Security in Ad Hoc Networks: a General Intrusion Detection Architecture Enhancing Trust Based Approaches

Patrick Albers[1], Olivier Camp[1], Jean-Marc Percher[1], Bernard Jouga[2], Ludovic Mé[2], and Ricardo Puttini[2]

[1] École Supérieure d'Électronique de l'Ouest (ESEO),
4, rue Merlet de la Boulaye,
49000 Angers, France
{patrick.albers,olivier.camp,jean-marc.percher}@eseo.fr
[2] Supélec,
BP 81127,
Cedex 35511 Cesson Sévigné, France
{bernard.jouga,ludovic.me,ricardo.puttini}@supelec.fr

Abstract. In the last few years, the performances of wireless technologies have increased tremendously thus opening new fields of application in the domain of networking. One of such fields concerns mobile ad hoc networks (MANETs) in which mobile nodes organise themselves in a network without the help of any predefined infrastructure. Securing MANETs is just as important, if not more, as securing traditional wired networks. Existing solutions can be used to obtain a certain level of security. Nevertheless, these solutions may not always be suitable to wireless networks. Furthermore, ad hoc networks have their own vulnerabilities that cannot be tackled by these solutions. To obtain an acceptable level of security in such a context, traditional security solutions should be coupled with an intrusion detection mechanism.
In this paper we show how ad hoc networks can be, to a certain extent, secured using traditional techniques. We then examine the different intrusion detection techniques and point out the reasons why they usually cannot be used in an ad hoc context. Finally, we go through the requirements of an intrusion detection system for ad hoc networks, and define an adapted architecture for an intrusion detection system for manets.

1 Introduction

Mobile ad hoc (or spontaneous) networks (manet) are IP networks made up of a collection of wireless and mobile nodes communicating via radio links. They do not depend on any predefined infrastructure or centralised administration to operate [5] and could, for example, find applications in the case of networks created for the needs of participants to a conference or meeting [9], students and teachers in a classroom, rescuers in a search and rescue operation, soldiers on a battlefield ...

Ad hoc networks can either be standalone networks (this would be the case of a spontaneous network created for the needs of a meeting between participants away from their home network -for example an airport or a hotel lounge) or peripheral networks connected, for instance, to a wired local area network or to the Internet (this would certainly be the case for a virtual classroom where students need to intercommunicate and may need to access documentation on the web or in the school's digital library).

Ad hoc networks being spontaneous and mobile, their configuration should be done with as little user intervention as possible and the nodes should be able to rely on an adapted routing algorithm to exchange information across the network. Furthermore, ad hoc networks should offer the necessary security level for user applications.

Due to the lack of an underlying infrastructure, basic functionalities, such as routing, configuration of the hosts or security management cannot rely on predefined or centralised entities to operate, and must be carried out in a distributed manner. For instance, in the case of security, the nodes cannot rely on network architecture based defense techniques such as centralised firewalls. Each node thus becomes a point of vulnerability and must assume, by itself, it's own security.

Routing and auto configuration are two fields closely examined by the *manet* (mobile ad hoc networking), *zeroconf* (zero configuration networking), and *ipng* (IP next generation) IETF working groups[1], but, research concerning the security aspects of these mechanisms still seems to be immature.

Security requirements in wireless networks are nonetheless identical to those in wired networks and ad hoc networks should offer mechanisms to achieve the following security services: Authentication, access-control, confidentiality, integrity and non-repudiation.

This paper focuses on the specific security requirements of such new generation spontaneous IP networks and shows how an adapted intrusion detection mechanism can be implemented to increase their security.

The article is organised as follows: Section 2 presents the wireless ad hoc context and its vulnerabilities. In Sect. 3, we see why some of the existing security solutions used in wired networks are not adapted to the decentralised nature of manets. We show that a certain security can be obtained with a trust based mechanism and point out the remaining vulnerabilities. After going through the requirements for an IDS for manets, Sect. 4 proposes a suitable architecture for such a system. Finally, we conclude in Sect. 5, and present our future works aimed at preventing attacks on the network's routing protocol.

