**Available Online at www.ijcsmc.com**

**International Journal of Computer Science and Mobile Computing**

**A Monthly Journal of Computer Science and Information Technology**

**ISSN 2320–088X**

***IJCSMC, Vol. 3, Issue. 7, July 2014, pg.506 – 511***

**RESEARCH ARTICLE**

# Implemented IDS Version of Advanced Artificial Neural Networks (ANN) using Layered Method

**Mr. Prashant D. Somwanshi[1], Dr. S. M. Chaware[2]**
[1]M.E. (CSE), [2]HOD, Department of Computer Engineering
TSSM'S Bhivarabai Sawant College of Engineering and Research, Narhe, Pune, India
[1] prashant.somwanshi@gmail.com
[2] smchaware@gmail.com

## Abstract

***Security is a big issue for all networks in today's enterprise environment. Hackers and intruders have made many successful attempts to bring down high-profile company networks and web services. Many methods have been developed to secure the network infrastructure and communication over the internet; among them are the use of firewalls, encryption, and virtual private networks. Intrusion detection is a relatively new addition to such techniques. Using intrusion detection method we can find the intrusion signature in a network and must perform efficiently to manage with the large amount of network traffic. Existing system are label these two issues of correctness and efficiency using Conditional Random Fields with Layered Approach. We show that high attack detection correctness can be achieved by using Conditional Random Fields and high efficiency by implementing the Layered Approach. In this review paper, we are modifying existing system and newly use modified ANN algorithm to improve the efficiency and correctness to detect intrusion than previous one.***

***Keywords: IDS, Layered Approach, ANN's method, Host Based***

## 1. Introduction

Now a day, intrusion detection is one of the high priority and demanding tasks for network administrators and security expert. There is a need to protect the networks from known intrusion (attacks) and also to find new and unseen attacks by developing more reliable and efficient intrusion detection systems. First purpose of intrusion detection system is to detect intruders; that is, unexpected, unwanted or unauthorized people or programs in a network.

Since 1980's after the dominant research from Anderson [1] intrusion detection system was began. Intrusion detection systems are divided into network based, host based, or application based categorized. [2]. Also, intrusion detection systems can also be classified as signature based or anomaly based depending upon the method of attack detection. The signature-based systems to detect intrusion from previously known attacks while the anomaly-based systems learn from the normal data collected when there is no inconsistent activity [2].

After the introduction in Section 1, existing work with highlighted on various methods used for intrusion detection are described in Section 2. We describe implemented method layered approach with ANN method in Section 3. And in section 4, mention system architecture of proposed system. In Section 5, describes results and discussion and section 6 describes conclusion and future work. Finally, section 7 mention references.

## 2. Existing System

The domain of intrusion detection and network security has been around since late 1980s. Since then, a number of methods and frameworks have been proposed and many systems have been built to detect intrusions. In this section, we briefly discuss existing techniques and frameworks. Naive Bayes classifier is one of the intrusion detection methods. It provides lower attack detection accuracy when the selection features are dependable [3]. Decision trees have also been used for intrusion detection [3]. The decision trees select the best features for each decision node during the construction of the tree based on some well-defined criteria [5]. In 2010[4], Kapil Kumar Gupta, Baikunth Nath and Ramamohanaroa Kotagiri presented "A frame work using a layered approach for intrusion detection". They have addressed two main issues of ID i.e. accuracy and efficiency by using conditional random fields and layered method [6]. They have shown that layered Conditional Random Fields (CRFs) have very high attack detection rate than previous one [7].

## 2.1 Limitations of Existing system

- Existing system is based on off-line system which was less efficient and accurate.
- Static off-line database.
- Layered approach of CRF is not support pipelining of layers in multi-core processors.
- Using Naïve Bayes when selection features are dependable then attack detection accuracy is lower.
- Using decision tree algorithms additional research is needed in data mining to improve, automate, and simplify decision tree method for use in industry.

