Computer Physics Communications 180 (2009) 1795–1801

Contents lists available at ScienceDirect

Computer Physics Communications

www.elsevier.com/locate/cpc

A research using hybrid RBF/Elman neural networks for intrusion detection system secure model

Xiaojun Tong [a,*], Zhu Wang [b], Haining Yu [a]

[a] School of Computer Science and Technology, Harbin Institute of Technology, Weihai, 264209, China
[b] College of Information, Harbin Institute of Technology, Weihai, 264209, China

ARTICLE INFO

Article history:
Received 17 August 2008
Received in revised form 2 May 2009
Accepted 6 May 2009
Available online 14 May 2009

PACS:
05.45.+b

Keywords:
Intrusion detection
Hybrid RBF/Elman neural network
Memory of events
Anomaly detection
Misuse detection

ABSTRACT

A hybrid RBF/Elman neural network model that can be employed for both anomaly detection and misuse detection is presented in this paper. The IDSs using the hybrid neural network can detect temporally dispersed and collaborative attacks effectively because of its memory of past events. The RBF network is employed as a real-time pattern classification and the Elman network is employed to restore the memory of past events. The IDSs using the hybrid neural network are evaluated against the intrusion detection evaluation data sponsored by U.S. Defense Advanced Research Projects Agency (DARPA). Experimental results are presented in ROC curves. Experiments show that the IDSs using this hybrid neural network improve the detection rate and decrease the false positive rate effectively.

1. Introduction

With the rapid development of internet, the security of internet becomes a vital problem. In order to prevent hacks from intruding internet systems, intrusion detection becomes one of the "hottest" subjects. Intrusion detection systems detect and prohibit the abnormal actions.

By building profiles of authorized computer users, the neural network can be trained to classify the incoming computer traffic into authorized traffic or not authorized traffic. Current research in the area of Intrusion Detection based on Neural Networks shows encouraging results [1–4]. The Intrusion Detection based on Neural Networks can improve the detection rate efficiency and strengthen the ability of self study. But there are also many problems that need to be solved.

An MLP neural network is used widely in intrusion detection systems. But an MLP neural network cannot restore the memory of past events and it takes a long time to train an MLP neural network because of its non-linear mapping of global approximation [5]. An RBF/Elman hybrid neural network is presented in this paper to solve the above problem. An Elman neural network is used to restore the memory of past events. An RBF neural network uses a local recessionary exponential function to approximate the non-linear input and output locally [6,7]. Compared with the MLP neural network, an RBF neural network takes less time to be trained.

A hybrid neural network is presented for both misuse and anomaly detection in this paper. The approach is evaluated against the DARPA intrusion detection evaluation database. The experiments' results demonstrate that the hybrid neural network can detect intrusions with higher detection rate and lower false positive rate than current other IDSs based on a neural network.

The rest of this paper is organized as follows. Section 2 designs an intrusion detection system using the hybrid RBF/Elman neural network. In Section 3, the performance of the hybrid RBF/Elman neural network is evaluated via both theoretical analysis and experimental tests. Finally, Section 4 concludes the paper.

2. Using neural network for intrusion detection system

2.1. Encode the data for input to the network

Neural network has been used widely for intrusion detection system. Compared with those early works on "individual behavior", here a hybrid neural network is presented to distinguish normal behaviors from abnormal behaviors by the profile of "software behavior" which means the profile of privileged processes' system calls.

In order to use a neural network for an intrusion detection system, five major issues have to be addressed: how to encode the

* Corresponding author.
E-mail address: tong_xiaojun@163.com (X. Tong).

