2012 IEEE 14th International Conference on High Performance Computing and Communications

Security Enhancement in Wireless Sensor Networks using Machine Learning

Aswathy B. Raj[1], Maneesha V. Ramesh[2], Raghavendra V. Kulkarni[3] and Hemalatha T.[4]
Amrita Center for Wireless Networks and Applications, Amrita Vishwa Vidyapeetham
Kerala, India
aswathy.braj13@gmail.com[1], maneesha@am.amrita.edu[2], arvie@ieee.org[3], hemalatha@am.amrita.edu[4]

***Abstract*—Ensuring the security of wireless sensor networks (WSNs) is vital for monitoring real-time systems. One of the major security flaws experienced by WSNs is denial of service (DoS) which can even lead to the breakdown of the complete system or to wrong decisions being made by the system that can cause adverse results. This research work focuses on two techniques for detecting a DoS attack at a medium access control (MAC) layer. Our research compares and evaluates the performance of two major machine learning techniques: neural network (NN) and support vector machine (SVM). Vanderbilt Prowler is used for simulating the scenarios. In the simulations, normalized critical parameters and their corresponding probabilities of DoS attack are computed in 50 trial runs. These normalized critical parameters and their corresponding probabilities of DoS attack are used as training inputs in NN and SVM approaches. The simulation results clearly show that SVM provides better accuracy compared to NN, 97% accuracy by SVM and 91% accuracy by NN. The simulation also shows that SVM takes much less time to detect and determine the probability of a DoS attack, 0.25 seconds by SVM and 0.75 seconds by NN. All these results clearly show that SVM performs better than NN when used for detecting the probability of DoS attack in WSNs.**

Keywords- Wireless Sensor Networks; Security; Denial of Service; Neural Network; Support Vector Machine

I. INTRODUCTION

A sensor is an object used to gather information about a physical object or the occurrence of events. Together, many sensors can be used to collect data and communicate wirelessly to a processing station. A Wireless Sensor Network (WSN) is formed when these sensors are deployed cooperatively to monitor large physical environments. Major constraints for WSN include: security, energy (where sensor nodes are powered through either batteries or solar power), memory, computational capability and communication bandwidth. When addressing the security concerns for WSNs there are a variety of unique challenges [1]. Security enhancement techniques have computational, communication and storage requirements which further constrain sensor nodes. Moreover, it is impractical to have a central point of control in sensor networks because of their resource constraints and network dynamics. Therefore, the development of a decentralized security solution is integral for WSN optimum efficiency. Many WSNs are left unattended since they operate in remote and hard-to-reach locations. So it is difficult to continuously monitor and prevent sensor nodes from attacks. WSNs are employed in a variety of applications such as disaster relief operations, biodiversity mapping, machine surveillance, precision agriculture, military, machine and healthcare. WSNs are vulnerable to a variety of attacks. Ensuring the security of WSNs in such applications is vital.

This paper compares two machine learning techniques namely: Neural Network (NN) and Support Vector Machine (SVM), for detecting and counteracting to the DoS attacks launched by adversaries, thereby enhancing the security of WSNs. NN and SVM, enable nodes to monitor (for the key parameters of an attack), and to stop working if attacks are detected.

The rest of the paper is organized as follows: A survey of DoS attacks and their countermeasures are described in section II. The details of security enhancement using NN and SVM are described in section III. The simulation results showing the security enhancement of WSN using the two approaches and their performance analysis are presented in section IV. Finally conclusion and future work are contained in section V.

II. RELATED WORK

Different types of attacks in WSN are identified and categorized by H. K. Kalita and A. Kar [2]. Adversaries can launch DoS attacks that disrupt the services of WSNs. DoS attacks can occur at any layer of a typical WSN [3] [4]. The prevention of DoS attacks is considered a major issue in security of ad-hoc sensor networks [5]. Although, cryptographic authentication mechanisms were found to be effective at combating DoS attacks [4], they cannot be used by WSNs because of resource limitations. However, DoS attacks can be overcome by identifying misbehaving nodes [5]. A MAC protocol based on fuzzy logic system [6] and a hybrid intelligent intrusion detection system [7] can be used to identify misbehaving nodes, and thus for detecting DoS attacks. While the MAC protocol based on fuzzy logic system [6] only uses the fuzzy interference approach for taking a decision, the hybrid intelligent intrusion detection system [7] additionally uses the NN based approach to learn the attack definitions. But this hybrid system [7] has partly solved the problem to recognize attacks.

