Volume 8, No. 3, March – April 2017
ISSN No. 0976-5697
International Journal of Advanced Research in Computer Science
RESEARCH PAPER
Available Online at www.ijarcs.info

Intrusion Detection in Networks Security: A New Proposed Min-Min Algorithm

Parveen Sadotra (CEH)
Research Scholar,Department of computer Application
Career Point University,Kota, Rajasthan, India

Dr. ChandrakantSharma
Assistant Professor,Department of computer Application
Career Point University,Kota, Rajasthan, India

***Abstract*:** We propose a method to detect intrusions in computer network security. The main idea is to reuse the already available system information that is generated at various layers of a network stack. To the best of our knowledge, this is the first such approach for intrusion detection in computer network security. Wireless Sensor Network consists of large number of nodes, which will be in distributed nature. Security is a very important consideration while designing a Wireless Sensor Network. So, an Advanced Intrusion Detection System has been proposed where the Heterogeneous Hybrid Intrusion Detection System (H-HIDS), Intrusion Detection Prediction based are implemented in various stages in order to assure maximum possible security from the Intrusions.

Keywords:intrusion detection, network security, anomaly, Min-Min algorithm, Neural network**.**

I. INTRODUCTION

Nowadays Network security is an emerging topic [1, 2]. The network security is generally deployed in a critical and hostile environment where the human labor is not implicated. Some of the trendy applications of WSN are fire response, traffic monitoring, military command etc. [1, 2, 3, and 4]. Different types of network topologies such as star, tree, mesh etc. are used for communication in WSN. In a cluster based hierarchical approach, concentration of sensor nodes forms a cluster and one node among them acts as a Cluster Head (CH). The CH assumes to have a larger battery and acts as a supervisor node for communication between other nodes. All CH in the network are connected to a Base Station (BS), which is a single decision making authority. [5][8] One of the cluster topology is depicted in figure 1

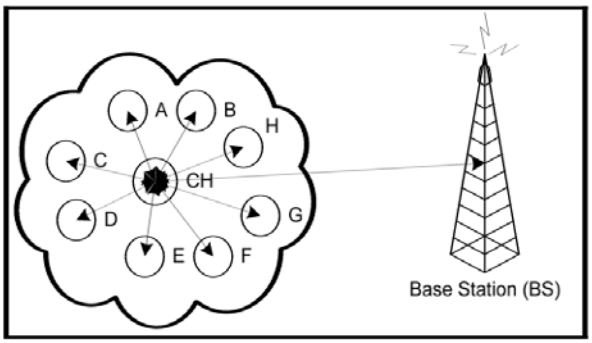


Fig. 1.1 Cluster in a sensor network

II. BACKGROUND OF THE RESEARCH

Intrusion detection system (IDS) provides two primary benefits: Visibility and Control [1]. The combination of these two benefits makes it possible to create and enforce anenterprise security policy to make the private computer network secure. Visibility is the ability to see and understand the nature of the traffic on the network while Control is the ability to affect network traffic including access to the network or parts thereof. Visibility is paramount to decision making and makes it possible to create a security policy based on quantifiable, real world data. Control is key to enforcement and makes it possible to enforce compliance with security policy.

The idea of detecting the intrusions or system misuses by looking at some kind malicious patterns in the network or user activity was initially conceived by James Anderson in his report titled *"Computer Security Threat Monitoring and Surveillance"* [2] to US Air Force in the year 1980.

In the year 1984, the first prototype of Intrusion Detection System, which monitors the user activities, named "Intrusion Detection Expert System" (IDES), was developed. In the year 1988, "Haystack" became the first IDS to use patterns and statistical analysis for detecting malicious activities, but it lacked the capabilities of real time analysis. Meanwhile, there were other significant advances occurring at University of California Davis' Lawrence Livermore Laboratories. In the year 1989, they built IDS called "Network System Monitor" (NSM) for analyzing the network traffic. This project was subsequently developed into IDS named "Distributed Intrusion Detection System" (DIDS). "Stalker" based on DIDS became the first commercially available IDS and influenced the growth and trends of future IDS. In the Mid 90's, SAIC developed "Computer Misuse Detection System" (CMDS), a host based IDS. US Air Force's Cryptographic support center developed "Automated Security Incident Measurement" (ASIM), which addressed the issues like scalability and portability.

