A Covariance Matrix Based Approach to Internet Anomaly Detection

Shuyuan Jin[1], Daniel So Yeung[1], Xizhao Wang[2], and Eric C.C. Tsang[1]

[1] Department of Computing, HongKong Polytechnic University, HongKong
{cssyjin, csdaniel, csetsang}@comp.polyu.edu.hk
[2] School of Mathematics and Computer Science, Hebei University, Baoding, China
wangxz@mail.hbu.edu.cn

Abstract. Detecting multiple network attacks is essential to intrusion detection, network security defense and network traffic management. This paper presents a covariance matrix based detection approach to detecting multiple known and unknown network anomalies. It utilizes the difference of covariance matrices among observed samples in the detection. A threshold matrix is employed in the detection where each entry of the matrix evaluates the covariance changes of the corresponding features. As case studies, extensive experiments are conducted to detect multiple DoS attacks – the prevalent Internet anomalies. The experimental results indicate that the proposed approach achieves high detection rates in detecting multiple known and unknown anomalies.

1 Introduction

Detecting multiple network attacks is essential to intrusion detection, network security defense and network traffic management. For example, effective detection of multiple attacks can guarantee the good performance of an intrusion detection system (IDS). All of the on-line intrusion-prevention systems (IPS), such as Internet Security Systems (ISS) Proventia G Series, NetScreen Technologies' NetScreen-IDP 100 and TippingPoint, have some level of attack detection mechanisms to identify malicious traffic [7]. Detecting multiple attacks also helps the Internet Service Providers (ISP) to effectively manage the traffic and improve the Quality of Service(QoS) to end-users.

Generally speaking, two kinds of strategies exist in the field of intrusion detection: misuse and anomaly. Misuse detection utilizes signature-matching techniques to indicate the presence of an attack. It is only effective and practical to detect already-known attacks. Anomaly detection utilizes the significant deviation from the normal profile to identify suspicious behaviors. Compared with misuse detection, anomaly detection approaches offer an advantage of identifying unknown attacks.

In the context of Internet anomaly detection, statistical detection approaches are widely employed. Normally these statistical methods utilize the first-order statistical inferences of network features provided by the monitoring devices [1] [2] [3]. In this paper, we present a second-order statistical method to detect the cumulative changes exhibited by the network packet sequences of equal and fix length. The detection approach utilizes the covariance matrix to model the observed network packets. Totally different from traditional anomaly detection techniques, where covariance matrix

D.S. Yeung et al. (Eds.): ICMLC 2005, LNAI 3930, pp. 691 – 700, 2006.

structure is estimated to analyze the noise [4] [5] [6], the covariance matrix based detection approach presented in this paper utilizes the difference among covariance matrices directly in the detection. Under the effect of different thresholds for different classes obtained by training, the significant changes among covariance matrices are revealed in detecting different types of attacks.

Our design makes use of the following basic facts. First, the covariances among different first-order network features have specific meanings in the network engineering. For example, the covariance changes among first-order features (such as SYN and FIN) will indicate the ongoing phenomenon, i.e., a SYN flooding attack [10]. Second, as a statistical variable, covariance or covariance matrix should be calculated from a collection of data. Facing the large volume of network traffic, it is more reasonable to consider the traffic within a determined time interval or a fixed sequence length.

The rest of this paper is organized as follows. Section 2 describes the anomaly detection approach in details. Section 3 validates the performance of our approach by experiments. Section 4 gives some discussions and draws a conclusion.

2 Approach

2.1 Detection Algorithm

We regard the problem of detecting multiple attacks as a multi-class classification problem. The classifier should be able to not only distinguish multiple known classes, but also identify the unknown classes.

