2-1-2014

Energy Prediction Based Intrusion Detection In Wireless Sensor Networks

Nancy Alrajei
Oakland University

George Corser
Oakland University, gpcorser@oakland.edu

Huirong Fu
Oakland University, fu@oakland.edu

Ye Zhu
Cleveland State University, y.zhu61@csuohio.edu

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 2, February 2014)

Energy Prediction Based Intrusion Detection in Wireless Sensor Networks

Nancy Alrajei[1], George Corser[2], Huirong Fu[3], Ye Zhu[4]
[1,2,3]Oakland University, Computer Science and Engineering Department, Rochester, Michigan 48309
[4]Cleveland State University, Electrical and Computer Engineering Department, Cleveland, Ohio 44115

***Abstract*—A challenge in designing wireless sensor networks is to maximize the lifetime of the network with respect to limited resources and energy. These limitations make the network particularly vulnerable to attacks from adversaries. Denial of Service (DOS) is considered a severely damaging attack in monitoring applications when intruders attack the network and force it to lose its power and die early. There are intrusion detection approaches, but they require communications and calculations which waste the network's limited resources. In this paper, we propose a new intrusion detection model that is suitable for defending against DOS attacks. We use the idea of energy prediction to anticipate the energy consumption of the network in order to detect intruders based on the each individual node's excessive usage of power. Our approach does not require a lot of communications or calculations between the nodes and the cluster head. It is energy efficient and accurate in detecting intruders. Simulations show that our energy aware intrusion detection approach can effectively detect intruders based on energy consumption rate.**

***Keywords*—energy consumption, intrusion detection, probabilistic approach, energy map, energy dissipation**

I. INTRODUCTION

Wireless sensor networks (WSN) consist of individual nodes that are able to perceive their environment, communicate with nearby nodes via radio broadcast, and perform computations based on information gathered from their surroundings. By monitoring states of individual nodes in a WSN it is possible to predict power consumption and, if excessive, detect intruders.

The main goal of such network is to perform distributed sensing tasks for applications, such as environmental mentoring. The deployment of sensor network in hostile environments, combined with limitations of the sensors, limited power, and memory and computation resources makes them vulnerable to variety of attacks. The network lifetime is the time span from the deployment to the instant when the network is considered non-functional. Once these sensors are deployed, it is almost impossible to conduct regular maintenance.

Due to the fact that network may consist of a very large number of nodes or the nodes may be in an environment in which human intervention is difficult or undesirable. So battery reserve becomes a valuable resource since it cannot be replaced.

The wireless sensor network is vulnerable to attacks due to its limitations and lack of structure. The most common attack in monitoring networks is Denial of Service (DOS) attacks where intruders attack parts of the network and flood them by requests forcing the sensor nodes to deplete their power which makes them unavailable to perform their primary function of monitoring their environment. We define intruders as nodes in the network that are engaged in intensive and useless activities that do not contribute to the main purposes of the network, which is environmental monitoring. So the need for an intrusion detection mechanism has become an essential part of the network protocols and algorithms since firewalls alone do not succeed to stop such intruders from interrogating the boundaries of the network.

A. Problem Statement

No solutions to date to the WSN DOS detection problem have considered variable energy consumption of states, behavior of nodes, and the messaging required to enable the solution. First, there are works that focus on the area of energy consumption in WSN, but most of the approaches do not consider the characteristics of the nodes. Naïve methods assume that nodes lose the same amount of power in each activity it conducts in the network, such as sensing, transmitting, and receiving data. It is important to recognize the difference in power consumption in each state, such as in the state of transmitting data, a sensor node consumes the highest power rate over all other states. The state of sensing consumes less than transmitting but more than sleeping.

Second, we found that intrusion detection studies focus on the nature of the behavior of nodes, whether it is anomaly detection [15] or signature based detection [14], it does not take energy consumption into account.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 2, February 2014)

Third, we also found that intrusion detection in general requires a large number of messages transmitted to the base station or the cluster heads. We noticed intrusion detection approaches for DOS attacks in monitoring applications consume a lot of energy for data transmission and processing, with high false negative or false positive rates. In monitoring applications, energy should be managed wisely to extend the lifetime of the network. So this brings up the need for an efficient intrusion detection that is low in data transmitting and processing, and as efficient as expensive approaches in terms of catching the intruder as early as possible. We are looking for less communications back and forth with the cluster head or the base station, and less data processing.