978-0-7695-4749-7/12 $26.00 © 2012 IEEE
DOI 10.1109/HPCC.2012.186

Marti Hearst [8] pointed out that the use of complex algorithms like neural networks in real-world applications are harder to analyze theoretically but SVM can learn more precisely and it is simple enough to be analyzed mathematically.

Our research compares the performance of NN and SVM for detecting and counteracting to the DoS attacks launched by adversaries on the MAC layer of WSN. NN is trained using backpropagation (BP) algorithm. The attack definitions are learned by the sensor nodes and hence on detecting attacks they will stop working till the adversary moves away from their vicinity. Applying the machine learning techniques NN and SVM to improve WSN security creates a distributive WSN security mechanism that, in face of an attack, requires only the victim sensor node to temporarily shut itself down, irrespective of its neighbor nodes. The victim node then subsequently reactivates when the attack is over.

III. DETECTION OF DOS ATTACKS

A DoS attack is an attempt by an adversary to degrade the network's services. In a DoS attack, malicious nodes can degrade the services provided by legitimate nodes by flooding them with requests. The MAC layer of the WSN is based on Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol. This CSMA/CA protocol relies on the exchange of ready-to-send (RTS) and clear-to-send (CTS) control packets. When a source node has data to send, it initiates the process by sending an RTS packet. The RTS packet silences any nodes that hear it. If a target node receives an RTS it responds with a CTS. Like the RTS packet, the CTS packet silences the nodes in the immediate vicinity. Once the RTS/CTS exchange is complete, the source node transmits data without worry of interference from any other nodes. The data packets are positively acknowledged. Different types of DoS attacks occur at various layers of protocol stack. Three types of DoS attacks that exist at MAC layer include: the collision attack, the unfairness attack and the exhaustion attack [9].

- Collision Attack: Before sending RTS/CTS packets, all nodes sense the channel for determining whether that channel is busy or idle. The sensor nodes will carry out data transmission only if the channel is idle– in order to prevent a collision occurring while sending data packets. Under this condition, adversaries can conduct attacks by flooding the sensor network with packets thereby causing collision.
- Unfairness Attack: All nodes have the same priority to access the same channel. The channel is assigned to nodes on the basis of a first come first serve (FCFS) policy that means the first tried node is given access to the channel first. Under this condition, adversaries transmit large numbers of packets without waiting or waiting for a short time. This prevents legitimate nodes from using the common channel.
- Exhaustion Attack: When a sensor node receives an RTS control packet, the node acknowledges an RTS control packet with a CTS control packet. Since the attackers are normal nodes, the legitimate nodes are not able to distinguish whether the RTS packet is sent by normal nodes or attackers. Under this condition, the adversaries transmit large number of RTS packets to normal nodes, which in turn acknowledge them with CTS packets. This results in the exhaustion of battery life at receivers.

The work in [10] uses the following critical parameters for detecting the probability of attack:

- R_c (Collision Rate): R_c is the number of collisions detected by a node in a second.
- R_r (RTS arrival rate): R_r is the number of RTS packets received successfully by a node in a second.
- T_w (Average waiting time): T_w is the waiting time of a packet in MAC buffer before transmission.

The above critical parameters are periodically monitored for different probability of attack ranging from 0.1 to 1. It is observed that the value of T_w is negligible on comparing with the values of R_r and R_c. Hence the critical parameters R_r and R_c are used for detecting the probability of DoS attack.

In the neural network (NN) based approach, the inputs represent the parameters R_c and R_r and the corresponding probability of attack is represented as the targets to the multilayer perceptron (MLP). The MLP is trained by using backpropagation algorithm. At each node, MLP is implemented with predefined weights and biases which are obtained from trained MLP. Every minute, each node passes its computed values of R_c and R_r to its MLP which produces an output (that is the calculated probability of attack at that particular node). If the MLP's output (that is the calculated probability of attack at that particular node) is greater than a preset threshold value S_{TH}, then the node temporarily shuts itself down, and subsequently reactivates when the attack is over.

