A Self-Organization Agent-based Architecture for Power-Aware Intrusion Detection in Wireless Ad-Hoc Networks

T. Srinivasan1, Vivek Vijaykumar2, and R. Chandrasekar3

Department of Computer Science and Engineering
Department of Information Technology
Department of Information Technology
Sri Venkateswara College of Engineering
Sriperumbudur, India
1tsrini@svce.ac.in, 2vivekvvk@gmail.com, 3chandra85@gmail.com

Abstract—In this paper we propose SAPID-A Self-organized Agent-based architecture for Power-aware Intrusion Detection in wireless ad-hoc networks. We utilize an agent-based architecture that conserves the available bandwidth and segregates SAPID into two phases. Based on a power level metric and a hybrid metric that determine the duration and kinds of traffic that can be supported by a network-monitoring node, potential ad-hoc hosts are identified by repetitive training using an Adaptive Resonance Theory module. The agent architecture primarily consists of a Kohonen Self-Organizing Map to identify the appropriate patterns and recognize anomalies by way of unauthorized users in the network. A UNIX based session information file is utilized for testing and reporting possible intrusion attempts to the decision and action modules. Comprehensive experiments were carried out to clearly delineate and analyze the performance of the architecture.

Index Terms—Ad-hoc networks, adaptive resonance theory, agent-based architectures for intrusion detection, self-organizing maps

I. INTRODUCTION

The application of wireless communication has become vast with the growing technology of ad-hoc networks gaining prominence in recent years. This developing technology calls for various proposals for researching into newer areas of existing ad-hoc networking techniques. One such proposal includes an efficient mechanism for intrusion detection in ad-hoc systems. Intrusion detection systems have been an active area of research and development since some time. Increase in speed and complexity of computer networks, as well as public access to them, have sharpened the need for effective intrusion detection approaches [2]. Here we describe an innovative approach for intrusion detection by succeeding in developing a method for selecting a node in the network as a monitoring node and then provide efficient agent-based intrusion detection system architecture.

Unlike most approaches, these networks could recognize potentially harmful activity in a network without utilizing the predefined attack signatures that limit the effectiveness of the traditional systems.

An Intrusion is defined as “any set of actions that attempt to compromise the integrity, confidentiality, or availability of a resource” [25]. Several algorithms have been published in recent years to deal with intrusion detection, which have incorporated the essence of wireless nature of wireless networks. Intrusions in wireless networks amount to interception, interruption or fabrication of data transmitted across nodes. Intrusion into a wireless network is possible if an intruder node attempts to access nodes. Ad-hoc networks are particularly prone to such dangers, considering the dynamic nature and geographically distributed nature of the nodes. Furthermore, Ad-hoc networks can be classified on the basis of their dynamic nature as minimally mobile or highly mobile [5]. In this paper, we primarily focus on minimally mobile networks, where the power levels and other set of metrics such as duration of the node usage, availability of the network, channel utilization, quality of service are absolutely critical in determining the kind of processes that can be sustained.

In this paper we concentrate on agent based systems for intrusion detection. The intrusion detection job is executed by the agents on select hosts whose suitability for the job is determined based on several factors broadly classified into two categories. We describe a mechanism called Self-organized Agent based architecture for Power-aware Intrusion Detection (SAPID). SAPID provides an efficient mechanism for selecting nodes from the set of nodes in the network for network monitoring activity. The later half of the paper discusses about agents that are distributed across the monitoring nodes. The main aim of this paper is to provide an efficient method for distributing the monitoring activity among those nodes that have maximum remaining battery power and those which are less prone to traffic and make less utilization of the channel. This kind of system is setup because monitoring nodes must perform their activities without any inhibition and must posses enough battery power for sustaining their regular activities as well as for monitoring.

