Liu and Zhu *EURASIP Journal on Wireless Communications and Networking* (2019) 2019:109
https://doi.org/10.1186/s13638-019-1419-z

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

A new intrusion detection and alarm correlation technology based on neural network

Yansong Liu[1,2] and Li Zhu[1*]

Abstract

With the continuous development of computer networks, the security of the network has become increasingly prominent. A major threat to network security is the intrusion of information systems through the network. Intrusion detection of the traditional intrusion detection and alarm technology is not sufficient. Based on neural network technology, this paper studies the intrusion detection and alarm correlation technology. Based on the research on the working principle and workflow of the existing intrusion detection system, a new neural network-based intrusion detection and alarm method is proposed. A neural network-based intrusion detection and alarm system is designed and implemented. Through the experiment of the system prototype, the results show that the intrusion detection and alarm system based on the neural network has a higher detection rate and a lower false alarm rate for intrusion behaviors such as denial of service attack and has higher detection ability for unknown attack behaviors.

Keywords: Network security, Neural network, Intrusion detection, Alarm

1 Introduction

With the rapid development of network technology and the advent of the network era, network security has been becoming more and more important [1]. Determining a complete network security policy is an important means to ensure network security [2, 3]. One of the most popular network security models is the PPDR model, which focuses on network security policies. It includes four parts: policy, protection, detection, and response [4].

Intrusion detection and alerting techniques are the primary means of implementing the "detection" part of the model [5]. Intrusion detection and alerting, as a proactive security protection technology, provides a real-time protection against internal attacks, external attacks, and misuse, intercepting and responding to intrusions before the network system is compromised [6]. However, with the diversification of intrusion technology, the traditional expert system-based intrusion detection and alarm technology gradually exposes its shortcomings such as the inability to detect unknown attacks or even variants of unknown attacks [7]. Therefore, the implementation methods for intrusion detection should be diversified, and it is a general trend to integrate intelligent technology into intrusion detection and alarm systems [8].

In this paper, we use the advantages of artificial neural network self-learning, self-adaptation, nonlinearity, and robustness to study a simple and effective intrusion detection and alarm system based on artificial neural network. This is also the purpose of studying the system, that is, designing and implementing intrusion detection and alarm model, which not only can detect known attacks, but has good detection capabilities for unknown attacks.

2 Related work

2.1 Intrusion detection technology

An intrusion is any act that attempts to compromise the integrity, confidentiality, or availability of computer resources. At present, network security is becoming increasingly prominent [9]. In the first half of 2018, the statistics of Tencent Security Anti-Virus Lab showed that the total number of viruses blocked by the PC side was one billion times [10]. The total number of viruses increased by 30% compared with the number of viruses

* Correspondence: magiccobble@stu.xjtu.edu.cn
[1]Xi'an Jiao Tong University, Xi'an 710049, Shaanxi, China
Full list of author information is available at the end of the article

intercepted by Tencent Security Anti-Virus Lab in the second half of 2017. The Trojan virus was intercepted nearly 170 million times. In April and June, the peak of the virus was intercepted, and the number of interception was 180 million times, as shown in Fig. 1.

Intrusion detection is the detection of intrusion [11]. It collects information from several key points in a computer network or system and analyzes this information to discover whether there are violations of security policies and signs of attacks in the network or system [12]. The combination of software and hardware for intrusion detection is the intrusion detection system (IDS). The intrusion detection system is mainly composed of the following parts:

1. Data collection and audit data reflect the status of information and pass it to the detector.
2. The detector is responsible for analyzing and detecting the intrusion and issuing a warning message.
3. Knowledge base provides the necessary data information support.
4. The controller responds manually or automatically according to the alarm information.

In recent years, major progress has been made in directions for intrusion detection technology:

1. Distributed intrusion detection and general intrusion detection architecture.
2. Application layer intrusion detection
3. Intelligent intrusion detection.
4. Evaluation methods for intrusion detection.
5. The combination of network security technologies.

Classification is based on intrusion detection methods. Two basic analytical methods are commonly used to analyze events and detect intrusion behaviors, namely misuse detection and anomaly detection. Both analytical methods have their own advantages and disadvantages [13]. The most effective method should be the use of misuse detection by the subject technology and assist in the use of anomaly detection.

2.2 Artificial neural network technology

Artificial neural network technology is an intelligent information processing technology that simulates the processing, storage, and processing mechanism of the human brain [14]. The artificial neural network is a highly complex large-scale nonlinear adaptive network system composed of a large number of simple processing unit neurons. Through the corresponding learning algorithm, the relationship between the features and data contained in the data set is abstracted to the nerve. The form distribution of the state of the element and the strength of the connection between the neurons implements a mapping from input to output.

