Egyptian Informatics Journal (2012) 13, 225–238

Cairo University

Egyptian Informatics Journal

www.elsevier.com/locate/eij
www.sciencedirect.com

ORIGINAL ARTICLE

A comparative performance evaluation of intrusion detection techniques for hierarchical wireless sensor networks

H.H. Soliman [a], Noha A. Hikal [b], Nehal A. Sakr [b,*]

[a] *Communications Engineering Dept., Faculty of Engineering, Mansoura University, Mansoura, Egypt*
[b] *Information Technology Dept., Faculty of Computer Science and Information Technology, Mansoura University, Mansoura, Egypt*

Received 30 April 2012; revised 18 September 2012; accepted 8 October 2012
Available online 21 November 2012

KEYWORDS

Intrusion detection system;
Wireless sensor network;
Security evaluation metrics;
Clustering;
Discrete Wavelet Transform;
Support Vector Machine

Abstract An explosive growth in the field of wireless sensor networks (WSNs) has been achieved in the past few years. Due to its important wide range of applications especially military applications, environments monitoring, health care application, home automation, etc., they are exposed to security threats. Intrusion detection system (IDS) is one of the major and efficient defensive methods against attacks in WSN. Therefore, developing IDS for WSN have attracted much attention recently and thus, there are many publications proposing new IDS techniques or enhancement to the existing ones. This paper evaluates and compares the most prominent anomaly-based IDS systems for hierarchical WSNs and identifying their strengths and weaknesses. For each IDS, the architecture and the related functionality are briefly introduced, discussed, and compared, focusing on both the operational strengths and weakness. In addition, a comparison of the studied IDSs is carried out using a set of critical evaluation metrics that are divided into two groups; the first one related to performance and the second related to security. Finally based on the carried evaluation and comparison, a set of design principles are concluded, which have to be addressed and satisfied in future research of designing and implementing IDS for WSNs.

1. Introduction

A wireless sensor network, shortly WSN, is an innovative large scale network that is made up of spatially distributed autonomous sensors, which cooperatively collect information through infrastructure less wireless network. The development of WSN was originally motivated by military applications. But recently, they are employed in different fields and for different objectives, including civilian application areas, environment monitoring, habitat monitoring, healthcare applications, home automation,

* Corresponding author.
E-mail addresses: hhsoliman@yahoo.com (H.H. Soliman), nmhikal@yahoo.com (N.A. Hikal), nsakr_cs@yahoo.com (N.A. Sakr).

Peer review under responsibility of Faculty of Computers and Information, Cairo University.

Production and hosting by Elsevier

1110-8665 © 2012 Faculty of Computers and Information, Cairo University. Production and hosting by Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.eij.2012.10.003

and traffic control [1]. While benefiting from the convenience and useful applications of these networks, they are exposed to various types of attacks, since the usage of WSN has a special nature; usually deployed in remote places, left unattended, and use publically accessible communication channels. These attacks are commonly classified according to the affected communication layer protocol [2]. So securing the WSN is a necessity and a challenging work.

A number of traditional security mechanisms have been proposed for securing these networks such as secure routing [3] and data aggregation protocols [4], but they cannot alone ensure enough security for WSN, since it is possible for an attacker to compromise a sensor node and inject false data into the WSN. Moreover, prevention-based techniques arises including authentication and data encryption [5], but they are not enough for ensuring data security; since whenever broken through compromised nodes, the attacker could extract security sensitive information (e.g. secret key). So these techniques act as a first line of defense. Consequently, developing an intrusion detection system, or IDS, as a second line of defense became a necessity. Since it has the ability to monitor the network activity in order to detect any malicious action or intruder and reacts as quickly as possible to the occurring attacks. Many approaches were introduced for developing IDS [6–14]. Most of the existing solutions have strengths and weakness points. Concluding that, despite all efforts it is impossible to have a completely secure system.

This paper introduces a comparative evaluation study for the newly and recently applied anomaly based IDS. Different techniques are investigated. For each technique, the main idea and the related functionality is briefly presented and evaluated. The different techniques are compared based on critical evaluation metrics. Moreover, the strengths and weakness of each technique are compared and discussed. The rest of this paper is organized as follows. Section 2 introduces a complete background of the IDS functionality and classification. Section 3 briefly analyzes and evaluates the recently and common anomaly-based IDS for hierarchy WSN architecture, focusing on both the operational strengths and weakness. Section 4 presents a comparison and evaluation results based on a set of critical performance and security metrics. Finally, Section 5 introduces a conclusion based on the carried out comparison and suggests ideas to enhance the performance of IDS in future researches.

2. Intrusion detection system classification

In order to discuss applying IDS for the hierarchical WSN, the nature of the WSN must be considered first. Since the WSN is characterized by its limited resources, it implies many constrains compared to a traditional computer network. These constrains can be summarized as [15]: (i) Node constraints; memory size, energy levels, and computing capability, (ii) Network constraints; bandwidth, unreliable communications, and (iii) Physical limitations; due to remote management it is widely exposed to be tampered.

Therefore, the deployment of IDS must be visualized through various aspects. The existing techniques classify the IDS according to [16]: (i) Detection methods, (ii) IDS architecture, and (iii) Decision making methods. Each one of these categories is classified itself into different classes. Fig. 1 summarizes the classification procedure. Practically, the IDS must combine a feature from each class, or sometimes more than one feature (hybrid).

3. The studied anomaly based-IDSs for hierarchical WSN

This paper mainly targets comparing the different anomaly-based IDSs for hierarchical WSN architecture. Selecting anomaly-based as a detection method returns to its methodology which is characterized by being flexible, resource friendly, and its ability to detect unknown attacks compared to the other methodologies (signature, specification) that need complicated expression, computing and sizeable memory which WSNs usually cannot afford [17]. Moreover, majority of the proposed schemes in this research direction focused on the hierarchical architecture of WSN. Since it is recommended to assign special equipped nodes (cluster heads) to carry out the IDS processing overhead, in addition to using multi-level transmission which reduce the communication complexities, those not accomplished in other WSN architectures.

This section briefly explains the recently and common anomaly-based IDS for hierarchical WSN architecture. These systems are organized according to their detection technique as: Data mining and artificial intelligence, Hybrid detection techniques, and Statistical-based techniques. At the end of this section, the strengths and weaknesses of the studied IDSs are presented in Table 1.

The investigated techniques that undergo the category of data mining and artificial intelligence are: IDS based on Fuzzy C-Means [19], IDS based on supervised learning using Back Propagation Neural Network [20], and IDS using Agglomerative Clustering [21] and. The investigated techniques that undergo the category of hybrid detection are: IDS based on self organized map NN with K-means clustering [22] and another one based on self organized map NN with Discrete Wavelet Transform [23]. Finally, the investigated techniques that undergo the category of statistically-based techniques are: IDS using the Naïve Bayesian classifier [24] and the IDS based on Support Vector Machine [25].

3.1. IDS using Fuzzy C-Means clustering

In [19], an anomaly-based IDS using Fuzzy C-Means clustering (FCM) with hierarchical network architecture was introduced, that detect routing attacks caused by abnormal flows of data. The basic idea of FCM clustering analysis is to gather similar data in a cluster where the same types of data should be near, and the different types should be very far. The cluster heads are responsible for collection of all regions detection information to be conveyed to the base station at last for detection. The proposed method works as follows:

(a) *Network assumption*: The network is assumed to have a set $S = \{s_1, s_2, \ldots, s_n\}$ to form n clusters.
(b) *Feature collection*: A set of identified traffic features are extracted from each traffic flow.
(c) *Data preprocessing*: The properties of the collected data are quite different and different characteristics have different metrics. So each property value should be normalized.

