CHAPTER 6

INTRUSION DETECTION SYSTEM FOR NETWORK AND HOST

6.1 INTRODUCTION

Intrusion Detection is an emergent trade used against illegal network activities. There is no absolute solution to detect attacks but combinations of different tools and methods do give a quite good result. To give better accuracy in terms of detection, the individual type of IDS is not sufficient. In order to achieve a higher detection rate and low false alarms, more than one type of IDS is required. Like, in this work, NIDS and HIDS are used to detect anomalies using supervised learning. The SVM with GNP classifier is used both in the NIDS and HIDS to classify the data into normal or abnormal.

6.2 SYSTEM ARCHITECTURE FOR NIDS AND HIDS

The proposed system architecture shown in Figure 6.1 has two major functions, namely, Incremental Support Vector Machine and Genetic Network Programming. The ISVM mainly does the sample selection according to the RBF- kernel function. The GNP rule creator creates the rule to classify the dataset into normal or attack pattern.

The KDD CUP 1999 Dataset for NIDS or Widows Registry for HIDS is taken as the training data and that data set is preprocessed by RBF
based kernel function used in ISVM. In case of NIDS, 9 features out of 41 attributes from KDD CUP 1999 Dataset have been extracted for further process. Fuzzification is applied on the training dataset to enhance the rule creation. Genetic Network Programming is applied to the fuzzified dataset and the trained Network Intrusion Detection Model has been produced. The training dataset is used to create the rules and generated signatures are stored for pattern matching. In the testing phase the test dataset has been pattern matched with the NIDS model and finally the data are classified into normal attack or abnormal attack.

![System Architecture for NIDS and HIDS](image)

**Figure 6.1 System Architecture for NIDS and HIDS**

### 6.3 IMPLEMENTATION DETAILS

In this work, support vector machine is used to preprocess the data using RBF-Kernel function. After preprocessing the data set is classified into the positive kernel and negative kernel. Thus the sample dataset is selected by the kernel function. Fuzzification is applied to the sample dataset and that divides the attributes into subattributes which are used for creating the rules.
easily. Then the genetic network programming is used to create the rules for the intrusion detection in the network. Genetic Network Programming is a step-by-step process which has processing node and judgement node. Processing nodes are used to pass or just process the rules and the judgement nodes are used to check the rules. If the first rule is satisfied it will then go to next processing node or another judgement node until the rule is created.

### 6.3.1 RBF Kernel Function

The RBF Kernel function is used in the support vector machine. For each subset of the dataset the kernel function is calculated with the help of this kernel function, the dataset is classified into the positive kernel and negative kernel as shown in Figure 6.2. Positive kernel is only used for the further process where negative kernel is not of interest. Normally the kernel function uses the formula given in (Yi et al 2011).

\[
k(u, v) = \exp\left(-\frac{(u - v)^2}{2\sigma^2}\right)
\]  

(6.1)

where u, v are the attributes from the dataset.

In this proposed work, the RBF kernel function has been derived using the equation (6.1) as

\[
k(u, v) = \exp\left(-\frac{\|u - m\|/ s - (v - m)/ s\|^2}{2\sigma^2}\right)
\]

(6.2)

The equation (6.2) is mathematically proved according to Kuhn-tucker condition.
\begin{align*}
k(u,v) &= \exp\left(-\frac{(u - m)/s}{2\sigma^2}\right) \exp\left(-\frac{(v - m)/s}{2\sigma^2}\right) \\
&\times \exp\left(-\frac{(u - m)/s \cdot (v - m)/s}{\sigma^2}\right) \\
F(u) &= \exp\left(-\frac{(u - m)/s}{2\sigma^2}\right) h(u,v) \quad (6.3)
\end{align*}

where \( m \) is the mean and \( s \) is the mean square deviation.

Normally, the radial function is calculated by finding out the mean and medians of any two attributes. The mean or medians are calculated by Euclidian distance. The proposed system has taken source bytes and destination bytes as \( x \) and \( y \) and count as \( s \). The source bytes and destination bytes are used to calculate the mean value that is used to calculate the radial basis function.

