Incorporation of Human Resistant System and Advance Network Security System to improve Computer Security

Mr.P.Balamuthukumar

Assistant Professor, Department of Computer Science, NIFT-TEA College of Knitwear Fashion, Tirupur, Tamilnadu, India

Abstract- **Numerous of the present security systems do not provide satisfactory level of protection aligned with ever- increasing threats. The main reason for their collapse is the use of point solutions to protect hosts and reactive approach against intrusions. Here we studied and apply human immune system, which survives under dynamic changing conditions and provides protection against biological viruses and bacteria. By taking immune system as an analogy, we propose an end-to-end network security system using mobile agents with some mandate. Our solution not only overcomes limitations of traditional security solutions, but also enhances overall security by providing protection at each and every level of attack timeline. But the challenge in implementing such system is how to manage mobile agents in such a way that they are always acting as immune agents for our system. It functions in proactive and also reactive manner and has ability to learn and improve its strategies, equivalent to what human immune system does against viruses and bacteria.**

Index terms- **Intrusion Detection, Intrusion Response, Vulnerability Analysis, Mobile Agents, Immune System, Intrusion Deterrence**

1. INTRODUCTION

Information systems are required to survive in different environments in order to continuously provide their services. They must be adaptable to different dynamic environments in order to provide reliable services. We have studied the systems that survive in nature in order to understand how to provide survivability to IT systems. One example of a natural system that survives in different dynamic environments is human immune system [3]. At the lowest level the human body consists of cells. These cells form tissues. The tissues combine to form organs and organs are combined to form complete systems, like immune, digestive, and reproductive system [4]. Cells in the immune system are produced by special areas in the body, like the thymus and the bone marrow [4]. There are different damaging agents (i.e. viruses, bacteria) that can destroy the body. But the immune system is able to identify, locate and remove these damaging agents what allows the body to survive and maintain itself for many years [4]. The immune system enables humans to survive in different environments[1].

Over the past few years research community and commercial product vendors came up with many solutions to protect network from intrusions: antivirus, firewall, spy-ware and authentication mechanism. These solutions still face the challenges of inherent system flaws, OS bugs, and social engineering attacks. Intrusion Detection Systems (IDS) fail to provide adequate protection due to the fact that they cannot detect and respond to all intrusions in real-time, because most of them require customization and human reaction by system administrators. It is very difficult for a system administrator to analyze large logs generated by network traffic, identify the attack, and respond in real-time. Traditional IDSs open a window of opportunity for attacker, because of the delay in attack identification and response by system administrators. The major drawback in all available solutions is their methodology of protection. First, the methodology is reactive: reaction starts when there is already an intrusion in progress. Second, there is no learning mechanism at a network level to study and learn about intrusions and provide protection against the same intrusion to the rest of the network[1]. Third, there are no preventive measures taken against foreseeable threats that can turn into intrusions based on existing vulnerabilities in the system, which become the cause of zero-day attacks.

A number of researchers have applied the immune system features in securing information systems ([1] [4], [5], [6]). The following features of the immune system are applied in information security systems: learning to detect new viruses; detecting viruses locally; identifying viruses; classifying and eliminating viruses autonomously; multiple layered protection system; different cells being able to detect different viruses and few ‘self’ cells being able to detect multiple viruses; and remembering discovered viruses.

There is a need to revisit existing methodologies with an intension to improve them by applying the concept of immune system to achieve comprehensive security for information systems. In this paper we present the system that functions in six stages to secure information systems against intrusions. Our system is based on the concepts of prevention, deterrence, detection, response, and learning as similar as in human immune system. We used mobile agents along with different sensors to achieve the desired results of our comprehensive security system. As the first step our system detects and eliminates vulnerabilities, which are usually being exploited by intruders. Second, our system provides access control mechanism that blocks all illegal accesses based on predefined security policy. However, it is possible that attacks overcome the above two preventive measures. Therefore, as the third step, we have mechanism to detect intrusion attempts in real-time based on Intrusion Detection System (IDS). As the forth step, an Intrusion Response System (IRS) is activated in response to alerts generated by the IDS in order to limit intrusions and damages. The fifth step is use of Post Intrusion Vulnerability Analysis mechanism, which detects the cause of intrusion by identifying the vulnerability that had been exploited by the attacker. Once that specific vulnerability has been identified, the system applies preventive measures against the likely exploitation of the same vulnerability. Finally, all hosts in the network must be kept up-to-date with respect to security technologies, security configurations, and updates installed. The following figure shows our approach using the attack timeline based on the immune system concept.