[1] Official charters can be found at:
http://www.ietf.org/html.charters/ipngwg-charter.html
http://www.ietf.org/html.charters/manet-charter.html
http://www.ietf.org/html.charters/zeroconf-charter.html

2 Ad Hoc Networks

Even though research in securing manets is not a major issue today, we consider that the development of ad hoc networks will depend, of course, on an efficient routing algorithm and a performing autoconfiguration mechanism, but also on an acceptable security level for the users.

After giving an overview of the current work around mobile ad hoc networks, this section goes through their vulnerabilities.

2.1 Background

Routing: Two approaches of uni-cast routing algorithms are considered for ad hoc networks [12]: Proactive, table driven algorithms and reactive, source initiated, on-demand routing protocols. Both approaches rely on full cooperation of the nodes in the network.

Despite a lot of effort put on this subject the choice of a standard protocol still has to be made. The performances of both types of protocol are very dependent on the nature of the underlying network; its density and size, the mobility of the nodes, the type of traffic it deals with... So far, none of the specified protocols is suitable for all possible network configurations. We think that a possible solution could be to consider a mixed approach capable of dynamically adapting itself to the current situation.

Autoconfiguration: When a host wishes to join an existing ad hoc network it needs to configure itself for this particular network. Such a configuration consists in choosing an IP address, identifying the network's DNS servers, routing protocol,...

The native IPv6 functionalities offer this type of autoconfiguration mechanism and an implementation of a similar mechanism can be thought of for an ad hoc network functioning under IPv4. In both cases, just like routing, autoconfiguration is based on a collaborative participation of the network.

Security: Even though security is stated as being one of the concerns of the MANET working group [5], literature on the subject is rather poor and this issue is only treated in a few papers [17, 13, 18]. It seems that security concerns have been postponed until the specification of a routing protocol for ad hoc networks has been achieved. Nevertheless, we are convinced that security problems cannot be considered separately and must be taken into account for the specification of all the functionalities of the network (routing, autoconfiguration,...). Indeed, existing mechanisms providing the main security services may not easily be adaptable to the cooperative, distributed and spontaneous nature of ad hoc networks.

2.2 Ad Hoc Network Vulnerabilities

Ad hoc networks suffer from the same kinds of vulnerabilities as their wired counterparts: Passive eavesdropping, spoofing, replay, denial of service... Some

of these vulnerabilities are accentuated in a wireless context and new vulnerabilities intrinsic to this type of network arise.

Vulnerabilities accentuated by the ad hoc context: The topology of an ad hoc network is defined by the geographical position and by the wireless emission ranges of its hosts. A consequence of this is that these networks do not have a clearly defined physical boundary and thus no entry point. Access-control to the network, as it is traditionally achieved by a LAN's firewall, thus becomes a tricky point to deal with. Attention should thus be placed on the problems of IP masquerading and passive eavesdropping, and a protection against these attacks should be implemented.

Other types of attacks such as the popular denial-of-service (DOS) attacks, frequently encountered on wired networks, can of course be transposed to the wireless context. The vital routing and autoconfiguration services are vulnerable to such attacks.

Vulnerabilities specific to the ad hoc context: Among the intrinsic vulnerabilities of ad hoc networks, some reside in their routing and autoconfiguration mechanisms. Both these key functionalities of ad hoc networks, are based on a full trust between all the participating hosts.

In the case of routing, the correct transport of the packets on the network relies on the veracity of the information given by the other nodes. The emission of false routing information by a host could thus create bogus entries in routing tables throughout the network making communication difficult. Furthermore, the delivery of a packet to a destination is based on a hop by hop routing and thus needs total cooperation from the intermediate nodes. A malicious host could, by refusing to cooperate, quite simply block or modify the traffic traversing it.

By fooling the routing algorithm or even by choosing a strategic geographic positioning, a host can thus control the traffic to and from entire parts of the network.

