Expert Systems with Applications 38 (2011) 15234–15243

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

An Integrated Intrusion Detection System for Cluster-based Wireless Sensor Networks

Shun-Sheng Wang, Kuo-Qin Yan *, Shu-Ching Wang *, Chia-Wei Liu

Chaoyang University of Technology, 168, Jifong E. Rd., Wufong Township, Taichung County 41349, Taiwan, ROC

ARTICLE INFO

Keywords:
WSN
Intrusion Detection System
Misuse detection
Anomaly detection

ABSTRACT

A Wireless Sensor Network (WSN) consists of many low-cost, small devices. Usually, as they are deployed to an open and unprotected region, they are vulnerable to various types of attacks. In this research, a mechanism of Intrusion Detection System (IDS) created in a Cluster-based Wireless Sensor Network (CWSN) is proposed. The proposed IDS is an Integrated Intrusion Detection System (IIDS). It can provide the system to resist intrusions, and process in real-time by analyzing the attacks. The IIDS includes three individual IDSs: Intelligent Hybrid Intrusion Detection System (IHIDS), Hybrid Intrusion Detection System (HIDS) and misuse Intrusion Detection System. These are designed for the sink, cluster head and sensor node according to different capabilities and the probabilities of attacks these suffer from. The proposed IIDS consists of an anomaly and a misuse detection module. The goal is to raise the detection rate and lower the false positive rate through misuse detection and anomaly detection. Finally, a decision-making module is used to integrate the detected results and report the types of attacks.

1. Introduction

Advances in wireless communication and miniature electronics have enabled the development of small, low-cost, low-power sensor nodes (SNs) with sensing, computation and communication capabilities. Therefore, the issues of Wireless Sensor Networks (WSNs) have become a popular subject for research (Akyildiz, Su, Sankarasubramaniam, & Cayirci, 2002). WSN is a non-infrastructure network, and through the mass deployment of SNs, a WSN is formed. However, because of the limited energy of sensors, the sensors will communicate with each other by multi-hop to reduce energy consumption of sensors (Akkaya & Youngish, 2005).

The major function of WSN is to collect and monitor the related information about a specific environment, such as business, military, healthy care, and environment surveillance (Akyildiz et al., 2002; Laerhoven1, 2004). The SNs detect the surrounding environment or the given target and deliver the data to the sink using wireless communication. The data are then analyzed to find out the state of the target. However, due to the design of their hardware, WSNs suffer from many resource constraints such as low computation capability, small memory and limited energy (Akyildiz et al., 2002; Mhatre, Rosenberg, Kofman, & Shroff, 2004; Onat & Miri, 2005).

Two of the most common topology of WSNs, Flat-based Wireless Sensor Network (FWSN) and Cluster-based Wireless Sensor Network (CWSN) (Bace & Mell, 2001), are shown in Figs. 1 and 2, respectively. However, a large amount of the information is generated by multi-hop communication and the energy consumption is raised in FWSN, such as SPIN (Heinzelman, Kulik, & Balakrishnan, 1999). CWSN is a popular network topology in WSN. In a CWSN, all SNs are clustered, and a cluster head (CH) is elected to manage the operation of its own cluster (Heinzelman et al., 1999; Lindsey & Raghavendra, 2002; Manjeshwar & Agrawal, 2001, 2002). CH should aggregate data from all SNs sensed from a specific target. Therefore, CWSN efficiently reduces the amount of information in the entire network. The advantages of CWSN are a decrease in energy consumption, an increase in the network scale, and a prolonged network lifetime. Many protocols of CWSN have been proposed, such as LEACH (Heinzelman, Chandrakasan, & Balakrishnan, 2000), TEEN (Manjeshwar & Agrawal, 2001), APTEEN (Manjeshwar & Agrawal, 2002), and PEGASIS (Lindsey & Raghavendra, 2002).

Because WSNs are composed of numerous low-cost and small devices, and usually deploy to an open and unprotected area, they are vulnerable to various types of attacks (Rajasegarar, Leckie, & Palaniswami, 2008; Wang, Attebury, & Ramamurthy, 2006; Zhenwei & Tsai, 2008). For example, when a WSN is applied to the battlefield, SNs are invaded by the enemy and destroyed. Thus, the security of the WSN needs to be considered. A prevention mechanism is used to counteract well-known attacks. It establishes a corresponding prevention method, according to the

* Corresponding authors. Tel.: +886 4 23323000x4218; fax: +886 4 23742337 (S.-C. Wang).

E-mail addresses: sswang@cyut.edu.tw (S.-S. Wang), kqyan@cyut.edu.tw (K.-Q. Yan), scwang@cyut.edu.tw (S.-C. Wang), s9614640@cyut.edu.tw (C.-W. Liu).

0957-4174/$ - see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2011.05.076

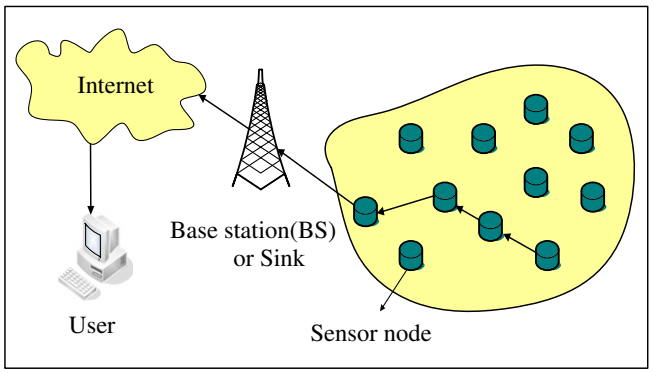

Fig. 1. Flat WSN.

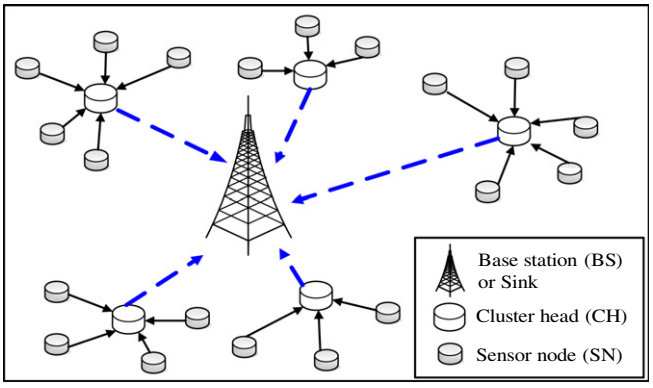

Fig. 2. Cluster-based WSN.

characteristics of an attack. However, prevention mechanisms cannot resist overall attacks. Therefore, the attacks need to be detected. An Intrusion Detection System (IDS) is used frequently to detect the packets in a network, and determine whether they are attackers. Additionally, IDS can help develop the prevention system through acquired nature of attacks.

