

Analysis of Distributed Intrusion Detection Systems using Mobile Agents

Nita Patil

Datta Meghe College of Engineering,
Airoli , Navi Mumbai-400708
nspatil_10@rediffmail.com

Shreya Patankar

Datta Meghe College of Engineering,
Airoli , Navi Mumbai-400708
shreyanpatankar@yahoo.co.in

Chhaya Das

Datta Meghe College of Engineering,
Airoli , Navi Mumbai-400708
chhaya_mdas@yahoo.com

Kshitija Pol

Datta Meghe College of Engineering,
Airoli , Navi Mumbai-400708
kshitija_pol@yahoo.com

Abstract

The goal of IDS is to analyze events on the network and identify attacks. The increasing number of network security related incidents makes it necessary for organizations to actively protect their sensitive data with the installation of intrusion detection systems (IDS). Detecting intrusion in distributed network from outside network segment as well as from inside is a difficult problem. Intrusion detection system must analyze a large volume of data while not placing a significant added load on the monitoring systems and networks. The deficiency of centralized intrusion detection systems leads the idea of mobile agents. In an agent based IDS idea, there is no central station, therefore no central point of failure. Agents can detect and take predefined actions against malicious activity. The system shows a superior performance compared to central sniffing IDS techniques, and saves network resources compared to other distributed IDSs that activate too many sniffers causing bottlenecks in the network. This is one of the major motivations to use the distributed model based on Mobile Agent platform. This paper presents survey of Distributed Intrusion Detection System based on Mobile agents. It also includes an overview of several agent based intrusion detection implementation.

Key Words- Mobile agent, IDS.

1. Introduction.

The growing importance of network security is shifting security concerns towards the network itself rather than being host based. Security services must be

evolving into network-based and distributed approaches to deal with heterogeneous open platform and support scalable solution [1]. The intrusion detection technology is the process of identifying network activity that can lead to a compromise of security policy. IDS must analyze and correlate a large volume of data collected from different critical network access points. This task requires an IDS to be able to characterize distributed patterns and to detect situations where a sequence of intrusion events occurs in multiple hosts.

2. Basis of IDS.

The two major variants of intrusion detection systems (IDS) have emerged, namely host and network based approaches. Host based systems collect local data from sources internal to a computer, usually at the OS level. This has the advantage of collecting high quality data directly at the source (e.g. kernel). Unfortunately, some attacks cannot be detected at a single location.

Distributed intrusions may leave innocent marks at each single host and can only be identified when combining data from a number of different machines.

Network based variants monitor packets on the wire by setting the network interface to promiscuous mode and analyzing network traffic. Therefore they have some possibilities to correlate activities that occur at different hosts, but suffer from scalability problems in case of high network load and have problems when encrypted communication is used.

A new approach is the development of distributed architectures, where sensors (host and network based) collect data, preprocess it and send it to a centralized analyzing station which is able to relate this input.

Intrusion Detection has been achieved by following two different strategies of analysis [8] .

- Anomaly detection: relies on models of "normal" behaviors of a computer system. Behavior profiles may be focused on users, applications or networks. Anomaly detection compares the defined profiles against the actual usage patterns to detect "abnormal" activity patterns. These patterns will be considered as intrusions.

- Policy detection: relies on a set of attack descriptions called attack signatures.

2.1 IDS requirements.

In [7], authors have defined a set of desirable characteristics for IDS by focusing two themes: functional and performance requirements.

In the following section, we summarize some of these characteristics.

2.1.1 Functional requirements

- IDS must continuously monitor and report intrusion,
- IDS should have a very low false alarm rate,
- IDS should provide enough information to repair the system in the case of detection of intrusion. Notice that this characteristic depend on IDS goals. In fact, many IDS solutions focus only on alerting administrators without suggesting any corrective actions.
- IDS must detect and react to distributed and coordinated attacks. This detection feature is one of the most difficult because it needs a huge distributed amount of information in addition to the hard task of synchronization between different hosts.
- The IDS should be adaptive to network topology and configuration changes.

2.1.2 Performance requirements.

- Intrusion should be detected in real-time as it should be reported immediately in order to minimize network damage,
- The IDS must be scalable in order to handle additional computational and communication loads.

2.2 Limitations.

The most common IDS shortcomings include the following:

- High number of false positives
- Lack of efficiency: usually when an IDS is faced with a very large number of events in the network, it slows down a system or drops network packets.

- Vulnerability to attacks: Hierarchical structures attackers the opportunity to harm the IDS by cutting off a control branch or even by tacking out the root command.

