International Journal of Security and Its Applications
Vol.8, No.5 (2014), pp.227-242
http://dx.doi.org/10.14257/ijsia.2014.8.5.21

A SVM-based IDS Alarms Filtering Method

Yun Liu *, Kun-Peng Xia and Jian-Xun Zhao

School of Electronic and Information Engineering
Key Laboratory of Communication and Information Systems, Beijing Municipal Commission of Education
Beijing Jiaotong University, Beijing, 100044， China
bshen@bjtu.edu.cn, 610092885@qq.com, zhaojianxun124@gmail.com

Abstract

In view of the existing IDS are widespread the problem of high false alarm rate, this paper proposes a kind of alarm information filtering method of IDS based on support vector machine (SVM). The method consists of two parts, training, and data prediction. Model training including parsing command line parameters, read the training sample, select the appropriate penalty coefficient, kernel function and kernel parameter, statistical types and the number of each type of sample, sample training data grouping, using the minimum sequence optimization algorithm C - SVM classifier model. Training data to predict including read alarm data and based on the model of C - SVM classifier model calculation values of decision alarm data. Theoretical analysis and experimental data show that the rational selection of kernel function and kernel parameters and the training data set, this method can effectively reduce the intrusion detection system false alarm rate.

Keywords: *Network Security, Threat Traceback , Intrusion Detection, SVM*

1. Introduction

The Internet and its derived various applications in people's life are playing a more and more important role with the rapid development of information technology. At the same time, various network threats also bring a very serious challenge. In recent years, the spread of the virus, information disclosure, data theft and other forms of network threat caused by the vulnerability damages to the network data confidentiality, integrity and availability seriously.

The Research results of the 360 Internet Security Research Center and the Gartner Group show that: the main task of the current network security equipment is to filter the illegal application and to prevent and detect the threatening behavior. The traditional passive safety protection mechanism will be gradually failure with the rapid development of network threat technology, especially the advanced persistent threats (APT) to the growing popularity of targeted. Vahid Aghaei-Foroushani's research show us: the reason why network threat situation is becoming more and more serious is because the lack of effective threat Traceability Technology and The proliferation of attack tool on the Internet cause the hackers to attack the network without risk of being discovered. As long as we establish an effective traceability mechanism for threatening behavior, most of the network threats will disappear. In summary, to propose an efficient and reliable threat Traceability Technology has a very positive meaning.

ISSN: 1738-9976 IJSIA

2. Research Status

Currently, research on threat information traceability technology focused on how to improve the real-time processing of threat detection ability, how to reduce the threat detection engine's false alarm rate and how to effectively trace back the attack.

a) How to improve the real-time processing of threat detection ability.

Jung-Sik Sung.et al proposed a deep packet inspection algorithm base on TCAM (Ternary Contents Addressable Memory, TCAM), which can handle gigabit network traffic at a 9 Mbit TCAM. Tian Song, *et al.*, proposed a kind of ACC algorithm based on CDFA(Cached Deterministic Finite Automate) and AC algorithm, which uses NSA(Next State Addressing) technology, which is take up less memory, to store the Transition Rules of FSA (finite state automata, FSA), greatly improves the efficiency of memory and detection engine ability of real-time processing. Zhiping Cai.et al proposed a kind of Uniformed Parallel Detection.

Architecture (UPDA) which is based on traffic division. The test results about the high performance of IDS based on NetMagic and UPDA show that: The detection effect of the system compared with the traditional single engine IDS were similar, but the packet loss rate (less than 1.6%) is far lower than the traditional single engine IDS’s(6.64%).

b) How to reduce the threat detection engine's false alarm rate.

There is a widespread problem of high rate of false positives in the existing IDS because of following the principle of” Rather than false positives, not omission”. A huge number of false positives information greatly reduce the value of the alarm information, and flood the real alarm data. Ziyan Qin.et al proposed the alarm information correlation model after they researched and analyzed the causes of high false alarm rate detection engine. The model outputs real alarm message by pretreatment, aggregation and association, extraction and integration of three steps. Ying-Dar Lin. et al proposed a Creditability-based Weighted Voting (CWV) Scheme after they studied the rate of false positives and non-response rates of distributed IDS. In this scheme, they distribute different credibility to intrusion detection system In different locations, measure critical level of IDS alarm message In different locations by using the comprehensive results of IDS credibility and IDS alarm events, effectively solve the problem of distributed IDS high false alarm rate by filtering out those critical low-level alarm. Gupta D., *et al.*, proposed the Post-processor for IDS alerts using Knowledge-based Evaluation (PIKE). The system scores the context (*e.g.*, operating system type, server type, pre-installed application, *etc.*,) and knowledge base (*e.g.*, exploitable vulnerabilities in device, the pattern of exploiting the vulnerabilities and The damage after the fall of equipment, *etc.*,) firstly, then, determine whether an alarm is a high-risk alarm according to the relationship between alarm points and threshold value set by the system.

c) How to effectively trace back the attack.

The ultimate goal of threat trace back is to locate the attack position and reconstruct attack path, currently the focus of the research direction is how to make use of logging, packet marking, ICMP tracing method and the methods of attack graph to effectively restore attack path and accurately locate the position of attackers through the router. Vahid Aghaei Foroushani, *et al.*, proposed an IP tracing method based on Deterministic Flow Marking (DFM). The method can accurately locate the attack position, even if the attacker used the fake source IP address, network address translation or proxy server. This method has the advantage of strong extensibility, easy to implement, can almost thousands of distributed real-time tracking attack, and the disadvantage is that it takes a lot of CPU and memory. The Intention-driven iTrace Model (a method against the DOS attack and distributed ICMP, which is proposed by Alireza Izaddoost, *et al.*,) use the Intention flags to decide whether to produce the ICMP traceback packet. As long as the victim end collected enough ICMP traceback packets like this, we can accurately back the attack source location. Chunying Wang, *et al.*, proposed a technology that can automatically generate and analyze attack graph, which is based on SMCA (Symbolic Model Checking Algorithm, SMCA) and Attack Graph. The advantages of this technology is a high degree of automation, trace back effect is also very good, but need access to the entire network topology, once an attacker to break the deployment of the system can access the entire network.

3. A SVM-based IDS Alarms Filtering Method

3.1. Analysis of the Causes of High False Alarm Rate

a) Mode matching rules is not perfect.

