An Host Anomaly Detection Algorithm Based on Bayesian Tree

Yaning Zheng [2, b], Wujun Yao[1, a]

[1] Xi'an Modern Chemistry Research Institue, Xi'an, 710086, China

[2] Department of electronic technology Engineering University of Chinese Armed Police Force, Xi'an, 710086, China

[a]email: 78013810@qq.com, [b]email: sensornet@163.com

Keywords: Intrusion detection; Bayesian tree; Local System Service

Abstract. The naive Bayes algorithm in intrusion detection have the problem of high internal dependence and the data "broken" in decision tree, in order to solve the problem, this paper combines the advantages of section in decision tree and multi-evidence fusion in naive Bayes, uses the Windows Native APIs related data as data sources, using the Native APIs sequence produced by key process, construct the process service predicting model based on the Bayesian tree algorithm, and uses U-test method as the anomaly detection algorithm. The experimental results show that the model can effectively detect abnormal host, and the time complexity is lower, is suitable for online detection.

Introduction

With the development of the host anomaly detection study, more and more modeling method are proposed, the fixed-length sequence research is just one of the commonly used methods. Forrest [1] using a set of fixed short sequence model to represent the process of normal execution model, Lee[2] establish the model with the mothod of data mining, Helman [3] establish statistical model by calculating the frequentness of sequence in the abnormal or normal data. The thought of these methods are trying to predict whether a child sequence is produced by normal process, or its probability that produced by normal process. In traditional search mode method, the sequence prediction is only two values, namely, "0" or "1". Because of the internal dependence of Native APIs sequences, this paper tries to construct a model to depict the sequence prediction probability distribution.

For a long time, establishing probability prediction model is an important research direction. Eskin[4] constructed the prediction model with sparse markov tree, which predicted the emergence probability of the first N system call through the last N-1 system calls, the low probability is considered abnormal. This method took only a probability value, there is no use of prediction probability distribution. Tatara[5] thought sequence have some non-consecutive features, so the sequence property is regarded as random variables, using bayesian network method to predict the probability distribution of the Nth call, this method is opposite to the traditional idea which thought dependence exists between the sequence properties, although have a certain originality, the results are not very ideal.

Bayesian network is a good probability prediction method, but need to assume that all properties as random variables. Naive Bayesian classifier needs to assume that all properties are independent of each other. Bayesian tree algorithm is proposed in this paper, which properly use the dependence between the sequence, and alleviate the naive bayes algorithm for property independence requirements, through the last N-1 sequence calls, predict probability distribution of the Nth call to establish model, using the hypothesis test method for anomaly detection algorithm. Experimental results indicate that this model could effectively distinguish between normal and abnormal data.

Data source as the main basis of abnormal diagnosis in intrusion detection system, its quality is essential to the effectiveness of the detection system, choose a proper data source is the precondition of establishing intrusion detection model. Good data source should satisfy easy

collection, can distinguish, moderate amount of data and real-time requirements.

Windows Native APIs related data is a kind of host intrusion detection data sources which meet the requirements, it is also the interface to process access system resources, and a fine-grained description of process behavior. This paper uses the Windows Native APIs related data as data sources, constructing the key process behavior model with service sequence number.

Process Behavior Model Based on Bayesian Tree

A. The Bayesian Tree Algorithm

Naive Bayes and decision tree is two common classification algorithm, which are used in intrusion detection. For example, [5] established the forecast model of the system calls using Bayesian network, the decision tree algorithm is used to looking for continuous mode in ref. [6].

Under the condition of a given category, Naive Bayes algorithm have good performance when the properties are independent of each other, because the classifier classification decision after considering multiple property and evidence. However, naive Bayesian classifier needs strong attribute independence assumption, if this condition is not met, will reduce the classification accuracy. As a result of the decision tree algorithm to produce the leaves of a tree contains relatively few samples, so we assume that the sample data roughly satisfy the conditional independence assumption, we replace the decision tree leaf nodes with Naive Bayesian classifier, such improved to some extent solve the influence of the data internal dependencies. Decision tree is another kind of fast algorithm, used to find discontinuous mode in [6], but the method is derived based on the recursive partitioning, that can produce a "broken" problem: data sets break up according to the test data, when the segmentation greater than 20 layers, often appear only a small amount of data can be used to make decisions, and led to the decrease of the prediction performance. To avoid divided into very small data set, under the condition of relative error lower than 5% and the nodes more than 20 examples, Bayesian Tree algorithm segmentation is deemed to be meaningful, such definition to some extent relieves the problem of "broken".