2.3 Mobile agent advantages.

Instead of using static components in a IDS, mobile agent based systems has the following advantages.

Overcoming Network Latency - Since agents operate directly on the host, where an action has to be taken, and their response is faster than hierarchical systems, where the actions are taken by central coordinator.

Reducing Network Load - Instead of sending audit data from sensors to central stations, sending the code of the agent may cause little network load, because audit data may become huge amounts.

Autonomous Execution - In order to prevent letting the whole network undefended, when a part of the IDS fails, agents can work autonomously even if their creators don't operate anymore.

Platform Independence - Where the agents run on the agent platform, they are independent from the platform of the host.

Dynamic Adaptation - The system can be reconfigured at run-time because of the agents dynamic behavior.

Static Adaptation - When a new attack signature has to be added to the IDS, the algorithm of the agents can be updated without restarting the whole system.

Scalability - Mobile agents reduce the computational load on the system by dividing it different hosts.

3. Intrusion Detection System.

3.1 AAFID (Autonomous Agents For Intrusion Detection).

The Architecture for Intrusion Detection Using Autonomous Agents (AAFID) implements a host based hierarchical design to deal with disadvantages of centralized systems [9] [16]. The system consists of three layers. Each layer calls methods of the layer below. At the base level *agents* collect information, search for suspicious packets and forward collected data to *transceiver*, which exists in the upper layer and once in every host. Transceivers are like agent managers, they control and configure agents on the host, and channel information, most probably exclude unnecessary parts of the collected information and forward it to *monitors*.

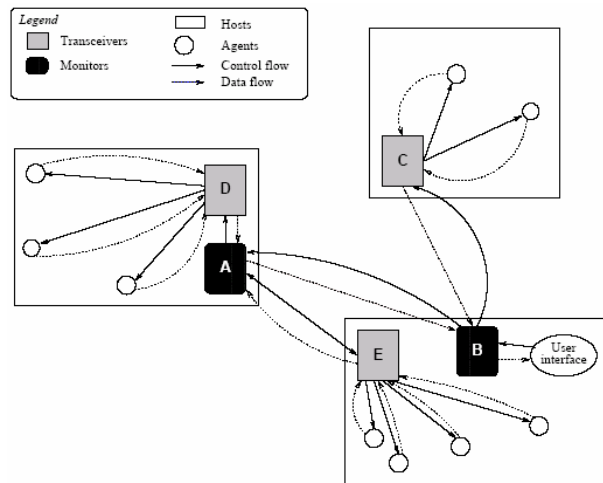


Figure 1: AAFID system

In the higher level, each monitor collects data from one or more transceivers and analyzes their input. As each host has one transceiver, monitors collect data from multiple hosts. Monitors can be set hierarchically, having root monitor at the top, which provides user interface to control the IDS. In order to make agents to communicate with each other, audit information is delivered to agents by the Audit Router. Audit Router holds a database of running agents and their interests. Agents register to this database and get a handle to read the entries of other agents. To sum up, in the AAFID architecture, agents are used to collect and preprocess information which is needed to detect intrusions in the system. They are not intelligent programs, but due to genetic programming they inherit their experience. In addition transceivers and monitors is aimed to make the system scalable, which allow detection of distributed attacks. The major disadvantage of AAFID architectures is the delay in the detection of the intrusion caused by the layers between agents and the monitor and also monitors are the single point of the failure.

3.2 IDA – Intrusion Detection Agent System.

IDA is an implementation of the mobile agent approach in intrusion detection [3] [16]. According to the system designer, attackers follow four common steps to gain unprivileged access to remote machines. In the first step they scan for machines and ports, in the second stage they try to use vulnerabilities of common services. Following stage is the mark stage where the attacker gets actually access to the system and later in the masquerade stage they try to hide their actions by trying to erase logs, etc.

IDA is aimed to detect intrusions by scanning marks left by the intruder, who is actually in the mark

stage. The MLSI, which stands for Mark Left by Suspected.