Mode matching rule is the core of the IDS based on feature signature, imperfect matching rules will not only result in under-reporting of real aggression, still can cause to the aggression of a large number of false positives. While due to the mode matching rules are written by experienced information security professionals and through strict performance tests before the official launch, so the false alarm which is caused by imperfect matching rules in the share is not large proportion of all false alarms. With the application and development of regular expression in the mode matching rules and the gradually improvement of mode matching algorithm, this kind of false alarm will be less and less.

b) There are indeed attack packets in the net flow, but they will not cause harm to the target network.

For example, an attacker use a particular vulnerability to attack, the attack should only be harmful to a specific kernel version of Linux system, the use of other version of the kernel of Linux and Windows system will not cause any influence. If IDS detect the attack taking advantage of the vulnerability, System will give alarm even if there is no host which is installed in the Linux system in the network environment the system protects. Obviously, this is a false alarm and this is a main cause of high false alarm rate. While we can use the alarm information correlation model, machine learning and other techniques to secondary alarm information filter to reduce the high rate of false alarm in the IDS. In this paper, we use machine learning techniques to reduce the high false alarm rate with its advantages of High degree of intelligent, automated ability, high classification accuracy.

3.2. Alarm Filtering and Support Vector Machine (SVM)

IDS alarm filtering is a binary classification problem, in other words, it Screens out the true alarm signal and filters out the false alarm from vast amounts of alarm data. The true alarm signal is small sample relative to the false alarm.

The n vector which is composed of alarm information is non-linear, n refers to the number of characteristic properties of the false alarm information, n = 41 In the KDD99 data set. When n is large, there is a problem of "dimension disaster" if we calculate the vector. Therefore, the general classification algorithm is difficult to realize effective filtering false alarm to IDS alarm message, while, the SVM can solve these problems. SVM is an algorithm based on the classification of the small sample learning. When calculating, linearly non-separable data in low-dimensional Space can be translated into linearly separable data in high-dimensional by adopting the proper kernel function, then accurately classified by using linear classifier in the high-dimensional space. Based on the characteristics of Alarm information small sample, nonlinear and high dimension and SVM's advantages on dealing with data classification problem in small sample, nonlinear and high dimension, we proposes a IDS alert filtering method based on SVM.

3.3. C - SVM Classifier

3.3.1. The Mathematical Model of the C – SVM

A SVM-based IDS alarms filtering method

$(x_i, y_i)$is alarm data set given by detection engine, vector x_i ($x_i \in R^n$) refers the i th alarm data in the alarm message, i = $1,2,3,\cdots,N$, N is the number of alarm data, n refers to the number of characteristic properties of the false alarm information.

If n dimensional vector w and constant b make system of inequalities (3-1) set up,

$$\begin{cases}(w\cdot x_i)+b\ge 0, & y_i=+1\\(w\cdot x_i)+b\le 0, & y_i=-1\end{cases} \qquad (3\text{-}1)$$

We say alarm data set is linearly separable in n dimensional space, hyper plane$(w\cdot x)+b=0$ is classification hyper plane, and classification decision function is:

$$f(x) = sgn((w\cdot x)+b) \qquad (3\text{-}2)$$

Make the distance between hyper plane $(w\cdot x)+b=1$ and $(w\cdot x)+b=-1$ to 2Δ, Δ refers classification interval. Set x_j to a point on the hyper plane$(w\cdot x)+b=0$, Δ is the distance between x_j and $(w\cdot x)+b=1$, $\Delta=\frac{|(w\cdot x_j)+b-1|}{||w||}$, as $(w\cdot x)+b=0$, $\Delta=\frac{1}{||w||}$

We call it the optimal separating hyper plane if the hyper plane (w·x)+b=0 can maximize the classification interval Δ, also known as the Maximum Margin Hyper plane. As shown in Figure 1.

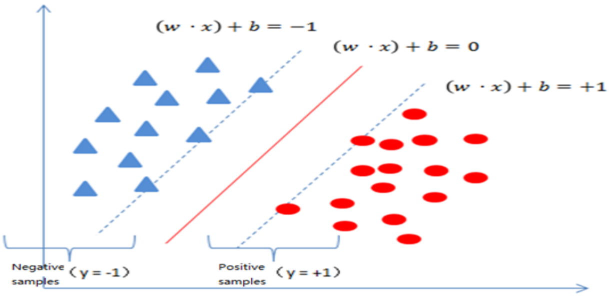


Figure 1. The Optimal Hyper plane

The SVM classifier consists of two model training and data prediction. Model training is to solve the optimal model. Data prediction is the process of classifying data according to the decision function value of the data.

Solving w and b, we get

$$w^*=\sum_{i=1}^{N}\alpha_i^* y_i x_i$$

$$b^*=y_i\left(1 - \left(w^*\cdot x_i\right)\right)=y_i-\sum_{i=1}^{N}\alpha_i^*\left(x_i\cdot x_j\right)$$

Enter the formula 3-2 with w^* and b^*, So the decision function of SVM classifier is:

$$f(x)=sgn\left(\left(w^{*}\cdot x\right)+b^{*}\right)=sgn\left(\sum_{i=1}^{N}\alpha_i^{*}y_i\left(x_i\cdot x\right)+b^{*}\right) \qquad (3\text{-}3)$$

$x_i(i = 1,2,3,\cdots,N)$ is support vector, x refers to sample points to be predicted. The sample points are classified as a true alarm if f(x)>0, as a false alarm if f(x) <0, and directly discarded if f(x)>0.

Introducing kernel function, slack variable and penalty coefficient to SVM classifier to solve the problems of that alarm data in the original input space is nonlinear and that data points deviated from the sample space cause by noise.

The "dimension disaster" problem which appears in the high-dimensional space complex calculations can be avoided by using the kernel function method.

As shown in Figure 2 slack variable ε_i $(\varepsilon_i \geq 0)$ refers to some kind of sample points deviate from the sample space distance $\varepsilon_i = \max(0,1 - y_i f(x_i))$.