The mean value is calculated from the relation,

\[ M_j = \frac{1}{n} \sum_{i=1}^{n} L_{ij}, (j = 1,2,\ldots,n) \quad (6.5) \]

where \( m = (m_1, m_2, \ldots m_j, \ldots, m_k) \) and \( s = (s_1, s_2, \ldots s_j, \ldots, s_k) \) are the mean value and the mean square deviation of attributes, \( k \) is the dimension of sample vectors.

The mean square deviation is calculated by the formula,

\[ S_j = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (L_{ij} - m_j)^2} \quad (6.6) \]
These kernel values are stored into another table with the required attributes. These attributes are used to create the rules for the intrusions. Not all attributes are used to create the rules.

A support vector machine is a concept in statistics and computer science for a set of related supervised learning methods that analyze data and recognize patterns, used for classification and regression analysis. The sample set is selected according to the SVM concept.

### 6.3.2 GNP based Rule Creation

The rule creation is based on genetic network programming. Genetic Network programming is the method shown in Figure 6.3 to create the rules by passing the attributes to the processing node and judgement nodes. Judgement nodes are the nodes to compare the attributes with already created rules. Processing nodes are the nodes to just pass the attribute from one node to another node.

There are two key points that have been taken into consideration, one is how to select samples to construct the reserved set, and the other is how to assign weights to each sample. The decision-function of ISVM is determined by the support vectors and then determines which samples are most likely to be support vectors. The algorithm for Reserved Set-Incremental Support Vector machine is given below.

**Algorithm: Reserved Set-Incremental Support Vector Machine**

<table>
<thead>
<tr>
<th>Input</th>
<th>KDD CUP 1999 Dataset</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output</td>
<td>Sample Dataset</td>
</tr>
</tbody>
</table>

Algorithm:

1. Begin
2. For each dataset
3. Do
   a. Find mean M;
   b. Find mean square deviation S;
   c. Find kernel function k(u,v);
   d. If(k(u,v)>0)
      i. Set as positive kernel;
      ii. Store the dataset;
      iii. Fuzzify the dataset;
   e. Else
      i. Set as negative kernel;
      ii. Reject the dataset;
4. Return sample dataset;
5. End

The algorithm for Rule creation with Genetic network programming is given below.

**Algorithm: Genetic Network Programming**

**Input** : Sample Dataset  
**Output** : Generated Rules  
**Algorithm:**

1. Begin
2. For each fuzzified dataset
3. If the attribute is judgement node
   a. Send attributes to processing node;
   b. Check the conditions;
   c. Generate the rules;
4. Else
   a. Send attributes to next judgement node;
5. Repeat the process until rules are generated;
6. End
The KDD CUP 1999 Dataset is too large for the first part of the experiment, so we selected a subset of KDD, about 10% of it, as the input of the proposed system. To avoid the imbalance of the dataset, the sample set are preprocessed to regenerate new training and test set. The preprocessed dataset is produced by the ISVM method. The training time increases according to the size of the dataset increases.

In the training dataset 9 attributes are taken instead of 41 attributes which reduce the noise as well as the training time. The selected attributes are server count, count, source bytes, destination bytes, diff_host_srv_rate, dst_host_port_rate, service, flag, proto_type. In the testing phase, the dataset is compared with the training data model and produce the result according to the type of the attack. The attacks are classified based on the rules generated using the combination of attributes like service, flag, source bytes and destination bytes.

6.4 PERFORMANCE EVALUATION

This experiment has been done through the 10% of the KDD CUP 1999 Dataset. The dataset is preprocessed by incremental SVM using RBF kernel function and rules are generated using genetic network programming. The proposed system detects misuse activities with significant improvement in terms of high detection rate and low false positive rate. This can be compared with traditional SVM method and results are shown in Figures 6.2 and 6.3.
From the results shown in the graphs, it can be observed that the proposed RS-ISVM with GNP provided better detection rate and reduced false positive rate when it is compared with the RS-SVM algorithm.