Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

HIDS:DC-ADT : An Effective Hybrid Intrusion Detection System based on Data Correlation and Adaboost based Decision Tree classifier

Ali Raeeyat [1], Hedieh Sajedi [2]

1 Department of Computer Engineering, Islamic Azad University, Dezfoul Branch, Dezfoul, Iran

2Assistant Professor of Computer Science, Tehran University, Tehran, Iran

Abstract
Due to the rapid development of computer networks, intrusions and attacks into these networks have grown, and occur in various ways. Thus, usually an intrusion detection system can play an important role in security protection and intruders' accessibility to network prevention. In this paper, a new hybrid approach, which is called HIDS:DC-ADT, is used to design proposed detection engine. In the proposed intrusion detection system, the anomaly detection engine is responsible to detect new and unknown attacks and the misuse detection engine is responsible to protect anomaly detection system. Through this, it is assured that collected data and patterns be safe for anomaly detection system. In the intrusion anomaly detection using statistical correlation method that is of data correlation methods, normal behavior of network is analyzed statistically by KDD-Cup99 data-set. Further, the Data Correlation Graph (DCG) has been proposed to show behaviors deviation of normal behavior. In misuse detection, Principal Components Analysis (PCA) is used to dimensionality reduction. More, a new classification method by Adaboost algorithm using base classifier of decision tree C4.5 has been introduced for classification. Simulation results show that this hybrid system can reach a competitive accuracy and efficiency.

Keywords: HybridIntrusion Detection System, Data Correlation, Data Correlation Graph, Adaboost Algorithm, Decision Tree, Principle Component Analysis.

I. Introduction

With the fast development and popularity of Internet, the security of networks has been a focus in the current research. Nowadays, much attention has been paid to intrusion detection system (IDS) which is closely linked to the safe use of network services. There are mainly two types of intrusion detection systems namely anomaly detection and misuse detection. Anomaly detection system builds normal behavior profile for users and system actions, monitors the deviation of current event with respect to the recognized profile. This approach doesn't depend on the characteristics of attacks. However, it needs a large set of training data from system event log to build normal behavior profile and usually signals many false alarms (FA). Misuse detection system detects intrusions by matching system behaviors with known attacks, of which the behavioral features are exhaustively studied and well-defined. The wholeness of known attacks determines the efficiency of

this method. Since it can only detect attacks known earlier, the systems must be updated with newly discovered attack signatures [1]. To improve the performance of IDS, we propose a hybrid intrusion detection system (HIDS), which uses both methods. In the proposed intrusion detection system, the anomaly detection engine is responsible to detect new and unknown attacks and the misuse detection engine is responsible to protect anomaly detection system. Through this, it is assured that collected data and patterns be safe for anomaly detection system.

Intrusion detection systems have to collect and relate alert information from different sources to spot complete attack scenarios. The process of collecting and relating alert information is called alert correlation. Recently, alert correlation gained momentum and a number of academic and commercial correlation approaches have been suggested. However, there is no consensus on what this process is or how it should be implemented or evaluated [5].

Some systems use distinctive IDSs and then correlate the results and the similar alarms. This method aims at attaining higher-level descriptions of attacks or a more condensed view of the security issues highlighted during the analysis without losing security-relevant information[7][6]. Alarm correlation based IDSs only determine the relation and correlation between alarms, produced by IDS' sensors, but there are some other systems that focus on alert correlation. Some of these systems, presented in [16], do not use independent IDSs. For example in [16] some correlated alert create a new Meta-Alert to achieve higher-level descriptions of attacks.

In this paper, for construct anomaly detection engine our method is data correlation. Data correlation means associating sets of events acknowledged through different means and applying knowledge to conclude whether they are related, and if so, in what manner and to what degree. As the quantity of correlation between two features is enlarged, the relation between features is more justifiable. We calculate the correlation between features in the normal traffic with statistical methods. If this value is larger than a defined threshold value, the correlated feature pairs are considered to be comprised in a correlation relation graph. This method reduces processing load of anomaly detection engine and the set of features that are required for intrusion detection.

Also, with the ever-increasing network traffic and variation of intrusions, data mining technologies have been introduced to IDSs. Kayacik In [8] proposed a hierarchical Self Organized Map (SOM) for intrusion detection. They utilized the classification capability of the SOM on selected dimensions and specific attention was given to the hierarchical development of abstractions. The reported results showed that there was an increase in attack detection rate. Yongiin Liu et al. [15] have created a classifier by using a decision tree as its base classifier. The classification accuracy of this algorithm was little improved than SOM algorithms. Weiming Hu et al. [13] have proposed an Adaboost based algorithm for network intrusion detection system which used decision stump as a weak classifier. The decision rules are provided for both categorical and continuous features and some provision was made for handling the overfitting. The key difference between our proposed work and that of Weiming Hu et al. [14] is that they have used decision stump as a weak learner, while we use Decision Tree as weak classifier.