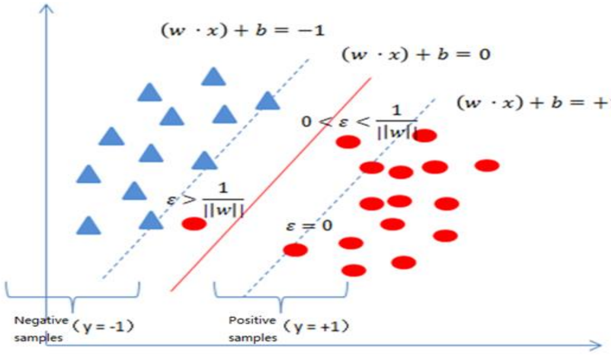

$(w \cdot x) + b = -1$
$(w \cdot x) + b = 0$
$(w \cdot x) + b = +1$
$0 < \varepsilon < \frac{1}{||w||}$
$\varepsilon > \frac{1}{||w||}$
$\varepsilon = 0$
Negative samples (y = -1)
Positive samples (y = +1)

Figure 2. The Slack Variables

Penalty coefficient C can control objective function to achieve balance between “Looking for maximum classification hyper plane” and “ensuring the data point deviation is minimal”. The value of C refers to the tolerance to slack variable.

After the introduction of slack variable and penalty coefficient, decision function of the SVM classifier is:

$$f(x)=sgn\left(\sum_{i=1}^{N}\alpha_i y_i K\left(x_i, x\right)+ b\right) \qquad (3\text{-}4)$$

$K(\cdot)$ is the chosen kernel function, x_i is support.

vector, $a_i\,(a_i \neq 0)$ is Lagrange multiplier of x_i The selection criteria of optimal vector are as follows:

$$y_i((w\cdot x_i))+b\begin{cases} >1, & \alpha_i= 0 \\ <1, & \alpha_i= C \\ = 1, & 0<\alpha_i<C \end{cases} \qquad (3\text{-}5)$$

3.3.2. The Working Process of the C – SVM Classifier

After introducing kernel function, slack variable and penalty coefficient to SVM classifier, the working process of C – SVM classifier is shown in Figure 3.

3.3.3. Solving a_i and b by using the SMO Algorithm

We use the SMO (Sequential Minimal Optimization, SMO) algorithm to implement the training process. As shown in Figure 4.

3.4. Feature Selecting by PCA

PCA (Principal Component Analysis), a kind of effective means of data dimension reduction, is a method which is eliminating the linear correlation properties and keeping the non-liner correlation properties in set of properties by using orthogonal transformation.

We set part of the KDD99 data set as training set, extract 22 major properties from 41 properties by PCA.

The main steps are as follows:

Step 1: calculate sample average $\overline{x_j}$ and variance $\mathrm{var}(x_j)$.

$$\overline{x}_j=\frac{1}{M}\sum_{i=1}^{M}x_{ij} \quad , \quad i=1,2,\cdots,n\,;\ j=1,2,\cdots,p \tag{3-6}$$

$$var\left(x_j\right)=\frac{1}{n\text{-}1}\sum_{i=1}^{n}\left(x_{ij}-\overline{x}_j\right)^2,\quad i=1,2,\cdots,n\,;\ j=1,2,\cdots,p \tag{3-7}$$

Step 2: normalization of original data.

$$x_{ij}^{*}=\frac{x_{ij}-\overline{x}_j}{\sqrt{var\left(x_j\right)}},\quad i=1,2,\cdots,n\,;\ j=1,2,\cdots,p \tag{3-8}$$

Step 3: calculate sample covariance matrix R.

$$r_{ij}=\frac{1}{n\text{-}1}\sum_{t=1}^{n}x_{ti}x_{tj},\ \ i、j=1,2,\cdots,p \tag{3-9}$$

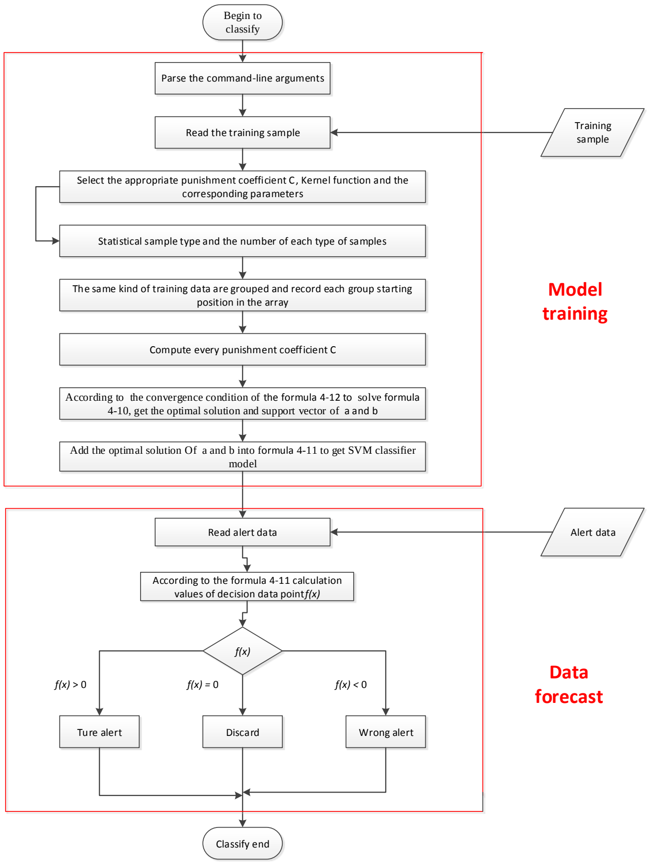

Begin to classify
Parse the command-line arguments
Read the training sample
Training sample
Select the appropriate punishment coefficient C, Kernel function and the corresponding parameters
Statistical sample type and the number of each type of samples
The same kind of training data are grouped and record each group starting position in the array
Model training
Compute every punishment coefficient C
According to the convergence condition of the formula 4-12 to solve formula 4-10, get the optimal solution and support vector of a and b
Add the optimal solution Of a and b into formula 4-11 to get SVM classifier model
Read alert data
Alert data
According to the formula 4-11 calculation values of decision data point f(x)
f(x)
f(x) > 0
f(x) = 0
f(x) < 0
Data forecast
Ture alert
Discard
Wrong alert
Classify end