SAPID is organized into two different phases, the first phase talks of node selection and the second phase talks of agent architecture for intrusion detection. It takes into account the importance of using neural networks for intrusion detection. Neural networks provide a wide range of advantages and it finds application in many areas such as pattern recognition and pattern
classification. These two main applications of neural networks are taken into consideration in this paper.

The first phase of SAPID incorporates a popular type of neural network architecture namely the Adaptive Resonance Theory (ART). ART acts like a filter [26] for pattern matching and selecting the most suitable pattern from the list of patterns. In SAPID we classify the possible metrics associated with wireless networks, as mentioned in the earlier part of section I, into two broad categories. They are the Power metric (PLANE) derived from [5] and a hybrid metric which comprises of other metrics. The two broad categories of metrics form a pair and play a vital role in deciding the process of selection of nodes for network monitoring or intrusion detection purposes. Initially a list of nodes satisfying the Power metric criteria are chosen and then they are scaled down to a single node based on the hybrid metric. The process of selection of nodes is an iterative process taking place until the entire network is covered. Each of the selected nodes monitors only its adjacent nodes and those adjacent nodes are marked as when they are selected so that redundancy is avoided. The end of an iteration in ART culminates in its association with a training module. The training module trains the neurons of the selected node to fish for a set of adjacent nodes that it monitors. Thus the ART provides an optimized approach to the selection process.

The second phase of SAPID describes an efficient method for performing agent-based operations such as monitoring and decision-making. Here again we use one of the more recent concepts in neural networks namely the self-organizing nature of neurons [23] that is an unsupervised learning process. The monitoring nodes learn the environment on a continuous basis and keep accumulating knowledge about the network. To achieve this we use the Self Organization Map (SOM). Here the agents identify various patterns that are associated with the network when an authentic user logs in or when he logs out. When there is a variation in the regular pattern, the SOM classifies the patterns as regular or unusual ones. The unusual ones depict the intrusion by an unauthenticated user. Based on the pattern classification the monitoring nodes take the necessary decisions like terminating a critical process or disconnecting a user. Thus by utilizing the applications and advantages of neural networks we were able to develop an efficient mechanism for power-aware intrusion detection using an agent-based architecture.

II. RELATED WORK

The intrusion problem can typically be tackled by adding additional intrusion detection layers on top of the protocol, or through alterations to the wireless protocol itself. For the former style of enforcing security, two types of intrusion detection systems (IDS) are typically used, as a reminiscence of wired intrusion detection techniques [25]. Network-based systems (NIDS) can be passive or active systems, listening in on network traffic. By capturing and examining individual packets flowing through a network, NIDS can analyze across all layers of the network protocol and are able to look at the payload within a packet, to see which particular host application is being accessed and with what options, and raise alerts when an attacker tries to exploit a bug in such code, by detecting known attack signatures. NIDS often require dedicated hosts or special equipment, and thus can be prone to network attacks. Further considerations are discussed in [6] and [7]. Host-based intrusion detection systems [8] and [9] monitor each individual host by running on each host. They are able to detect actions such as repeated failed access attempts or changes to critical system files, and normally operate by accessing log files or monitoring real-time system usage [5]. To ensure effective operation, host IDS clients have to be installed on every host on the network, tailored to the specific host configuration.

Host-based systems require dedicated processes to run for network monitoring, and, as their name suggests, are not bandwidth dependent. The disadvantage of such comprehensive host-based systems is that they can considerably slow down the hosts that have IDS clients installed. To circumvent these problems agent-based lightweight models were proposed for wireless networks, which are more bandwidth efficient, and provide a heuristic approach to intrusion detection. Protocol-based security measures provide for encryption mechanisms and other extensions such as one-way hash chains used in [4], to deal with routing update attacks. The important concepts of self-organizing maps are an area of latest research providing much promise through its dynamism and flexibility in training of the data thereby accounting for many kinds of attacks prevalent in today’s rapidly growing information age.