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

Decision tree, among others, makes simple and effective predictive models by training a large set of sample data, therefore provides more accurate detection results. Decision tree can be used to optimize detection rules of present IDSs, hence reduce the workload of manual analysis of intrusions.

In this paper, an Adaboost algorithm for misuse intrusion detection system with decision tree as weak classifier is proposed. A benchmark dataset is used in this experiment to prove that boosting algorithm can greatly improve the classification accuracy of weak classification algorithms. Also, before we use Principal Components Analysis (PCA) to dimensionality reduction. Principal Component Analysis is method used for feature extraction, data used in intrusion detection issue are high dimensional in nature. It is desirable to reduce the dimensionality of the data for easy examination and further analysis.

II. Materials and Methods

A. Data Set

The experimental data used in this paper is a benchmark database downloaded from KDD-Cup99 (http://kdd.ics.uci.edu/databases/kddcup99). This database contains a standard set of network visit data, which contains a wide diversity of intrusion simulation in the US military network environment. KDDcup99 data consist of two data sets, which are the full data set (18 M, 743 M Uncompressed) and the 10% subset (2.1 M, 75 M Uncompressed). The latter is chosen to be the experimental data set as our object. Each data consists of 41 features.

The classification of the attack behavior is a 5-class problem, and each network visit belongs to one of the following behavior: normal, denial of service (DOS), unauthorized access from a remote machine (R2L), unauthorized access to local supervisor privileges (U2R), probing, surveillance and other probing.

B. Data Preprocessing

One notes that the redundancy in the KDD99 data set is incredibly high. Observably, such a high redundancy certainly influences the use of data. By deleting the repeated data, the size of data set is reduced from 494,021 to 145,586.

In the other hand, there is a problem of symbolic attributes like name, protocol, service, and a flag. The correlation process is completely meaningless on nominal features and then these features will not participate in the correlation process. Therefore, features are listed, convert to numeric attributes through conversion table.

C. Data Correlation

All the devices, whether designed at prevention or detection, produce enormous volumes of audit data. Firewalls and other devices logging network connection

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

information are especially culpable of generating vast masses of data. Many miscellaneous data formats are used for those log files and audit trails. Also, a percentage of events produced by network IDS and IPS are false alarms and do not map to actual threats. Additional problem is that thedifferent devices might report on the same things happening on the network, but in a different way, with no obvious way of figuring the truth of their relationship. There is a definite need for a consistent analysis framework to identify diverse threats, order them and learn their influence on the target system.

Correlation is defined as relationships between entities. Data correlation may be defined to enhance the threat identification and the assessment process by looking not only at individual data's, but also at their sets. Chuvakin in [3] generally categorized correlation as rule-based or statistical.

1) Statistical correlation

A rule-based correlation engine has some prior knowledge of the attack, and it is capable to define what is actually detected in exact terms, based on that. Statistical correlation does not employ any prior knowledge of the malevolent activity, but instead relies upon the knowledge of normal activities, which has been gathered over time. Ongoing events are then rated by a built-in algorithm and may also be compared to the collected activity patterns, to discriminate normal from suspicious behavior.

In this paper, we propose extra data correlation method that calculates the correlation value between features with statistical analysis of a normal behavior. This approach of data correlation processes the collected statistical samples of features from normal data instances during train phase.

A feature may fluctuate in different observations. We can find correlation value between two features with different pairs of observation. Calculation of correlation value between such features gives us the extent of relation between them.

In random samples of statistical population, n observation of X and Y variables are represented by (X_i, Y_i) pairs, for $i = 1, 2, \dots, n$. These pairs have equal bi-variable distribution and different pairs are independent of each other. A simple relation between $X$and Ycreates some points around the straight regression line. We use Pearson Correlation Coefficient to define the value of correlation between two features:

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{[\sum_{i=1}^{n}(X_i - \bar{X})^2][\sum_{i=1}^{n}(Y_i - \bar{Y})^2]}} \quad (1)$$

In which a (X_i, Y_i) pair is an observations of X and Y random variables. $\bar{X}, \bar{Y}$are mean values for X and Y respectively.$r$has a value in $[-1, 1]$. While $r = 1$ means that all points of (X_i, Y_i) pairs are on a straight line with a positive slope.$r = -1$means that all points of (X_i, Y_i)pairs are on a straight line with a negative slope. When r approaches from these two values to zero, the degree of correlation decreases; such that there is no correlation in zero point [3].

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

D. Data Correlation Graph

We inspected features that are correlated such that the value of their correlation coefficient is more than threshold level. Hence, the features that haven't this requirement are not considered for examination. We selectonly 97278 of normaldata instances of KDD-Cup99 dataset for our statistical analysis. We calculatedthe correlation coefficient between these features,and finally introduced some of them as optimumfeatures for anomaly intrusion detection.