Intruder is defined by suspicious activities like opening a root shell, trying to change important files, etc. The IDA has a simple structure; it consists of a central manager on each network segment, sensors and multiple kinds of agents. The central manager is responsible for collecting information from sensors and analyzing it. The sensors, as usual monitor network traffic according to MLSIs. The first kind of agent, *Information Gathering*

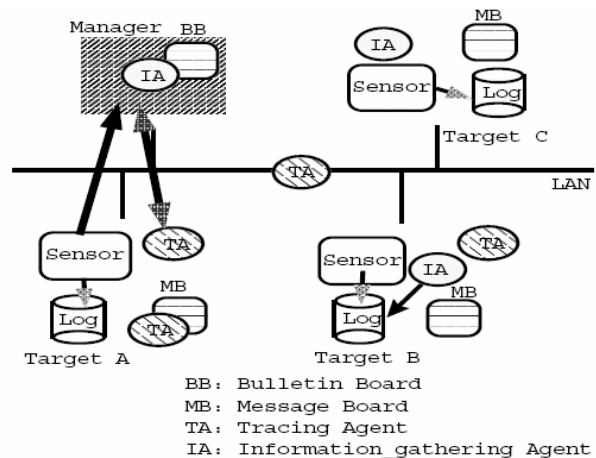


Figure 2: Structure of IDA

Agent (IDA), which exists once for every MLSI, is responsible for examining logs in detail. Another type of agent, *Tracing Agent* finds the source of the attack. It works in coordination with IDAs. Providing agents to communicate with each other and the manager is the *Communication Boards* duty. IDA uses agents to collect information about the intrusion and the intruder, but the management is still centralized. The main disadvantage of the architecture is scalability, because managers can deal with only a limited amount of sensors and agents.

3.3 IMA-IDS.

Intelligent Mobile Agents for Intrusion Detection System: The architecture of IMA-IDS is defined in terms of four agents [5].

Collector agent: This kind of agent will be cloned and distributed throughout the network. This agent patrols the network and collects all the events occurring in the host.

Correlator agent: It collects critical information and sends it to the appropriate analyzer agent without passing through the manager agent. Each correlator agent uses a set of rules that clearly specifies the crucial events, contexts and analyzer agents concerned by an urgent reporting event mechanism.

Analyzer agent: It performs several kind of analysis such as classical signature detection, anomaly detection and a new security protocol analysis based on abstract interpretation.

Manager agent: This agent gathers collected information and distributes it to analyzer agents.



Figure 3: IMA-IDS Architecture

All the collector agents report their results to the manager agent which transmits them to the analyzer agents. The analyzer agent performs a higher level analysis and correlation (Anomaly and Policy detection). The analyzer agents report their results to the manager, and they generate alarms if they detect any anomaly.

In order to communicate, agents are able to know all information about the other agents created and running in the network (their locations, their number, and their identifier) by sending a request to the manager agent. The manager agent uses the following two agents:

- **Registry Agent:** being present on all hosts running agents, it maintains information about the agents running in the host.
- **IdsHost Agent:** it keeps track of all created and running agents.

For communication agents subscribe to one or more multicast message list and implement handlers for communication messages.

3.4 Micael.

Micael is another agent based approach, where decision making, distributed, autonomous agents have the role of investigating intrusions [10][16]. Its architecture consists of the following elements.

Sentinels are static agents existing on every host. They do not have any knowledge about different attacks, but they are against distributed attacks (DoS etc.).

Detachments are mobile agents to investigate intrusions. When a possible attack is detected, they are moved to the related host and start to examine log files

in detail. In case they decide that there is an attack, they can perform various actions from disconnecting the host to counterattacking to the intruder.

Headquarters are centralized agents collecting data from sentinels and creating new detachments if needed.

In an example scenario, the intrusion is detected as it follows.

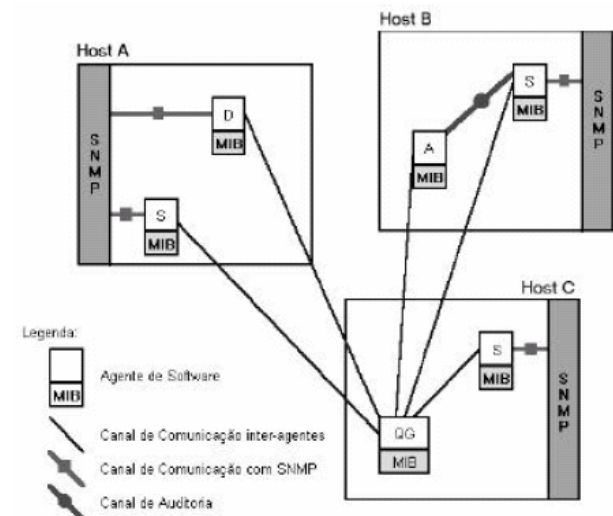


Figure 4: An Example of Micael System

First sentinels look into data for a suspicious activity, when they find, they request for a detachment from the headquarters. The detachment is created by headquarters and sent to the host. After it arrives to the host, it analyzes information more in detail and if decides to an attack, it take actions and then returns the result to headquarters.