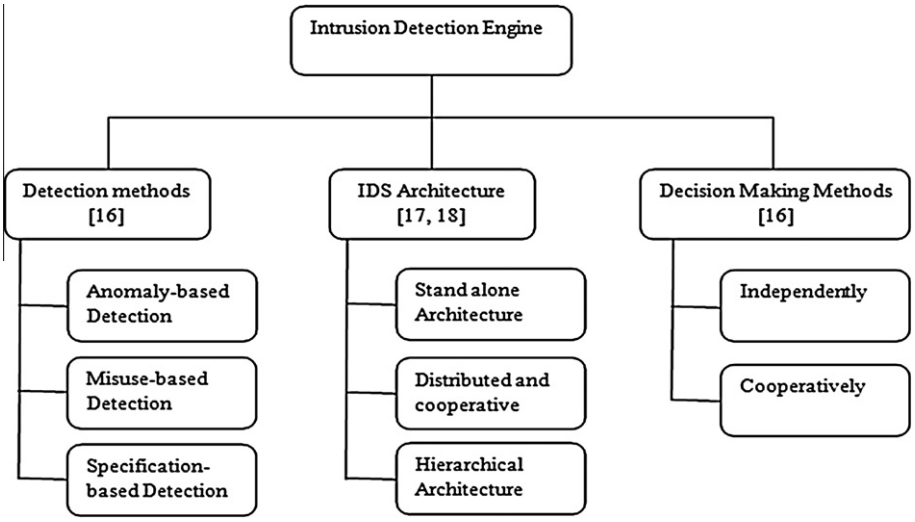


Figure 1 IDS engine classification tree.

(d) *Anomaly detection using Fuzzy C-Means*: Intrusion detection is executed at the base station using FCM algorithm as explained in Fig. 2.

Although this algorithm performs a preliminary clustering, it has a problem arising when the sample pool of data is not ideal. So Fuzzy C-Means algorithm should be improved to avoid this problem as shown in Eqs. (1)–(3):

– The constraint condition should be relaxed to

$$\sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik} = n \tag{1}$$

– Now we have the cluster centers as

$$V_i = \frac{\sum_{k=1}^{n}(u_{ik})^m x_k}{\sum_{k=1}^{n}(u_{ik})m} \tag{2}$$

– The membership value of U_m to

$$u_{ik} = n \times \frac{1}{\sum_{j=1}^{c}\sum_{l=1}^{m}\left(\frac{d_{ik}}{d_{jl}}\right)^{\frac{2}{m-1}}} \tag{3}$$

3.2. A hybrid detection using Back-Propagation Neural Network

A hybrid intrusion detection system (HIDS) combining both anomaly and misuse detection methods in a heterogeneous cluster-based WSN was introduced in [20]. Due to the heterogeneous nature of this cluster-based WSN, the capability of cluster heads is greater than other common nodes to accommodate the processing required for the detection process and decrease the consumption of energy. The proposed model is installed in each cluster head for intrusion detection of its entire cluster using information aggregated from nodes within the cluster. This HIDS consist of three models as mentioned below, the relation between these models is shown in Fig. 3.

(a) *Anomaly detection model*: Used to filter a large number of packet records by adopting a rule based method, using a set of expert defined rules to analyze the packets and distinguish which ones are abnormal
(b) *Misuse detection model*: Used for further detection of abnormal packet detected by anomaly model using Back Propagation Neural Network (BPN). The established flow chart of misuse detection model is shown in Fig. 4.
(c) *Decision making model*: Integrates the output of anomaly and misuse models and determines if an intrusion occurred or not based on a given set of predefined rules. Finally, the output of decision model is reported to the administrator at base station.

3.3. Distributed detection using Agglomerative Clustering

In [21], an anomaly detection technique based on distributed Agglomerative Clustering approach was proposed, whose main goal is to identify faulty nodes with low computational overhead, besides reducing the communication overhead through using hierarchical architecture which save energy. Being distributed, the algorithm operates in two different levels: local (at leave nodes) and global (at central nodes). At the lowest level of the hierarchy, each sensor apply local clustering to remove anomalous from its collected readings and send the cluster summaries to its next higher level,then cluster heads collect these summaries and perform clustering till reaching the base station.

This detection method use Agglomerative Clustering algorithm to create clusters then the determination of anomalous and sparse clusters depends on a novel formulation considering two properties of the cluster: cluster density and distance from other clusters. For detecting the intruders, each node at each level performs four steps as explained below:

Table 1 Strengths and weaknesses of the studied IDSs.

	Strengths	Weaknesses
Detection using Fuzzy C-Means clustering	– Improved FCM has better robustness than FCM algorithm – Achieves acceptable clustering results at the absence of wild point – Clustering results are less sensitive to the number of predetermined clusters by relaxing the restrictive condition of membership	– Increased communication overhead – Number of clusters created by improved FCM is more than created by FCM, affecting time complexity that depends on the number of clusters
A hybrid detection using Back-Propagation Neural Network	– Increased detection accuracy as a result of hybridization – Reduced energy consumption through making detection at cluster heads – Ability to identify type of attack	– Central point of failure at cluster heads – Increased communication overhead between sensor nodes and cluster head – There must be an efficient way to select relevant features – There must be a learning method for anomaly rule-base instead of expert experience
Distributed detection using Agglomerative Clustering	– Efficiency, scalability and energy saving – Low communication overhead	– Dependence of survival score determined by user – High computation complexity – Low detection accuracy
Detection using Multi-agent and Refined Clustering	– Flexible, easy to expand and energy saving through adopting multi-agent and different strategy of detection for common node and cluster heads – No central point of failure as each node maintains its own management agent	– Increased communication overhead needed for transferring full data records between nodes – Increased computation complexity needed for SOM – Dependency on cluster variance; detection rate is reduced if the variance of the cluster representing intrusion is large
Detection using Self Organizing Map and Discrete Wavelet Transform	– High detection accuracy – Reduced computation overhead	– Central point of failure and suspect -able to various types of attacks as anomaly detection is accomplished only on base station – Increased communication overhead – There is no management plane for re-electing the cluster head in case of its failure
Distributed detection using Naïve Bayesian classifier	– High detection accuracy and low false positive rate – Low computation complexity	– Central point of failure as anomalous detection is accomplished only at cluster heads – Increased communication overhead required for sending full data from common nodes to cluster heads – There must be a learning method for misuse detection instead of rules defined by expert
Detection using Support Vector Machine	– High detection accuracy through combination between SVM classifier and signature-based detection – Reduced energy consumption by transmitting support vectors between nodes instead of all captured data – No central points of failure since all nodes have the same capability of detection	– There must be an efficient way to select relevant features instead of delete one at a time and rank the important one – There must be a learning method for misuse detection instead of rules defined by experts

(a) *Generate partitions*: Using a hierarchical clustering algorithm similar to the fixed-width algorithm to cluster the data and treat each cluster as a separate partition. The algorithm computing candidate partitions is illustrated in Fig. 5.

(b) *Compute inter cluster distance for each cluster in the cluster set*: For each cluster C_i, compute inter cluster distance CD_i using Euclidean distance, cluster size $Count_i$, and the average inter cluster distance ICD_i as shown in Fig. 6.