0010-4655/$ – see front matter
doi:10.1016/j.cpc.2009.05.004

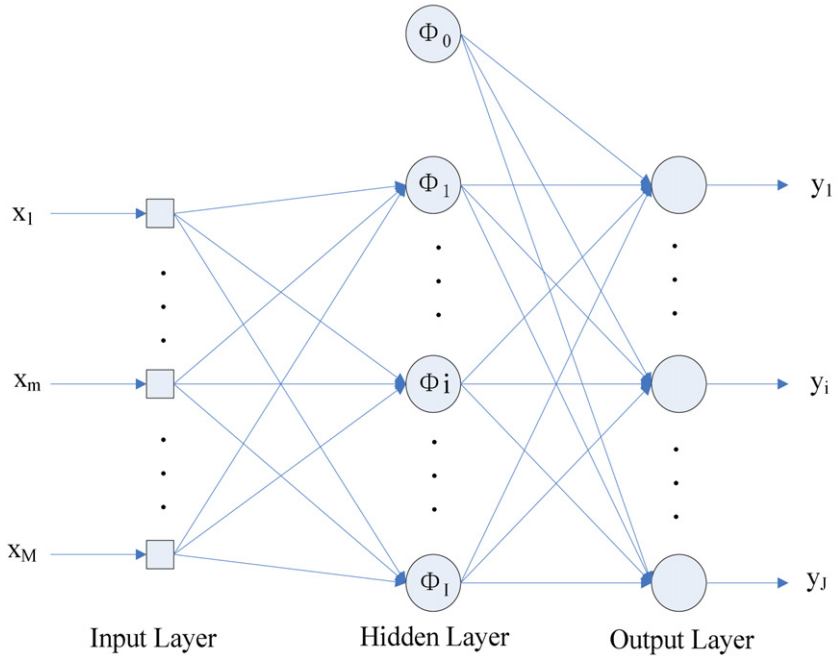

Fig. 1. RBF network in pattern classification.

data for input to the network, what network topology should be used, how to train the networks, how to perform anomaly detection with a supervised training algorithm, and what to do with the data produced by the neural network.

To solve the above issues, the format of BSM audit data have to be recognized. A piece of BSM audit data record contains seven fields at most [8]. These fields hold full information for every "software behavior". To use a neural network for intrusion detection, we have to encode BSM audit data to vectors that a neural network can input [9]. The steps is as follows:

(1) Get the trace of privileged processes' system calls.
(2) Slide a window of size "L" across the trace, recording each unique sequence of length "L" that is encountered.
(3) Encode each unique sequence into a string consisted of "0" and "1", recording each string.
(4) Pick up several "exemplar" strings from the strings, each "exemplar" string is different from each other, its value of the distance metric with other strings is bigger.
(5) To encode a string of data, the distance metric is used to measure the distance from the data string to each of several "exemplar" strings. The encoding then is consisted of a set of measured distances. A string could then be thought of as a point in a space where each dimension corresponds to one of the exemplar strings, and the point is mapped in the space by plotting the distance from each dimension.

2.2. RBF neural network

The radial basis function (RBF) neural network is a kind of feed forward neural network. RBF neural network is embedded in a three layers neural network, where each hidden unit implements a radial activated function. The input units implement the data input to the network. The output units implement a weighted sum of hidden unit outputs. The input into a RBF network is non-linear while the output is linear. Due to their non-linear approximation properties, RBF networks are able to model the complex mappings, which perception neural network can only be modeled by means of multiple intermediary layers [6].

An RBF neural network bases on the Cover Theorem. The probability that classes are linearly separable increases when the features are non-linearly mapped to a higher dimensional feature space. Radial basis functions are embedded into a three-layer feed-forward neural network. Such a network is characterized by a set of inputs and a set of outputs. Between the inputs and outputs there is a layer of processing units which is called hidden units. Each of them implements a radial basis function. The network is used as a pattern classification, as shown in Fig. 1.

As shown in Fig. 1, the number of units is M, I and J in input layer, hidden layers and output layers individually. The basis function of the number "i" unit in hidden layer is $\phi_i(X, t_i)$, and $t_i = [t_{i1}, t_{i2}, \ldots, t_{iM}]$ for $i = 1, 2, \ldots, I$, which is the center of basis function. The weights between hidden units and output units are presented as w_{ij}.

On the assumption that the set of training data is $X = [X_1, X_2, \ldots, X_k, \ldots, X_N]^T$, where a piece of training data is $X_k = [x_{k1}, x_{k2}, \ldots, x_{km}, \ldots x_{kM}]^T$, for $k = 1, 2, \ldots, N$. Real output is $Y_k = [y_{k1}, y_{k2}, \ldots, y_{kj}, \ldots, y_{kJ}]^T$, for $k = 1, 2, \ldots, N$. The output is $d_k = [d_{k1}, d_{k2}, \ldots, d_{kj}, \ldots, d_{kJ}]^T$ for $k = 1, 2, \ldots, N$. When input is X_k, the real output of the number of "j" unit in output layer is given by:

$$y_{kj}(X_k) = w_{0j} + \sum_{i=1}^{I} w_{ij}\phi(X_k, t_i), \quad j = 1, 2, \ldots, J \tag{1}$$

The Gaussian activation function is used as basis function in this paper, which is given by:

$$\phi(X_k, t_i) = G(\|X_k - t_i\|) = \exp\left(-\frac{1}{2\sigma_i^2}\|X_k - t_i\|\right) = \exp\left(-\frac{1}{2\sigma_i^2}\sum_{m=1}^{M}(x_{km} - t_{im})^2\right) \tag{2}$$

where $t_i = [t_{i1}, t_{i2}, \ldots, t_{iM}]$ is the center of the Gaussian activation function, and σ_i is the variance of Gaussian activation function.