B. Related work

When we surveyed the literature, we focused on both energy consumption and intrusion detection in WSN. The literature covers a variety of approaches used to detect intruders [11-19]. Some of the common approaches catalog patterns of attacks seen in the past and focus on watching the incidence of behavior consistent with these patterns. These include the detection of wormhole attacks, routing holes, or detection of particular operations, like routing, localization, etc. All these approaches tend to be taxing in terms of storage and computation which is an important issue for sensor nodes with their limited storage and processing capacity. Furthermore, the patterns that they cataloge tend to be generic rather than application specific [17].

For example in [11][17] the intrusion detection process consist of a two-step approach: In the first step, a profile is created to characterize intruder behavior. In the second phase, while the network is operating, the observed behavior is compared with what has been catalogued and flagged if it matches catalogued abnormal behavior.

When dealing with malicious intrusions, the network is constantly at risk with new enemies (that may use a different pattern from the ones in the catalog) As a result of this approach, intrusion detection can be divided according to what they protect. That is not effective in the very specialized application specific context of sensor networks [11].

The energy studies in WSN are mostly related to designing energy efficient routing protocols in sensor networks. For directed diffusion, proposed in [9], all communications are neighbor to neighbor with no need for a node addressing mechanism. It is energy efficient because each node can do aggregation and caching.

Another protocol is LEACH (Low Energy Adaptive Clustering Hierarchy) [6][20]. The main idea of LEACH is to consider the local data fusion in each cluster. Each cluster head, instead of each sensor node, directly sends fusion data to the base station. The clustering infrastructure is used because of the high correlation between data from nodes located close to each other. This allows all data from nodes within the cluster to be processed locally using data aggregation techniques, can be used to combine several correlated data into a smaller set of information that maintains the effective. Therefore, much less actual data needs to be transmitted from the cluster to the base station.

PEGASIS [6] is another protocol that builds a chain using a greedy algorithm to route the data to the leader of all nodes. BCDCP [6] creates a minimum spanning tree by the base station to build the routing path and gains more energy efficient than PEGASIS.

Another energy efficient protocol explained in [7] is TEEN (Energy Efficient Sensor Network Protocol), which is reactive, event-driven protocol for time-critical applications. A node senses the environment continuously, but turns radio on and transmits only if the sensor value changes drastically, so members only send data to the cluster head only if data values are in the range of interest.

The estimation of the sensor node energy consumption is a multi-parameter problem, depending on the hardware characteristics and the protocol decisions throughout the OSI layer range. Microprocessors are considered the primary power consumer. The power consumed by the microprocessor is related to the frequency of its clock, the voltage supply, and the time that is needed to perform a given task, which in turn is related to the code that is executed, and to the power saving features that are implemented [2].

In [1] they proposed an energy prediction model that we adopted in this paper. Knowing the states of sensor nodes (sense, send, receive, sleep) in the network from the past history, and the amount of energy consumed by each state, is enough to predict its energy dissipation rate. This model will be explained in details in the next section.

C. Contribution

The contribution of this paper is a prediction-based intrusion detection scheme. We adopted the generic prediction model from [1] and used it to detect intruders in the network based on their energy consumption. Our solution is based on the premise that intruders are nodes tend to consume energy more than normal. We will explain this concept in details and show how we adopted it to build an efficient and economic intrusion detection model.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 2, February 2014)

The paper is organized as follows. In Section 2 we present the energy prediction model that we used. Next, in Section 3, we describe our solution, energy prediction based intrusion detection. Then, in Section 4, we list the performance metrics to evaluate our model. In Section 5 we evaluate our proposed solution via simulation and analyze the results. Finally, we conclude in Section 6.