Suppose that we have samples from R already known classes: $\omega_1, \omega_2, \ldots, \omega_R$. For each class ω_r $(1 \le r \le R)$, its training set ω_r consists of all the corresponding covariance matrices calculated on the sample sequences of equal, fixed length. For example, when selecting the sequence length as n, M_r^1 is obtained by calculating all the samples $x_1^1, x_2^1, \ldots, x_n^1$ in the temporal sequence T_1; and M_r^l is obtained from the samples $x_1^l, x_2^l, \ldots, x_n^l$ in the temporal sequence T_l. A covariance matrix M_r^l based on all the samples $x_1^l, x_2^l, \ldots, x_n^l$ in the temporal sequence T_l is defined as:

$$M_l = \begin{pmatrix} \sigma_{f_1^l, f_1^l} & \sigma_{f_1^l, f_2^l} & \cdots & \sigma_{f_1^l, f_p^l} \\ \sigma_{f_2^l, f_1^l} & \sigma_{f_2^l, f_2^l} & \cdots & \sigma_{f_2^l, f_p^l} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{f_p^l, f_1^l} & \sigma_{f_p^l, f_2^l} & \cdots & \sigma_{f_p^l, f_p^l} \end{pmatrix} \tag{1}$$

$$\sigma_{f_u^l, f_v^l} = \frac{1}{n} \sum_{k=1}^{n} \left(f_u^{l,k} - \mu_{f_u^l} \right) \left(f_v^{l,k} - \mu_{f_v^l} \right) \tag{2}$$

where

f_u and f_v is the first-order features of the observed network packet

$\mu_{f_u^l}$ $(1 \le u \le p)$ is the expectation of f_u

$f_u^{l,k}$ is the value of f_u in the k-th observation during the l-th time interval

u is the number of features $(1 \le u \le p)$

k is the number of observations during T_l $(1 \le k \le n)$

l is the number of time intervals, such as $T_1, T_2, \ldots, T_l, \ldots$ $1 \le l \le \infty$

Assume that we can get a total of l covariance matrices for class $\omega_r : \{M_r^1, M_r^2 ..., M_r^l\}$ according to the above described procedure. The classifier will assign a class label to a presented M^{obs} in the detection according to the following discrimination function:

$$\begin{cases} if\ Dist(M^{obs}, E(\omega_r); \delta_r) = [0]_{p \times p}, & M^{obs} \in \omega_r \\ if\ \forall r, Dist(M^{obs}, E(\omega_r); \delta_r) \neq [0]_{p \times p}, & M^{obs} \in unkown\ class \end{cases} \tag{3}$$

where $E(\omega_r)$ is the mean of class ω_r and δ_r is the settled threshold matrix of class $\omega_r, 1 \le r \le R$. The dissimilarity function $Dist(M^{obs}, E(\omega_r); \delta_r) = [0]_{p \times p}$ is defined as

$$\forall m_{ij}^{obs} \in M^{obs}, \forall \bar{m}_{ij}^r \in E(\omega_r), \forall \delta_{ij}^r \in \delta_r, \forall d_{ij} \in Dist(M^{obs}, E(\omega_r); \delta_r)$$

$$\begin{cases} d_{ij} = 0 & if\ \left| m_{ij}^{obs} - \bar{m}_{ij}^r \right| \le \delta_{ij}^r \\ d_{ij} = 1 & if\ \left| m_{ij}^{obs} - \bar{m}_{ij}^r \right| > \delta_{ij}^r \end{cases} \tag{4}$$

Equations (3) and (4) mean that for an observed covariance matrix M^{obs}, the classifier will classify it to the same label as the average of any one of known classes ω_r, if and only if $Dist(M^{obs}, E(\omega_r); \delta_r) = [0]_{p \times p}$. For example, M^{obs} will be considered as normal when M^{obs} is δ_N-matrix nearest to the center of the normal class ω_N in all $p(p+1)/2$ different positions in the difference matrix of M^{obs} and ω_N. If we could not find any one of known classes to M^{obs} to be δ_r-matrix nearest, M^{obs} will be determined as the novelty.