III. METHODOLOGY

The IDS have been implemented in organizations to collect and analyze various types of attacks within a host system or a network. In addition, to identify and detect possible threats violations, which involve both intrusions, which are the attacks from outside the organizations and misuses that are known as the attacks within the organizations. In this paper, we proposed the integrated model which involves a combination of the two systems Intrusion Detection (ID) and Intrusion Prevention (IP) adding to those getting benefits from well-known techniques: intruder Detection (ID) which is totally different from most of the recent works

that focused only on using one system, either detection or prevention and also using either Intruder detection or Signature based detection. Some works even used a hybrid method which is a combination of both such as the work presented in [7] where the researchers used ID based on Signature but even then, their method was not provided with prevention capabilities. On the other hand, in our case, we proposed to use our approach IDPS, which not only can detect the attack but also can further stop it using the capabilities of prevention system, whichhas not been utilized in the previous works. Therefore, the proposed system can outperform the hybrid system of [7] [8]in terms of preventing the attack from conducting any bad action through blocking the event and saving that threat with the other signatures in order to be observed by Signature Based Intrusion Detection for next time so that it can be detected earlier. Finally, deploying such integrated model in the Wireless environment will reduce the probability of risks than the normal system or even than other systems, whichare just provided with Intrusion Detection methods.

IV. PRESENT APPROACH

We show that these protocols can accommodate much fewer competing nodes within a region in a network infested with hidden terminals are validated through MATLAB simulations. We introduce a framework to address the fairness problem inherent in wireless networks using IEEE 802.11 and propose a Min-Min algorithm in wireless network. Itcan be used to track discourse and to predict smooth discourse trajectories with applications to problem solving, which would be transmission of a data packet and its acknowledgment is preceded by sending and receiving packets between a pair by using Mac layer process for target tracking after that sending and receiving nodes conversational analytics, to realize the framework which improve throughput and failure probability as compare to existing scheme by using MATLAB simulation.

In our proposed work, we defined different scenarios over existing protocol are as successful transmissionin this method. It proposed a new back-off procedure in which the channel allocation method is different. In this method, the channel is allocated as per new back-off means whenever anomaly is occur at channel the contention method allocated channel as per Fibonacci based mathematical function. The Min-Min algorithm behaves as a mathematical function in which contention window size increment in a predefined manner.

EVALUATION OF PHAD AND SNORT

PHAD and Snort are evaluated on the 1999 DARPA off-line IDSevaluation data set [4]. The set consists of network traffictcpdump files. The week 3, 4 and 5 data is taken as input set.

A. COMPONENTS OF SNORT

Snort is logically divided into multiple components. These components work togetherto detect particular attacks and to generate output in a required format from the detectionsystem. A Snort-based IDS consists of the following major components:

- Packet Decoder
- Preprocessors
- Detection Engine
- Logging and Alerting System
- Output Modules

Snort is open source IDS which is available free of any costs can get it from http://www.snort.org. It is based on the rules which the call it snort rules which is regularly updated.