Features that have necessary correlation value and participate in graphs are effective features in our anomaly intrusion detection system. The first step for building graphs is to create correlation matrix of features. Each entry of this matrix is the correlation coefficient between two features that are calculated using (1).For example, $DCG_{i\times j}$ represents thecorrelation value between i and j that theformerfeature is in the i-th row and the latter one is in thej-th column.

We selected the entrances that are greater than our defined threshold. Since the correlation coefficient is a value in $[-1, 1]$and $0 \leq |r| \leq 1$, we intuitively considered 0.5 as the suitable threshold.

DCG is a graph for modeling a set of features that make an equivalence class under the data correlation-relation. Actually, this graph is just a way to illustrate the correlated features and analyzing the data correlation-relations. In [2], the algorithm of constructing the DCG is described.

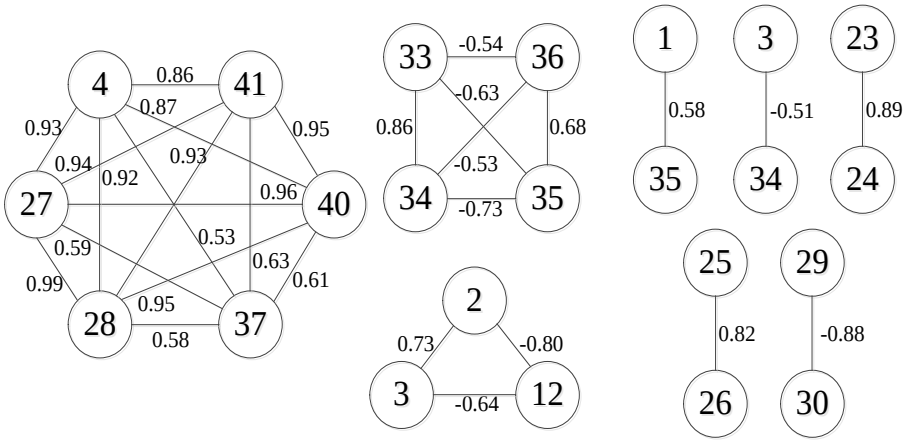


Figure. 1.Data Correlation Graph of Normal Data Instances.

Figure 1 shows DCG of normal data instances. These graphs have been created based on correlation matrix of 41 features examined in normal data instances. A correlation matrix of each behavior is a $n \times n$matrix, which the entrance that lies in the i-th row andthe j-th column of a matrix is the correlation coefficient between features i and j. As we can see, only 20 features have been contributed in these graphs. Nodes of DCG are theindices of features defined in KDD and the correlation value of two features has been

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

shown beside the corresponding edge between them. Table 1 shows what feature each number refers to.

If a DCG has more nodes it will be more useful for detection engine, as it checks further binary correlation relations between features and increases the percentage of detection. For example, the graph which has 6 nodes is more useful than the other ones because of the number of their members. Two-member graphs are not as useful as three, four-or-six-member graphs are for detection engine because they just examine one binary correlation relation between two features.

TABLE I. NAME OF CORRELATED FEATURES OF NORMAL'SDCG

Feature Name	Feature Number
Duration	1
Protocol_type	2
Service	3
Flag	4
Logged_in	12
Count	23
Srv_Count	24
Serror_Rate	25
Srv_Serror_Rate	26
Rerror_Rate	27
Srv_Rerror_Rate	28
Same_Srv_Rate	29
Diff_Srv_Rate	30
Dst_Host_Srv_Count	33
Dst_Host_Same_Srv_Rate	34
Dst_Host_Diff_Srv_Rate	35
Dst_Host_Same_Src_Port_Rate	36
Dst_Host_Srv_diff_Host_Rate	37
Dst_Host_Rerror_Rate	40
Dst_Host_Srv_Rerror_Rate	41

E. Principle Component Analysis

Principal Component Analysis (PCA) is a technique for dimensionality reduction and multivariate analysis [9]. Its applications contain data compression, image processing, visualization, exploratory data analysis, pattern recognition, and time series prediction. PCA popularity is derived from three properties. To begin with, it is an optimal linear scheme for compressing high dimensional vectors into lower dimensional vectors and later reconstructing the original set. Secondly, the model parameters are directly computed from data - by diagonalizing the sample covariance matrix. Finally, compression and decompression are easy to accomplish with the given model parameters - they need matrix multiplication alone. A multi-dimensional hyper-space is typically hard to visualize. The purpose of unsupervised learning approaches is reduced dimensionality, scoring observations on a composite index and clustering similar

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

multivariate attribute observations. Multivariate attributes can be summarized by two or three variables which are graphically displayed with minimum information loss and are so useful in knowledge discovery. As visualization of multi-dimensional space is difficult, PCA is used to reduce dimensionality of *d* multivariate attributes into two or three dimensions. PCA summarizes variations in correlated multivariate attributes to non-correlated components, each being of a particular linear combination of original variables. Consequently extracted non-correlated components are known as Principal Components (PC) and they are estimated from the original variables eigenvectors. Therefore PCA objective is attainment of parsimony and reduction in dimensionality through extraction of the smallest number components that lead to the most variation in original multivariate data. And this data should also be summarized with little information loss. In PCA, PC extractions can be made through original multivariate data set or by using a covariance matrix when the original data set is unavailable. In deriving PC, the correlation matrix instead of the covariance matrix might be used especially when differing dataset variables are measured with differing units or if differing variables have different variances. Use of a correlation matrix is equal to standardizing variables to zero mean and unit standard deviation.