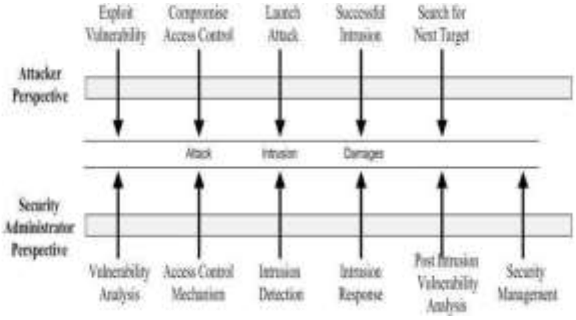


Fig -1: Security Solution along Attacks Timeline [1]

2. RELATED WORK

Misuse Intrusion Architecture: Prevent, Detect, Monitor and Recover Internal Fraud

The article [1][2] describes architecture against misuse intrusion used for preventing, detecting, monitoring and recovering frauds that are caused by employees. The architecture is based on a holistic approach that considers different types of factors, both technical and non-technical. The architecture has managerial, operational, technical and other controls. The measures used for prevention of employee frauds include perimeter defence technology, recruitment screening, segregation, training, and supervision [2]. Measures for detecting frauds include internal auditing, authorization and also technical measures including both network and host based intrusion detection systems. The measures for monitoring are placed at the network, operating system, and application levels [3]. At the network level measures include monitoring access to sensitive assets, monitoring downloads, spamming, and network games. At the operating level monitoring is on system calls, on CPU usage, audit trails, and file access. At the application level monitoring is done on interactions of users with different applications like requests and responses, access patterns, user inputs, and application inputs [3]. Recovery measures include fraud evidence collection and interpretation measures, which include evidence identification and collection, analysis, storage, preservation, transportation and presentations in courts [3].

Integrated Innate and Adaptive Artificial Immune Systems applied to Process Anomaly Detection

In the doctoral thesis [4] artificial immune systems are applied to solve security problems in software systems. The biological immune properties were applied in increasing the performance of artificial immune systems (AIS). The thesis presents nine design principles for the second generation's artificial immune systems. The first principle is that artificial immune systems are represented as autonomous agents. The second principle states problems when AIS are represented as antigens or external (intrusion) signals. The third principle states that the aim of the second generation AIS is to maintain themselves and their environments. The fourth principle defines the functions of agents being to capture antigens, to process, to present, to recognize, to monitor, process and produce signals [4]. The fifth design principle states that agents have a life cycle. The sixth design principle states that agents communicate with the environment at multiple levels. The seventh design principle states that signals can be externally or internally produced. The eight design principle states that receptors can be specific, internal or external signals. Te last principle states that agents can specialize in specific tasks [4].

Architecture for Intrusion Detection using Autonomous Agents (AAFID)
AAFID [18], proposed at Purdue in 1998, is an agent based hierarchal architecture for IDS. It decomposes the traditional IDS into lightweight autonomous cooperating agents, which can easily reconfigure. Autonomous agents used in AAFID project, using static and special purpose agent platform, are used to dynamically reconfigure IDS components. The other thing worth noticing is that AAFID is based on a hierarchal architecture, which is vulnerable to direct attacks. If any of the internal nodes is compromised, the whole branch is disabled. Secondly, the transfer of large logs across the hierarchy also overloads the network traffic.

Automated detection of vulnerabilities in privileged programs by execution monitoring (ADVP)
ADVPP project[1] [9] worked on detection of vulnerability exploitations in privileged programs by monitoring operational audit trails. Their work is based on the assumption that privilege programs are more likely to exploit vulnerabilities. They have introduced the Program Policy Specification Language based on a simple predicate logic and regular expressions.