The autoconfiguration mechanism also brings up new vulnerabilities. This functionality, whether it uses the ICMPv6 *router solicitation/neighbour solicitation* messages [4, 7] or a similar autoconfiguration principle above IPv4, is vulnerable to false replies. Both these processes use information given by the nodes on the network to, either calculate an IP address or verify that a particular address is not already used. In the case of Duplicate Address Detection (DAD), a danger exists that a malicious node pretends using any of the addresses chosen by the incoming host, thus denying it the right to join the network.

Finally, ad hoc nodes usually being battery powered, by forcing a host to relay packets, an attacker could make it empty its battery creating a new type of DOS attack: The "sleep deprivation torture" attack [13].

3 Security and Ad Hoc Networks

As in wired networks, the main security services (access control, authentication, confidentiality, integrity and non-repudiation) should be provided by manets.

Existing solutions for the security of wired networks could be applied to wireless environments. Nevertheless, the intrinsic characteristics of wireless networks (absence of centralised infrastructure, mobility of the nodes, limited bandwidth, ...) may limit their application. For instance, key certification as implemented in a public key infrastructure (PKI) usually relies on a point of centralisation. Such centralised architectures are not adapted to the dynamic topology of manets. In such networks, the mechanisms implementing continuous services, such as security functions, should be distributed.

In what follows, we go through the different existing network security solutions and point out where attention should be placed when implementing them in the ad hoc context. We show how a simple trust based mechanism sometimes provides a solution to security requirements.

3.1 Existing Security Solutions

Traditional mechanisms, such as asymmetric cryptography, one way hash functions and other techniques implementing authentication, confidentiality, integrity and non-repudiation can be used whether in a wired or wireless network. On the other hand, access control, which for us stands for firewalling, seems somehow more difficult to enforce in an ad hoc network. We will see that a trust based security policy can help solve this problem at the network level. On the other hand, applicative firewalling, as achieved by proxies, cannot be considered in manets because of their centralised nature.

According to the security goals to be achieved, several mechanisms can be implemented on different network layers.

Most existing mechanisms are based on cryptography and certification must be implemented to secure key exchanges. This point is particularly important in an environment prone to "Man in the Middle" attacks such as manets. A certification mechanism can be implemented in many ways ranging from a simple physical exchange of keys, to a more sophisticated PKI based exchange [13]. The choice depends on the configuration of the network and the required security.

3.2 Managing Security with Spontaneous Communities

As we have seen, a serious breach in the security of ad hoc networks resides in their lack of a clear physical boundary. This makes them particularly vulnerable to eavesdroppers. We show here how *communities* can prevent this type of attacks.

Communities are defined by a relation of trust between participants. Such trust could be persistant, and a host could be configured so as to belong initially to a number of native communities. It may also dynamically define transient trust relations with other nodes so as to create a spontaneous community (similar to L.M.Feeney's "spontaneous networks" [9]).

A host could belong simultaneously to both types of communities.

Communities can be a first solution when tackling security problems in ad hoc networks. Indeed, if the trust relationship implies that the hosts in a community can securely exchange keys, then authentication, confidentiality, integrity and non-repudiation can easily be provided by the network. A host is thus able to establish secure unicast and multicast communications with other members of its community. Furthermore, a community based policy may define the levels of security imposed on exchanges inside and between communities and tackle the problem of access-control. In a way, a community's network can be seen as a virtual subnetwork involving the nodes of the community and capable of protecting itself from eavesdropping by encryption. A node's access to the subnetwork thus depends on whether it belongs to a given community and not simply be a matter of wireless transmission range.

With such a principle, a node belonging to several communities will have, as many security policies. Care should be taken to manage such multiple policies, since one particular communication may be covered by several restrictions.

3.3 Remaining Vulnerabilities

Communities obviously rely on the trustworthiness of their members for their security and such mutual trust must be guaranteed throughout their existence.