In the SVM based approach, the probability of an attack is divided into two classes, namely Low and High. The SVM classifier is trained using the critical parameters R_c and R_r taken from these two classes. Every minute, each node passes its critical parameters such as R_c and R_r to the trained SVM classifier to classify the probability of attack as either Low or High. The node shuts itself down if it detects the High probability of attack and subsequently reactivates when the attack is over.

A. NN Based Approach

1) Architecture: MLP is the type of neural network used. MLP is a feed forward NN in which neurons are arranged in many layers. The structure of MLP used is shown in Figure 1. It has one hidden layer. X_1 and X_2 are the input units. The output unit Y_1 and hidden unit Z_1 have biases. The bias on output unit Y_1 is denoted by W_{01}. The bias on hidden unit Z_1 is denoted by V_{01}. The activation function applied to hidden layer is hyperbolic tangent sigmoid function. The activation function applied to output layer is linear function.

2) Training Algorithm: MLP is trained using BP algorithm [11]. It involves three stages:

- Feedforward of the input pattern
- Calculation and backpropagation of associated error
- Adjustment of weights

During feedforward, each input unit $X_i (i = 1, 2)$ receives input signal x_i and broadcasts it to the hidden unit Z_1. Z_1 aggregates its weighted input signals as expressed in (3).

$$z_{in} = v_{01} + \sum_{i=1}^{2} x_i v_{i1} \tag{3}$$

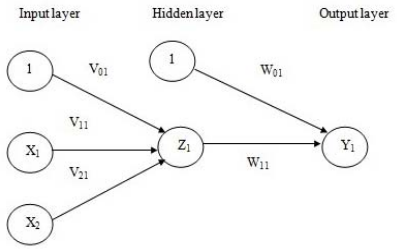

Figure 1. Structure of MLP

Z_1 applies its activation function to compute its output signal as expressed in (4) and send this signal to output unit Y_1.

$$z_1 = f(z_{in}) \tag{4}$$

The output unit Y_1 sums its weighted input signals as expressed in (5).

$$y_{in1} = w_{01} + z_1 w_{11} \tag{5}$$

Y_1 applies its activation function to compute its output signal as expressed in (6).

$$y_1 = f(y_{in}) \tag{6}$$

During training, the output unit Y_1 compares its activation y_1 with its target value t_1 to determine the associated error. Based on this error, the factor δ_1 is computed. δ_1 is used to propagate the error at output unit Y_1 back to hidden unit Z_1. It is used to update the weights between the output and the hidden layer. Similarly weights between hidden layer and the input layer are updated. Many epochs are required for training a NN by BP. The mathematical basis for the BP algorithm is gradient descent.

B. SVM Based Approach

SVM is a statistical learning method used for various purposes like classification, de-noising, pattern recognition etc [12]. In this paper we use SVM based classification technique for identifying the probability of attack. The learning machine is given a set of input data for training. The training data is a binary labeled data indicating lower and higher probability of attack. The learning machine tries to find a maximally separating hyper plane and two bounding planes, which separates the two classes '*L*' or '-1' (Lower probability of attack) and '*H*' or '+1'(Higher probability of attack).

Consider the training set of pairs (x_i, d_i), $i = 1, ..., n$ where x is n-dimensional input vector and d is m-dimensional target vector in which $d_i \in \{-1, +1\}$ indicates the class to which x_i belongs. The maximally separating hyper plane is represented as $w^T x - \gamma = 0$ and the bounding hyper planes as $w^T x - \gamma = \pm 1$, where w is n-dimensional coefficient vector, and 'γ' is a bias term which is scalar. The input vector belonging to '+1' or '*H*' class satisfies the constraint $w^T x - \gamma \geq 1$ and the vector belonging to '-1' or '*L*' satisfies the constraint $w^T x - \gamma \leq -1$. However in our scenario, we except few errors, so there is a chance that some of the input vector will be deviated from their respective bounding plane. A positive quantity called slack variable, ξ is added or subtracted to the input vector to satisfy the constraints. Thus the new constraints are written as,

$$\begin{aligned} w^T x - \gamma + \xi \geq 1 \\ w^T x - \gamma - \xi \leq -1 \end{aligned} \tag{7}$$