2.2.1 Basic principles of neurons

The structure of the neural network is determined by the basic processing unit and its interconnection method. The basic processing unit of the connection mechanism structure and neurophysiological analogy is often referred to as neurons [15, 16]. Each neuron model that constructs the network simulates a biological neuron, as shown in Fig. 2.

The neuron unit consists of a plurality of inputs, $i = 1, 2, \ldots, n$ and an output y. The intermediate state is

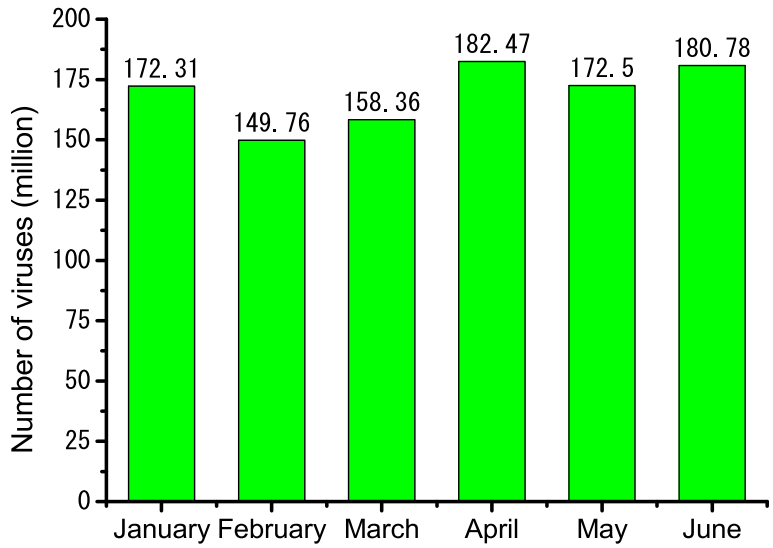

Fig. 1 Number of viruses intercepted by the PC

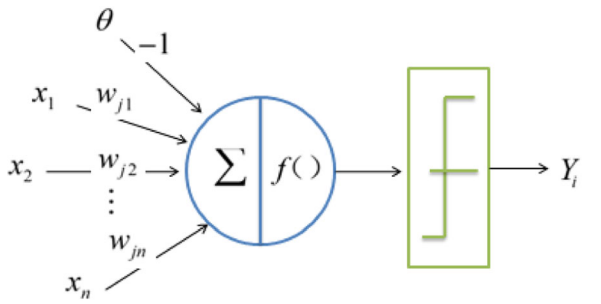

Fig. 2 Neuronal model

represented by the weight of the input signal and the output is:

$$y_i(t) = f\left(\sum_{j=1}^{n} w_{ij}x_j - \theta i\right) \tag{2.1}$$

In Eq. (2.1), θi neuron unit bias (threshold), w_{ij} connections weight coefficient (for the excited state, w_{ij} positive values, for the suppression state, w_{ij} negative values), n is the number of input signals, y_i neuron output, t is the time, $f()$ output transform function, sometimes called excitation or excitation function, often using 0 and 1 binary function or sigmoid function, these three functions are continuous and non-linear. A binary function can be represented by Eq. (2.2):

$$f(x) = \begin{cases} 1, x \geq x_0 \\ 0, x < x_0 \end{cases} \tag{2.2}$$

A conventional sigmoid function can be represented by Eq. (2.3):

$$f(x) = \frac{1}{1+e^{-ax}}, 0 < f(x) < 1 \tag{2.3}$$

The commonly used hyperbolic tangent function replaces the conventional sigmoid function because the output of the sigmoid function is positive, and the output value of the hyperbolic tangent function can be positive or negative. The hyperbolic tangent function is shown in Eq. (2.4):

$$f(x) = \frac{1-e^{-ax}}{1+e^{-ax}}, -1 < f(x) < 1 \tag{2.4}$$

2.2.2 The basic structure of artificial neural networks

1. Recursive network: In a recursive network, multiple neurons are interconnected to organize an interconnected neural network. The output of some neurons is fed back to the same or anterior neurons. Therefore, the signal can flow from the forward and reverse directions. The Hopfield network, the Elman network, and the Jordan network are representative examples of recursive networks. A recursive network is also called a feedback network. In Fig. 3, the state of V_i node, the input (initial) value of X_i nodes, and the output value after convergence of X'_i, $i = 1, 2, \ldots, n$.
2. Feed forward network: The feed forward network has a hierarchical layered structure, which consists of some layers of interconnected neurons that do not have interconnections. The signal from the input layer to the output layer flows through a one-way connection; the neurons are connected from one layer to the next, but there is no connection between the neurons in the same layer.

In the figure, the solid line indicates the actual signal flow and the broken line indicates the back propagation. Examples of feed forward networks are multilayer perceptron (MLP), learning vector quantization (LVQ) networks, cerebellar model connection control (CMAC) networks, and data processing methods (GMDH) networks.

