A Performance Analysis of Chasing Intruders by Implementing Mobile Agents

Omid Mahdi Ebadati E omidit@gmail.com
Dept. of Computer Science
Hamdard University
New Delhi, India

Harleen Kaur harleen@jamiahamdard.ac.in
Dept. of Computer Science
Hamdard University
New Delhi, India

M. Afshar Alam aalam@jamiahamdard.ac.in
Dept. of Computer Science
Hamdard University
New Delhi, India

Abstract

An Intrusion Detection System in network fetches the intrusions information from systems by using Mobile Agents aid. Intrusion Detection System detects intrusions based on the collected information and routes the intrusion.
The intelligent decisions on communications permit agents to gain their goals more efficiently and provide more survivability and security of an agent system. The proposed model showed a formal representation of information assurance in agent messaging over a dynamic network by probability of redundant routes.
The proposed Intrusion Detection System, chase intruders and collect information by Mobile Agents. Our proposed architecture is an information exchange method and chasing intrusion along with a method by implementing Mobile Agents.

Keywords: Intrusion Detection System (IDS), Hybrid Intrusion Detection System (HyIDS), Mobile Agent (MA).

1. INTRODUCTION

Computer network attacks generally are divided into two types: break in from outside of local area network and the other attack from inside the local area network. Therefore, it is rarely happen in either case that intruders can directly attacks from their own host [1]. The reason is obvious: intruders want to conceal their origin.
Intruders prefer to attack the minimum secured hosts first, then smoothly approaching secured host with stronger protection, consequently try up to reach their target hosts. Typically the users or administrators do not notice about intrusion attacks, either on the target hosts or the intermediate hosts, and the administrators cannot chase the origin of an intrusion even the intrusion has been detected or the network connection has logged out. Intrusion Detection has

the ability to analyze data in real time to detect the intruder and block the attacks when they occurred.
There are various types of Intrusion Detection Systems and each one has different activity and different ways of detection. Practically detecting the intrusion is more complex than a simple definition.
With the Mobile Agents over a network environment that support it, it can be penetrated possibly harmful agents that called intruders. By increasing the damages of intruder's, motivation of a large amount of research on detection part especially by Mobile Agent team, also increased. The Mobile Agent team, deploy their chasing over the network just by detection of an intruder.
Here, we are defining the different types of Intrusion Detection System:
A Network Intrusion Detection System (NIDS) which identify the intrusion by monitoring the network traffic; it has an independent platform of detection [2] and can monitor multiple hosts. Network Intrusion Detection Systems approach access to network traffic by connecting to a hub network or switch, to configure the mirroring port, or network tap such as a snort, which is an example of NIDS.
A Protocol based Intrusion Detection System (PIDS) is based on communication protocol between different connected devices, such as user or system and the server. PIDS has agents that settle down on server front-end that analyzes and monitor the network traffic.
Other type of IDS is Application Protocol based Intrusion Detection System (APIDS). The APDIS is based on a system or agents that located within a number of servers, and try to analyze and monitor the communication on specific application protocols.
A Host based Intrusion Detection System (HIDS) is basis on an agents on a host [12,13,14] that identify intrusions by various measurement like file system modifications such as binaries, password files, analyzing system call, application logs and other host activities and states.
The last model of Intrusion Detection System is Hybrid Intrusion Detection system (HyIDS). HyIDS is a combination of two or more approaches of Intrusion Detection Systems. In case of implementing a combine specification of intrusion detections, we can have a new comprehensive view of network detection.