3. IMMUNE SYSTEM FEATURES FOR INFORMATION SECURITY SYSTEMS

Every information system is recommended to have deterrence measures, prevention measures, detection measures, response measures, and recovery [20].
The immune system has the following features [5] that can be applied to information security systems:

1. Distributed[1] – The T-cells and B-cells can detect infections and viruses locally without doing any global coordination. In this work we model this feature by having mobile agents act as cells in Vulnerability Analysis System (VAS), Intrusion Detection System (IDS), Intrusion Response System (IRS), and Security Management (SMS).
2. Multi-layered[1] – The immune system has multiple defence layers, defence in depth. This is modelled by having multiple protection at the boundary of a system and inside the system and redundancy protection for all sub-systems [5] [16].
3. Diversity[1] – with diversity, vulnerabilities in one system are less likely to be widespread [5] [16]. Diversity could be provided by having agents doing a variety of functions; autonomy – the immune system does not require outside maintenance or management. It autonomously classifies and eliminates foreign cells and it repairs itself by replacing damaged cells. This behavior is the core of mobile agent technology, mobile agents act autonomously based on their task specifications and achieve the desired goals.
4. Adaptability[1] – the immune system is able to detect and to learn to detect new foreign cells and retains the ability to recognize previously detected foreign cells through immune memory [5] [16]. This feature is not new in computer systems, though determining with 100% that a certain program is malicious is a hard problem. We model this feature by enabling mobile agents to learn to adapt to the new intrusions and attacks by using the conceptual architecture of the artificial immune model [16].
5. Dynamically changing coverage[1] – The immune system has a limited amount of cells for detection in any moment. There are about 1016 foreign cells in the environment where humans spend their lives and these foreign cells must be

detect by the limited quantity of detectors of the immune system [5] [16]. The immune system solves this by maintaining a random sample of its detectors that circulates throughout the body. This is modelled in our system by having one agent– based system that detects multiple intrusions, abnormalities and viruses, analyze multiple vulnerabilities, and perform multiple functions in the Vulnerability Analysis System, Intrusion Detection System, Intrusion Response System, and Security Management System.

6. Identification[1] – The immune systems marks all the cells that belong to the body as 'self' [5] [16]. We model this feature in our system by providing 'self' identities to all the objects of a system and by registering them in the database. There are agents that monitor the system and when they find objects that are not having 'self' identities are handled according to the system's policy.

We apply the negative algorithm and cloning algorithms

[5] [16] to generate and train mobile agents for the four subsystems vulnerability analysis system, intrusion detection system, intrusion response system, and the security management sub systems. This genetic information library database, shown in Figure 2, contains genes that have been predetermined based on the a-priori knowledge [17]. These genes are combined to form different solutions like the way you combine Lego blocks to form some solution [17]. The gene libraries provide information for the Enhance-MagicNET agent creator [18]. as shown in Figure 2. Enhance-MagicNET system acts like a bone marrow in the human body in which the B-cells are created [16] [17]. Enhance-MagicNET combines genetic expressions from the database of genetic expressions and artificial immune algorithms from the database of artificial immune system algorithms to generate agents. These software agents are given specialized features for Vulnerability Analysis System(VAS), Intrusion Detection System (IDS), Intrusion Response System (IRS), and Security Management System (SMS), as shown in Figure 3. Agents are tested by using a negative selection algorithm [5] [16] to make sure that they detect only objects that do not belong to the system. Those agents that pass this test are allowed to perform different tasks in the system. The agents that perform better according to specified criteria in the policy are cloned and stored into memory. Future agents are created using the features in memory. In this way the system learns how to adapt to the environments. The next section describes the implementation of the system.

