Mobile Agent Based Adaptive Intrusion Detection and Prevention Systems

[1]Ameya Gangamwar, [2]Anand Kanani, [3]Vivek Singh, [4]Rachana Srivastav and [5]Deven Shah

***Abstract*- The proposed system using mobile agents, provides the computational security by constantly roaming the internal infoways of an organization, presents a solution to detect both external and internal intrusions. Also the Intrusion Prevention Systems (IPS) is being incorporated at the borders of the networks to add to the functionality of the IDS and provide a proactive approach.**

***Index Terms*—Intrusion Detection and Prevention Systems, Mobile Agents, Mobile Agent based IDS**

I. Introduction

INTRUSION detection and Prevention systems (IDPS) have become an essential component of computer security to detect attacks before causing widespread damage. The number of security-breaking attempts originated inside the organizations is increasing steadily. As the idea of filtering the traffic at the "entrance door" (by firewalls, for instance) is not completely successful using the static monitoring agents, the use of other technologies should be considered to increase the defense capacity of a network.

Present-day intrusion detection and prevention systems have the following shortcomings:

Centralization or Partial Distribution [1][3][4]: Many of existing IDS systems perform data collection and analysis using a monolithic architecture. This means that there is a single point of data collection and analysis. If this one weak point is successfully subverted, the attacker obtains considerable power to gain access to the whole network.

Static Reconfiguration [1]: It is difficult to reconfigure or add capabilities to the IDS. Changes and additions are usually done by editing a configuration file, adding an entry to a table or installing a new module and usually needs the system to completely be restarted before the additions take effects.

Vulnerability to Direct Attacks [1]: Because of its critical role, the IDS itself is a primary target for attackers. An attacker can cut off a control branch of the IDS or even decapitate the entire IDS.

Typically, critical components reside on hardened platforms to resist direct attacks. Nevertheless, survivability techniques such as redundancy and dynamic recovery are lacking in current implementations.

Limited Response Capabilities [1]: Security systems are traditionally focused on detecting attacks. An automatic response reduces the time window an attacker has before being countered by a human to limit further system damage. Automatic responses include the IDS ability to trace an attacker through the attacked network, to respond at the target, respond at the source, to collect evidence about the attack from the host and network components, and to isolate the source and/or the target.

Limited Flexibility, Adaptability, and Extensibility [2]: Typically, IDSs have been designed for a specific environment and have proved difficult to use in other environments. Adaptability is the ability of the system to detect slightly different patterns. Traditional IDSs cannot adapt to different patterns of usage. Moreover, adding modules to existing running IDSs needs tough work such as reconfiguring system files, often the IDS needs to be completely restarted in order to make changes and additions take place.

High False Positive/Negative Rates: False alarms are high and detection recognition is not perfect. Lowering the detection thresholds to reduce the false positive raises the false negative rates and thus leaving some attacks undetected. Also high detection thresholds cause overwhelming the detection engine with huge false negatives alert files.

The approach suggested by us provides a hybrid IDPS which uses mobile processing units to capture and analyze relevant data asynchronously and independently from the main machine. Roaming the internal network, the agents are capable of detecting both internal and external attacks. The proposed approach is highly secure against attacks targeting the IDS itself since it keeps on working even if a part of it is compromised due to its distributed capabilities.

II. Architecture

The Enterprise Network will be consisting of many hosts, workstations, general-purpose servers, critical servers running different operating systems and other networking devices like switches, routers, firewalls, etc (henceforth will be referred as elements).

Ameya Gangamwar is a student of final year I T Engg., S.P. College of Engg., Mumbai, India. (Email: ameya.gangamwar@gmail.com).

Anand Kanani is a student of final year I T Engg., S.P. College of Engg., Mumbai, India. (Email: kanani_anand@yahoo.com).

Vivek Singh is a student of final year I T Engg., S.P. College of Engg., Mumbai, India. (Email: vivek.singh@yahoo.com).

Rachana Srivastav is a student of final year I T Engg., S.P. College of Engg., Mumbai, India. (Email: rachana.srivastav@gmail.com).

Deven Shah is a faculty in Information Technology Dept. of S.P. Institute of Technology, Mumbai, India (email: devenshahin@yahoo.com).

The objective is to provide a framework for centralizing, organizing, and improving detection and display for monitoring security events within the enterprise. As discussed in the abstract, the outcome will be an integration of various open-source security and auditing tools not only working as NIDS but also HIDS working as signature-based as well as heuristic anomaly detection and try to figure out most difficult-to-know zero-day events. Now the network plan is studied and the network elements are categorized into one of the security category by the administrator.

