An Active Network Approach for Security Management

Ahmed Eddaoui[†] and Abdellatif Mezrioui [††]

[†] FSTM, Mohammedia Morocco [††] INPT, Rabat Morocco

Summary
Networks becomes more complex practically in terms of offered services such as electronic commerce. As a result, networks are more and more subject to various kinds of complex security attacks. Existing security system responses have reached their limits in detecting and defending against various network attacks because current attacks are decentralized, automated and intelligent and these systems are passive in response to network attack in that they are limited to being local; and there is no automated, network wide response against detected attacks. Some drawbacks of existing systems reveal the necessity of designing a new generation of systems adapted to dynamical environment. In order to deal with these requirements, active networks approach provides interesting characteristics; it is a novel approach that gives networks and services flexibility and spontaneity. With an active network in place, we can build a more active and dynamic attack response by pushing the countermeasures near the source of attack. This paper describes this approach.

Keywords:
Active Networks, Control Architecture, Network Security.

Introduction

Networks becomes more complex practically in terms of offered services such as electronic commerce. Moreover, the number of individual users, government agencies and companies with Internet access is expanding rapidly. As a result, networks are more and more subject to various kinds of complex security attacks; Distributed Denial of Service (DDOS) shows such feature clearly [1]. It is known that the only way to secure completely a private network is to make it unreachable. However, even if this solution was undertaken for many years, today it is not possible to close private network especially for business purpose. Therefore, security management of these new networks requires more intelligence and sophistication. The focus of our work concerns one critical security management issue that is attack detection and response.

Existing security management system, such as intrusion detection system [2], firewalls [3] and Honeypot [4], have reached their limits in detecting and defending against various network attacks [5],[6]. This is because current attacks are intelligent [7], and these systems are passive in response to network attack in that there are limited to being local; and there is no automated, network wide response against detected attacks. These systems rely on manual response techniques involving network administrators. These drawbacks of existing systems reveal the necessity of designing a new generation of system adapted to dynamical environment. In fact, flexibility, adaptability and distribution are the main features to be addressed in a suitable architecture that fulfils these requirements.

In order to deal with these requirements, active networks approach provides interesting characteristics [8]. It is a novel approach that gives networks and services flexibility and spontaneity. It allows intermediate node to execute programs that are dynamically deployed on the network [9],[10]. The introduction of the active network approach seems so promising to enable network nodes to perform intelligent and dynamic behavior. This approach appears an appropriate candidate to make balance between requirements, flexibility and adaptability for attack detection and response.

With an active network in place, we can build a more active and dynamic attack response by pushing the countermeasures near the source of attack, where they can produce better results [24],[25]. The active network attack detection and response is automatic and effective in security management; but it is not yet realistic system and requires more research [26],[27]. In this paper we propose a flexible attack detection and response framework that is based on active network technology, which allows dynamically deploying and distributing security tasks among different nodes, such as attack information, response techniques and protection techniques.

The structure of this paper is organized as follows : in the next section, we examine some existing security services, including their advantages and drawbacks, and we present an overview of active networks technology. The section 3 describes our active security architecture for attack detection and active response. To demonstrate the potential of our solution, the section 4 presents a scenario of attack response. Finally we conclude and indicate some directions that could be followed in future work.

Manuscript received May 25, 2006.
Manuscript revised May 30 , 2006.

2 Background

Recent researches have studied security management specially defending against the network security attacks. Typical solutions to prevent attacks involve the set of the tow actions, detection of the attack and an efficient response to it. For detection of the attack, an evolution of the packet filters, abnormally based scanning and signature based detection would be a good start. For response to the attack, traffic blocking is most effective when applied near the source of the attack. Active network seems to be an efficient approach to improve attack detection and response capabilities [26],[27], but it is not a realistic system and require more research, since all the immediate nodes must be active. To remedy to this drawback we present an architecture that is partially active and compatible with existing networks.

2.1 Security Services

In this part, we present a short introduction to some existing security services, including their advantages and drawbacks. We choose Firewalls and Intrusion Detection Systems (IDS), because they have a relationship with our proposition.