II. Energy Prediction Model

A. Prediction based energy map

The information about the remaining available energy in each part of the network is called the energy map and can aid in prolonging the lifetime of the network [8]. This knowledge is helpful in many applications, such as reconfiguration algorithms, query processing, and data fusion. It is also important to know where to increment the deployment of nodes in the low energy areas. Figure 1 shows an energy map. Light shaded areas represent regions with more remaining energy and regions short of energy are represented by dark shaded areas.

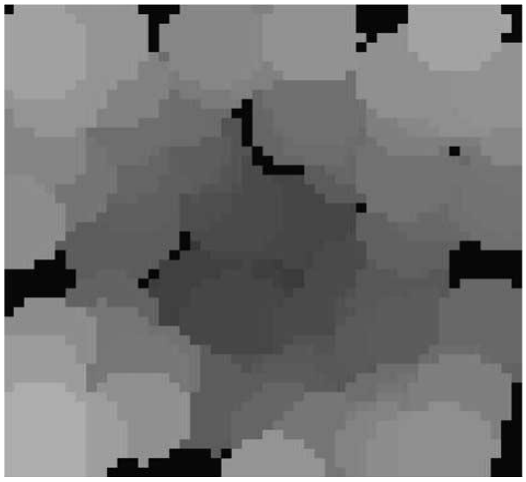

Figure 1: Energy map of a wireless sensor network[1]

B. Probabilistic Model Based on Markov Chains

In order to predict the dissipated energy, we studied Markov chains probabilistic model from [1][8]. Each sensor node can be modeled by Markov Chain with states $X_1 \ldots X_4$ that represent the states of a node during the time. For example X_1 = Sense, X_2 = Send, X_3 = Receive, X_4 = Sleep. If a sensor node has M operation modes, at each time the node is in state i, there is a fixed probability, P_{ij}, that is, in the next time-step, it will be at state j. This probability can be represented by the *transition probability matrix*, P, which is the matrix consisting of the one-step transition probabilities, P_{ij}.

$$P_{ij} = Pr\{X_{n+m} = j \mid X_n = i\} \quad (1)$$

Then a transition probability matrix P is generated for each node with M rows and M columns, each element of row i column j represents the probability Pij. Which can be calculated based on a node's history, one way find it is the number of times the node was in state i and went to state j divided by the total number of time steps the node was in state i. The expected probability matrix Pij for each node will look like this:

$$\begin{matrix} P_{11} & P_{12} & \ldots & P_{1M} \\ P_{21} & P_{22} & \cdots & P_{2M} \\ . & . & \cdots & . \\ P_{M1} & P_{M2} & \ldots & P_{MM} \end{matrix}$$

The m-step transition probability $X = \{ X_1, X_2, \ldots X_m\}$ is the probability of transitioning from state i to state j in m steps.

$$P_{ij}^{m} = Pr\{X_{n+m} = j \mid X_n = i\} \quad (2)$$

The m-step transition matrix whose elements are the m step transition probabilities $P_{ij}^{(m)}$ is denoted as $P^{(m)}$. It can be found by multiplying the single-step probability matrix by itself m times,

$$P^{(m)} = P.P.P \ldots P = P^{m} \quad (3)$$

With the knowledge of probabilities P_{ij}^{m} for all nodes and the initial state of each node, it is possible to use these probabilities to predict the energy drop of a sensor node.

The first step is to calculate for how many time-steps a node which is currently in state i will stay in state s. As $P_{is}^{(t)}$ represents the probability of a node currently in state i will be in state s after t transitions, for any state s the number of time steps a node will stay in state s can be collected as:

$$\sum_{t=1}^{T} P_{is}^{(t)} \quad (4)$$

For a node that is currently in state i, its expected amount of energy spent in the next T times, $ET(i)$

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 2, February 2014)

$$E^{T}_{(i)} = \sum_{s=1}^{M} \left(\sum_{t=1}^{T} P_{is}^{(t)} \right) * E_s \qquad (5)$$

Where E_s is the amount of energy dissipated by a node that remains onetime-step in state s .Using E^T each node can find its energy dissipation rate (ΔE) for the next T time steps. Each node then sends its available energy and its (ΔE) to the monitoring node. The monitoring node maintains estimation for the dissipated energy at each node by decreasing the value (ΔE) periodically for the amount of remaining energy of each node. The better the estimation the node can do, the fewer the number of messages necessary to obtain the energy information.