Figure 3. The Flow Chart of C-SVM Classifier Train

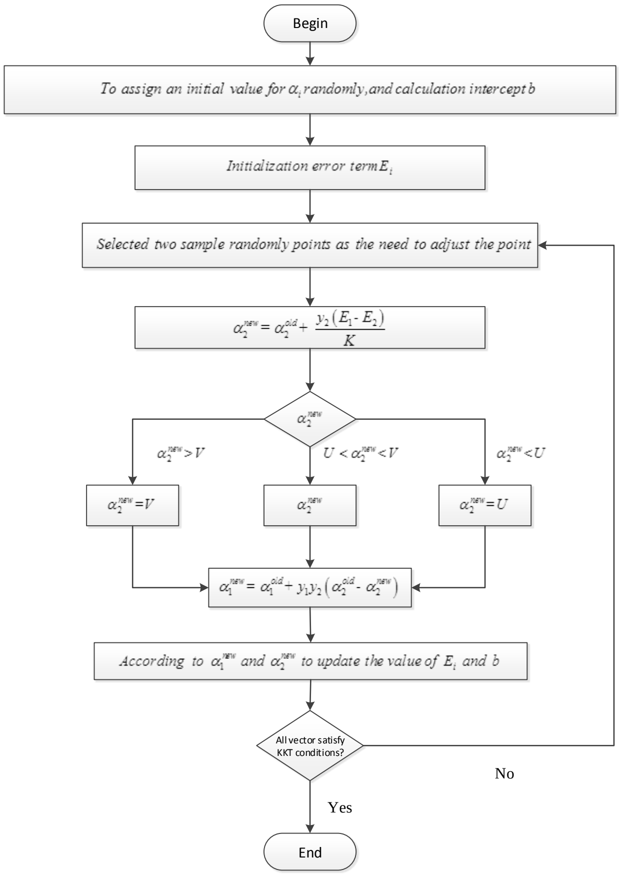

Begin
To assign an initial value for α_i randomly,and calculation intercept b
Initialization error term E_i
Selected two sample randomly points as the need to adjust the point
$\alpha_2^{new} = \alpha_2^{old} + \frac{y_2(E_1 - E_2)}{K}$
α_2^{new}
$\alpha_2^{new} > V$
$U < \alpha_2^{new} < V$
$\alpha_2^{new} < U$
$\alpha_2^{new} = V$
α_2^{new}
$\alpha_2^{new} = U$
$\alpha_1^{new} = \alpha_1^{old} + y_1 y_2 (\alpha_2^{old} - \alpha_2^{new})$
According to α_1^{new} and α_2^{new} to update the value of E_i and b
All vector satisfy KKT conditions?
No
Yes
End

Figure 4. SMO Algorithm

Step 4: calculate eigenvalue $\lambda_i, i = 1,2,\cdots,p$ of sample covariance matrix R and eigenvector $\alpha_i, i = 1,2,\cdots,p$ corresponding to each eigenvalue.

Step 5: sequence eigenvector according to the eigenvalues' descending order.

Step 6: Select principal component According to the cumulative contribution rate of characteristic value

$$\text{Cumulative contribution rate} = \frac{\lambda_i}{\sum_{i=1}^{p} \lambda_i}$$

3.5 Parameter Optimization Method

The penalty coefficient and kernel parameter are the keys that determine the classifier performance. At present the most commonly used parameter optimization method includes genetic algorithm [15], the grid search method [16] and particle swarm optimization algorithm [17]. Considering the complexity of the algorithm, calculation cost, convergence time and cost of implementation, we choose the grid search method in our paper, and use K-fold cross-validation method to Verify the optimization results(As shown in Figure 5).

3.6 False Alarm Filtering Method based on SVM

As shown in Figure 6, the main steps are as follows:

Step 1: Choose data sets, we use the part of KDD99 data set which is after demarcated in this paper.

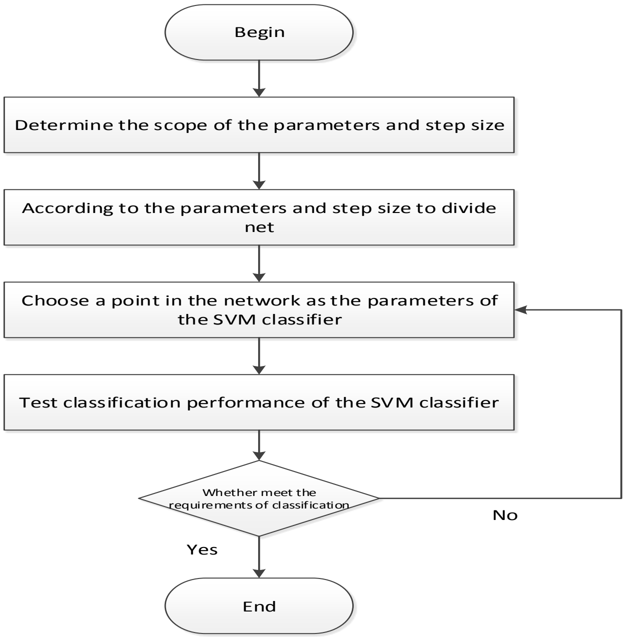

Begin
Determine the scope of the parameters and step size
According to the parameters and step size to divide net
Choose a point in the network as the parameters of the SVM classifier
Test classification performance of the SVM classifier
Whether meet the requirements of classification
No
Yes
End

Figure 5. Grid Search Algorithm

Step 2: Data preprocessing, including: a) quantize the string properties in the KDD99 data set. (Protocol type property TCP,UDP and ICMP were replaced by 1,2,3 respectively; Service type property aol, auth, etc. were replaced by 1,2,3,⋯,70; Normal behavior is 1, Threatening behavior is -1); b) normalize the data quantized.

Step 3: Principal component analysis, get data sets the data set of dimension reduction according to the principal component by PCA.

Step 4: Dataset division, divide data set of dimension reduction to the training set and testing set.

Step 5: Choose kernel function and value of K, construct the C-SVM classifier model.

Step 6: Parameter optimization with grid search algorithm and cross-validation method.

Step 7: Test the C-SVM classifier with the test set, which we get from the training set. If the classification accuracy meets the requirements, we use the classifier to filter the false alarm information, or back to step 3.

Step 8: Extract the source data for alarm filtering from alarm information according to the principal component, get the data set of dimension reduction.

Step 9: Normalize alarm data of dimension reduction.

Step 10: use the C-SVM classifier that we get to classify the normalized alarm data, filter the false alarm and output the true alarm.