The IDS acts as a network monitor or an alarm. It prevents destruction of the system by raising an alarm before the intruder starts to attack. The two major modules of intrusion detection include anomaly detection and misuse detection (Kemmerer & Vigna, 2002; Murali & Rao, 2005). Anomaly detection builds a model of normal behaviour and compares it with the detected behaviour (Kemmerer & Vigna, 2002; Rajasegarar et al., 2008). Anomaly detection has a high detection rate, but the false positive rate is also high. The misuse detection detects the attack type by comparing the past attack behaviour and the current attack behaviour. The misuse detection has high accuracy but low detection rate. Especially, the misuse detection cannot detect unknown attacks, which are not in the model base. Many researchers discuss a module of hybrid detection to have the advantages of both anomaly detection and misuse detection (Bace, Mell, 2001; Depren, Topallar, Anarim, & Ciliz, 2005; Kemmerer & Vigna, 2002). This combination can detect unknown attacks with the high detection rate of anomaly detection and the high accuracy of misuse detection. The Hybrid Intrusion Detection System (HIDS) achieves the goals of high detection rate and low false positive rate.

CWSN is the most general network topology in WSNs; hence, the mechanism of intrusion detection for a heterogeneous CWSN is discussed in this study. Since the works and the importance of each node are different, the probabilities of suffering attacks are also different. The sink has the most important role in WSN, because it has to collect the data from all CHs, and then the data are aggregated to the administrator to analyze. CHs are also important components; they have to aggregate the sensed data from all SNs in their own cluster. However, the works of SNs are just to sense the data in their region, so the damages after the attacks are only minor. Base on these concepts, it is clear that the sink and CHs are the most attacked targets by enemies, so the security of them needs more attentiveness. In addition, the capabilities of all sensors in CWSN are heterogeneous (Mhatre & Rosenberg, 2004; Wang, Wang, Xie, Wang, & Agrawal, 2008). The sink has to collect the data from all CHs in the whole network for administrator, so it needs better computing and more energy to work. CHs have more demanding works than other SNs, so the capability of CHs is better than other SNs. On the other hand, SNs are not as capable as CHs, because SNs just need to sense data in their region. Overall, the capability of the sink is better than CH, and CH is better than SN in CWSN.

In this study, an Integrated Intrusion Detection System (IIDS) is discussed in a heterogeneous CWSN. According to the different capabilities and probabilities of attacks on them, three separate IDSs are designed for the sink, CH and SN. For the sink, an Intelligent Hybrid Intrusion Detection System (IHIDS) is proposed, which has learning ability. The proposed IHIDS combines anomaly and misuse detection, and the goals of high detection rate and low false positive rate. The anomaly detection model can filter a large number of normal packets first, and then the abnormal packets are forwarded to the misuse detection model to identify the type of attacks. However, if the misuse detection model cannot identify the type of attack, it is then forwarded to the learning mechanism of IHIDS to learn the new type of attacks. For CHs, a HIDS is proposed, which has the same detection models with IHIDS, but there is no learning ability in HIDS. The goals of HIDS are to detect attacks efficiently and avoid resource wasting. However, HIDS would retrain the behaviour of new attacks, which have been detected and classified from IHIDS. For SNs, a misuse IDS is proposed; it uses the attack model to match the packets fast and then to find attacks. Because the resources of SN are less than other devices, such as the sink, CHs, a simple and fast detection method in SN is adopted to avoid overwork, and to save resources for the purpose of safety. In short, the IHIDS not only decreases the consumption of energy but also efficiently reduces the amount of information. Therefore, the lifetime of CWSN can be prolonged.

The remainder of this paper is organized as follows: Section 2 introduces the common attacks in WSN and the analytic tools of intrusion detection. In Section 3, the proposed methods and architecture of our research are introduced. The simulation results used to evaluate the performance of the proposed system are presented in Section 4. Finally, the conclusion and future work are discussed in Section 5.

2. Related work

In this section, the common attacks in WSN are introduced, and the analytic tools of intrusion detection used in our research are presented, including rule-based, BPN and ART.

2.1. Attacks in WSN

Attacks can be classified into two main categories, based on the objectives of intrusion (Su, Chang, & Kuo, 2007): (1) To destroy the connectivity of a network by sending malicious control messages. (2) To exhaust the link bandwidth of a network and to consume the energy of SNs. Therefore, the common attacks in WSN are analyzed and classified according to their features.

Table 1
The comparisons of different attacks in WSN.

Attack name	Behaviour	Number of attacks	Influence area
Spoofed, Altered, Replayed Routing Information	Route updating misbehaviour	Single	Local area
Select Forward	Data forwarding misbehaviour	Single	Local area
Sinkhole	Route updating misbehaviour	Single	Local area or whole network
Sybil	Route updating misbehaviour	Single	Local area or whole network
Wormholes	Route updating misbehaviour	Multiple	Local area
Denial of service	Data forwarding misbehaviour	Single or multiple	Whole network
Hello floods	Route updating misbehaviour	Single	Whole network
Acknowledgment spoofing	Route updating misbehaviour	Single	Local area

The comparison of attacks in WSN is shown in Table 1 (Karlof & Wagner, 2003; Su et al., 2007; Wang et al., 2006; Wood & Stankovic, 2002). However, the majority of attack behaviour consists of the route updating misbehaviour, which influences data transmission. In the application of CWSN, the data are sensed and collected by SNs, and delivered to CH to aggregate. The aggregated data are then sent to sink from CH. Therefore, this paper designs and analyzes three different IDSs for SN, CH and the sink.

2.2. Analytic tool of intrusion detection

The proposed IHIDS in our research not only efficiently detects attack but also avoids the waste of resources. First, a large number of packet records are filtered by using the anomaly detection module, and then the misuse detection module is used to complete the whole detection. By training the mode of normal behaviour, the anomaly detection module detects the normalcy of current behaviour, as determined by the rules. The misuse detection module determines whether the current behaviour is an attack, and the Back Propagation Network (BPN) is used to classify the attacks.

2.2.1. Rule-based method

Rule-based presents the thoughts of an expert (Philippe, 1997). Because human thought is very complicated, the knowledge could hardly be presented by algorithms. Therefore, a rule-based method is used to analyze results. The rules are defined by an expert, through his experience and observation. Additionally, the rules are logged in a rule base after they have been defined. The basic method of expression of rule is "if...then", that means if "condition" is established and then the "conclusion" will occur. The methods of inference are classified into two categories:

(1) Forward chaining which determines a conclusion from a "cause." The advantages of this method are that the accuracy of reason is high and the false positive rate is low. However, its weakness is that it cannot infer new events, which have not been defined before.
(2) Backward chaining which determines a "cause" from a "conclusion." The advantage of this method is that it guesses all possible causes. However, its weakness is a high false positive rate.

Up to this point, most of the IDSs assume the rule-based method. This changes well-known attack behaviours or normal behaviours into rules, so as to determine if a communication is an intrusion or a normal behaviour. It quickly and accurately determines behaviours. However, when the rules are increased incessantly, the performance decreases. In our research, the rule-based method is applied to our IDSs, and the method of forward chaining is used to make an inference. The current behaviours are matched with the rules to find the intruder.

