Distance Measures for Anomaly Intrusion Detection

Wei Wang, Sylvain Gombault

▶ **To cite this version:**

Wei Wang, Sylvain Gombault. Distance Measures for Anomaly Intrusion Detection. SAM'07: the 2007 International Conference on Security and Management, Jun 2007, Las Vegas, United States. pp.17 - 23. hal-01898076

HAL Id: hal-01898076
https://hal.science/hal-01898076

Submitted on 18 Oct 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Submitted to

SAM'07- The 2007 International Conference on Security and Management

Title of the paper:

Distance Measures for Anomaly Intrusion Detection

The details of the authors:

Wei Wang and Sylvain Gombault
Department of Networks, Security and Multimedia
GET/ENST Bretagne
2, rue de la Châtaigneraie
35512 Cesson Sévigné Cedex, France
Tel: +33-299127045 (for W. Wang) and +33-299127037 (for S. Gombault)
Fax: +33-299127030
E-mail: wang.wei@enst-bretagne.fr, sylvain.gombault@enst-bretagne.fr

We declare that Dr. Wei Wang will be presenting the paper (if accepted) at Las Vegas based on the financial support of the DADDi project and of the department.

Keywords of the paper:

Intrusion detection; masquerade detection; distance measures, k-nearest nearest neighbor; chi-square test

General Chair of WORLDCOMP'07
Dear Prof. Hamid R. Arabnia,

We would like to submit the attached original research full manuscript, entitled "Distance Measures for Anomaly Intrusion Detection", for your consideration for possible presentation in SAM'07. We believe that this manuscript well suits the topics of SAM'07 and has some contributions to the related fields.

Should you have any questions please contact us.

Sincerely,

Dr. Wei WANG (Corresponding author)

Distance Measures for Anomaly Intrusion Detection

Wei Wang, Sylvain Gombault

Department of Networks, Security and Multimedia
GET/ENST Bretagne
2, rue de la Châtaigneraie
35512 Cesson Sévigné Cedex, France
Wang.Wei@enst-bretagne.fr, Sylvain.Gombault@enst-bretagne.fr

Abstract--*Instead of using the plain frequency of audit data, this paper presents several novel cross frequency weights to model user and program behaviors for anomaly detection. The frequency weights are plain Term Frequency (TF) and various term frequency-inverse document frequency (tfidf) defined as Ltfidf, Mtfidf and LOGtfidf respectively. Nearest Neighbor (NN) and K-NN methods with Euclidean and Cosine distance measures as well as Chi-square test method based on these frequency weights are used for anomaly detection. Extensive experiments are performed based on command data from Schonlau et al. and the results show that the LOGtfidf weight gives better detection performance compared with plain frequency and other types of weights, and Eculidean distance gives better detection performance compared with Cosine distance measure while the Chi-square test consistently returns the worst results. By using the LOGtfidf weight, the simple NN method achieves the better masquerade detection results than the other 7 methods in literature. The LOGtifidf weight improves the detection results with 27.9% than plain TF and improves with 30.8% than Ltfidf based on the NN method. The sendmail system call data from University of New Mexico (UNM) are used as well and the results also demonstrate the effectiveness of the LOGtfidf weight for detection of anomalous program behavior.*

Keywords--Intrusion detection; masquerade detection; distance measures, k-nearest nearest neighbor; chi-square

1. Introduction

Intrusion detection is an important technique in the defense-in-depth network security framework and has become a widely studied topic in computer security in recent years [1]. In general, the techniques for intrusion detection fall into two major categories depending on the modeling methods used: signature-based detection and anomaly detection. Anomaly detection has been an active research area because of its capability of detecting new attacks [2]. In most computing environments, the behavior of a subject (e.g. a program, a user or a network element, etc.) is observed and analyzed via the available computer audit data [1]. It is always a big challenge to extract important and suitable features that best characterize behavioral patterns of a subject so that abnormality can be clearly distinguished from normal activities.

