The intrusion detection algorithm using statistic technology

Ming Gu
Dept. of Software, ShenZhen Polytechnic, ShenZhen, China 518055
Email: gum_guming@hotmail.com

The intrusion detection algorithm calculates the deviation distance using Chi-square test statistic. The amount of calculation can be greatly reduced. Sample data coming from operating system LINUX and Windows8 was presented and compared. The results of this study show that the algorithm achieves the 0% false alarm rate and the 100% detection rate for abnormal intrusion plots. All intrusion plots are detected at the first or second audit event.

Keywords: Computer security; Algorithm; Intrusion detection; Statistic technique; Operating system.

Introduction

Signature recognition and anomaly detection [1, 2, 3] are two main types in currently existing intrusion detection techniques. The shortcoming of former is to identify of the new invasion. The latter is to build a Profile of information system under normal circumstances. By comparing the difference called off which is observed system behavior and the Profile, we can determine there is invasion when off to meet certain restrictions. Therefore, anomaly detection can not only detect known intrusion, can also detect unknown intrusions.

In these existing abnormal detection technologies, normal profile can be described in case of formal logic [1], statistical distribution [2,3], string [4] and so on. Although these technologies can, to some extent, detect intrusion, but there are the following limitations. First, test string does not be generated for some non-normal string. Second, the robustness to noise is not enough. The noise in the normal string may lead to match of the normal string and test string. False detection is produced. The noise in the abnormal string may lead to false match of the normal string and test string. Neglect detection is produced. Third, it is difficult to enumerate all the normal behavior of the information system, especially when the object of the information system is more complex. Fourth, because the objects behavior in the information system is dynamic, it's difficult to describe the changing object behavior with formal logic or production rules in advance.

In order to overcome these shortcomings, abnormal detection [2] based on statistics is used to represent the desired objects and changing normal working behavior. This research is divided into single-variable and multi-variable. Because invasion includes multiple objects and behaviors, and affects the measurement of multiple behavior0s, process control technology of multi-variable is relatively effective and practical for intrusion detection.

Commonly used multi-variable process control technology includes Hotelling's T2 and so on [5,6]. In theory, these techniques can be used to monitor and detect abnormal information systems. In the actual application, because the amount of data calculated is limited, these technologies have two difficulties. First, the intrusion detection involves a large number of multi-dimensional process data. The practical application of these techniques is limited. Second, in order to ensure early prediction and early warning, the processing of each event requires minimal time delay in intrusion detection. These techniques have limitations in this respect. The limitation is reflected in computing capacity. For example, Hotelling's T2 needs to calculate a covariance matrix and its inverse matrix. If the frequency of event occurring is high, calculate of covariance matrix and its inverse matrix will take the computer a long time. It will lead to warning delay of abnormal detection and normal warnings do not be achieved.

Therefore, the low computational multivariate anomaly detection technique is very necessary and practical. Statistic of Hotelling's T2 is measure statistic distance from the observation point to average evaluation of multivariate normal distribution.

We adopted the Chi-square test statistic [7] to calculate the deviation distance. The amount of calculation can be greatly reduced. The purpose of proper warning and early warning for anomaly detection can be achieved.

The principle of the Chi-square test technique

The formula of calculating deviation distance in Chi-square test statistics is as following:

$$\mathbf{x}^2 = \sum_{i=1}^{n} \frac{(\mathbf{x}_i - \mathbf{E}_i)^2}{\mathbf{E}_i} \tag{1}$$

$\mathbf{x}_i$ is the observation value of variable i.

$\mathbf{E}_i$ is the expected value of variable i and is a statistical constant. N is the number of variables. The closer the observation value is to expected value, the less the value of $\mathbf{x}^2$ is.

In our testing technology, the sample average value is used to express the estimate of expected value:

$$\overline{\mathbf{x}_i} = (\sum_{i=1}^{n} \mathbf{x}_i) / n \tag{2}$$

When the sample average value replaced the expected value, the equation (1) becomes the following formula:

$$\mathbf{x}^2 = \sum_{i=1}^{n} \frac{(\mathbf{x}_i - \overline{\mathbf{x}_i})^2}{\overline{\mathbf{x}_i}} \tag{3}$$

According to the central limit theorem of statistic, when the number of variables is large enough (e.g. greater than 30), $\mathbf{x}^2$ used as the sums of squares between observed and expected value is close to the normal distribution. According to the characteristics of the normal distribution, the interval [μ-Zα/2σ, μ + Zα/2σ] includes (1-α)% of the total $\mathbf{x}^2$ value, where μ and σ are mean and deviation from the overall value of $\mathbf{x}^2$, α is the confidence interval, Zα / 2 is a value of standard normal distributing list.