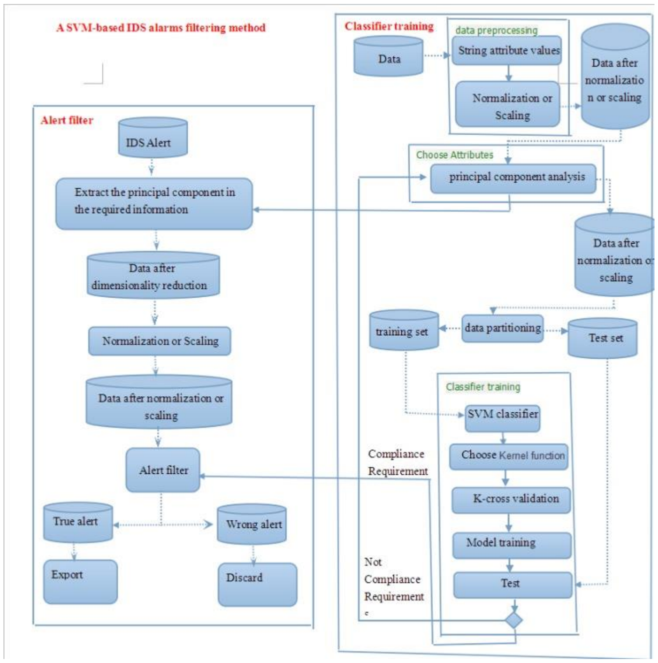


Figure 6. The Architecture of Alarm Filtering based on SVM

4. Model Simulation and the Result Analysis

4.1. Experimental Environment

The experimental environment is shown in the Table 1.

4.2. The Experimental Data Set

Table 1. Experimental Environment

hardware	Thinkpad T420 processor: Intel Core i5-2430 CPU 2.40GHz Installing memory: 6.00G OS: Windows 7 64 - bit flagship SP1
software	IBM SPSS Statistics 19.0 Libsvm-3.1 Gridregression.py

Table 2. The Result of PCA

	Initial Eigenvalues			Extraction of sum of squares loaded		
	summation	variance %	accumulation %	summation	variance %	accumulation %
1	9.215	25.597	25.597	9.215	25.597	25.597
2	4.523	12.564	38.162	4.523	12.564	38.162
3	2.996	8.324	46.485	2.996	8.324	46.485
4	2.634	7.318	53.803	2.634	7.318	53.803
5	1.607	4.465	58.268	1.607	4.465	58.268
6	1.195	3.320	61.588	1.195	3.320	61.588
7	1.145	3.181	64.769	1.145	3.181	64.769

8	1.129	3.137	67.906	1.129	3.137	67.906
9	1.021	2.836	70.742	1.021	2.836	70.742
10	1.002	2.782	73.525	1.002	2.782	73.525
11	.996	2.766	76.291	.996	2.766	76.291
12	.992	2.756	79.047	.992	2.756	79.047
13	.965	2.682	81.729	.965	2.682	81.729
14	.925	2.568	84.297	.925	2.568	84.297
15	.864	2.401	86.698	.864	2.401	86.698
16	.846	2.351	89.049	.846	2.351	89.049
17	.777	2.158	91.207	.777	2.158	91.207
18	.722	2.005	93.212	.722	2.005	93.212
19	.716	1.988	95.200	.716	1.988	95.200
20	.399	1.108	96.308	.399	1.108	96.308
21	.364	1.011	97.319	.364	1.011	97.319
22	.334	.927	98.246	.334	.927	98.246

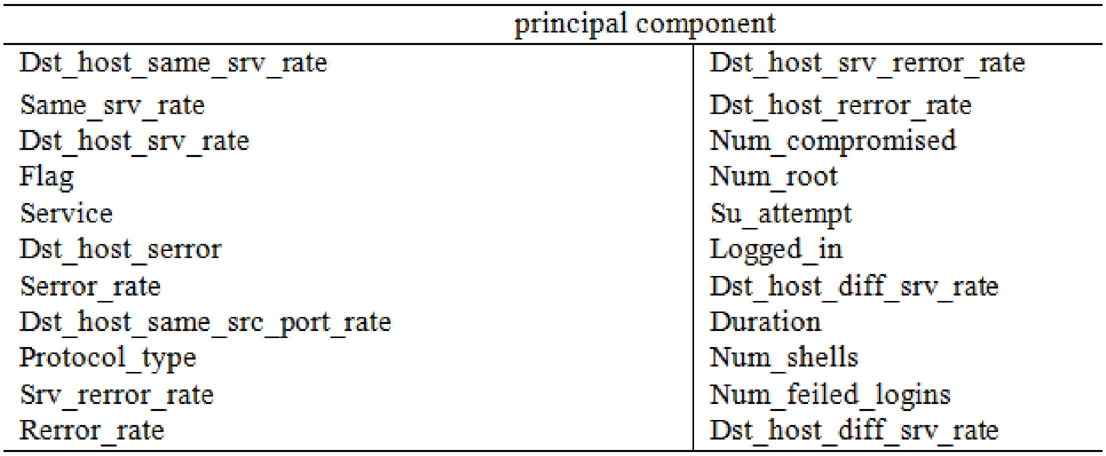
KDD99 data set is a benchmark data set to detect performance of IDS, the data set recorded more than 500 ten thousand Internet connections. There are 493,645 Internet connections in the 10% training subset, including 97,278 normal connections and 39,6743 abnormal connections. The abnormal connections include 22kinds of attacks. There are 311,029 internet connections in the testing subset, including 60,593 normal connections and 250,436 abnormal connections.

In order to verify the alarm filtering method proposed in this paper, Take 20,773 data from the KDD99data set as experiment data, including 16,958 normal connections and 3,814 network attacks. Detect the experiment result by the Snort, there are 12,582 alarms. After calibrate the alarms, get 3,752 true alarms, 8,830 false alarms, fail to declare 62 alarms. False alarm is 70.1796% of the total number of alarm.

4.3. Get Principal Component by using PCA

We pick out 22 principal components from 41 properties of KDD99 data set by using PCA. The results are shown in the Table 2 and Table 3.