Most previous work in anomaly detection considered two probabilistic attributes of activities in computer systems and networks, namely, the transition attributes and the frequency attributes of the audit data. One can also call these two attributes as dynamic models and static models [3] or time series and non-time series [4]. In 1996, Forrest et al. [5] introduced an anomaly detection method called *stide* (Sequence TIme-Delay Embedding) by using a fixed length of system calls invoked by active and privileged processes. Profiles of normal program behavior were built by enumerating all fixed length of distinct and contiguous system calls that occur in the training data sets and unmatched sequences in actual detection are considered anomalous. This method can be considered as using the transition information contained in the audit data and was also called as methods. In subsequent research, most research in detecting anomalous program behavior has used fixed length sequences of system calls as observable (*n-gram*). For example, Lee et al. [6] used data mining approach to study a sample of system call data and characterize the sequences contained in normal data by a small set of rules. The sequences violating those rules were then treated as anomalies. Warrender et al. [7] proposed a Hidden Markov Model (HMM) based method for modeling and evaluating invisible events. This method was further studied by many other researchers [3, 8-9]. Lee et al. [1] used information-theoretic measures for anomaly detection. Masquerade detection is as important as anomalous program behavior detection. Masquerades are people who impersonate other people on the computer [10] and relatively difficult to be detected. Schonlau et al. [10-11] attempted to detect masquerades by building

normal user behavioral models using truncated command sequences. Experiments with six masquerade detection techniques [10-11]: Bayes one-step Markov, Hybrid multi-step Markov, IPAM, Sequence-Match, Compression and Uniqueness, were performed and compared. The first five methods are mainly based on the transition information of user command data.

The frequency attributes of audit data have also been widely used for intrusion detection. Liao and Vemuri [12] developed an intrusion detection method by using the text categorization techniques based on the frequency attributes of system calls. In subsequence research, Hu et al. [13] applied robust Support Vector Machines (SVM) for intrusion detection based on the frequency attributes of system call data. Zhang et al. [14] modified the conventional SVM, Robust SVM and one-class SVM respectively for intrusion detection also based on the frequency property of system call data. Chen et al. [15] developed Artificial Neural Networks (ANNs) and SVM based methods for detecting potential system intrusions by the frequency property of system call data. Yeung et al. [3] used information measure for detecting anomalous user and program behavior based on the frequency attributes of computer audit data. In our previous work, we employed various data reduction techniques, e.g., Non-negative Matrix Factorization (NMF) [16], Principal Component Analysis (PCA) [17-18] and Self Organizing Maps (SOM) [19], to reduce high dimensional data for intrusion detection with high efficiency and low use of system resources. These techniques are also based on the frequency attributes of computer audit data.

In general, intrusion detection methods based on the transition information model temporal variation of the audit data. The intrusion detection methods using the frequency information, on the other hand, convert the temporal sequences into some non-temporal representation typically in the form of multidimensional feature vectors with no time dimension. Our previous work [19] is consistent with Ye's work [20] and indicates that considering the transition information of audit data can improve detection accuracy but have to sacrifice some real-time performance compared to using the frequency information. In practice, audit data in intrusion detection problem is typically very large. For example, in colleting system calls of *sendmail* on a host machine, only 112 messages produced a combined trace with the length of over 1.5 million system calls [5]. Fast processing of massive audit data in real-time is therefore essential for a practical Intrusion Detection System (IDS) so that actions for response can be taken as soon as possible. However, intrusion detection methods considering the transition information of audit data usually require much time to train the models and to detect intrusions by processing a large amount of data. For example, it took Hidden Markov Models (HMM) approximately two months to train an anomaly detection model with a large data set [7]. This is clearly not adequate for real-time intrusion detection. On the other hand, intrusion detection methods only taking account of frequency information usually cannot achieve good detection accuracy [19-20]. In this paper, we propose a novel intrusion detection method by using frequency weights that not only consider the frequency information of events in each sequence of audit data, but also consider the distribution of the event in the whole data. We may call this kind of data preprocessing as considering cross frequency information of audit data. The weights are originally from information retrieval and text mining and were known as *tfidf* (term frequency - inverse document frequency). These weights almost do not increase the computation expense and are thus suitable for real-time detection. Plain Term Frequency (*TF*) and various frequency weights defined as *Ltfidf*, *Mtfidf* and *LOGtfidf* are used in this paper for feature transformation. Several distance measures, namely, Nearest Neighbor (NN) and K-NN with Euclidean distance and with Cosine distance as well as Chi-square test are used for masquerades detection based on the four weight schemes. Extensive experiments are based on user command data from Schonlau [10-11] and results show that based on the *LOGtfidf* frequency weight of audit data, even simple NN method can achieve the better masquerade detection results than the other 6 methods in [10-11] and also than our previous results using NMF [16]. The *LOGtifidf* weight improves the detection results with 27.9% than plain frequency *TF* and improves with 30.8% than *Ltfidf* based on the same NN method. *Sendmail* system call data from University of New Mexico (UNM) are used as well for further validating the *LOGtfidf* weights and the results also demonstrate its effectiveness for detection of anomalous program behavior.