Micael is a successful implementation of the agent based approach with several advantages. Its agents are not used only collect data; they can also react to incidents. In addition the whole system is written in Java, therefore it is very portable, can run on any platform supporting Java.

3.5 IA_DIDS (Intelligent Agents for Distributed Intrusion Detection System).

The Specialized Local Agent performs several kinds of attack analysis such as signature detection, anomaly detection and performed global analysis, for detecting distributed attacks [4]. Suspicious network activities are captured by Snort sensor and log files are generated. *Filter Agent* is agent responsible for filtering specialized security events from the log files based on event rules. *Intrusion Event Rule* is a set of requirements that will trigger an alert.

The *Analyzer Agent* analyses the events database. It looks for the local events selected by the *Interpreter Agent*. These patterns are retrieved from *Events DB*.

Analyzer Agent has the pattern matching task to confirm the occurrence of the given events in Event DB. Then, it reports a search results to the *Interpreter Agent* using its Specialized Local Agent.

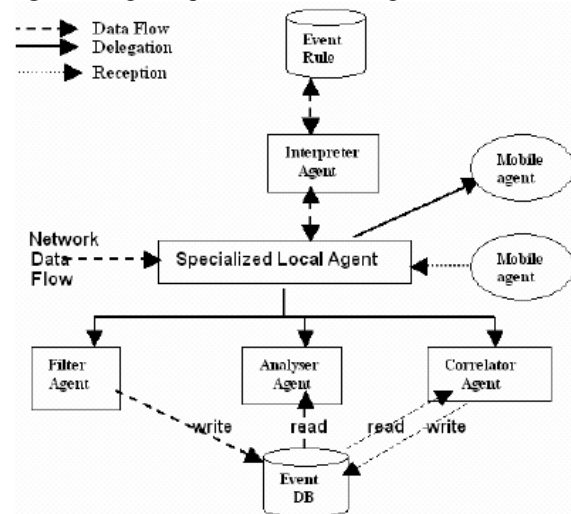


Figure 5: Multi-Agents Architecture

Interpreter Agent queries the events database to search the occurrence of some event, and accesses to database events to store the occurrence of the external event received from other network nodes using *Mobile Agent*. *Interpreter Agent* transmits the alert actions to the specialized local agent.

3.6. Safe MA-IDS.

In this system the mobile agent platform Gypsy is selected because it satisfies the needs of MA-IDS [12]. Gypsy uses the Java security mechanisms for secure class loading. Furthermore, it uses Java Sandbox security model including code signing and class loading. Here some new parts are added and some Changes are made in the architecture and implementation of existing Gypsy platform. There is an agent transmission security module which is designed to build on the benefits of public-key Cryptography, symmetric-key cryptography and message authentication codes (MACS).

First, each agent is encrypted using a symmetric key algorithm with a one-time session key. Then, this session key is then encrypted using a public key algorithm, implementing the authentication of the other side to the originator.

Finally, the MAC about important contents of the agent is computed, for receiver to verify the integrity of the agent upon the arrival. Another module is the agent persistence module, which is responsible for storing mobile agents to help the There are three cases in which agents need to be resuscitated, agents may become lost during migration to other hosts, agents may suffer

malicious attack or deletion and the MA Platform the agents reside in may be at fault. In this module agents are stored with their data and status at regular intervals. If an accident occurs, the agent persistence module will try to regenerate an agent with the latest stored data and status information, then reintroduce it into the network. The advantage of the system is that Autonomous agents may continue operating despite the failure of other failures in the system, which avoids the compromise of the entire IDS even if one component fails or is attacked. Moreover, Mobile agents in this system are capable of evading attackers and resurrecting themselves if killed.

3.7 Preemptive DIDS.

This system is a network-based preemptive distributed intrusion detection system using mobile agents [11]. Packets are diverted to various types of agents strategically placed around the network. Suspected packets are blocked before they reach the destination when an intrusion is detected and the policy mechanism verdicts for blockage.

There are five types of agents in this architecture; they are the gateway agent, the controller agent, the detection agent, the policy agent, and the home agent

A gateway agent is an agent that resides in the gateway between the internal network and the external network. It is responsible for grabbing packets from the external network and sending them to one of the controller agents in the internal network for the intrusion detection. The decision to send a particular packet to which controller agent depends on (1) the workload of the host computer where the controller agent resides, (2) the session which the packet belongs to, and (3) the cluster which the controller agent is involved.

A controller agent is an agent residing in computers in the internal network. It receives packets either from the home agents (outgoing/internal traffic) or the gateway agents (incoming traffic). Then it transfers the received packets to the detection agents for the intrusion detection.