By sample average value $\mathbf{x}^2$ and standard deviation S_x^2, the total field $\mathbf{x}^2$ can be estimated from the sampled data. According to 3-Sigma control limits [5], abnormal detection limit controlled should be set to $[\overline{\mathbf{x}^2} - 3S_x^2, \overline{\mathbf{x}^2} + 3S_x^2]$. For abnormal detection of information systems, we are concerned about the relatively large value of $\mathbf{x}^2$. Deviation of observed and expected value is large when the value of $\mathbf{x}^2$ is large. At this time, an exception may occur. Control of the upper limit of abnormal detection is set to $\overline{\mathbf{x}^2} + 3S_x^2$.

Therefore, for monitoring observation of an exception, when calculating the value of $\mathbf{x}^2$ is greater than $\overline{\mathbf{x}^2} + 3S_x^2$, an abnormal warning of information system can be sent out.

In formula (3) of Chi-square test statistics, in order to simplify the calculation, the relationship between multiple variables does not be considered. It can avoid the matrix operations, reduce the time of abnormal detection and improve the efficiency of abnormal detection. It is a practical and effective method.

Algorithm of Chi-square test technology for abnormal detection

Definitions

EWMA (Exponentially Weighted Moving Average) is a statistical technique [5]. The vector ($\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_n$) is defined at the time period [t-k, t]. According to the EWMA technology, $\mathbf{x}_i(t)$ is the value of i' th component of vector in observation time t. The calculation of $\mathbf{x}_i(t)$ is as following:

$$\mathbf{x}_i(t) = \lambda * 1 + (1-\lambda) * \mathbf{x}_i(t-1) \quad (4)$$

Formula (4) described that the event i occurred in the observation time t.

$$\mathbf{x}_i(t) = \lambda * 0 + (1-\lambda) * \mathbf{x}_i(t-1) \quad (5)$$

Formula (5) described that the event i did not occur in the observation time t. λ in formula (4) and (5) is the Smoothing constant, or known as the decay rate. The weight of the λ is λ in observation time t. The weight of the λ is（1-λ）in observation time t-1. The weight of the λ is λ(1-λ)k in observation time t-k. We initialized $\mathbf{x}_i(0) = 0$ and took the typical λ value is 0.3.

Algorithm description

Algorithm based on following idea. With a group of test data during normal information systems operation, the following steps 1 to 6 is excited and $\mathbf{x}^2 + 3S^2{}_x$ is be Calculated and used as an upper limit value of system exception warning. At any time, an abnormal warning is sent out when the calculated $\mathbf{x}^2$ value is greater than the upper limit value in following step 1 to 5.

Step 1: In the time period [t-k, t], the description of a relationship between time and event type is shown in Fig.1.

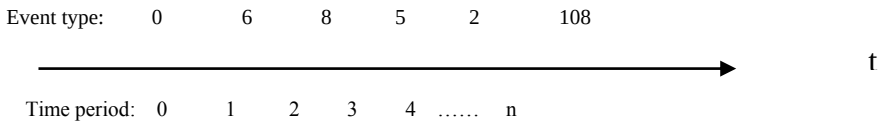

Fig. 1 relationship between time and event type.

Step 2: Assuming t = 0, no event occurs, the component value of vector $(\mathbf{x_1}, \mathbf{x_2}, \ldots \mathbf{x_n})$ is all of zero.

Step 3: According to the formula (4) and (5), the value of vector $(\mathbf{x_1}, \mathbf{x_2}, \ldots \mathbf{x_n})$ was calculated at different time ti. For example:

$\mathbf{x_6}$ =0.3*1+0.7*0=0.3, where t = 1

The other component values of vector $(\mathbf{x_1}, \mathbf{x_2}, \ldots \mathbf{x_n})$ are all of zero.

$\mathbf{x_8}$ = 0.3 * 1 +0.7 * 0 = 0.3, $\mathbf{x_6}$ = 0.3 * 0 +0.7 * 0.3 = 0.21, where t = 2

In a similar way, the other components value of vector $(\mathbf{x_1}, \mathbf{x_2}, \ldots \mathbf{x_n})$ can be calculated at time ti.

Step 4: In order to obtain $\overline{\mathbf{x}_i}$ (denoted as $\overline{\mathbf{x}_{k,i}}$) at time ti, we use the following recursive calculation:

$$\overline{\mathbf{x}_{k,i}} = ((k-1)\overline{\mathbf{x}_{k-1,i}} + \mathbf{x}_{k,i})/k \quad (6)$$

In formula (6), K is the number of current event, namely, the value of ti. For example, as shown in Fig. 1, in time t = 3, k = 3. $\overline{\mathbf{x}_{k-1,i}}$ denoted $\overline{\mathbf{x}_i}$ at time ti-1 and $\mathbf{x}_{k,i}$ denoted $\mathbf{x}_i$ at time ti.