2.2 Threshold Determination

The determination of multiple thresholds for multiple known classes is a complex optimization problem. Especially, that the threshold is a matrix as proposed in Equations (3) and (4) makes its determination more difficult. Here we propose a relatively simple but practical threshold determination algorithm. The algorithm attempts to set every entry of the threshold matrix as a value which covers all the variances of the corresponding covariance changes. Especially, the maximum statistic of the covariance changes is utilized as the initial value of each element in the threshold matrix. The aim of the threshold matrix determination algorithm is to achieve the minimal

misclassification rate for each training class. We increase or decrease the threshold matrix by multiplying the threshold matrix with different multipliers. Correspondingly, the classification precision rate and misclassification rate will change with different product threshold matrices. Generally, we can obtain a set of product threshold matrices which can make the misclassification rate achieve the minimum. We select the minimal multiplier from the product set as the preferred threshold matrix multiplier. The preferred threshold matrix is the product of the preferred threshold matrix and the initial threshold matrix. In order to illustrate the threshold matrix determination process, a realization of threshold matrix determination for normal class is provided in Fig. 1. The threshold matrix determination of other attack classes will have the similar process. Because a misclassification of any normal sample will signal a false alarm, the false alarm rate is used in Fig. 1 instead of misclassification rate.

Notations

ω_r : Traing Data Set, $1 \le r \le R$

ϖ_r : Testing Data Set, $1 \le r \le S,\ S > R$

N: Normal Class Label, $N \in [1,R]$

ω_N: Traing Data Set of Normal Class

ϖ_N: Testing Data Set of Normal Class

δ_N : Threshold Matrix of Normal Class

Assume ω_N has a total of l covariance matrix training samples

$\forall \delta_{ij}^N \in \delta_N,\ \forall m_{ij}^N \in \omega_N, \forall \delta_{ij}^N \in \delta_N,$

init $\delta_{ij}^N = \max_{k=1}^{l} \left| m_{ij}^N - \bar{m}_{ij}^N \right|$, *multiplier*=0.2;

do {

$\delta_N = \delta_N * multiplier$

for each $M \in \varpi_N$

if $Dist(M, E(\omega_N); \delta_N) \neq [0]_{p \times p}$ count=count+1;

endfor

FAR(False Alarm Rate) = count/$\|\varpi_N\|$

multiplier=*multiplier*+0.01;

} until (*FAR* is converged or minimum)

Fig. 1. A realization of threshold determination algorithm for normal class

2.3 Detection Rules

As an on-line detection system, our detection approach employs a rule-like sequential detection procedure. Assume there are R classes in the training stage and we have obtained R threshold matrices according to the above threshold determination algorithm.

We label the training classes as follows. The normal class is labeled as class ω_1 and the attack classes are labeled based on the sample numbers they have. The more samples the attack class has, the smaller its label is. Therefore, a labeled training class sequence will be obtained as $\omega_1, \omega_2, \omega_3, ..., \omega_R$. The detection process is demonstrated in Fig. 2.

$\forall M^{obs}$

if $Dist(M^{obs}, E(\omega_1); \delta_1) = [0]_{p \times p}$, then $M^{obs} \in \omega_N$

elseif $Dist(M^{obs}, E(\omega_2); \delta_2) = [0]_{p \times p}$, then $M^{obs} \in \omega_2$

elseif $Dist(M^{obs}, E(\omega_3); \delta_3) = [0]_{p \times p}$, then $M^{obs} \in \omega_3$

… …

else $M^{obs} \in$ unknown attack

Fig. 2. A rule-like realization of on-line detection

3 Experiments

3.1 Data and Feature Used

The dataset we use is KDD CUP 99 Dataset at http://kdd.ics.uci.edu/databases /kddcup99/kddcup99.html. It is constructed based on the raw data of TCP dump data from 1998 DARPA evaluations [11] for the purpose of network intrusion detector competition. The datasets contain a total of 24 training attack types, with an additional 14 types in the test data only. In our experiments, only the DoS attack classes which contain records greater than 200 are considered. The main reason is that it is not always possible to formulate a classification model to learn the anomaly detector with "insufficient" training data [12]. There are totally 6 types of DoS attacks in our experiments which include 3 known DoS attack types and 3 unknown DoS attack types. We use 3/5 samples as training set and 2/5 samples as testing set for each selected class. The detailed dataset description in our experiments is presented in Table 1.