The rest of this paper is organized as follows. Next section describes the frequency weights and the distance measures for anomaly intrusion detection. Experiments are described and the results are summarized and discussed in Section 3. Concluding remarks follow in Section 4.

2. Distance measures for anomaly intrusion detection

2.1. Feature transformation with various frequency weight schemes

Feature transformation is usually the first step for

anomaly detection. To facilitate comparison, we used various frequency weight schemes for feature transformation. The first one is the plain frequency of events in audit data and we call it as *term frequency* (*TF*). We call the second scheme as *Ltfidf* (Liao's term frequency - inverse document frequency) as it was first used by Liao and Vemuri [12]. The other two schemes have never been used for intrusion detection by now and we called them as *Mtfidf* (Mean tfidf) and *LOGtfidf* (LOG tfidf), respectively. These four frequency weight schemes are described below and the notation and terminology used in this paper are listed in Tab. 1.

Table 1. The notation and terminology

N	Total number of sequences in the observation data set
M	Total number of distinct events in the observation data set
f_{ij}	Frequency of event i in sequence j
n_i	Number of times that event i appears in the observation data set.
s_i	Number of sequences containing event i
X	Test sequence
T	Training sequences in the observation data set

2.1.1. Plain Term Frequency (TF)

Plain Term frequency (*TF*) may itself be used as the basis for feature selection for intrusion detection. Nearly all the current frequency based intrusion detection methods used this kind of measures for feature transformation [12-20]. It can be defined as

$$tf_{ij} = f_{ij} \tag{1}$$

2.1.2. Ltfidf (Liao's term frequency - inverse document frequency)

Liao and Vemuri [12] first used this kind of measure for intrusion detection based on system call data. In subsequent research, Zhang and Shen [14] and Chen et al. [15] also used the same measure for intrusion detection based on system call data. It is defined as

$$Ltfidf_{ij} = \frac{f_{ij}}{\sqrt{\sum_{l=1}^{M} f_{lj}^2}} \times \log\left(\frac{N}{n_i}\right) \tag{2}$$

2.1.3. Mtfidf (Mean term frequency - inverse document frequency)

The *Mtfidf* has been widely used in information retrieval and text mining [21] and we propose to use this scheme for intrusion detection in this paper. It is defined as

$$Mtfidf_{ij} = f_{ij} \times \log\left(\frac{N}{s_i}\right) \tag{3}$$

2.1.4. LOGtfidf (LOG term frequency - inverse document frequency)

LOGtfidf is a revised scheme for feature transformation. The logarithm of the *tf* is to amend unfavorable linearity. It is defined as

$$LOGtfidf_{ij} = \log(0.5 + f_{ij}) \times \log\left(\frac{N}{s_i}\right) \tag{4}$$

2.2. Distance measures for anomaly intrusion detection

2.2.1. K-Nearest Neighbor (KNN)

K-Nearest Neighbor (k-NN) is a method for classifying objects based on closest training examples in the feature space. It is simple but has been demonstrated effective for many classification tasks [22]. For a given k, KNN ranks the neighbors of a test vector X among the training sample, and uses the class labels of the k most nearest neighbors to predict the class of the test vector. Euclidean distance and Cosine distance are usually used for measuring the similarity between two vectors. The Euclidean distance measure and cosine distance measure are respectively defined as follows:

$$dis_{eu}(X,T_j) = \left\| X - T_j \right\| = \sqrt{\sum_{i=1}^{M} (x_i - t_{ij})^2} \tag{5}$$

$$dis_{\cos}(X,T_j) = \frac{X'T_j}{\left\|X\right\|\left\|T_j\right\|} = \frac{\sum_{i=1}^{M} x_i \times t_{ij}}{\sqrt{\sum_{i=1}^{M} x_i^2}\sqrt{\sum_{i=1}^{M} t_{ij}^2}} \tag{6}$$

where x_i is the i-th variable in the test vector X; T_j is the sequence j in the training data set and t_{ij} is the i-th variable in sequences T_j.

