Category-Based Selection of Effective Parameters for Intrusion Detection

Peyman Kabiri[†] and Gholam Reza Zargar[††],

School of Computer Engineering, Iran's University of Science and Technology
16846-13114, Tehran, Iran

Summary
Existing intrusion detection techniques emphasize on building intrusion detection model based on all features provided. In feature-based intrusion detection, some selected features may found to be redundant and useless. Feature selection can reduce the computation power requirements and model complexity. This paper proposes a category-based selection of effective parameters for intrusion detection using principal components analysis method. In this paper, 32 basic features are selected from TCP/IP header. Tcpdump from DARPA 1998 dataset is used in the experiments as the test data. Principal Components Analysis (PCA) method is used to determine an optimal feature set. Experimental results show that feature reduction can improve detection rate for the category-based detection approach while maintaining the detection accuracy within an acceptable range. Feature reduction will speed up the training and the testing processes for the attack identification system considerably. Results presented in this paper show that normal state of the network and category of the attacks can be identified using a small number of a carefully selected network features.

Key words:
Intrusion Detection; Principal Components Analysis; Data Dimension Reduction; Feature Selection

1. Introduction

Nowadays, as more people make use of the internet, their computers and valuable data in their computer systems become a more interesting target for the intruders. Attackers scan the Internet constantly, searching for potential vulnerabilities in the machines that are connected to the network. Intruders aim at gaining control of a machine and to insert a malicious code into it. Later on, using these slaved machines (also called Zombies) intruder may initiate attacks such as worm attack, Distributed Denial-of-Service (DDoS) attack and probing attack [1].
Intrusion Detection Systems (IDS) are a subset of security management systems that are used to discover inappropriate, incorrect, or anomalous activities within computers or networks. During the past few years, existing IDSs take a variety of approaches to the task of detecting intrusion attempts. In order to develop such a system, several datasets are collected and then shared, so that, it can be used for this purpose. IDS use a dataset of signatures to analysis network traffic. This dataset includes data from the network traffic plus several other network parameters. In general, the volume of data is large; it includes thousands of traffic records with a number of various features such as the length of the connection, type of the protocol, type of the network service and lots of other information. Theoretically and ideally, the ability to discriminate attack from normal behavior should be performed better if more features are included in data analysis. However, results are sometimes unexpected because not every feature in the traffic data is relevant to the intrusion detection task. Among the large amount of features, some of the features might be irrelevant or with poor prediction ability with regard to the target patterns. Some of the features might be redundant due to their high inter-correlation with one or more of the other features in the dataset [2]. These problems not only hinder the detection speed but also decline the detection accuracy and performance of the IDS [3]. Achieving a better overall detection performance, any irrelevant and redundant features should be discarded from the original feature space. Selecting a meaningful subset of features from network traffic data stream is therefore a very important and indispensable task at the early stages of an intrusion detection process.
Feature reduction and identification of Effective Network Features for Probing Attack Detection performed in [4].
Feature reduction can be performed in several ways [5, 6 and 7]. This paper proposes a method based on TCP/IP header parameters. In the proposed approach, Principal Components Analysis (PCA) is used as a dimension reduction technique.

2. Related Works

MIT Lincoln Lab's DARPA intrusion detection evaluation dataset is employed to design and test intrusion detection systems. In 1999, recorded network traffic from the DARPA'98 Lincoln Lab dataset [8] was summarized into network connections with 41-features per connection. This formed the KDD'99 intrusion detection benchmark in international Knowledge Discovery and Data mining tools competition [9].
Chebrolu et al [5], identified important input features in building IDS that are computationally efficient and

Manuscript received September 5, 2009
Manuscript revised September 20, 2009

effective. In their reported work, they have investigated performance of three feature selection algorithms, i.e. Bayesian networks (BN), Classification and Regression Trees (CART) and an ensemble of BN and CART.
Sung and Mukkamala [6], have exploited SVM and Neural Network to identify and categorize features with respect to their importance in detection of specific kinds of attacks such as probing, DoS, Remote to Local (R2L), and User to Root (U2R). They have also demonstrated that elimination of these less important and irrelevant features did not reduce the performance of IDS significantly. Sung and Mukkamala [10] have analyzed data from large network traffic since it causes a prohibitively high overhead and often becomes a major problem for the IDS.