2. RELATED WORK

Using Mobile Agents in Intrusion Detection System being conducted by various researchers including Wayne Jansen, Peter Mell, which worked on reification of their own model on specific network [15], Mohamad Eid, proposed immunity components [16], Omid Mahdi Ebadati, Harleen Kaur, M. Afshar Alam, proposed secured route in NIDS [17], Christopher Krugel, Thomas Toth conducted their work on Sparta and Micael [18], Shunji Okazawa, Midori Asaka, Atsushi Taguchi, proposed detection in an intermediate node [19], Wayne A. Jansen, proposed a new design with more robust [24].
By introducing the intrusion detection to the industry, it is become a part of detection technology, and different companies commercially perfecting the existing intrusion detection techniques. Finally the research part focuses on the most unsealed part that those are:
1- Mechanisms of attack response.
2- Architecture for distributed Intrusion Detection Systems and standards of intrusion detection inter operability.
3- New paradigms for performing intrusion detection by Intrusion Detection System.

3. INTRUSION DETECTION SYSTEM ARCHITECTURE

In a large scale to deploying an Intrusion Detection System over a network, network traffic will be ultimately high though the high volume of the system logs will regularly transferred, in this case large number of information which has been collected, are unrelated to intrusion. Therefore in such cases Intrusion Detection System on a large network environment does not have efficient functionality. For the solution of this problem we adopted a Mobile Agent paradigm in developing IDS. Mobile Agents autonomously [10] route to the target systems for just collecting the intrusion related information and eliminate the requirement of transferring logs to the server.

Through the TCP/IP protocol Intrusion Detection System can deploy on the local area network. The proposed Intrusion Detection System typically consists of various elements like a sensors, administrator, notice boards, flag/message boards, information collector agents, and chasing agents.

Information collector agents— Information collector agents during the normal process gathering the information from the specific network. This information which has been collected may contain intrusion also but the recognition of this matter is unrelated to this part.

Administrator— The administrator of intrusion detection has the duty of recognition of intrusions. After information collector agents gathered the information over the specify network, the administrator try to detect the intrusions among them. The administrator manages the notice boards and mobile agents and tries to making interface between system and administrators. The detection part according to the information weight which entered by agents on the notice board and exist in manager accumulator can find out the intrusion. If the weights are more than a set threshold administrator concludes that it is an intrusion.

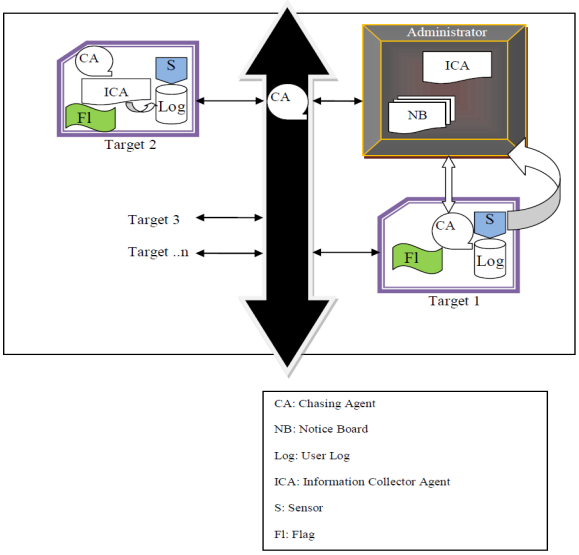


FIGURE 1: Intrusion Detection System Architecture

Notice boards and sensors— Notice boards by the aim of sensors try to specify the information and get direction to the administrator.

The sensor on each agent by logging which conduct by system monitoring send the report of collected logs to the administrator.

Chasing agent— Chasing agent is specifically to route and trace the path of intrusion and try to locate the origin of intrusion. This chase can be done by detection of remote logged on, to the target host. The compromised nodes in network can detect by chasing agent.

Intrusion Detection System functionality— The speed of detection part which depend on platform of detection and different algorithm that can use in the system shows the performance of IDS. If the functionality of IDS wants to be acceptable, speed of detection and reliability of that must be increase. The requirement performance of IDS can depend on, real time detection of anomalous event and security breaches and an immediately report is necessary to reduce the corruption and loss on the network to have the confidential data.