2.2.2. Back-Propagation Network (BPN)

Back-Propagation Network (BPN) is the most typical and the most general model to use in a neural network (Philippe, 1997). BPN is a model of supervised learning, through the specific environment to get the training data, which includes input and output variables. However, BPN learns from input and output variables inherent to the mapping rules to deduce what kind of variables these are. BPN is more suitable for diagnosis and prediction, etc.

A network structure of BPN includes many layers, and each layer has several processing units. The network structure of BPN consists of three layers, such as an input layer, a hidden layer, an output layer and many links between each layer. The input layer is used to get the outer environmental messages, and by the intersect computing in the hidden layer, a corresponding output is gotten from the output layer.

In the progression of BPN training, when all training data have been trained, the procedure is completed and it is called one epoch. The BPN learns training data repeatedly, and tunes the weights between layers continuously, through many epochs, until the output of the network is similar to the target value and a convergence is achieved.

2.2.3. Adaptive Resonance Theory (ART) Network

Grossberg proposed Adaptive Resonance Theory (ART) Network in 1976 (Carpenter & Grossberg, 1988). ART is a model of unsupervised learning, through the specific environment to get the training data, which only have input variables. Therefore, ART learns the rules of clustering using these training data, so as to infer which cluster the new data belong to. However, if the new data do not belong to any current clusters, and then ART would add a new cluster for the new data.

The network structure of ART consists of an input layer, an output layer and many links between these two layers. The input layer is used to input the input variables; the number of input variables is assigned according to a specific environment. The output layer is used to present the output variables; the number of output variables is only one before learning, but the number would increase progressively during learning. Finally, ART would achieve convergence after the number of output layer is stable. By the intersect computing between the links, ART could classify these data and obtain a corresponding output from the output layer.

In order to construct an IHIDS, a method is adopted which could deal with a large number of data to keep the system stable. In addition, because attacks vary with time, our IDS needs the capability of on-line learning. Therefore, ART is adapted to the learning mechanism of the proposed IHIDS in our research.

3. Research method and architecture

In this research, a mechanism of intrusion detection created in CWSN is proposed. According to varied capabilities and probabilities of suffer attack among sink, CH, and SN, three individual IDSs are designed. The research architecture of our study is shown in Fig. 3, an IHIDS for the sink, a HIDS for CH, and a misuse IDS for SN are proposed. There is a feedback mechanism between the sink and CH; HIDS will be retrained for the new type of attacks, which have been detected and classified by IHIDS.

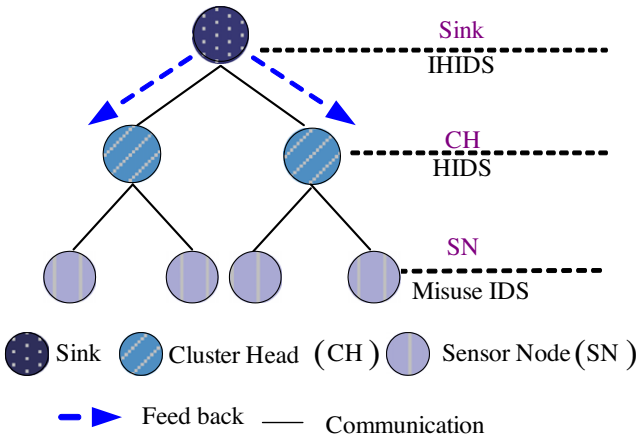

Fig. 3. Research architecture.

As a whole, an Integrated Intrusion Detection System (IIDS) for CWSN is designed. It can provide the system to resist intrusions, and process in real time by analyzing the attacks. The provided models and the adopted methods of IIDS are shown in Table 2.

3.1. Intrusion detection for the sink

As the resource of the general SNs are limited, the techniques which need better computing and more energy could not be implemented (Dasilva et al., 2005; Wang et al., 2006). In this research, an IHIDS is designed with an enormous resource of the sink, which combines anomaly and misuse detection. The proposed IHIDS not only can achieve a high detection rate and low false positive rate, but also can learn and add new classes by learning mechanism in real time when it suffers unknown attacks.

IHIDS proposed in this study consists of four models as shown in Fig. 4. First, the anomaly detection module is used to detect packets as normal or abnormal. Then, the misuse detection module screens the abnormal packets for type detection. Finally, the results of the two detection modules are integrated by the decision-making module to determine any intrusion and the type of intrusion, and return the same to the manager to follow-up treatment.

3.1.1. Anomaly detection model

The anomaly detection model is used as the first line of defence in IHIDS. Only few packets are actually attacks. The anomaly detection model acts like a filter. Abnormal packets are delivered to the misuse detection model for further detection.

Because the anomaly detection uses a defined model of normal behaviour, a packet is determined to be abnormal by the system when the current behaviour varies from the model of normal behaviour. As a result, the anomaly detection usually determines the normal communication as abnormal communication, and creates the problem of erroneous classification. However, it seldom marks an abnormal communication as a normal communication. Therefore, the anomaly detection model is used to filter a large number of packet records first, and make further detection with the misuse detection model, when the amount of information decreases. In other words, a large number of normal packets can be filtered in the anomaly detection model, and thus a waste of resources is avoided.

In CWSN, a large number of packets are to be detected, where most of them are normal. As a filter, packets will be screened first by anomaly detection module. If an abnormal packet is found, then the misuse detection module does further detection. The anomaly detection module detects the intrusion comparing the current behaviour with the normal behaviour. However, if the current behaviour and normal behaviour patterns vary, the system will be misjudged as abnormal.

In a CWSN, it is necessary for the packets to establish normal patterns of behaviour for monitoring the status of packets. Therefore, in this study, the rule-based analysis method is used to build anomaly detection modules and the corresponding rules are defined by experts. The flow of construction can be divided into three steps, as follows:

Step 1: Analysis of network packets sent by the history. In CWSN, the packets, which pass through the sink, are sent from the neighbour of CH. Therefore, the past packets that communicate on sink are collected to analyze, and the packet is divided into two types as normal or abnormal.
Step 2: Feature selection. Looking for identification of key features issued to distinguish between normal or abnormal packet.
Step 3: The establishment of anomaly detection rules. Based on the definition of a normal packet and the selected features, the rules are created. Then, the well-established rules are stored in the knowledge base.

In CWSN, when all CHs communicate with the sink, all the packets through the sink have to be screened by the anomaly detection module to determine whether there is any abnormal packet. If any abnormal packets are found, they are transferred to the second phase, where the misuse detection module looks for any misjudgement that has happened.

3.1.2. Misuse detection model

The misuse detection module utilizes various models of well-known attack behaviours, so we should build a model base according to these behaviours. Because the performance in most techniques of intrusion detection is promised through training data, BPN with the supervised learning method is adopted by this study. BPN learns the corresponding relations between input and output variables, and tunes the corresponding weight. It can bring down the error for inference to a minimum to achieve a high accuracy. Thus, BPN achieves high accuracy for our IHIDS through massive trainings.

