CHAPTER 4

HYBRID INTRUSION DETECTION SYSTEM BASED ON CLUSTER WITH MOBILE AGENT

4.1 INTRODUCTION

Intrusion detection is defined as the process of intelligently monitoring the events occurring in a computer system or network and analyzing them for signs of violation of security policies. The primary aim of intrusion detection system is to protect the availability and confidentiality (Siqueira & Abdelouahab 2006) of network information system. Sun et al (2007) showed that the data collection process used by the MA is distributed and it does not focus on a specific network. The proposed intrusion detection system is based on the Graph based Intrusion Detection System (GRIDS) that makes use of a tree-format structure to reduce the network complexity. The tree format structure improves the speed of the MA thus enabling all the nearby and remote nodes to be analyzed and finalized at distribution observation regions.

Within AAFID, the nodes of the IDS are placed in the focus of a tree structure. Mediators in AAFID are not mobile (Sun et al 2007). Recent DIDS typically make use of distributed mechanisms to gather information which is then passed on to the hub processor center. In case of a large wide area network, the distributed information collected and stored in the hub processor center is enormous. This leads to delay in network transmission time, network traffic problems, out of order delivery of information from
the server in response to the requests. This DIDS is completely distributed and causes single point of failure because of the central processing node (Ning 2001).

Mitrokotsa et al (2000) proposed a dynamic clonal based on selection algorithm that reduces false positive rate. Identifiers are fixed in mediators with message devices and the messages sent by the mediators, new mediators act accordingly. Using the mobile characteristics of mediators, the recognizable mediator can move to local host, and thus it can trim down network load which can get better real-time capability. The entire model is fully distributed.

The intrusion detection system is used to detect attacks and its placement in the network. Hence, they need more time to detect an intrusion. The architecture of the Traditional Intrusion Detection System (TIDS) is used in the lower layer of the architecture proposed by (Sun et al 2007). The TIDS architecture deals with the information being collected on each machine separately. The information from every machine is then forwarded to the registered file. In the event of node failures, the amount of information lost is more. This has led to the development of DIDS (Anjum et al 2003).

Mostly on the network side, encrypted communication alone is randomly used. Networks based variants monitor packets on the channel by setting the network interface to promiscuous mode and analyze network traffic (Thangavel & Thangaraj 2011). Therefore they have some possibilities to correlate the activities that occur at different hosts, but suffer from scalability problems in case of high network load and encounter problems when encrypted communication is used. However, they need more time for communication between agents to detect intrusions. For instance, the coefficient procedure proposed by (Barika et al 2009) have added mobile agent in a distributed environment to detect intrusions.
The procedure proposed by (Cao & Zheng 2008) mentions that an agent doesn't only work independently, but is also a controlled to perceive the change in environment and behaves accordingly. But the entire agent will cooperate in detecting intrusions in some circumstances.

The existing security solutions are very complex and costly. What is rapidly needed is a flexible, adaptable and affordable security solution, which provides greater autonomy. Therefore, it is necessary to review the way security system architectures are designed in investigating new technologies that could help make easier and cost-effective new solutions.

This chapter presents a new distributed procedure, called Hybrid Intrusion Detection System based on cluster with mobile agent and is organized as follows. Section 4.2 describes cluster formation by using a standard election algorithm. Section 4.3 proposes the system architecture of cluster based intrusion detection system. Section 4.4 describes the fundamental concepts of Dynamic Distributed Intrusion Detection System with Mobile Agent (DDIDS-MA) model. Section 4.5 describes the system architecture of DDIDS-MA. Section 4.6 describes the procedure of intrusion detection mechanism. In Section 4.7, the performance of the proposed procedure is analyzed and compared with the existing distributed procedures. Finally, Section 4.8 contains the concluding remarks.

4.2 CLUSTER FORMATION
stability and should give the lesser number of acceptable errors. The proposed
cluster formation strategy puts forth a selection process that is easy and which
does not require group calculation or nearby node information. Each node in
the cluster has to contribute in the selection procedures by throwing their
eagerness.

Figure 4.1 shows how the clusters are formed, how clusters heads
are chosen and how the cluster heads of different clusters interact with
each other (Samad et al 2005) and (Huang & Lee 2003). The election
algorithm is carried out using the following steps (Zhang & Lee 2000) and
(Kachirski & Guha 2002).

![Figure 4.1](image)

Figure 4.1 (Continued)
Each node finds its neighbors by sending alert messages frequently and identifies whether the adjacent node is active or not from its response. Thus every node collects its neighbor information.

At the start of the algorithm, every node does intrusion detection, for thus behaving as a single node cluster.