SVM aims to try for maximum margin between the bounding planes and minimum number of input vectors contributing to error [13]. Maximum margin is achieved by minimizing $\frac{1}{2} w^T w$ and minimum error is achieved by minimizing $\sum_{i=1}^{m} \xi_i$. The primal form of formulation is given by,

$$\min_{w,\xi,\gamma} \frac{1}{2} w^T w + C \sum_{i=1}^{m} \xi_i \tag{8}$$

subject to constraints,

$$\begin{aligned} d_i(w^T x_i - \gamma) + \xi_i - 1 \geq 0, 1 \leq i \leq m \\ \xi \geq 0, 1 \leq i \leq m \end{aligned} \tag{9}$$

where '*C*' known as penalty parameter controls the weightage for maximum margin and sum of error. The value of '*C*' gives good generalization power for the classifier. The primal form given in (8) is converted to dual form and then solved using quadratic programming. The solutions are got in terms of Lagrangian multipliers. From these Lagrangian multipliers, primal variables ξ, γ and w are computed. For good result, the input space is mapped to a higher dimensional space $\varphi(xi)$and then hyper plane is maximally separated in that space. The dual form of formulation is given by,

$$\min_u L_D(u) = \frac{1}{2} u^T Q\, u - e^T u \tag{10}$$

subject to constraints

$$0 \leq u \leq Ce \tag{11}$$

where $D = diag(d)$, $Q = DKD$, K is the kernel matrix and $e^T\xi$ is the sum of non negative errors.

In this paper, we have used Gaussian radial basis function (GRBF) kernel given by,

$$K(x_i x_j) = exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right) \tag{12}$$

'*C*' and 'σ' values of GRBF are fined tuned to achieve maximum accuracy. The decision function for the testing data is given by,

$$f(x) = sign(\sum_{i=1}^{m} d_i u_i K(x_i, x) - \gamma) \tag{13}$$

Performance of SVM is evaluated and it is described in section IV.

I. EXPERIMENTATION AND RESULTS

A. Numerical Results

The WSN scenario is simulated using the probabilistic wireless network simulator (Prowler) [14]. A WSN scenario for observing the critical parameters is shown in Figure 2. It involves 25 sensor nodes having unique IDs from 1 through to 25. The nodes exchange data through an RTS/CTS control scheme. Each node attempts to transmit a packet every 0.25 seconds with a probability '*P*'. At the receiver, collision happens when two nodes transmit simultaneously. The number of RTS packets a node receives, in one minute, is measured as the request rate, R_r. The average number of collisions, in one minute, is measured as the collision rate, R_c. For each different probability of DoS attack, (ranging from .1 to 1), the values of these R_r and R_c parameters are observed and averaged over 50 trial runs. Shown in Table I, these R_r and R_c values are then normalized. The probability of attack is the measure of a suspicion of an attack. From Table I, it is observed that R_c and R_r increase with the increase in the probability of attack. Figure 3 is a graph showing the normalized values of these parameters. Normalized values of these R_c and R_r parameters and their corresponding probabilities are used as training inputs in NN based approach and SVM based approach.

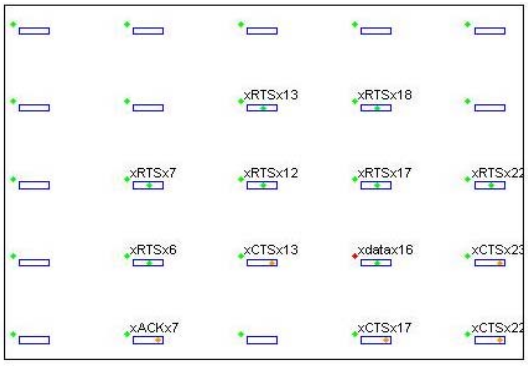

Figure 2. WSN scenario used for computing critical parameters

TABLE I. CRITICAL PARAMETERS AVERAGED OVER 50 TRIAL RUNS

Probability of Attack	R_r	R_c
0.1	393.9	110.2857
0.2	506.2	124.44
0.3	535.74	135.12
0.4	652.4	163.34
0.5	717.4	177.4
0.6	921.74	220.02
0.7	991.74	239.96
0.8	1056.48	260.34
0.9	1131.34	280.88
1	1190	301.66