In anomaly detection, each sequence of the

observation data set is first transformed into a data vector respectively based on the plain *TF* or various cross frequency weight schemes defined in Equations (1-4). Suppose there are *M* distinct events in total in the observation data set, each sequence can then be expressed as a vector with *M* dimensions. The distance between a new test vector *X* and each vector in the training data set is calculated by using Euclidean distance and Cosine distance defined in Equation (5-6). The distance scores are sorted and the *k* nearest neighbors are chosen to determine whether the test vector is normal or not. In anomaly detection, we average the *k* closest distance scores as the *anomaly index*. If the *anomaly index* of a test sequence vector is above a threshold ε, the test sequence is then classified as abnormal. Otherwise it is considered as normal.

2.2.2. Nearest Neighbor (NN)

Nearest Neighbor (NN) is a slight modification of *k* Nearest Neighbor (KNN) presented in previous Section 2.2.1., when *k*=1. NN is simpler than KNN but usually is as effective as KNN for some classification tasks. Similarly, the closest distance between a test vector *X* and each vector in the training set is found and used as *anomaly index* for anomaly detection. The test vector is classified as abnormal if its *anomaly index* is above a pre-defined threshold ε.

2.2.3. Chi-square test

Chi-square distance test (also called as X^2 test) is a multivariate statistical technique. For a given test vector *X*, the X^2 test statistic is given by the equation:

$$X^2 = \sum_{i=1}^{M} \frac{(x_i - \bar{t}_i)^2}{\bar{t}_i} \qquad (7)$$

Where x_i is the *i*-th variable in the test vector *X* and $\bar{t}_i$ is the averaged *i*-th variable of all the training vectors. The distance of a test vector *X* from the center of the normal data population can be measured by X^2 test and are considered as *anomaly index* for the test vector. When the *M* variables are independent and *M* is large (e.g., greater than 30), the X^2 statistic follows approximately a normal distribution according to the central limit theorem [20, 23]. We compute the mean and standard deviation of the X^2 population as $\overline{X}^2$ and $\sigma_{\overline{X}^2}$ and set a threshold based on a zone of some combinations of $\overline{X}^2$ and $\sigma_{\overline{X}^2}$, e.g., $[\overline{X}^2 - \alpha\sigma_{\overline{X}^2}, \overline{X}^2 - \alpha\sigma_{\overline{X}^2}]$, where α is a variable parameter. For a test sequence *X*, if its *anomaly index* is outside of the zone, it is then classified as abnormal.

3. Experiments

3.1. Masquerade detection by profiling user behavior based on command data

3.1.1. Data set

The command data sets collected by Schonlau et al. [10-11] are used in our experiments for masquerade detection. The command data consists of user names and the associated command sequences (without arguments). 50 users are included with 15000 consecutive commands for each user, divided into 150 blocks of 100 commands. The first 50 blocks are uncontaminated and used as training data. Starting at block 51 and onward, some masquerading command blocks, randomly drawn from outside of the 50 users, are inserted into the command sequences of the 50 users. The goal is to correctly detect the masquerading blocks in the user community. The data used in the experiments are available for downloading at http://www.schonlau.net/intrusion.html.

3.1.2. Experiment results

In the experiments, we first convert each block of data into a feature vector based on the four weights. NN, KNN with Euclidean distance and Cosine distance as well as Chi-square test are then used respectively for masquerade detection. We use the same threshold for all the users for NN and KNN methods and use different thresholds for different users for Chi-square distance test based on the zone defined in Section 2.2.3. There is no updating during the training and detection steps in our experiments. Receiver Operating Characteristic (ROC) curves are used to evaluate the masquerade detection performance based on different distances with various frequency weights. The ROC curve is the plot of Detection Rates (DR), calculated as the percentage of masquerades detected, against False Alarm Rates (FAR), calculated as the percentage of normal blocks falsely classified as masquerades. There is a tradeoff between the DR and FAR and the ROC curve is obtained by setting different thresholds. Points nearer to the upper left corner of the ROC curve are the most desirable, as they indicate high DR with correspondingly low FAR.

For evaluating the performance of different weights, we plot ROC curves of the results shown in Fig.1 base on NN method by using Euclidean distance measure with four

weights. It is easily observed from the figure that *LOGtfidf* and *Mtfidf* are much better than *TF* and *Ltfidf* in terms of detection accuracy. In details, *LOGtfidf* is better than *Mtfidf* and *TF* is better than *Ltfidf*. We also plot the ROC curves based on the results of KNN (k=10) method by using Cosine distance measure with four weights in Fig. 2 and the results are consistent with those of Fig. 1.