The Intrusion Detection System should not busy by different deployment on the system and not to interfere with the normal operation on it. The important thing is that, the agents aware of the consumption of network resources, and the tradeoff between additional levels of security that monitoring the network and checking the agents performance.

Scalability of Intrusion Detection System due to the new computing devices which added to the network is another issue of performance, and IDS must be able to handle the communication load and these computations [8].

4. FUNCTIONALITY OF IDS IN MOBILE AGENTS

There is a certain platform of Mobile Agent that can work on variety of hosts and systems. In this platform not all but at least hosts and network devices must be installed with Mobile agent platform. Mobile Agent platform give the capability of implementation various applications and simply can assume companies and organizations [9]. Mobile Agents are not preinstalled on all the systems because, it has not that much popular to be, and so if it is, it needs to contrast for IDS scheme and assume on host based Intrusion Detection System. Generally this scheme is too expensive and it is unusual to install Mobile Agent platform or Java virtual machine on all hosts for a general purpose.

Mobile Agent Merits

Previously Mobile Agents have not the ability to move in different environments. As virtual machines and interpreter has been introduced, it got the possibility for heterogeneous environment and platform but however there is a limitation support for preservation and resumption of the execution state.

Mobile Agents although carry a deal of autonomy and role well in operating disconnection and the failure of platforms or home platform, that the agent confidentially provide the security services and reduce their functionality.

The fault tolerant of Mobile Agent can be increased when agents moving from a machine to another machine therefore, the Mobile Agents always working on the trusted machine and restrict on its functionality, so its environments will be a trusted platform and a safe home.

Mobile Agents platform designers mostly face tradeoff between fault tolerance and security. An example of security risk can be multi hop in Mobile Agent that cause of different types of architecture which have been built in client server centralization models that need those agents return to a central server before moving to another machine/host machine [20]. Even though having a security risk of Mobile Agents shows the vulnerable and failing of MA for central server.

There is some keynotes about Mobile Agents like network latency over coming, network load reduction, execution and autonomy asynchronization, composition and structure, scalability, dynamic adoption, fault tolerance behavior and robust, hydrogenise environment operations.

Mobile Agents have many capabilities to enhance with Intrusion Detection System technology. The first obvious one is mobility which is the most important characteristics of MA. Agent lending capability to Intrusion Detection System can be another one. Agent application and technology mimic collections of intelligent individual and autonomous. Individual classes have special applications and each class can operate individually and independently than each other and each one individually exchange the information and talk to other when they meet individually. This paradigm is clearing the traditional programming where Master Logic Unit controls slave unit that have not autonomy and work according to command of Master Logic Unit.

Traditional approach which work on multiple units with set communication channel and set duties cannot response well, suppose each unit rely on other unit to perform their job and there is not central controller. In case of a unit seas the other unit are not enough intelligent to function and sort out the problem. The traditional distribution programming paradigm upon function reliability to component can only work well, and even by using redundant components a small number of back offs can attack and make it disable with attackers. Even though the implementation of the traditional design and solution can efficiency solves many problems. A great contrast to traditional design is agent technology that attempt to give understanding of environment along with independently decision making and authority to each agent.

Chasing Agents on Network Environment

The agents take their way and decide their destination and their path to them by use of notice board. Chasing agents perform as follow:

First of all, agents know that on each system there is a process and user ID, therefore dispatching process to a target system by chasing a process ID [11] is the first perform.

Secondly, the chasing agent tries to find out the logs on target systems, if it detects any, it will determine its destinations from the information on user's log.

Thirdly, referring to the notice board on the target system is the next refer of chasing agent.

Fourthly, it will check whether on notice board there is any information related to login session that intend the agent to chase, if there is any related to that session on the notice board and its already exist, it shows that another agent has already tried and traced this user, so the chasing agent update the notice board, and returns to the administrator.
Fifthly, if there is no information pertains to login session, the agent enters to such information and tries to get into the target system from the user logged. If the agent comprehensive that the origin of intrusion is that the target system which has been chased, the chasing agent return to the administrator.
Sixthly, if not, the steps one to six again should repeat by the agent.