2.3 Typical model of artificial neural network

Among the dozens of neural network models, Hopfield network, back propagation (BP) network, adaptive resonance theory (ART) network, and LVQ network are used.

2.3.1 Hopfield Network

The Hopfield network is the most typical feedback network model, but it is one of the most studied models. The Hopfield network is a self-associative network of single layers composed of the same neurons without learning functions. The training of the Hopfield network is only one

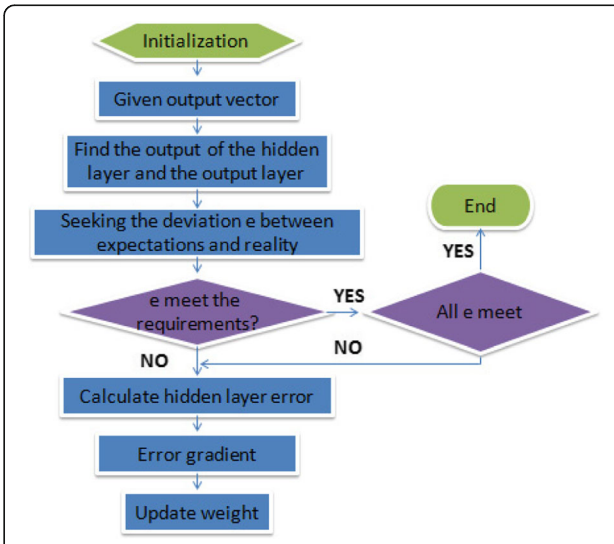

Fig. 3 The system flow based on neural network

step. The weight of the network is directly specified as follows:

$$w_{ij} = \begin{cases} \frac{1}{N}\sum x_i^c x_j^c, i \neq j \\ 0, i = j \end{cases} \quad (2.5)$$

When an unknown mode is input to this network, set its output initial value that equal to the component of the unknown mode, namely:

$$y_i(0) = x_i, 1 \leq i \leq N \quad (2.6)$$

Starting from these initial values, the network iterates according to the following equation until a certain minimum energy voltaic state is reached, i.e., its output is stable at a constant value:

$$y_i(k+1) = f\left[\sum_{j=1}^{N} w_{ij} y_i(k)\right], 1 \leq i \leq N \quad (2.7)$$

Where f is a hard limit function defined as:

$$f(x) = \begin{cases} -1, x < 0 \\ 1, x > 0 \end{cases} \quad (2.8)$$

2.3.2 Back propagation network

The BP network is a multi-layer image network that is reversed and can correct errors. When the parameters are not appropriate, the network can converge to a small mean square error so it is one of the most widely used networks. It can be used for language synthesis, recognition, and adaptive control. The training of BP network is divided into two processes: forward propagation and reverse propagation.

Forward propagation process is like this. The weight w_{ki} and the neuron threshold θ_k of each layer of the initialization network are set to be small random numbers not equal to zero. Where w_{ki} represents the weight between the k-th neuron of the current layer and the i-th neuron of the next layer, and θ_k represents the closed value of the k-th neuron of the current layer, according to the provided samples, the calculation of each layer of the neural network and the output of the neurons is as follows:

$$y_k = \phi\left(\sum_{j=1}^{p} w_{kj} - \theta_k\right) \quad (2.9)$$

Where $x_1, x_2, \ldots, x_p$ all are inputs to the neuron, and $\phi()$ is the Sigmoid function. There are two types of $\phi()$:

$$\phi(v) = \frac{1}{1 + e^{-ov}} \quad (2.10)$$

$$\phi(v) = \frac{1 - e^{-v}}{1 + e^{-v}} \quad (2.11)$$

This paper adopts the first form and takes the coefficient $\sigma = 1$ and calculates the errors of the current learning samples:

$$E = \sum_{i=1}^{m} (d_i - y_i)^2 \quad (2.12)$$

where di is the standard output of the sample and m is the number of output neurons.

Back propagation process is like this. The weight and threshold are adjusted according to the output error term of each layer of neurons. The j-th neuron of the output layer uses the formula as follows, where d_1 is the standard output of the sample:

$$\delta_j = y_j\left(1 - y_j\right)\left(d_i - y_j\right) \quad (2.13)$$

The k-th neuron of the hidden layer uses the formula:

$$\delta_k = o_k(1 - o_k)\sum_{j=1}^{m} \delta_j w_{kj} \quad (2.14)$$

Where o_k the actual output of the neuron is, δ_j is the output error of the j-th neuron in the output layer, and w_{kj} is the weight between the k-th neuron of the hidden layer and the j-th neuron of the output layer; the adjustment formula is:

$$w_{kj} = w_{kj} + \eta \delta_j O_k \quad (2.15)$$

where η is the learning speed; the threshold adjustment formula is:

$$\theta_j = \theta_j + \eta \delta_j \quad (2.16)$$

Determine whether the learning sample has been learned. If it has not been completed, enter the forward propagation phase. If the sample has been learned, calculate the cumulative errors of all samples, if $E < \varepsilon$ is satisfied. Then, the training process is over and enters the test phase; otherwise, the sample is re-learned and transferred to the forward propagation phase. The formula for calculating the cumulative error is:

$$E = \frac{\sqrt{\sum_{i=1}^{P} E_i}}{m \times p} \quad (2.17)$$

where p is the number of sample learning, and m is the number of neurons in the output layer.