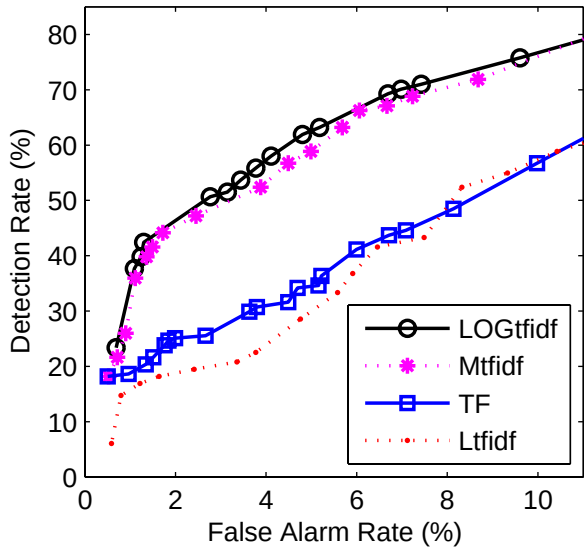

Fig. 1. ROC curves for the NN method by using Euclidean distance with four different frequency weights.

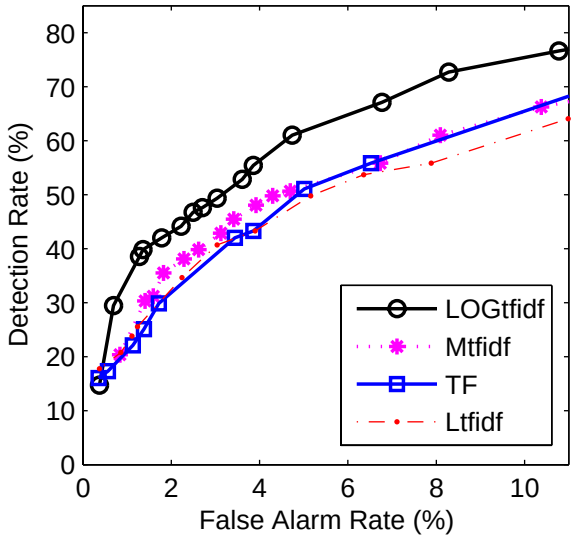

Fig. 2. ROC curves for the KNN method (k=10) by using Cosine distance measure with four different frequency weights.

For comparing the detection performance of different distance measures, we also plot ROC curves shown in Fig. 3 for the five distance measures using the *LOGtfidf* as it has been demonstrated as the most effective weight.

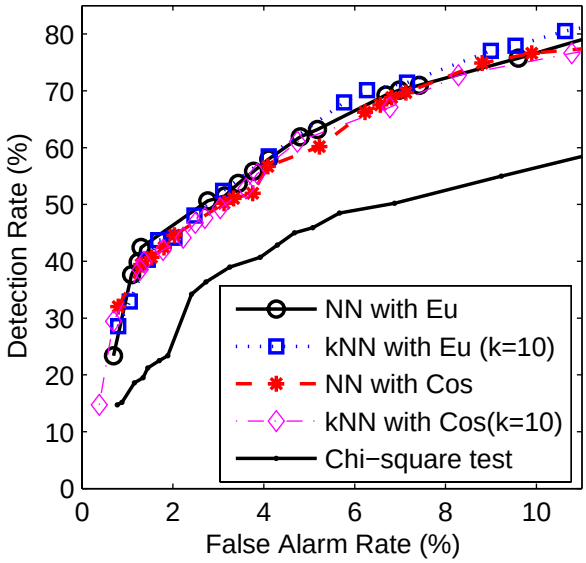

Fig. 3. ROC curves for different distance measures using the *LOGtfidf* weight.

From Fig. 3, it is observed that the NN and KNN methods outperform Chi-square test method for masquerade detection. The Euclidean distances are a little better than Cosine distance in terms of detection accuracy. Fig. 3 also indicates that the simple NN method with Euclidean distance measure can give a good performance for the detection.