The PCA model can be represented by:$u_{mx\,1} = W_{mxd}\, x_{dx\,1}$
Where *u*, an *m*-dimensional vector, is a projection of *x* - the original *d*-dimensional data vector (*m* <<*d*).

PCA technique is applied to the KDD-Cup99 dataset with variance covered 0.95 and maximum attribute name 5. Consequently 19 features extracted out of 41 features as shownin Table 2.

TABLE II. FEATURE EXTRACTED BY PCA TECHNIQUE

No	Feature
1	-0.292dst_host_same_srv_rate-0.292dst_host_srv_count-0.288same_srv_rate+0.287service-0.269flag
2	0.395srv_rerror_rate+0.395rerror_rate+0.395dst_host_srv_rerror_rate+0.395dst_host_rerror_rate-0.261dst_host_srv_serror_rate
3	0.467logged_in-0.39dst_host_count-0.356count + 0.262srv_diff_host_rate-0.256srv_count
4	0.54 num_compromised +0.539num_root +0.475su_attempted + 0.317num_access_files+0.239root_shell
5	-0.687is_guest_login-0.685hot+0.099dst_host_srv_diff_host_rate-0.097duration-0.096dst_host_diff_srv_rate
6	-0.619dst_host_diff_srv_rate-0.563duration-0.345diff_srv_rate-0.139flag+0.128hot
7	0.477num_shells+0.444num_failed_logins+0.425urgent+0.409root_shell+0.377num_file_creations
8	-0.537num_failed_logins+0.479num_shells-0.457urgent + 0.346 num_file_creations+0.263root_shell
9	0.633dst_host_srv_diff_host_rate+0.54 dos + 0.249srv_diff_host_ rate-0.24logged_in-0.191dst_host_count
10	0.953wrong_fragment-0.159duration-0.122num_file_creations + 0.103num_shells-0.09src_bytes
11	0.994src_bytes+0.083wrong_fragment-0.039num_file_creations + 0.031dst_bytes+0.027num_shells
12	-0.852dst_bytes +0.373urgent +0.226num_file_creations + 0.162 num_access_files-0.13num_shells
13	-0.702num_file_creations+0.429num_shells-0.307dst_bytes + 0.248diff_srv_rate+0.201root_shell
14	0.82 DoS-0.406dst_host_srv_diff_host_rate+0.167logged_in+0.16 dst_host_count-0.149srv_diff_host_rate
15	0.524num_failed_logins-0.515urgent-0.388diff_srv_rate + 0.366 duration-0.241dst_bytes
16	0.532diff_srv_rate-0.47duration+0.414num_failed_logins-0.409 urgent+0.275num_file_creations
17	-0.68root_shell+0.512num_shells+0.46 num_access_files + 0.15num_failed_logins+0.109srv_diff_host_rate
18	0.869srv_diff_host_rate-0.336dst_host_srv_diff_host_rate + 0.217dst_host_count-0.147diff_srv_rate+0.119root_shell
19	0.74 num_access_files+0.398root_shell-0.28num_file_creations-0.224num_compromised-0.222num_root

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

F. Adaboost Algorith

G.

AdaBoost is a machine learning algorithm, can be used in aggregation with many other learning algorithms to improve their performance [15]. It calls a weak classifier repetitively in a series of rounds. The pseudo code of our Adaboost algorithm is given in Figure 2.

Input: Sequence of m training examples
Let the set of training sample data be $\{(x_1, y_1), \ldots, (x_i, y_i), \ldots, (x_m, y_m)\}$with labels y_i {Normal, Dos, Probe, R2L, U2R}, where x_i denotes i^{th} feature vector and m is the size of the dataset. Let T be the number of iterations.
Initialize the weights $D_t(i) = 1/m$ for all i.
Repeat for $t = 1, 2, \ldots, T$ the following steps

(1) Call the weak classifier, and provide it with the instances of distribution D_i

(2) Calculate the error rate $\varepsilon_c$for each category of attacks on each round of the hypothesis

$$\varepsilon_c = Pr_{i\sim Dt}[h_t(x_i) \neq y_i] = \sum_{h_i(x_i)\neq y_i} D_t(i) \qquad (2)$$

If ε_c>0.5, then set T=t-1 and abort loop. Here $\varepsilon_c$is the error rate for each category of attacks.