2.3.3 Adaptive resonance theory (ART)

The ART network is also a self-organizing network model. The adaptive resonance theory (ART) network has different versions including ART-1 version for handling binary input. New versions, such as ART-2, are capable of handling continuous value inputs.

The alert vector v_i constitutes a short-term memory of the network. v_i and w_i are related, w_i is a normalized copy of v_i, i.e.,:

$$w_i = \frac{v_i}{\varepsilon + \sum v_{ji}} \tag{2.18}$$

2.3.4 Learning vector quantization network

A learning vector quantization (LVQ) network consists of three layers of neurons, the input conversion layer, the hidden layer, and the output layer.

3 Methods

For the application of artificial neural networks in intrusion detection and alarm, the predecessors have done some research work. The main function of the intrusion detection and alarm system is to detect the intrusion behavior in the computer network or computer system. The main actions include data collection, data clustering, behavior judgment and classification, alarm time, and response to intrusion time. But in the final analysis, there are only two cores of intrusion detection systems:

1. Based on the existing attack knowledge, it is speculated whether the current context-related time is a suspicious attack behavior.
2. Based on the existing knowledge of normal system activities, it is speculated whether the current context with context is within the scope of normal system activity.

The above knowledge is the basis on which the intrusion detection system relies. In practical applications, it exists in the form of system logs and intrusion attack means.

3.1 System design

The intrusion detection alarm model based on the artificial neural network mainly includes the following modules: network packet interception module, protocol parsing module, message parsing module, artificial neural network detection, and response module, as shown in Fig. 3:

The network packet interception module is responsible for intercepting data packets from the network, including data frames, IP packets, and packets transmitted on the network. This part of the work can be done by some software, such as Lipcap and Wincap. The protocol parsing module mainly performs protocol parsing and rule matching. The preprocessing module is responsible for the numerical conversion function and converting network packets of different protocol layers into corresponding mathematical vectors as input signals of the neural network engine. The network data packets are classified, and each type of data packet is detected by using a suitable neural network engine to determine whether it is an attack behavior, such as Web submodule and Telnet submodule.

3.2 Selection of neural network types

This design uses BP network, and the full name of BP network is Back Propagation Network, BP network training algorithm originally developed by Werbos, which is an iterative gradient algorithm for solving the minimum mean square errors between the actual output and the expected output of the feedforward network. The BP network is a multi-layer mapping network that transmits backwards and corrects errors. When the parameters are not appropriate, the network can converge to a small mean square error, so it is one of the most widely used networks.The learning algorithm process of BP network:

1. Forward transmission of information

Assuming the BP network has a total of L layers, for a given p samples, the expected output of the network is:

$$T_d = \left[T_{d1}, T_{d2}, \ldots T_{dp}\right] \tag{3.1}$$

When the p-th sample is input, the operational characteristics of the j-th neuron in the $l(l = 1, 2, \ldots, L-1)$ layer in the network are:

$$net_{jp}^{l} = \sum_{i=1}^{n_{l-1}} W_{ji}^{(l)} O_{ip}^{(l-1)} - \theta_j^l \tag{3.2}$$