In this research, a three-layer BPN is adopted for our misuse detection module of IHIDS that includes an input layer, a hidden layer and an output layer. The structure of the misuse detection model is shown in Fig. 5. We use abnormal packets, which were determined by anomaly detection module, as the input vector. The number of processing units in input layer is determined by the selected features for the packets. In addition, the number of processing units in hidden layer is designed through averaging the input layer units and the output layer units. After analysis, there are eight common attacks in CWSN, including Spoofed/Altered/Replayed Routing Information, Select Forward, Sinkhole, Sybil Attack, Wormholes, Denial of Service, Hello Floods and Acknowledgment Spoofing. Nine processing units in the output layer represent eight different attacks and one normal behaviour, to determine whether the inputted packet is an intrusion and make a classification.

Table 2
The adopted methods of IIDS.

Model	IDS		
	Sink IHIDS	CH HIDS	SN misuse IDS
Anomaly detection model	Rule-based	Rule-based	
Misuse detection model	BPN	BPN	Rule-based
Decision making model	Rule-based	Rule-based	
Learning mechanism	ART		

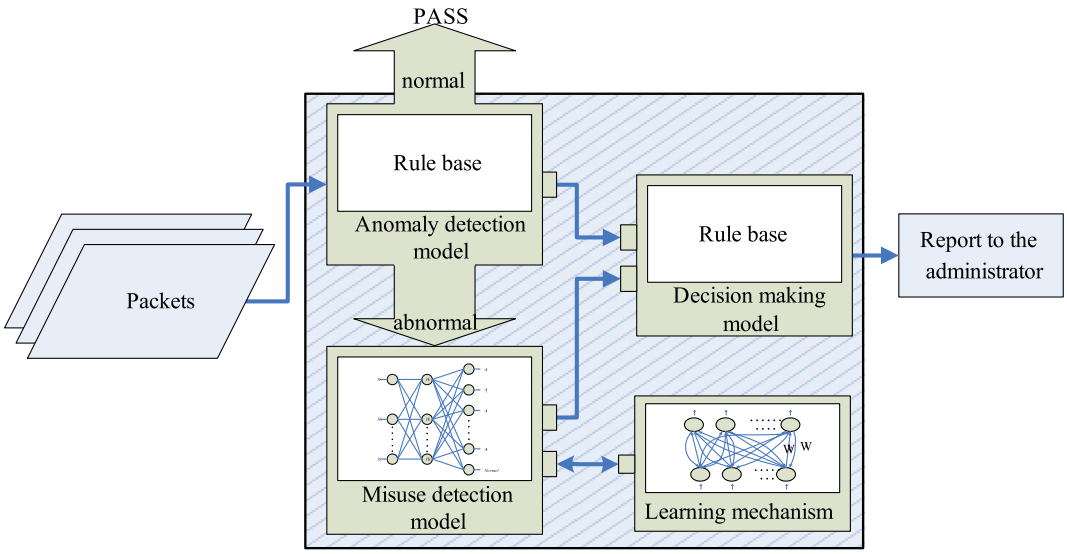

Fig. 4. System architecture of IHIDS.

The historical records of packets are collected, which passed through sink in CWSN, as the sample data for training. Most of packets are normal in CWSN. This makes the training data unbalanced. In other words, the abnormal packets will be neglected by the BPN due to the low occurrence rate. In order to avoid this problem, the training data are filtered through the anomaly detection module first, and then the abnormal packets will be separated which are taken for training.

Before sending the training data to BPN, the training data were normalized into a recognizable form of BPN. In other words, the packet records are converted into a stream binary value, and then sent to BPN. To get a better convergence, the learning rate is set to 0.5 or between 0.1 and 1.0 (Philippe, 1997). The actual learning rate is determined through simulation. Additionally, we assign values between 0 and 1 as the weights and biases randomly. Then, the training data are fed into BPN, computing the actual output through the method of feed forward. The error and correction of output and hidden layers are calculated through the method of back propagation. To update the weights and biases of network, until all training data have been used, this period is called one epoch. The training data can be learned repeatedly and tune the weights between layers continuously, through many epochs, until the output of network is similar to the target value, and the training is complete.

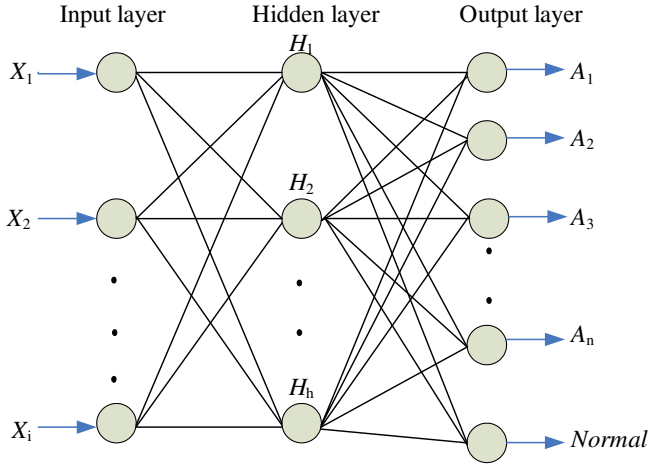

Fig. 5. The structure of the misuse detection model of IHIDS.

All abnormal packets, which were determined by the anomaly detection module, are subjected to misuse detection module. First, the abnormal packets are converted into binary value in a pre-processing step, and the binary value is sent into the misuse detection module to calculate the output. Finally, the results of detection are delivered to the decision-making module for integration.

3.1.3. Decision making model

The decision making module is used to integrate anomaly detection and misuse detection modules to detect whether the attack is of intrusion or invasion type. The rule-based method is adopted to establish the decision making model, using the rules to combine the outputs of two detection models, and its main advantages are that it is very simple and fast. The model used rules as shown in Table 3.

3.1.4. Learning mechanism

Attacks are variable with the development of users' behaviours and the advance of information technology. Therefore, if the capacity of the system cannot continue to improve, existing IDS systems will not provide security. Therefore, in order to solve this problem, our proposed IHIDS must be an intelligent detection system. When the system encountered new attacks, the learning mechanism has the ability to detect and learn to create new types of detection. However, the packets that cannot be correctly classified by misuse detection model will be regarded as an unknown attack. In addition, these packets will be transferred to the learning mechanism to learn and produce a new type of detection.

The ART is adopted to construct the learning mechanism of IHIDS, because ART could deal with a large number of data to keep the system stable, and it has the capability of on-line learning for variable attacks. In other words, ART could perform detection and learn new attacks simultaneously.

The ART is used to cluster unknown attacks for our learning mechanism that includes an input layer and an output layer, and the structure of the learning mechanism is shown in Fig. 6. The packets are used, which the misuse detection model cannot identify, as the input vector, and the number of input nodes is determined through the selected features for packet. The number of output node is only one at the beginning. However, more

Table 3
The rules of the decision making model of IHIDS.

Rules
If anomaly detection model detects an attack and misuse detection model does not detect the same attack, then the detected attack is not an attack and it is an erroneous classification
If anomaly detection model detects an attack and misuse detection model detects the same attack, then the detected attack is an attack and the attack mode is classified
If anomaly detection model detects an attack and misuse detection model finds it to be an unknown attack, then the detected attack is a new attack

different types of cluster will be generated by ART, thus the number of output node increases, while each output node represents a new type of attack.