All the nodes calculate a hash value based on the election parameters such as computational power, power level and a random number and broadcast these hash values.

The original values of the election parameters are then broadcast.

The nodes compare the values obtained in the above two steps and verify that they are the same.

All nodes run the election algorithm and calculate the same values for the head node.
The cluster head node maintains a member list which is used for storing information about each node.

The mobile agent periodically checks the connectivity of links and is also responsible for periodic election timer checks.

Failure of links causes a node to be out of the network.

When the cluster head node moves out of the network then all the nodes in the cluster enter into the re-election procedure.

When the master node has been attacked either through the results of the signature detection or through anomaly detection, it calls for re-election after isolating the master node.

Periodic re-election takes place after a timeout to ensure fairness in service and avoid a single-point control and monitoring.

4.3 SYSTEM ARCHITECTURE

Figure 4.2 shows the simple architecture of an intrusion detection system with a mobile agent based on clustering. In this architecture, creation of a mobile agent, response agent, control agent, detection agent and manage agent is done.

4.3.1 Mobile Agent

Mobile agent has many benefits. Mobile Agent is able to decrease the network load. This agent has the capacity to provide an energetic relation between the client and the server and hence has the ability to identify even small errors in the network. The mobile agent is responsible for creating
awareness about the IDS to the cluster head by storing the complete information about how the cluster head has to behave. Thus the MA provides a clear picture of the IDS to the cluster head. A mobile agent is a kind of a typical mediator, which supervises and manages roaming mediators on the network.

---

Figure 4.2 Architecture of Mobile Agent Based Intrusion Detection System with Cluster Head

4.3.2 Response Agent and Control Agent

Response and Control agent executes in the server and takes the role of a manager on the server side. Mobile agent travels through the entire network to identify the attacks (Li et al 2009). If any attack is identified by the control agent, then it moves to the respective network. It then analyzes and checks whether the attack is similar to known intrusions. If there is a match, then the response agent becomes active and performs the necessary corrective action. If it does not match, then the attack is identified as an unknown attack and the details about it are stored on the server.
In future, if the control agent encounters an attack very similar to this known attack, then it contacts the response agent to perform its task. Depending on the type of attack, the response agent and control agent ask the manage agent to take over the manage agent is responsible for removing the corresponding node. In order to calculate the IDS inequality may be used the Chebyshev's inequality model specified in equation (4.1),

\[
P(X - \mu) \geq \varepsilon \leq \varepsilon^2
\]

Hence according to this formula X term is the number of clusters which should be used, \(\mu\) represents the number of clusters to be connected and \(\varepsilon^2\) represents the value in the network where the identifiers are to be detected.

4.3.3 Cluster Head

The cluster-head maintains an election interval timer for managing the elections. Every node in the cluster participates in the election process by casting their vote and shows their willingness to become the cluster-head. The node showing the highest willingness becomes the cluster-head until the next timeout period. The cluster management responsibility is circulated among the capable members of the cluster for load balancing and fault tolerance and must be fair and secure.

4.4 FUNDAMENTAL DDIDS WITH MOBILE AGENT (DDIDS-MA) MODEL

The agent is an entity that is able to accomplish its assigned task without manual intervention and supervision under certain conditions. It is self-adaptable, intelligent and collaborative. Dynamic detection refers to the process of identifying and locating the problems in a distributed environment dynamically.
Figure 4.3 shows the basic architecture of Dynamic Distributed Intrusion Detection System based on the mobile agent model. The IDS also makes the process of finding the users who try to access the authorized events in the system / network. In this system, each DDIDS-MA has a connection with the other agents for information sharing about an intrusion and to take the necessary steps to solve the current problem arising in the network. At the same time this will help to neglect an unwanted event. DDIDS-MA consoles will have a control of every agent in a network and will support report preparation.

![Diagram of DDIDS-Mobile Agent Model]

**Figure 4.3 Fundamental DDIDS-Mobile Agent Model**

In the TIDS, every node acted as individual IDS, thereby increasing the network load and degrading the network performance. There are two kinds of agents: static agent and mobile agent (Cao & Zheng 2008). Static agent is the first proposed (Jun et al 2006) agent technology which is applied in the area of intrusion detection. Static agent, that is to say, the agent that resides in a fixed position or some fixed platform.
Mobile agent is an entity capable to move from one node to another in the network. It permits to spread dynamically the server interfaces managed on the different sites. It guarantees a big resistance to network breakdowns; it also permits saving of bandwidth since negotiations between mobile agents and the server consists of an exchange of local messages that don’t bypass the network. Instead of using static components in IDS, mobile agent based systems provide the following capabilities: get the better of network latency; have natural network loads; agents can work individually; use the usual methods to structure and design an IDS; work in various surroundings; the structure can be reconfigured at run-time; ability to react dynamically in unfavorable situations and updating can be done without restarting the whole system. This study which proposes the Dynamic Distributed Intrusion Detection System based on Mobile Agent (DDIDS-MA) improves the system processing and system networking.