(c) *Identify the candidate partitions containing outliers*: It identifies candidate partitions that contain outliers depending on two criteria:
   – If inter cluster distance ICD_i is more than a standard deviation away from the overall average ICD.
   – If the cluster size is more than median absolute deviation smaller than the median count.

(d) *Determining anomalous clusters from candidate partitions*: In this step, the anomalous clusters are computed from the candidate partitions. For each candidate parti-

Input:

$X = \{x_1, x_2, \ldots \ldots, x_n\}$	A limited set belong to p-dimensional Euclidean space R_p.
$d_{ij} = \|x_i - v_j\|$	The distance between x_i and v_j , ,
U	The fuzzy c-classified matrix of finite set
V	The collection of X cluster centers
c	The clustering number, $1 < c < n$
m	The weight exponential, $m \in (1, +\infty)$
ε	A given tolerance
n	Number of input patterns
t	Step number

Output:

The minimum values of U and V.

Step 1. Initialize n, m, c, t = 0, $V^{(t)} = \{v_1, v_2, \ldots., v_c\}$

Step 2. Compute $U^{(t)}$ and $U^{(t+1)}$, $\forall$ k, i (k = 1, 2, …, n, j = 1, 2, …., c)

If $d_{kj} \neq 0$ then $u_{kj} = 1/\sum_{i=1}^{c} \left[\frac{d_{kj}}{d_{ki}}\right]^{2/m-1}$

Else if $j = i$ then $u_{kj} = 1$

Else if $j \neq i$ then $u_{kj} = 0$

End If

Step 3. Compute $V^{(t+1)}$ according to $U^{(t)}$ and $U^{(t+1)}$

$v_i = \sum_{k=1}^{n} u_{ki} x_k / \sum_{k=1}^{n} u_{ki}$ i = 1, 2, ……., c

Step 4. Compute matrix norm to compare $V^{(t)}$ and $V^{(t+1)}$

If $\|V^{(t+1)} - V^{(t)}\| < \varepsilon$ then cease operations

Else $t = t + 1$ and return to step 2

End If

Figure 2 Fuzzy C-Means clustering algorithm.

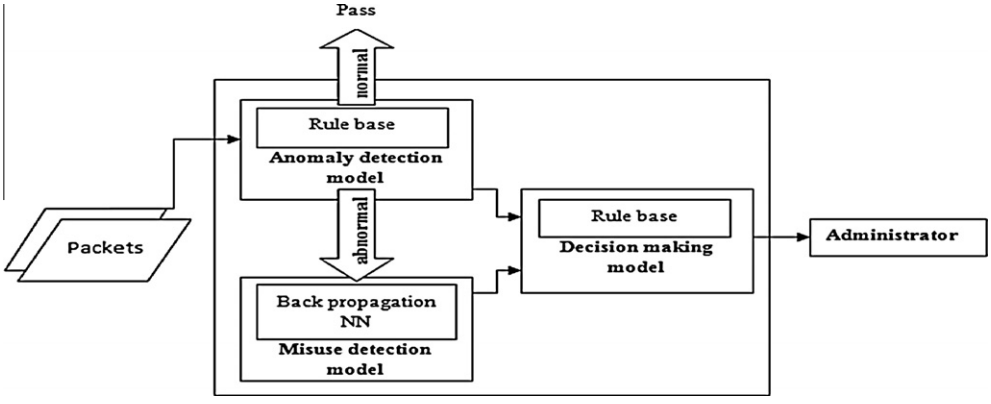


Figure 3 Architecture of Back Propagation based IDS.

tion, when the criteria mentioned in previous step coincide except that the survival score exceeds L (a predefined value), then the clusters are promoted as anomalous clusters. The algorithm for computing anomalous partitions is illustrated in Fig. 7.

3.4. IDS using Multi-agent and Refined Clustering

In [22], the author's proposed a hierarchical architecture combined with both cooperative decision making technique and anomaly-based IDS. The proposed detection method is called refined clustering. The word ''refined'' refers to the way of clustering, since it applies two cascaded degrees of clustering, Self-Organizing Map (SOM) neural network as unsupervised clustering technique to roughly cluster the samples, followed by the supervised K-means clustering technique to refine clustering of the previous stage, resulting in a number of cluster heads and member nodes.

This model employs the agent strategy; it adopts multiple agents to achieve different modules of intrusion detection. Four kinds of agents are installed on each node to cooperate. Each node will execute different operations of detection according to its place (cluster header or member node). Finally, they collaborate with each other to detect attacks. The agents installed on each are divided into four categories depending on their role.