3. Data Reduction and feature selection with PCA

PCA is a mathematical method that transforms a number of possibly correlated variables into a new set of uncorrelated variables called principal components. The first principal component stands for the highest variability in the dataset.
In many datasets, the first several principal components have the highest contribution to the variance in the original dataset. Therefore, the rest can be disregarded with minimal loss of the information value during the dimension reduction process. The transformation works as follows [11].
Given a set of observations x_1, x_2, …, x_n where each observation is represented by a vector of length *m*, the data set is thus represented by a matrix $X_{n\times m}$

$$X_{n\times m}=\begin{bmatrix} x_{11} & x_{12} & . & . & . & x_{1m} \\ x_{21} & x_{22} & . & . & . & x_{2m} \\ . & . & . & . & . & . \\ x_{n1} & x_{n2} & . & . & . & x_{nm} \end{bmatrix}=[x_1,x_2,...x_n] \quad (1)$$

The mean is defined by the expected value. This is explained in equation (2).

$$m=\frac{1}{n}\sum_{n=1}^{n}x_i \quad (2)$$

The covariance matrix is defined in equation (3).

$$\sum x=\frac{1}{n-1}\sum_{n=1}^{n}(x_n-m)(x_n-m)^t \quad (3)$$

The covariance matrix is one of the most important mathematical concepts in the analysis of data.

If the data in the new co-ordinate system is presented by *y*, then it is desired to find a linear transformation G of the original co-ordinates, such that expression (4) is correct.

$$y=Gx=D^tx \quad (4)$$

Replacing G with D^t will make any future comparison of principal components with other transformation methods, much simpler.

The covariance matrix in the y space is defined by the equation (5)

$$\sum y=D^t\sum xD \quad (5)$$

where $\sum x$ is the covariance of the data in x space. Since $\sum y$ needs to be diagonal, D can be recognized as the matrix of eigenvectors of $\sum x$, providing D is an orthogonal matrix.
$\sum y$ is a diagonal matrix (6). Thus, $\sum y$ can be identified as the diagonal matrix of eigenvalues of $\sum x$.

$$\sum y=\begin{bmatrix} \lambda_1 & 0 & 0 & 0 \\ 0 & \lambda_2 & 0 & 0 \\ 0 & 0 & . & 0 \\ 0 & 0 & 0 & \lambda_n \end{bmatrix} \quad (6)$$

Let n to be the dimensionality of the data. The covariance matrix is used to calculate $\sum y$ that is a diagonal matrix. $\sum y$ is sorted and rearranged in the form of $\lambda_1>\lambda_2>....>\lambda_n$ so that the data exhibits maximum variance in y_1, the next largest variance in y_2 and so on, with minimum variance in y_n.

4. The Dataset and Pre-processing

The Dataset that was used in the reported work and the pre-processing applied on the dataset, are presented in the following sections.

5. The Dataset used in this work

In this work, 32 basic features are extracted from TCP/IP protocols [12](see Table 1). Basic features are also called packet header features. These features can be derived from packet headers without inspecting the payload. In the reported work, Tcpdump from the DARPA'98 dataset is used as the input dataset.

DARPA'98 dataset provides around 4 gigabytes of compressed Tcpdump data [13] for 7 weeks of the network traffic [14]. This dataset can be processed into about 5 millions of connection records each about 100 bytes in size. Dataset contains payload of the packets transmitted between hosts inside and outside a simulated military base. [1]BSM audit data from one UNIX Solaris host for some network sessions were also provided. DARPA 1998 Tcpdump dataset [14] was preprocessed and labeled using two class labels, e.g. normal and attack. The dataset contains 13 different types of attacks that are broadly categorized in five groups such as probing, DoS, U2R, R2L and anomalous behavior. Intention is categorize different intrusion methods into a number of categories. This approach aims to summarize the intrusion method into a few similar approaches. Following the proposed approach, system will be able to deal with variations of the different attacks within each category. Considering the DARPA'98 dataset, there are five main categories of attacks proposed in this paper. The proposed attack categories are listed and described in the following sections.

5.1 Denial of Service (DoS) Attacks

A denial of service attack is a class of attacks in which the attacker consumes computing or memory resources in such a way that the targeted system will be unable to handle legitimate requests, or denies legitimate user access to a machine. Apache2, Back, Land, Mail bomb, SYN Flood, Ping of death, Process table, Smurf, Teardrop, Udpstorm and Neptune attacks are some examples of the Dos attack.