RBF network has to learn three parameters: the center of radial basis function, the variance of radial basis of function and the weight. Here the self-organizing learning algorithm of selecting the

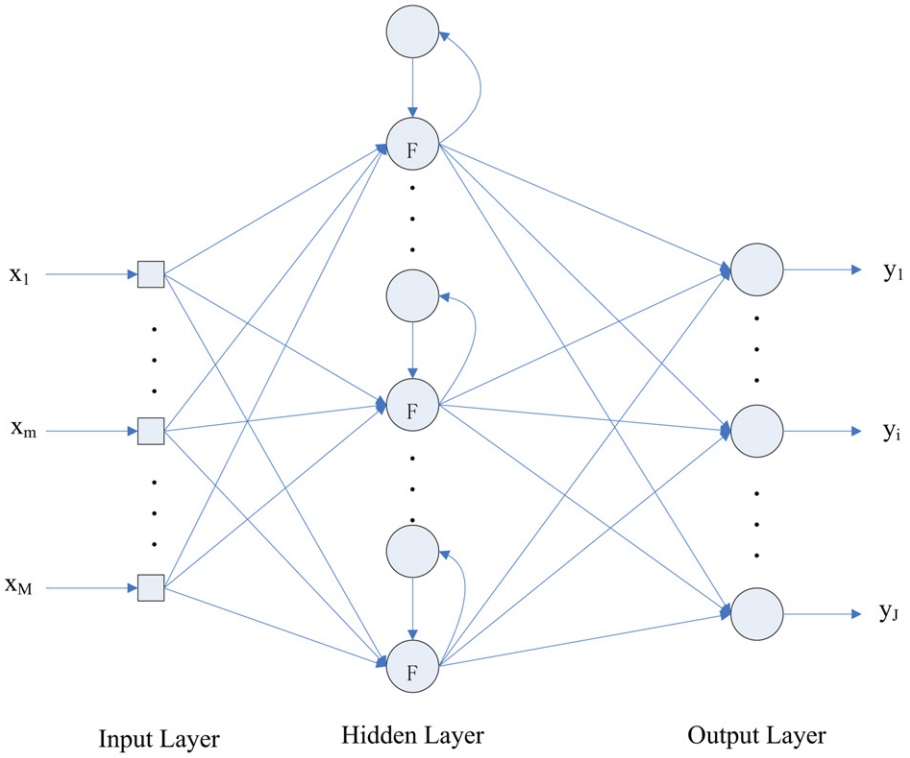

Fig. 2. Elman network in memory of past events.

radial basis function center is used in this paper. The algorithm contains two steps: the first step is self-organizing learning, and the second step is to train the network with LMS (Least-Mean-Square) algorithm.

2.3. Elman neural network

As shown above, the neural network intrusion detection systems must have the ability of keep memory of recent events in order to identify dispersed and collaborative attacks. The Elman neural network fits this purpose well.

Elman recurrent network is a well-known recurrent topology, developed by Jeffrey Elman. An Elman network has a set of context nodes. Each context node receives input from a single hidden node and sends its output to each node in the layer of its corresponding hidden node. Since the context nodes depend only on the activations of the hidden nodes from the previous input, the context nodes retain state information among inputs. The Elman neural network used in this paper is shown as Fig. 2. Here we use the dynamic BP algorithm to train the Elman neural network [7].

The expression of Elman non-linear state space is as follows:

$$\begin{cases} x(t) = f\left(W^A x_c(t) + W^B u(t-1)\right) \\ x_c(t) = x(t-1) \\ y(t) = g\left(W^C x(t)\right) \end{cases} \tag{3}$$

where $x(t)$ is the output of hidden layer, $y(t)$ is the output of output layer, $u(t-1)$ is the input of Elman network, W^A is the weight of connection between context units and hidden layer, W^B is the weight of connection between input layer and hidden layer, W^C is the weight of connection between hidden layer and output layer, $f(.)$ and $g(.)$ are the activation functions for hidden layer and output layer.