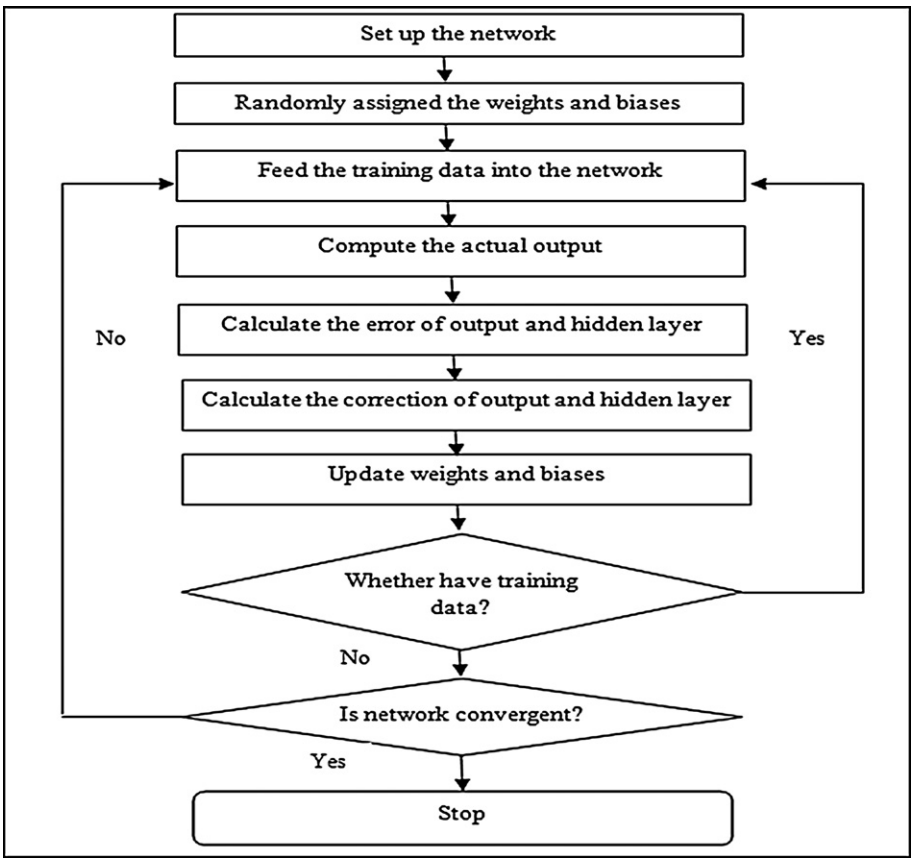


Figure 4 The established flow chart of misuse detection model.

```
Input:
   w          Cluster radius
   X          Set of incoming patterns
   K          Cluster centers
   counter_i  Number of points in cluster i
   counter    Number of clusters
   centroid_i Center of cluster i
   centroid   Final cluster centers
   d_j        Euclidean distance
   added      Does pattern belongs to a cluster

Output:
   Centroid, Counter

Step 1. Initialize w
Step 2. For (j=1: numPoints(X))
          // Compute distances
          d_j=dist (x_j, X)
        End For
Step 3. For (i=1: numPoints (K))
          Initialize counter_i=0;
        End For
Step 4. Initialize
          Newclusters=0;
          Newcentroid=0;
          Newcounter=0;
Step 5. For ( j=1:numPoints(X))
          Set Added <- false
          For (l=1: numPoints (K))
            If ((abs (d_j-k_l)) <=w)
                // add pattern x_j to cluster l
                Insert x_j to 1th cluster C_l
                // increase number of clusters
                Update counter=counter+1
                // update center of cluster l
                Update centroid_l=mean (C_l)
                Added <- true
                Break;
            End If
          End For
          For (l=1: numPoints (Newcluster))
            If ((abs (dj-newclusterl)) <=w)
              Insert xj to the 1th cluster of Newcluster
              Update Newcounter_l=Newcounter_l+1
              Update Newcentroid_l=mean(Newcluster_l)
              Added <- true
              Break;
            End If
          End For
          If (added<-false)
            Newcluster=Newcluster+1
            Newcentroid (Newcluster) =xj
            Newcounter (Newcluster) =1
            Add xj to Newcluster (Newcluster)
          End IF
          Return  (Centroid+Newcentroid),
                  (Counter+Newcounter))
        End For
Step 6. Return (Centroid, Counter)
```

Figure 5 The algorithm computing candidate partitions.

Input:
Nc The total number of clusters
$d(c_i,c_j)$ The Euclidean distance between c_i ,c_j
K The number of nearest neighbor clusters
CD_i Inter cluster distance of cluster c_i
ICD_i The average inter cluster distance of c_i

Output:
CD_i, ICD_i

For each cluster c_i, $1< i, j < Nc$
$CD_i = \{d(c_i,c_j),\ i,j = 1 \ldots \ldots Nc,\ j \neq i\}$
$ICD_i = \frac{1}{k} \sum_{j=1\neq i}^{k} d(c_i,c_j)$ *when* $k \leq |c| - 1$
End For

Figure 6 The algorithm computing inter cluster distance for each cluster.

Input:
ICD_i () Average inter cluster distance of clusters
$Counter_i$ () Set of count of data points in clusters
C () Set of Cluster centroids (c_1, c_2, ..., c_n)
O Set of outlier clusters
Count Set of count of data points of outlier clusters

Output:
O, Count

Step 1. Initialize i=1
Step 2. While C list is not empty do
// Initialize counter
cnt = 1
If ($Counter_i$ < mean (counter) + MAD (counter))
If (ICD_i > (mean (ICD) + SD(ICD)))
$O_{cnt} = C_i$
$Count_{cnt} = Counter_i$
Cnt = cnt + 1
End If
i++
End If
End While
Step 3. Return (O,Count)

Figure 7 The algorithm computing anomalous partitions.

(a) *Sentry agent*: It is responsible for monitoring all the activities of the node. If it is located in common nodes it monitors the information of cluster head, and if located in cluster head it collect the data from the whole cluster and the information from the neighboring cluster headers.
(b) *Analysis agent*: It is considered the most important agent in the model, since it is responsible for receiving and analyzing the data collected by sentry agents and judging if an intrusion happens. This function is done by adopting two clustering levels; a rough detection (using SOM neural network) followed by a refined clustering level (using K-means clustering technique).
   – *First stage: SOM clustering*. First the network was created then it will be trained. After training the neural network, we get the U-map (Unified Matrix map), from which we can notice the nodes representing normal behavior and the others representing abnormal. From the recognition of theses nodes we can get number of clusters and cluster centers.
   – *Second stage: K-means clustering*. From the output of previous stage, we can refine the clustering using the algorithm shown in Fig. 8.
(c) *Response agent* It will be activated only when the analysis agent discovers an intrusion, and takes the corresponding response measures under specific circumstances.
(d) *Management agent*: It is installed at each node and is responsible for managing, maintaining, and harmonizing the functions of the three other agents in the node.

The detection mechanism are divided into two scenarios, monitoring of cluster headers to common nodes and monitoring of common nodes to cluster headers depending on a set of predefined set of attributes as explained in the paper.

3.5. IDS using Self Organizing Map and Discrete Wavelet Transform

In [23], an anomaly-based IDS using a combination of Discrete Wavelet Transform (DWT) and a competitive learning SOM neural network was proposed, in order to detect anomalies accurately. The reason of this combination comes from ability of the SOM neural network to extract statistical regularities from the input data vectors and encode them in weights without supervision. But SOM requires processing time ascending with the input data size, therefore, the DWT is applied first to gather sufficient features of input data then fed these features to SOM network.

The proposed model is installed in the base station that operates as the following stages:

(a) *Data preprocessing using DWT*: In each sensor node DWT is applied to its reading. DWT separates the data signal into detail coefficients and approximate coefficients through signal decompositions process, described in:

$$a_{j+1}^{DWT}(k) = \sum_{n} h_0(n-2k) a_j^{DWT}(k) \tag{4}$$

$$d_{j+1}^{DWT}(k) = \sum_{n} g_0(n-2k) d_j^{DWT}(k) \tag{5}$$

where a_{j+1}^{DWT}, d_{j+1}^{DWT} represent approximate and detailed coefficients, respectively, both are convolved with h_0 and g_0 representing the wavelet function and scaling function respectively, n is the time scaling index and k is the frequency translation index for wavelet level j.

(a) *Un supervised Clustering using SOM*: In the base station, SOM is built for unsupervised clustering. The basic SOM consists of a regular grid of map units or neurons, each neuron denoted by i has a set of layered neighboring neurons and maintain a weight vector m_i. The SOM network is trained iteratively using the coefficients obtained from previous stage representing SOM data set.
(b) *Anomaly detection*: After training the SOM using wavelet coefficients, anew observation data set can be considered abnormal if the distance between the weight vector of the winning neuron and the new state vector given by:

Input:
- n Number of readings
- c Number of clusters
- $\bar{X}$ Set of cluster centers
- X Set of readings
- dx_i The distance between x_i and the center of its cluster.

Output:
Final cluster centers

Step 1. Set initial cluster centers
$\bar{X}_j = X_j$, j = 1,……,c
Step 2. Classify each pattern about the cluster centers
For each $x_i \in \{x_1, x_2, \ldots\ldots, x_n\}$
For each $\bar{X}_j$ $(j = 1,2, \ldots., c)$
If $(dx_i > distance(x_i, \bar{X}_j))$
$dx_i = distance(x_i, \bar{X}_j)$
$x_i \in cluster\ j$
End If
End For
Recalculate the center of each cluster
End For
Step 3. Repeat the above steps till the center of each cluster doesn't change

Figure 8 K-means clustering algorithm.

$e^{\mu} = \|x^{new} - m_c^{\mu}\|$, is greater than a certain percentage $p = 1 - \alpha$ of the distances in the distance distribution profile.

3.6. Distributed detection using Naïve Bayesian classifier

In [24], the authors proposed an anomaly detection framework for WSN using agent-based learning and distributed data mining technique. First, the network is structured into two-tier hierarchical topology, with different capabilities for sensors at each tier. According to these capabilities, nodes are divided into two types: Forwarding nodes; for activity sensing and data forwarding to higher-tier nodes, and Cluster heads; responsible for collecting and processing data from lower-tier.