(3) Calculate the reweight value for each category of attack instances by using the equation,

$$\beta c = \varepsilon_c / 1 - \varepsilon_c \qquad (3)$$

(4) Update distribution $D_t$for each category of attacks:

$$D_{t+1}(i) = \begin{cases} \beta c \dfrac{D_t(i)}{Z_t} & (h_t(x_i) = y_i) \\ \dfrac{D_t(i)}{Z_t}(else) \end{cases} \qquad (4)$$

Where $Z_t$is normalization constant.

(5) Repeat the steps from (2) to (4) for all category of attacks with multiple combination of weak classifiers

Output: final hypothesis

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

$$h_{fin}(x) = \underset{y \in Y}{\arg max} \sum_{ht(x)=y} log\frac{1}{\beta c} \qquad (5)$$

Let us write the error $\varepsilon_i$of $h_t$as$1/2 - y_t$. Then $y_t$shows how much better of weak learner than random guessing. in [10] have proven that the training error $\varepsilon_c$of the final hypothesis is at most

$$\varepsilon_c = \prod\left[2\sqrt{\varepsilon t(1-\varepsilon t)}\right] = \prod t\sqrt{1-4\gamma 2} \leq \exp(-2\sum_t \gamma 2) \qquad (6)$$

From above equation (6), we can conclude that the training error of boosting algorithm drops exponentially fast.

Figure. 2.Adaboost Algorithm.

H. Decision Tree

A decision tree offers a decision procedure to determine the class of a given instance. In the massive area about decision trees, also known as classification trees or hierarchical classifiers, at least two pivotal works are to be mentioned, those by Quinlan [12] and those by Breiman [4]. The first synthesizes the experience gained by people working in the area of machine learning and describes a computer program called ID3, which has developed in a new system, namedC4.5 [11].A decision tree is a tree that has three main components: nodes, arcs, and leaves. Each node is labeled with a feature attribute which is most informative amongst the attributes not yet considered in the path from the root, each arc out of a node is labeled with a feature value for the node's feature and each leaf is labeled with a category or class.

We use the C4.5 algorithm [11] to construct the decision trees where Shanon Entropy is used to measure how informative is a node. The selection of the finest attribute node is based on the gain ratio *GainRatio(S, A)* where *S* is a set of records and *A* a non-categorical attribute. This gain describes the expected reduction in entropy due to sorting on A. It is calculated as the following [16]:

$$Gain(S,A) = Entropy(S) - \sum_{v \in Values\ (A)} \frac{|(S_v|}{|S|} Entropy(S_v) \qquad (7)$$

In general, if we are given a probability distribution $P = (P_1, P_2, \ldots, P_n)$ then the information conveyed by this distribution, which is called the Entropy of P is:

$$Entropy(P) = -\sum_{i=1}^{n} P_1 log_2 P_1 \qquad (8)$$

If we consider only $\mathrm{Gain(S, A)}$then an attribute with many values will be automatically selected. One solution is to use GainRatio instead [17]

$$GainRatio(S,A) = \frac{Gain(S,A)}{SplitInformation(S,A)} \qquad (9)$$

Where

$$SplitInformation(S,A) = -\sum_{i=1}^{c} \frac{|S_1|}{|S|} log_2 \frac{|S_1|}{|S|} \qquad (10)$$

Where $S_i$is a subset of *S* for which *A* has a value v_i.

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

III. The Proposed Hybrid Intrusion Detection System

We now present the HIDS:DC-ADT as shown in Figure 3.The model comprises of 4 seminalmodules: (1) Data preprocessing. Data streams need to be preprocessed to meet the input requirements of data correlation and data mining algorithms.(2) Misuse detection. Preprocessed data is sent to PCA unit to extract features. Then the data analyzed using the Adaboost algorithm based C4.5 as a classifier to decide if the data instance is an intrusion. The result is sent to the 'Evaluation and comparison module'. (3) Anomaly detection. Simultaneously pre processed data is sent to data correlation unit to calculate the correlation between features and select the correlate and informative features. Then the DCG model used for show behaviors deviation of normal behavior is built. The result of this module also sent to the 'Evaluation and comparison module'. (4) Evaluation and comparison module. To determine whether a data instances an intrusion. An instances recognized as an intrusion if and only if both detection methods decide it is an intrusion.