III. Energy Prediction Based Intrusion Detection

In monitoring application the most damaging type of attack is Denial of Service (DOS), when the goal of the attacker is to build a connection with the nodes and overload them with requests forcing them to deplete their power. For that reason, intruders require significant amount of energy for their intensive interaction with their surrounding nodes. But the non-compromised nodes lose power only for being engaged in legitimate functions.

We adopted the approach explained in the previous section to be able to measure the energy consumed in the network, and under our assumption that the intruder is more active than other nodes we want to test if we can detect them based on the power consumption by itself and how accurate it is.

Each node locally constructs its own transition probability matrix based only on its past history of the period $[t\text{-}5, t]$ where t = current time. In this case, P_{ij} will be the number of times a node was in state i and went to state j divided by the total number of time-steps the node was in state i. This information is sent to the cluster head every fixed time intervals, which is 5. With this matrix, each node uses Equation *(4)* to find its energy dissipation rate for the next $t + 5$ time period. The result of this matrix will give us the predicted energy drop of a sensor node.

We have six possibilities of states X_1, X_2... X_6 a node can be in, each states represents what a node can be doing in a single time step. Radio communication can be either sending or receiving a message. See Table 1.

TABLE I
Six States of a Node

State VAR	Sensing	Radio	Energy consumed in a time step
X1	on	off	1
X2	on	sending	3
X3	on	receiving	3
X4	off	sending	2
X5	off	receiving	2
X6	off	off	0.1

X1 is the starting state for all nodes. After some fixed time interval *t*. each cluster head queries the nodes. The nodes receive the query and respond with their values. The energy consumed by a sensor node if it stays in each operating states for one time step is [1, 3, 3, 2, 2, 0.1] for $X_1, \ldots X_6$ respectively, as listed in table 1. These values are inspired by the fact that the most expensive operation in sensor network is radio transmission, then sensing, but if they are not doing any of these two, they go to the sleeping mode, which consumes a very minimum energy.

The cluster head is responsible for:

- Aggregating the query from its member first, then with other cluster heads and then transmits it to the base station.
- Choosing the subset of nodes to interrogate, rather than querying all the nodes. For that purpose, it calculates the Usefulness metrics as shown in eequation (6) for its members. This tells the cluster head which nodes are more useful than the others, and based on the result, the cluster head will choose a subset of its members who have high usefulness values.
- Intrusion detection, a cluster head watches its nodes and is responsible for identifying any abnormal behavior among these nodes. In our approach, it is responsible for calculating the energy dissipation rate for its members.

In our solution we use the following two strategies combined when the cluster head selects and queries nodes to ensure maximum energy saving, selective querying and information value based transinformation.

A. *Selective Querying*

This approach explained in [10], the idea is to query the minimum number of nodes that is enough to tell us accurate answer about the query. The selective querying is the cluster head responsibility that is based on the following premise.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 2, February 2014)

Given a query *Q* computing some aggregate function *f*, given a set *S* of sensor nodes, it is possible to find a relatively small– subset S_0 of S such that $f(S_0)$ provides us with a good approximation of *f(S)*. The ideal of implementation of this premise is to select the S_0 that contains the right size of nodes and right contents of data. This subset should guarantee us two things, first, it should detect emerging event, second, it should include nodes that have proven to be the most relevant in the recent past.

We define S_0 the subset of nodes from each cluster to be queried S_0 such that it includes the scrutiny set and the exploratory set- the scrutiny set makes 75% of S_0 - contains nodes that show high relevance in the recent past, normally it includes nodes concentrated around the emerging event so that we ensure high accuracy of the query. On the other hand the exploratory set - makes 15 % of S_0 - is a randomly selected set of nodes that is to give an opportunity to other nodes to be queried where an event might take place, so it gives a wider picture of the whole network. The selection of nodes is done using the next strategy.

B. Information Value Based Transinformation

The idea of information value is to select a number of nodes to query in each iteration that show high information value and terminating the other ones that show low information value. The information collected from one node is not useful by itself, instead the values of all the nodes aggregated together that what matters in monitoring applications.

