An Efficient Technique of Intrusion Detection for Large Number of Malicious Nodes in MANET using a Tree Classifier

D Rajalakshmi 1, K Meena 2

1 Department of Computer Science and Engineering
Sri Sairam Institute of Technology, Chennai, India.
1, 2 Department of Computer Science and Engineering
VelTech Rangarajan Dr Sagunthala R&D Institute of Science and Technology
Chennai, India.

Abstract - A mobile ad hoc network (MANET) is generally defined as a network that has many free or autonomous nodes, often composed of mobile devices or other mobile pieces that can arrange themselves in various ways and operate without strict top-down network administration. Mobile Ad hoc Networks are particularly prone to malicious behavior. Lack of any centralized network management or certification authority makes these dynamically changing wireless structures extremely vulnerable to infiltration, eavesdropping, interference and so on. Intrusion detection systems (IDSs) are used in MANETs to observe actions to detect any intrusion in the susceptible network. In this paper we propose an efficient scheme for analyzing and detecting the malicious nodes using a hybrid classification technique. Using a Tree Classifier technique we detect the malicious node with high accuracy and less error as well as help to increase the performance of the system.

Keywords - MANET, Intrusion Detection, Data mining, Classification.

I. INTRODUCTION

Past few years, have observed a rapid growth in the field of mobile computing due to explosion of inexpensive, widely available wireless devices. Thus, it has opened vast opportunity for researchers to work on Ad Hoc Networks.

Fig. 1. A Mobile Ad hoc Network (MANET)

In a MANET, nodes within one another’s wireless transmission range can communicate directly; however, nodes outside one another’s range have to rely on some other nodes to relay messages [1]. Thus, a multi-hop scenario occurs, where several intermediate hosts relay the packets sent by the source host to make them reach the destination node.

MANET is one that comes together as needed, not necessarily with any support from the existing infrastructure or any other kind of fixed stations [2]. This statement can be formalized by defining an ad hoc network as an autonomous system of mobile hosts (MHs) (also serving as routers) connected by wireless links, the union of which forms a communication network modeled in the form of an arbitrary communication graph. This is in contrast to the well-known single hop cellular network model that supports the needs of wireless communication by installing base stations (BSs) as access points. In these cellular networks, communications between two mobile nodes completely rely on the wired backbone and the fixed (BSs). In a MANET, no such infrastructure exists and the network topology may dynamically change in an unpredictable manner since nodes are free to move.

As for the mode of operation, ad hoc networks are basically peer-to-peer multi-hop mobile wireless networks where information packets are transmitted in a “store-and-forward” manner from a source to an arbitrary destination,
via intermediate nodes as shown in Fig.1. As the MHs move, the resulting change in network topology must be made known to the other nodes so that outdated topology information can be either updated or removed. For example, MH2 in Fig.1 changes its point of attachment from MH3 to MH4, other nodes in the network should now use this new route to forward packets to MH2 [3].

The issue of symmetric and asymmetric links is one among the several tasks encountered in a MANET. Another essential issue is that different nodes often have different mobility patterns. Some MHs are highly mobile, while others are primarily stationary. It is difficult to predict a MH's movement and pattern of movement [3].

The dynamic nature of MANETs makes network open to attacks and unreliability. Routing is always the most significant part for any networks. Each node should not only work for itself, but should also be cooperative with other nodes. MANETs are vulnerable to various security attacks [4]. Hence, finding a secure and trustworthy end-to-end path in MANETs is a genuine challenge.

II. RELATED WORK

Mobile Ad hoc wireless networks have attracted a lot of attention over the last few years, because of the increasing demand for ubiquitous connectivity. The design of the mobile ad hoc networks seems to require novel approaches, since they have peculiar characteristics which differ substantially from those of fixed networks or cellular networks, for which well-established design techniques already exist [6].

Authentication is a common prevention-based approach used in MANETs to reduce intrusions. However, it cannot eliminate intrusions because there are always some weak points in the system. In MANETs, a malicious node can launch deny of service (DoS) or disrupt the routing mechanism by generating error routing messages. For these types of attacks, intrusion detection can serve as a second wall of defense and is of paramount importance in high security networks.