Table 3. The Primal Components



principal component	
Dst_host_same_srv_rate	Dst_host_srv_rerror_rate
Same_srv_rate	Dst_host_rerror_rate
Dst_host_srv_rate	Num_compromised
Flag	Num_root
Service	Su_attempt
Dst_host_serror	Logged_in
Serror_rate	Dst_host_diff_srv_rate
Dst_host_same_src_port_rate	Duration
Protocol_type	Num_shells
Srv_rerror_rate	Num_feiled_logins
Rerror_rate	Dst_host_diff_srv_rate

4.4. Gaussian Kernel SVM Classifier Performance

Gaussian kernel function, the expression is: $K(x,z) = exp\left(\frac{\|x-z\|^2}{2\sigma^2}\right), \sigma > 0$ .The main parameters are penalty coefficient C and Kernel function g. The value range of the parameter C is [0, 0.5 × 1010], step pitch is 1. The value range of the coefficient $g = \frac{1}{2\sigma^2}$ is [0, 0.5 × 104], step pitch is 10^{-4}, optimal parameter C_{best} = 2589, g_{best} = 0.1131 are calculated by using the grid search method and cross validation. To test the performance of the classifier, parameter is pick out from the point near C_{best}, g_{best} with the KDD99 data set.

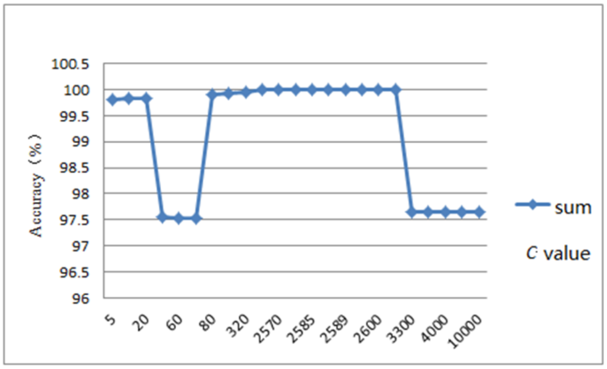


Figure 7. The Line Chart of Classifier's Performance with Different C(g=1/22)

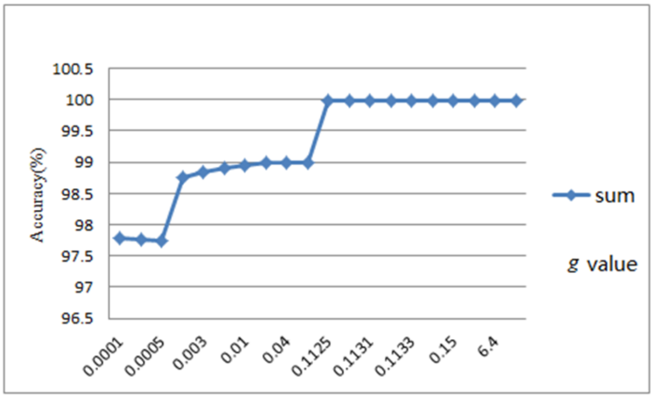


Figure 8. The Architecture of Alarm Filtering based on SVM

As shown in the Figure 7 and Figure 8, the optimal parameter C_{best} = 2589, g_{best} = 0.1131, the iterations number of the SVM classifier #iter = 36786, the number of support vector nSV=177, the number of support vector containing slack variable nBSV=58, Classification accuracy is 99.9954%. Classify 12,582 alarm data with the new classifier, get 3,878 alarm data, including 129 false alarms, The False Alarm Rate is 3.3265%, it is reduced by 95.26%, compared with the original 70.1796%.pitch is 10^{-4}, optimal parameter C_{best} = 2589, g_{best} = 0.1131 are calculated by using the grid search method and cross validation. To test the performance of the classifier, parameter is pick out from the point near C_{best}, g_{best} with the KDD99 data set.

4.5 Polynomial Kernel SVM Classifier Performance

The general expression of polynomial kernel function is:
$K(x,z) = (g(x \cdot z) + r)^d, d \in Z^+, g > 0, r \in constant$

The main parameters are penalty coefficient C and Kernel function g, d, r. The value range of the parameter C is [0, 0.5 × 1010], step pitch is 1, The value range of the parameter g is [0, 0.5 × 104], step pitch is 10^{-4}, The value range of the parameter d is[0, 10], step pitch is 1. The value range of the parameter r is [-0.5 × 104, 0.5 × 104], step pitch is 10^{-2}, optimal parameter: C_{best} = 545, g_{best} = 0.0303, d_{best} = 2, r_{best} = 45.62. To test the performance of the classifier, parameter is pick out from the point near C_{best}, g_{best}, d_{best} and r_{best} with the KDD99 data set.

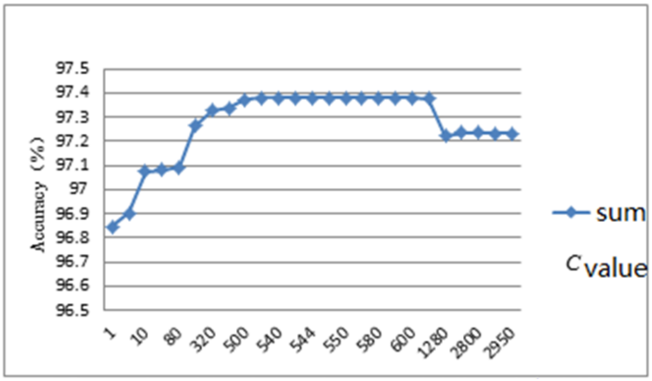


Figure 9. The Line Chart of Classifier Performance with Different C(g=1/22, d=3, r=0)

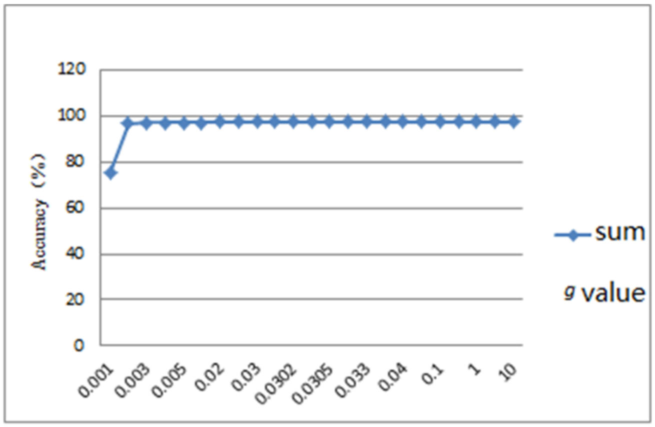


Figure 10. The Line Chart of Classifier Performance with Different g(C=545, d=3, r=0)