2.1.1 Firewall

The primary function of the firewall is to control access to services and hosts. The firewall is based on security policies and is the main line of defense against attack between a protected sub network and a less trusted network. Its purpose is to restrict traffic entering and leaving at one carefully controlled point, and is responsible to apply the security policies defined, by controlling and restricting the traffic. The security policies are static rules (source address, destination address, protocol, port, etc) [3]. Generally, firewall cannot protect against new threats, viruses and malicious insiders, and does not provide dynamic reconfiguration.

2.1.2 Intrusion Detection System (IDS)

IDS attempts to detect and response to malicious activity targeted at computing and networking resources [2]. In general, IDS can be classified as Network Intrusion Detection System (NIDS) or Host Intrusion Detection System (HIDS). The fundamental difference between them is the source of the activity that they monitor and analyze to detect intrusions. HIDS monitors activities on a host or end system, while NIDS monitors network traffic. HIDSs are used to protect critical network servers or other individual systems containing sensitive information. NIDSs are used to monitor activities on a specific network segment. There are two evaluation methods to detect an attack, anomaly and signature detection. The anomaly detection observes network traffic for anomalies; it permit the detection of unknown attacks, but it generates false alarm rate. The signature detection scans network traffic for known attack traffic patterns. The IDS is usually passive in response to network based attacks; because the response has been an afterthought and is generally limited to logging, notification and disconnection at local host. The current IDS process neither provides the needed real time attacks response nor scales with network. In order to be effective in today's internet, network based attack response has to be network wide and automatic.

2.2 Active Networks

2.2.1 Definition

Active Network is a new concept, emerged from the broad DARPA Community in 1994-1995 [11]. It provides a new way of thinking about a network environment. Traditionally, the function of a network has been to deliver packets from endpoint to another, with only routing and header processing occurring within the network. In an active network, program can be injected into the devices making them active in sense that their behavior can be dynamically defined, and it can perform computations on individual packets. There are two approaches to building an active network, the discrete (or programmable), and the integrated (or encapsulated) approach.

The Discrete Approach (or programmable) approach: it consists to separate data packet processing from the injection of programs into the node; users send a program to a node as they would to a host; this program would then be stored at the node; when a packet arrives at the node, the corresponding program is selected using some header information and then executed. When a new version of the program is necessary, or if a different type of processing is required, the user can send new program to the node to replace the old one. ANTS [12]. DAN [13] and HAPA [14] are pursuing this approach.

The integrated (or encapsulated) approach: it consists to integrate the program with every packet. In this case every packet that is sent contains both data and programs. When the packet arrives at a node, its contents evaluated using the program in the packet. SmartPackets, Active IP Project [15] and architecture M0 [16] are pursuing this approach.

2.2.2 Node Architecture

The architecture of an active network node can be divided into three separate components [17],[18]: Node Operating

System (Node OS), The Execution Environments and Active Applications.

The Node OS is an operating system for active node. It is responsible for the management and allocating of resources like CPU and memory in the active node. It also provides a set of interfaces to the Execution environment to access and use these resources. There are several efforts in the building of a generic Node OS – Bowman [19], Joust [20] and JanOS [21].

The Execution Environments are environment for execution of active applications. There are one or more Execution Environments per node; each Execution Environment accepts valid programs and packets, either executing them or modifying their state, and emits one or more new packets or programs. The Execution Environment is essentially the active network's programming environment and is often centered on a particular language or model.

The Active Applications are code injected in network and executed at an active node. This code may come from a provisioned library, a capsule or fetched on demand.

3 System Architecture

In this section, we will describe the components of the system that will support our approach. This system is an attacks detection system and also a response system. Its goal is twofold :

1. Detect ingoing attacks by monitoring the traffic.
2. Response to the attacks in two ways:
   a. Response locally by taking the initiates countermeasures.
   b. Response in the network by pushing the global countermeasures nearby the attack source (Fig 1).

