Architecture for Intrusion Detection System with Fault Tolerance Using Mobile Agent

Chintan Bhatt[1] , Asha Koshti[2] ,Hemant Agrawal[3] , Zakiya Malek[4],
Dr Bhushan Trivedi[5]

MCA Dept.,GLS Institute of Computer Technology, Ahmedabad, India

chintanbh@gmail.com[1] , aasha.kosti@gmail.com[2],
hemant_agrawal25@yahoo.co.in [3], bhtrivedi@gmail.com[4],
zakiya.malek@glsict.org [5]

Abstract

This paper is a survey of the work, done for making an IDS fault tolerant.Architecture of IDS that uses mobile Agent provides higher scalability. Mobile Agent uses Platform for detecting Intrusions using filter Agent, co-relater agent, Interpreter agent and rule database. When server (IDS Monitor) goes down, other hosts based on priority takes Ownership. This architecture uses decentralized collection and analysis for identifying Intrusion. Rule sets are fed based on user-behaviour or application-behaviour.This paper suggests that intrusion detection system (IDS) must be fault tolerant; otherwise, the intruder may first subvert the IDS then attack the target system at will.

Keywords

Fault tolerance, Mobile Agent, Intrusion Detection System

1. INTRODUCTION

Fault tolerance is a means of achieving dependability, working under the assumption that a system contains faults, and aiming at providing the specified services in spite of their presence. Implementing an effective intrusion detection capability is an elusive goal, not solved easily or with a single mechanism so there are many distributed intrusion detection system architecture proposed using autonomous or mobile or multi agent [7]. These IDS took the advantage of distributed system using Mobile agent. But they have some disadvantages related to performance of System executing IDS and its mobile agent platform on various hosts, security concern of agents, agent's lack of a prior knowledge on heterogeneous environment.

In most commonly found attack scenarios, an intruder first gains access to a single host by exploiting the flaws in existing software applications, computer viruses, or through some misconfigured applications. Using that compromised host, attempts are made to gain access to other hosts in the network [10].therefore an IDS is needed to detect and respond whenever anything unusual is happening with the computer resources. Intrusion detection systems (IDSs) were conceived of as a form of expert system that observes patterns of activity in user accounts and notifies a system administrator if anything unusual is detected. As an IDS provides the security mechanism but what happens when IDS itself get compromised? Or what difficulties can arise when server goes down? So a fault tolerance mechanism is needed to implement the IDS.

DOI : 10.5121/ijnsa.2011.3513

Here the paper describes an architecture that is more concerned about the fault tolerance of the IDS. In that system Administrator once specifies the back-up hosts if Server goes down. Some of hosts on network virtually combined and also make one logical IDS Using Mobile Agent if there is corruption or any disruption of service by actual IDS monitor. If IDS may be attacked first, after it has been subverted, the system is left defenseless. Hence, it is important to make an IDS fault tolerant.

Here in this paper sections are: (I) Literature Review (II) System Architecture (III) Fault Tolerance mechanism with the architecture (IV) Implementation Direction (V) Discussion and results

This paper discusses the fault tolerance with the effectiveness of Mobile Agent Platform.

2. LITERATURE REVIEW

We have reviewed various Intrusion detection systems using with or without different types of agent. They represented frameworks, architectures and their advantages & disadvantages. Some also represent other implementation of IDS.

The initial architecture had the hierarchical structure that remains to date, included monitors, transceivers and agents, and was used to implement the first prototype of the system [4]. Various network computing paradigms that support communication between entities in a distributed computer system and Different messaging scheme is defined in [1] they talk about centralized data collection and analysis units. Communication among mobile agents and Agent Migration are in [2].

There are four broad categories of security threats related to the use of mobile agents & countermeasures are discussed in [3, 4, 5, and 6]. Within IDS design improvements, there are three categories of research: new detection paradigms, new architecture paradigms, and improvements over existing designs [11].Because Intrusion detection is proven technology and the Intrusion Detection System architectures commonly used in commercial and research systems have a number of problems that limit their scalability or efficiency.therefore commercial companies are mostly perfecting existing intrusion detection techniques.Please see references for further reading.

3. SYSTEM ARCHITECTURE

The presented intrusion detection system Architecture is designed by keeping in mind the notion of flexibility, scalability, platform independence, reliability and most important one is the fault tolerance.

It contains A. Mobile Agent Platform (MAP) B. Mobile Agent (MA)

3.1 Mobile Agent Platform (MAP)

It provides execution environment to MA. MAP is like virtual machine on different operating system. It performs security checks for Mobile agents.