Notice board information
The information can enter by chasing agent on the notice board of target system are: name of the system, process ID, the user logs timing, leaving process ID or the name of target and where it has been chased by agent, chasing agent ID, chasing agent trace time.
In case, if the chasing agent wants to discontinue a trace, on the notice board it should be enter its ID and time of performing. It will help when the agent return to the administrator, because the chasing route and the reason has been logged already.
For above case (agent's return) we can have three possibilities and they are: if the chasing is not perform so the intrusion cannot chase, mostly is the cause of not installation the Intrusion Detection System on the following system, the other reason can be whether the system itself is the origin of intrusion and the last can be chasing that intrusion by another agent.
As it has mention above also, if another agent is chasing the intrusion the concurrent agent send the name and ID of that agent of the following system to the administrator. The trace part is shown in Figure 2.

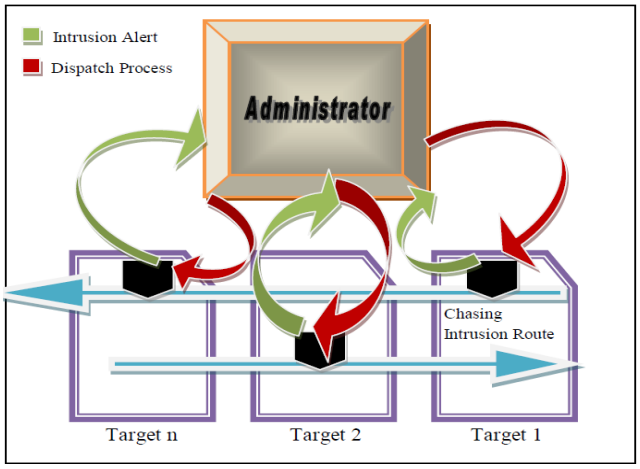


FIGURE 2: Intrusions Chasing

5. EXPERIMENT AND EVALUATION

Optimal Time Algorithm for Contiguous Search in Trees
This section is devoted to the description of a linear-time algorithm returning an optimal contiguous search strategy for weighted trees.
Let T = (V, E) be a tree with n nodes. The quest for a minimal strategy can be restricted to monotone strategies satisfying the "single entry point" constraint, i.e., initially, at time *t = 0*, all searchers are in the same node x_0 called home base, and the first step consists to clear a link incident to the home base. The choice of the home base a_ects is the number of searcher of monotone strategies. Hence, let $cs_x(T)$, called search number of T from node *x*, denote the minimum number of searchers of monotone strategies starting from $x \in V$.
$cs(T) = min_{x \in V} \{cs_x(T)\}$ [23, 21, 22]. Any monotone strategy will induce an ordering on the neighbors (and thus on the incident links) of each node, where the ordering depends on whether one neighbor becomes occupied (and thus safe) before another. This ordering depends solely on