## 2.2 Comparative Analysis between Existing system and Proposed system

Following table shows the comparison between existing approach and proposed approach:

**Table 1.1:** Comparative Analysis

| Sr No. | Existing Approach Using CRF Algorithm | Proposed Approach Using ANN Algorithm |
|---|---|---|
| 1 | Layered Approach Used | Layered Approach Used |
| 2 | Based on Static Database (KDD '99) | Dynamic Database Used( i.e. Runtime file generated by object serialization method) |
| 3 | Existing Attacks Categories: DOS, U2R, R2L, Probe, Normal etc. | Modified Attacks Categories:<br>TCP, UDP, ICMP, Normal etc. |
| 4 | Addresses two issues: Accuracy and Efficiency. | Resolved these issues. |
| 5 | Consider some features (i.e. duration, flag, protocol_type etc.) of each attacks types. | Consider all features (i.e. duration, flag, protocol_type etc.) of each attacks types. |
| 6 | To implement layered approach using CRF algorithm. | To implement layered approach using ANN algorithm. |

## 3. Implemented Method

Due to the above limitations we have to develop modified or advanced artificial neural network (ANN) algorithm to improve intrusion detection system better than previous. Neural network algorithms are merging now a day as a new artificial intelligence technique that can be applied to real life problems. Neural networks are a form of artificial intelligence that uses multiple artificial neurons, networked work together to process information. This type of network has the capability to learn from structure, and extend results from data that has been previously entered into the network's knowledge base. This ability makes neural network applications really valuable in intrusion detection [8].

## 3.1 Algorithm of ANN

Algorithm of ANN is divided into two parts:

### 3.1.1 Training

**Step 1**: Select the number of **n** layers.

**Step 2**: Select all features for each layer.

**Step 3**: Train a separate model with ANN for each layer using the all features selected from Step 2.

**Step 4**: Manage trained models sequentially such that only the connections labeled as normal are passed to the next layer.

### 3.1.2 Testing

**Step 5**: For each (next) test instance perform Steps 6 through 9.

**Step 6**: Test the instance and label it either as attack (intrusion) or normal.

**Step 7**: If the instance is labeled as attack, block it and identify it as an attack represented by the layer name at which it is detected and go to Step 5. Else pass the sequence to the next layer.

**Step 8**: If the current layer is not the last layer in the system, test the instance and go to Step 7. Else go to Step 9.

**Step 9**: Test the instance and label it either as normal or as an attack (intrusion). If the instance is labeled as an attack, block it and identify it as an attack corresponding to the layer name.

## 3.2 Mathematical model for proposed system

Let's consider the general mathematic definition of a single neuron is as follows:

$$\mathbf{y(x) = g(\textstyle\sum_{i=0}^{n} wi\ xi)} \text{ .................. } \mathbf{1.1}$$

- **x** is a neuron with n input dendrites **($x^0$…$x^n$)** and one output axon **y(x)** and where **($w^0$…$w^n$)** are weights determining how much the inputs should be weighted.
- g is an activation function that weights how powerful the output (if any) should be from the neuron, based on the sum of the input.
- If the artificial neuron should mimic a real neuron, the activation function g should be a simple threshold function returning 0 or 1.
- The output from the activation function is either between 0 and 1, or between -1 and 1, depending on which activation function is used.
- The inputs and the weights are not restricted in the same way and can in principle be between -∞ and +∞, but they are very often small values centred on zero.

According to multilayer neuron network:

- Two different kinds of parameters can be adjusted during the training of an ANN, the weights and the **t** value in the activation functions. This is impossible and it would be easier if only one of the parameters should be adjusted.
- To manage with this problem a bias neuron is invented. The bias neuron lies in one layer, is connected to all the neurons in the next layer, but none in the previous layer and it always emits 1.
- Since the bias neuron emits 1 the weights, connected to the bias neuron, are added directly to the combined sum of the other weights (by equation 1.1), just like the **t** value in the activation functions.
- A modified equation for the neuron, where the weight for the bias neuron is represented as $\mathbf{w_{n+1}}$, is shown in equation 1.2.

$$\mathbf{y(x) = g(w_{n+1}\textstyle\sum_{i=0}^{n} wi\ xi)} \text{....................} \mathbf{1.2}$$

- Adding the bias neuron allows us to remove the t value from the activation function, only leaving the weights to be adjusted, when the ANN is being trained.
- We cannot remove the **t** value without adding a bias neuron, since this would result in a zero output from the sum function if all inputs where zero, regardless of the values of the weights.