The below images shown the components of the SNORT

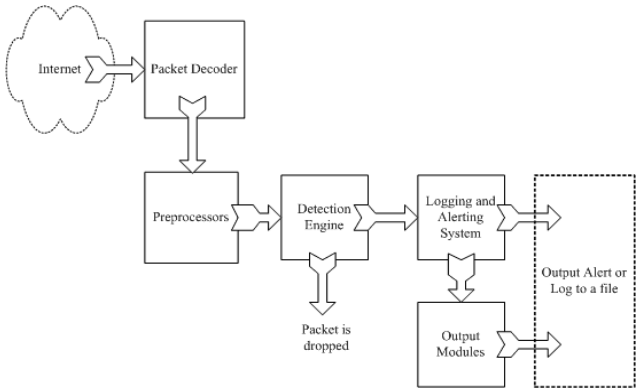


Fig.1.2snort components

Packet decoder
It takes the packets from different interfaces and sends it to preprocess or to the detection engine, these interfaces can be Ethernet, serial line internet protocol, etc.

Preprocessors
It is very important for IDSs, by applying the preprocessing you can easily analyze the packets, find the abnormal packets, and it generate some alerts so it is playing important role for intrusion detection systems.

Detection engine
The most important part of the IDSs is this part, the detection engine has the responsibility of detecting any stranger activities which might be exist on the packets, it applies the snort rules or this detection .so if any of these packets has matched any rules then immediate action will be taken else it will drop that packet.so, it means packet will be logged or generating an alert. The stronger IDS have strong detection engine. Whenare, you building,an IDS keep these on mind- Set of the rules as much as it can detect all of the possible attack so it means should be updated.

- The machine which you installed the snort should be have some features.
- Also, the internal buses should be very fast
- Also, the network load.

Logs and alerts
After the detection engine checking the packets so it might log the activity or also alert and log this alert into tcp dump file or txt file or whatever form, you can easily manage the logs by changing the location or whatever action you want to do.

B. Packet Header Anomaly Detection (PHAD)

We developed an anomaly detection algorithm (PHAD) that learns the normal ranges of values for eachpacket header field at the data link (Ethernet), network (IP), and transport/control layers (TCP, UDP, ICMP). PHAD does not currently examine application layer protocols like DNS, HTTP or SMTP, so itwould not detect attacks on servers, although it might detect attempts to hide them from an applicationlayer monitor like *snort* by manipulating the TCP/IP protocols.

An important shortcoming of all anomaly detection systems is that they cannot discern intent; they canonly detect when an event is unusual, which may or may not indicate an attack. Thus, a system shouldhave a means of ranking alarms by how unusual or unexpected they were, with the assumption that therarer the event, the more likely it is to be hostile. If this assumption holds, the user can adjust thethreshold to tradeoff between a high detection rate or a low false alarm rate. PHAD is based on theassumption that events that occur with probability *p* should receive a score of 1/*p*Packet header anomaly detector (PHAD) is the first anomaly based approach added to Snort as apreprocessor in this study. PHAD is different from other network-based anomaly detection systems by tworeasons. Firstly, it models protocols rather than the user behavior because the majority of the attacks exploitprotocol implementation bugs and can only be understood by detecting unusual input and output. Secondly, ituses a time-based model, assuming a quick change in a short time in the network statistics. PHAD reduces falsealarm rate by flagging only the first anomaly as an alarm.

C. Network traffic anomaly detector (NETAD)

Network traffic anomaly detector (NETAD) is second anomaly-based approach added to Snort as a preprocessorin this study. The NETAD also models packets as PHAD. NETAD operates in two phases: First is thefiltering of incoming client sessions to distinguish beginning of sessions. Second is the modeling phase. Filteringphase eliminates the traffic up to 98–99%. Elimination simplifies the traffic for the modeling phase. Thus, onlythe traffic data which evidence of attacks are included in is passed to the modeling phase.

Combining PHAD and NETAD to signature-based IDS Snort: -

Snort's preprocessor architecture has been used to combine PHAD and NETAD with Snort. Snort hasdetected 27 attacks out of 201 attacks available in IDEVAL data. Snort + PHAD have detected 12 attacks and **Snort + PHAD + NETAD** detected 18 attacks in existing work reference [12]. It is clear NETAD is added as a pre-processor Snort becomes amore powerful IDS.