Certain protocols such as that in [4] seem to improve on the base protocol but comparisons with other approaches is still in its infancy and the increased cryptographic overhead in some applications cannot be justified. Examples of protocol-based security extensions and measures can be found in [3], [4], [11], [12] and [13]. There have also been many approaches that use Neural Networks for intrusion detection.

A Network-based neural network detection system in which packet-level network data was retrieved from a database and then classified according to nine packet characteristics and presented to a neural network was developed by [21]. Self-Organizing Maps (SOMs) have also been used as anomaly intrusion detectors in [24]. In that work, a SOM was used to cluster and then graphically display the network data for the user to
determine which clusters contained attacks. An effective combination of Adaptive Resonance Theory and Self Organizing Maps is up to our knowledge uniquely proposed in this paper.

III. SYSTEM ARCHITECTURE

In tune with the requirements of choosing the monitoring nodes and performing the agent-based intrusion detection, SAPID is organized into two phases. The first phase defines an efficient method of selecting monitoring nodes for the network. The second phase describes the hierarchical agent architecture for the monitoring nodes.

A. Node Selection

The selection of the monitoring nodes is essentially based on the two parameters or metrics. The two metrics are then utilized in an Adaptive Resonance Theory (ART) module and trained to get the optimum nodes satisfying both the metrics. A central feature of all ART systems is a pattern matching process that compares an external input with the internal memory of an active code [26]. ART matching leads either to a resonant state, which persists long enough to permit learning, or to a parallel memory search. If the search ends at an established code, the memory representation may either remain the same or incorporate new information from matched portions of the current input. If the search ends at a new code, the memory representation learns the current input. The first parameter chosen is a power metric derived from [5] in accordance with the requirements of a minimally mobile system where power considerations are an important aspect of the system. This metric identifies the duration for which a node can support a network monitoring activity. The second is a combination of three metrics and is hence called as a hybrid metric denoted by $H(m)$. It comprises of the following: $T_d$ (Duration for which the node is active), $B_w$ (Bandwidth utilization), $QoS$ (Quality of service) and $A$ (Availability or the time duration for which the node uses the channel)

$$H(m) = \{T_d,B_w,QoS,A\}$$  \hspace{1cm} (1)

The power metric is calculated as per [5] described below. It is given by $PLANE$ (Power Loss/Availability for Network Monitoring Estimate) a node-specific measure of the mean power loss per node for running the network-monitoring agent. $PLANE$ can directly be related to the wireless protocol used, mean number of wireless links for the specific node, average node maintenance energy consumption, and the battery power remaining. $PLANE$ ultimately estimates the duration the node can last on the same power without replenishment.

$$PLANE = \frac{BPR}{TEC}$$  \hspace{1cm} (2)

Where $BPR$ is the total battery remaining and $TEC$ is the total energy consumption before a node is selected as the network-monitoring node. The $PLANE$ value is calculated for each of the nodes and the resulting values are arranged in descending order. The advantage of this arrangement is that only the most suitable nodes are chosen to enable efficient monitoring of the network. A specific range is chosen from the list of values. The range has been calculated experimentally here as follows. Here $PLANE_{max}$ and $PLANE_{min}$ denote the upper and lower values of $PLANE$ respectively. The optimum range is then represented as given in equation (5).

$$\Delta d = \frac{PLANE_{max} - PLANE_{min}}{2}$$  \hspace{1cm} (3)

$$PLANE_{lower} = PLANE_{max} - \Delta d$$  \hspace{1cm} (4)

$$Range = \{PLANE_{max}, PLANE_{lower}\}$$  \hspace{1cm} (5)

Thus the input vector $I(v)$ for the ART is given by the following equation,

$$I(v) = \{PLANE_{max,PLANE_{lower}, H(m)}\}$$  \hspace{1cm} (6)