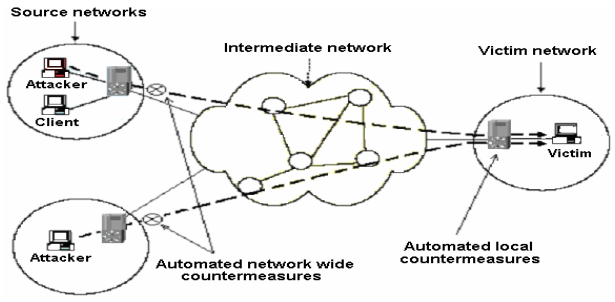

Fig. 1. Dynamic, Network wide attack response

Ideally, attacks should be stopped as close to the sources as possible. Push the responses at the attack source have several advantages :

- Congestion avoidance: restraining attack streams near the source preserves Internet resources that are usually overwhelmed by the attack traffic. This reduces overall congestion and increases resources available to legitimate users.
- Small collateral damage: many response systems respond to the attack by filtering or rate-limiting all traffic to the victim. Legitimate traffic thus suffers collateral damage. Moving attack response closer to the sources reduces the range of legitimate traffic adversely affected by the response, as the traffic from uncompromised source networks proceeds to the victim unhampered.

The architecture that implement our approach is composed of the following two main parts : node architecture and network architecture.

3.1 Node Architecture

It is the architecture of the security module installed on the host to be protected. It is composed of some components, as depicted in Fig 2:

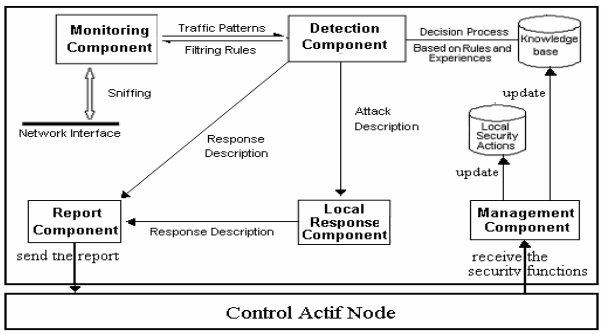

Fig. 2. Node System Architecture.

3.1.1 Monitoring Component (MC)

This component consists in providing a report of network traffic, and transmits it to the Detection Component, i.e. builds and gives a representative view of detailed data that passes through the network interface. It operates by filtering and analyzing the traffic data stream, based on the filtering rules provided by the DC. This report is used to identify the attack and to determine appropriate security actions to defend against it.

3.1.2 Detection Component (DC)

This component is responsible for detecting and identifying attack against the node. It receives the traffic report form Monitoring Component (MC); this report provides a global picture of the data stream. The decisions

process of DC is based on security rules and expertise stored in its knowledge base. Some attacks are violation of these rules. The role of the DC then is to identify these security violations and recognize the attacks that can occur by using the signature analysis.
In addition, DC provides a history traffic patterns and can support historical queries; this allows a baseline to be established, from which anomalies can be detected and suspicious activity identified.

3.1.3 Local Response Component (LRC)

This component is responsible for generating adequate responses, it defines the reactions when an attack is detected; i.e.: defines the security actions that must be taken locally in this attack situation. It is responsible for taking the appropriate local security actions (e.g., kill processes and connections associated with attacks, install filtering rules, disable the user account, modify a host's policy) based on the attack type and local policy constraints (e.g., never disable HTTP between 8 h and 18 h).

3.1.4 Report Component (RC)

This component is responsible for building and sending a report that contains attack description and description of the node's response to the manager of the network called Control Active Node (CAN). On detecting an attack, the node determines the appropriate response based on the attack type; the vulnerability of the component under attack and on the local policy constraints. The description of the attack responses are added to the attack description prior to sending the report to the CAN. This report enables the CAN to gain a better largely picture of the situation; and to determine an optimal global response.

3.1.5 Management Component (MC)

This component performs an initial check each new arrived security function. If the security function passes this check, then the MC integrates it into the system. A security function can for example include signature detection functionality, anomaly detection algorithms or any kind of security action.