Design Architecture

The proposed research design architecture can be divided into three phases of development namely, data collection and pre-processing; Known and unknown attack detection; and Prevention as shown in Fig.

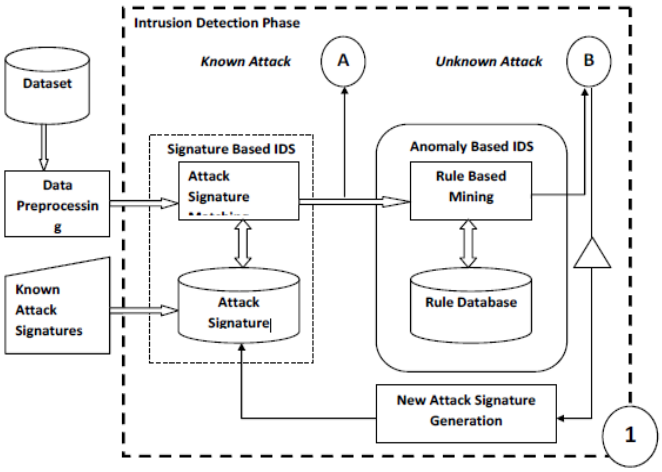
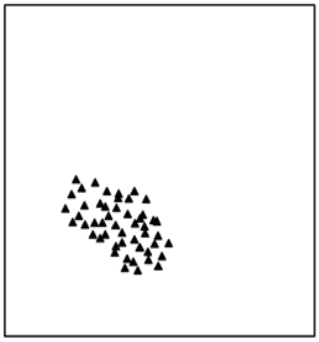


Fig1.3.: Intrusion Detection Phase of System Architecture

V. PROPOSED NOVEL ALGORITHM MIN-MIN NN (MIN_MIN NEURAL NETWORK)

we present a possible application ofneural networks as a component of an intrusion detection system. Neural network algorithms are emerging nowadays as a new artificial intelligence techniquethatcan be applied to real problems. We present an approach of user behavior modeling that takes advantage of theproperties of neural algorithms and display results obtained on preliminary testing of our approach.

The IDS-NN algorithm consists of two main phases – the specific training set construction and the neural network training process. The trained neural network is applied in the network communication system to detect intrusion attempts. During the supervised training process, the neural network has to be confronted with instances of both normal and intrusion classes. However, in case of an anomaly IDS, future intrusion data vectors are unknown at the time of training. It is only assumed that they will be different from the pattern of the recorded normal behavior. Hence in the first phase of the IDS-NN algorithm, the intrusion instances are randomly created in the attribute space.Since the real intrusion vectors are unknown ahead, they will be uniformly generated within the whole attribute space.



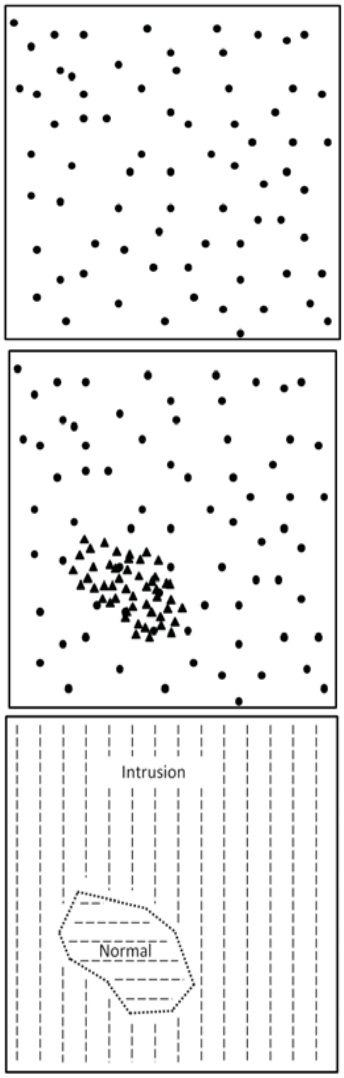


Fig. 1.4. Illustrative example of the training set construction and the cluster boundary modeled by the neural network. Recorded instances of normal behavior (a) and simulated intrusion instances (b) are combined together into a training set (c). The neural network models the classification function and the cluster boundary (dotted line) during the supervised learning process (d).