In this framework, sensor nodes sense the action and then report to their corresponding cluster head to be processed then cluster heads send sensed data file to base station. The data collected at cluster heads may contain erroneous or wrong information (anomaly), so before sending the data file to base station, cluster heads need to detect

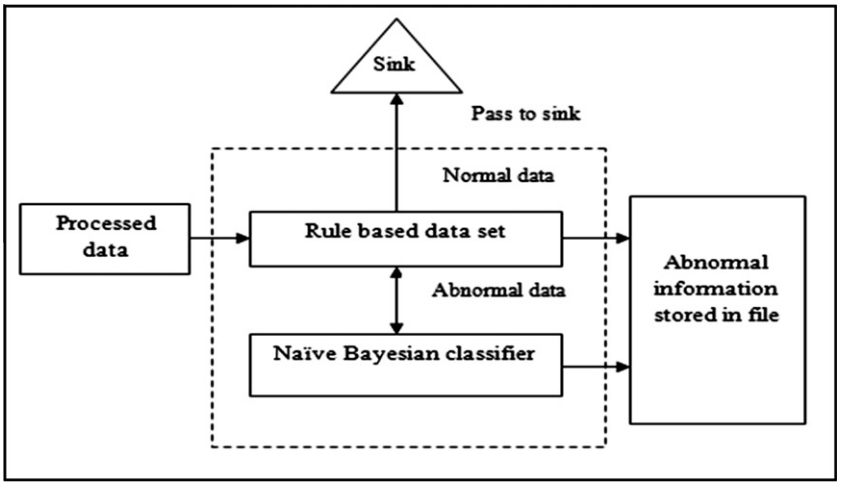


Figure 9 Internal agent architecture in Naïve Bayesian classifier based IDS.

Input:
Processed n data set files
Output:
Percentage of abnormal data stored in a file

Step 1. Read processed data set file for each cluster head;
Step 2. Call naïve Bayesian classifier program for training the classifier for abnormal event detection, and store that information into a test file;
Step 3. Test this file with the classifier model and write them to an output file
Step 4. Calculate the percentage of abnormal data;

Figure 10 Naïve Bayesian classifier algorithm.

anomalous and remove them. The distributed detection process is accomplished through the agent residing between base station and each cluster head. This agent performs detection using two modules as discussed below and shown in Fig. 9.

(a) *Misuse detection module*: This module compares the given data with a predefined rules using rule-based method.
(b) *Anomaly detection*: It is activated only if anomalous is detected by previous module to further detect using Naïve Bayesian classifier. The algorithm explaining how anomaly detection works is shown in Fig. 10.

3.7. Detection based on Support Vector Machine

In [25], Sedjelmaci and Feham proposed a novel hybrid IDS for cluster-based heterogeneous WSN. This hybridization involves anomaly detection using Support Vector Machine (SVM) and misuse detection using rule-based data set. In order to prolong the network life time, firstly the hierarchical architecture divides the network into a set of clusters, each one having cluster head. Secondly, for each cluster only a predefined number of nodes N employs the IDS, $N = 1.6r^2d$, where d refers to the network density and r to the communication range. This novel IDS comprises three modules as mentioned below and shown in Fig. 11.

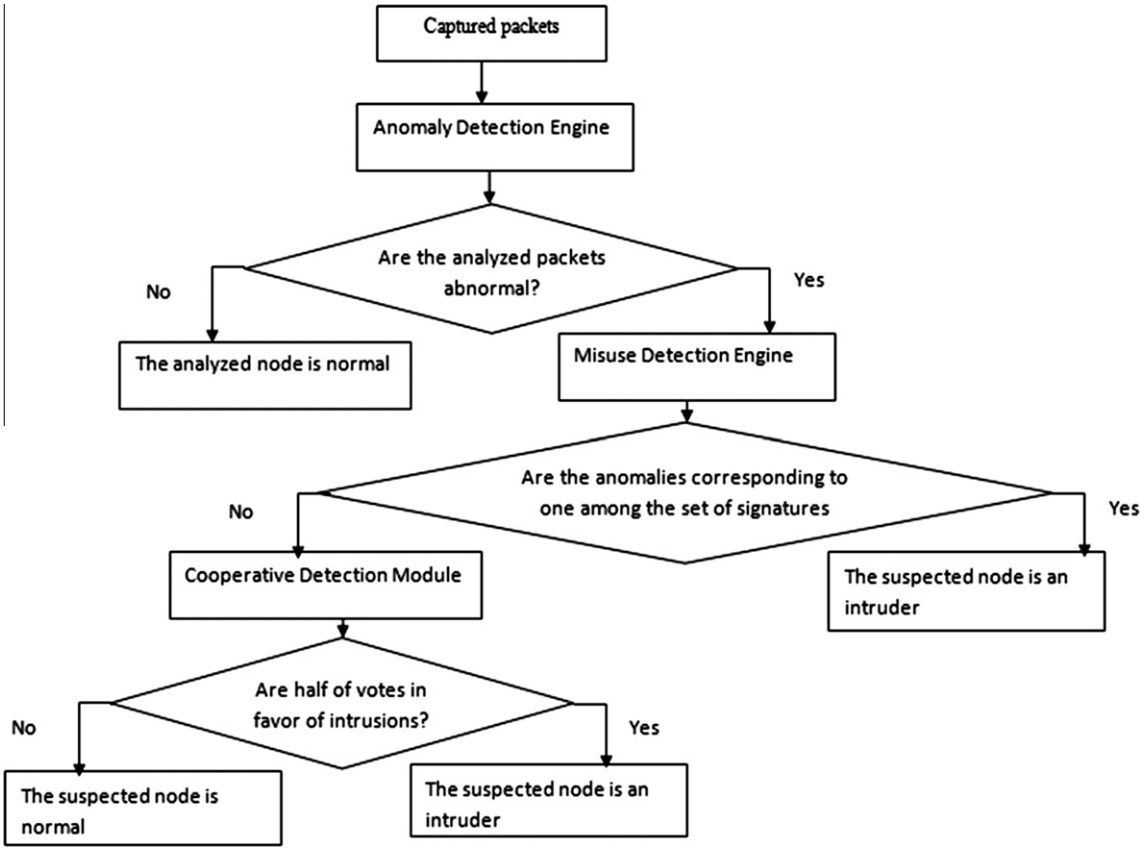


Figure 11 The flow chart of IDS framework in SVM-based IDS.

(a) *Data collection module*: From which IDS nodes gather the packets within their radio range.
(b) *Hybrid intrusion detection module*: That involves two detection engines, anomaly and misuse detection engine.
   – *Anomaly detection engine based on SVM*. The detection is accomplished in two stages. The first stage; involves training the SVM locally at each IDS node, then computes support vectors that will be sent to adjacent IDS node situated in the same cluster. When a node receives these vectors it combines them with its own and so on, until all IDS nodes in the same cluster reaches the same trained SVM. Afterwards, all cluster heads exchange their own vectors, and then communicate these vectors to their IDS. As a result of this cycle, a global support vectors resides in each IDS node. These global vectors are then used in the SVM testing stage.
   – *Misuse detection engine*. It is activated only if abnormality is detected by anomaly phase, using a set of predefined attack signatures for further checking. If a match occurs, then intrusion exists. Otherwise, the third module is lunched.