If we use the Naive Bayesian classifiers instead of the decision tree leaves, can combine the advantages of decision tree section and Naive Bayesian multi-property evidence accumulation. Two algorithms, the former stress on probability calculation, the latter is based on information theory, compromise preferably. Kohavi[7] presented Bayesian tree algorithm in 1996, and proved that the classification performance of Bayesian tree algorithm is better than that of Bayesian algorithm and decision tree algorithm.

There is a certain correlation between Native APIs sequence. This correlation can be used to design fixed-length sequence as input of Bayesian tree algorithm, and the output is the probability distribution of next possible call of this sequence. In other words, we can predict the next probability distribution of a process through its observation sequence. When we evaluate a new process to determine its operation situation, with a to our algorithm calculate the prediction probability of each child sequence that produced by sliding window cutting algorithm of its father process. If the actual probability distribution of child sequence is obviously different with the maximum probability distribution sequence, we think that the subsequence is not produced from normal process, namely it is abnormal.

B. Process Behavior Model

Data preprocessing: Dividing the Native APIs sequence of collected key process into several independent parts according to the different threads, and cutting it to fixed-length sequences with sliding window.

Definition 1: Collection $\sum = \{x_1, x_2, \cdots, x_l\}$, size is l, which each element is a Native API number. The collection $C = \{c_1, c_2, \cdots c_{|c|}\}$, size of $|c|$, is the set of all possible values of $\sum$. $s = (x_i, x_{i+1}, \cdots, x_{i+N-1}), (x_i \in \sum, 1 \le i \le l - N + 1)$ is a Native API sequence, data set D which is composed of m training sequences is the of input Bayesian tree algorithm. For s, the last $N-1$ properties is defined as non-class property X ($X \in R^{N-1}$, R is a nonnegative integer), the N[th] property is defined as

a class attribute.

The first step of Bayesian tree is to establish a tree $T=\{t_1,t_2,\cdots t_k\}$, which contains k nodes, $t_i(1\le i\le k)$ is a node of the tree, D_{t_i} is the data set in t_i node. All properties are discrete, so gain ratio method is used to select the property $X_p(1\le p\le N-1, X_p\in X)$, where each select the property with maximum gain ratio (R) as the current node each time, all possible values of the attribute as the branch of the node and recursive spanning tree. If there is no property whose gain ratio is obviously superior to others, we use a naive Bayesian classifier was as a leaf node. The gain ratio of property X_p of the node t_i is:

$$R(X_p)=G(X_p)/S(X_p) \tag{1}$$

$$G(X_p)=I(D_{t_i})-In_{X_p}(D_{t_i}) \tag{2}$$

$$I(D_{t_i})=-\sum_{j=1}^{N}(n_j(t_i)/n(t_i))\cdot\log_2(n_j(t_i)/n(t_i)) \tag{3}$$

$$In_{X_p}(D_{t_i})=-\sum_{j=1}^{N}(|D_{t_i}^j|/n(t_i)\cdot I(D_{t_i}^j)) \tag{4}$$

$$S(X_p)=-\sum_{j=1}^{N}((|D_{t_i}^j|/n(t_i))\cdot\log_2(|D_{t_i}^j|/n(t_i))) \tag{5}$$

$D_{t_i}^{\ j}$ represents the data set that X_p take the j^{th} possible value, $n(t_i)$ represents the data number of data set t_i, $n_j(t_i)$ represents the number of nodes in t_i that belongs to X_p takes the j^{th} value , $e(t_i)$ represents the data number that not belongs to multi-class in t_i.

D_{t_i} which in leaf node t_i is the learning sample of the Bayesian classifier, $n(Y_j)$ represents the number of samples where $Y=c_j(1\le j\le|c|)$ in D_{t_i}.

Prior probability and posterior probability are shown follow.

$$P(c_j)=\frac{n(Y_j)}{n(t_i)},\quad p(c_j/X)=\frac{P(X/c_j)P(c_j)}{P(X)}\quad(1\le j\le|c|) \tag{6}$$

The probability distribution $P(Y|X)$ of class property under t_i is resulting from formula (6).