### 3.2.1 Activation Function

The activation usually uses one of the following functions:

#### 3.2.1.1 Sigmoid Function

The sigmoid is also very useful in multi-layer networks, as the sigmoid curve allows for differentiation (which is required in Back Propagation training of multi layer networks). Or if you're into math type explanations:

$$f(x) = \frac{\mathbf{1}}{\mathbf{1} + e^{-\beta x}}$$

#### 3.2.1.2 Step Function

A basic on/off type function, if 0 > x then 0, else if x >= 0 then 1 Or a math-type explanation is:

$$f(x) = \begin{cases} 0 \text{ if } 0 > x \\ 1 \text{ if } x \geq 0 \end{cases}$$

## 4. System Architecture of Proposed System

In proposed system architecture, Main Host systems consider as Main server and that system provides internet facility to other Sub-Host system by using proxy setting. In other words, Internet facility not directly accessible to sub-host system; it goes from Main Server or Host. So our IDS project will be run on the Main Host systems that system provides internet facility to other system. Also runtime dynamic database will be created to stored intrusion patterns.

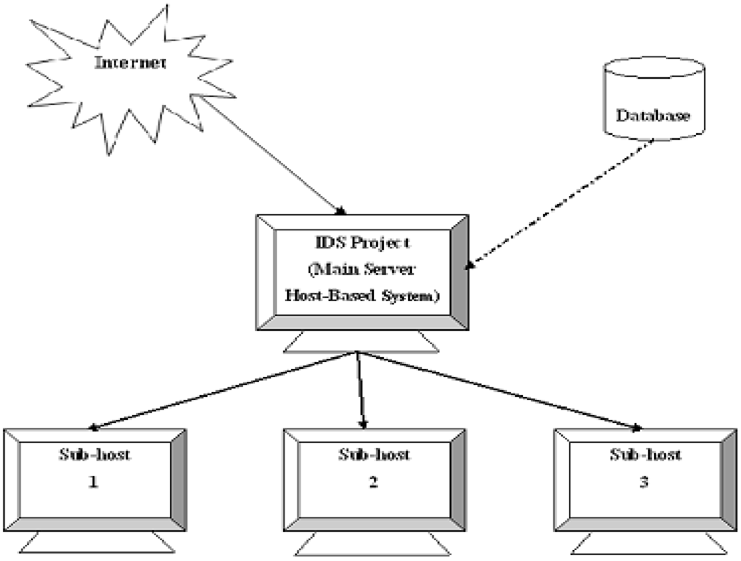

Fig 1.1: System Architecture of the Proposed System

### 4.1 Advantages of proposed system

- To detect all types of attacks (intrusion).
- More secure, reliable, accurate and efficient than previous one.
- Scope is unlimited.
- Host based online real time IDS advanced layered ANN's method.
- Dynamic real time database.
- Execution time is less.

## 5. Results and Discussion

### 5.1 Simulation Setup

Proposed IDS system run on several different computers in a networks running by attacking module setup. By attacking module assign IP address to Host Based System (i.e. actual IDS system run) and attacking module run on different machine and send thousands of samples attacks to check the intrusion detection accuracy of Host system. It gave the above 99% accuracy to detect the intrusion and time duration needed to detect intrusion is in microsecond (i.e. we cannot calculate it) because Host Based IDS System is Real Time Online System i.e. system detect the intrusion online or real time. In other words, Real Time Online System means IDS system run on Internet to detect the intrusion on online fashion.

The aim of this test was to evaluate situation-specific performance in networks. We developed custom simulator software to run simulations. In our simulation, system performance in the network is shown.

### 5.2 Host system performance in the network

Result tables shows data regarding the no of packets present in network and total time required to detect malicious packets.

**Table 1.2:** Packets comparison table

| Sr no. | No. of packets | Time/ Duration (seconds) |
|---|---|---|
| 1 | 10000 | 0.99 |
| 2 | 7000 | 0.80 |
| 3 | 5000 | 0.70 |
| 4 | 2000 | 0.30 |

As the no. of packets is increasing, the intrusion detection duration also increases.

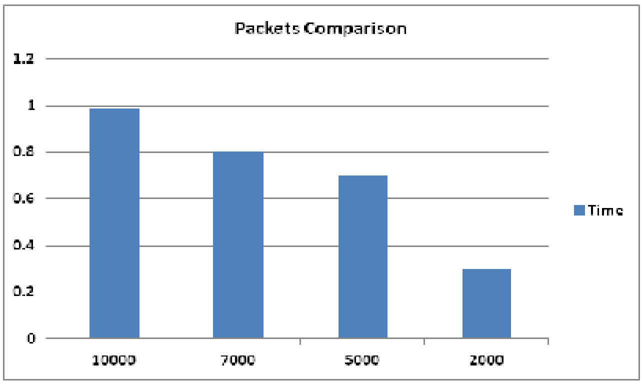

Fig 1.2: Comparison of No. of Packets verses Time

Total no. of packets present in a network verses total time required to detect intrusion packets as shown in above graph.