An IDS continuously or periodically monitors the current subject activities, compares them with stored normal profiles and/or attack signatures, and initiates proper responses [7]. Basically, IDSs can be categorized as network-based or host-based. Network-based IDSs are not suitable for MANETs since they need to monitor or collect data that go through the network hardware interface. Host-based IDSs, which rely on data generated by users or programs located on the hosts, are good candidates for MANETs [7].

Security Requirements of MANET:

1. Availability: Availability ensures that the desired network services are available whenever they are needed. Systems that ensure availability seek to combat denial of attacks (DoS) and energy starvation attacks. As all the devices in the network depend on each other to relay messages, DoS attacks are easy to perpetrate.
2. Authorization and key management: Due to little or no infrastructure, identifying users is not an easy task. There are problems with TTP Schemes and identity based mechanisms for key agreement. A password authenticated multi-party Diffie-Hellman key exchange seems to overcome many problems of the generic protocol.
3. Confidentiality and Integrity: Data confidentiality is a core security primitive for ad hoc networks. It ensures that the message cannot be understood by anyone other than the authorized personnel. Data integrity denotes the immaculateness of data sent from one node to another. It ensures that a message sent from node A to node B was not modified during transmission by a malicious node C. If a robust confidentiality mechanism is employed, ensuring data integrity may be as simple as adding one-way hash to encrypted messages. In addition to malicious attacks, integrity may be compromised because of radio interference, etc.
4. Non-Repudiation: Non-repudiation ensures that the origin of a message cannot deny having sent the message. It is useful for detection and isolation of compromised nodes. When a node A receives an erroneous message from a node B, non-repudiation allows A to accuse B using this message and to convince other nodes that B is compromised.

III. SYSTEM ARCHITECTURE

An intrusion detection system always has its core element - a node will act as a central node, each node they can enter the network and do the operations after that they can leave the network. During this process some malicious nodes are entering in this private network. These nodes are identified and monitored by this secure intrusion detection system. To performing all steps this information will be validated further decision-making process.
Monitoring System: In the world of intrusion detection, to focus on detecting attacks and clearly anomalous activity. Another important component of a complete intrusion detection solution is basic network monitoring. Network monitoring collects information or connections. This allows us to identify unauthorized services being used within a network. Using network monitoring systems establish a better overall security.

Data Collection: Data collection is the process of gathering and measuring information on targeted variables.

Data Pre-processing: To reduce data as much as possible without any information loss, and required specialized planning, training and testing. These issues derived from the system analysis are [8]:

i) To provide an optimal and efficient computing data for IDS.
ii) To filter false rates and improve detection rates.
iii) To discover attack patterns and display appropriate data types for administrators to make policies.

Based on security audit, the number of intrusion models, the particular intrusions are recognized. There are six components of intrusion detection:

1. Subjects - Initiators of activity on a target system-normally users.
2. Objects - Resources managed by the system-files, commands, devices, etc.
3. Audit Records - Generated by the target system in response to actions performed or attempted by subjects on objects-user login, command execution, file access, etc.
4. Profiles - Structures that characterize the behavior of subjects with respect to objects in terms of statistical metrics and models of observed activity. Profiles are automatically generated and initialized from templates.
5. Anomaly Records - Generated when abnormal behavior is detected.
6. Activity Rules - Actions taken when some condition is satisfied, which update profiles, detect abnormal behavior, relate anomalies to suspected intrusions, and produce reports.

IV. THE PROPOSED SYSTEM

An intrusion detection system (IDS) is a device or software application that monitors a network or systems for malicious activity or policy violations. Any malicious activity or violation is typically reported either to an administrator or collected centrally using a security information and event management (SIEM) system. While anomaly detection and reporting is the primary function, some intrusion detection systems are capable of taking actions when malicious activity or anomalous traffic is detected, including blocking traffic sent from suspicious IP addresses.
A. Decision Tree Classifier:

Decision tree builds classification or regression models in the form of a tree structure. It breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches. Leaf node represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data.

Decision tree is a model that is both predictive and descriptive. A decision tree is a tree that displays relationships found in the training data. The tree consists of zero or more internal nodes and one or more leaf nodes with each internal node being a decision node having two or more child nodes. The training process that generates the tree is called induction. It should be noted that there to be a balance between the number of training samples and the number of independent attributes. The quality of training data usually plays an important role in determining the quality of the decision tree. If there are a number of classes, then there should normally be sufficient training data available that belongs to each of the classes.