3.2 Network Architecture

3.2.1 Active Plans

The components of an active node (NodeOS, Active Applications and Execution Environments) interact together to ensure the dynamic deployment and the execution of the actions response to attacks. To keep a compatibility with the traditional network, the active nodes will be deployed only on some specific point of the network. Therefore we will have to manage a network that formed of two principal plans: active plan and transport plan [22],[23]. In order to control and manage the dynamic deployment of the active code, we introduce a third plan called control plan, and a fourth plan called administrative plan. This architecture in plans, shown in the Fig 3, achieves these goals by the introduction of two new types of active nodes : the Control Active Node (CAN) the and Administrative Active Node (AAN).

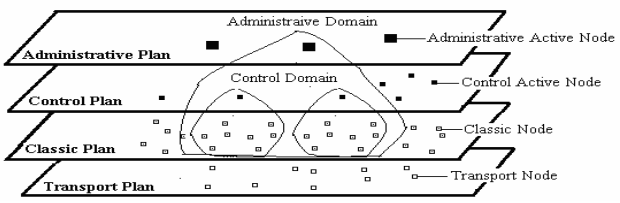

Fig. 3. Active plans

By making a regrouping of the nodes according to their functionalities we obtain :

- Administrative Plan : this plan gathers the administrative active nodes. An administrative active node is responsible for managing and administering the control nodes in its administrative domain.
- Control Plan : this plan gathers the control active nodes. A control active node is responsible for controling the dynamic deployment of the active security module in its control domain.
- Classic Plan : this plan gathers the classic nodes; that can be a server or a simple client.
- Transport Plan: this plan gathers the transport routers, with transport the packets without any treatment.

3.2.2 Administrative Domains and Control Domains

In order to control and allow easy and dynamic deployment of the active actions response, we propose a logical architecture of control. This architecture divides the network into administrative domains and control domains. A control domain is a trusted network area identified by a CAN, whereas an administrative domain is a networked area that is controlled and administrated by an AAN. The NAC is a programmable node that can manage and control security systems, such as detect a breach of security, and the system response to the attacks. The CAN makes a decision about the security policy and distributes it to the security systems within its control domain, and is responsible to control the dynamic

deployment of active security module in its domain. The administrative domain is a network area that includes many control domains. It is controlled and administered by the AAN. The AAN is a security system that is managed by an authorized organization. It is responsible for distributing security patches or updating related security software in its administrative domain. Fig 4 shows that control domains A and B are independent, and they are included in administrative domain C. The CAN or the AAN is an active node that can adapt itself to different states in runtime. It can dynamically change its own functions.

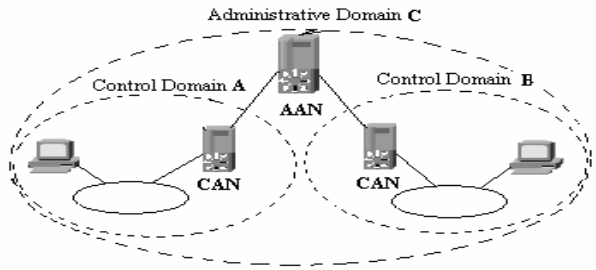

Fig. 4. Administrative Domain and Control Domains

3.2.3 Role of an Administrative Active Node

The AAN is responsible for administering and controling the entire CANs in its administrative domain. It is responsible for updating the security functions in its administrative domain. Furthermore this node can play the role of an authority of certification, whose public key must be distributed to all CANs which want to receive the authenticated active code.

3.2.4 Role of Control Active Node

The CAN is both a central location for receiving a copy of all active reports sent by nodes in its control domain, and it is also a central component that can extend and direct the response to any component that it controls. It can make better global decision for attack response. The CAN is composed of the following components (Fig 5) : the global response component, the dynamic deployment component and the control management component.

