developed for other applications can be made to fit the requirements of multi-agent systems to provide authentication, confidentiality, and integrity.

Anonymity is an issue that can be useful to tackle in the context of secure communication. Since anonymity in many ways interferes with security properties, finding ways to provide (limited) anonymity in combination with security can prove challenging. However, by using proxies it

should be possible to provide limited anonymity for agents without losing relevant security properties.

D. Delegation

Delegation, being an important concept for agents, needs to be properly addressed. With a public key infrastructure in place, delegation can be done through various types of certificate; including attribute certificates [31] for delegation of rights, and issuing of ‘traditional’ public key certificates [31] for delegation of signing rights [32].

VI. FUTURE WORK

Within the work carried out by the Core 2 programme of Mobile VCE (www.mobilevce.com) we are currently developing a security model and architecture for a multi-agent system within a mobile setting, where mobile applies to users as well as to terminals. We will further analyse the possible PKI requirements for a MAS in this environment as well as requirements on delegation and secure communication. Protocols will be specified to support the necessary security features.

We are also planning to carry out further research on security mechanisms for mobile agents. This appears to be the most challenging area concerning agents and multi-agent systems.

ACKNOWLEDGEMENTS

The work reported in this paper has formed part of the Software Based Systems area of the Core 2 Research Programme of the Virtual Centre of Excellence in Mobile & Personal Communications, Mobile VCE, www.mobilevce.com, whose funding support, including that of the EPSRC, is gratefully acknowledged. More detailed technical reports on this research are available to Industrial Members of Mobile VCE.