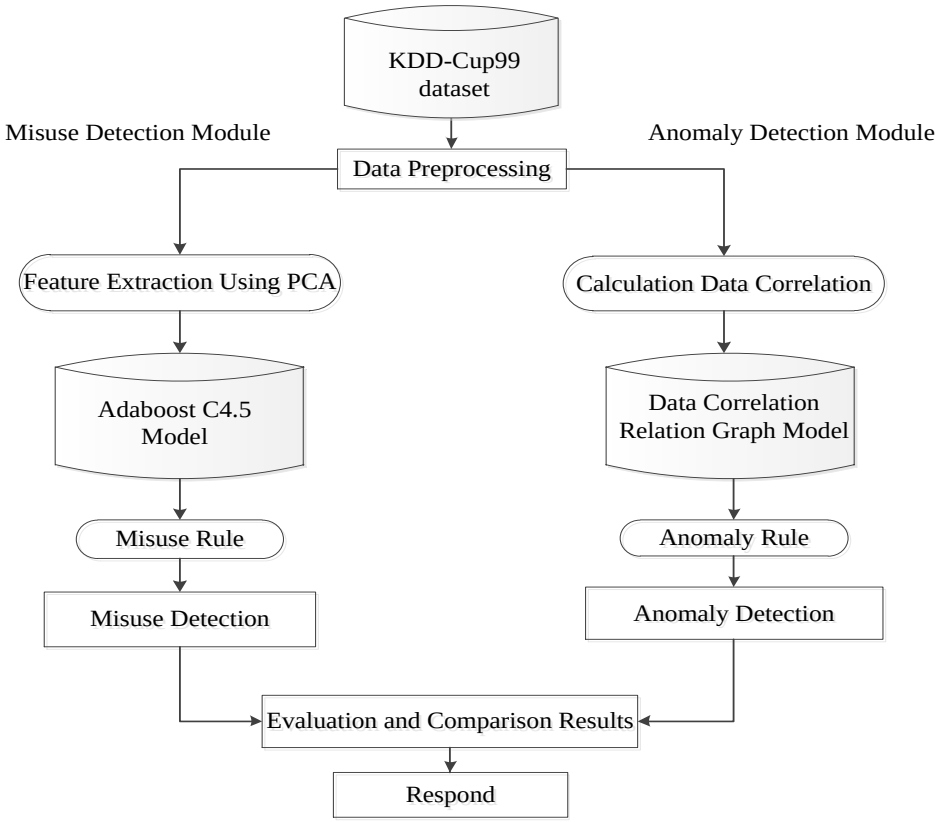


Figure. 3.The Proposed HIDS:DC-ADT Model.

IV. Results and Analysis

A. Anomaly Detection

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

Now we'll introduce our anomaly detection engine, and the results will be represented. Features of all normal data instances created 8DCGs such that one of them had 6 nodes, another's had 4 and 3 nodes and the others had only 2 nodes. A detection engine that employs these features should be based on the correlation relation and the deviation from them. Correlation relation has a much influence on the regression line of two statistical features. As the correlation value between two features increases, scattering of points around the regression line of them will be decreased. For correlation values near 1 or -1, we can say that they are over regression line completely.

We also consider a confidence interval for every regression line. This interval determines the acceptable deviation for every regression equation. Every pair that has greater interval from regression line cannot justify this relation. There is only one regression equation and confidence interval for two-nodes DCGs that determine the relation between two features. Each regression equation between two features can be shown as$Y = aX + b$, in which $X$is independent variableand Y is dependent variable. For every X, thevalue of Y is always between two limitations; i.e. a aslower limit and b as upper limit with$P(a < y < b) = 95\%$.

The acceptable width of deviation of Y defined as$|a - b|$and confidence interval of each relation is half ofthis absolute value. In this paper we use the min intervals to achieve acceptable detection rate and false alarms, as we will show in Table 3. The severity of detection engine depends on this interval, as well as the number of false alarms.

In this paper, a data instances anomaly "if most of regression relations of DCG are rejected". First, we examined 494021 KDD data instances with only six-member DCG. Then these data instances evaluated with four and three member DCG. Table 3 shows the results of this approach. Due to the result of table 3 we construct anomaly detection engine for our HIDS with six-member graph.

TABLE III. ANOMALY DETECTION RESULT

Compared DCG of Normal Data Instances	DR	FA	Accuracy
Six-Member Graph	95.7%	1.3%	97.2%
Four-Member Graph	83.5%	5.9%	81.4%
Three-Member Graph	52.4%	4.1%	76.2%

B. Misuse Detection

Now training and test for misuse detection engine can bebegun. On the other hand, classifier is evaluated with 10-fold cross validation, which is a technique for estimating the performance of a classifier.

First, on the basis of PCA algorithm, misuse detection models are built using C4.5. Then we use Adaboost algorithm with C4.5 as a weak classifier. The overall accuracy of two classifiers is shown in figure 4. It can be seen that, the accuracy of PCA:DT

Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

combination with the Adaboost algorithm is comparatively better than the single PCA:DT classifier. The detection rate result is shown in Figure 5.