This newly generated intrusion vector dataset is combined with the recorded normal behavior. Fig. 3.3 (a) – 3.3(c) illustrates the construction of the training dataset.

In the second phase of the IDS-NN algorithm, aFeed-forward neural network is trained using a specific combination of the Error Back-Propagation and the Levenberg-Marquardt algorithm [9] [10] [11]. An example of athree-layer feed-forward neural network. The output of the input layer is directly determined by the input vector *p:*

$$\vec{a}^0 = \vec{p} \quad (1)$$

The net input of neuron *I* in layer *k+1*is calculated as:

$$n^{k+1}(i) = \sum_{j=1}^{Sk} w^{k+1}(i,j)a^k(j) + b^{k+1}(i) \quad (2)$$

Here *Sk*denotes the number of neurons in layer *k* , $w^{k+1}(i, j)$is the weight of the connection from neuron *j* in layer *k*, b^{k+1} *(i)*is the bias of neuron *I* and *ak* (*j)* is the output from neuron *j* in layer *k*.

The output of neuron *I* in layer *k* +1 is:

$$a^{k+1}(i) = f^{k+1}\left(n^{k+1}(i)\right) \quad (3)$$

Here f^{k+1} is the activation function of neuron *i*. For an *L* layer

neural network, the task of the LM algorithm is to minimize the total error:

$$E = \sum_{p=1}^{P}\sum_{m=1}^{M}\left(d_{pm} - a_{pm}^{L}\right)^2 \quad (4)$$

Which can be reduced to

$$E = \sum_{p=1}^{P}\sum_{m=1}^{M}\left(e_{pm}\right)^2 \quad (5)$$

Here *P* and *M* are the number of patterns and the number of outputs respectively, and *dpm*denotes the desired output.

The weight update rule for the LM algorithm is derived from the Newton's method and written as:

$$\Delta w = A^{-1}g \quad (6)$$

Here *A* and *g* are the Hessian and the gradient respectively. For the error function *E*, which is a sum of squares, the Hessian and gradient can be computed as follows:

$$A \cong 2J^T J \quad (7)$$

$$g = 2J^T \vec{e} \quad (8)$$

Here *e*_constitutes the error vector and *J* is the Jacobian of the partial derivative of error with respect to the weights. The Jacobian matrix can be computed by a modified EBP algorithm [11]. The matrix form of the Hessian and the gradient is written as:

$$A = \begin{bmatrix} \frac{\partial^2 E}{\partial w_1^2} & \frac{\partial^2 E}{\partial w_2 \partial w_1} & \cdots & \frac{\partial^2 E}{\partial w_n \partial w_1} \\ \frac{\partial^2 E}{\partial w_1 \partial w_2} & \frac{\partial^2 E}{\partial w_2^2} & \cdots & \frac{\partial^2 E}{\partial w_n \partial w_2} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 E}{\partial w_1 \partial w_n} & \frac{\partial^2 E}{\partial w_2 \partial w_n} & \cdots & \frac{\partial^2 E}{\partial w_n^2} \end{bmatrix} \text{ and } g = \begin{bmatrix} \frac{\partial E}{\partial w_1} \\ \frac{\partial E}{\partial w_2} \\ \vdots \\ \frac{\partial E}{\partial w_n} \end{bmatrix} \quad (9)$$

The LM method solves the problem with ill-defined Jacobian matrix by introducing an identity matrix *I* and learning parameter μ. The LM weight update rule is definedas:

$$\Delta \vec{w} = \left[J^T J + \mu I\right]^{-1} J^T \vec{e} \quad (10)$$

For μ =0 the LM becomes the Guass-Newton method, whereas for larger values of μ the algorithm is reduced to the steepest decent algorithm. Initially μ is set to *0.001*. If the total error (5) increases, 10 multiply μ. In case of error reduction,10 divide the learning parameter.