### 5.3 Packet detected ratio in the network

According to layer (i.e. Normal, TCP, UDP, ICMP) the packets detection ratio is following way:

**Table 1.3:** Packets Detected Ratio

| Sr no. | No. of packets detected in each layer | Layer Name |
|---|---|---|
| 1 | 10000 | Normal |
| 2 | 7000 | TCP |
| 3 | 5000 | UDP |
| 4 | 2000 | ICMP |

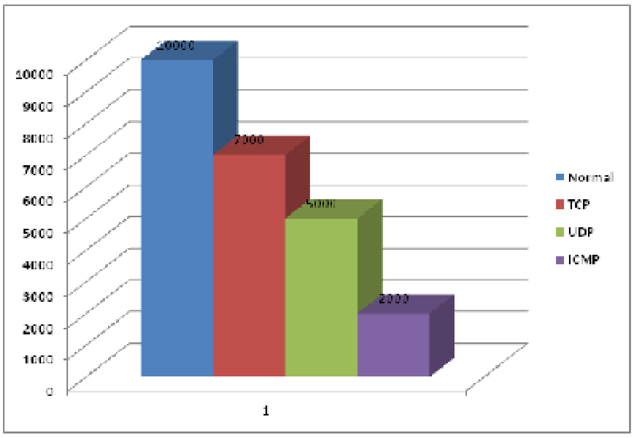

Fig 1.3: Packets detection ratio in the network

Above graph shows the total no. of packets detected in layer.

## 6. Conclusions and Future Work

Implemented IDS system can help in identifying an attack once it is detected at a particular layer, which expedites the intrusion response mechanism, thus minimizing the impact of an attack. Also, IDS system has the advantage that the number of layers can be increased or decreased depending upon the environment in which the system is deployed, giving flexibility to the network administrators.

This paper shows implemented the new advanced artificial neural network (ANN) method to detect all types of attacks (intrusion) by host based online real time system. It is more accurate and efficient than previous one. The performance result shows that our proposed work overcomes the existing work with variations of differences.

## References

[1] J.P. Anderson, Computer Security Threat Monitoring and Surveillance, http://csrc.nist.gov/publications/histo ry/ande80.pdf, 2010.

[2] R. Bace and P. Mell, Intrusion Detection Systems, Computer Security Division, Information Technology Laboratory, Nat'l Inst. of Standards and Technology, 2001.

[3] N.B. Amor, S. Benferhat, and Z. Elouedi, “Naïve Bayes vs. Decision Trees in Intrusion Detection Systems,” Proc. ACM Symp. Applied Computing (SAC ’04), pp. 420-424, 2004.

[4] Kapil Kumar Gupta, Baikunth Nath and Ramamohanaroo kotagiri, “A layered approach using conditional random fields for intrusion detection”, IEEE Tranc. on Dependence and secure computing, Vol.7, 2010.

[5] Jeff Markey. Using Decision Tree Analysis for Intrusion Detection: A How-to Guide. SANS Institute.

[6] K.K. Gupta, B. Nath, and R. Kotagiri, “Network Security Framework,” Int’l J. Computer Science and Network Security, vol. 6, no. 7B, pp. 151-157, 2006.

[7] K.K. Gupta, B. Nath, and R. Kotagiri, “Conditional Random Fields for Intrusion Detection,” Proc. 21st Int’l Conf. Advanced Information Networking and Applications Workshops (AINAW ’07), pp. 203-208, 2007

[8] Herv´e Debar, Monique Becke, Didier Siboni, “A Neural Network Component for an Intrusion detection system,” Proceedings of the IEEE Symposium on Research in Security and Privacy, pp. 240–250, 1992.

## Authors

**Prashant D. Somwanshi** received B. E. degree in Computer Engineering from North Maharashtra University, India in 2008. He is currently pursuing M. E. degree in Computer Engineering from University of Pune. He is working in teaching fields since last 2 years and he has industrial experience of 3 years.

**Dr. S. M. Chaware** received Ph.D with specialization in Information and Network Security, Mobile and Wireless Network. He is working in teaching fields since last 18 years and HOD at TSSM’s BSCOER, Pune under University of Pune, India.