MAP has security measurement module for security related things. It is done when some other MA come to MAP. It applies Authorization and authentication mechanism when MA comes.

MAP contains platform for running filter agent, co-relator agent. It also contains detection engine. Detection engine in turn contains memory segment for storing log or some data for detection purpose and module for Interpreter Agent.

Filter Agent → It Filter the packet and give appropriate packet or data to Agent. Filter Agent is agent responsible for filtering specialized security events from the log files. It examines the packets for well-known attack events and stores all its characteristics into Log files. Log files contain events. An event is an indication of intrusion. A security event is characterized by its signature, its type, location, and a temporal attribute representing the event occurring moment.

Co-realtor Agent → it does tasks of collecting log for detection by co-relating other-to-agent's data running on different platform. Using Co-realtor agent, Server or any host can also detect malfunctioning or compromised host by collecting data from different hosts.

Detection Engine → Detection engine check for suspicious behaviour by using rule or statistical data. It main task is of detecting threats and unusual things. The heart of our detection mechanism is the Interpreter Agent. It analyzes events for detecting complex local attacks, and uses the Correlate agent with the Mobile Agent for determining whether some suspicious activities in different node can be combined to be a distributed intrusion.

Rule Files in Detection Engine is set of rule files or statistical data related to user-behaviour or application-behaviour on System. Rule Files are plug-able. New files can be added or updated. Interpreter Agent can also generate files from learning System Environment.

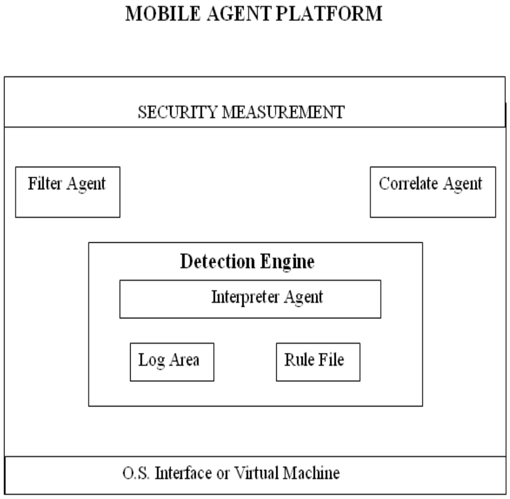

Figure 1: Mobile Agent Platform

3.2 Mobile Agent (MA)

A system can be distributed over any number of hosts in a network. Each host can contain any number of agents that monitor for interesting events occurring in the host. It contains manipulation flags and security related data. Manipulation flags are one kind of access list using that MA can use or create separate thread for its own use from MAP's filter agent module and co-relator agent module. Some MA can not create thread of co-realtor agent module or some only of co-realtor agent.

MA's main task is to initiate actions from MAP to detect Intrusion. It continuously reads communication channels and system logs. It uses various agent and can invoke it as its own thread of correlating, filtering and interpreting (i.e. machine learning) the things coming to its system.

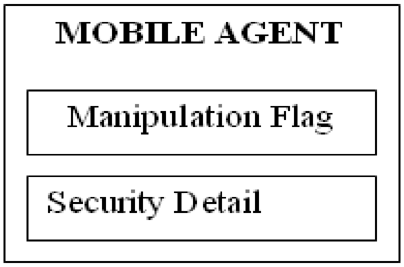

Figure 2: Mobile Agent

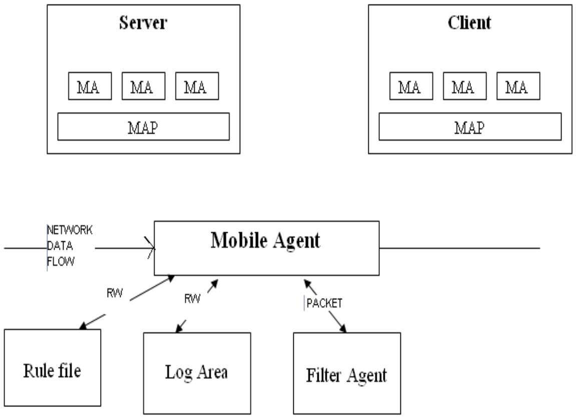

Figure 3: Client –Server Using Mobile Agent

4. WORKING OF ARCHITECTURE WITH FAULT TOLERANCE

When server and host boots up, both should starts MAP. Some MAP can themselves create MA, but for some host server sends mobile agent.

Without MA, MAP can't monitor traffic and any activity or event. Only Detection Engine starts and tries to identify pattern based on log and rules file which it has.