Taking *N-1* elements repeatedly from collection *C* and sort them, then generate conditional collection $\Phi=\{(x_i,x_{i+1},\cdots x_{i+N-1})\,|\,x_i\in C,1\le i\le p(|c|,N-1)\}$, arrangement number is $p(|c|,N-1)=|c|^{N-1}$. Conditional probability distribution set of class property is defined as $\Omega=\{P(Y/X)\,|\,Y\in C,X\in\Phi\}$, From the collection have repeatedly take N - 1 orderly do get conditions collection, arrangement number. Class attribute set is defined as the probability distribution, the elements in Φ and Ω are one to one correspondence. Ω is just the prediction model generated by Bayesian tree algorithm.

C. Time Complexity Analysis of Bayesian Tree Algorithm

We assume that after be pre-processed, the training data include m instances (m sequences), N attributes, the depth of tree with m leaves is $O(\log(m))$, therefore, the computational complexity of decision creation tree is $O(mN\log(m))$. For Bayesian tree, given m instances, N attributes and $|c|$ calibration values, then the complexity of attribute selection is $O(mN^2|c|)$. Generally, the number of attributes is N means the length of sliding window is less than $O(\log(m))$, and the variety of $|c|$ is few and set, therefore, we can predict that time complexity analysis of Bayesian Tree algorithm increases linearly with the size of instance number m.

Anomaly Detection Algorithm

Sorting elements in the collection C ascend according to the frequency $f(c)$, makes $f(c_1)<f(c_2)\cdots<f(c_{|c|})$. The instance attribute values in Φ is ascending order. For sequence to be detected, s, the former $N-1$ sequence members is the conditional sequence, the serial number of the sequence members in C is marked as $O(p_1,p_2,\cdots p_{N-1})$, so position of s in Φ is

$p_1 \cdot |c|^{N-2} + p_2 \cdot |c|^{N-3} + \cdots + p_{N-1}$, therefore, we can quickly get the probability distribution of s in Ω' from database that saved by the model according to its position.

To the sequence s to be detected that size of l, take out $P(Y / x_i, x_{i+1}, \cdots, x_{i+N-2})$ from Ω', and assume that x_n is actual value of Y, x_j is the value of Y to make B_i maximum,

$A_i = P(x_n / x_i, x_{i+1}, \cdots, x_{i+N-2})$, $B_i = \max P(x_j / x_i, x_{i+1}, \cdots, x_{i+N-2})$ $(1 \le i \le l)$.

For the sequence s that size of l, we can get $(A_1, A_2, \cdots A_l), (B_1, B_2, \cdots B_l)$, then cut them with sliding window that size of I, bring $\tilde{A}_i = (A_i, A_{i+1}, \cdots A_{i-I+1}), \tilde{B}_i = (B_i, B_{i+1}, \cdots B_{i-I+1})$ $(1 \le i \le l + I - 1)$, do U test on them.

Assume that $\tilde{A}_i$ and $\tilde{B}_i$ respectively represent the sample of first and second order matrix, μ_1, μ_2 is the mother mean, the capacity of the sub-sample, mean and variance respectively represent as $n_A, \overline{A}_i, S_A^2$ and $n_B, \overline{B}_i, S_B^2$. On the mother samples, we assume that $H_0 : \mu_1 = \mu_2$.

If n_A, n_B are large, $U = \dfrac{|\overline{A}_i - \overline{B}_i|}{\sqrt{\dfrac{S_A^2}{n_A} + \dfrac{S_B^2}{n_B}}}$ approximately obey standard normal distribution $N(0,1)$.

For a given significant level:

If, $|\overline{A}_i - \overline{B}_i| \ge u_{\frac{\alpha}{2}} \sqrt{\dfrac{S_A^2}{n_A} + \dfrac{S_B^2}{n_B}}$, then refuse H_0, namely view the two mother samples have significant differences, so let $H_i = 1$;

Else if $|\overline{A}_i - \overline{B}_i| < u_{\frac{\alpha}{2}} \sqrt{\dfrac{S_A^2}{n_A} + \dfrac{S_B^2}{n_B}}$, then accept H_0, namely view the two mother samples do not have significant differences, so let $H_i = 0$.