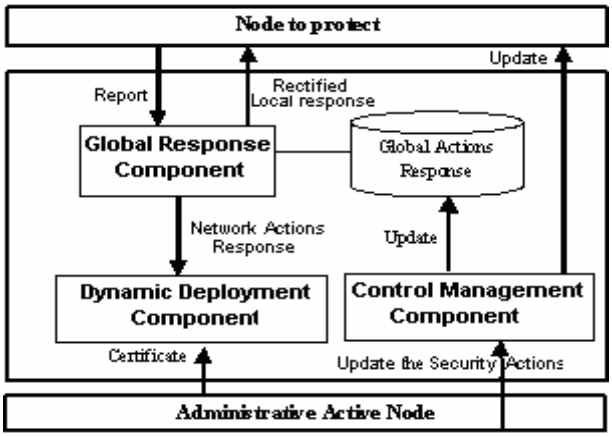

Fig. 5. Control Active Node Architecture

This central component will have the information needed to assess coordinated and distributed attacks. In addition, the CAN can only affect the behavior of devices within its domain. It can request other CANs to prove additional responses. Furthermore this node is responsible for answering to the requests of the execution of active code coming from other CANs. Moreover it will answer only the authenticated and authorized requests coming from CANs. The components of a CAN are described in the following paragraphs.

3.2.4.1 Global Response Component

This component is responsible for generating the global actions response. The report of attack and local response description, enables the CAN to gain a better largely picture of the situation and to determine an optimal global response. The global response can be either a rectification of the local response or a network action response (Fig 6).

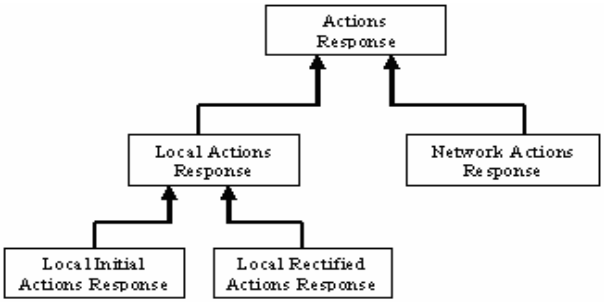

Fig. 6. Actions response structure

Once this component has determine an optimal response, it sends directives of the rectifications back to nodes whose response requires altering or removing an unnecessary response (e.g., open up a service that was blocked at firewall), or take other security action (e.g., extend the

duration of a blocking rule). The CAN may request local security action, such as disable a user account or modify a host's policy. The network actions response are sent to the Dynamic Deployment Component.

3.2.4.2 Dynamic Deployment Component

This component is responsible for converting the network actions response into active code, and then deploy it dynamically in the network nearby the attack source. The active code is a program that can be written in different programming language, and can be executed in different execution environments. During the deployment, nodes along the network receive and execute the active code, and possibly return values or forward it along the others nodes. The active code is deployed into the network and reprograms and reconfigures nodes and routers.

4 Attack Detection And Response Scenario

In order to illustrate our approach, we present here a scenario of attack detection and response (Fig 7 and Fig 8). The topology of the network consists of two administrative domains (AD), each ones contains two control domains (CD), and each CD contains several classical nodes as shown in Fig 7. In normal operation the monitoring component filters the traffic and sends the traffic report to the detection component.
When the attacking host initiates an attack from the CD2 in administrative domain A, to the server in CD6 in administrative domain C; the system reacts as depicted below :

1. The detection component detects the attack by knowing its signature. Then it contacts the response component and provides it by the attack description that was encountered (IP address, TCP port, etc.)
2. On detecting an attack the response component generates and executes the attack response locally. e.g. Block the malicious traffic from the specific IP address.
3. The report component generates the report that contains the attack description and the response description; then sends it to the control active node CAN 1 (step 1).

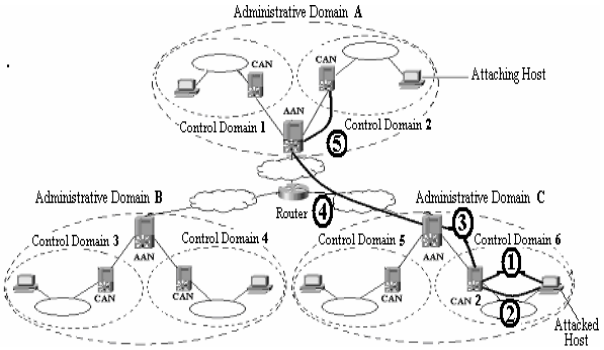

Fig. 7. Topology of the network

4. The CAN 6 receives the report; analyses it and make global response for this situation. Then sends back the rectified local response as active security module to attacked host (step 2), and pushes the active code corresponding to the network response nearby the attacking host, where it can produce better results (step 3, 4, and 5). The active code is deployed dynamically by using the active network infrastructure to their destination CD2. After passing successfully the authentication and the authorization mechanism, it can change the node configuration if it has the permission to do that. The malicious traffic is thus blocked near its source and can't no more reaches the attacked host.