Based on the constructed training dataset, the training of the neural network is driven by two assumptions:

1) The intrusions can appear anywhere in the attribute space (including within the cluster of normal behavior);

2) There is a cluster of normal behavior somewhere in the attribute space.

By attempting to minimize the classification error, the training algorithm eventually finds the boundary of the normal behavior class. Anything located outside of the class is therefore considered an intrusion. Fig. 1.4 (d) describes thelearned classification function.

The steps of the IDS-NNM algorithm are as follows:

Step 1: Construct an ordered sequence *ST* of attribute vectors *vi*_ using the information from packet headers. The vectors are order time-sequentially:

$$S_T = \{\vec{v}_0, \vec{v}_1, \ldots, \vec{v}_N\} \quad (11)$$

Here, *v*0 and*vN* are the first and the last recorded packets in the sequence, respectively.

Step 2: Extract sequence *SW* of window-based feature vectors *rj*_ from sequence *ST*. This extraction of window-based attributes can be described as:

$$f(\vec{v}_i, \vec{v}_{i+1}, \ldots, \vec{v}_{i+\beta-1}) \rightarrow \vec{r}_j, \quad i, j \in \{0,1,\ldots,N-\beta+1\} \quad (12)$$

Where β denotes the length of the window.

Step 3: Create set S_w*of normal behavior training instances by assigning each feature vector *rj* class label *lNorm*.

$$S_W^* = \{(\vec{r}_j, I_{Norm})\}_{j=1,2,\ldots,N-\beta+1} \quad (13)$$

Step 4: Create randomly generated set *IW* of simulated intrusion vectors uniformly distributed over the window based attribute space.

$$I_W = \{\vec{r}_0, \vec{r}_1, \ldots, \vec{r}_M\}$$

Where *M* is the number of generated intrusion vectors.

Step 5: Create set I_w*of the intrusion training instances by assigning each feature vector *rk* class label *lIntr*.

$$I_W^* = \{(\vec{r}_k, I_{Intr})\}_{k=1,2,\ldots,M}$$

Step 6: Combine setsS_W*and I_W*into a single training dataset *T*:

$$T = S_W^* \cup I_W^*$$

Step 7: Propagate the training dataset *T* to the output of the neural network using (1), (2) and (3).

Step 8: Using the modified EBP compute the Jacobianmatrix.

Step 9: Calculate the weight update vector *wi*by solving (10).

Step 10: Update the network weights and the learning parameter w:

Step 11: If predefined convergence criteria is not met, go to step 7.

VI. MIN-MIN ALGORITHM

Min-Min algorithm selects the smaller tasks to be executed first. Min-Min Algorithm as an improved intruder detector algorithm is introduced on a base of Min-min algorithm in order to improve the intrusion detection activity.

There are two phase in the Min-Min algorithm. In the first phase, it finds the minimum execution time of all attacks. Then in the second phase, it chooses the task with the least execution time among all the tasks. The algorithm proceeds by assigning the attack to the resource that produces the minimum completion time. The same procedure is repeated by Min-Min until all tasks of attacks are scheduled.

Min-Min begins with the set MT (Meta task) of all unassigned attack. As shown in algorithm steps, firstly it computes minimum completion time CTij for all tasks in MT on all resources (lines 1-3). Then two main phases of this algorithm begin. In the first phase, the set of minimum expected completion time for each task in MT is found (lines 5-6). In the second phase, the task with the overall minimum expected completion time from MT is chosen and assigned to the corresponding resource (lines 7-8). Then this task is removed from MT and the process is repeated until all tasks in the MT are mapped (lines 9-11). It is also one of the scheduling algorithms implemented in [6].