the choice of the starting node x, and corresponds to the parent/child relationship in T_x where T_x denotes the tree T when rooted in node x. Hence, in the following, we will refer to a strategy for T starting from x also as the strategy for T_x, and alternatively denote $cs_x(T)$ by $cs(T_x)$.
Given a rooted tree T_x and one of its nodes y, let $T_x[y]$ denote the subtree of T_x rooted in y, and let $cs(T_x[y])$ denote the contiguous search number of $T_x[y]$ from y. The most important property of monotone strategies is given by the following lemma.
Let $y_1, y_2; : : : ; y_k$ be the $k \geq 2$ children of y in T_x, and assume, w.l.o.g., that $cs(T_x[y_i]) \geq cs(T[y_{i+1}])$ for all $i < k$. Then $cs(T_x[y]) = max\{cs(T_x[y_1]; cs(T_x[y_2]) + \omega(y)\}$:
Proof. Clearly $cs(T_x[y]) \geq cs(T_x[y_1])$, otherwise $T_x[y_1]$ cannot be cleared. If $cs(T_x[y_1]) > cs(T_x[y_2]) + \omega(y)$, then $cs(T_x[y_1])$ searchers suffice to clear $T_x[y]$ by visiting y_1 last among the children of x, and by letting $\omega(y)$ searchers occupying node y while the other subtrees are visited. Indeed, from Eq. 1, $\omega(\{y,y_i\}) \leq \omega(y_i) \leq cs(T_x[y_i])$ for every i.
Hence, let $cs(T_x[y_1]) < cs(T_x[y_2]) + \omega(y)$. Let β be a contiguous search strategy which uses $b = cs(T_x[y_2]) + \omega(y) - 1$ searchers to clear $T_x[y]$. If $Tx[y_2]$ is cleared before $T_x[y_1]$, while the $cs(T_x[y_2])$ searchers are clearing $T_x[y_2]$, y will be occupied by at most $\omega(y) - 1$ searchers, and incident to $\{y,y_1\}$ which is contaminated. Thus, β does not satisfy the condition of Theorem 1. Similarly, if $T_x[y_1]$ is cleared before $T_x[y_2]$, then, while the $cs(T_x[y_1])$ searchers are clearing $T_x[y_1]$, at most $\omega(y)-1$ searchers will be at y since by definition $cs(T_x[y_1]) \geq cs(T_x[y_2])$, and $b = cs(T_x[y_2])+ \omega(y)-1$.
Since y is incident to $\{y,y_2\}$ which is contaminated, y becomes unsafe, and β is hence not monotone. Therefore, strictly more than $cs(T_x[y_2])+ \omega(y)-1$ searchers are required for a monotone strategy if $cs(T_x[y_1]) < cs(T_x[y_2])+ \omega(y)$. On the other hand, $cs(T_x[y_2])+ \omega(y)$ searchers clearly suffice to clear $T_x[y]$ by visiting y_1 last among the children of y, since at least $\omega(y)$ will stay at y making it safe while the other subtrees are visited. Links incident to y are cleared since, again from Eq. 1, $\omega(\{y,y_i\}) \leq \omega(y_i) \leq cs(T_x[y_i])$ for every i.
A straightforward application of Lemma allows to compute $cs(T_x)$ in $O(n)$ time, resulting in an $O(n^2)$ time algorithm for computing $cs(T)$ for any tree T. We show that the complexity can be reduced to $O(n)$, and more importantly that a minimal search strategy can also be computed in linear time.

Search /* Returns a strategy α */
(a) Set $\alpha = \emptyset$; /* empty sequence */
(b) Choose a node x such that $cs(T_x)$ is minimum;
/* start with $cs(T)$ searchers at the homebase*/
Let $y_1; y_2; : : : ; y_{deg}(x)$ be the $deg(x)$ neighbors of x where, w.l.o.g., $cs(T_x[y_i]) \geq cs(T_x[y_i+1])$;
(c) For $i = deg(x)$ down to 1, apply
Move$(x, y_i, cs(T_x[y_i]))$;
Move (u, v, q) /* searching a subtree rooted at v */
(1) $\alpha = \alpha|(u, v, q)$ where | denotes the concatenation operation; /* transfer q searchers from u to v */
Let $w_1; : : : ;w_{deg(v)-1}$ be the $deg(v) - 1$ children of v in T_x where, w.l.o.g., $cs(T_x[w_i]) \geq cs(T_x[w_{i+1}])$;
(2) For $i = deg(v) - 1$ down to 1, apply
Move$(v, w_i, cs(T_x[w_i]))$;
(3) $\alpha = \alpha \mid (v, u, q)$; /* return q searchers from v to u */