In the same way as a majority of attacks against wired networks originate from within it, it is to fear that attacks from inside a community will not be rare. A mechanism should therefore assure that all the nodes in a community behave according to the security policy and deserve to be trusted. Moreover, we have seen that a mechanism such as routing performs best if it can rely on the cooperation of as many hosts as possible. In the case of a community evolving in an open environment containing unknown nodes this would most certainly lead to relying on strangers for data exchanges. The community should thus try to detect malicious nodes so as to exclude them when performing cooperative actions.

Existing mechanisms, implemented through the use of communities, can only provide a certain security to ad hoc networks. Complementarily to these mechanisms mentioned above, we feel that intrusion detection must be implemented to obtain an optimal security level in such networks.

After going through an overview of existing solutions to intrusion detection and pointing out the requirements in the ad hoc context, we will propose a suitable architecture for such wireless, mobile environments.

4 Intrusion Detection

In order to provide a way of securing the vital network functionalities without affecting their efficiency, we propose to use intrusion detection. In this paper we will not investigate the detection techniques (this will be part of our future work), but we will describe a suitable architecture for an intrusion detection system (IDS) for manets. Furthermore, we will show how such an IDS can, in this context, be considered as a preventive security tool.

4.1 Characteristics of Intrusion Detection Systems

An IDS collects and analyses audit data to detect unauthorised uses and misuses of computer systems [8].

To present the characteristics of IDS we will use the following criteria, defined by the IBM labs in Zurich [6].

- *Audit source location:* The data to be analysed may be obtained on the host, in application or system log files by *Host Based Intrusion Detection Systems* (*HIDS*) or from the network (for instance, by placing sniffers on interconnection equipment) by *Network Based Intrusion Detection System* (*NIDS*).
- *Methodology of detection:* Two approaches are used for the detection of intrusions: Anomaly detection and misuse detection. With anomaly detection, the system knows the user's standard profile and detects deviations from this reference. Misuse detection, on the other hand, relies on the signature of attacks. Even though, commercial products tend to prefer signature based detection, neither of the two techniques has really proven better than the other and research in this field still remains active.
- *Computing location:* Most IDS use a centralised architecture to gather and analyse audit data. Some of them use agent technology to realise a local pre-analysis prior to centralising the data.
- *Usage frequency:* An IDS can collect and analyse data at regular intervals or provide a continuous intrusion detection service. The latter is particularly needed by an open environment such as Internet where intrusions should be detected "on the fly".
- *Response to intrusions:* When an intrusion is detected the system may react in different ways. Most systems generate an alarm informing the administrator, who decides of the reaction to have. A more sophisticated response consists in a corrective action (a new rule in a firewall, disconnection of suspicious connections,...) to prevent an identical future attack.

4.2 Requirements of an IDS for Ad Hoc Networks

An IDS for ad hoc networks, independently from network specifications, should:

- *not introduce a new weakness* for the system. Ideally it should ensure its own integrity,
- *need little system resources* to run and should not degrade the system performances by introducing overhead,
- *run continuously* and remain transparent to the system and the users,
- *use standards* to be cooperative and open. The specifications of such standards are based on proposals by the IETF *Intrusion Detection Working Group* (IDWG)[2]. Recently this work was aimed at defining standards for

[2] The IDWG official charter can be found at:
http://www.ietf.org/html.charters/idwg-charter.html

IDS: A standard alert format *Intrusion Detection Message Exchange Format* (IDMEF) and a protocol for transporting such alerts *Intrusion Detection Exchange Protocol* (IDXP).
- *be reliable* and minimize false positives and false negatives detections.

Specific requirements must also be considered in the mobile wireless and ad hoc context. As for routing and autoconfiguration, the intrusion detection mechanism should also be distributed [11].

A distributed hierarchical IDS architecture such as the one found in IDA (*Intrusion Detection Agent System*) [1], AAFID (*Autonomous Agents For Intrusion Detection*) [2] GrIDS (*Graph-Based Intrusion Detection System*) [14] cannot be considered here, as the root of the hierarchy would introduce a point of centralisation.