B. Adaptive Resonance Theory Training

The Adaptive Resonance Theory module compares the power of the nodes with that of the power range and selects those nodes that fall under the range. The first stage output of the ART is a set of nodes whose power metric value lies in the range. Now the second metric namely the $H(m)$ of the selected nodes is compared with the vigilance parameter to activate a single node. The node that is selected in this process is most optimized one. The activated node is then subjected to a training module. In this phase the node is trained to select nodes that are adjacent to the monitoring node. Adjacent in this case means nodes that are within single hop radius with respect to the monitoring node. The ART module is a recursive module and selects a few monitoring nodes in such a way that the covers all.

$$\Delta d = \frac{PLANE_{max} - PLANE_{min}}{2}$$  \hspace{1cm} (3)

$$PLANE_{lower} = PLANE_{max} - \Delta d$$  \hspace{1cm} (4)

$$Range = \{PLANE_{max, PLANE_{lower}}\}$$  \hspace{1cm} (5)

Thus the input vector $I(v)$ for the ART is given by the following equation,

$$I(v) = \{PLANE_{max,PLANE_{lower}, H(m)}\}$$  \hspace{1cm} (6)

B. Adaptive Resonance Theory Training

The Adaptive Resonance Theory module compares the power of the nodes with that of the power range and selects those nodes that fall under the range. The first stage output of the ART is a set of nodes whose power metric value lies in the range. Now the second metric namely the $H(m)$ of the selected nodes is compared with the vigilance parameter to activate a single node. The node that is selected in this process is most optimized one. The activated node is then subjected to a training module. In this phase the node is trained to select nodes that are adjacent to the monitoring node. Adjacent in this case means nodes that are within single hop radius with respect to the monitoring node. The ART module is a recursive module and selects a few monitoring nodes in such a way that the covers all.
C. Agent Architecture

In this phase we describe a hierarchical agent based system architecture given in [6] and adapted for SAPID. Agents are nothing but lightweight processes that are executed at various nodes. Not all nodes possess the same architecture. This agent architecture is for the monitoring nodes alone. The architecture is shown in Fig 2. The various stages for the agent based system are listed below.

![Agent Architecture for SAPID](image)

**Monitoring Agent.** The monitoring agent performs an activity of looking at the various session data present in the session information file of a UNIX based system. This module recognizes all the changes the log file pattern to detect intrusion detection [17]. Repetitive patterns of changes in the file are stored into a matrix. The session information file contains the following default fields.

<table>
<thead>
<tr>
<th>Field Number</th>
<th>Field Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>User Name</td>
</tr>
<tr>
<td>2</td>
<td>Connection Type</td>
</tr>
<tr>
<td>3</td>
<td>Dev</td>
</tr>
<tr>
<td>4</td>
<td>PID</td>
</tr>
<tr>
<td>5-10</td>
<td>Time Connection Mode</td>
</tr>
<tr>
<td>11-16</td>
<td>Time Logged In</td>
</tr>
<tr>
<td>17-22</td>
<td>Time Connection Terminated</td>
</tr>
<tr>
<td>23</td>
<td>Host</td>
</tr>
<tr>
<td>24</td>
<td>Class</td>
</tr>
</tbody>
</table>

**TABLE I**

**DEFAULT FIELDS OF SESSION INFORMATION FILE**

The next stage of monitoring involves in vectorization of the data matrix. Our discussion to this point has glossed over the issue of how incoming data streams are packaged for presentation to the SOM, which accepts fixed-length vectors as input [23]. The invention of schemes by which packets, transactions, and data might represent streams numerically represents both a challenge as well as an opportunity. The choice of which traffic features to represent, and how to translate them into numbers, will unavoidably involve highlighting certain aspects of network activity while making others obscure or even invisible to the intrusion detector.

It is to be noted that vectorization must be developed which increase as much as is possible, the contrast between innocent and malicious activity. The output of vectorization is a form suitable for representation in the Feature selection module that incorporates a Self-Organizing Map.