(c) *Cooperative detection module*: Performs a voting mechanism within the same cluster in order to make a better decision about the suspected nodes. If attack exists, then attack signatures are updated with this novel attack.

4. Simulation and results

To compare the previously discussed IDSs, a series of experiments were conducted to simulate and evaluate each technique. The algorithms simulations are done in Matlab and the standard KDDCup'99 intrusion detection dataset [26] is used for WSN simulation. The features of the standard KDDCup'99 dataset consist of 34 types of numerical features and 7 types of symbolic features, according to different properties of attack. This dataset includes many attack behaviors, classified into four groups: Probe, Dos, U2r and R2l. kddcup. data_10_percent.gz data set is used as training and testing dataset in all experiments. For training, 10,000 records are used with 5000 normal record and 5000 abnormal (including Dos and Probe). Moreover, testing is accomplished many times using 1000 record per time including normal and abnormal (including new attacks R2l and U2r) records, such that decreasing normal percentage and increasing abnormal percentage over time.

The main objective is to compute the accuracy of anomaly detection process for each technique based on a set of different evaluation metrics. The standard IDS evaluation metrics are classified into two categories [27]; the first one, measures the detection performance and the second is assigned to measure the capability of the IDS to provide security to network. The performance metrics include: (i) Technique processing overhead; measured as computational complexity and resources usage [27]. (ii) Communication overhead, and (iii) fair distribution of the processing workload among the network nodes. On

Table 2 Comparison among the studied IDSs on the basis of performance evaluation metrics.

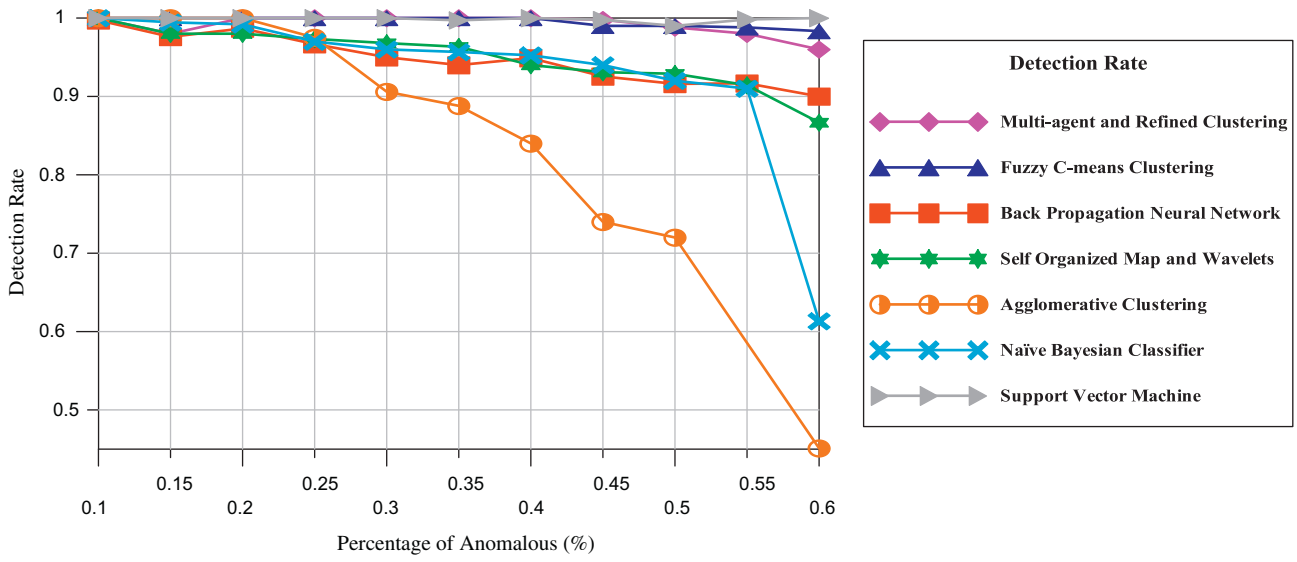
IDS methodology	Processing overhead		Communications overhead	Un fair workload distribution
	Time complexity	Space complexity		
Detection using Multi-agent and Refined Clustering	*SOM:* [28] $O(NDR)$ *K-means:* [30] $O(NCID)$	*SOM:* [29] $O(N^2)$ *K-means:* [30] $O((N + C)D)$	Full data records are exchanged between nodes	N/A
Detection using Fuzzy C-Means clustering	$O(NDC^2I)$ [31]	$O(ND + NC)$ [31]	A set of predefined features extracted from packets are exchanged	N/A
Detection using Back Propagation Neural Network	$O(R^2T)$ *per cycle* [32]	$O(RT)$ *per Cycle* [32]	Full data records are exchanged between nodes	Cluster heads are unfairly overloaded
Detection using Self Organizing Map and Wavelets	*Wavelet:* [33] $O(N)$ *SOM:* $O(NDR)$	*Wavelet:* [33] $O(N)$ *SOM:* $O(N^2)$	Full data records are transferred to base station	N/A
Detection using Agglomerative Clustering	$O(N^3)$ [30]	$O(N^2)$ [30]	Only clustering summaries are exchanged between nodes	N/A
Detection using Naïve Bayesian classifier	$O(ND)$ [34]	$O(DVC)$ [34]	Full data records are transferred to cluster heads	Cluster heads are unfairly overloaded
Detection using Support Vector Machine	$O((ND)^3)$ [35]	$O((ND)^2)$ [35]	Only support vectors are exchanged between nodes	Some common nodes are unfairly overloaded

the other hand, three different security metrics are used to evaluate each technique namely, detection rate D_R, false alarm rate F_A and false positive rate F_P, as defined in Eqs. (6)–(8) [21].

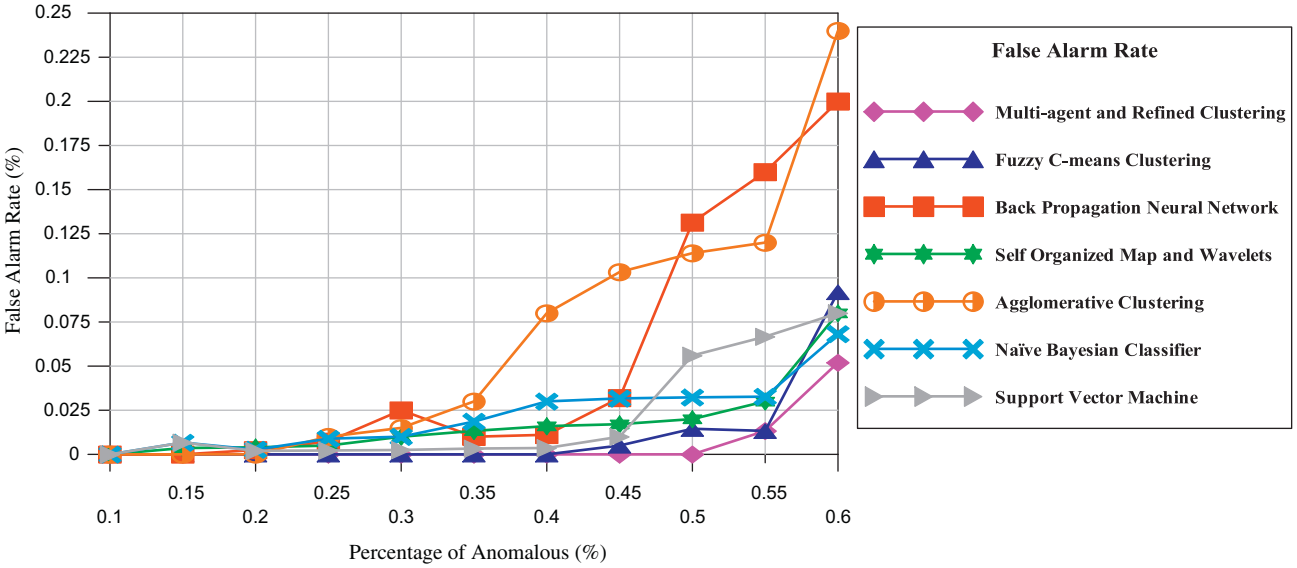
- *Detection rate* (D_R): is defined as the ratio between the numbers of correctly detected anomalous measurements to the total number of anomalous measurements.