REFERENCES

- [1] Bernard Burg, “Towards the deployment of an open agent world,” in *Journées Francophones d’Intelligence Artificielle Distribuée et de Systèmes Multi-Agents (JFIADMSA2000)*, Hermes, Ed., October 2001.
- [2] M. Genesereth and R. Fikes, “Knowledge interchange format, version 3.0 reference manual,” Tech. Rep. Logic-92-1, Computer Science Department, Stanford University, USA, 1992.
- [3] N. R. Jennings, K. Sycara, and M. Wooldridge, “A roadmap of agent research and development,” *Autonomous Agents and Multi-Agent Systems*, vol. 1, no. 1, pp. 275–306, 1998.
- [4] N. R. Jennings and M. Wooldridge, “Intelligent agents: Theory and practice,” *The Knowledge Engineering Review*, vol. 10, no. 2, pp. 115–152, 1995.
- [5] J. E. White, “Telescript technology: The foundation for the electronic marketplace,” Tech. Rep., General Magic Inc, 2465 Latham Street, Mountain View, CA 94040, 1994.
- [6] J. R. Galliers, *A Theoretical Framework for Computer Models of Cooperative Dialogue, Acknowledging Multi-Agent Conflict*, Ph.D. thesis, Open University, UK, 1988.
- [7] Jeffery S. Rosenschein and Michael R. Genesereth, “Deals among rational agents,” in *The Ecology of Computation*, B. A. Huberman, Ed., pp. 117–132. North-Holland Publishing Company, Amsterdam, 1988.
- [8] J. M. Bradshaw, “An introduction to software agents,” in *Software Agents*, Cambridge, MA, 1997, pp. 3–46, AAAI Press, Menlo Park, Calif., USA.
- [9] K. Decker, M. Williamson, and K. Sycara, “Matchmaking and brokering,” in *Proceedings of the Second International Conference on Multi-Agent Systems (ICMAS-96)*, December 1996, p. 432.
- [10] Gio Wiederhold, “Mediators in the architecture of future information systems,” *IEEE Computer*, vol. 25, no. 3, pp. 38–49, March 1992.
- [11] H. P. Nii, *The Handbook of Artificial Intelligence*, vol. IV, chapter XVI, Blackboard Systems, pp. 1–82, Addison-Wesley, New York, 1989.
- [12] Vesna Hassler, *Security Fundamentals for E-commerce*, Artech House, 2000.
- [13] Fritz Hohl, “A model of attacks of malicious hosts against mobile agents,” in *Proceedings of the ECOOP Workshop on Distributed Object Security and 4th Workshop on Mobile Object Systems: Secure Internet Mobile Computations*, 1998, pp. 105–120, <http://mole.informatik.uni-stuttgart.de/simc98.ps.gz>.
- [14] Reticular Systems Inc, San Diego, CA, USA, *Agentbuilder Reference Manual*, version 1.3 rev. 0 edition, April 2000, Available from <http://www.agentbuilder.com>.
- [15] S. Posland, P. Buckle, and R. Hadingham, “The FIPA OS agent platform: Open source for open standards,” in *the 5th International Conference and Exhibition on the Practical Application of Intelligent Agents and Multi-Agents*, Manchester, UK, April 2000, pp. 355–368.
- [16] FIPA, “FIPA security SIG request for information,” February 2001, <http://www.fipa.org>.
- [17] S. Posland and M. Calisti, “Towards improved trust and security in FIPA agent platforms,” in *Autonomous Agents 2000*, June 2000.
- [18] Chelliah Thirunavukkarasu, Tim Finin, and James Mayfield, “Secret agents - a security architecture for the KQML agent communication language,” in *Intelligent Information Agents Workshop held in conjunction with Fourth International Conference on Information and Knowledge Management CIKM’95*, Baltimore, December 1995, pp. 176–184, IEEE Comp. Soc. Press.
- [19] Qi He, Katia P. Sycara, and Timothy W. Finin, “Personal security agent: KQML-Based PKI,” in *Proceedings of the 2nd International Conference on Autonomous Agents*, Katia P. Sycara and Michael Wooldridge, Eds., New York, 1998, pp. 377–384, ACM Press, New York, USA.
- [20] William Farmer, Joshua Guttmann, and Vipin Swarup, “Security for mobile agents: Authentication and state appraisal,” in *Proceedings of the European Symposium on Research in Computer Security (ESORICS)*. 1996, number 1146 in LNCS, pp. 118–130, Springer-Verlag, Berlin.
- [21] Lars Rasmussen and Sverker Jansson, “Simulated social control for secure internet commerce,” in *New Security Paradigms ’96*. September 1996, pp. 18–26, ACM Press.
- [22] Giovanni Vigna, “Protecting mobile agents through tracing,” in *Proceedings of the Third ECOOP Workshop on Operating System support for Mobile Object Systems*, Finland, June 1997, pp. 137–153.
- [23] Bennet Yee, “A sanctuary for mobile agents,” in *Secure Internet Programming*, Jan Vitek and Christian Jensen, Eds., New York, NY, USA, 1999, number 1603 in Lecture Notes in Computer Science, pp. 261–274, Springer-Verlag Inc.
- [24] U. G. Wilhelm, S. Staamann, and L. Buttyán, “On the problem of trust in mobile agent systems,” in *Symposium on Network and Distributed System Security*. March 1998, pp. 114–124, Internet Society.
- [25] Niklas Borselius, Chris J. Mitchell, and Aaron Wilson, “On mobile agent based transactions in moderately hostile environments,” in *Advances in Network and Distributed Systems Security, Proceedings of IFIP TC11 WG11.4 First Annual Working Conference on Network Security, KU Leuven, Belgium*, B. De Decker, F. Piessens, J. Smits, and E. Van Herreweghen, Eds. November 2001, pp. 173–186, Kluwer Academic Publishers, Boston.
- [26] Fred B. Schneider, “Towards fault-tolerant and secure agency,” in *Eleventh International Workshop on Distributed Algorithms*. September 1997, number 1320 in LNCS, pp. 1–14, Springer-Verlag, Berlin.
- [27] Fritz Hohl, “Time limited blackbox security: Protecting mobile agents from malicious hosts,” in *Mobile Agents and Security*, Giovanni Vigna, Ed. 1998, number 1419 in LNCS, pp. 92–113, Springer-Verlag, Berlin.
- [28] Tomas Sander and Christian Tschudin, “Towards mobile cryp-

- tography,” in *Proceedings of the IEEE Symposium on Security and Privacy*, Oakland, CA, May 1998, pp. 215–224, IEEE Computer Society Press.
- [29] Tomas Sander and Christian Tschudin, “Protecting mobile agents against malicious hosts,” in *Mobile Agents and Security*, Giovanni Vigna, Ed. 1998, number 1419 in LNCS, pp. 44–60, Springer-Verlag, Berlin. Available from <http://www.icsi.berkeley.edu/~sander/publications/MA-protect.ps>.
- [30] James Riordan and Bruce Schneier, “Environmental key generation towards clueless agents,” in *Mobile Agents and Security*, G. Vigna, Ed. 1998, vol. 1419 of LNCS, pp. 15–24, Springer-Verlag, Berlin.
- [31] International Telecommunication Union, “X.509, information - technology - Open systems - Interconnection - The Directory: Public-key and attribute certificate frameworks,” 2000, also ISO International Standard 9594-8.
- [32] Niklas Borselius and Chris J. Mitchell, “Certificate translation,” in *NORDSEC 2000 - 5th Nordic Workshop on Secure IT Systems*. 2000, pp. 289–300, Reykjavik University, Reykjavik, Iceland, 12/13 October 2000.