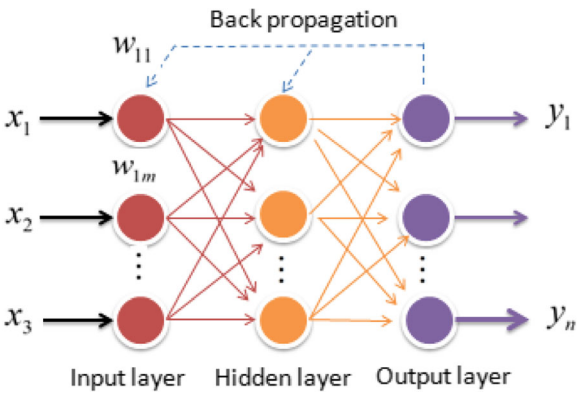

Fig. 4 Forward network structure

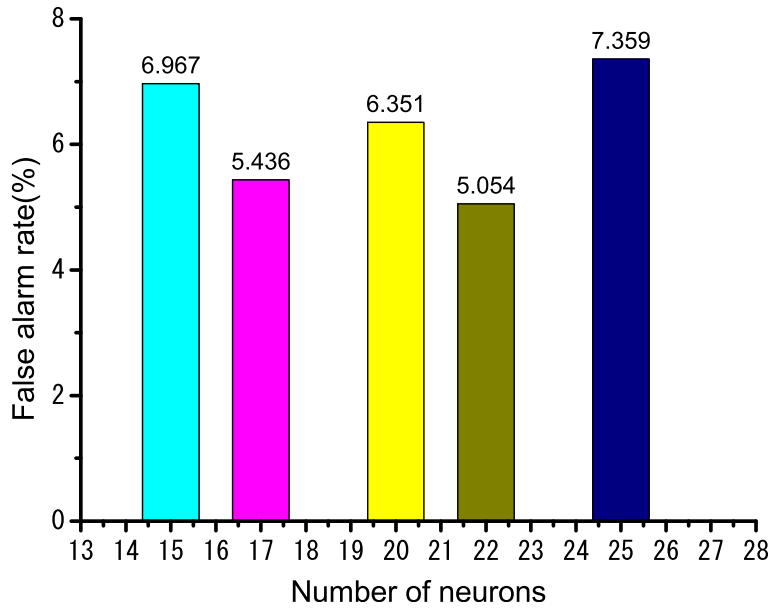

Fig. 5 BP neural network false positive rate

$$O_{jp}^{(l)} = f_l(net_{jp}^{(l)}) \tag{3.3}$$

Where w_{ji} is the connection weight of neuron i to neuron j; n_{L-1} is the number of nodes of the $L-1$ th layer; $O_{jp}^{(l-1)}$ is the current input of neuron j; $O_{jp}^{(l)}$ is the output of neuron j; f_l is a nonlinear different non-decreasing function, generally taken as an S-type function, namely:

$$f_l(x) = \frac{1}{1+\mathrm{e}^{-x}} \tag{3.4}$$

And for the output layer, there is

$$O_{jp}^{(l)} = f_L\left(net_{jp}^{(l)}\right) = \sum_{i=1}^{n_{l-1}} W_{ji}^{(l)} O_{ip}^{(L-1)} - \theta_j^L \tag{3.5}$$

The purpose of neural network learning is to achieve in each sample

$$E_p = \frac{1}{2}\sum_{i=1}^{m}\left(T_{jdp} - \hat{T}\right)^2, (p = 1, 2, \ldots, p) \tag{3.6}$$

(where m is the number of output nodes) to the minimum, thus ensure the total network error

$$E = \sum_{p=1}^{p} E_p \tag{3.7}$$

It is minimized, where $T_{jdp}, \hat{T}_{jp}$ is the expected output and actual output of the jth node of the output layer.

2. Using the gradient descent method to determine the changes of weights and the back propagation of errors

The gradient algorithm is used to correct the network weight and threshold. The weight coefficient iteration equation for the first layer is:

$$W(k+1) = W(k) + \Delta W(\mathrm{k}+1) \tag{3.8}$$

$$W = \{w_{ij}\} \tag{3.9}$$

where k is the number of iteration

$$\text{Make } \Delta_p w_{ji}\infty = \frac{\partial E_p}{\partial w_{ij}^{(l)}} = -\frac{\partial E_p}{\partial net_{jp}^{(l)}}\frac{\partial net_{jp}^{(l)}}{\partial w_{ij}^{(l)}} = -\frac{\partial E_p}{\partial net_{jp}^{(l)}} O_{ip}^{(l-1)} \tag{3.10}$$

Table 1 Input vector

Duration	Protocol type	Serve type	Link status	SR byte	RS byte	Land	Wrong frame	Urgent

Table 2 Detect sample type distribution

Sample type	Normal	Neptune	Udpstorm	Process Table	Back
Number of samples	54,669	400	2000	1232	689
Sample type	Apache2	Teardrop	Warezmaster	Shprocesstable	POD
Number of samples	1500	90	215	1111	584
Sample type	Smurf	Selfping	Mailbomb	Warezclient	Syslogd
Number of samples	112	137	200	50	1
Sample type	Crshiis	Tcpreset	Land	Dosnuke	
Number of samples	4	6	1	9	

The normal samples were randomly selected from 54,669 records. The attack data was selected from one attack of various attack types, and the total number of attack samples was 8341.

$$\text{Make } \delta_{pj}^{(l)} = -\frac{\partial E_p}{\partial net_{jp}^{(l)}}, \text{ then there is} \tag{3.11}$$

$$\Delta_p w_{ji} = \eta \delta_{pj}^{(l)} O_{ip}^{(l-1)}, \text{ where } \eta \text{ is learning step.} \tag{3.12}$$

3.3 BP network design

The intrusion detection model is modeled by using a multi-layer forward network. The structure of the multi-layer forward network is shown in Figs. 4 and 5.