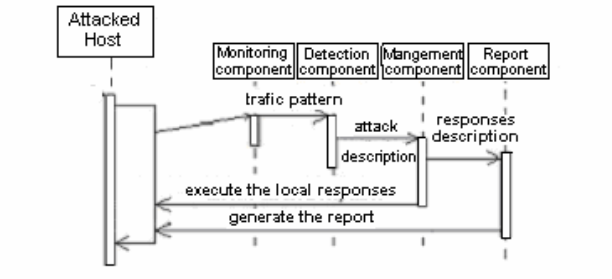

Fig.8-a. Monitoring, Detection, Management and Report functionalities.

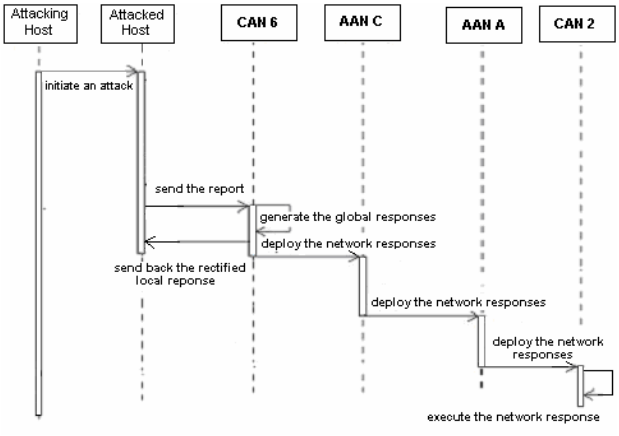

Fig.8-b. Local and network respons functionalities.

5 Conclusion

The active network approach has initiated an intense effort of investigation in a number of networking areas. This paper presents our ideas and thoughts on future network security by using advantage and properties of active networks. We have presented our approach architecture which introduces the notions of control domain, administrative domain, control active node and administrative active node.. This architecture provides automatic, dynamic and network wide responses against network attack. The principal idea is to dynamically deploy the attack responses on the network which must support active nodes. The active responses are free to move in the network where they can produce better results. To illustrate our architecture, we have presented a scenario of attack detection and response. As a perspective we will continue to deepen the concepts and the notions of this architecture and to proceed after to its implementation it in order to validate it.

References

[1] Carnegie Mellon University, CERT Incident Note IN-99-07: “Distributed denial of service tools”, CERT/CC, 1999. Available: http://www.cert.org/incident notes/IN-99-07.html.

[2] B G. Helmer, J. Wong, V. Honavar, L. Miller, “Lightweight agents for intrusion detection”, J. Syst. Software, Iowa State University, November 2000.

[3] D.chapman, D.Zwicky, “Building Internet firewalls”, O’Reilly & Associates, INC. 1996.

[4] The Honeynet Project. “Know your enemy”. Addison Wesley, 2001.

[5] C. Kaufman, R. Perlman, M. Speciner, “Network Security: PRIVATE Communication in a PUBLIC World”, Prentice-Hall, Englewood Cliffs, NJ, 1995.

[6] S. McClure, J. Scambray, G. Kurtz, “Hacking Exposed Network Security Secrets and Solutions”, McGraw-Hill, New York, 1999.

[7] K. Houle, G. Weaver, “Trends in Denial of Service” v1.0, Report of the CERT/CC, October 2001. Available at : http://www.cert.org/archive/pdf/DoS trends.pdf.

[8] R. Boutaba, A Polyrakis, A. Fernandez Casani, “Active networks as a developing testing environment for network protocols” in Annals of telecommunications, N°5-6, 2004.