So one reading from individual node is not our interest here, rather we are interested in knowing its contribution to answer the query. In our case we use this concept to define Usefulness, a time-variable correlation between a query *Q* and a sensor node *Ni*, up to the current time t. The transinformation of *Q* and *Ni* is denoted:

$$U(Q,Ni,t) = \sum_{[t-\delta t,t]} p(q,mi)\log\left(\frac{p(q,mi)}{p(q)\,p(mi)}\right) \quad (6)$$

Where *mi* is the message associated with node *Ni* and $p(x)$ is the probability of *x*, and $p(x,y)$ is the joint probability of *x* and *y*, where *x* is a sensor node value and *y* is query value given the discrete probability distribution calculated over period Δt. We focus on recent history over a selected time interval of length δt. The relevance of a node to the *query* will vary over time, because natural phenomena are continuous over time and space, we assume that the relevance of a node at present time is highly correlated with its relevance over the recent history. This is explained in details in [10][16].

IV. Performance Metrics

To evaluate our scheme, we used two performance metrics, mean square error and energy dissipation rate.

A. Mean Square Error (MSE)

To determine the accuracy of this method, we calculated the mean square error between the actual energy power consumption for the whole network, and the predicted one. Error rate for node *i* at time *t* is as follows.

$$MSE_{i,t} = (E_{i,t} - \bar{E}_{i,t})^2 \quad (7)$$

This is the difference between the estimated power consumption $\bar{E}_{i,t}$ and the real power consumption $E_{i,t}$. The model is expected to have a low difference between the two values.

B. Energy Dissipation Rate

The model is expected to show different Energy Dissipation Rate for intruders than other nodes to detect them. See prior equation (5).

V. Simulation And Analysis

We used Matlab simulation to study how accurate and effective our approach is. We have 200 nodes divided into 4 clusters. These nodes are placed on a 100 x 100 grid. Nodes are first placed randomly in the field, and then divide the field into 4 equal clusters as in Figure 2.

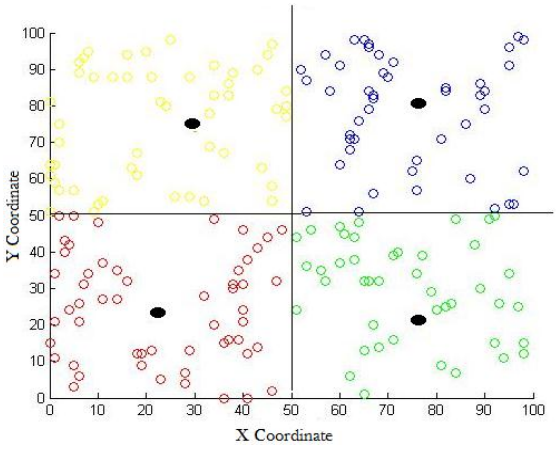

Figure 2: All nodes in 4 clusters. Filled nodes are cluster heads.

A cluster head is picked based on the location of a node to be in middle of the cluster. All input values (the data) for the nodes are simulated using equation 8. In this equation the event is centered at *(a, b)*, with a peak in value at that point and exponential decline as we get further from the center *(a, b)*.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 2, February 2014)

$$f(x,y) = h * e^{(-(x-a)^2 - (y-b)^2)/w} \quad (8)$$

Where h is the range of the phenomenon, or the height of the peaks in the data. *w*, the radius of the phenomenon or the width. The smaller is *w*, the narrower is the peak, and the steeper is the decline. With a large *w*, the data changes more slowly. *(a, b)* the location of max or the center of the peak, moves with time along with *h* and *w*. In other words, *a* and *b* are in fact functions of time *t.*

We have used linear movement pattern, as shown in Figure 3. The interest is to find the MAX reading and the max value is moving linearly in each iteration.