Each packet of unknown attacks is inputted to the learning mechanism, and ART calculates the matching values for each output node, and then it finds the winning output node to calculate the similar value; finally, it does the vigilance test for winning output node. If the similar value of winning output node is greater than vigilance value, this means the inputted packet matches to the output node, therefore, it belongs to this cluster, and ART just has to update weights. On the other hand, if the similar value of winning output node is smaller than vigilance value, this means the inputted packet is not similar to the connected weights; hence, it does not belong to this cluster. ART has to find the next winning output node to see if it can pass the vigilance test. Otherwise, ART will generate a new output node, and it means a new attack. In addition, to define the suitable vigilance value, the sample data need to be tested by simulation.

In order to add new detection classes, the data of cluster are taken to retrain BPN of the misuse detection model when the value of cluster member achieves the defined threshold.

3.2. Intrusion detection for CH

HIDS proposed in this study consists of three modules as shown in Fig. 7. First, the anomaly detection module is used to filter the normal or abnormal packet. Then, the abnormal packets are judged through the misuse detection module for type detection. Finally, the results of the two detection modules are integrated by the decision-making module to determine whether there is intrusion and the type of intrusion, and return to the manager for follow-up treatment.

HIDS and IHIDS are similar, but the most difference between them is that HIDS does not have a learning mechanism; therefore, it cannot learn and classify new attacks immediately when it suffers unknown attacks. The main reason is that the resources of CH are less than the sink. Because of greater resources the sink has, there is no limit when using the resources. If CH consumes too much resource of computing and energy to detect intrusion, then the lifetime of network becomes shorter. Therefore, to reduce the workload of HIDS, there is no learning mechanism in it. However, the classes of attacks are updated using the feedback mechanism between CH and the sink.

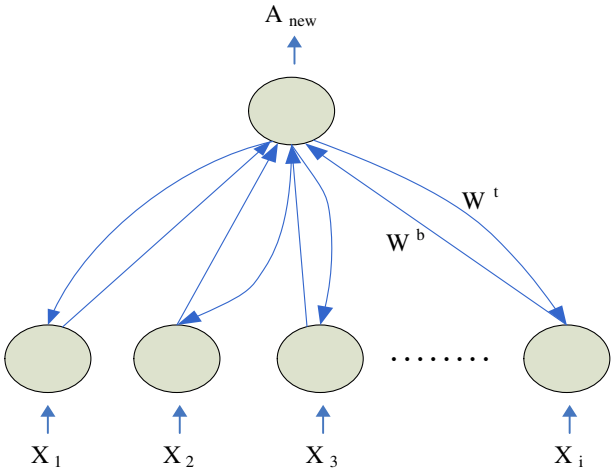

Fig. 6. The structure of the learning mechanism of IHIDS.

The feedback mechanism feeds the data of new attacks, which the learning mechanism of IHIDS provides to the misuse detection model of HIDS for retraining. This way not only does IHIDS get the same performance that HIDS spends additional resources to learn new attacks, but also it saves on resources. When HIDS gets the feedback message from the learning mechanism of IHIDS, the misuse detection model of HIDS is retrained using the data of new attacks at the next training for adding new detection classes.

Because the anomaly detection model, misuse detection model and decision making model in HIDS are same as those in IHIDS, the details of system structure are not described again.

3.3. Intrusion detection for SN

SN is seldom attacked by enemies, and its resources are less compared other devices. Therefore, a simple and fast method is designed for SN, since the damages are very little when SN suffers attacks. In this research, a misuse IDS is designed in SN and it is composed of only one model, as Fig. 8 shows. Misuse IDS determines whether a packet is an attack or not and identifies the category of attack by rules. To avoid SN overwork, and to save resources for the purpose of safety, the intrusion detection in SN is performed through these simple rules.

To protect itself from overwork, SN has to use its resources efficiently while detecting attacks and avoid wasting its resources. Because misuse detection has higher accuracy than anomaly detection, the misuse detection is adopted to do intrusion detection. This way not only is the problem of erroneous classification by anomaly detection avoided but the consumption of resources that false alarms cause is also decreased. In addition, the rule-based method is adopted to analyze behaviours, since it does not need complicated computing to infer, so the speed of matching is fast. Therefore, this method is suitable for our requirement that saves resources for the purpose of safety.

Due to the construction and adopted methods in misuse IDS are much similar to those in anomaly detection model of IHIDS and HIDS, the details of system structure are not described again.

4. Experiment

In this section, the proposed architecture is evaluated through simulation. However, the corresponding rules in anomaly detection module are defined by experts. Therefore, the correctness is judged by experts. Thus, in the main validation experiments conducted in this study, the use of BPN classification performance of the misuse detection module is evaluated. In addition, the learning mechanism, which is adopted by ART to classify the unknown attacks, is simulated when BPN could not identify the attacks.

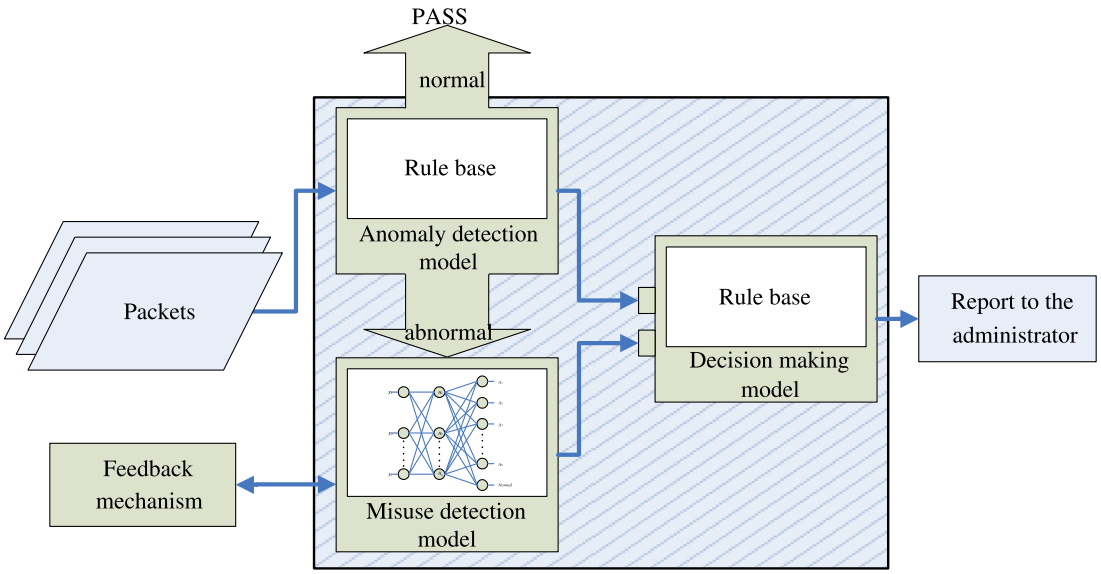

Fig. 7. System architecture of HIDS.