MA does specialized monitoring for specific rule or protocol. It invokes Filter Agent for filtering traffic and data. MS stores information in log area to read other MA on same MAP. It uses rule file as well as log file. It may also indirectly use rule files interacting with detection engine.

Mobile agent platform resides on server as well as client machines but MAP on client machine has less rights that of server machine. If server goes down than the few difficulties arise with it:1) MAP on client will not be able to communicate therefore new intrusion can not be find out 2)database will also not be updated with the new rules3) If client host does not have MA then also server can not send it. 4) Client MAP can compromise

Here lots of difficulties have been find out so for removing all such difficulties a new technique is proposed: In this technique, first we will assign the priority to the every MAP client on the sequential basis. So if server goes down then the collectively the group of MAP client can take the in charge of the server's activities and serve as a server itself. So the priority of the MAP client will decide this.

Each MAP client will have the priority value associated with it. It resides in the Security measurement field in the MAP. But How MAP client will be able to know that it has the maximum priority? For that every MAP client will check its own priority and wait for the maximum priority host group to become the server.

Another problem is how the client MAP will be able to know that server is down? For that the technique is: when the client MAP will not receive any response after sending the request three or more times to the server the client MAP will know that server is down. Now the client's responsibly is to broadcast this message to every other client in the network that runs MAP.

For broadcasting the message to every MAP client, you should make the MAP client that communicates with other clients in the network and the maximum priority group will act as a server but what if the maximum priority host group is down? In that case second maximum priority host group become server and so on. But how the second priority group will know that the first priority groups MAP client are down? For that we will use the TTL (time to live) for that much of time a group should wait for becoming a server and whoever become a server should broadcast that "mine group is server". Now the scenario is when server comes back .the server should broadcast the message and take the charge from the backup server.

5. IMPLEMENTATION DIRECTIONS OF IDS

More alternatives lead to more confusion. In an environment where technology poses both opportunity and risk, it's essential that you make your design and purchase decision wisely. The difference between selecting a technology that meets the long-term needs of an enterprise and making a choice that the enterprise many later come to regret is not a matter of happenstance. Such critical decision requires an understanding of the needs of the business, and these needs must be represented in terms of a balanced, objective set of criteria and methods.

The basic idea is, when snort detect an intrusion, an aglet will read the log file created by snort containing the alert message of the intrusion, then the aglet will inform tahiti server about this, tahiti server will then dispatch a new agent to the targeted PC and close the port of the targeted PC in order to prevent the intrusion from occurring.

5.1 Detection Engine

We used snort as Detection Engine because snort is freely available and light-weight. It is and open source, packet sniffer / packet logger / network IDS. Snort's beefiness comes from its intrusion detection capabilities which match packet contents to an intrusion rule. Snort is a signature-based IDS and uses rules to check for errant packets in your network. It also displays all the different network packets.

5.2 Mobile Agent Platform

Tahiti server will work as a MAP. This module is responsible for allowing the network administrator to monitor and control the mobile agent mobility. It will provide a framework for execution to the mobile agent. It will help to instantiate the mobile agent and move from one machine to another machine in the network. It will also help to log the information about mobile agent arrived and created. It will display the display the summary of the mobile agent instantiated or arrived in the system. It will also allow the administrator to control the life of the mobile agent by activating, deactivating, cloning, disposing it.