The framework consists of 5 main components :

1. Central Server (CS)
2. Mobile Agent Platform (MAP)
3. Mobile Agent (MA)
4. Software Repository (SR)
5. Zone Listeners (ZL)

A. Central Server:

The central server, as shown in Fig.1, has the following functions:

1) Create/Destroy mobile agents and communicate with them

2) Normalize heterogeneous data received from various sources into a proper format and store it in the Event DB

3) Update the Signature DB with the latest signatures of attacks

4) Perform network-wide correlation and anomaly detection using signature & event DB and the algorithm used for this purpose will be similar to that of OSSIM (Open Source Security Information Management)

5) Provide a User Interface which acts as a control panel for the admin for management of entire system

6) Periodically reconfigure the external or internal (at the gateways) Intrusion Prevention System (IPS) if required

The Central Server is placed in a secure location so that it cannot be traced.

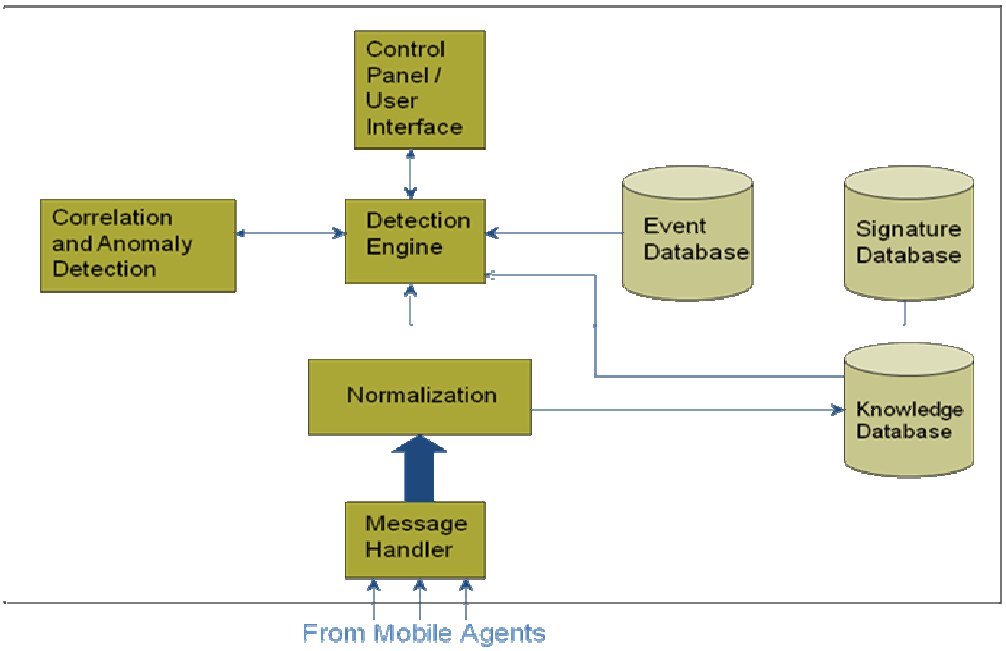

Fig.1. Block diagram of Central Server

B. Mobile Agent Platform

The platform is responsible for receiving the mobile agents, generating child mobile agents, and dispatching agents back into the network. The platform is a small server program that will reside in each required host within the network and will be responsible for managing the mobile agent life cycle.

C. Mobile Agent

The Mobile Agents are the agents which roam inside the network and do the work of detection in a n almost autonomous manner. The block diagram of mobile agent is shown in Fig.2. The Mobile Agents perform the following activities:

1) Agent arrives at the host and checks the security category of the host

2) Depending upon the security category of the host, the MA downloads required tools from the Repository Server and installs them on the host when it reaches the host for the first time

3) On the further visits, it gathers logs and alerts and determines the current security level at the level at the host. The actions taken depending on the security level are mentioned in Table 1

TABLE I
Actions taken on current security level

Level	Action
1	Send entire log + alerts
2	Send summary of log + alerts
3	Send alerts only
4	Don't send any summary or alerts

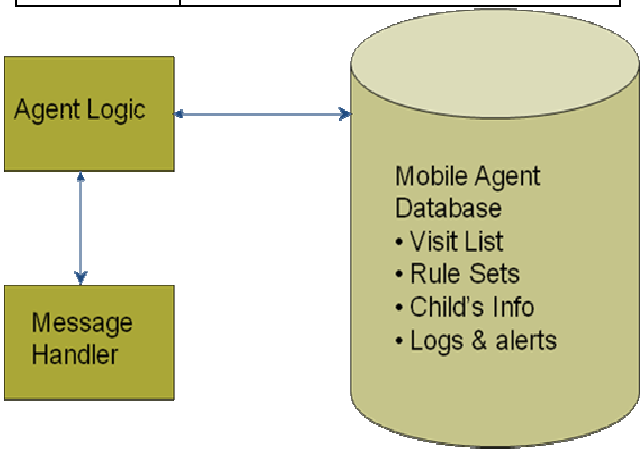

Fig.2. Mobile Agent Block Diagram

D. Software Repository

This is a NFS or FTP server which stores the setups of the tools which are installed on the network elements by the mobile agents.