The activation functions which is sigmoid function is given by:

$$f(x) = \frac{1}{1+e^{-ax+b}} \tag{4}$$

$$g(x) = kx \tag{5}$$

Formula (3) deduces that

$$x_c(t) = x(t-1) = f\left(W^A_{t-1} x_c(t-1) + W^B_{t-1} u(t-2)\right) \tag{6}$$

$x_c(t)$ depends on weight W^A_{t-1}, W^B_{t-1} which comes from different time. This deduction is a dynamic process.

2.4. Hybrid RBF/Elman neural network

An RBF neural network demonstrates the potential of its detecting individual instances of possible misuse. However, most of attacks are composed of a series of misuse events. New prototypes must be designed to identify temporally dispersed and possibly collaborative attacks, such as DOS or probing.

Elman networks are employed to keep memory of events as they occur in a large stream of events. The hybrid RBF/Elman neural network model takes an output of RBF as input of an Elman network, so that the number of input units of Elman network must be equivalent to the number of output units of RBF network. An Elman network can keep memory of past misuse events. Therefore, each output of RBF network can be restored by a Elman network. When a classification result of an input is analyzed by RBF, the result will be feed forward and can be restored by the Elman network connected to the output units of RBF network. The hybrid network realizes classification with memory of recent events using the real-time classification of RBF network and the memorial functionality of Elman network. The topology of the hybrid RBF/Elman neural network is shown in Fig. 3.

In performance, the context nodes of Elman are initially set to 0. Recurrent connection weights, which are from context nodes to hidden nodes, are fixed values from 0 to 1. Processing consists of the following sequence of events. Both the input nodes and context nodes activate the hidden nodes, and then the hidden nodes feed forward to activate the output nodes. At time t, the first input

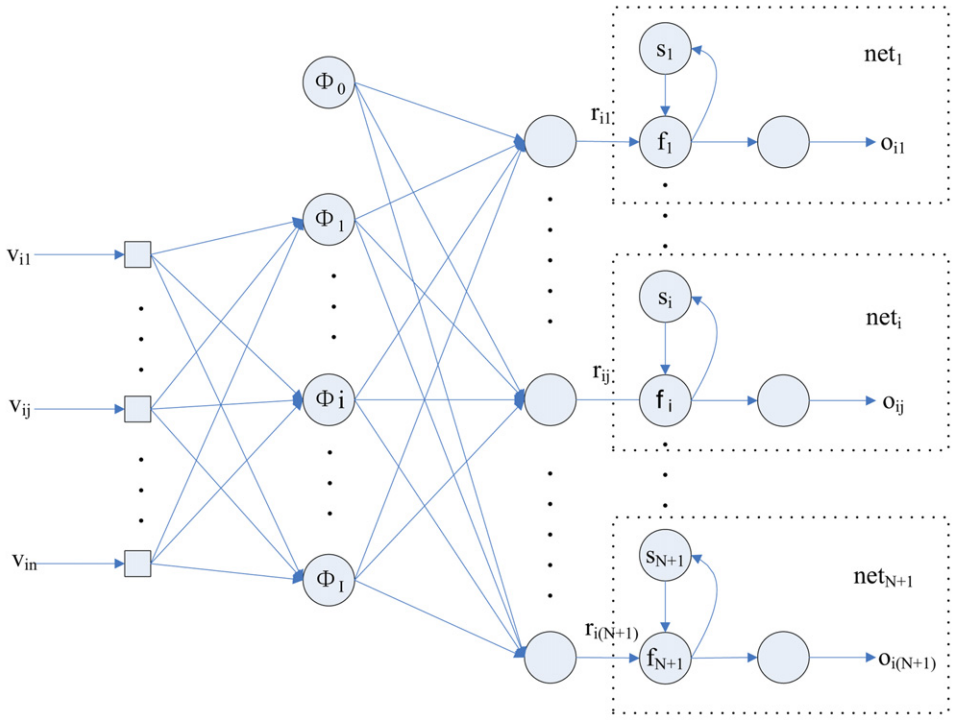

Fig. 3. The hybrid RBF/Elman neural network model.

in the sequence is received by the input nodes. The hidden nodes also feed back to activate the context nodes. This constitutes the forward activation. At the next time step $t+1$ the above sequence is repeated. This time the context nodes contain values, which provide the network with memory.