We employ all the 9 time-based traffic features in our experiments. They are the features named *count, serror_rate, rerror_rate, same_srv_rate, diff_srv_rate, srv_count, srv_serror_rate, srv_rerror_rate* and *srv_diff_host_rate*. A detailed description of the 9 time-based traffic features is provided in [13].

Table 1. Data set description

Type	Training samples	Testing Samples
Normal	94722	63148
Smurf	266929	177952
Back	1981	1320
Neptune	99122	66080
apache2	0	794
mailBomb	0	5000
processtable	0	759

3.2 Results

In our experiment, the sequence length parameter n is set to 200 with a sliding window of 50, while the preferred threshold δ_r for each known classes is obtained in the training procedure described in Section 2.2.

As we know, different threshold matrix corresponds to different classification precision rate in detecting different class. Here we take the detection of the normal class as an example. Fig. 3 illustrates the performance of different threshold matrix used in detecting the normal class, in terms of different pairs of detection rate and false alarm rate.

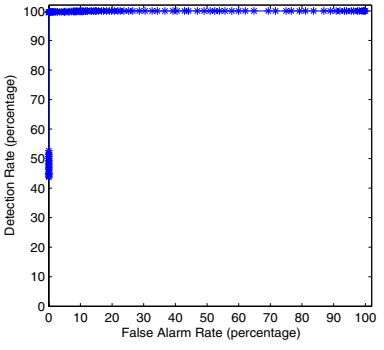

Fig. 3. ROC curve of the covariance matrix based detection approach

***Remarks**:* Fig. 3 shows that the covariance matrix based detection approach achieves very high detection rates with very low corresponding false alarm rates. Two reasons contribute to this high detection results. One is the dataset itself. As we know that many flaws exist in the KDD CUP dataset [14], the normal traffic provided in the dataset is somewhat too simple. The other reason is that the threshold utilizes a matrix rather than a scalar to evaluate the covariance changes. Each entry in the threshold matrix evaluates the changes of the covariance of two corresponding features. Therefore, if the observed covariance matrix (e.g. the samples of the attack class) should not be labeled as the provided class profile (e.g. the profile of normal class), it is very easy to happen that the changes of some elements in the observed covariance matrix exceed the corresponding element in the threshold matrix, which will result in the failure of labeling the observed samples as the label of the provided profile. However, we will also notice that each entry of the initial threshold matrix is settled as the maximum statistic of the covariance changes, which increase the opportunity of labeling the observed sample of covariance matrix as its corrected class label. Therefore, the false alarm rate of the normal class or the misclassifications rates of the attack classes will be very low while the classification precision rate will be very high in the covariance matrix based detection approach.

Table 2 shows some pairs of false alarm rate and detection rate in details when different multipliers of the initial threshold matrix for normal class are used (refer to Fig. 1 for detailed threshold determination algorithm for the normal class).

Table 2. Different pairs of false alarm rate and detection rate under different threshold matrices of the normal class

Threshold Multiplier	False Alarm Rate	Detection Rate
1.15	0	99.60%
1.05	0.32%	99.66%
0.95	2.78%	99.70%
0.85	7.94%	99.84%
0.75	10.88%	99.86%
0.65	16.68%	100.00%

In order to show the performance difference of using different threshold matrices in detecting normal class, we list the classification precision rates in detecting testing samples of the normal class, under different threshold matrix with different multipliers of 1.15 and 0.65 (refer to Table 2), respectively. The classification precision results are shown in Table 4, where *New* represents the unknown attack class. Each entry in Table 3 shows the classification rate of detecting the testing samples of normal class as different known classes. For example, the entry of (2,2) in Table 3 means that the classification rate of detecting the testing samples of the normal class as the normal class is 100% under the threshold matrix with multiplier 1.15.