According to the data processing module, the nine vectors are shown in Table 1:

Therefore, the number of neurons in the input layer of the network is 9. And because there are only two output results, attacked and normal, the number of neurons in the output layer is only one. Here, the output is set 1 when attacked, and the output is 0 when the system is normal. Since the actual output of the neural network is generally not an integer of 0 or 1, it is assumed that the system is attacked (or normal) when the output value approaches 1 (or 0) within a certain accuracy range.

Commonly used BP improvement algorithms mainly include additional momentum method, adaptive learning rate method and momentum-adaptive learning rate adjustment algorithm.

1. Additional momentum method

The additional momentum method allows the network to consider not only the effect of the error on the gradient, but the influence of the trend on the error surface when correcting its weight. The weight and threshold adjustment formula with additional momentum factor is:

$$w_{ij}(k+1) = (1-mc)\eta\delta_i p_j + mc\Delta w_{ij}(k) \tag{3.13}$$

$$\Delta b_i(k+1) = (1-mc)\eta\delta_i + mc\Delta b_i(k) \tag{3.14}$$

The conditions for using the momentum method in the design of training programs are:

$$mc = \begin{cases} 0 & E(k) > E(k-1) * 1.04 \\ 0.95 & E(k) < E(k-1) \\ mc & \text{others} \end{cases} \tag{3.15}$$

$E_k$is the square of the error of the k-th step.

2. Adaptive learning rate

For a particular problem, choosing the right learning rate is not an easy task. It is usually obtained from experience or experiments, but even then, the learning rate that works well at the beginning of the training is not necessarily suitable for later training. The following formula gives an adjustment formula for the adaptive learning rate:

$$\eta(k+1) = \begin{cases} 1.05\eta(k) & E(k+1) < E(k) \\ 0.7\eta(k) & E(k+1) > 1.04E(k) \\ \eta(k) & \text{others} \end{cases} \tag{3.16}$$

E_k is the range of the k-th error squared and the initial learning rate $\eta(0)$ can be very random.

3. Momentum-adaptive learning rate adjustment algorithm

When the aforementioned momentum method is used, the BP algorithm can find the global optimal solution, and when the adaptive learning rate is adopted, the BP algorithm can shorten the training time, and the two methods can also be used to train the neural network.

4 Experience

The two main metrics for evaluating the performance of intrusion detection and alerting systems are detection

Table 3 BP neural network training results

Number of neurons	15	17	20	22	25
Number of training	454	308	226	79	49
Final error	0.002918	0.002930	0.002997	0.002995	0.002980

Table 4 Attack sample detection comparison

Type of data	Number of Neurs				
	15 (%)	17 (%)	20 (%)	22 (%)	25 (%)
Neptune	97.50	95.50	97.50	96.00	98.25
Process table	39.04	36.12	38.47	30.08	43.10
Back	86.50	76.34	81.86	70.25	87.52
APache2	90.40	62.20	84.93	59.53	95.95
Teardrop	76.67	44.44	68.89	38.895	78.89
Warezmaster	53.48	48.37	52.56	46.51	59.07
Sshprocesstable	62.47	53.11	60.22	50.45	65.08
POD	99.49	98.46	99.32	97.95	99.49
Smurf	31.25	34.82	32.14	39.29	23.21
Mailbomb	26.50	18.50	22.50	16.00	28.50
Udpstorm	0	0	0	0	0
Selfping	0	0	0	0	0
Warezsilent	0	0	0	0	0
Syslogd	0	0	0	0	0
Crashiis	0	0	0	0	0
Land	0	0	0	0	0
Dosnuke	0	0	0	0	0

The false positive rate of the normal data samples is shown in Fig. 5

rates and false positive rates. The intrusion detection system should have a high detection rate and a low false positive rate at the same time. Simply pursuing a single indicator cannot effectively improve the performance of the intrusion detection system.

Select test data as shown below in Table 2.

The pair of training times and final errors is shown in Table 3. The detection rate of various attack samples is shown in Table 4.

Network convergence rate was tested. The BP neural network gradient descent algorithm is used, and the convergence curve is shown in Fig. 6.

It can be seen from Fig. 6 that when the network is trained by the default gradient descent algorithm, the number of iterations exceeds 1000 times before the network converges. And the closer the optimal value, the faster the network convergence speed. Using the improved BP algorithm, the convergence curve is shown in Fig. 7.

As can be seen from Fig 7, when the training network is close to optimal, the network converges rapidly and greatly reducing the number of iterations.

The experimental results show that the neural network-based intrusion detection and alarm system has a high detection rate for denial of service attacks with a high time continuity. However, for a transient attack against a protocol or system vulnerability, or for an attack in which the characteristics of the intrusion are presented in the data segment of the packet, the network traffic characteristics and the network connection state are similar to the normal network behavior, but the extracted features cannot reflect it. The characteristics of this attack are therefore cannot be detected. This type of attack exploits feature-based detection and will have good results. For the attacks that do not appear in the training set, there is a high detection rate, which indicates that the artificial neural network-based intrusion detection can detect the unknown attack and the variant of the known attack well. In this respect, its advantages are obvious.