[9] A. Jeffrey, I. Wakeman, “A survey of semantic techniques for active networks”, School of Cognitive and Computing Science, University of Sussex, 1997.

[10] D. Raz, Y. Shavitt, “Active networks for efficient distributed network management”, IEEE Commun. Mag. 38 3 2000 138–143.

[11] D.Tennehouse. J.Smith. W.Sincoskie. D.Wetherrall. G.Minden. “A survey of Active Network research”. In IEEE communications Magasine.35.n 1 pp.80.86. January 1997.

[12] D. Wetherrall, “ANTS: A tool kit for Building and Dynamically Deploying Networks Protocols”, In IEEE openarch 98, San Francisco April 1998

[13] D. Decasper, B. Platttner,”DAN: Distributed code caching for Active networks”, Proc. IEEE INFOCOM ’98, San Francisco, CA 29 March-2 April 1998.

[14] R. Kilany, A. Serhrouchni, “Using Distributed compenent Model for Active service Deployment”, ISCS 2002.

[15] D. Wetherrall, D. Tennehouse,”The Active-IP option “ in the 7th ACM SIGOPS European workshop.

[16] A. Banchs, “Multicasting Multimedia Streams with Active Networks”, ICSI technical report 97-050

[17] K. Calvert, S. Bhattacharjee, E.W Zegura, J. Sterbenz, “Directions in active networks“, In IEEE communications.1998.

[18] K. Calvert, “Architectural Framework For Active Networks”, v1.0 July 27, 1999; Available at : http://www.cc.gatech.edu/projects/canes/papers/arch-1-0.pdf

[19] S. Merugu, S. Bhattacharjee, E. Zegura and K. Calvert, "Bowman: A Node OS for Active Networks", presented at Proceedings of IEEE Infocom, 2000.

[20] J. H. Hartman, L. L. Peterson, A. Bavier, P. A. Bigot, P. Bridges, B. Montz, R. Piltz, and T. A. Proebsting, "Joust: A Platform for Liquid Software", IEEE Network Magazine, special issue on Active and Programmable Networks, vol. 32, pp. 50-56, 1998.

[21] P. Tullmann, M. Hibler and J. Lepreau, "Janos: A Javaoriented OS for Active Networks", IEEE Journal on Selected Areas of Communications, vol. 19, 2001.

[22] R. Kilany, M. Riguidel, A. Serhrouchni, D. Zebiane, “A control Architecture for Active Network”, SoftCom2001, Available at : http://www.fesb.hr/SoftCOM/2001/

[23] R. Kilany, D. Zebiane, M. Riguidel, A. Serhrouchni, “A control architecture for ANTS”, ifip Workshop on IP and ATM Traffic Management WATM 2001 and EUNICE 2001

[24] D. Sterne, K. Djahandari, R. Balupari, W. Cholter, B. Babson, B. Wilson, P. Narasimhan, A. Purtell, D. Schnackenberg, S. Linden, “Active network based DDoS defense”, in: Proceedings of the DARPA Active Networks

Conference and Exposition (DANCE'02), San Francisco, CA, 2002, pp. 193–203.

[25] D. Schnackenberg, H. Holliday, R. Smith, K. Djahandari, D. Sterne, "Cooperative Intrusion Traceback and Response Architecture (CITRA)", in: Proceedings of the Second DARPA Information Survivability Conference and Exposition (DISCEX II), CA, June 12–14, 2001.

[26] S. KARNOUSKOS "Community aware network security and a DDOS response system". in Annals of telecommunications, Tome 59, N°5-6, may/june 2004, pp. 525-542.

[27] A.Hess, M.Jung, and G. Schafer. "FIDRAN: A flexible Intrusion Detection and Response Framework for Active Network". In Symposium on computers and communications 2003.

Ahmed EDDAOUI received the DESA degree in computer sciences Engineering from Faculty of Science of Rabat, Morocco in 2003. He is preparing his PhD degree in the field of networks security using active network approach.

Abdellatif MEZRIOUI received the PhD degree in the field of software process modelling from EMI Rabat, Morocco in 2001. He is a professor at the INPT since 1995. His actual reserch domains are networks security and software process modelling.