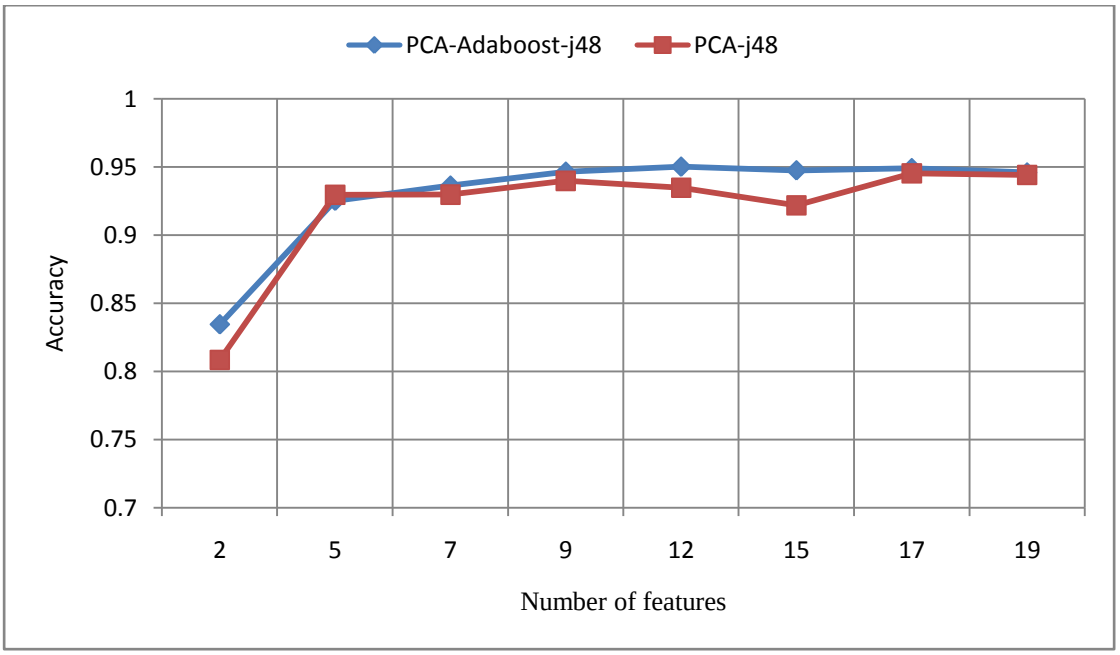
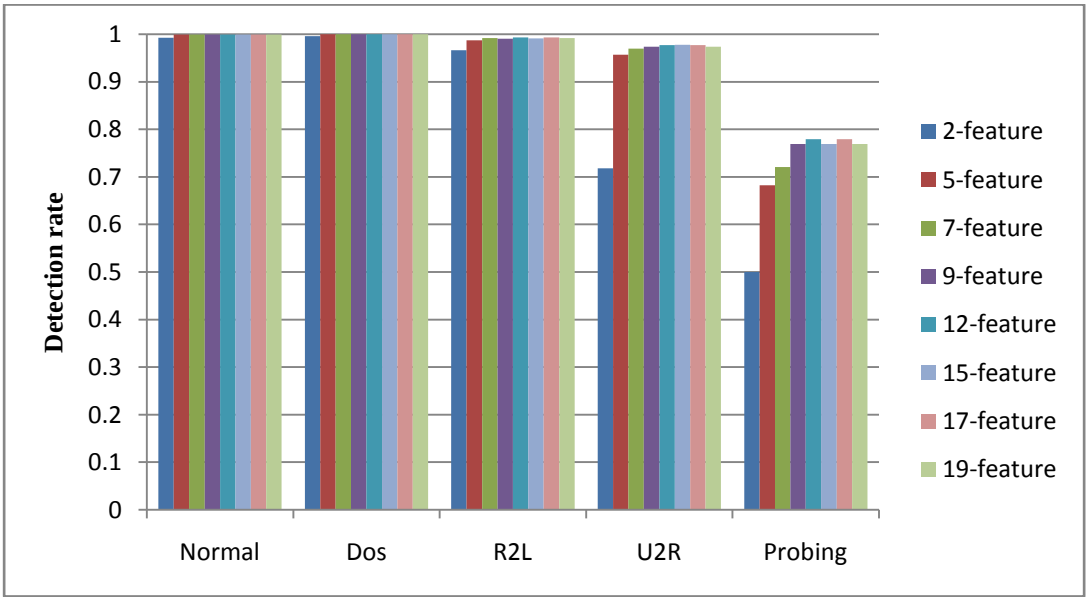


Figure. 4.Accuracy with chosen PCA features.

The experiment result shows that with 12-feature have the highest accuracy, detection rates with low false alarm show in table 5.We construct our misuse detection engine with this approach.



Journal of Academic and Applied Studies
Vol. 2(12) December 2012, pp. 19-33
Available online @ www.academians.org
Special Issue on Computer Science

ISSN1925-931X

Figure. 5.Detection rate with chosen PCA features using adaboost based C4.5 classifier.