A simple solution is for each host to only rely on itself for intrusion detection. The audit data can thus be gathered and computed locally and, to have a broader knowledge of its environment a node may also request complementary information from others. It could also be informed of a local intrusion detected by another node. This cooperation will benefit from standard data exchange format and intrusion alert protocol.

With such a distributed architecture a host joining the network will do so with its own IDS, thus reinforcing the global intrusion detection mechanisms.

4.3 A General Architecture for Ad Hoc Intrusion Detection

Communities, as they are described in Sect. 3, should provide a means of making sure that its members can be trusted and that they respect the community's security policy. Moreover, they should be able to select extra community nodes, on which to rely, for cooperative network functionalities. They should, at the least, be capable of detecting malicious nodes so as to exclude them from such distributed mechanisms.

Such detections can be carried out by an adapted intrusion detection mechanism. Yet, it must be distributed throughout the network.

We propose to achieve the distribution of the intrusion detection mechanism by implementing a *Local Intrusion Detection System* (LIDS) on each node. In order to make this detection a global concern for the community, the different LIDS coexisting within it, should collaborate. This would extend each LIDS's vision of the network. Such a general architecture was proposed in [17] and fits well the needs of manets. To have, yet a broader knowledge of its environment, a LIDS may also take into account the intrusions detected by other members of its community. The different LIDS in a community will thus exchange two types of data:

- Security data: To obtain complementary informations from collaborating hosts.
- Intrusion alerts: To inform others of a locally detected intrusion.

We will now describe this architecture with regards to the characteristics presented in 4.1.

– **Audit source and computing location:** We propose to use SNMP (Simple Network Management Protocol) data located in MIBs (Management Information Base), as an audit source for LIDS. Such a data source provides several advantages:

  • it is independent from the operating system,
  • it can be extended in order to collect and store additional data relative to network activities, operating system or applications [16, 3],
  • if an SNMP agent runs on a node, the cost of the collection of local information needs no additional resources,
  • the standard representation of the data collected on each node facilitates cooperation between LIDS.

  To confirm a detected suspicious action, a LIDS may need complementary external information. Considering the unreliability of the UDP transport protocol underlying SNMP and the dynamic topology of manets, we propose to use mobile agents to transport SNMP requests to remote hosts. To fully benefit from the mobile agent framework, agents should also be autonomous and adaptative. We will now focus on these characteristics:

  - *Mobility*: Mobile agents, as opposed to traditional approaches, where data are transported towards the computation location, bring the code to the data. This aspect, allows asynchronous execution of the agent on a remote host.Another point, interesting for manets, is that mobility can significantly reduce exchanged data.
  - *Autonomy*: Mobile agents are given a mission upon their creation. For instance, an agent could be created to trace an incoming *telnet* connection. It should be capable of achieving this aim without any help from its LIDS,
  - *Adaptability*: An agent should adapt its behaviour according to the information it gathers while achieving its mission. For instance, the tracing agent should be able to decide, on its own, what actions to undertake in order to reach the source of the, possibly cascading, incoming connection.

  The above factors allows an LIDS to delegate to an agent a specific mission which will be achieved in an autonomous and asynchronous manner. This is particularly interesting in the ad hoc context, in which connections may be unreliable.
  As shown by Fig. 1, all external or internal communications with an LIDS rely on a common communication framework. If the framework can understand IDMEF and IDXP messages delivered by the network, then, it will facilitate cooperation with other open IDS. Any IDS capable of using the standards implemented by the framework may thus act as a remote data source either providing security data or intrusion alerts.
  Several types of data collecting agents coexist in a LIDS and each of them has a specific function:

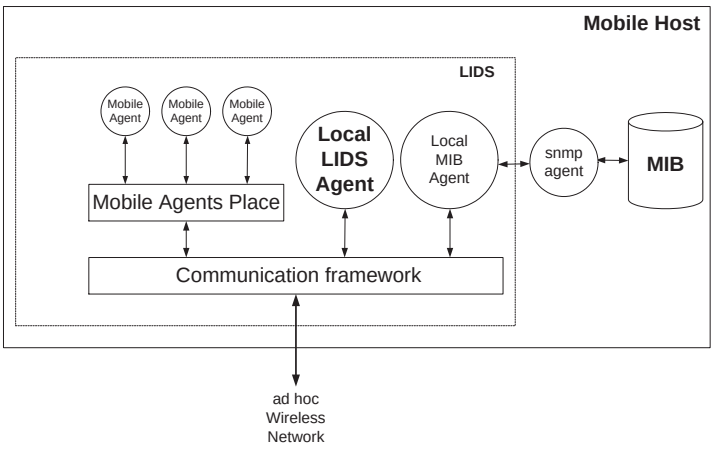

Fig. 1. LIDS architecture

- A Local LIDS Agent, is in charge of local intrusion detection and response. It also reacts to intrusion alerts provided by other nodes in order to protect itself against this intrusion.
- Mobile Agents collect and process data on remote hosts. They may decide to transfer the results of a computation back to their home LIDS or to migrate to another node for further investigation. The security control of these agents can be taken in charge by the *Mobile Agent Place*. An agent should also be able to protect itself from malicious mobile agent places [10].
- The local MIB agent provides a means of collecting MIB variables for, either, mobile agents or the Local LIDS Agent. If SNMP runs on the node the Local MIB Agent will simply be an interface with the running SNMP agent. Otherwise, an SNMP based agent should be developed specifically, to allow optimised updates and retrieval of the MIB variables used by intrusion detection. The Local MIB agent would in that case act as an interface between the LIDS and this tailor-made agent.

– **Methodology of detection:** In the proposed architecture, the Local LIDS Agent could use either misuse or anomaly detection. We will first consider the implementation of an LIDS based on the signatures of attacks, using MIB variables to define signatures. However, both techniques should be considered to offer the best intrusion detection model.
– **Response to intrusion:** As soon as a LIDS detects an intrusion locally, it should inform the other nodes of the network. Locally, the user could choose to refuse connections with the suspicious node, to exclude it when performing cooperative actions, or to exclude it from its community until it re-authenticates itself. By being informed of intrusions on remote hosts, a LIDS can act as a preventive security tool and prevent the intruder from attacking it.

– **Usage:** For the best security in an ad hoc network, all its LIDS should run and cooperate continuously.

The above defined architecture seems well adapted to the specificities of manets. Indeed, mobile agents provide solutions to the relatively low performances of manets and their use make the global architecture evolutive. Furthermore, a node joining the network, does so with its own LIDS. It is, in that way, able to participate in the global detection mechanism, thus reenforcing it. If all LIDS in a community have similar detection capabilities, scalability of the community's global IDS is guaranteed.

5 Conclusion and Future Works

In this paper we showed how a simple trust based mechanism coupled with a mobile agent based intrusion detection system could ensure the security services required by users in a manet. These mechanisms are complementary; communities give a framework allowing the definition of security policies, whereas intrusion detection ensures that such policies are not violated.

We have implemented a feasibility prototype of the above architecture [15]. It detects incoming *telnet* connections and reacts if they originate from outside the community's network. The first results obtained with this prototype are encouraging and the architecture seems well adapted to ad hoc networks. We are now studying the detection of five main attacks against the OLSR (Optimised Link State Routing) protocol (a table driven ad hoc routing protocol) and the response to give to such intrusions. An implementation of OLSR is available for IPv4 and it's adaptation to IPv6 is on its way. Our final goal is to use a fully IPv6 manet in order to benefit from the security and autoconfiguration aspects it proposes.

Acknowledgments

This paper is the result of the work we are carrying out in the context of the RAHMS (Secured Multiservice Ad Hoc Networks) project, funded by the French Ministry of Industry and Telecommunication, and managed by 6-Wind[3].

The authors would like to thank all the members of the project team: Catherine Devic from EdF, Eric Carmes, Vladimir Ksinant and Jean-Mickael Guerin from 6Wind, and Lionel Evon and Nicolas Jordan from Netcentrex.