B. Security against DoS attack using NN based approach

The normalized values of critical parameters, R_c and R_r plotted in Figure 3 are given as inputs and their corresponding probability of attack are given as targets to the MLP. The MLP is trained by BP algorithm to get the desired output value (which is in fact the probability of attack) that matches the targets. Figure 4 shows the values of inputs (normalized R_c and R_r) plotted against target values (t). Figure 5 shows the values of inputs plotted against the outputs (y) obtained from training the MLP by BP. From this trained MLP, best trainable parameters are obtained and a MLP is created. Each node checks the probability of attack using its own MLP every minute. If a node detects any attack, then it shuts down. A node that has temporarily shut itself down continues to calculate its probability of attack every minute and once there is no longer a threat of attack it will resume its normal function. The WSN scenario used for detecting DoS attack is illustrated in Figure 6.

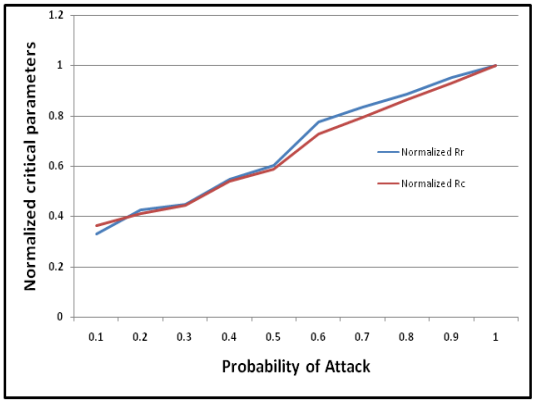

Figure 3. Normalized critical parameters averaged over 50 trial runs

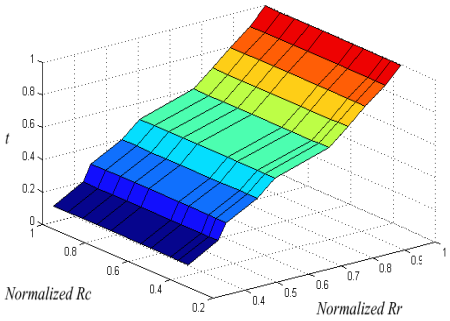

Figure 4. Normalized critical parameters versus target values (t) used for training the MLP

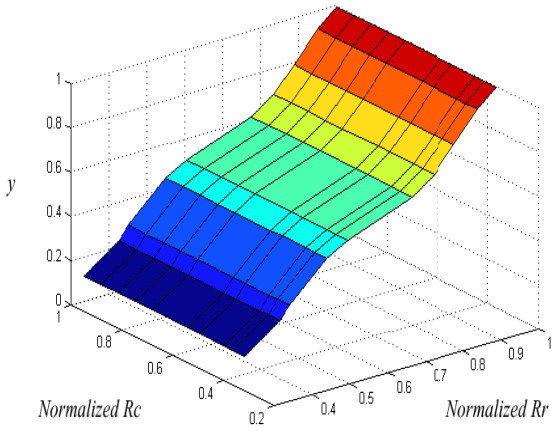

Figure 5. Normalized critical parameters versus output values (y) obtained from trained MLP