Based on the same data set, Schonlau et al. [10-11] has used Bayes one-step Markov, Hybrid multi-step Markov, IPAM, Sequence-Match, Compression and Uniqueness for masquerade detection. We have also used NMF for masquerade detection based on the plain *TF* weight of the same data [16]. Fig. 4 and Tab. 2 show the results for the NN method with Euclidean distance by using the *LOGtfidf* weight along with the results from another 7 methods in [10-11, 16]. It is observed that based on the *LOGtfidf* weight, even the simple NN method achieves the best results among the other 7 methods. By using the same NN method, the *LOGtifidf* weight improves the detection results with 27.9% than plain frequency *TF* and improves with 30.8% than *Ltfidf* based on the same data set.

Table 2. The false alarm rate and missing alarm rate with comparison.

Method		False alarm (%)	Missing alarm (%)
Compression		5.0	65.8
Sequence-Match		3.7	63.2
IPAM		2.7	58.9
Hybrid multi-step Markov		3.2	50.7
Bayes one-step Markov		6.7	30.7
Uniqueness		1.4	60.6
NMF		1.9	57.5
NN with *LOGtfidf*	ε=1.38	1.3	57.5
	ε=0.67	6.6	30.0
NN with *TF*	ε=0.60	1.3	79.7
	ε=0.35	6.7	56.7
NN with *Ltfidf*	ε=1.30	6.6	58.4
	ε=1.88	1.3	83.1

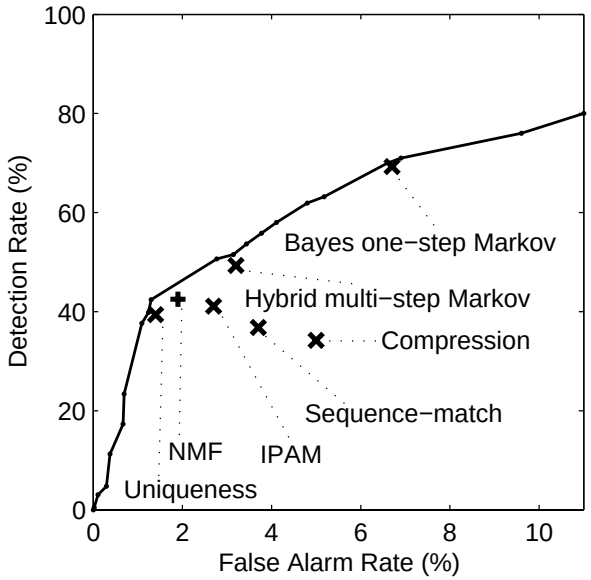

Fig. 4. ROC curves for the NN method using the *LOGtfidf* weight along with the results from other 7 methods.

3.2. Anomaly detection by profiling program behavior based on system call data

3.2.1. Data set

We used another data set to test the performance of the *LOGtfidf* weight with NN classifier as it has achieved good results on the command data for masquerade detection. The data set is *sendmail* system call sequences, collected in a UNIX-based host at UNM by Forrest et al. [5], since they were widely used for testing many other intrusion detection models. In the experiments, we used CERT synthetic *sendmail* data in which one sequence of normal data named "sendmail.int.gz" and 12 sequences of abnormal data including 4 syslog attacks and 2 unsuccessful intrusions (sm5x and sm565a) are used for testing. The data sets are available at http://www.cs.unm.edu/immsec/, and the procedures of generating the data are also described on the same website.

3.2.2. Experiment results

There are 147 normal traces and 34 abnormal traces of system calls in total in the data set. Each trace of the data corresponds a single process. In the experiments, all the system calls associated to the same process is grouped together. We then used *LOGtfidf* weight to convert each process into a vector. Intrusion detection is to identify whether the vector is normal or anomalous. 45 normal processes are randomly selected for training and the other 102 normal processes and 34 abnormal processes are used for testing. Fig. 5 shows the experiment results for the *sendmail* system call data based on NN method by using Euclidean distance with *LOGtfidf* weight.