5 Results and discussion

This paper starts from the current network security situation, analyzes the importance and development status

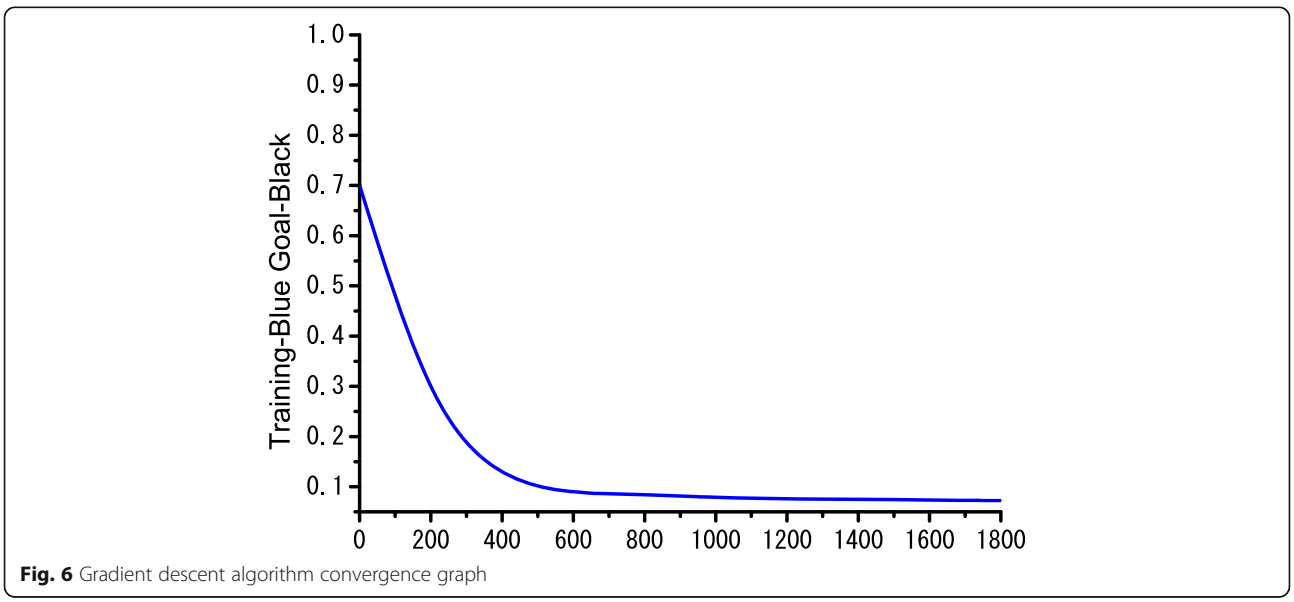

Fig. 6 Gradient descent algorithm convergence graph

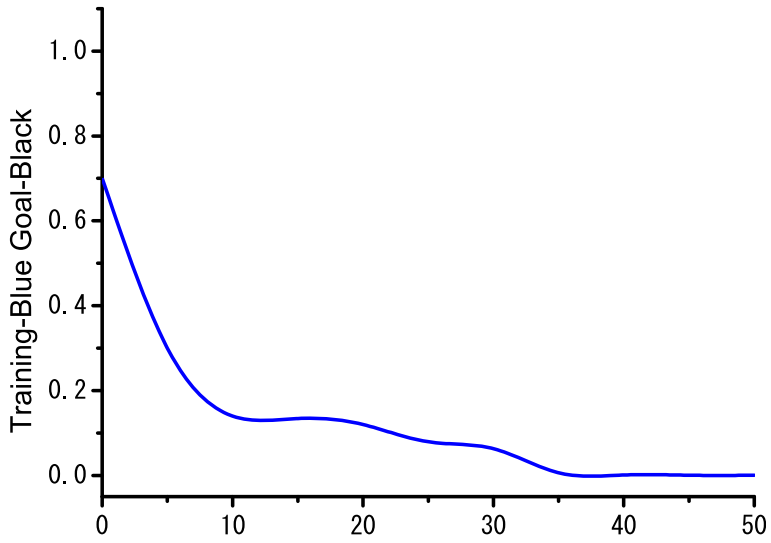

Fig. 7 Improved BP algorithm convergence graph

of intrusion detection, and solves the problems faced by traditional intrusion detection systems in detecting denial of service attacks. Based on the artificial neural network, this paper studies and implements a new intrusion detection of the alarm system. In this intrusion detection system, detection features including attributes related to time, connection characteristics and service types. These features show the difference between attack packets and normal packets, whether it is effective for known attacks or unknown attacks, which greatly improves the detection rate of denial of service attacks. The neural network-based intrusion detection and alarm system is proved to have obvious advantages compared with the traditional intrusion detection system. The system implemented in this paper, which is an attempt and preliminary exploration in the development direction of network intrusion detection. It is hoped that the new anomaly detection model can be applied to the actual security defense system, and the existing network security model is enriched and improved. The intelligent intrusion detection system provides a theoretical and experimental basis.