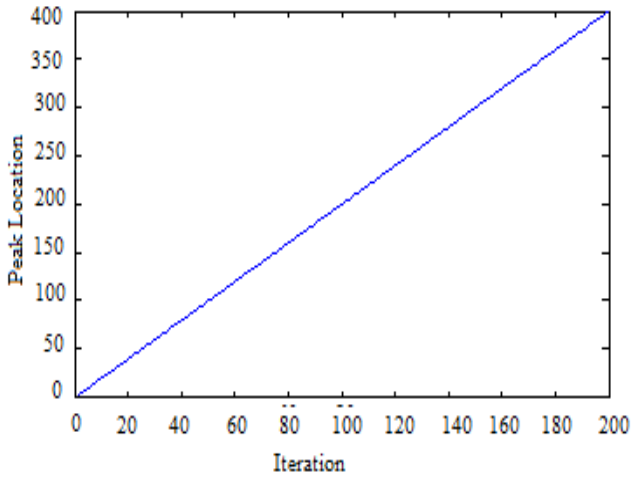

Figure 3: Linear event movement

The cluster head calculates the usefulness for each node using equation 6 every time interval t = 5 along all the iterations. The default usefulness value is 0, the usefulness for all intruders remains around 0.

We cannot use usefulness alone to detect intruders, because we query only 12% of the nodes in each iteration so a lot of the nodes are not being interrogated at all specially nodes located far away from the event, thus their usefulness remains at the default value. The nodes interrogated are the nodes were selected based on selective querying approach (scrutiny set and exploratory set). These nodes are responsible for answering the query, then continue sensing for the next 5 iterations, in the end of that time, cluster heads re-calculate the usefulness of these nodes pick a new set to query based on the results of high usefulness, a node might be selected over again if it shows high usefulness.

For the rest of the nodes in the cluster that were not selected by the cluster head to sense and respond to the query will automatically go to sleep, that is to save the nodes energy, therefore extend the lifetime of the network.

The intruders are selected randomly among other nodes, and make them act like cluster heads, in the sense that they are always in state X_4 and X_5 alternately, so they send continuous queries to their neighbors, and receive their readings .

Nodes shift states (sensing, receiving, sending, and sleeping). If a node was selected from the cluster head to be queried it will switch its state to communicate back and forth with the cluster head.

All nodes keep track of the past short history of their states. So each node will be modeled using Markov Chain. Each node submits to the cluster head every time interval of length δ their current amount of power, so the cluster head predicts the energy dissipation for each node for the next δ time intervals using equation (5). A record of predicted power usage and actual power used is kept at the cluster head.

Any abnormal power consumed should raise a flag that this node is suspicious. The detection part is the difference between the actual and the predicted power. A threshold is set to separate node from intruders.

We plot *MSE* for all the nodes, as in Figure 4. Intruders have lower *MSE* than other nodes, which means the prediction of their power consumption is close to the real power consumed.

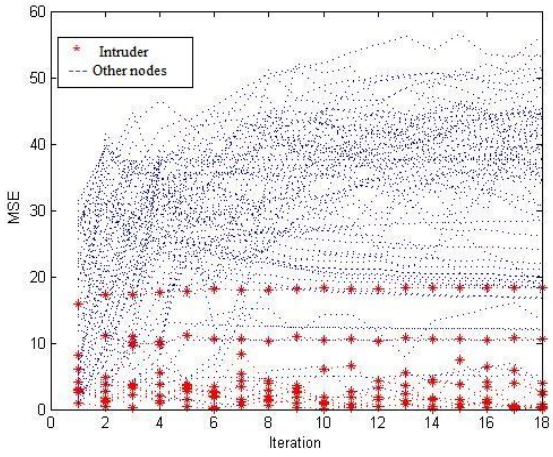

Figure 4: Mean square error for all nodes

In Figure 5 we plotted the average consumption of power for the whole network and the estimated one. That also shows that they are close to each other. At the beginning of the simulation values of estimated power are calculated after few iterations to collect historical data, which explains why the dotted line starts after 2.