**Feature Selection.** Here pre-processing of the vectorized data is carried out for pattern recognition. The pattern recognition module identifies the changes in the log file. The changes highlight the intruder’s attack on the log files. The feature selection is performed using SOM. The SOM is a continuous learning module where the monitoring nodes build a SOM. The SOM keeps building up horizontally and the learning process continues simultaneously.

**Kohonen Model of Self-Organizing Map.** The Kohonen model [23] employs a two-dimensional neuron layer. This layer innervated with certain number of inputs (Axons), which carry the input signals. The signal inhibits the neurons of the layer via synaptic connections. In this paper we make the SOM to identify the pattern and then categorize the patterns based on their similarity. Thus the SOM provides a visual input to the user at the monitoring node that shows grouping of similar type of intrusion that is detected based on the patterns of attack on the data in the log file.

**Training for the Self-Organizing Map.** The SOM keeps learning the environment time and again to detect the change in the log file pattern. After a series of training runs the output obtained shows the best possible representation of the patterns discovered from the session information file. The training may be performed every time an agent is assigned to a node or it may refer to its previous pattern discovery output depending on whether there is a marked change in the structure of the system. In this way SAPID optimizes the process of intrusion detection. The next stage in the architecture is the Decision making phase.

**Decision Phase.** A decision is taken by the monitoring nodes based on the set of patterns obtained. Every node will decide the intrusion level on a host-level basis. Certain nodes will collect intrusion and make collective decisions about the network level intrusion.
Action Phase. Every node has an action module that is responsible for resolving various intrusion situations on a host such as killing a process and for expedite execution of the appropriate decisions taken from the decision phase.

IV. EXPERIMENTAL RESULTS

To evaluate the efficiency of SAPID a series of experiments were conducted both to test the accuracy of the algorithm in choosing the network monitoring nodes as well as its ability to perform the intrusion detection using the agent architecture on a series of data present at the chosen nodes. The first part of the evaluation process was inspired by the work of [5]. The final part was derived from [21] and adapted for SAPID.

A. Experimental Setup

Comparisons between single-hop and multiple-hop radius for allocating network-monitoring nodes provides a neat measure of the tradeoff involved vis-à-vis the number of nodes needed. The density of the network clearly plays a major role, since the more the number of adjacent nodes per node, the fewer the network monitors needed to verify their authenticity [5].

The systems for testing the approaches were typically UNIX based and consisted of “session” information collected over a period of time. Many attacks generate different patterns than normal requests. Since the features described above are designed to capture the behavior of the requests, the attacks, when examined using the agents, will have large deviations than the normal requests [6].

This fact is not considered for our study, as the aim was to test the overall performance of SAPID. As evident form the succeeding graphs the percentage of nodes selected as network-monitoring nodes varies with the changing density and hop radius of the network.

B. Experimental Results

![Fig.4. Performance in sparse wireless networks with low average number of adjacent nodes per node using SAPID.](image)

![Fig.5. Performance in Dense Wireless Networks with high average number of adjacent nodes per node using SAPID.](image)

Fig.4 and Fig.5 show the performance of SAPID in sparse and dense wireless networks with the minimum criteria for a dense network being the number of adjacent nodes greater than 7. We can infer from Fig.5 that the number and percentage of the nodes selected as network monitoring nodes decreases as the network grows in size in tune with the results of [5]. Also evident is the fact that the number of nodes selected as network monitoring nodes stabilizes to near constant levels even with increasing network sizes. We tested the performance of SAPID for a UNIX based system with the session information taken over a period of time as explained previously. The results of Table2 show the correct predictions of the data present in the system and predictions in case of two different types of attacks and also a combination of both the attacks that entail anomalies in data.

The columns in Table2 are the correct prediction of normal intensity of data, the incorrect prediction of normal intensity of data, the correct prediction of attack intensity, and the incorrect prediction of attack intensity respectively. Our training and testing in this area was limited however, because our dataset did not contain many instances of the same attack. In this respect we found that using a SOM created a uniform grouped input for detection [23] for dynamic inputs.