The hybrid RBF/Elman neural network keeps a memory of recent events by incrementing the value of the context node of Elman network, while slowly decreasing its value. When the RBF network identifies a misuse intrusion, the Elman network will quickly accumulate a large value in its context unit. Similarly, when the RBF network identifies a normal output, the Elman network will decrease the value of its context back down to zero. As a result, the hybrid neural network emphasizes misuse intrusions that are closely temporally co-located and diminishes the values of those that are sparsely located. If the value of the context node rises the above threshold, an attack is considered to be appearance.

The advantage of using an Elman network is that it allows occasional misuse behavior, which is to be expected during normal system operation, but it is sensitive to large numbers of temporally co-located misuse events, which one would expect if an attack really happens. The RBF/Elman approach is similar to the functionality of leaky bucket algorithm used in [4], while the difference between the two approaches is that the prior one is a complete neural network approach, but leaky bucket algorithm is not.

Through the above analysis, the hybrid RBF/Elman neural network model can be used in both anomaly detection and misuse detection. The number of units in output layer is "$N+1$", one presents for abnormal behavior. The others are used for misuse detection, where N is equivalent to the number of different kinds of misuse behaviors. Each of the units presents for a kind of misuse behavior.

2.5. Hybrid RBF/Elman neural network algorithm

The set of vector got from preprocessor is

$$V = \{v_1, v_2, \ldots, v_i, \ldots, v_k\} \tag{7}$$

Select one from the set at random is

$$v_i = [v_{i1}, v_{i2}, \ldots, v_{ij}, \ldots, v_{in}]^T \tag{8}$$

where "n" is the number of input units of the RBF network. With the input V_i, the output of RBF network is

$$R_i = [r_{i1}, r_{i2}, \ldots, r_{ij}, \ldots, r_{i(N+1)}]^T \tag{9}$$

where "$N+1$" is the number of output units of the RBF network. The weight connects the outputs of the RBF to the input of the Elman is

$$W = (w_1, w_2, \ldots, w_i, \ldots, w_{N+1})^T \tag{10}$$

The recurrent weights of the Elman are

$$W' = (w'_1, w'_2, \ldots, w'_i, \ldots, w'_{N+1})^T \tag{11}$$

The number of Elman network is "$N+1$", which is

$$Net = (net_1, net_2, \ldots, net_i, \ldots, net_{N+1})^T \tag{12}$$

The values in context units of Elman is

$$S = [s_1, s_2, \ldots, s_i, \ldots, s_{N+1}]^T \tag{13}$$

With the value given above, the output of the Elman is

$$O_i = [o_{i1}, o_{i2}, \ldots, o_{ij}, \ldots, o_{i(N+1)}]^T \tag{14}$$

The thresholds for each alert:

$$T = (t_1, t_2, \ldots, t_i, \ldots, t_{N+1})^T \tag{15}$$

The approach to get O_{ij} is given by:

$$o_{ij} = r_{ij} * w_i + s_i * w'_i \tag{16}$$

Table 1
Modules description.

Modules	Realization
Sensor module	DARPA BSM Data Set I/O reader
Preprocessor module	Map the short system calls to vector used for neural network
Pattern classification module	RBF neural network
Event memory module	Elman neural network

Input: The BSM audit data
Output: Intrusion alert
Algorithm:

1. process the BSM audit data to retrieve set V, where each item is a vector
2. build a hybrid network, whose outputs number is $N+1$, one used for anomaly detection and the others used for misuse detections
3. for $V_i \in V$ do
4. RBF process V_i to get a result R_i
5. for $v_{ij} \in V_i$ do
6. if $r_{ij} <= t_i$ then $s_i = 0$
7. else then $\Delta s = r_{ij} * w_i$, $s_i += \Delta s$
8. end if
9. $o_{ij} = r_{ij} * w_i + s_i * w'_i$
10. set net_1 used for anomaly detection
11. set $net_2 \sim net_{N+1}$ used for misuse detection
12. for $o_{ij} \in O_i$ do
13. if $o_{ij} > t_i$
14. if $j == 1$ then output an anomaly alert
15. else output a misuse alert,
16. type is which "j" mapped
17. end if
18. end if
19. end for
20. end for
21. end for

2.6. A modularization model

The hybrid model is consisted of several modules, several sensor modules are used to collect data set, a preprocessor model, a pattern classification module and several event memory modules. Every module is packaged with the O-O (Object-oriented) theory.