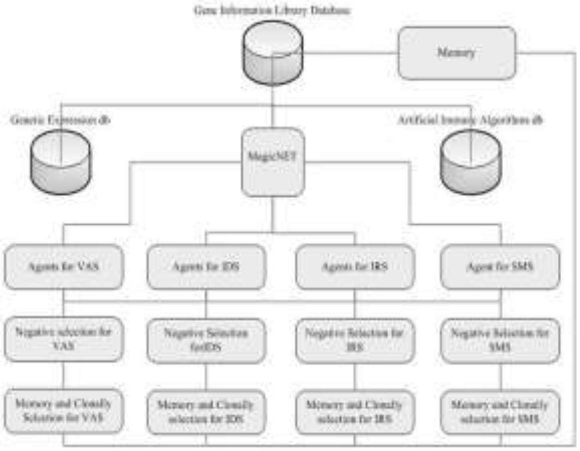

Gene Information Library Database
Memory
Genetic Expression db
MagicNET
Artificial Immune Algorithms db
Agents for VAS
Agents for IDS
Agents for IRS
Agent for SMS
Negative selection for VAS
Negative Selection forIDS
Negative Selection for IRS
Negative Selection for SMS
Memory and Clonally Solution for VAS
Memory and Clonally selection for IDS
Memory and Clonally selection for IRS
Memory and Clonally selection for SMS

Fig -2: Creation of Immune agents for IDP System

4. SYSTEM OVERVIEW

We have developed the complete system in our so called Enhance-MagicNET Environment. Enhance-MagicNET is comprehensive Mobile Agents System that provides features like creation, appraisal, trust evaluation, privileges assignment, and secure adoption, and killing of malicious mobile agents and the secure runtime support for mobile agents' execution [18].

The proposed network security system comprises four subsystems. Figure 3 shows the layered architecture of the system. The four subsystems are equivalent to the human immune system functional concept. These subsystems are as follows:

a. Vulnerability Analysis System (VAS)
- i.Pre-Intrusion Vulnerability Analysis ii.Post-Intrusion Vulnerability Analysis iii. Vulnerabilities Record Management

b. Intrusion Detection System (IDS)
- i.Host based Intrusion Detection ii. Distributed Intrusion Detection

c. Intrusion Response System (IRS)
- i.Host based Intrusion Response ii. Distributed Intrusion Response

d. Security Management (SMS)

- Vulnerability Rule Updater

Moreover, the following three components work with these systems to provide comprehensive network security system.

a. Mobile Agents (MAs)

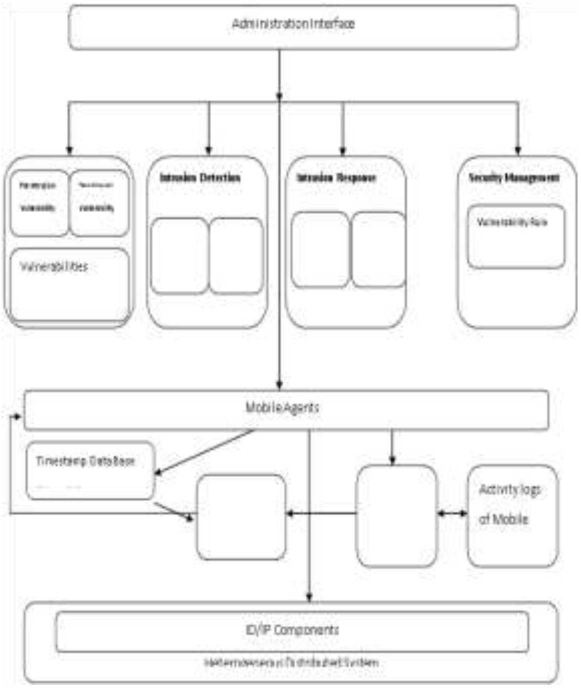
b. ID\IP Components

c. MagicNET Management Console (MMC)

The presented solution is tightly coupled with the number of mobile agents which provide flexibility, scalability, platform independence and reliability to the system. Moreover, mobile agents can execute asynchronously, autonomously and can adapt dynamically at different stages of attacks timeline and thus provide a robust security mechanism. At the abstract level, these mobile agents are acting as middleware between ID/IP components and the overall system's core functionality.

MMC provides graphical interface to Security Administration Station, in short SecAdmin, and to remote servers. At SecAdmin, MMC helps security administrator to execute different tasks and observe activities and results of mobile agent's processing. At remote servers, it provides mechanism to display information about different agents visiting and performing different actions.