Attack Detection

The goal of Intrusion Detection System is detecting intrusion as many as possible because always detecting all intrusions is not possible, and IDS tries to precise the route of chasing of intrusion, to make it efficiently.
By increasing the internet accessibility, intrusions and cracking tools that distributed on the internet the appropriate evaluation of IDS is much difficult to examine. Another important factor that classifies the internet connection also is bandwidth of connected internet network. In proposed Intrusion Detection System we try to obtain such cracking tools that aimed at local attacks on the net and because of our limitation in trend of cracking tools which are available, we just try with Linux Fedora operating system machines with the limit number of network speed (less than six hundred Kbps). Our evaluation results has been tested the buffer overflow %54,

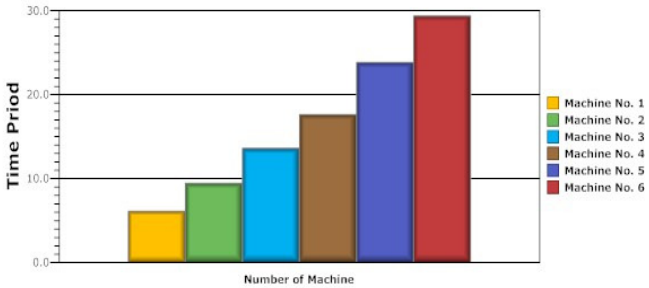
password sniffing %7, route shell execution %3, file mode changing %2, file creation %27.4, routing %2.5 and administrator password sniffing %4.

Mobile Agents Performance Analysis

To initial a tracing and established a performance of an agent the Intrusion Detection System should determine the number of trigger. Generally the number of chasing agent and number of occur trigger on the administrator are same. In our investigation rate the whole events per day it was around (39 in 105617), so the rate of 0.000369 events per day is investigated on our machines. The average size of chasing agent excluding the information is 1.9 KB and 2.3 KB including the information for information collector agents and 4.1 KB for chasing agent.

The time period since a sensor trigger till chasing agent pertaining that return to administrator contained in an intrusion route in each case, there is a number of target system which is measured. Time period for agents' authentication also is calculated and it shows that including encryption and authentication, it takes around 1.4 times longer than when we have only authentication. The measured time is contain the process of chasing agent on each target and the transportation time period between targets and agent which become too short by using proposed algorithm, and it is around 0.07 second.

The chasing agents round trip, time period on a number of machines is as follows (Figure 3).



	Number of Machine
Machine No. 1	6.114
Machine No. 2	9.452
Machine No. 3	13.601
Machine No. 4	17.625
Machine No. 5	23.845
Machine No. 6	29.364

FIGURE 3: Chasing Agent, Round Trip-Time Period Diagram

6. CONCLUSIONS

Mobile Agent technology offers much related field in Intrusion Detection System. The autonomous components and MA seem an obvious usefulness in IDS and other applications as well.

However practically, hardly we can find out the technology of Mobile Agent and its beneficiary, but the technology is provided valuable capabilities and although in practice moves a running program from a platform to another one, without Mobile Agent system face us the barriers.

Mobile Agents can enter main stream by shows its performance, security, emerging technology, and widely use in different hardware platforms.

In this paper we described how intrusions chased by emerging Mobile Agents and Intrusion Detection System. However Intrusion Detection System does not indiscriminately in entire the network collect the information, but collect pertain information by chasing the logged in user.

The proposed architecture of Intrusion Detection System by implementing chasing route optimal time algorithm is briefly explained in this paper. The round trip time on a number of machines for chasing the agent shows that the round trip type is not that, much increased when the numbers of machines are increased, and the ramp of increasing is not as same as number of machine's ramp.

Proposed Intrusion Detection System does not collect unrelated information to intrusions and just focused on the amount of information which is gathered by agents, and administrator which has been approved them, consequently it will help to reduce the using of the system.

7. REFERENCES

[1] W. R. Cheswick, S. M. Bellovin. "Firewalls and Internet Security: Repelling the Wily Hacker," Addison Wesley Publishing Company (1994)

[2] Northcutt, S. Intrusion Detection - The Big Picture. SANS Institute, 1999, www.sansstore.org.