<table>
<thead>
<tr>
<th>Types of Attacks</th>
<th>Correct Predictions</th>
<th>False Negatives</th>
<th>Correct Attack Predictions</th>
<th>False Positives</th>
</tr>
</thead>
<tbody>
<tr>
<td>Union of Attacks</td>
<td>57%</td>
<td>3%</td>
<td>95%</td>
<td>5%</td>
</tr>
<tr>
<td>Attack A</td>
<td>55%</td>
<td>5%</td>
<td>92%</td>
<td>8%</td>
</tr>
<tr>
<td>Attack B</td>
<td>54%</td>
<td>6%</td>
<td>91%</td>
<td>9%</td>
</tr>
</tbody>
</table>

TABLE II
NORMAL AND ATTACK PREDICTIONS BASED ON VARIOUS TYPES OF ATTACKS
In this approach we have presented a self-organized agent based architecture for power aware intrusion detection (SAPID). It provides an efficient way of choosing the network monitoring nodes based on power considerations as well as other parameters such as availability of the system, channel utilization and Quality of Service by training an Adaptive Resonance Theory module to filter out the best suited nodes. Redundant nodes are prevented from being selected as it leads to increasing computation overhead. An agent based lightweight model is used to monitor the selected nodes and represent the patterns present in the data in a self-organizing map to recognize the anomalies in them. Advantages of using an agent based intrusion detection system are manifold with the most important ones being conserving bandwidth and providing a heuristic approach to intrusion detection.

V. CONCLUSION

In this approach we have presented a self-organized agent based architecture for power aware intrusion detection (SAPID). It provides an efficient way of choosing the network monitoring nodes based on power considerations as well as other parameters such as availability of the system, channel utilization and Quality of Service by training an Adaptive Resonance Theory module to filter out the best suited nodes. Redundant nodes are prevented from being selected as it leads to increasing computation overhead. An agent based lightweight model is used to monitor the selected nodes and represent the patterns present in the data in a self-organizing map to recognize the anomalies in them. Advantages of using an agent based intrusion detection system are manifold with the most important ones being conserving bandwidth and providing a heuristic approach to intrusion detection.

REFERENCES


<table>
<thead>
<tr>
<th>Running Time</th>
<th>Sequence Detection Rate</th>
<th>Detection Rate</th>
<th>False Alarm</th>
<th>False Alarm Rate</th>
<th>Intrusion Detection Rate</th>
<th>Intrusion False Alarm Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>10000</td>
<td>98%</td>
<td>2%</td>
<td>98%</td>
<td>1%</td>
<td>98%</td>
<td>1%</td>
</tr>
<tr>
<td>20000</td>
<td>95%</td>
<td>5%</td>
<td>98%</td>
<td>2%</td>
<td>98%</td>
<td>2%</td>
</tr>
<tr>
<td>30000</td>
<td>90%</td>
<td>8%</td>
<td>97%</td>
<td>3%</td>
<td>98%</td>
<td>3%</td>
</tr>
<tr>
<td>40000</td>
<td>95%</td>
<td>4%</td>
<td>95%</td>
<td>3%</td>
<td>98%</td>
<td>3%</td>
</tr>
<tr>
<td>50000</td>
<td>95%</td>
<td>4%</td>
<td>97%</td>
<td>2%</td>
<td>98%</td>
<td>2%</td>
</tr>
</tbody>
</table>

Table 3 shows the sequence detection rates and the intrusion detection rates for increasing durations of the test run in seconds. The above results show that although the system has converged with a smaller trace run using SAPID, the efficacy does not change with increasing durations and stabilizes at a particular value. Almost all intrusions are detected and the false alarm rates are also low and average around 3%. The detection results demonstrate that this approach can work well on different wireless networks with increasing sizes and durations of time.