Sequence anomaly detection decide by the sum of each 15 consecutive H_i $(1 \le i \le l)$ values, that is using LFC method. The greater sum (closer the 15) means greater abnormal degree.

Experimental Environment and Data

Hardware environment: Core2 CPU, 2G DDR memory; Software environment: Windows XP, matlab7.1.We select data of several key processes and build model, the data as shown in table 3-1.

Table1. Experiment data

Process	Normal	
	Thread	Call Number
inetinfo	128	721993
Serv-u	111	385415
lsass	119	256787

For general fixed-length mode, $N = 6$ is regarded as be the best balance of detection capability and efficiency. But the space complexity of our model is $o(|c|^{N-1})$, small N is better for reducing scale of model. When $N < 4$, detection ability is insufficient, but when $N = 4$, the experiment shows that our method still has good capability of anomaly detection. In order to balance the detection ability and the model scale, in this paper, we take the sliding window with $N = 4$ and cut data, choose the fourth columns as the class attribute. As shown in figure 1, in which * represents class attribute.

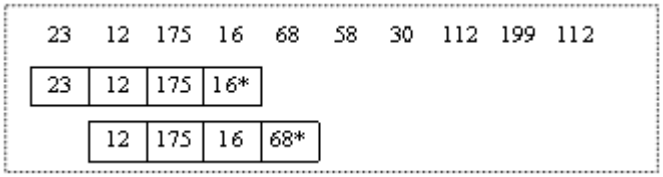

Fig.1. Data Preprocessing

The probability predicted model (part) as shown in table 2, the former 3 columns is the first three attribute values of the sequences, and the other columns are the corresponding probability for the fourth property take different values.

Table2. Prediction Model

Former 3 calls			Probability distribution of the fourth call								
			0	1	10	12	16	23	24	26	…
24	68	56	0.01	0.01	0.01	0.01	0.01	0.01	0.414	0.01	…
68	56	168	0.011	0.011	0.011	0.011	0.011	0.011	0.011	0.011	…
56	168	136	0.013	0.013	0.013	0.013	0.013	0.013	0.013	0.013	…
199	32	130	0.006	0.006	0.006	0.006	0.006	0.006	0.006	0.006	…
199	32	130	0.006	0.006	0.006	0.006	0.006	0.006	0.006	0.006	…

A. Normal Data Prediction

Divide the normal data into 10 subsets, use 9 subsets as the training set each time, and use one as test set. For the data in test set, predict the fourth property with maximum probability by the former three properties, and compared with the fourth call number in sequence, if it is same then consider classification is correct, else think classification is error. Classification accuracy is ratio of right classification in the total test serials. The results as shown in table 3, it can be seen that the model has good prediction ability for normal data.

Table3. The predict the results for normal data

Process	Train call number	Test call number	accuracy
Inetinfo	644003	77990	91.1%
Serv-u	345802	39613	90.3%
Lsass	220100	36687	90.4%

B. Anomaly Detection Results

Abnormal data in figure 2 and figure 3 is several possible loopholes scanned by software, includ Unicode vulnerability, FrontPage extensions, trying to get SAM file, attempts to acquire PcAnyWhere password files, and CGI vulnerability scanning data. As shown in the figure below, the detection algorithm can properly distinguish between the normal and abnormal process.