$\overline{\mathbf{x}_{1,6}} = ((1\text{-}1)\ \mathbf{x}_{1-1,6} + \mathbf{x}_{1,6}) / 1 = (0 * 0 + 0.3) / 1 = 0.3$, the rest values of $\overline{\mathbf{x}_{1,i}}$ are all zero. where t = 1, k = 1;

$\overline{\mathbf{x}_{2,8}} = ((2-1)\ \overline{\mathbf{x}_{2-1,8}} + \mathbf{x}_{2,8}) / 2 = (1 * 0 + 0.3) / 2 = 0.15$

$\overline{\mathbf{x}_{2,6}} = ((2-1)\ \overline{\mathbf{x}_{2-1,6}} + \mathbf{x}_{2,6}) / 2 = (1 * 0.3 + 0.21) / 2 = 0.255$

Rest values of $\overline{\mathbf{x}_{2,i}}$ are all zero. where t = 2, k = 2;

Step 5: according to formula (3), respectively, the values $\mathbf{x}_i^2$ is obtained for each time ti. Where, m is the number of times, 0 <= i <= m.

Step 6: $\overline{\mathbf{x}^2}$ and S_x^2 is calculated based on the following formula (7) and (8).

$$\overline{\mathbf{x}^2} = (\sum_{i=1}^{m} \mathbf{x}_i^{\,2}) / m \quad (7)$$

$$S^2{}_x = sqrt(((\sum_{i=1}^{m} (\mathbf{x}_i^{\,2} - \overline{\mathbf{x}^2})^2) / (m-1)) \quad (8)$$

Experimental application results in LINUX and Windows 8 systems

Experimental data source

We have a class of experimental data from the LINUX system. The simulation is for the host intrusion. The system has a program called BSM (Basic Security Module) security system. BSM supports security monitoring host by recording security-related events.

Records of BSM security include the event type, user ID, group ID and process ID and so on. Abnormal detection is only to extract the event type from BSM. In

the BSM audit events, there are all 284 different types of events. The maximum value of the vector $\mathbf{x}_i$ is 284 in above algorithm.

In order to detect intrusion, the Profile of normal event needs to be established. The audit data are also needed. Our sample data contain 5019 audit events. 2613 audit events are used to train normal Profile. The remaining 2,406 audit events are used for test.

To obtain the audit data of invasion event, we simulated a number of invasion scenario. For example: guess password, steal the root user's permissions and so on. These invasion scenarios contain 2225 audit events. These 2225 audit events are also used for testing.

The other experimental data came from a Windows 8 host. Normal and abnormal invasion is simulated. The "Event Viewer" of "Administrative Tools" in the "Control Panel" is selected. Event type from "Security Log" was extracted. Using the 2026 audit events, the normal Profile is trained. For testing stage, processing is different from LINUX system. The amount of data is small for each test. However, the number of testing is added. Specifically, for the normal audit data, 289 event types are used. 8 times repeating extraction is done and then average value is get. For the audit data of intrusion, 42 audit event types are used. 8 times repeating extraction is done and then average value is get. Table1 is the experimental data.

Table1 Source of experimental data

	The number of audit events in training stage	Testing stage	
		The number of normal audit events	The number of invasion audit events
LINUX system host	2613	2406	2225
Windows 8 system host	2026	289*8	42*8

Analysis of experimental results

The detection software that can read the training and test files was developed using Visual C + +6.0. The software calculate the maximum, minimum, average and standard deviation values of $\mathbf{x}^2$ according to the algorithm provided above , and display the number of abnormal alarm. Table 2 is the experimental results of LINUX system host using the training and test data. Table 3 is the experimental results of Windows 8 system.

Table 2 Experimental data of LINUX system

Detection rate for invasion event	Detection rate for invasion plot	Upper limit $\overline{\mathbf{x}^2}+3S_x^2$	Testing data	Minimum value of $\mathbf{x}^2$	Maximum value of $\mathbf{x}^2$	average value of $\mathbf{x}^2$	standard deviation values of $S^2{}_x$
			Normal data	0.41	4.93	1.74	0.91
76%	100%	6.81	Invasion data	0.97	69988	3121	7322

Table 3 Experimental data of Windows 8 system

Detection rate for invasion event	Detection rate for invasion plot	Upper limit $\overline{\mathbf{x}^2}+3S_x^2$	Testing data	Minimum value of $\mathbf{x}^2$	Maximum value of $\mathbf{x}^2$	average value of $\mathbf{x}^2$	standard deviation values of $S^2{}_x$
			Normal data	0.0272	6.3176	1.1817	0.9814
78%	100%	4.18	Invasion data	0.0608	12.0753	2.8106	2.7015

The upper limit value $\overline{\mathbf{x}^2}+3S_x^2$ of testing data obtained with the LINUX OS is 6.81. Experiments show that when the amount of testing data is changed, the upper limit value has little change. The range of change is around 6.81. For the LINUX system, 6.81 can be used as the upper limit of anomaly detection warning. Experiments also show that there is no invasion signal when completely normal with a known type of event detection. There is no invasion of warning signals. False alarm rate 0f false positive was 0%. When the invasions of 2225 were detected, the warning signal to obtain 1691, the audit event detection rate was 76% (1691/2225), as shown in Table 2.