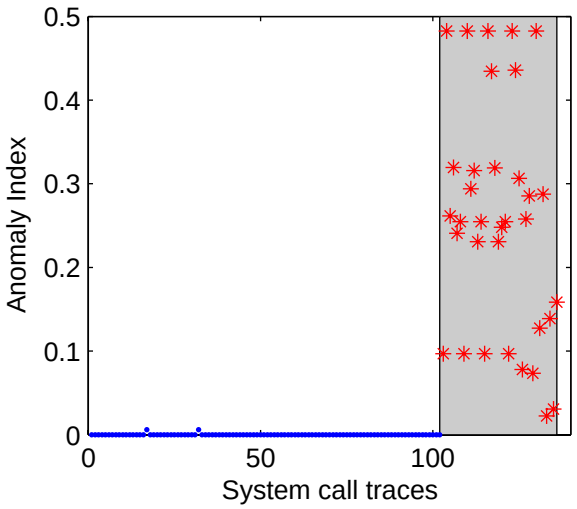

Fig. 5. Testing results for the NN method by using Euclidean distance with *LOGtfidf* weight based on *sendmail* system call data. The y-axis represents the *anomaly index* and x-axis represents system call trace (process) number. The stars (*) in the gray shading indicate attacks and dots (•) with no shading stand for normal data.

From Fig.5, it is seen that the abnormal data can be 100% distinguished from the normal data without any false alarms by using NN distance with *LOGtfidf* weight. This shows that the *LOGtfidf* have some robustness for anomaly intrusion detection.

4. Concluding remarks

Most current anomaly intrusion detection methods consider the transition or the frequency information of audit data. However, the methods considering the transition property of audit data needs high computational expense and this may not be suitable for real-time detection. The methods taking account of the frequency attributes of audit data have the capability of processing massive data for real-time detection but have to sacrifice some detection accuracy [18, 20] and this may reduce the effectiveness of an IDS. In this paper, we propose novel anomaly detection methods based on several frequency weights, e.g., the *LOGtfidf* and *Mtfidf* weights, which not only consider the frequency of each event in its sequence, but also takes account of how important the event is to the whole data set. The importance increases proportionally to the number of times an event appears in a sequence but is offset by the frequency of the event in the whole data set. For example, if an event appears in a sequence with a high frequency but seldom appears in the whole data, the *LOGtfidf* and *Mtfidf* score of the event become bigger and this helps a lot for detection of abnormal sequence of audit data. As these kinds of weight consider the frequencies of each event in its sequence and also in the whole data set, we may call these weights as cross frequency weights. Using the cross frequency weights are essentially more effective than only using the plain frequency attributes of audit data for anomaly intrusion. In addition, the computation cost of the cross frequency weights is almost as the same as the computation expense of the frequency and is low overhead. In this way, by using the cross frequency weights, the detection accuracy can improve a lot while the computation expense almost does not increase so that an effective IDS can be developed for real-time detection.

Plain *TF* weight and several novel cross frequency weights, namely, *LOGtfidf*, *Mtfidf* and *Ltfidf* are proposed for feature transformation and the Nearest Neighbor (NN) and KNN with Euclidean distance and Cosine distance as well as Chi-square test are employed for anomaly intrusion detection. Command data from Schonlau et al. [10-11] are used for validating the weights and the different distance measures. Experiment results show that the *LOGtfidf* and *Mtfidf* weight are better than plain term frequency (*TF*) and *Ltfidf* in terms of detection accuracy. For detection algorithms, the Euclidean distance measure is better than Cosine distance measure while Chi-square test consistently returns the worst results. Based on the *LOGtfidf* weight, even the simple NN method can achieve the better results than the other 7 methods in [10-11] and in [16]. The *LOGtifidf* weight improves the detection results with 27.9% than plain *TF* and improves with 30.8% than *Ltfidf* based on the NN method. The *sendmail* system call data from UNM are used as well and results show that the simple NN method with *LOGtfidf* weight is also effective for detection of anomalous programs.

Research in progress is on finding more effective weights for extracting valuable features of audit data to increase the detection accuracy. The ways how to combine the frequency attributes with the transition information of audit data to achieve lower false alarm and missing alarm rates are also being investigated.

Acknowledgements

The authors thank Ms. Xiangliang Zhang, Laboratoire de Recherche en Informatique, Université Paris Sud for the valuable suggestions and discussions.

The research in this paper was supported by French Ministry of Research (CNRS ACI-SI), Dependable Anomaly Detection with Diagnosis (DADDi) project.

References

[1] W. Lee and D. Xiang, "Information-theoretic measures for anomaly detection". In Proceedings of the 2001 IEEE Symposium on Security and Privacy, pp. 130-143, May 2001.

[2] D. E. Denning, "An intrusion-detection model". IEEE Transactions on Software Engineering, vol. 13, no.2, pp. 222-232, 1987.

[3] D. Y. Yeung and Y. Ding, "Host-based intrusion detection using dynamic and static behavioral models". Pattern Recognition, vol.36, no.1, pp.229-243, 2003.