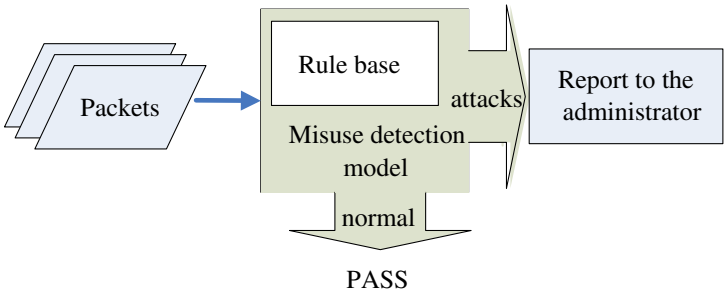

Fig. 8. System architecture of misuse IDS.

4.1. Data collection

As the real sample cannot be acquired in CWSN for intrusion detection, the KDDCup'99 dataset (kddcup, 1999) is used as the sample to verify the performance of the misuse detection module. The KDDCup'99 dataset, referred by Columbia University, was arranged from intrusions simulated in a military network environment at the DARPA in 1998. It was performed in the MIT Lincoln Labs, and then announced on the UCI KDDCup, 1999 Archive.

The features consist of 34 types of numerical features and seven types of symbolic features, according to different properties of attack. Additionally, the KDDCup'99 dataset includes many attack behaviours, classified into four groups: Probe, DoS, U2R, and R2L. It also includes a kind of normal communication. Therefore, these five behaviours are used in the experiment for the classification of HIDS.

The attacks of Spoofed, Altered, or Replayed Routing Information, Sinkhole, Sybil, Wormholes, and Acknowledgment Spoofing need to make a probe step before they begin to attack, so they would be classified as Probe attacks. Select Forward, which uses illegitimate data forwarding to make an attack, is known as a DoS attack. Sinkhole, Wormholes, and Hello Floods are caused by inner attacks, and are therefore classified as U2R. Spoofed, Altered, or Replayed Routing Information, Sinkhole, Sybil, Wormholes, Hello Floods, and Acknowledgment Spoofing use the weakness in the system to make an attack, so they would be classified as R2L.

In this research, we use the kddcup.data_10_percent.gz as our sample of training and testing dataset in experiment (kddcup, 1999). This includes 10% data in the KDDCup'99 dataset and the total number of communication records is 494,021. It randomly samples 30,000 records as training data and 15,000 records as testing data. However, the sample set of Probe, U2R, and R2L is small; hence, the whole records are sampled. Moreover, two-thirds of these records are taken as training data, and one-third as testing data. While other sample sets are sampled according to their ratio from kddcup.data_10_percent.gz, they are classified to Normal and DoS type separately (kddcup, 1999). The Normal accounts for about 20% and the DoS accounts for about 80%. The data distribution and ratio in kddcup.data_10_percent.gz and corrected.gz are shown in Table 4. In addition, the data sampling number and ratio for training and testing data are shown in Table 5.

Table 4
The data distribution and ratio in original dataset.

Category	Data			
	kddcup.data_10_percent.gz		Corrected.gz	
	Amount of total data	Ratio (%)	Amount of total data	Ratio (%)
Normal	97,278	19.69	60,593	19.48
Probe	4107	0.83	4166	1.34
DoS	391,458	79.24	229,853	73.90
U2R	52	0.01	228	0.07
R2L	1126	0.23	16,189	5.20
Total	494,021	100	311,029	100

Table 5
The data sampling number and ratio for training and testing data.

Category	Data			
	Training data		Testing data	
	Sample of training data	Ratio (%)	Sample of testing data	Ratio (%)
Normal	4943	16.48	2472	16.48
Probe	4107	13.69	2053	13.69
DoS	19,772	65.91	9886	65.91
U2R	52	0.17	26	0.17
R2L	1126	3.75	563	3.75
Total	30,000	100	15,000	100

4.2. The simulation design of BPN

In this subsection, the flow of experiment, feature selection, data pre-processing and the training BPN model are presented.

4.2.1. The flow of experiment

In this simulation, the training and testing data are sampled first from the KDDCup'99 dataset. In addition, the feature selection method proposed by Jong, Marchiori, Sebag, and van der Vaart (2004) is adopted to filter some unimportant and noise features to decrease the data dimension. Then, the data are normalized through the pre-processing step, which are used to train the BPN model, and then make a test. The simulation flow chart of BPN is shown in Fig. 9.

Not every feature has decisive effects on the output of classification. Some features even make classification errors. Therefore, feature selection is an important factor that can affect the performance of IDS. In this research, the feature selection method proposed by Jong et al. (2004) is adopted. Therefore, the data dimensions and the complications can be reduced.

Before training the BPN model, it must be normalized for the training data, and the training date must be converted to a data type which is recognizable by BPN. However, the original state is normalized for the training data, and 24 types of features are chosen (Carpenter & Grossberg, 1988). To achieve normalization, these 24 features are converted into binary value. We design a corresponding binary value to transfer the original value for the symbolic data. Additionally, the corresponding target value is classified into five groups: Normal, Probe, DoS, U2R, and R2L, which translates to 00001, 00010, 00100, 01000 and 10,000, respectively.

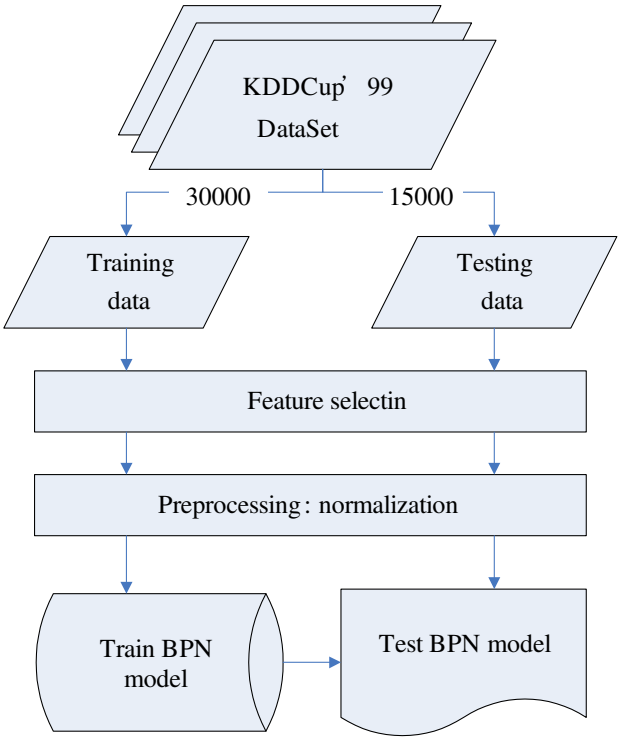

Fig. 9. The simulation flow chart of BPN.

BPN is a network model of supervised learning, inputting training data which have target values to make training, learning the training data repeatedly, and tuning the weights between layers continuously, until the output of network is similar to the target value, and training is completed. In the training process, original weights and biases are assigned from 0 to 1 randomly. Through the error back-propagation to find out the correction, it would stop until the network gets a convergence. The allocation of each layer in the three-layer BPN is shown in below:

(1) Input layer: According to the 24 types of features, chosen by Jong et al. (2004). The features are transferred into 95 binary values, and 95 neurons of input layer are produced.
(2) Output layer: The outputs have five types, including Normal, Probe, DoS, U2R and R2L. Therefore, five neurons of output layer are produced.
(3) Hidden layer: Rely on the mean method, adding the number with input units and output units, and dividing it by 2, to get the number of hidden layer unit. The 50 neurons of hidden layer are produced.