E. Zone Listener

As mentioned earlier, since the network is divided into sub-networks, each sub-network's traffic will be routed through a ZONE-LISTENER. A zone listener will have tools like snort and nessus, which will perform the function of Network Intrusion Detection System.

III. Working

This system has a main server which is responsible for creating the agents, updating their rule sets and deploying them.

A. State Diagram of the main mobile agent in the System:

The main mobile agents are deployed by the central agent in all the sub networks/zones. The state diagram as shown in Fig.3 gives the actual working of the MAIN mobile agent. The working is summarized below:

1.When the system is functional, the Main mobile agent (MMA) is deployed by the Central Server (CS) into the network

2.The MMA reaches the first terminal and then checks the statistic which indicates the number of its visits to that terminal.

CASE 1:

3.If it finds out that it is its first visit to that node then it goes a step further and then checks the category of that terminal.

4.After getting the information about the category, the MMA gives the link to the FTP server to the terminal and then a predefined set of tools for that category is downloaded and installed.

5.The MMA then moves to the next terminal.

CASE 2:

6.If it finds out that it is not its first visit to that node then it gathers the logs generated by the system and calculates the security level.

7.It sends that level to the CS and waits for an acknowledgement.

8.After getting an acknowledgement ACK, the MMA creates a child mobile agent (CMA).

9.Now after the CMA is sent to the CS, the MMA waits for a response signal from the CS indicating that the CMA has reached the CS safely.

10. After getting this response, the MMA moves on to the next terminal.

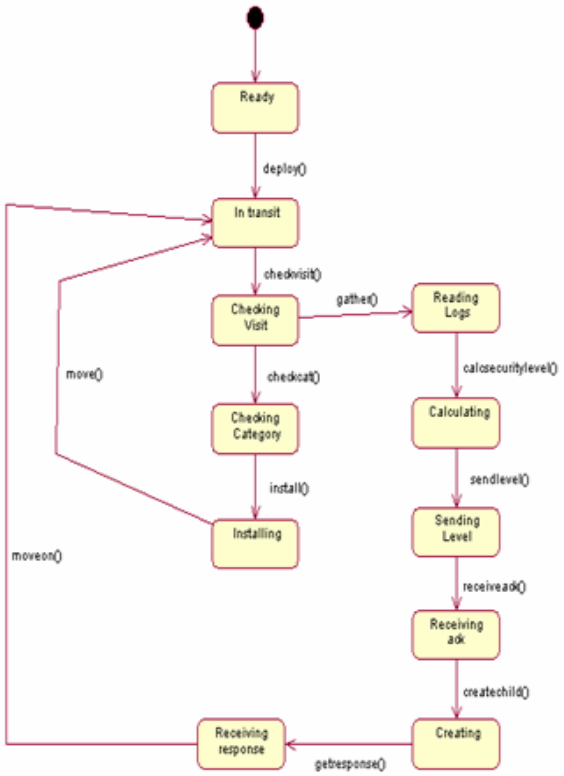

Fig.3.Block Diagram of States of Main Mobile Agent

B. State Diagram of the Child mobile agent in the System:

The child mobile agents are created by main mobile agents in order to send data to central server. The state diagram as shown in Fig.4 gives the actual working of the CHILD mobile

agent. The working is summarized below:

1. After the Child mobile agent (CMA) is created by the Main mobile agent, the CMA collects the data that is to be sent to the Central Server.
2. The CMA is then sent to the CS.
3. After passing the data to the CS, the CMA sends a signal to the MMA indicating that the data transfer is over.
4. Then the CMA dies.

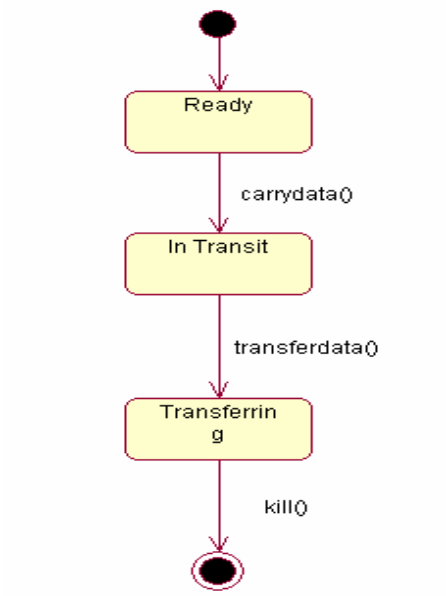

Fig.4.Block Diagram of States of Child Mobile Agent

The main advantage of this system is that it is neither completely centralized nor completely distributed. The mobile agents communicate with the main server at every step by sending messages which keeps the main server aware of their presence. Similarly, because of the same working mechanism the mobile agents too are aware of the main server's presence.