International Journal of Emerging Technology and Advanced Engineering
Website: www.ijetae.com (ISSN 2250-2459, ISO 9001:2008 Certified Journal, Volume 4, Issue 2, February 2014)

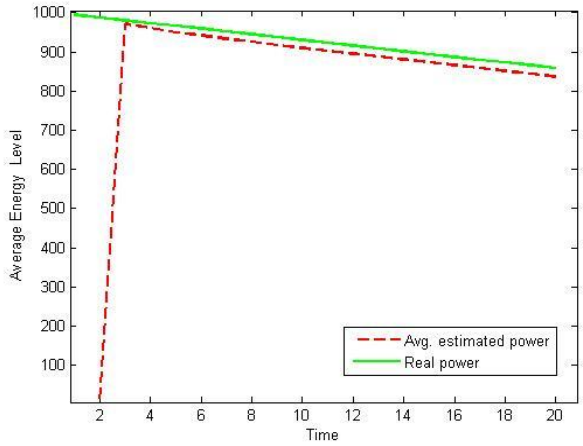

Figure 5: Average estimated power (dotted) vs. average real power (solid)

The dotted lines in Figure. 5 represent the average estimated power level for all nodes along all the iterations. The solid line is average of the real (actual) power level for all nodes. The estimated power line is close to the real (actual) one, which indicated that our model can predict closely how the network is going to lose its power. Even we notices that the two lines decline in parallel with each other.

In Figure. 6 shows the energy Dissipation Rate for all nodes. Clearly, we see (squares) intruder's dissipation rate energy drops significantly comparing with other nodes, due to their excessive functions. From Figure. 6 we can tell the intruders from non-intruders by setting a threshold at the energy dissipation rate after 18 iterations we can clearly distinguish intruders from others, the threshold is set to be 700.

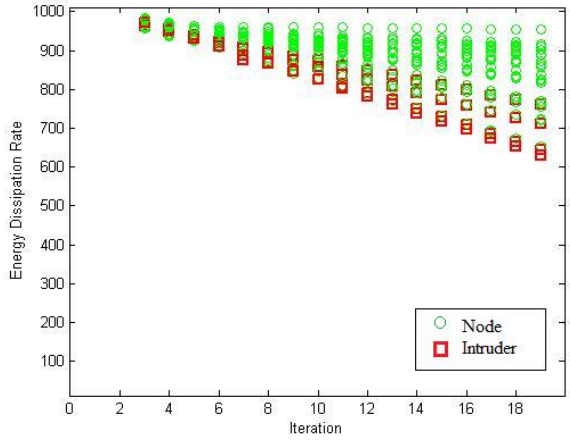

Figure 6: Power level for all nodes along all the iterations

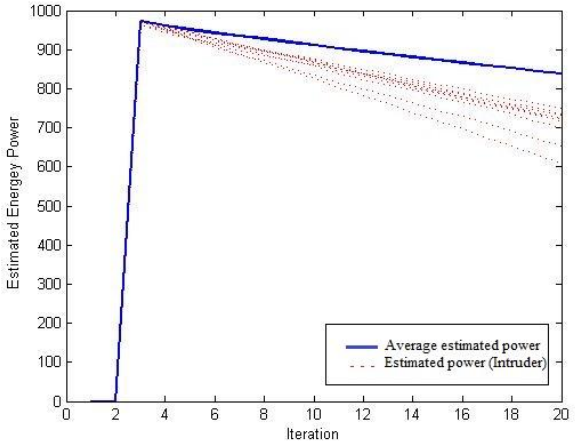

Figure 7: Average estimated power vs. estimated power for each intruder

In Figure. 7 if look at average estimated power for all nodes which is the solid line, and the estimated power for all intruders (dotted lines), we see that they are significantly less than the average.

From comparing the average estimated power with the estimated power for each intruder separately, we can tell how much the intruders consume more energy than normal.

VI. Conclusion

In this paper, we proposed an efficient intrusion detection that is based on energy prediction in wireless sensor networks. We combine different approaches in our simulation to minimize energy waste, we used the probabilistic approach to predict energy consumption in senor network, as well as the naive approach to calculate the power level.

We injected intruders into the network that does not participate anything to the query but perform one type of denial of service attack. Intruders are highly active but not useful.

We tested if we can detect intruders based on the predicting model we have. Intruders are expected not participate in answering the query as their history tells so, but instead they engage themselves in intensive activates with other nodes that make them significant in our simulation results due to their huge power consumption.