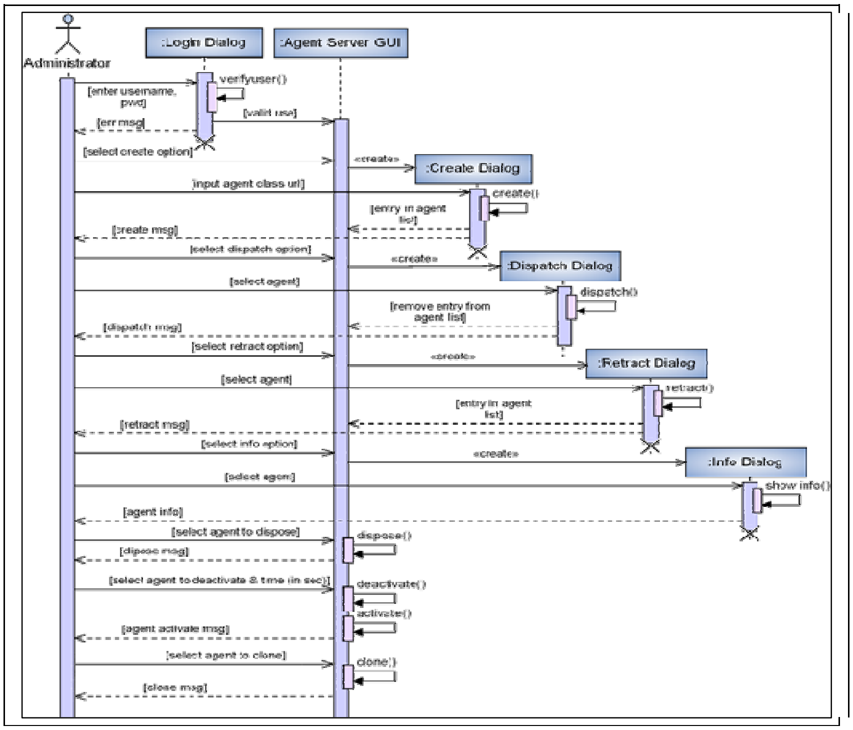

Figure 4: Sequence Diagram for MAP

5.3 Mobile Agent

Mobile Agent based system is implemented using Aglets. Java based mobile agent toolkit developed by IBM Now available as open source under the IBM public license The name originates from AGENT + APPLET = AGLET.

Security Details: For authenticating user and the owner of the mobile agent it will use digital signature. The users have to create his private key using the key-tool utility and login by providing his key-alias as login name and key password as his password.

5.4 Data Storage: XML

Only the information about the IDS node and snort rule is required to be store, so no big need for the database. The issues of licensing create a problem if database server is used. So the best alternative is XML. It is platform independent and easy and efficient XML data manipulation APIs are available.

5.5 Filter Agent

Filter Agent based system is implemented using Aglets.This module will allow the user(s) of the MA-DIDS to manage the different IDS Node(s) across all the networks to be monitored. The user(s) can send specific request to static agents on IDS node(s) to filter particular packets from the logs generated by IDS. This module will allow the user(s) to view the gathered logs as per requirements based on different criteria like protocol, source/destination IP address or ports, contents of packets etc. This module will allow the user(s) to correlate the logs gathered from different networks to get the details of attacks that are carried out in all the networks.

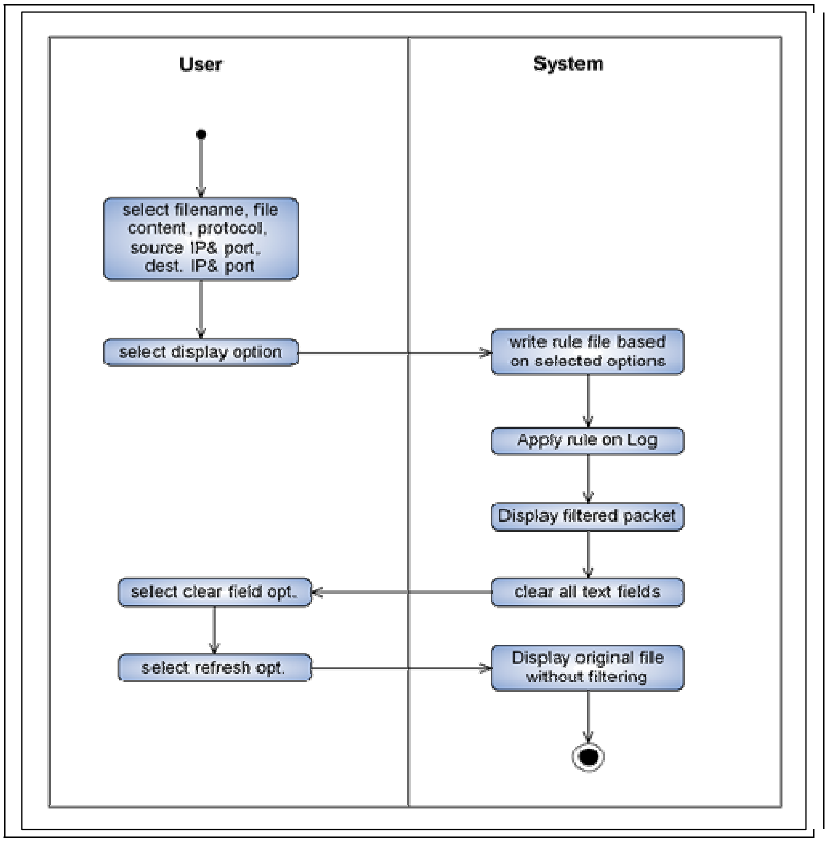
User
System
select filename, file content, protocoll, source IP& port, dest. IP& port
select display option
write rule file based on selected options
Apply rule on Log
Display filtered packet
clear all text fields
select clear field opt.
select refresh opt.
Display original file without filtering

Figure 5: Activity Diagram for Filter Agent

6. DISCUSSION AND RESULTS

The system is configured as follows:

MAP server in the network will have the snort with winpcap for the sniffing of packets. It also has the Tahiti server which will act as MAP and aglet API. It needs J2SDK installation for the java API and key-tool utility. For the authentication purpose you need to first generate your username and password with help of java key-tool utility.