4.5 SYSTEM ARCHITECTURE

Based on the merits of agent technology, this study puts forward a dynamic distributed intrusion detection system based on mobile agent. It consists of Host Negotiator, Net Negotiator and Mobile Negotiator.

4.5.1 Network Connection

Here, the term network denotes the connection between particular systems or machines (Benattou & Tamine 2005). A single network consists of many numbers of systems that to connect internally. Each and every individual system has its own address. If one system in the network malfunctions/fails, information can be gathered from the neighboring system. This technique is adopted by systems connected in a similar manner in all the remaining networks. Similarly the remaining system also follows the same procedure for sharing the certain information or messages.
Every workstation in the model is connected to a switch. Switches communicate with each other through a router. If a fault is found, this information will not be conveyed by the switch or the router. This is where the DDIDS plays a major role. Here, the network plays an important and major role in information sharing between the systems. By using DDIDS-MA the efficiency of the system increases and information can also be transferred very easily.

### 4.5.2 Host Negotiator

The main operation of the host negotiator system is to protect the system from its place. When unauthorized actions are not to be determined, the host negotiator generates an ID event and transmits it to a centralized database through the mobile agent. Every type of ID event indicates a kind of possible attack. In Figure 4.4 the detailed architecture of DDIDS-MA along with the workstation and agents is shown. The interlinks between all kinds of DDIDS Mobile agent system will share the intrusion data with the support of console window. This console window is useful for the administrator to gather information about the intrusions happening in the network.

Host negotiator intrusion detection systems to evaluate data that originates on computers, such as application and operating system event logs and file attributes. These host event logs contain information about file accesses and program executions associated with inside users. If protected correctly, the event logs may be entered into the database to support the action of computer intruders in the future. The host negotiator system is designed to prevent inside intruders, but can’t effectively prevent an outside intruder. An inside intruder who misuses his privileges, will be detected by a host negotiator. Host negotiator system mainly concentrates on misuse of privilege.
Figure 4.4 System Architecture of DDIDS-MA

The host negotiator system deals with very low real-time response and cannot efficiently safeguard against one-time disastrous events. Host negotiator system stands out when it comes to determining the level of cooperation after maltreatment. They usually maintain large databases that store the historical information about the prosecution of the misuse.
4.5.3 Net Negotiator

The role of the net negotiator is to detect network intrusions. It is installed in the neuralgic places of the network, for example, at the entrance of the network or on the server. The net negotiator supervises the network traffic; a record all suspected events in a database, responds to intrusions and also installs a mobile agent on it. Both net negotiator and host negotiator are used for detecting intrusions. But the way in which they perform the detection is entirely different. Net negotiator is effective in real-time detection and in providing the response.

Net negotiator intrusion detection can also be effective in detecting long-term attacks from outside the firewall; the insider won’t need to achieve any of the transactions normally detected by network negotiator intrusion detection. To put it in simple words, host negotiator intrusion detection detects insider misuse while network negotiator intrusion detection detects outsider misuse. Also, network negotiator intrusion detection focuses more on abuse of problems whereas an insider does not have to exploit problems because they are already in the network and have their own privileges.

4.5.4 Agent System

The agent system consists of a mobile negotiator for the continuous observation of the host negotiators and net negotiators. Most of the useful and practical mobile negotiator-based application is based on using dynamic itineraries for free roaming agents, in which the host platforms are discovered at runtime. The mobile negotiator visits all the hosts in the network and collects the related attack data information (Yongle et al 2003). For an information retrieval application, the mobile agent has to migrate to a group of nodes in a particular sequence, and the goal is to maximize the information extracted while keeping the freshness of data and minimizing the travelling
time. Then it correlates the collected information with the data it has received from the other host agent or net agent that has generated the same type of ID event.

Most importantly, it should not be affected by the failure of the network components (Yang et al 2006). The mobile agents are allowed to carry the monitoring tasks to wherever they are needed. The MA sends out alerts when it detects intrusion. On one hand, it informs every host it has visited to take the corresponding response measures and indicates the alerts, on the other hand, it transmits these related information to the centralized database and shows them on the user interface. Multiple mobile agents are used to collect data from the nodes of different network partitions and thus maintaining the deadline to an affordable limit. The data available to the mobile agent is concatenated with the retrieved data from the host and the encrypted information is forwarded to the server administrator.