Table 3 General notations used in Table 2.

Notation	Meaning	Notation	Meaning
N	Number of data records(vectors) at time t	T	Maximum number of activation changes
C	Number of clusters/classes	V	Values for each feature
R	Total number of neurons	I	Maximum number of iterations
D	Dimension of input vectors (n. of features)		

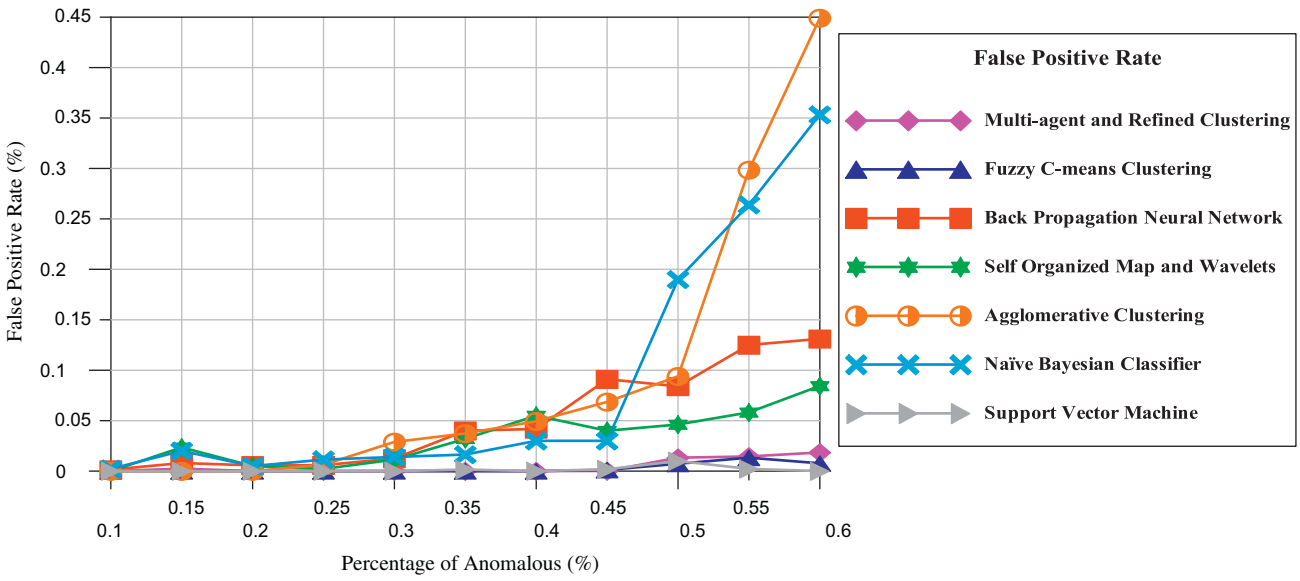


(a) Detection Rate (D_R)



(b) False Alarm Rate (F_A)

Figure 12 Security evaluation of the studied IDSs.



(c) False Positive Rate (F_P)

Fig. 12 (*continued*)

$$D_R = \frac{\text{Number of correct classified anomalous mesurements}}{\text{Total number of anomalous measurements}} \times 100\% \quad (6)$$

- *False alarm rate* (F_A): is the ratio between the numbers of normal measurements that are incorrectly misclassified as anomalous to the total number of abnormal measurements.

$$F_A = \frac{\text{Number of misclassified normal measurements}}{\text{Total number of anomalous measurements}} \times 100\% \quad (7)$$

- *False positive rate* (F_P): is the ratio between the numbers of abnormal measurements that are incorrectly misclassified as normal to the total number of normal measurements.

$$F_P = \frac{\text{Number of misclassified abnormal measurements}}{\text{Total Number of normal measurements}} \times 100\% \quad (8)$$

A successful anomaly detection algorithm should achieve high D_R, low F_A and low F_P.

4.1. Performance evaluation

In this section, the studied IDS techniques are tested and compared according to the standard performance evaluation metrics. The main objective is to specify a successful anomaly detection algorithm for a hierarchical WSN architecture that achieves low processing and communication overheads while fairly distributes the workload on the nodes as possible. The comparative performance evaluation results are shown in Table 2 with its notations declared in Table 3. In addition, it can be noticed that:

1. Anomaly-based IDSs that employ different neural networks techniques introduce time and space complexity as a nonlinear function in many variables. Therefore, those techniques exhaust resources.
2. Detection based on Wavelet and Bayesian classifier shows superiority in terms of time and space complexity. Since their complexities are either linear functions with its inputs or a few number of variable are involved.
3. Regarding communication overhead, IDS based on Support Vector Machine shows superiority in term of communication overhead. Since instead of exchanging all the data records, they exchange the main cluster centers (support vectors) only.
4. Most of them fairly distribute workload among nodes, prolonging life time of the network. Except those employing Back Propagation, Naïve Bayesian and Support Vector Machines.
5. Generally, classification based on proper feature selection is one of the important factors which affect the performance of IDS.

4.2. Security evaluation

Recalling that, the successful anomaly detection algorithm should achieve high D_R, low F_A, and low F_P, as defined in Eqs. (6)–(8) respectively. Results of security evaluation metrics are compared through curves shown in Fig. 12a–c respectively. The x-axis specifies the percentage of anomalous attack which refers to the ratio of the number of anomalous attack to the total number of measurements collected at the sensors. The y-axis specifies the security evaluation metric. We notice that:

1. As seen in Fig. 12a, most of the studied algorithms successfully detect the anomalous attacks with a very close performance ratio up to 25% of increase in the anomalous data. After that, some of them rapidly deteriorate as the percentage of anomalous increase, while others show robustness against the increase of anomalous percentages.
2. The superiority of SVM intrusion detection technique, FCM clustering and Multi-agent Refined Clustering IDS, respectively, can be clearly observed. They show robustness against the increase of anomalous till 60%.

3. Regarding F_A and F_P ratios, the comparative curves are shown in Fig. 12b and c respectively. The results of comparisons enforce the detection rate results. Since the performance ratios of the studied techniques have low F_A and low F_P till 35% of increase in the percentage of anomalous.
4. IDS based on SVM classification, FCM clustering, and refined clustering techniques respectively, shows robustness against increase in the percentage of anomalous in terms of F_A and F_P.