The hybrid model is easy to be extended and updated. For example, if a new intrusion event occurred, the pattern classification module that adapts the new event is trained and a new interface for the event is produced. A new memory module is used to alert this intrusion events connect to the interface. In this paper each module is shown in Table 1.

SVMs, MPM, soft computing and other pattern classification technology can be used as a pattern classification module to instead of RBF neural network. New technology may reach more accurate results [12]. RBF neural network is employed as a pattern classification module, it also works effectively. In this paper, hybrid model is the most important point. This model improves the capability of IDS effectively. The model is not sensitive to occasional misuse behavior, but it is sensitive to temporally co-located intrusion events. The model also has the ability to detect temporally dispersed and possibly collaborative attacks, such as DOS or probing.

3. Experimental results

The anomaly and misuse detection systems were tested on the same test data (1999 DARPA Intrusion Detection Evaluation Data Sets). There are 128 intrusive sessions in the data set [10,11].

Since the optimal number of hidden nodes for a program was not known before training, for each program, networks were trained with 10, 20, 25, 30, 35, 40, 50, and 60 hidden units. Before training, network weights were initialized randomly. However, initial weights can have a large values, but unpredictable, effect on the performance of network trained. In order to avoid poor performance due to bad initial weights, for each program, for each number of hidden nodes, 5 networks were initialized differently and trained. Therefore, for each program, 40 networks were trained. To select one of the 40 to keep, each was tested on DARPA Data Sets which is not used for training. The network can classify data most accurately and the results can meet the standards. DARPA Intrusion Detection Evaluation Data Sets contain many kinds of intrusion events. The data sets can be a standard to evaluate the precision of neural network. So the network chose can work well in this system.

The sensitivity of the system can be easily changed, when the recurrent connection weight and the threshold of the output nodes are varied. A recurrent connection weight of 1 results in all past events being retained in memory. A recurrent connection weight of 0 results in all of the past events but the current one being forgotten. The recurrent connection weight can be varied from 0 to 1.

The performance of the IDS should be judged in terms of both the ability to detect intrusion events. We used receiver operating characteristic (ROC) curves to compare intrusion detection ability to false positives rate. A ROC curve is a parametric plot, in which the parameter is the sensitivity of the system to what it perceives to be insecure behavior. The curve is a plot of the likelihood that an intrusion is detected, against the likelihood that a non-intrusion is misclassified for a particular parameter, such as a threshold. The ROC curve can be used to determine the performance of the system for any possible operating point. The ROC curve allows the end user of an intrusion detection system to assess the trade-off between detection ability and false positive rate in order to properly tune the system for acceptable tolerances.

Following the three steps can get a better result:

(1) Test several different the RBF neural networks with different hidden layer units numbers and each network was initialized differently for several times.
(2) Adjust the weights of the neural network by training the neural network under a high precision.
(3) Adjust the thresholds used in this system, the best one is kept.

By above means, the IDS can reach a better detection result both in anomaly detection and misuse detection.

Different recurrent connection weights produced different ROC curves. Fig. 4 displays two ROC curves of anomaly detection, for the recurrent connection weight is 0.75, a detection rate of 91.4% can be achieved with a false positive rate of only 3.1%. When the recurrent connection weight is 0.25, detection rate is 93%. At 93% detection rate, the false positive rate is only 2.6%.

Fig. 5 displays two ROC curves of misuse detection, for the recurrent connection weight of 0.75, detection rate is 94.1%. At 94.1% detection rate, the false positive rate is 2.3%. When the recurrent connection weight is 0.25, a detection rate of 95.3% can be achieved with a false positive rate of 1.4%. The best detection results for each type of intrusion in Solaris are shown in Table 2.

Compared with the results of the other three neural network models presented in [5] and SOM (self-organizing map)/RPROP (resilient back propagation) presented in [13], the results are shown in Table 3. Compared with the other three neural network models, the detection rate of the hybrid network is the highest and the false positive rate is the lowest. The result of SOM/RPROP is closed to the hybrid network, but it is only used for misuse detection.