4.6 PROCEDURE OF INTRUSION DETECTION MECHANISM

This section, first proposes the concept of cluster formation in intrusion detection system. After the formation of clusters, the procedure for installing the mobile agent on every host in the network is proposed. Given below is the detailed description of cluster formation with mobile agent.

4.6.1 K-Means Algorithm

Input : The number of clusters K and a dataset for intrusion detection

Output : A set of K-clusters that minimize the squared error criterion.
Algorithm

Step 1 : Initialize K clusters (randomly select k elements from the data).

Step 2 : While cluster structure changes, repeat from step 2.

Step 3 : Determine the cluster to which the source data belong use election algorithm to elect a head node.

Step 4 : Determine the node based on the various parameter levels such as computational power and use Euclidean distance formula.

Add element to cluster with min (Distance (xi, yj)).

Step 5 : Calculate the means of the clusters.

Step 6 : If suspicious node is found then remove that node from the network and change cluster centroids to means obtained using Step 3.

The K-means clustering is a classical clustering algorithm. After an initial random assignment to K clusters, the centers of clusters are computed and assigned to the clusters with the closest centers. The process is repeated until the cluster centers do not significantly change. Once the cluster assignment is fixed, the mean distance of an example to cluster centers is used as the score. Using the K-means clustering algorithm, different clusters are specified and generated for each output class. K-means clustering algorithm has been used in an attempt to detect anomalous user behavior, as well as unusual behavior in network traffic.

Two problems are inherent to K-means clustering algorithms. The first is determining the initial partition and the second is determining the optimal number of clusters. As the algorithm iterates through the training data, each cluster’s architecture is updated. In updating clusters, elements are moved from one cluster to another. The updating of clusters causes the values
of the centroids to change. This change is a reflection of the current cluster elements. Once there are no changes to any cluster, the training of the K-Means algorithm is complete. The K cluster centroids are created and the algorithm is ready for classifying traffic. For each element to be clustered, the cluster centroids with the minimal Euclidean distance from the element will be the cluster for which the element will be a member.

4.6.2 DDIDS-MA Algorithm

**Step1** : Check the connection state of every node or host in a system.

**Step2** : Check whether the host is connected to the switch.

**Step3** : Check whether the switch is connected to DDIDS-MA in the Network.

**Step4** : Make an interlink between all IDS in the network.

**Step5** : Each IDS in the switch has to check the intrusion in the network.

```plaintext
if (intrusion)
    Update into corresponding IDS
else
    Repeat step5
```

**Step6** : Share the IDS with all network IDS.

**Step7** : DDIDS Mobile Agent has to connect with network IDS.

**Step8** : The entire process (normal/abnormal behavior) is updated on the server as well as communicated to all nodes in the network.
4.7 IMPLEMENTATION DISCUSSION AND RESULTS

In this experiment network plays a very important role in order to connect each and every system. LAN is the most widely used network in almost all places, hence a LAN environment with a bandwidth of 100 Mbps is used for this experiment. The attack activities are simulated with the target as a host on the LAN. All types of attacks are performed on the proposed Dynamic Distributed Intrusion Detection System with Mobile Agent (DDIDS-MA) architecture.

The earlier methods of intrusion detection (TIDS & MDIDS) showed an increase in the fault acceptance level whenever there was an increase in fault overhead. But the proposed method (DDIDS-MA) is best when compared to the above two as a decrease in the fault acceptance level is seen even when there is an increase in fault overhead. This is evident from Figure 4.5.

![Figure 4.5 System Faults Detection Efficiency](image)

**Table 4.1** shows the different techniques that have been implemented for intrusion detection in a network.
Table 4.1 Comparison with Different IDS Techniques

<table>
<thead>
<tr>
<th>Techniques</th>
<th>Fault Overhead %</th>
<th>Fault Acceptance %</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIDS</td>
<td>35</td>
<td>60</td>
</tr>
<tr>
<td>MDIDS</td>
<td>63</td>
<td>40</td>
</tr>
<tr>
<td>DDIDS-MA</td>
<td>70</td>
<td>15</td>
</tr>
</tbody>
</table>

This system is provided with high detectivity. By the introduction of Mobile Agents, the system is able to analyze and classify in real-time the state of intruding activity, so that it can detect the attack correctly. The proposed design fault detection rate is higher than the existing systems. This is due to the shared connections for information sharing between IDS, thereby providing a perfect solution. This system is also provided with high flexibility and fault tolerance. To prevent allowing the whole network go undefended, when a part of the IDS fails, agents can work autonomously even if their creators don’t operate anymore.