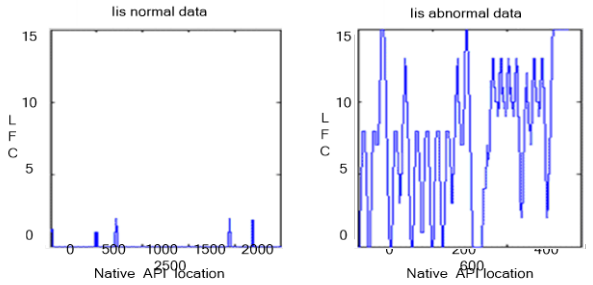

Fig.2. Iis detection figure

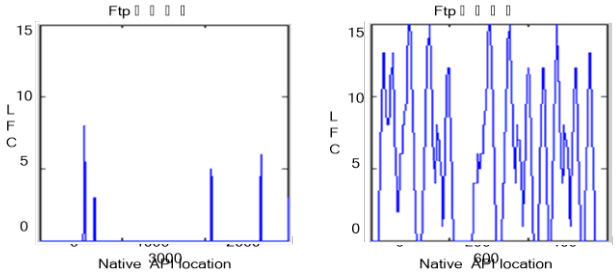

Fig.3. Ftp detection figure

The false alarm rate and missing alarm rate are two important indications for evaluation of intrusion detection system. In the course of calibration system sequence, the first type of error is false alarm, namely intrusion behavior has not been detected; The second type of error is missing alarm, namely normal sequence is estimated abnormal. All been studied so far, abnormal tend to appear in local centralized system call sequences [8], so the LFC is set a threshold value in this paper. When LFC value is lower than specified threshold value, despite abnormal, also believes that the sequence is normal; When the LFC value exceeds the threshold, then that the sequence of anomalies. Obviously, lower threshold will help decrease the missing alarm rate, but at the same time, the false alarm rate will increase. On the other hand, when the threshold rises, the false alarm rate will reduce but missing alarm rate will rise.

Table 4 and table 5 shows when the window length is 4, false alarm rate and missing alarm rate comparison of our algorithm and Forrest. For convenience, we convert the threshold range into 0-15.

Table4. Comparison of false alarm rate

process	Forrest		Our algorithm	
	false alarm rate	threshold	false alarm rate	threshold
Inetinfo	0%	4-15	0%	7-15
Serv-u	1%	5-15	1%	7-15
Lsass	1%	5-15	1%	8-15

Table5. Comparison of missing alarm rate

process	Forrest		Our algorithm	
	false alarm rate	threshold	false alarm rate	threshold
Inetinfo	1%	1-5	0%	5-9
Serv-u	1%	1-6	1%	4-8
Lsass	1%	1-6	0.5%	4-9

On premise of lower rate of false alarm and missing alarm, we can the compare the threshold range intersection of two tables, it can be seen that our algorithm has larger threshold choice space.

The time complexity of Forrest is $O\left(|s|+|s|R_A|\Phi|\right)$, R_A is the rate of abnormal sequence, $|\Phi|$ is the model number. Our algorithm can compute the location of probability distribution in the probability distribution table and then obtain the probability data directly, so time complexity is $o(|s|)$. It can be seen that our algorithm have lower time complexity, so it has some advantages in case of much abnormal, and it is suitable for online detection.

Conclusion

In this paper, using the key process of Native APIs sequence, we build process service prediction model based on Bayesian tree algorithm, and using U testing method for anomaly detection, the experimental results show that the model is well for anomaly detection. It is suitable for online detection because of lower time complexity. The space complexity of our detection model space is , if and N is large, for some key process model may require large storage space, therefore, we further research will introduce database and optimization technology, solving the problem of storage model.

Acknowledgement

Our research was sponsored by the Nature Science Foundation (Project No. 61272492)

References

[1] Dorothy E. Denning. An Intrusion-Detection Model. IEEE Transactions on Software Engineering[J]. 1987,13(2).

[2] S. Forrest, S. A. Hofmeyr, et al, A sense of self for Unix processes. IEEE Symposium on Computer Security and Privacy[C]. Los Alamos, CA, 1996.pp.120-128.

[3] P. Helman and J. Bhangoo. A statistically base system for prioritizing information exploration under uncertainty. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans[C].1997:449–466.

[4] E. Eskin, W. Lee, and S. Stolfo, Modeling system call for intrusion detection using dynamic window sizes. In Proceedings of the 2001 DARPA Information Survivability Conference & Exposition[C]. Anaheim:CA, June 2001.

[5] Kohei TATARA,et al. A Probabilistic Method for Detecting Anomalous Program Behavior, Workshop on Information Security Applications(WISA04)[C]. Aug, 2004.

[6] Li Naijie. Research on Intrusion detection Based on Windows Host Behavior[D]. Xi'an jiao tong Universtiy,2006.

[7] Kohavi R. Scaling up the accuracy of Naive-Bayes classifiers: A decision-tree hybrid. In: Simoudis E, Han J, Fayyad UM, eds. Proc. of the 2nd Int'l Conf. on Knowledge Discovery and Data Mining[C]. Menlo Park: AAAI Press, 1996. 202-207.

[8] J. P. Anderson, Computer Security Threat Monitoring and Surveillance[R].James P Anderson Co. Fort Washington, PA.Apr. 1980.