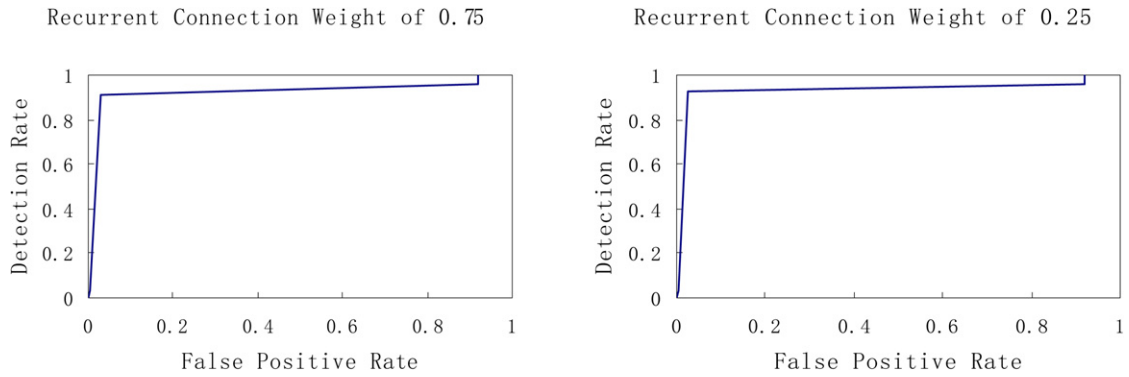

Fig. 4. Anomaly detection results for two different recurrent connection weights.

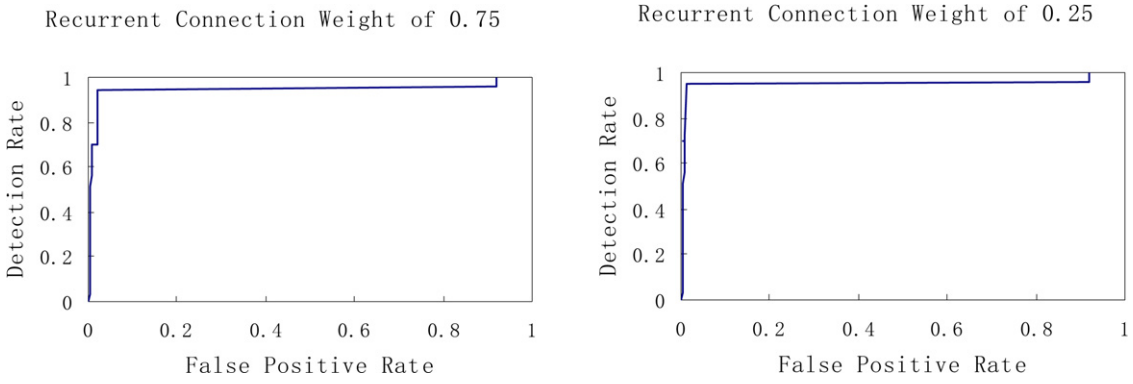

Fig. 5. Misuse detection results for two different recurrent connection weights.

Table 2
Detection results.

Name	Category	Total instances	Instances detected
portsweep	Probe	13	13
ipsweep	Probe	11	11
queso	Probe	8	8
illegal-sniffer	Probe	14	14
neptune	DoS	11	10
pod	DoS	4	4
processtable	DoS	1	1
smurf	DoS	5	3
dosnuke	DoS	4	3
selfping	DoS	5	5
syslogd	DoS	4	4
tcpreset	DoS	3	3
warezclient	DoS	3	3
dict	R2L	5	5
ftpwrite	R2L	3	3
guest	R2L	4	4
httptunnel	R2L	4	2
xlock	R2L	3	3
xsnoop	R2L	6	6
eject	U2R	3	3
fdformat	U2R	3	3
ffbconfig	U2R	4	4
ps	U2R	3	3
secret	DATA	4	4
Total		128	122

The hybrid network can improve the detection rate and decrease false positive rate effectively in both anomaly detection and misuse detection. Through the above results, the hybrid RBF/Elman neural network model can get higher detection rate and lower false positive rate that both are used for misuse detection and anomaly detection.

Table 3
Detection rate and false positive rate of four different network.

Types	Anomaly detection		Misuse detection	
Neural network model	Detection rate	False positive rate	Detection rate	False positive rate
MLP/leaky bucket	77.3%	2.2%	90.9%	18.8%
Elman/leaky bucket	77.3%	0%		
SOM/RPROP			95%	4.6%
RBF/Elman	93%	2.6%	95.3%	1.4%

Table 4
Detection rate and false positive rate for DOS and probing of three different network.