[4] S. Axelsson, "Intrusion detection systems: A survey and taxonomy". Technical Report 99-15, Chalmers Univ., March 2000.

[5] S. Forrest, S. A. Hofmeyr, A. Somayaji, and T. A. Longstaff, "A sense of self for Unix processes". In Proceedings of the 1996 IEEE Symposium on Research in Security and Privacy, pp. 120-128. IEEE Computer Society Press, 1996.

[6] W. Lee and S. Stolfo, "Data mining approaches for intrusion detection". In Proceedings of the 7th

USENIX Security Symposium, pp. 79-94, San Antonio, TX, Jan 1998.

[7] C. Warrender, S. Forrest, and B. Pearlmutter., "Detecting intrusions using system calls: Alternative data models". In Proceedings of 1999 IEEE Symposium on Security and Privacy, pp. 133-145, 1999.

[8] S. B. Cho and H. J. Park, "Efficient anomaly detection by modeling privilege flows using hidden markov model". Computers & Security, vol.22, no.1, pp.45-55, 2003.

[9] W. Wang, X. Guan, and X. Zhang, "Modeling program behaviors by hidden Markov models for intrusion detection". In Proceedings of the Third International Conference on Machine Learning and Cybernetics (ICMLC'2004), pp. 2830-2835, 2004.

[10] M. Schonlau and M. Theus, "Detecting masquerades in intrusion detection based on unpopular commands". Information Processing Letters, 76, pp. 33-38, 2000.

[11] M. Schonlau, W. Dumouchel, W.-H, Ju, A. F. Karr, M. Theus and Y. Vardi, "Computer Intrusion: Detecting Masquerades". Statistical Science, vol.16, no.1, pp.58-74, 2001.

[12] Y. Liao and V. R. Vemuri, "Using text categorization techniques for intrusion detection". In 11th USENIX Security Symposium, pp. 51-59, 2002.

[13] W. Hu, Y. Liao, and V.R. Vemuri, "Robust support vector machines for anomaly detection in computer security". In Proceeding of the 2003 International Conference on Machine Learning and Applications, 2003.

[14] Z. Zhang and H. Shen, "Application of online-training svms for real-time intrusion detection with different considerations". Computer Communications, vol.28, pp. 1428-1442, 2005.

[15] W. H. Chen, S. H. Hsu, and H. P. Shen, "Application of svm and ann for intrusion detection". Computers & Operations Research, vol.32, pp. 2617-2634, 2005.

[16] W. Wang, X. Guan, and X. Zhang, "Profiling program and user behaviors for anomaly intrusion detection based on non-negative matrix factorization". In Proceedings of 43rd IEEE Conference on Decision and Control (CDC'04), pp. 99-104, 2004.

[17] W. Wang, X. Guan, and X. Zhang, "A novel intrusion detection method based on principal component analysis in computer security". In Advances in Neural Networks-ISNN2004, International IEEE Symposium on Neural Networks, volume 3174 of Lecture Notes in Computer Science (LNCS), pp. 657–662, Dalian, China, August 2004.

[18] Y. Bouzida and S. Gombault, "Intrusion Detection Using Principal Component Analysis". In proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics, Orlando, Florida, 2003.

[19] W. Wang, X. Guan, X. Zhang, and L. Yang, "Profiling program behavior for anomaly intrusion detection based on the transition and frequency property of computer audit data". Computers & Security, Elsevier, vol. 25, no 7, pp. 539-550, 2006.

[20] N. Ye, X. Li, Q. Chen, S. M. Emran, and M. Xu, "Probabilistic techniques for intrusion detection based on computer audit data". IEEE Transactions on Systems, Man, and Cybernetics, vol.31, no.4, pp.266-274, 2001.

[21] B. Tang, M. Shepherd, E. Milios, and M.I. Heywood, "Comparing and combining dimension reduction techniques for efficient text clustering". International Workshop on Feature Selection for Data Mining - Interfacing Machine Learning and Statistics in conjunction with 2005 SIAM International Conference on Data Mining, Newport Beach, California, 2005.

[22] R. O. Duda, P. E. Hart, and D. G. Stork. Pattern Classification. China Machine Press, Beijing, 2nd edition edition, 2004.

[23] R. A. Johnson and D. W. Wichern, "Applied Multivariate Statistical Analysis". Prentice Hall, 2002.