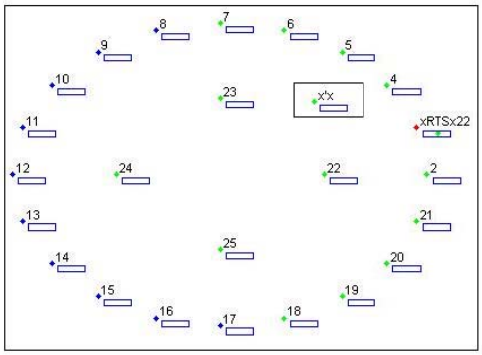

Figure 6. WSN scenario used for DoS attack detection

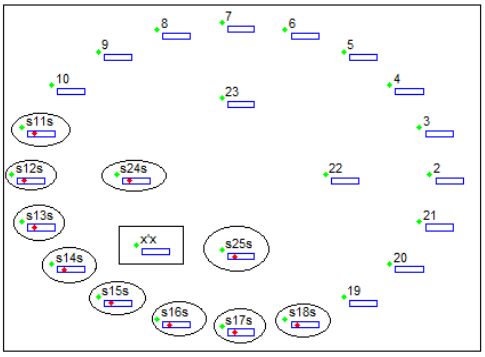

Figure 7. Simulation showing stopping of nodes on attack detection using NN

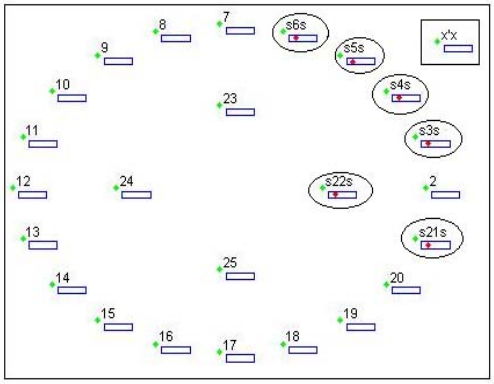

Figure 8. Simulation showing stopping of nodes on attack detection using SVM

Figure 6 shows a scenario where node 1 is marked 'xx' is the adversary. The adversary is encircled in a rectangle. The adversary can be moved to any place in the scenario. In this scenario, node 1 (xx) induces a constant probability attack. The normal nodes transmit packets through the CSMA/CA mechanism. Every minute, each node's MLP senses the parameters R_c and R_r and produces the output that can shut down and subsequently reactivate itself once the threat has gone. In Figure 7, it can be seen that all nodes which have detected attack have stopped working. These nodes are shown with red LEDs ON. They are marked in circles. When these attacked nodes do not detect any attack in the next minute, then they become active and begin their normal data transmission.

C. Security against DoS attack using SVM

The normalized critical parameters, in Figure 3, are grouped into two classes of probability of attack namely Low and High. These R_c and R_r values are given to SVM for training. Every minute, each node gives its values of critical parameters, R_c and R_r to the trained SVM to classify these parameters R_c and R_r into one of two classes. If these parameters fall in High probability of attack class, then the node stops working. In Figure 8, it can be seen that all nodes that have detected attack have temporarily stopped working. These temporarily shut down nodes are shown with red LEDs

ON and are marked in circles. When these attacked nodes do not detect any attack in the next minute, then they become active and carry out data transmission.

D. Security against DoS attack in bursty traffic

The NN based approach is used for detecting an attack in bursty traffic. In previous works, traffic conditions are static that means the probability of an attack is constant. In bursty traffic, the probability of attack varies i.e. the probability of attack varies with change in time. In Figure 9, it can be seen that all nodes which have detected attack have temporarily stopped working. These temporarily shut down nodes are shown with red LEDs ON and are marked in circles. When these attacked nodes do not detect any attack in the next minute, then they become active and carry out data transmission.

E. Performance Analysis

Table II gives the comparison between NN and SVM based approach for DoS attack detection, based on their accuracy and time taken for determining the probability of DoS attack. SVM achieves 97% accuracy whereas NN achieves 91% accuracy. From the critical parameters computed at the sensor node, while NN takes 0.75 seconds for determining the probability of attack, SVM takes only 0.25 seconds in determining the probability of attack.

V. CONCLUSION AND FUTURE WORK

Two machine learning techniques, SVM and NN are used for DoS attack detection. They are based on supervised learning. NN is a distributed parallel system capable of solving problems that a linear program cannot solve. SVM training finds a global minimum for kernel based methods. From the performance analysis between the two techniques, it is observed that SVM achieves 97% accuracy and NN achieves 91% accuracy in detecting DoS attack. When NN takes 0.75 seconds, SVM takes only 0.25 seconds for determining the probability of attack in WSNs. These results demonstrate that SVM is distinctly faster than NN whist still producing a higher accuracy.

In future, this work will be extended by using unsupervised machine learning techniques or different neural architectures such as generalized neuron or radial basis functions for detecting DoS attacks. The security mechanism against DoS attack will be implemented for a network of MICAz sensor nodes.