Table 3. Classification precision rates of detecting the testing samples of the normal class as known classes and unknown attack

Threshold Multiplier	normal	smurf	back	neptune	New
1.15	100.00%	0	0	0	0
0.65	83.32%	0	0	0	16.68%

In order to show the overall performance of the covariance based detection approach in detecting multiple known and unknown attacks, we list the classification results in Table 4. Because the threshold matrix determination algorithm proposed in Section 2.2 is only a practical solution, we only use 1.15 as the threshold multiplier for the normal class and simply use the initial threshold matrix as the preferred threshold matrix for each known attack class.

Table 4. Classification results of distinguishing multiple known and unknown classes using the preferred threshold in the threshold determination algorithm

	normal	smurf	back	neptune	New
normal	100.00%	0	0	0	0
smurf	0	100.00%	0	0	0
back	86.96%	0	4.35%	0	8.70%
neptune	0	0	0	98.86%	1.14%
apache2	0	0	0	0	100.00%
mailBomb	0	0	0	0	100.00%
processtable	0	0	0	0	100.00%

Table 5 summarizes the total classification precision rates of the results listed in Table 4, for the different whole dataset of 4 training classes (refer to column *Training*), 7 testing classes (refer to column *Testing*), 3 known attacks (refer to column *Known*) and 3 unknown attacks (refer to column *Unknown*), respectively. The row of *number of samples* presents the sample count of the dataset represented by the column in terms of the covariance matrix, where the covariance matrix samples are obtained under the fixed and equal sequence length 200 with a sliding window 50. The total precision rate is calculated as the number of correctly classified samples divided by the total samples.

Table 5. Total classification results of distinguish multiple known and unknown classes

	Training	Testing	Known	Unknown
Number of Samples	9241	6277	4897	121
Total Precision Rate	99.62%	99.41%	99.24%	100.00%

4 Discussions

The experimental results in Section 3 show that the covariance matrix based detection approach can achieve a high detection rate, high classification precision rate for the normal class, known attack and unknown attack classes. These experimental results in this extended paper are much better than that presented in our original paper [15]. In the original paper, the threshold is realized based on a scalar value, which evaluates the Euclidean distance of the difference matrix between an observed covariance matrix and the profile covariance matrix; while the threshold in this extended paper is based on a matrix where each entry is realized based on the maximum statistics of the covariance changes of two corresponding features. Therefore, the detection performance has improved a lot. It is true that the dataset employed in this paper has some bias on the detection results, because the flaws exist and the data is somewhat a little simple [14]. However, the detection results verify the effectiveness of employing the covariance matrix in the DoS attack detection; particularly, employing a matrix rather than a scalar to evaluate each entry of the observed matrix sample will greatly increase the effectiveness of the detection.

As the detection approach proposed in this paper utilizes statistical covariance matrix directly, another more relevant detection approach we want to mention is the Mahalanobis-distance based detection approach (M-detector). Both methods take into account variance and covariance of the variables measured, but our proposed method is very different from Mahalanobis Distance base detector in the following aspects:

- Similarity measurement: M-detector evaluates the M distance between the observed sample and the different means of different classes, while our method evaluates the difference of matrices between the covariance matrix of samples and the covariance matrix of different classes.
- Feature space: the feature space of M-detector consists of the samples of signals, while the feature space of our proposed method consists of the covariance matrix of a group of sampled data.