A1. Algorithm:

1. Let the set of training data be S. If some attributes are continuous-valued, they should be discretized. The attributes are categorized and transformed into A, B, C and D or more descriptive labels may be chosen. Once that is done, put all of S in a single tree node.
2. If all instances in S are in the same class, then stop.
3. Split the next node by selecting an attribute A from the independent attributes that best divides or splits the objects in the node into subsets and create a decision tree node.
4. Split the node according to the values of A.
5. Stop if either of the following conditions is met, otherwise continues with step 3:
   a) If this partition divides the data into subsets that belong to a single class and no other node needs splitting.
   b) If there are no remaining attributes on which the sample may be further divided.

A2. Entropy:

\[ E(S) = \sum_{i=1}^{c} -p_i \log_2 p_i \]

Information Gain:

\[ \text{Info}(D) = - \sum_{i=1}^{m} -p_i \log_2 (p_i) \]  

\[ \text{InfoA}(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \text{Info}(D_j) \]

\[ \text{Gain} (A) = \text{Info}(D) - \text{InfoA}(D) \]

B. Confusion Matrix:

A confusion matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. The confusion matrix shows the ways in which your classification model is confused when it...
makes predictions. It gives us insight not only into the errors being made by a classifier but more importantly the types of errors that are being made.

**TABLE 1. CONFUSION MATRIX FOR A BINARY CLASSIFIER.**

<table>
<thead>
<tr>
<th>Terms</th>
<th>Class 1 Predicted</th>
<th>Class 2 Predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class 1 Actual</td>
<td>TP</td>
<td>FN</td>
</tr>
<tr>
<td>Class 2 Actual</td>
<td>FP</td>
<td>TN</td>
</tr>
</tbody>
</table>

Here: Class 1 is Positive, Class 2 is Negative

Definition of the Terms:
- **a** Positive (P): Observation is positive (for example: is an apple).
- **b** Negative (N): Observation is not positive (for example: is not an apple).
- **c** True Positive (TP): Observation is positive, and is predicted to be positive.
- **d** False Negative (FN): Observation is positive, but is predicted negative.
- **e** True Negative (TN): Observation is negative, and is predicted to be negative.
- **f** False Positive (FP): Observation is negative, but is predicted positive.

Classification Rate/Accuracy: is given by the relation:

\[ \text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \]

However, there are problems with accuracy. It assumes equal costs for both kinds of errors. A 99% accuracy can be excellent, good, mediocre, poor or terrible depending upon the problem.

Recall: can be defined as the ratio of the total number of correctly classified positive examples divide to the total number of positive examples. High Recall indicates the class is correctly recognized (small number of FN).

Recall is given by the relation:

\[ \text{Recall} = \frac{TP}{TP + FN} \]

Precision: To get the value of precision we divide the total number of correctly classified positive examples by the total number of predicted positive examples. High Precision indicates an example labeled as positive is indeed positive (small number of FP). Precision is given by the relation:

\[ \text{Precision} = \frac{TP}{TP + FP} \]

High recall, low precision: This means that most of the positive examples are correctly recognized (low FN) but there are a lot of false positives.

Low recall, high precision: This shows that we miss a lot of positive examples (high FN) but those we predict as positive are indeed positive (low FP)

F-measure: Since we have two measures (Precision and Recall) it helps to have a measurement that represents both of them. We calculate an F-measure which uses Harmonic Mean in place of Arithmetic Mean as it punishes the extreme values more. The F-Measure will always be nearer to the smaller value of Precision or Recall.

\[ F\text{- Measure} = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}} \]

Let us consider an example now, in which we have infinite data elements of class B and a single element of class A and the model is predicting class A against all the instances in the test data.

Here, Precision is 0.0, Recall is 1.0. Now, if Arithmetic mean is 0.5 and Harmonic mean is 0.0, then when taking the arithmetic mean, it would be 50% correct. Despite being the worst possible outcome! While taking the harmonic mean, the F-measure is 0.