The upper limit is 4.18 when the test data obtained with Windows 8. The value is less than 6.81 of LINUX systems. However, consistent with the LINUX system, when the normal amount of test data sample sizes is changed, the values of a little change around 4.18 can be used as the upper limit of anomaly detection warning of Windows 8 system. Experiments show that, when 289 normal event types known is used to detected, the results of 8 times is consistent. No invasion of warning signals occurs. Therefore, false positive rate of audit event is 0%. When the 42 invasion events were detected, the average detection rate of 8 times is 78%. The result of a particular experiment is shown in Table 3.

An invasion plot may contain multiple event types. Experiments of two systems prove that if 2225 invasion events (or 42) are arranged to each invasion plot, there is warning signal for each invasion plot. The detection rate is 100% for all invasion plots. There is 73% invasion plot is detected in the first audit event. Warning signal occurs in the first audit event of invasion plot. The other 27%

invasion plot is detected in the second audit event. Warning signal occurs in the second audit event of invasion plot.

Using different quantity of data in two different systems, the experiment result is that there are 0% false positive rate and 71 ~ 75% detection rate for invasion plot. The smallest detection rate is 35% and the maximum detection rate is 86%. There will be warning signal occurs in early period of invasion plot.

0% false positive rate and 71 ~ 75% detection rate mean that the invasion contains some normal activities. We do not expect that every audit event of invasion plot is detected. However, Table 2 and Fig. 2 show that intrusion is different from the normal activities.

In our study, the behavior of the set of every audit is defined as the plot. Detection rate depends of plot. The maximum detection rate of normal plot is 0% and minimum detection rate of intrusion plot is 35%. Decision threshold can be chosen between detection rates 0% and 35% in which the warning signal occurs. Using the decision threshold, we can clearly distinguish the invasion plot and normal plot. The detection rate of 100% and 0% false positive rate can be gotten.

On the other hand, people want that the warning signal occurs in the early period of invasion plot. Three levels which include normal, forecast and formal warning are established between accurate detection and early warning. For example, three different sounds can be used to denote different levels. When the first warning signal is received, the "prediction" level is triggered which causes the system administrator's attention. The detection rate is continued to accumulate. If the decision threshold is arrived, the formal warning level is triggered. If the plot is finished and accumulated rate does not exceed the threshold decision, forecast level is canceled and normal level is set.

In short, the experiment results show that the detection rate of 100% and 0% false positive rate can be gotten based Chi-square statistics. The invasion plot can be detected in the moment of the first or second audit event occurring.

Summary

We further study is to collect the test data of different systems and increase the total amount of test data. The rationality and correctness of upper limit value which are very important data for anomaly detection will be verified. In order to prove validity and practicability of Chi-square technique for the information system in anomaly detection, real-time monitoring software will be developed for real-time monitoring of information systems.

Acknowledgement

This research was financially supported by the Software Specialty of ShenZhen Polytechnic.

References

[1] K.C, G. Fink, K.Levitt, Execution monitoring of security critical programs in distributed systems, A specification-based approach, Proceedings of the 1997 IEEE Symposium on Security and Privacy[C], Oakland,CA: IEEE Computer Society Press1997, pp.134–144.

[2] HS. Javitz, A. Valdes, The SRI statistical anomaly detector," Proceedings of the 1991 IEEE Symposium on Research in Security and Privacy, Oakland,CA, IEEE Computer Society Press,1991.

[3] Y. Jou, F. Gonh ,C. Sargor, X. Wu, S. Wu, H. Chang , F. Wang F, Design and implementation of a scalable intrusion detection system for the protection of network infrastructure, Proceedings of the DARPA Information Survivability Conference and Exposition. Los Alamitos, CA, IEEE Computer Society, 2000, pp. 69–83.

[4] S. Forrest, SA. Hofmery, A. Somayaji, Computer immunology. Communications of the ACM, 1997, 40(10), pp.88–96.

[5] D. C. Montgomery, Introduction to statistical quality control, Fourth Edition. New York, John Wiley & Sons, 2001.

[6] Yu Wangfeng, Feng Caoshou, etc., Method of Detecting Application-Layer DDOS Based on the Out-Linking Behavior of Web Community, Journal of Software,2013(6), pp.1263-1273.

[7] N. Ye, Q. Chen , An anomaly detection technique based on a Chi-square statistic for detecting intrusions into information systems, Quality & Reliability Engineering, 2001,17(2), pp.105-112.