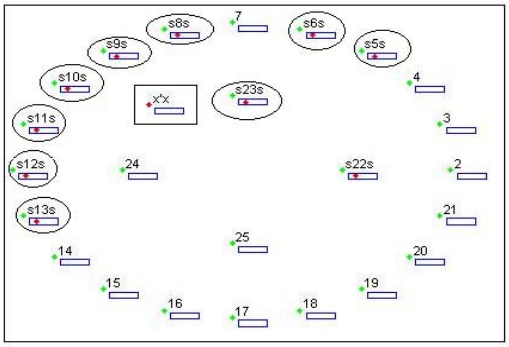

Figure 9. Simulation showing stopping of nodes on attack detection in bursty traffic

TABLE II. PERFORMANCE ANALYSIS OF NN AND SVM

Performance Metric	*NN*	*SVM*
Accuracy(in percentage)	91.43	97.14
Time Taken(in seconds)	0.75	0.25

ACKNOWLEDGMENT

We would like to express our immense gratitude to our beloved Chancellor Shri. (Dr.) Mata Amritanandamayi Devi for providing motivation and inspiration for doing this research work.

REFERENCES

[1] W. Dargie and C. Poellabauer, "Fundamentals of Wireless Sensor Networks", Wiley, 2010.

[2] H. K. Kalita and A. Kar, "Wireless sensor network security analysis," in International Journal of Next-Generation Networks, vol. 1, no. 1, pp. 01-09, Decenber 2009.

[3] A. D. Wood and J. A. Stankovic, "Denial of service in sensor networks," IEE JNL, vol. 35, no. 10, pp. 54 – 62, October 2002.

[4] D. R. Raymond and S. F. Midkiff, "Denial-of-service in wireless sensor networks: Attacks and defenses," IEEE Pervasive Computing, vol. 7, no. 1, pp. 74 – 81, March 2008.

[5] F. Hu and N. K. Sharma, "Security considerations in ad hoc sensor networks," in Ad Hoc Networks, pp. 69 – 89, September 2003.

[6] Q. Ren and Q. Liang, "Fuzzy logic-optimized secure media access control protocol for wireless sensor networks," in IEEE IntemationaI Conference on Computational Intelligence for Homeland Security and Personal Safety, 2005, pp. 37 – 43, April 2005.

[7] D.-R. Tsai, W.-P. Tai, and C.-F. Chang, "A hybrid intelligent intrusion detection system to recognize novel attacks," in IEEE 13th Annual International Carnahan Conference on Security Technology, pp. 428 – 434, October 2003.

[8] Sutharshan Rajasegarar, Christopher Leckie, James C. Bezdek and Marimuthu Palaniswami, "Centered Hyperspherical and Hyperellipsoidal One-Class Support Vector Machines for Anomaly Detection in Sensor Networks", IEEE Transactions On Information Forensics And Security, vol. 5, no. 3, pp. 518 – 533, September 2010.

[9] Q. Ren and Q. Liang, "Secure media access control in wireless sensor networks: intrusion detections and countermeasures," in 15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, vol. 4, pp. 3025 – 3029, September 2004.

[10] R. V. Kulkarni and G. K. Venayagamoorthy, "Neural network based secure media access control protocol for wireless sensor networks," in International Joint Conference on Neural Networks, pp. 1680 – 1687, June 2009.

[11] L. Fausett, "Fundamentals of Neural Networks", Prentice-Hall.

[12] K. Soman, R. Loganathan, and V. Ajay, "Support Vector Machines and Other Kernel Methods".

[13] Hemalatha.T, S. Mithila.T, and Soman.K.P, "Comparative study of linear and quadratic programming versions of svm on various real life data sets," in International Journal of Recent Trends in Engineering, vol. 1, no. 2, pp. 23 – 25, May 2009.

[14] G. Simon, P. Vlgyesi, M. Marti, and kos Ldeczi, "Simulation-based optimization of communication protocols for large-scale wireless sensor networks,"in IEEE Aerospace Conference, vol. 3, pp. 3_1339 - 3_1346 March 2003.