TABLE IV.MISUSE DETECTION RESULT

Number of Features	DR	FA	Accuracy
12-feature	99.95%	0.04%	95.02%

C. Hybrid Intrusion Detection System

To compute the detection rate of our HIDS, we define false-positive rate and false-negative rate, which is given as follows:

- **Definition 1 (False-Positive Rate).** *The probability of detecting normal instances as intrusion ones is defined as false-positive rate* γ_p.
- **Definition 2 (False-Negative Rate).** *The probability of detecting intrusion instances as normal ones is defined as false-positive rate* γ_n.

From definition 1 and 2, there holds detection rate:

$$\gamma = 1 - \gamma_p - \gamma_n \qquad (11)$$

Therefore, the detection rate for the proposed hybrid intrusion detection system:

$$\gamma = 1 - 0.0134 - 0.0435 = 94.31\%$$

TABLE V
HIDS:DC-ADT Result

	DR	FA	Accuracy
Proposed HIDS:DC-ADT	94.31%	1.34%	96.48%

V. Conclusions

In this paper, we propose a hybrid intrusion detection system using both misuse and anomaly detection methods. In anomaly detection model, after data preprocessing, data is sent to data correlation unit to calculate the correlation between features and select the correlate and informative features. Then the DCG model used for shows behaviors deviation of normal behavior. Then linear regression with two features is calculated. We detected data instances as anomaly with calculating the deviation of each pairs from their linear regression equation. So if most of regression relations of DCG are rejected we conclude this data instance is anomaly .In misuse detection model, data is sent to PCA unit to extract features. Then the data analyzed using the Adaboost algorithm based C4.5 as a classifier to decide if the data instance is an intrusion. The result show that we achieve highest accuracy with 12 features extracted using PCA algorithm. Finally we show the result of proposed HIDS. Simulation results show that our proposed system provides a high detection rate and accuracy with low false-positive rate and false-negative rate.

References

[1] Ali Aydın, Halim Zaima, and Gökhan Ceylana, "A hybrid intrusion detection system design for computer network security", Computers & Electrical Engineering, May 2009, pp. 517-526.

[2] Amin Hassanzadeh, Babak Sadeghian, "Intrusion Detection with Data Correlation Relation Graph", Proceeding of IEEE conference in ARES 2008, Barcelona, Spain, March 3-5, 2008.

[3] Anton Chuvakin, "Event Correlation in Security",www.securitydocs.com, May 2004.

[4] BreimanL., FriedmanJ. H., OlshenR. A., and StoneC. J.. Classification and Regression Trees.1984.

[5] Christopher Kruegel, Fredrik Valeur, Govanni Vigna, "Intrusion Detection and Correlation: Challenges and Solutions", Springer (2005).

[6] Chyssler T., BurschkaS., "Alarm Reduction and Correlation in Intrusion Detection Systems", Proceedings of Detection of Intrusions and Malware & Vulnarability Assessment workshop (DIMVA), Gesellschaft f r Informatik, June (2004), pp. 9-24

[7] Chyssler T., Nadjm-TehraniS., "Alarm Reduction and Correlation in Defence of IP Networks", Proceedings of the 13th International Workshops on Enabling Technologies (WETICE04), IEEE Computer Society, June (2004), pp. 229- 234

[8] KayacikH. G., NZincir-HeywoodA., HeywoodM. I., On the capability of an SOM based intrusion detection systems, in Proc. Int. Joint Conference in Neural Networks. Vol. 3, 2003, 1808-1813

[9] Khaled Labib and V. Rao Vemuri,"An Application of Principal Component Analysis to the Detection and Visualization of Computer Network Attacks", A version of this paper appeared in the proceedings of SAR 2004

[10] MitchellT. M.. Machine Learning. McGraw Hill, 1997.

[11] QuinlanJ. R.. C4.5: Programs for machine learning. Morgan Kaufmann Publishers, 1993.

[12] QuinlanJ. R.. Induction of decision trees. Machine Learning, 1:1–106, 1986.

[13] Weiming Hu, Wei Hu, Steve Maybank, AdaBoost-based algorithm for network intrusion detection, IEEE Transactions on Systems, Man and Cybernetics, Vol. 38, April-2008, 577-583.

[14] Yoav Freund, Robert E.Schapire. A decision theoretic generalization of on-line learning and anapplication to boosting, Journal of Computer and System Sciences. Vol. 55, 1997, 119-139

[15] Yongiin Liu, Na Li, Leina Shi, Fangping Li, An intrusion detection method based on decision tree,International Conference on E-Health Networking, Digital Ecosystems and Technologies, 2010, 232-235.

[16] ZurutuzaU. and UribeetxeberriaR, "Intrusion Detection Alarm Correlation: A Survey", Proceedings of the IADAT International Conference on Telecommunications and computer Networks, 1-3 December, 2004