**TABLE 2. INTERPRETATION OF CONFUSION MATRIX**

<table>
<thead>
<tr>
<th>n=165</th>
<th>Predicted : No</th>
<th>Predicted : Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual: No</td>
<td>50</td>
<td>10</td>
</tr>
<tr>
<td>Actual: Yes</td>
<td>5</td>
<td>100</td>
</tr>
</tbody>
</table>

**TABLE 3. SIMPLIFIED CONFUSION MATRIX**

<table>
<thead>
<tr>
<th>n=165</th>
<th>Predicted : No</th>
<th>Predicted : Yes</th>
<th>Row Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actual: No</td>
<td>Tn = 50</td>
<td>FP = 10</td>
<td>60</td>
</tr>
<tr>
<td>Actual: Yes</td>
<td>Fn = 5</td>
<td>TP = 100</td>
<td>105</td>
</tr>
<tr>
<td>Column Total</td>
<td>55</td>
<td>110</td>
<td>-</td>
</tr>
</tbody>
</table>

Classification Rate/Accuracy:

\[ \text{Accuracy} = \frac{(TP + TN)}{(TP + TN + FP + FN)} = \frac{(100+50)}{(100+5+10+50)} = 0.90 \]

Recall: Recall indicates when it is actually yes, how often it predicts yes.

\[ \text{Recall} = \frac{TP}{(TP + FN)} = \frac{100}{(100+5)} = 0.95 \]

Precision: Precision indicates when it predicts yes, how often it is correct.

\[ \text{Precision} = \frac{TP}{(TP + FP)} = \frac{100}{(100+10)} = 0.91 \]

F-measure: F-measure = \((2\times\text{Recall}\times\text{Precision})/\text{(Recall + Precision)} = (2\times0.95\times0.91)/(0.91+0.95) = 0.92.\]

**V. EXPERIMENTS AND RESULTS**

Data visualization is a tool to convert "raw data" into meaningful images or graphics for effortless human comprehension or communication. Data visualization tools provide a single snapshot of complex relationships that are otherwise incomprehensible or difficult to interpret. The aim is to effectively convey or communicate the information content of (large volumes
of data as accurately as possible using graphical representations with minimal effort for human cognition. A side by side display of two data sets facilitates visual comparison, whereas a scatter plot of two correlated variables depict interdependency. Visualization display data in various forms, and allow a user to manipulate the data interactively.

In this section, we evaluate the performance of malicious node activities are monitored by the hybrid techniques using Tree classifier. Fig. 4 represents the root node divide the sub nodes and monitor the malicious node activity. Fig. 5 represents the margin curve Analysis in Real time and Fig. 6 represents the cost benefit analysis of malicious node behavior.
VI. CONCLUSION

In this paper, the unbalanced resource consumption of IDSs in MANET and the presence of malicious nodes have motivated us to propose an integrated solution for prolonging the lifetime of mobile nodes and for preventing the emergence of malicious nodes using a hybrid technique i.e. A Tree Classifier algorithm to measure the quality of a decision tree itself is an interesting problem. Classification Accuracy determined using test data is obviously a good measure but other measures may be used. These include average cost and worst case cost of classifying an object. A decision tree may be able to classify the training data 100% accurately but that does not imply that tree will be just as accurate on test data that was not part of the training set. It may be that the tree's performance on test data will be well below 100% if the training data was not a good sample of the data population.

REFERENCES

[15] Huanhuan Chen, Qiang Wang, and Yi Shen, “Decision tree support vector machine based on genetic algorithm for multi class
D RAJALAKSHMI et al: AN EFFICIENT TECHNIQUE OF INTRUSION DETECTION FOR LARGE NUMBER.


BIOGRAPHY

D. Rajalakshmi is a research scholar of Veltech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology, Chennai. She received the Undergraduate Degree in Information Technology from Anna University, in 2005, the Post Graduate degree in Computer Science and Engineering from Anna University, in 2010. Currently she is an Assistant Professor, Department of Computer Science and Engineering at Sri Sairam Institute of Technology, Chennai. She has more than 20 publications in National and International Conferences and international journal proceedings. Her areas of interest include Mobile Ad hoc Networks, Wireless Sensor Networks and Network Security.

K.Meena received her Ph.D Degree from the Computer Science and Engineering Department, Manonmaniam Sundaranar University, India in 2014. She received the B.E. (Electronics and Communication Engineering) and M.E. (Computer Science and Engineering) Degrees from Manonmaniam Sundaranar University in 2002 and 2009. She is currently Associate Professor in the Computer Science & Engineering Department, Veltech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Chennai. Her research interests include Machine Learning, Pattern Recognition, Image Processing, Biometrics and Cyber security. She has published 30 papers in International Journals, 7 in International Conferences and 11 in National Conferences.