Now, if the main mobile agent is killed by an intruder then it would not be able to send any messages to the main server. If the main server does not receive any missive from the main mobile agent then, after a particular period of time, it assumes that the main mobile agent for that particular zone is dead.

So, the main server would again create and deploy another main mobile agent with an updated rule set for that zone.

Consider another case where the main server is compromised. Here, the mobile agent may not receive any messages from the main server. In such a case, the mobile agent will assume that the main server has been compromised. So, it will generate only the alert and not create any child agent to send the data to the main server. It will then move on to the next node in the zone and continues with its operations. After sometime, if it realizes that the server is up again; it starts its normal operation of sending updates & alerts to the server.

C. Normalization

Normalization is aimed at unifying security events from all critical systems throughout the organization in a single format on just one console. Normalization requires a parser or translator familiar with the types and formats of alerts coming from different detectors. We have organized the database and adapt the forensic console in order to homogenize processing and display of all these events. That way we will be able to observe all security events for a particular moment in time-whether they come from a router, a firewall, an IDS, or a UNIX server - on the same screen and in the same format. When we have all network events centralized in the same database, we achieve a considerably comprehensive view of what's going on throughout the network, which, as we will see shortly, allows us to develop processes that enable us to detect more complex and widely dispersed patterns.

D. Correlation & Anomaly Detection

The correlation function can be defined as an algorithm that executes an operation on input data and returns output data. Here the information collected by our detectors and monitors to be specific is yet partial; it illuminates only small areas along the spectrum. Correlation gives the ability to take advantage of these systems and using a new layer of processing, fill in more areas along that infinite spectrum of all possible information about a network, thus giving a holistic view of the security status of the network.

IV. Conclusion

The system potentially reduces the massive amount of distributed log data moved among the inner nodes of conventional IDS. Having mobile IDS agents visit hosts and doing intrusion detection locally is well suited to the ability of mobile agents to move the computation to the data, thus reducing network load. Roaming the internal network, agents are capable of detecting attacks launched from within the network since the IDS will be capable of monitoring local traffic. Also due to control over external and internal IPS, intrusions can be prevented dynamically.

There is no single vulnerable point of failure. Agents roam the network continuously and thus are less suspicious to direct attacks. They operate independently and autonomously from where created.

The architecture is flexible since it is built on the concept of "Severity on Demand". This means that the system administrator can specify the severity levels of attacks, the attacks to look for, and the response mechanism depending on his/her own network needs.

V. References

[1] Mohamad Eid "Adaptive and Defense Mobile Agent Based Intrusion Detection System", March 2005.
[2] Eleazar Eskin, Matthew Miller, Zhi-Da Zhong, George Yi, Wei-Ang Lee, Salvatore Stolfo "Adaptive Model Generation for Intrusion Detection Systems", September 2002.
[3] Christopher Kruegel and Thomas Toth "Flexible, Mobile Agent based Intrusion Detection for Dynamic Networks", TUV-1841-2002-27, April 2002
[4] Christopher Kruegel, Thomas Toth and Engin Kirda "Sparta - A Mobile Agent based Intrusion Detection System", TUV-1841-2002-24, April 2002

VI. BIOGRAPHIES

Ameya Gangamwar a final year student of IT Engg., S.P. College of Engg., Mumbai, India. His areas of interests are Software Engg and Project Management.

(Email: ameya.gangamwar@gmail.com)

Anand Kanani is a final year student of IT Engg., S.P. College of Engg., Mumbai, India. His areas of interests are Networking & Digital Security.

(Email: kanani_anand@yahoo.com)

Vivek Singh is a final year student of IT Engg., S.P. College of Engg., Mumbai, India. His areas of interests are Finance and Management.

(Email: vivek.singh@yahoo.com)

Rachana Srivastav is a final year student of IT Engg., S.P. College of Engg., Mumbai, India.

(Email: rachana.srivastav@gmail.com)

Deven Shah is professor in IT Dept; S.P. Institute of Technology, Mumbai. He is currently pursuing PhD from NIT, Surat.

(E-mail: devenshahin@yahoo.com).