Abbreviations

PPDR: Public protection and disaster relief; LVQ: Learning vector quantization

About the authors

Yansong Liu, Master of Electronic and communications engineering, Associate Professor. Graduated from the Shandong University in 2009. Worked in Shandong Management University. He research interests include Network behavior analysis, intelligent information processing.
Li Zhu, Doctor of Computer science, Associate Professor. Graduated from the Xi'an Jiao Tong University in 2000. Worked in Xi'an Jiaotong University. He research interests include machine learning, novel computer networking and digital media understanding.

Funding

(1) national natural science foundation youth project, fund no. : 61602370. Research on the measurement and modeling method of information communication laws across social media for network hot events
(2) project of national natural science foundation of China, fund no. : 61773310, research on implicit identity authentication method of mobile intelligent terminal for tactile behavior recognition

Availability of data and materials

The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Author details

[1]Xi'an Jiao Tong University, Xi'an 710049, Shaanxi, China. [2]Shandong Management University, Jinan 250357, Shandong, China.

Received: 28 December 2018 Accepted: 28 March 2019
Published online: 02 May 2019

References

1. F. Hachmi, M. Limam, A. Improved, Intrusion detection system based on a two stage alarm correlation to identify outliers and false alerts. Lect. Notes Comput. Sci **9468**, 130–139 (2015)
2. N. Hubballi, V. Suryanarayanan, Review: False alarm minimization techniques in signature-based intrusion detection systems: A survey. Comput. Commun. **49**(8), 1–17 (2014)
3. G.H. Kim, H.W. Lee, in *International Conference on Computational Science and ITS Applications*. SVM based false alarm minimization scheme on intrusion prevention system (2006)
4. H.J. Liao et al., Intrusion detection system: a comprehensive review. J. Netw. Comput. Appl. **36**(1), 16–24 (2013)
5. O. Mazhelis, S. Puuronen, A framework for behavior-based detection of user substitution in a mobile context. Comput. Secur. **26**(2), 154–176 (2007)
6. A. Mohamed, M. Ahmed, S. Chau, in *IEEE International Symposium on Applied Machine Intelligence and Informatics*. A new adaptive evidential reasoning approach for network alarm correlation (2012)

7. A.A. Ramaki, M. Amini, R.E. Atani, RTECA: real time episode correlation algorithm for multi-step attack scenarios detection. Comput. Secur. **49**, 206–219 (2015)
8. G.A. Barreto et al., Condition monitoring of 3G cellular networks through competitive neural models. IEEE Trans. Neural Netw. **16**(5), 1064–1075 (2005)
9. A.S. Saratikov et al., Interactive wormhole detection and evaluation. Inf. Vis. **6**(1), 3–17 (2007)
10. G.C. Tjhai et al., A preliminary two-stage alarm correlation and filtering system using SOM neural network and K-means algorithm. Comput. Secur. **29**(6), 712–723 (2010)
11. S. Kabiraj, V. Topkar, R.C Walke, Going green: a holistic approach to transform business. Int. J. Manag. Inform. Technol. **2**(3), 22–31 (2010)
12. B. Zhang, X. Wang, Z. Zheng, The optimization for recurring queries in big data analysis system with MapReduce. Futur. Gener. Comput. Syst. (2017). https://doi.org/10.1016/j.future.2017.09.063
13. S. Yao, A.K. Sangaiah, Z. Zheng, T. Wang, Sparsity estimation matching pursuit algorithm based on restricted isometry property for signal reconstruction. Futur. Gener. Comput. Syst.. https://doi.org/10.1016/j.future.2017.09.034
14. W. Hua, D. Mu, Z. Zheng, D. Guo, Online multi-person tracking assist by high-performance detection. J. Supercomput., 1–19. https://doi.org/10.1007/s11227-017-2202-8
15. Y. Lin, X. Zhu, Z. Zheng, Z. Dou, R. Zhou, The individual Identication method of wireless device based on dimensionality reduction and machine learning. J. Supercomput., 1–18. https://doi.org/10.1007/s11227-017-2216-2
16. Zhigao Zheng, Zunxin Zheng. Towards an improved heuristic genetic algorithm for static content delivery in cloud storage. Comput. Electr. Eng. (2017). 2017–6–28. https://doi.org/10.1016/j.compeleceng.2017.06.011