Mobile Agents interact with number of ID/IP Components on remote host to accomplish their tasks. These components are in fact static agents that perform certain predefined tasks.



In our implementation we have used five IDP components: SNORT, Osiris, Windows firewall, jRegistry, System logs and Nessus [13][14]. We will not go into the details of these components due to space limitation.

Vulnerability Analysis System (VAS)

As the first step, VAS – Vulnerabilities Record Management provides security administrator with up-to- date and rich information about vulnerabilities. This has been achieved by using three vulnerabilities databases: NVD, OSVDB and Security Focus [10] [11], and generating IDP System's own database called Vulnerabilities DB.

Vulnerabilities DB is a rich database which contains relevant information about most of known vulnerabilities to date. VAS updates DB daily, weekly, or monthly, depending on a local security policy.

Second, Pre-Intrusion Vulnerability Analysis provides capability to query remote servers regarding their profile in terms of different software installed. It is necessary to acquire host profile for effective Vulnerability Analysis.

Two agents perform this activity:

Agent_Vulnerability_Messenger and Agent_ Vulnerability_Analyzer. VAS brings the remote host profile and queries Vulnerabilities DB for vulnerabilities. If they exist, it stores vulnerabilities in the Vulnerabilities DB and displays the result on the administrative interface.

Most of the IDS and Intrusion Response Systems (IRS) consider their task accomplished, after they detect and respond to intrusion. It can be easily inferred that intruder will try to exploit the same vulnerability at a different system in the network. Post-Intrusion Vulnerability Analysis provides capability to analyze the particular vulnerability that was being exploited by intruder in the most recent attack. The main purpose of this analysis is to identify the exploited vulnerability at a particular host and then apply preventive measure to the rest of the network against specific vulnerability. This activity is performed with the help of Agent_Post_Intrusion_ Vulnerability_Analyzer.

As soon as IRS finishes it's task and reports to the Administrative Server (the Server from where agents were launched the first time), the Administrative Server automatically launches Agent_Post_Intrusion_Vulnerability_Analyzer in order to analyze the particular host previously under attack by analyzing the system logs and logs generated by SNORT. Administrative Server stores vulnerability information in the Vulnerabilities DB in the Exploited Vulnerabilities Table.

Intrusion Detection System (IDS)

In an IDP system, a team of mobile agents is dispatched in the network in response to a suspicious alarm. They analyze the statistical data related to different intrusion attempts like DoS attack, DDoS, Doorknob-Rattling Attack [12] etc, and then request for reinforcement from the Security Administrator.

First SIDS, analysis of logs and reporting is performed on logs generated by system events and sensors, like SNORT or Osiris on remote hosts. Analysis of logs notifies Security Administrator about the intrusion.

Agent_IPS_Logs agent along with the Agent_Leader agent analyze these logs, filter them, extract relevant and useful information, and display results to the Security Administrator.

Second, Host–based Intrusion Detection (HID) subsystem monitors the network hosts for different types of intrusions and reports as soon as some intrusion occurs, so that appropriate response measures by SIRS can be applied. Host monitoring is being performed by continuously analyzing SNORT logs. Agent_Host_Monitoring initially is not present at remote hosts. Security administrator launches Agent_Host_Monitoring agent. It along with Agent_Leader reaches remote hosts and stays there permanently. They continuously investigate log entries generated by SNORT, OSIRIS and as soon. If they find any intrusion based on SNORT rules, they ask for help from the administrative server.

Third, Distributed Intrusion Detection (DID) sub-system works equivalent as Host–based Intrusion Detection, but it is activated by the detection of suspicious activity below the intrusion threshold, by ID components. Agent_Host_Monitoring provides the signal for the activation of DID. DID creates Suspicious Host List (SHL) [12], specific to each type of attack, and then logs the addresses of each host generating the same type of alert. Table 1 shows the structure of the SHL.