[3] Whitman, Michael, Herbert Mattord. Principles of Information Secuirty. Thomson, Canada, pp. 290-301 (2009)

[4] Anderson, Ross. Security Engineering. Wiley, New York, pp. 387-388 (2001)

[5] Kohlenberg, Toby (Ed.), Alder, Raven, Carter, Dr. Everett F. (Skip), Jr., Foster, James C., Jonkman Marty, Raffael, and Poor, Mike, "Snort IDS and IPS Toolkit," Syngress (2007)

[6] Barbara, Daniel, Couto, Julia, Jajodia, Sushil, Popyack, Leonard, and Wu, Ningning, "ADAM: Detecting Intrusions by Data Mining," Proceedings of the IEEE Workshop on Information Assurance and Security, West Point, NY (2001)

[7] M. Asaka, S. Okazawa, A. Taguchi and S. Goto. A method of tracing intruders by use of mobile agents. In 9th Annual Conference of the Internet Society (INET'99), 1999

[8] G. G. Helmer, J. S. K. Wong, V. Honavar and L. Miller. Intelligent agents for intrusion detection. In IEEE Information Technology Conference, 1998

[9] W. Jansen, P. Mell, T. Karygiannis and D. Marks. Mobile agents in intrusion detection and response. In 12th Annual Canadian Information Technology Security Symposium, 2000

[10] R.Gray, David Kotz, George Cybenko and Daniela Rus, "Security in a multiple-language mobile-agent system," In Giovanni Vigna (Ed.), Lecture Notes in Computer Science: Mobile Agents and Security, 1998

[11] W. R. Cheswick, S. M. Bellovin. "Firewalls and Internet Security: Repelling the Wily Hacker". Addison Wesley Publishing Company (1994)

[12] Computer Operations, Audit, and Security Technology (COAST). "Introduction to Intrusion Detection", www.cs.purdue.edu/coast/intrusion-detection/introduction.html

[13] Computer Security Institute. "The Cost of Computer Crime", www.gocsi.com/losses.htm.

[14] Computer Security Institute. "Intrusion Detection Resources", www.gocsi.com/intrusion.htm.

[15] Wayne Jansen, Peter Mell, "Applying Mobile Agents to Intrusion Detection and Response". NIST Interim Report, 1999

[16] M. Eid. "A new Mobile Agent-based Intrusion Detection System Using Distributed Sensors". In Third FEA Student conference, American University of Beirut, 2005

[17] Omid Mahdi Ebadati E., Harleen Kaur and M. Afshar Alam."A Secure Confidence Routing Mechanism Using Network-based Intrusion Detection Systems". In Second International Conference on Wireless Information Networks & Business information System (WINBIS'10), Nepal, 2010

[18] Christopher Krugel, Thomas Toth. "Applying Mobile Agent Technology to Intrusion Detection". In ICSE Workshop on Software Engineering and Mobility", 2001

[19] Midori Asaka, Shunji Okazawa and Atsushi Taguchi. "A Method of Tracing Intruders by Use of Mobile Agents", Wasedo University, 1999

[20] "Jumping Beans Security," Ad Astra Engineering, www.jumpingbeans.com/Security.htm

[21] Lali Barri`ere, Paola Flocchini, Pierre Fraigniaud, and Nicola Santoro. "Capture of an Intruder by Mobile Agents". In SPAA'02, Winnipeg, Manitoba, Canada, 2002

[22] D. Bienstock, P. Seymour. Monotonicity in graph searching. Journal of Algorithms, 12, 239-245, 1991

[23] J. Ellis, H. Sudborough, J. Turner. The vertex separation and search number of a graph. Information and Computation, 113(1):50-79, 1994

[24] Wayne A. Jansen. "Intrusion Detection with Mobile Agents". Mobile Agents Systems, National Institute of Standards and Technology, 2002, www.nist.gov