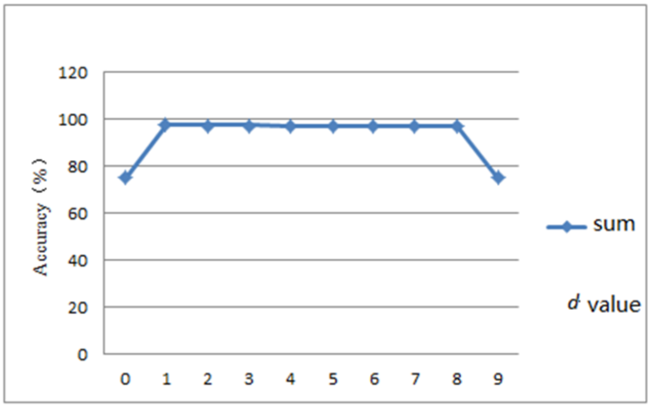


Figure 11. The Line Chart of Classifier Performance with Different d(C=545, g=0.0303, r=0)

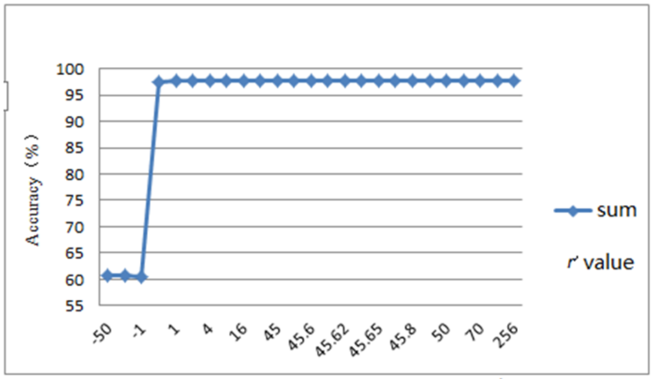


Figure 12. The Line Chart of Classifier Performance with Different r(C=545, g=0.0303, d=2)

As shown in the Figure 9-12, the optimal parameter C_{best} = 545, g_{best} = 0.0303, d_{best} = 2, r_{best} = 45.62, the iterations number of the SVM classifier #iter = 971, the number of support vector nSV=17, the number of support vector containing slack variable nBSV=0, Classification accuracy is 97.6787%. Classify 12,582 alarm data with the new classifier, get 3,962 alarm data, including 298 false alarms, The False Alarm Rate is 7.5214%, it is reduced by 95.26%, compared with the original 89.2826%.

4.6 Sigmoid Kernel SVM Classifier Performance

Sigmoid kernel function, also known as multilayer perceptron kernel function, is a kind of kernel function which based on the ideas of neural network, its expression is:

$$K(x,z) = \tanh(g(x \cdot z) + r), g > 0, r > 0$$

The main parameters are penalty coefficient C and Kernel function g, r. The value range of the parameter C is [0, 0.5 × 1010], step pitch is 1, The value range of the parameter g is [0, 0.5 × 104], step pitch is 10^{-4}, The value range of the parameter r is [0, 0.5 × 102], step pitch is 10^{-4}, optimal parameter : C_{best} = 1160, g_{best} = 0.0478, r_{best} = 0.7847. To test the performance of the classifier, parameter is picked out from the point near C_{best}, g_{best} and r_{best} with the KDD99 data set.

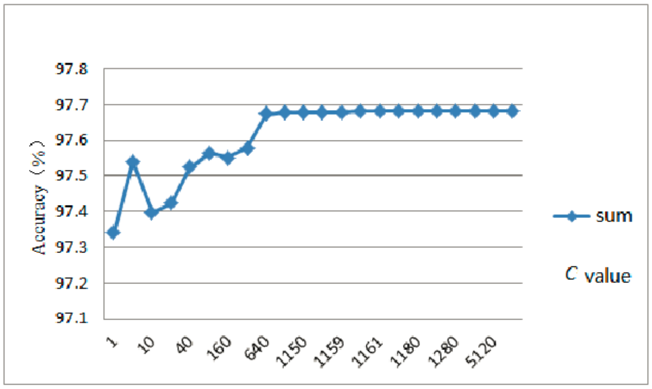


Figure 13. The Line Chart of Classifier Performance with Different C(g=1/22, r=0)

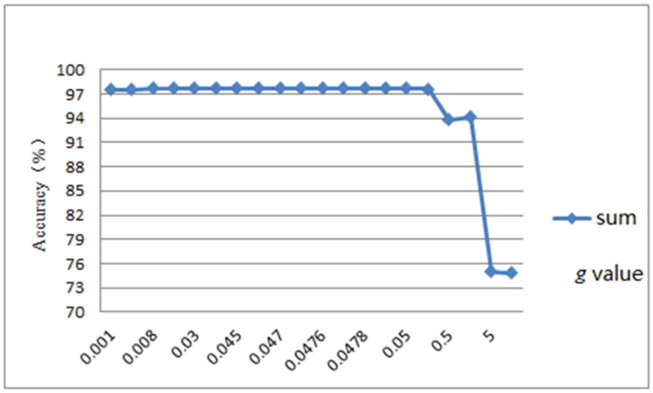


Figure 14. The Line Chart of Classifier Performance with Different g(C=1160, r=0)

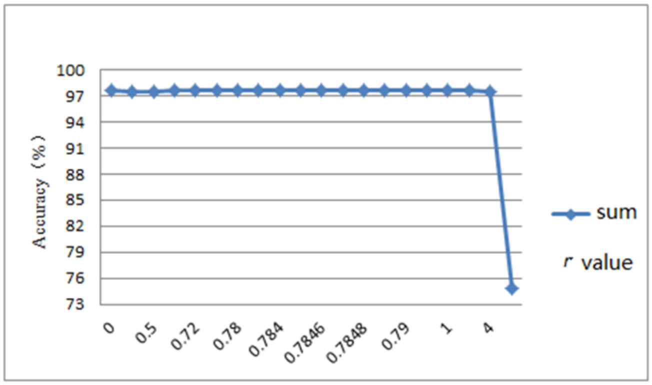


Figure 15. The Line Chart of Classifier Performance with Different g(C=1160, g=0.0478)

As shown in the Figure 13-15, the optimal parameter C_{best} = 1160, g_{best} = 0.0478, r_{best} = 0.7847, the iterations number of the SVM classifier #iter = 854, the number of support vector nSV =34, the number of support vector containing slack variable nBSV =23, Classification accuracy is 97.7141%. Classify 12,582 alarm data with the new classifier, get 3,894 alarm data, including 223 false alarms, the False Alarm Rate is 5.7268%, It is reduced by 95.26%, compared with the original 91.8397%.