Table -1: Suspicious Host List (SHL)

HOST IP	Attack Type	Proved	Response Applied
192.168.8.22	Doorknob-Rattling	True	False
192.168.8.17	Distributed Port Scanning	False	False

At this instance of time DID, launches Agent_Distributed_Instrusion_Analyzer (ADIA) along with Agent_Leader and Agent_Distributed_Messenger to visit each host in the SHL, analyze data from system logs, and then correlate and aggregate them with system logs of other entries of SHL to gather traces of the attack. If result is positive, then Agent_Distributed_Messenger immediately returns back to the Administration Server and marks Proved field as true in the SHL for hosts under the distributed intrusion attack, notified by a message to the administrator. Before launching an ADIA, DID creates a static agent i.e. Agent_SHL_Monitoring that constantly monitors SHL for any change in the Proved fields of different entries.

In order to understand the described scenario take an example of a Doorknob-Rattling Attack in which attacker tries few common password and username combinations on many computers that result in failed login attempts [11]. These failed login attempts, as events are logged into the system log. Agent _Host_Monitoring detects these events, marks them as suspicious, and reports to the administrator, where DID creates SHL, fills host IP field and marks attack type as Doorknob-Rattling. DID then launches Agent_Distributed_Intrusion_Analyzer along with Agent_Leader and Agent_Distributed_Messenger to visit each host in the SHL, and analyze data from system logs for Doorknob-Rattling attack. When it is detected that a remote host is under Doorknob-Rattling attack, it reports with the confirmation message to the DID through Agent_Distributed_Messenger and asks for reinforcements.

Intrusion Response System (IRS)

IRS is tightly coupled with the IDS and it is activated as soon as the IDS detects some intrusion at remote hosts for both cases i.e. Host–based Intrusion and Distributed Intrusion. The purpose of the IRS is to prevent the intrusion in real time. It launches

Agent_Intrusion_Response to counter the attack by any mean, like blocking the address from where the attack initiated, closing the port, shutting down service or program, or shutting down of remote host which is under attack in order to stop contamination in the network.

Agent_Intrusion_Response along with Agent_Leader is automatically launched (event based) from the Administrative Server to the host that reported an intrusion. Agent_Intrusion_Response reaches the desired host, creates the response, and then reports to the Administrator. The response is implemented using decision tables. Decision table is a mechanism that associates each attack with a specific response. It is the basis for static mapping and does not consider any other factor, except the attack type.

Agent_Intrusion_Response is using firewall to implement its responses against intrusions.

As soon as Agent_Distributed_Messenger mark, Proved field true Agent_SHL_Monitoring launches Agent_Intrusion_Response along with Agent_Leader. Agent_Intrusion_Response applies appropriate response to all hosts in the SHL that have confirmed intrusion and reports back to the Administrator.

Security Management System (SMS)

SMS also uses mobile agents to perform management tasks in the network. The purpose of these tasks is to keep all hosts in line with respect to number of security services,solutions, or security configurations. SMS performs four tasks.

First, test connectivity of remote host, second query remote host configurations, and third apply a number of security management tasks to remote hosts and exploited vulnerability and SNORT rule mapping. SMS performs these tasks with the help of a number of MAs as follows:

Test Connectivity: Security administrator tests the connectivity of remote hosts by launching Agent_Get_IP along with Agent_Leader. Agent_Get_IP reaches remote host if it is running and connected to network, gets its IP, and reports to the security administrator.

Host Configuration: has acquired with the help of Agent_Configuration_Inquiry that moves to the remote host, checks the configuration of remote host, and returns results to SecAdmin. Configuration of remote host includes OS type, OS version, user logged on, SNORT running status, number of SNORT rules and Osiris database generation date. These configuration items help security administrator to familiarize himself/herself with various security services running on remote host and their status.

Configuration Management: it is performed by Agent_Configuration_ Management (ACM) . ACM at remote host interacts with firewall and SNORT sensors in order to implement security management task. These security management tasks are to enable/disable firewall, enable/disable port, enable/disable services, enable/disable programs running, run SNORT and add SNORT rule.

Agent_Post_Intrusion_Vulnerability_Analy zer updates Vulnerabilities DB against exploited vulnerability, then the SMS launches

Agent_Vulnerability_Responder in the network that updates the rest of remote hosts with a SNORT rule that detect the exploitation of that specific vulnerability which was being exploited earlier.