4.2.2. The simulation results of BPN

The adopted system in this research is the AMD Athlon(tm) 64 X2 Dual Core Processor 5000+ 2.59 GHz PC with 2048MB RAM, Windows XP Professional version OS, and using the NNtool which is built in the MATLAB 7.1 to train the BPN model.

The performance of the experiment is evaluated by the detection rate (DR), the false positive rate (FP) and the accuracy, according to the formulas (1)–(3)

$$\text{Detection rate} = \frac{\text{Number of detected attacks}}{\text{Number of attacks}} \times 100\%. \quad (1)$$

$$\text{False positive rate} = \frac{\text{Number of misclassified connections}}{\text{Number of normal connections}} \times 100\%, \quad (2)$$

$$\text{Accuracy} = \frac{\text{Number of correct classified connections}}{\text{Number of connections}} \times 100\%. \quad (3)$$

Table 6
The performance evaluation of HIDS.

DR (%)	FP (%)	Accuracy (%)
90.96	2.06	91.26

Table 7
A table of detailed classification.

Category of attacks	Amount of correct detection/amount of sample	DR (%)
Normal	2421/2472	97.94
Probe	1568/2053	76.38
DoS	9641/9886	97.52
U2R	4/26	15.38
R2L	55/563	9.77

Table 8
The accuracies in different thresholds.

Threshold	Correct classification/amount of samples in which the output value is greater than the threshold	Accuracy (%)
0.8	13,602/14,705	92.50
0.9	13,588/14,601	93.06
0.95	13,563/14,514	93.45
0.99	13,369/14,053	95.13
0.999	11,642/11,964	97.31

According to the result of experiment that is shown in Table 6, the DR is 90.96%, the FP is merely 2.06%, and the accuracy is 91.26%. Analyzing each class of attacks shown in Table 7 and each individual performance, it can be observed that the DR figures of Probe, U2R, and R2L are not very good, even the DR of U2R is 15.38% and of R2L is 9.77%. This is because the training data of U2R and R2L are too less, and have resulted in the low detection performance. Moreover, there are many new attacks in these classes, so BPN would make an erroneous classification when it could not identify these attacks. Therefore, to keep the high DR for our IDS, our models need to be updated constantly by the learning mechanism.

4.3. The simulation of ART

In this subsection, the setting of BPN's threshold and the definition of ART's vigilance value are explained, and finally our simulation results are presented.

4.3.1. Define the threshold of BPN

The output value of BPN is between 0 and 1, and it takes the maximum output node as the classification, where it belongs. To assure the high accuracy of BPN, setting a threshold in BPN is a way to verify the maximum output value. Therefore, if the output value is lower than the threshold, we can define that the classification is not correct and it is an unknown attack, and using ART to reclassify these packets which are lower than the threshold, finally the whole accuracy will be improved.

To achieve the goal of the whole accuracy above 95%, and to avoid filtering too much data by a high threshold, we have to set a suitable threshold. The accuracies in different thresholds are shown in Table 8. We can see that a suitable threshold for our requirement is where the 0.99 threshold is, and the accuracy achieved is 95.13% and only 947 data were filtered. Otherwise, next there is the 0.999 threshold. Even though it could achieve 97.31% accuracy, it filtered 3036 data from 15,000 data. Therefore, the result is actually worse and results in the learning mechanism overload.

Therefore, the BPN's threshold is set as 0.99, we can define that the classification is not correct when the output value is under the threshold. There are 947 packets which are under the threshold in this experiment. We forward these packets to the learning mechanism to reclassify.

Table 9
Amount of clusters in different vigilance values.

Vigilance value	1	0.5	0.4	0.3	0.2	0.1	0.05
Amount of clusters	521	111	81	55	31	14	5

Table 10
The data distribution when vigilance value is set as 0.05.

Cluster	Data which have not passed BPN's threshold
1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 25, 26, 27, 29, 30, 38, 45, 48, 52, 54, 55, 56, 58, 76, 78, 80, 84, 93, 95...
2	24, 28, 35, 42, 53, 57, 59, 65, 77, 103, 124, 126, 127, 128, 130, 132, 135, 136, 137, 138, 141, 143, 145, 150, 151, 152, 153, 155, 158, 169, 177...
3	31, 32, 33, 34, 36, 37, 39, 40, 41, 43, 44, 46, 47, 49, 50, 51, 60, 61, 62, 63, 64, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 79, 81, 82, 83, 85, 86, 87, 88...
4	142, 144, 147, 148, 159, 160, 161, 162, 163, 165, 166, 173, 174, 175, 181, 184, 185, 186, 187, 190, 191, 196, 199, 200, 201, 202, 203, 205, 210...
5	263, 277, 279, 280, 281, 285, 286, 288, 290, 293, 294, 296, 297, 298, 300, 301, 302, 303, 304, 305, 306, 311, 312, 314, 315, 318, 319, 320, 321...

4.3.2. Define the vigilance value of ART

A packet is seen as unknown attack if the output value is under the threshold, and then it is forwarded to the learning mechanism to know its type. After filtering packets by BPN's threshold, there are 947 packets, which are delivered to ART for classification. The results of classification are verified through the plug-in ART tools in MATLAB 7.1 in this experiment.

In order to find a moderate number of clusters the vigilance value should be discussed. This is because vigilance value is an important factor to determine the number of clusters. Vigilance value is between 0 and 1, and to make different classifications, several values are set to perform it, so as to verify the result among classifications, as shown in Table 9. When the setting value is equal to 1, there are 521 clusters produced from 947 data; however, the number is too big since there are only five classes in BPN. Therefore, to control it to fit a moderate number is a problem which should be overcome. When the setting comes to 0.05, there are five clusters classified, and this number seems suitable for this research. Thus, the vigilance values are set as 0.05 to perform classification in this research.

4.3.3. The simulation results of ART

So far, we know that, from the experiments of BPN's threshold sets and ART's vigilance value, we can achieve the whole accuracy higher than 95% when BPN's threshold is set as 0.99, and avoid filtering too much data by a high threshold in our sample. In addition, by setting the vigilance value as 0.05, the clusters are controlled into a moderate range, and classifications are performed well. Table 10 shows that 947 data have not passed BPN's threshold. We classify these data into five clusters when the vigilance value is set as 0.05, and each cluster is a new attack.

When the number of a cluster reaches a defined threshold, we would take the data from this cluster to retain BPN of the misuse detection model, so as to add new detection class in BPN for this new attack. The threshold of the number of a cluster is defined according to the actual environment. We have to accumulate more samples in the environment in which attacks are variable, because enough samples could assure the performance of BPN. Otherwise, we just need enough samples in the environment in which attacks are stable, and BPN could classify accurately.