4.7 The Experiment Results Analysis

By the experiment result, we can know that:

a) As shown in the Table 13,the False Alarm Rates of Gaussian kernel function SVM classifier, Polynomial kernel function SVM classifier and Sigmoid kernel function SVM classifier are all less than 10%, they are all reduced by about 90%. Therefore, the false alarm filtering method based on SVM we proposed effectively reduces the false alarm rate alarm message.

b) The classification performance of Gaussian kernel function SVM classifier is the best one.

c) Polynomial kernel function SVM classifier and Sigmoid kernel function SVM classifier have a similar classification performance.

d) Kernel function and parameters is the key to classifier performance. For example, Classification accuracy of Gaussian kernel function SVM classifier is basically around 99.9%. Classification performance of Gaussian kernel function is generally better than the other ones'; the classification performance of Sigmoid kernel function has a big changes: it is best when C = 1160, g = 0.0478, r = 0.7847,it is worst when C = 1160, g = 10,r = 0.

5. Future work

The false alarm filtering method based on SVM we proposed effectively reduces the false alarm rate alarm message, but there is still something need to be improved: The SVM classifier is trained by the training of the training data set, while the internet connections is quite different in different kind of local area network, there is no existing general data set for all kinds of net. In the future we should strengthen the work on extracting data set in in different kind of local area network. SVM classifier need consume large amounts of RAM when calculating the Hessian matrix. How to calculate the Hessian matrix with less memory consumption is still an important subject in the study of the SVM.

Acknowledgements

This work was partially supported by the National Natural Science Foundation of China under Grant 61172072, 61271308, the Beijing Natural Science Foundation under Grant 4112045, the Beijing Science and Technology Program under Grant Z121100000312024, the Academic Discipline and Postgraduate Education Project of Beijing Municipal Commission of Education.

References

[1] L. Brown and W. Stalling, "Computer Security: Principles and Practice", William Stallings, **(2008)**.

[2] "360 Internet Security Center", Gartner. Enterprise security trends in the Internet age, **(2013)**.

[3] V. Aghaei-Foroushani, "On Evaluating IP Traceback Schemes: A Practical Perspective", Security and Privacy Workshops (SPW) IEEE, San Francisco, CA, **(2013)** May 23-24 pp. 127- 134.

[4] J.-S. Sung, S.-M. Kang and Y. Lee, "A Multi-gigabit Rate Deep Packet Inspection Algorithm using TCAM", Proceedings of the 2005 IEEE Global Telecommunications Conference (GLOBECOM '05), IEEE, **(2005)** December 2, pp. 453-457. St. Louis, USA, Piscataway, NJ, USA

[5] S. Tian, Z. Wei, W. Dongsheng, *et al.*, "A memory efficient multiple pattern matching architecture for network security", Proceedings of the 27th IEEE Conference on Computer Communications (INFOCOM'08), , IEEE Computer Society, **(2008)** April 15-17, pp. 166-170, Phoenix, USA, Washington, DC, USA.

[6] C. Zhiping, L. Shuhao, W. Han, *et al.*, "High Performance Parallel Intrusion Detection Algorithms and Framework", Journal of Frontiers of Computer Science and Technology, vol. 4, no. 7, **(2013),** pp. 289-303.

[7] Q. Ziyan and Z. Zengyi, "Analysis of False information in Net intrusion Detection System", 2007 Beijing University graduate student academic exchange communication and information technology conference proceedings, **(2008),** pp. 623-628.

[8] Y.-D. Lin, Y.-C. Lai, C.-Y. Ho, *et al.*, "Creditability-based weighted voting for reducing false positives and negatives in intrusion detection. Computers & Security, vol. 39, Part B, **(2013)** November, pp. 460–474.

[9] D. Gupta, P. S. Joshi, A. K. Bhattacharjee and R. S. Mundada, "IDS alerts classification using knowledge-based evaluation", International conference on communication systems and networks, **(2012)** January, pp. 1-8.

[10] V. A. Foroushani and A. N. Zincir-Heywood, "Deterministic and Authenticated Flow Marking for IP Traceback", 2013 IEEE 27th International Conference on Advanced Information Networking and Applications, **(2013)** March 25-28, pp. 397 – 404, Barcelona.

[11] A. Izaddoost, M. Othman, M. Fadlee and A. Rasid, "Accurate ICMP TraceBack Model under DoS/DDoS Attack", 15th International Conference on Advanced Computing and Communications, **(2007)** December 18-21, pp. 441-446, Guwahati, Assam.

[12] C. Wang, N. Du and H. Yang, "Generation and Analysis of Attack Graphs", International Workshop on Information and Electronics Engineering (IWIEE), Procedia Engineering, vol. 1, no. 29, **(2012),** pp. 4053–4057.

[13] A. Thomas, "RAPID: Reputation based Approach for Improving Intrusion Detection Effectiveness", Sixth International Conference on Information, **(2010).**

[14] J. C. Platt, "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines", Technical Report MSR-TR-98-14, **(1998)** April 21.

[15] S. Hussain, A. Olayemi and S.-S. Yeo, "Genetic algorithms for effective open port selection for a web filter", Personal and Ubiquitous Computing, vol. 8, no. 17, **(2013)** December, pp. 1693-1698.

[16] A. B. Jimenez, J. L. Lazaro and J. R. Dorronsoro, "Finding optimal model parameters by deterministic and annealed focused grid search", Neurocomputing, vol. 72, **(2009)** August13-15, pp. 2824–2832.

[17] M.-S. Leu, M.-F. Yeh and S.-C. Wang, "Particle swarm optimization with grey evolutionary analysis", Applied Soft Computing, vol. 10, no. 13, **(2013)** October, pp. 4047–4062.

Authors

Yun Liu, is a Professor in School of Electronic and Information Engineering in Beijing Jiaotong University where she received her PhD degree in the Communication and Information System. She is interested in Opinion Dyanamics, Network/Information Security, Computer Communication, and the Intelligent Transportation System.

Kun-Peng Xia, received his B.E. degree in Communication Engineering at the PLA Information Engineering University in 2011. Currently, he is a graduate student in the Department of Electronic and Information Engineering, majoring in Communication and Information Systems.

Jian-Xun Zhao, is a freshman in the Department of Electronic and Information Engineering at Beijing Jiaotong University.