The mapping between SNORT rules and Vulnerabilities: In ideal situations vulnerabilities should have available patches and security administrator should update network in advance. On the contrarily, usually there is a time gap between discovery of vulnerabilities and availability of their patches. This time gap can become a window of opportunity for an attacker. In order to cater this problem, as soon as intrusion attempt has discovered and responded to first time, a SNORT rule will be created for that specific vulnerability, Agent_Vulnerability_Responder will use it later. Currently in the IDP system the security administrator is performing this task, but in our future research intelligent MAs, based on network condition, would build SNORT rules.

Management of Mobile Agent

In proposed architecture there is possibility in which large numbers of mobile agents are created by our system and then probability of system fail is very high, also it may be possible that some mobile agents get influence by malicious user and worked as malicious mobile agent and if this situation is arise it becomes unmanageable condition for system analyst and administrator. So to avoid this situation some data base for every mobile agent is maintained in our proposed solution.

This metadata and activity log of every mobile agents contain following elements for mobile agent

1. Creation time of mobile elements.
2. Number of assignment performed by mobile agents.
3. Average time taken to perform or detect or intrusion.
4. Reliability factor, we can calculate this factor by considering above three points.
5. Malicious factor for mobile agent, it is basically number of anti-system activities done by mobile agents.
6. We can maintain timestamp for each mobile agent and kill mobile elements whose timestamp is greater than a fixed threshold value.

So by maintaining all above five elements Kill mobile agent sub module works daily, weekly or monthly depending on system definition. By this module we basically destroy all mobile agents who have positive or zero value for (Reliability Factor) - (Malicious factor).

5. CONCLUSIONS

This paper has presented network security system that applies features from the immune system to secure systems. The system has features that help an information security system learn to adapt to dynamic environments. This system uses secure mobile agents to protect systems and networks. These mobile agents are generated and tested using the negative and clonally selection algorithms. The secure mobile agents that pass these tests are allowed to perform the vulnerability analysis, intrusion detection, intrusion response, and security management services. We have implemented the system using secure mobile agents which take input from sensors like, SNORT, Firewall, Nessus, and Osiris. Mobile agents process the inputs from sensors and based on subsystem functionality perform various activities and finally output (protective measure) using sensors again.

Our implementation shows significant improvements in network security and as by deploying system, there is significant reduction in intrusions, as our system catered them using multifaceted approach, similar to the human immune system.

REFERENCES

[1] Muhammad Awais Shibli, Jeffy Mwakalinga and Sead Muftic†MagicNET: The Human Immune System and Network Security System: IJCSNS International Journal of Computer Science and Network Security, VOL.9 No.1, January 2009

[2] N.T. Baloyi. Misuse intrusion architecture: prevent, detect, monitor and recover employee fraud, The Proceedings of the Information Security South Africa 2005 New knowledge today conference. Sandtorn, South Africa.

[3] Furnell, S. Enemies within: the problem of insider attacks, Computer fraud & security. Volume 2004, issue 7, July 2004.

[4] Twycross, J.P. Integrated innate and adaptive artificial immune systems applied to process anomaly detection. University of Nottingham. 2007.www.cs.nott.ac.uk/~jpt/papers/phd-thesis.pdf

[5] A. Somayaji, S. Hofmeyr and S. Forrest. Principles of Computer Immune System, 1997 New Security Paradigms Workshop, ACM p75-82

[6] Symantec. The Digital Immune System, Enterprise- grade Anti-Virus Automation in the 21st century, 2008 www.symantec.com/avcenter /reference/dis.tech.brief. pdf

[7] Tenable Network Security, NESSUS http://www.nessus.org/nessus/ last retrieved 01 December 2008.

[8] Xinyou Zhang and Chengzhong Li, Intrusion Prevention System Design, Proceedings of the The Fourth International Conference on Computer and Information Technology (CIT'04)

[9] Ko, C. Fink, G. Levitt, K. Automated detection of vulnerabilities in privileged programs by execution monitoring, Proceedings of Computer Security Applications Conference, 1994. P 134-144.

[10] National Vulnerability Database, http:// nvd.nist.gov/ last retrieved December 01, 2008.