Whenever the tahiti server boots up, the administrator can send the agent on the network at any host with the of IP address of the host and ATP (aglet transfer protocol). Then we should start explicitly MAP for every host if you want to run Mobile agent on that host therefore mobile agent will be created on the server and MAP client will dispatch that mobile agent. It also takes rules file of the snort and stores all data in XML file. We have provided all option like dispatch, filter, start etc. for the mobile agent.

It also generate different report like log file, detected packet detail, host detail, rules file details. It gives alert to the network administrator when an attack is detected. It gets run on wireless or wired network. All the agents are autonomous and are not affected by any activity of another agents executing in the network.

7. CONCLUSION AND FUTURE WORKING

In this paper, we have proposed scalable distributed IDS which cover the flaws of the other models while using their useful features and fault tolerance. We have also tried to implement the IDS therefore implementation guidance is also given in the paper. This IDS uses specialized mobile agent which reduces monitoring tasks. We now want to stronger detection engine or IDS detection engine can be integrated with existing anti-virus system and use their rules or database to monitor host. Common communication format for exchanging rules or log between agents to remote agent are also in consideration.

REFERENCES

[1] Lange, D., Oshima, M. 1998. *Programming and Deploying Java Mobile Agents with Aglets* . Addison-Wesley.

[2] Rothermel, K., Schwehm, M. 1998. *Mobile Agents* . In Kent, A., Williams, J. (Editors) *Encyclopedia for Computer Science and Technology* . M. Dekker Inc. New York, USA

[3] Jansen, W. 1999. Mobile agents and security. In *Proceedings of the 1999 Canadian Information Technology Security Symposium* .

[4] Jansen, W. 2002. Intrusion detection with mobile agents. *Computer Communications* , 25(15): 1392-1401.

[5] Jansen, W., Karygiannis, T. 1999. Mobile agent security. Special Publication 800 19, National Institute of Standards and Technology (NIST).

[6] Jansen, W., Mell, P., Karygiannis, T., Marks, D. 1999. Applying mobile agents to intrusion detection and response. Interim Report 6416, National Institute of Standards and Technology (NIST)

[7] Intelligent Agents for Distributed Intrusion Detection System M. Benattou, and K. Tamine World Academy of Science, Engineering and Technology 6 2005

[8] Intrusion Detection & Prevention by Carl Endorf, Eugene Schultz and Jim Mellander ISBN:0072229543 TMH pub.

[9] INTELLIGENT INTRUSION DETECTION SYSTEM FRAMEWORK USING MOBILE AGENTS International Journal of Network Security & Its Applications (IJNSA), Vol 1, No 2, July 2009

[10] DIDMA: A Distributed Intrusion Detection System Using Mobile Agents Pradeep Kannadiga and Mohammad Zulkernine School of Computing Queen's University Proceedings of the Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and First ACIS International Workshop on Self-Assembling Wireless Networks (SNPD/SAWN'05) 0-7695-2294-7/05 © 2005 IEEE

[11] Applying Mobile Agents to Intrusion Detection and Response Wayne Jansen, Peter Mell, Tom Karygiannis, Don Marks National Institute of Standards and Technology Computer Security Division NIST Interim Report (IR) – 6416 October 1999

[12] An Architecture for Intrusion Detection using Autonomous Agents Jai Sundar Balasubramaniyan, Jose Omar Garcia-Fernandez,DavidIsaco, Eugene Spafford, Diego Zamboniy Center for Education and Research in Information Assurance and Security Purdue University CERIAS Technical Report 98/05 June 11, 1998

[13] A New Mobile Agent-Based Intrusion Detection System Using Distributed Sensors Mohamad Eid American University of Beirut, Department of Electrical and Computer Engineering

[14] Snort website: www.snort.org

Authors

Chintan Bhatt
MCA, SEM-5 Student GLSICT, Ahmedabad , India
Research Area :- Network Security

Asha Koshti
MCA, SEM-5 Student GLSICT, Ahmedabad , India
Research Area :- Network Security

Hemant Agrwal
MCA, SEM-5 Student GLSICT, Ahmedabad , India
Research Area :- Network Security

Zakiya Malek
Assistant Professor, GLSICT, Ahmedabad , India
Research Area :- Network Security

Dr. Bhushan Trivedi
Ditrector, GLSICT, Ahmedabad , India
Research Area :- Network Security