5. Conclusion and future works

In our research, an Integrated Intrusion Detection System (IIDS) is discussed in a heterogeneous CWSN. Three individual IDSs for the sink, CH and SN are designed according to varied capabilities and probabilities attacks that they suffer from. For the sink, an IHIDS is proposed which has the learning ability; it not only decreases the threat of attack in the system, but also learns and adds new classes by learning mechanism in real time when the sink suffers unknown attacks. For CHs, a HIDS is proposed which has the same detection models as IHIDS, but there is no learning ability in HIDS. Its goals are to detect attacks efficiently and avoid resource wasting. However, HIDS updates the classes of attacks using the feedback mechanism between CH and the sink. For SNs, a misuse IDS is proposed. A simple and fast method for SN is designed, to avoid SN overwork, and to save resources for the purpose of safety.

In this experiment, the performance of the misuse detection model is evaluated first, which is implemented by BPN. The simulation results present the performance of this method: the detection rate amounted to 90.96%, the false positive rate was 2.06% and its accuracy amounted to 99.75%. We can see that the performance of BPN is not good enough. Therefore, to keep the performance for our IDS efficient with a higher accuracy rate, the unknown attacks are classified by the learning mechanism, which is implemented by ART. With the help of ART, the simulation results reached an accuracy of 95.13%; also, five kinds of attack classes were added.

The method of feature selection is one of the important factors, which affects the performance of IDS. We adopt the proposed method of feature selection by Jong et al. (2004), but we can use other methods to select features in the future, such as data mining, to find identifiable features, instead of relying on the viewpoint of experts. Additionally, our rule-based method is also defined by the expertise of experts. We can use a method, which has the learning ability, and collocate with the selected features to provide our anomaly detection module with better performance and flexibility.

Acknowledgment

This work was supported in part by the Taiwan National Science Council under Grants NSC96-2221-E-324-021 and NSC97-2221-E-324-007-MY3.

References

Akkaya, K., & Youngish, M. (2005). A survey on routing protocols for wireless sensor networks. *Ad Hoc Network, 3*(3), 325–349.

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: A survey. *Computer Networks, 38*(4), 393–422.

Bace, R., & Mell, P. (2001). *Special publication on intrusion detection systems*. Tech. report SP 800-31, National Institute of Standards and Technology, Gaithersburg, Md.

Carpenter, G. A., & Grossberg, S. (1988). The ART of adaptive pattern recognition by a self-organizing neural network. *IEEE Computer, 21*(3), 77–88.

Dasilva, A. P. R., Martins, M. H. T., Rocha, B. P. S., Loureiro, A. A. F., Ruiz, L. B., & Wong, H. C. (2005). Decentralized intrusion detection in wireless sensor networks. In *Proceedings of the 1st ACM international workshop on quality of service and security in wireless and mobile networks* (pp. 16–23).

Depren, O., Topallar, M., Anarim, E., & Ciliz, M. K. (2005). An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks. *Expert Systems with Applications, 29*(4), 713–722.

Heinzelman, W. R., Kulik, J., & Balakrishnan, H. (1999). Adaptive protocols for information dissemination in wireless sensor networks. In *Proceedings of the 5th annual ACM/IEEE international conference on mobile computing and networking* (pp. 174–185).

Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H. (2000). Energy-efficient communication protocols for wireless microsensor networks. In *Proceedings of the 33rd annual Hawaii international conference on system sciences (HICSS-33)* (Vol. 2, pp. 1–10).

Jong, K., Marchiori, E., Sebag, M., & van der Vaart, A. (2004). Feature selection in proteomic pattern data with support vector machines. In *Proceedings of the computational intelligence in bioinformatics and computational biology (CIBCB'04)* (pp. 41–48).

Karlof, C., & Wagner, D. (2003). Secure routing in wireless sensor networks: Attacks and countermeasures. *Ad Hoc Networks, 1*(2-3), 293–315.

Kddcup (1999). <http://www.kdd.ics.uci.edu/databases/kddcup99/kddcup>.

Kemmerer, R. A., & Vigna, G. (2002). Intrusion detection a brief history and overview. *Computer, 35*(4), 27–30.

Laerhoven1, K. V. (2004). Medical healthcare monitoring with wearable and implantable sensors. In *Proceedings of the third international workshop on ubiquitous computing for pervasive healthcare applications*.

Lindsey, S., & Raghavendra, C. S. (2002). PEGASIS: Power efficient gathering in sensor information systems. In *Proceedings of the IEEE aerospace conference* (Vol. 3, pp. 1125–1130).

Manjeshwar, A., & Agrawal, D. P. (2001). TEEN: A protocol for enhanced efficiency in wireless sensor networks. In *Proceedings of the 15th international workshop on parallel and distributed computing issues in wireless networks and mobile computing* (pp. 2009–2015).

Manjeshwar, A., & Agrawal, D. P. (2002). APTEEN: A hybrid protocol for efficient routing and comprehensive information retrieval in wireless sensor networks. In *Proceedings of the international parallel and distributed processing symposium (IPDPS 02)* (pp. 195–202).

Mhatre, V., & Rosenberg, C. (2004). Homogeneous vs. heterogeneous clustered sensor networks: A comparative study. *Proceedings of IEEE International Conference on Communications* (6), 3646–3651.

Mhatre, V., Rosenberg, C., Kofman, D., & Shroff, N. (2004). Design of surveillance sensor grids with a lifetime constraint. *Lecture Notes in Computer Science, 2920*, 263–275.

Murali, A., & Rao, M. (2005). A survey on intrusion detection approaches. In *Proceedings of the first international conference on information and communication technologies* (pp. 233–240).

Onat, I., & Miri, A. (2005). An intrusion detection system for wireless sensor networks. *Proceedings of IEEE International Conference on Wireless and Mobile Computing, Networking and Communications* (3), 253–259.

Philippe, D. E. (1997). *Neural network models: Theory and projects*. London, New York.

Rajasegarar, S., Leckie, C., & Palaniswami, M. (2008). Anomaly detection in wireless sensor networks. *Wireless Communications, 15*(4), 34–40.

Su, W. T., Chang, K. M., & Kuo, Y. H. (2007). eHIP: An energy-efficient hybrid intrusion prohibition system for cluster-based wireless sensor networks. *Computer Networks, 51*(4), 1151–1168.

Wang, Y., Attebury, G., & Ramamurthy, B. (2006). A survey of security issues in wireless sensor networks. *IEEE Communications Surveys and Tutorials, 8*(2), 2–23.

Wang, Y., Wang, X., Xie, B., Wang, D., & Agrawal, D. P. (2008). Intrusion detection in homogeneous and heterogeneous wireless sensor networks. *IEEE Transactions on Mobile Computing, 7*(6), 698–711.

Wood, A. D., & Stankovic, J. A. (2002). Denial of service in sensor networks. *Computer, 35*(10), 54–62.

Zhenwei, Y., & Tsai, J. J. P. (2008). A framework of machine learning based intrusion detection for wireless sensor networks. In *Proceedings of IEEE international conference on sensor networks, ubiquitous and trustworthy computing* (pp. 272–279).