A detection agent is an agent responsible for the actual intrusion detection job. Each controller agent controls a number of detection agents. Each detection agent is only responsible for detecting certain types of intrusions. This makes it easier for updating when new types of intrusion is found or new types of detection method is invented.

The detection agents get the packet from the controller agent and the information from the leader controller agents in the cluster. Then, they determine whether the packet is an intrusion or not and estimate the false alarm rate. A home agent is an agent responsible for receiving (incoming) packets from and sending (outgoing) packets to the controller agent).

When it receives the packet and the detection result from the controller agent, it consults the policy agent on what action it should take. A policy agent is an agent responsible for what action a home agent or a gateway agent should take when an dropping the packet, letting the packet pass through but informing the SSO, or just logging the event down without informing the SSO, etc. Preemptive IDS has long been criticized for just reporting instead of stopping an intrusion in a timely fashion. In our design, if an intrusion is found and the policy agent verdicts for blockage, those suspected packets can be discarded before reaching the process in the destination host.

4. Conclusion.

Intrusion Detection Systems (IDS) plays an important role in achieving survivability of information system and preserving their safety from attacks. Centralized IDS consumes lot of network resources and is a single point of failure. So Mobile agent platforms is used to overcome deficiencies of centralized IDS, efficiently manage the system and dynamically adapt to network changes and event rules. The mobile agent based IDS shows superior performance than centralized IDS and is able to report intrusion instantly. In this paper we provide general definition of Intrusion detection system architecture, requirements and limitations. We also present survey of different Intrusion detection system.

5. References.

- [1] M. Eid, "A New Mobile Agent-Based Intrusion detection System Using distributed Sensors", *In proceeding of FEASC*, 2004.
- [2] C. Kruegel and T. Toth "Applying Mobile Agent Technology to Intrusion Detection", *technical report*, University of Vienna, TUV- 1841-2002-31, 2002.
- [3] Midori Asaka, Atsushi Taguchi, Shigeki Goto. The Implementation of IDA: An Intrusion Detection Agent System
- [4] M. Benattou, and K. Tamine. Intelligent Agents for Distributed Intrusion Detection System.
- [5] F. Barika, N. EL Kadhi, and K. Ghedira, "Intelligent and mobile agent for intrusion detection system: Ima-ids," in 1st ICICT Conference, 9 2003..
- [6] J. D. De Queiroz, L. F. R. Da Costa, and L. Pirmez. Micael: An autonomous mobile agent system to protect new generation networked application. In 2nd Annual Workshop on Recent Advances in Intrusion Detection, 1999.
- [7] W. Jansen, P. Mell, T. Karygiannis, and D. Marks. "applying mobile agents to intrusion detection and response". Technical report, NIST Interim Report - 6416, October 1999.
- [8] B. Mukherjee, T. L. Heberlein, and K. N. Levitt. Network intrusion detection. In IEEE Network, 1994.
- [9] J. S. Balasubramaniam, J. O. Garcia-Fernandez, D. Isacoff, E. Spafford, and D. Zamboni. An architecture for intrusion detection using autonomous agents. In Proceedings of the 14th Annual Computer Security Applications Conference, 1998.
- [10] J. D. De Queiroz, L. F. R. Da Costa, and L. Pirmez. Micael: An autonomous mobile agent system to protect new generation networked application. In 2nd Annual Workshop on Recent Advances in Intrusion Detection, 1999.
- [11] P.C.Chan and Victor K.Wei " Preemptive Distributed ntrusion Detection using Mobile Agents" .In Proceedings of the eleventh IEEE International Workshop on Enabling echnologies :Infrastructure for Collaborative Enterprises 1080-1383/02 2002 IEEE
- [12] Shao-chun Zhong, Qing-Feng Song, Xiao-Chun Cheng, Yan Zhang "A Safe mobile Agent System for distributed intrusion detection".In proceedings of the second international nference on machine learning and cybernatics, Xi'an, 2003 IEEE.
- [13] M. Roesch, "Snort: Lightweight Intrusion detection for networks", *A white paper* on the design features of snort 2.0, 2004.
- [14] J. Barrus and N. Rowe, "Distributed Autonomous-Agent Network- Intrusion detection and response System. *In proceeding of Command and Control research and Technologies Symposium*, 1998.
- [15] T. H. Ptacek and T. N. Newsham. "Insertion, evasion, and denial of service: Eluding network intrusion detection". Technical report, Secure Network Inc., January 1998
- [16] Haken Albag "Network and agent based Intrusion detection System"