4.7.1 Comparison of DDIDS-MA Method with other IDS Methods

Like TDIDS and MDIDS

The research is carried out on a wired network with a bandwidth of 100 Mbps. There are many numbers of attacks that has been tested in order to prevent them in the network. The attack activities are simulated with the target as a host on the LAN. This DDIDS-MA system not only blocks the application oriented issues but stops some of the network security issues. All types of attacks are performed on the proposed Intrusion Detection System (IDS) architecture. The DDIDS-MA system is also provided with high flexibility and expandability. Suppose if an agent is not working due to some problem, then the other agent in the same layer will do the work. Thus the flexibility and expandability has been improved.
Table 4.2 compares the performance of proposed DDIDS-MA with all distributed intrusion detection procedures in the literature in terms of alarm times and false alarm times. With these two performance measures, Table 4.2 clearly shows that the performance of the proposed algorithm is better than the existing algorithms. A connection is a sequence of TCP packets starting and ending at some well-defined times, between which data flows to and from a source IP address to a target IP address using some well-defined protocol. C# is used as the language for writing the code for the improved IDS. The results obviously expose that this method is able to achieve much higher detection rates with a negligible false detection rate. As the number of attacks increased the percentage of overall correctness is also increased.

Table 4.2  Performance Comparison of DDIDS-MA with Existing Intrusion Detection Algorithms for Detecting Generalized Alarm Times

<table>
<thead>
<tr>
<th>IDS Type</th>
<th>Alarm Times</th>
<th>False alarm times</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Set-1</td>
<td>Set-2</td>
</tr>
<tr>
<td>Cao &amp; Zheng (2008) Mobile agent Distributed Intrusion Detection System (MDIDS)</td>
<td>95</td>
<td>186</td>
</tr>
<tr>
<td>Dynamic Distributed Intrusion Detection System using Mobile Agent with Fault Tolerance (DDIDS-MA)</td>
<td>96</td>
<td>190</td>
</tr>
</tbody>
</table>

Normally, in any environment if the number of objects required for proper intrusion detection is greater than the minimum level then the system is proven efficient. Figures 4.6 and 4.7 shows the comparison result of TDIDS, MDIDS and DDIDS-MA.
The test compares three different intrusion detection systems: Traditional Distributed IDS (TDIDS), Mobile Agent based Distributed IDS (MDIDS) and Dynamic Distributed Intrusion Detection System using Mobile Agent (DDIDS-MA).

**Figure 4.6 System Results of TDIDS, MDIDS and DDIDS-MA**

**Figure 4.7 System Attack Detection Efficiency**
4.8 SUMMARY

The proposed model has the following advantages. The first and foremost advantage is that false positive rate is low and true positive rate is high by the adoption of the mobile agent selection algorithm. The second advantage is that real time potential is enhanced and bottleneck problem is overcome as the proposed model that is always up-to-date with information from the mobile agent. The third advantage is that dependability of the system is enhanced when compared with other hierarchical models it surmounts single point of failure. In addition, the cluster based IDS scheme was found to be simple and offers low overhead in terms of memory usage and number of messages exchanged. The fast, efficient and fair election process used for the selection of monitoring node has reduced the number of packets exchanged for cluster formation and intrusion detection. Finally DDIDS-MA provides high accuracy and low false rate alarm.

This system cannot be cheated easily even where there exists small pattern changes. It has increased the network efficiency. The system is robust and fault-tolerant. The final result is that known and unknown or new attacks are detected by the proposed architecture. The new improved dynamic distributed intrusion detection system with mobile agent provides detailed experimental results that demonstrate the advantage of the DDIDS-MA over other methods. To assess the performance of false alarm level in the network, 3 different data sets have been used to show that the proposed method outperforms the existing ones. There is some variation in the results for the attack detectors, when considering changes in clarification. During the experiment, it was observed that

- The improved dynamic distributed mobile agent intrusion detection system detects different kind of attacks.
• The improved dynamic distributed mobile agent intrusion detection system method shows a better performance for finding false alarm object and attack detection.

It was also observed that the edge flow method does not show encouraging results for the following variations.

• The performance of the intrusion detection method is reduced in large networks.

• The false alarm rate is not accurate in a heterogeneous environment.

Consequently, all these observations, drawbacks and advantages were analyzed. The next chapter focuses on the effective intrusion detection method that uses the concept of mobile agents. It describes the robust and stable tracking framework for finding intrusion. It also describes the integration of the cluster based method and dynamically distributed intrusion detection method using concepts of by mobile service agent.