Our model we proposed is not expensive as other intrusion detection schemes that require a lot of communication and calculations. It is efficient, cheap and accurate to detect intruders.

REFERENCES

[1] R. A. F. Mini, B. Nath, and A. A. F. Loureiro, "A probabilistic approach to predict the energy consumption in wireless sensor network," In IV Workshop ," In IV Workshop on Wireless Communication and Mobile Computing, Seo Paulo, Brazil, October 23-25 2002.

[2] G. Dimitriou, P.K. Kikiras , G.I. Stamoulis and I.N. Avaritsiotis, "A Tool for Calculating Energy Consumption in Wireless Sensor Networks," ISBN 978-3-540-29673-7, Pages 611-621, 2005

[3] S.Croce, F. Marcelloni, and M. Vecchio, "Reducing Power Consumption in Wireless Sensor Networks Using a Novel Approach to Data Aggregation," Oxford University Press of the British Computer Society. 2007.

[4] Ultra low power IEEE 802.15.4 compliant wireless sensor module, Humidity, Light, and Temperature sensors with USB, Datasheet (2/6/2006).

[5] NV Subramanian, "Survey on Energy-Aware Routing and Routing Protocols for Sensor Networks". Technical Report, Computer Science, University of North Carolina, Charlotte, 2004

[6] G. Huang, X. Li, and J. He, "Energy-efficiency analysis of cluster-based routing protocols in wireless sensor networks," in Aerospace Conference, IEEE, 2006, pp. 1-8.

[7] K. Akkaya and 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.

[8] F. Mini, M. do Val Machado, A. Alfredo F. Loureiro, and B Nath, "Prediction-based energy map for wireless sensor networks," Ad Hoc Networks 3(2): 235-253 (2005)

[9] C. Intanagonwiwat, R. Govindan, D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," Proceedings of MOBICOM, Boston, 2000, pp. 56–67.

[10] Yousuf F.Mili and N.Alrajei, "Information Theory Based Intruder Detection in Sensor Networks", 3rd Indian International Conference on Artificial Intelligence, Dec 2007.

[11] Demirkol, F. Alagoz, H. Delic, and C. Ersoy, " Wireless sensor networks for intrusion detection: packet traffic modeling", 10(1):22–24, January 2006.

[12] Ammari, H. M. and Das, S. K. 2009. " Fault tolerance measures for large-scale wireless sensor networks", ACM Transaction on Autonnmous and Adaptive Systems 4, 1 (Jan. 2009), pp. 1-28

[13] Srivastava P , Rai P, and Singh U, "Intrusion detection: An energy efficient approach in heterogeneous WSN", Intrenational conference on: Emerging Trends in Electrical and Computer Technology (ICETECT), March 2011 , Page(s): 1092 - 1096

[14] B. Yu and B. Xiao, "Detecting selective forwarding attacks in wireless sensor networks," in Proc. 20th Int'l Conf: Parallel and Distributed Processing Symposium, Apr. 2006.

[15] V. Bhuse and A. Gupta, "Anomaly intrusion detection in wireless sensornetworks," J. High Speed Netw., vol. 15, no. 1, pp. 33–51, 2006

[16] C. Shannon, "A mathematical theory of communication," Bell System Technical Journal, vol. 27, pp. 379–423,623–656, July, October 1948.

[17] J. Deng, R. Han, and S. Mishra, "Defending against path-based dos attacks in wireless sensor networks," In SASN '05: Proceedings of the 3rd ACM workshop on Security of ad hoc and sensor networks, pages 89–96, New York, NY, USA, 2005.ACM Press.

[18] K Ioannis, T. Dimitriou, and F.C. Freiling, "Towards intrusion detection in wireless sensor networks," Proc. of the 13th European Wireless Conference. 2007.

[19] Padmavathi, Dr G., and Mrs Shanmugapriya. "A survey of attacks, security mechanisms and challenges in wireless sensor networks." arXiv preprint arXiv:0909.0576 (2009).

[20] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660–670, 2002

[21] Ballarini, Paolo, Lynda Mokdad, and Quentin Monnet. "Modeling tools for detecting DoS attacks in WSNs." Security and Communication Networks (2013).