Neural network model	DOS		Probing	
	Detection rate	False positive rate	Detection rate	False positive rate
MLP /leaky bucket	35%	0%	29%	0%
Elman/leaky bucket	85%	0%	88%	0%
Snort	64.6%	0.32%	75%	0%
RBF/Elman	87.5%	0%	100%	0%

The hybrid neural network can detect the collaborative attacks effectively. Compared with other systems, the results for detecting DOS or probing are shown in Table 4.

4. Conclusions

In this paper, a hybrid RBF/Elman neural network is presented, which has four characteristics.

(1) It realizes classification with memory of recent events using the real-time classification of RBF and the memorial functionality of Elman network.

(2) It is a complete neural network approach, which is different from leaky bucket algorithm.
(3) The sensitivity of the system can be easily configured, which allows end users to tune the system for acceptable tolerances without having to retrain the neural network.
(4) This hybrid model is a modularization model, it's prone to extend and replace each module.

The capability of IDSs to identify DOS and probing attacks is enhanced. The results are shown in this paper from evaluating hybrid RBF/Elman neural network intrusion detection approaches in the Lincoln Laboratory/DARPA Intrusion Detection evaluation. The results demonstrate that the hybrid neural network can detect intrusions with higher detection rate and lower false positive rate than current neural network IDSs.

Acknowledgements

This work were supported by the Shandong Province Key Natural Science Foundation of China through the Grant number Z2006G01 and the Shandong Provincial Science and Technology Plan Project Foundation of China through the Granted number 2006110.

References

[1] A. Bivens, C. Palagiri, R. Smith, B. Szymanski, et al., Network-based intrusion detection using neural networks, in: Proceeding of ANNIE-2002, ASME Press, New York, 2002, pp. 579–584.
[2] Yang Ke, Wang Li-Ping, Fang Ding-Yi, Program behavior anomaly detection based on neural network, Dalian Ligong Daxue Xuebao (Journal of Dalian University of Technology) 45 (Suppl.) (2005) S136–S141.
[3] Morteza Amini, Rasool Jalili, Hamid Reza Shahriari, RT-UNNID: A practical solution to real-time network-based intrusion detection using unsupervised neural networks, Computers & Security 25 (6) (2006) 459–468.
[4] Guisong Liu, Zhang Yi, Shangming Yang, A hierarchical intrusion detection model based on the PCA neural networks, Neurocomputing 70 (7–9) (2007) 1561–1568.
[5] A.K. Ghosh, A. Schwartzbard, A study in using neural networks for anomaly and misuse detection, in: Proceedings of 8th USENIX Security Symposium, USENIX Association, San Washington, 1999, pp. 23–36.
[6] Felipe Miguel Aparicio Acosta, Radial basis function and related models: An overview, Signal Processing 45 (1) (1995) 37–58.
[7] Adem Kalinli, Seref Sagiroglu, Elman network with embedded memory for system identification, Journal of Information Science and Engineering 22 (6) (2006) 1555–1568.
[8] Sun Microsystems, Sun SHIELD Basic Security Module Guide, http://docs-pdf.sun.com/802-1965/802-1965.pdf.CA, 1995.
[9] S.A. Hofmeyr, S. Forrest, A. Somayaji, Intrusion detection using sequences of system calls, Journal of Computer Security 6 (3) (1998) 151–180.
[10] R.K. Cunningham, R.P. Lippmann, D.J. Fried, et al., Evaluating intrusion detection systems without attacking your friends: The 1998 DARPA intrusion detection evaluation, in: Proceedings of Third Conference and Workshop on Intrusion Detection and Response, San Diego, CA, 1999.10-21.
[11] R. Lippmann, J.W. Haines, D.J. Fried, et al., The 1999 DARPA off-line intrusion detection evaluation, Computer Networks 30 (2) (2000) 14–26.
[12] Srinivas Mukkamala, Guadalupe Janoski, Andrew Sung, Intrusion detection using neural networks and support vector machines, in: Proceedings of the International Joint Conference on Neural Networks, vol. 2, 2002, pp. 1702–1707.
[13] C. Jirapummin, N. Wattanapongsakorn, P. Kanthamanon, Hybrid neural networks for intrusion detection system, in: Proceedings of the International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC 2002), Thailand, July 2002, pp. 928–931.