References

1. M. Asaka, A. Taguchi, and S. Goto. The implementation of ida : An intrusion detection agent system. *In proceeding of 11th FIRST Conference-Brisbane-Australia*, 1999.

[3] 6Wind is a French company specialised in new generation networks, http://www.6wind.com

2. J. S. Ballasubramaniyan, J. O. Garcia-Fernandez, D. Isaco, E. Spafford, and D. Zamboni. Aafid - autonomous agents for intrusion detection. Coast Technical Report 98/05, Coast Lab. – Purdue University West Lafayette, 1998.
3. João B. D. Cabrera, Lundy Lewis, Ravi K. Prasanth Xinzhou Qin, Wenke Lee, and Raman K. Mehra. Proactive detection of distributed denial of service attacks using MIB traffic variables – a feasibility study. In *Proceedings of the 7th IFIP/IEEE International Symposium on Integrated Network Management, Seattle, USA*, 2001.
4. A. Conta and S. Deering. Internet control message protocol (ICMPv6) for the internet protocol version 6 (ipv6). Request for Comments (Standards track) 2463, IETF, 1998.
5. S. Corson and J. Macker. Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation consideration. Request for Comments (Informational) 2501, IETF, 1999.
6. H. Debar, M. Dacier, and A. Wespi. Towards a taxonomy of intrusion detection systems. IBM Zurich Research Laboratory, Ruschlikon, Switzerland, 1998.
7. S. Deering and R. Hinden. Internet protocol, version 6 (ipv6) specification. Request for Comments (Standards track) 2460, IETF, 1998.
8. Dorothy E. Denning. An intrusion detection model. In *IEEE Transactions on software engineering, Vol. SE-13, NO.2*, pages 222–232. IEEE, 1987.
9. L. M. Feeney, B. Ahlgren, and Assar Westerlund. Spontaneous networking: An application-oriented approach to ad hoc networking. *IEEE Communication Magazine*, 39(6), 2001.
10. D. Hagimont and L. Ismail. A protection scheme for mobile agents on java. In *Proceedings of the 3rd Anual ACM/IEEE International Conference on Mobile Computing and Networking*, pages 215–222, 1996.
11. Stefano Martino. A mobile agent approach to intrusion detection. Technical report, Joint Research Centre Institute for Systems, Informatics and Safety, Italy, 1999.
12. E. Royer and C. Toh. A review of current routing protocols for ad-hoc mobile wireless networks. *IEEE Personal Communications*, 1999.
13. Frank Stajano and Ross J. Anderson. The resurrecting duckling: Security issues for ad-hoc wireless networks. In B. Christiano, B. Crispo, and M. Roe, editors, *Security Protocols, 7th International Workshop Proceedings*, pages 172–194. Lecture Notes in Computer Science, 1999.
14. S. Staniford-Chen, S. Cheung, R. Crawford, M. Dilger, J. Frank, J. Hoagland, K. Levitt, C. Wee, R. Yip, and D. Zerkle. GrIDS – A graph-based intrusion detection system for large networks. In *Proceedings of the 19th National Information Systems Security Conference*, 1996.
15. Nguyên Chung Tiên. Validation d'une architecture à agents mobiles pour la détection d'intrusion (in french). Master's thesis, Institut de la Francophonie pour l'Informatique, Hanoi, Vietnam, 2001.
16. Feiyi Wang, F. Gong, F.S. Wu, and R. Narayan. Intrusion detection for link state routing protocol through integrated network management. In *Proceedings of the 8th International Conference on Computer Communications and Networks*, pages 634–639, 1999.
17. Y. Zhang and W. Lee. Intrusion detection in wireless ad hoc networks. In *Proceedings of the sixth annual international conference on Mobile computing and networking, MOBICOM'2000*, pages 275–283. ACM Press New York, USA, 2000.
18. L. Zhou and Z. Haas. Securing ad hoc networks. *IEEE Network Magazine, November-December 1999*, 13(6), 1999.