VII.CONCLUSION

Intrusion detection is currently attracting interest from both the research community and commercial companies. We have given background of the current state-of-the-art of IDS, based on a proposed taxonomy illustrated with examples of past and current projects. This taxonomy also highlights the recent work and covers the past and current developments adequately. Each of its technique has its own advantages and disadvantages. We believe that no single criterion can be used to completely defend against computer network intrusion. There is no single version of it that can be used as a standard solution against all possible attacks. It is both technically difficult and economically costly to build and maintain computer systems and networks that are not susceptible to attacks. The technique to be selected depends on the specifications of the type of anomalies that the system is supposed to face, the type and behavior of the data, the environment in which the system is working, the cost and computation limitations and the security level required.

An intrusion detection system is proposed which is based on the Min-Minneural network. The proposed system is evaluated using the KDD DARPA dataset and the classification accuracy, classification errors are taken as the performance parameter. The dataset is first preprocessed to remove duplicated instances, outlier and extreme values, and then convert nominal attribute to numeric. The Neural network concept is used for the Training and classification process and the Min-Min algorithm is used for detection attack. The proposed method is novel and it will have the online adaption capability, nonlinear reparability, can distinguish from the overlapping classes, nonparametric classification, and less training time requirement for training

compared to the traditional neural networks. The critical experimentation on the proposed system shows that the system performs well.

VIII. REFERENCE

[1] I. F. Akyildiz et al., "Wireless Sensor Networks: A Survey, "Elsevier Comp. Networks, vol. 3, no. 2, 2002, pp. 393–422

[2] G.Li, J.He, Y. Fu. "Group-based intrusion detection system in wireless sensor networks" Computer Communications, Volume 31, Issue 18 (December 2008)

[3] Michael Brownfield, "Wireless Sensor Network Denial of Sleep Attack", Proceedings of the 2005 IEEE Workshop on Information Assurance and Security United States Military Academy, West Point, NY.

[4] FarooqAnjum, DhanantSubhadrabandhu, SaswatiSarkar *, Rahul Shetty, "On Optimal Placement of Intrusion Detection Modules in Sensor Networks", Proceedings of the First International Conference on Broadband Networks (BROADNETS04)

[5] Parveen Sadotra *et al*, International Journal of Computer Science and Mobile Computing, Vol.5 Issue.9, September-2016, pg. 23-28

[6] K. Akkayaand M. Younis, ―A Survey of Routing Protocols in Wireless Sensor Networks,‖ in the Elsevier Ad Hoc Network Journal, Vol. 3/3 pp. 325-349, 2005

[7] A. Abduvaliyev, S. Lee, Y.K Lee, "Energy Efficient Hybrid Intrusion Detection System for Wireless Sensor Networks", IEEE International Conference on Electronics and Information Engineering, Vol.2, pp. 25-29, August 2010.

[8] Parveen Sadotra and Chandrakant Sharma. A Survey: Intelligent Intrusion Detection System in Computer Security. *International Journal of Computer Applications* 151(3):18-22, October 2016

[9] A. Araujo, J. Blesa, E. Romero, D. Villanueva, "Security in cognitive wireless sensor networks. Challenges and open problems", EURASIP Journal on Wireless Communications and Networking, February 2012.

[10] A. Becher, Z. Benenson, and M. Dorsey, \Tampering with motes: Real-world physical attacks on wireless sensor networks." in SPC (J. A. Clark, R. F. Paige, F. Polack, and P. J.Brooke, eds.), vol. 3934 of Lecture Notes in Computer Science, pp. 104{118, Springer, 2006.

[11] I. Krontiris and T. Dimitriou, \A practical authentication scheme for in-network programming in wireless sensor networks," in ACM Workshop on Real-World Wireless Sensor Networks, 2006

[12] M. Ali Aydın *, A. HalimZaim, K. GokhanCeylan "A hybrid intrusion detection system design for computer network security" Computers and Electrical Engineering 35 (2009) 517–526.