5. Conclusion and future work

This paper has evaluated and compared the latest anomaly based IDS applied for a hierarchal WSN. From the obtained results it can be concluded that: due to the WSN nature, it is difficult to distinguish the abnormal behavior from the normal one especially for uneven network traffic patterns. Therefore, it is highly recommended to depend on the data mining and artificial intelligence techniques. Since it has a dynamic ability to gather similar traffic patterns in a cluster and isolate the uneven ones. In addition, feature selection is one of the important factors which affect the performance of IDS. Also, the proper selection of clustering parameters can refine the isolation and enforce the decision making process. The decision of choosing an optimum IDS is a trade-off process between security and performance metrics.

For future work, it can be remarked that designing a new anomaly based IDS is a true challenge. Since, it must satisfy the performance aspects as well as the security aspects. Also, a newly feature selection methods can be adopted. In addition, relying on a newly data mining technique rather than traditional classifiers based on neural networks to properly select the clustering parameters can enhance the refining process.

References

[1] García-Hernández CF, Ibargüengoytia-González PH, García-Hernández J, Pérez-Díaz JA. Wireless sensor networks and applications: a survey. IJCSNS Int J Comput Sci Network Security 2007;7(3):264–73.

[2] Wang Y, Attebury G, Ramamurthy B. A survey of security issues in wireless sensor networks. IEEE Commun Surveys Tutorials 2006;8(2):2–23.

[3] Al-Karaki JN, Kamal AE. Routing techniques in wireless sensor networks: a survey. IEEE Wirel Commun 2004;11(6):6–28.

[4] Sang Y, Shen H, Inoguchi Y, Tan Y, Xiong N. Secure data aggregation in wireless sensor networks: a survey. In: The proceeding of the 7th international conference on parallel and distributed computing, applications and technologies; 2008. p. 315–20

[5] Xiao Y, Rayi VK, Sun B, Du X, Hu F, Galloway M. A survey of key management schemes in wireless sensor networks. Comput Commun 2007;30:2314–41 [Special issue on security on wireless ad hoc and sensor networks].

[6] Palpanas T, Papadopoulos D, Kalogeraki V, Gunopulos D. Distributed deviation detection in sensor networks. SIGMOD Rec 2003;32(4):77–82.

[7] Subramaniam S, Palpanas T, Papadopoulos D, Kalogeraki V, Gunopulos D. Online outlier detection in sensor data using non-parametric models. In: The proceeding of the 32nd international conference on very large data bases; September 2006. p. 187–98.

[8] Zhang Y, Yang W, Kim K, Park M. Inside attacker detection in hierarchical wireless sensor network. In: The proceeding of 3rd international conference on innovative computing, information and control; 2008. p. 594.

[9] Tiwari M, Arya KV, Choudhari R, Choudhary KS. Designing intrusion detection to detect black hole and selective forwarding attack in WSN based on local information. In: The proceeding of 4th international conference on computer sciences and convergence information technology; 2009. p. 824–8.

[10] Moshtaghi M, Rajasegarar S, Leckie C, Karunasekera S. Anomaly Detection by Clustering Ellipsoids in Wireless Sensor Networks. In: The proceeding of 5th international conference on intelligent sensors, sensor networks and information processing; 2009. p. 331–6.

[11] Khanna R, Liu H, Chen H. Reduced complexity intrusion detection in sensor networks using genetic algorithm. In: The proceeding of IEEE international conference on communications; 2009. p. 598–602.

[12] Agah A, Das SK, Basu K. A non-cooperative game approach for intrusion detection in sensor networks. In: The proceeding of IEEE 60th vehicular technology conference, vol. 6; 2004. p. 2902–6.

[13] Agah A, Das SK, Basu K, Asadi M. Intrusion detection in sensor networks: a non-cooperative game approach. In: The proceeding of 3rd IEEE international symposium on network computing and applications; 2004. p. 343–6.

[14] Su C, Chang K, Kuo Y. The new intrusion prevention and detection approaches for clustering-based sensor networks. In: The proceeding of IEEE wireless communications and networking conference, vol. 4; 2005. p. 1927–32.

[15] Zia T, Zomaya A. A security framework for wireless sensor networks. In: IEEE sensors applications symposium; 2006. p. 49–53.

[16] Krontiris I. Intrusion prevention and detection in wireless sensor networks. Dissertation; 2008.

[17] Mamun MSI, Kabir AFM. Hierarchical design based intrusion detection system for wireless ad hoc sensor network. Int J Network Security Appl (IJNSA) 2010;2(3):102–17.

[19] Wang T, Liang Z, Zhao C. A detection method for routing attacks of wireless sensor network based on fuzzy C-means clustering. In: The proceeding of 6th international conference on fuzzy systems and knowledge, discovery; 2009. p. 445–9.

[20] Yan KQ, Wang SC, Liu CW. A hybrid intrusion detection system of cluster-based wireless sensor networks. In: The proceeding of international multi conference of engineers and computer scientists, vol. 1; 2009.

[21] Chitradevi N, Palanisamy V, Baskaran K, Prabeela S. Efficient distributed clustering-based anomaly detection algorithm for sensor stream in clustered wireless sensor networks. Eur J Sci Res 2011;54(4):484–98.

[22] Huai-bin W, Zheng Y, Chun-dong W. Intrusion detection for wireless sensor networks based on multi-agent and refined clustering. In: The proceeding of international conference on communications and mobile computing; 2009. p. 450–4.

[23] Siripanadorn S, Hattagam W, Teaumroong N. Anomaly detection in wireless sensor networks using self organizing map and wavelets. Int J Commun 2010;4(3):74–83.

[24] Sa M, Nayak MR, Rath AK. A simple agent based model for detecting abnormal event patterns in a distributed wireless sensor networks. Int J Comput Sci Security (IJCSS) 2011;4(6):580–8.

[25] Sedjelmaci H, Feham M. Novel hybrid intrusion detection system for clustered wireless sensor network. Int J Network Security Appl (IJNSA) 2011;3(4).

[26] http://kdd.ics.uci.edu/databases/kddcup99/kddcup.

[27] Arora S, Barak B. Computational complexity: a modern approach. Cambridge University Press; 2009.

[28] Guan H, Turk M. The hierarchical isometric self-organizing map for manifold representation. In: The proceeding of IEEE conference on computer vision and pattern recognition; 2007. p. 1–8.

[29] Maiorana F. Performance improvements of a kohonen self organizing classification algorithm on sparse data sets. In: The proceeding of 10th WSEAS international conference on mathematical methods, computational techniques and intelligent systems; 2008. p. 347–52.
[30] Tan P, Steinbach M, Kumar V. Introduction to data mining: cluster analysis: basic concepts and algorithms. Pearson Addison-Wesley; 2006 [chapter 8].
[31] Hore P, Hall LO, Goldgof DB. Single pass fuzzy C means. In: The proceeding of IEEE international conference on fuzzy systems; 2007. p. 1–7.
[32] Rodríguez ES. The general back propagation Algorithm; 2005.
[33] Li T, Li Q, Zhu S, Ogihara M. A survey on wavelet applications in data mining. SIGKDD Explor Newslett 2002;4(2):49–68.
[34] Fleizach C, Fukushima S. A Naive Bayes classifier on 1998 KDD cup; 1998.
[35] Tsang IW, Kwok JT, Cheung P. Core vector machines: fast SVM training on very large data sets. J Machine Learn Res 2005;6:363–92.

Further reading

[18] Xie M, Han S, Tian B, Parvin S. Anomaly detection in wireless sensor networks: a survey. J Network Comput Appl 2011;34(4):1302–25.