Compared with other outlier detection methods, our proposed method takes advantage of the following desirable characteristics:

- The employment of the *covariance* statistics makes the dissimilarity prominent among the normal and different types of attacks.
- The covariance matrix used in our detection approach has specific meaning, which characterizes each attack in terms of dispersion of its own corresponding *first-order* feature pairs.
- The detection approach overcomes the drawback [7] of the dependency of data specific distribution in traditional IDES/NIDES based anomaly detection techinques.

More research needs to be done. For example, we need to know how to evaluate the effect of physical features on the covariance feature space. We also need to conduct more experiments on different datasets. But these future works do not undermine the discussion of anomaly detection in this paper. The proposed covariance based second order statistical detection approach can be served as a new tool in detecting multiple anomalies.

Acknowledgements

This research work is supported by the Hong Kong RGC Project Research Grant B-Q571 and the Hong Kong Polytechnic University Research Grant GT-891.

References

1. Feinstein, L., Schnackenberg, D.: Statistical Approaches to DDoS Attack Detection and Response. Proceedings of the DARPA Information Survivability Confe-rence and Expostion(DISCEX'03), April 2003.
2. Manikopoulos, C., Papavassiliou, S.: Network Intrusion and Fault Detection: A Statistical Anomaly Approach. IEEE Communications Magazine, October 2002.
3. Blazek, R. B., Kim, H., Rozovskii, B., Tartakovsky, A.: A Novel Approach to Detection of Denial-of-Service Attacks Via Adaptive Sequential and Batch-Sequential Change-Point Detection Methods. Workshop on Statistical and Machine Learning Techniques in Computer Intrusion Detection, June 2002.
4. Conte, E., Maio, A.De., Ricci, G.: Covariance matrix estimation for adaptive CFAR detection in compound-Gaussian clutter. IEEE Transactions on Aerospace and Electronic Systems, Volume: 38, Issue: 2, April 2002.
5. Yang, Z., Wang, X.: Blind turbo multiuser detection for long-code multipath CDMA. IEEE Transactions on Communications, Volume: 50, Issue: 1, Jan. 2002.
6. Conte, E., Maio, A.De., Ricci, G.: Recursive estimation of the covariance matrix of a compound-Gaussian process and its application to adaptive CFAR detection. IEEE Transactions on Signal Processing, Volume: 50, Issue: 8 , Aug. 2002
7. Ye, N., Emran, S,M., Chen, Q., Vilbert, S.: Multivariate Statistical Analysis of Audit Trails for Host-Based Intrusion Detection. IEEE Transaction on COMPUTERS, vol.51, No. 7, 2002.
8. G. Cormode, S. Muthukrishnan. What's New: Finding Significant Differences in Network Data Streams. IEEE INFOCOM 2004. March, 2004.

9. Estan, C., Varghese, G.: Data streaming in computer networks. In proceedings of workshop on Management and processing of Data Streams. http://www.research.att.com/conf/mpds2003/schedule/estanV.ps, 2003.
10. Jin, S., Yeung, D.: A Covariance Analysis Model for DDoS Attack Detection. Proceedings of IEEE ICC'2004. Paris, France, June 2004.
11. Lincoln Laboratories: 1999 DARPA Intrusion Detection Evaluation. http://www.ll.mit.edu/IST/ideval/index.html.
12. Lee, W. : A Data Mining Framework for Constructing Features and Models for Intrusion Detection Systems. Ph.D. dissertation, Columbia University,1999.
13. Lee, W., Stolfo, S.: A Framework for Constructing Features and Models for Intrusion Detection Systems. ACM Trans. Information and System Security, vol. 3, no. 4, pp. 227-261, Nov. 2000.
14. Mahoney M.V., Chan, P. K.: An Analysis of the 1999 DARPA/Lincoln Laboratory Evaluation Data for Network Anomaly Detection. RAID, pp. 220–237, 2003.
15. Jin S., Yeung, D., Wang, X., Tsang, E. C.C.: A Second-order Statistical Detection Approach with Application to Internet Anomaly Detection. IEEE International Conference on Machine Learning and Cybernetics, August, 2005.