5.1.1 Syn Flood Attack

TCP needs to establish a connection between a source host and a destination host, before any data can be transmitted between them. The connection process is called the three-way handshake (see Figure 1). In the first step, a SYN packet is sent from Source to the Destination node. Then Destination node sends a message to the Source with SYN and Ack Flags of it set. In the third step, Source sends a massage with its ACK flag set to Destination node. In this way a connection between source and destination is established. The third message may contain user payload data.

[1] Basic Security Monitoring (BSM)

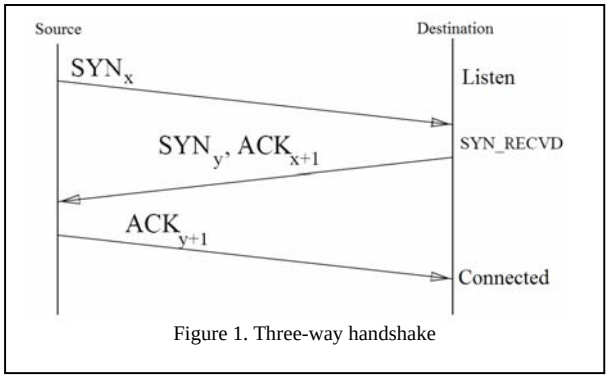

Figure 1. Three-way handshake

In a SYN flood attack a flood of TCP/SYN packets are sent to the victim machine, often with a forged sender address. Each of these packets is handled like a connection request, causing the server to create a half-open connection, by sending back a TCP/SYN-ACK packet, and waiting for a packet in response from the sender address.

Synflood is a DoS attack in which every TCP/IP implementation is vulnerable to some degree. Each half-open TCP connection made to a machine will cause the 'tcpd' server to add a record to the data structure that stores information describing all pending connections (see Figure 2). This data structure has size limit, and it may overflow by intentionally creating too many partially-open connections. The half-open connections data structure on the victim server system will eventually fill up. Here, unless the table is emptied, the system will be unable to accept any new incoming connections [15].

Normally, there is a time-out associated with a pending connection, so that, half-open connections will eventually expire and the victim server system will recover. However, the attacker system can simply continue sending IP-spoofed packets requesting new connections faster than the victim system can drop the pending connections. Christopher [16] believes that "Typical SYN flooding attacks can vary several parameters: the number of SYN packets per source address sent in a batch, the delay between successive batches, and the mode of source address allocation".

5.1.2 ICMP flood Attack

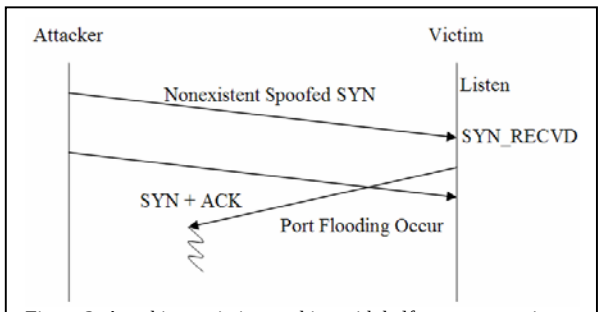

Figure 2. Attacking a victim machine with half-open connections.

A Smurf attack is a particular type of a flooding DoS attack over the public Internet. A DDoS ICMP flood attack occurs when the zombies send large volumes of ICMP_ECHO_REPLY packets to the victim machine. These packets signal the victim machine to reply and the combination of the traffic saturates the bandwidth of the victim's network connection. The method used is as follows:

- The attacking machine sends a ping request to one or more broadcast servers with a forged source IP address packets (IP address of the victim machine). Pinging is a mean to exploit the ICMP protocol, making it possible to test connections on a network by sending a packet and waiting for the response
- The broadcast server passes on the request to the entire network.
- All of the network's machines send a response to the broadcast server.
- The broadcast server redirects the responses to the target machine [17].

As such, when the attacking machine sends a request to several broadcast servers located on different networks, all of the responses from computers on the various networks will be routed to the target machine.

5.2 User to Root Attacks (U2R)

User to root exploits, are a class of attacks in which an attacker starts with accessing a normal user account on the system. Later on, intruder exploits system vulnerabilities to gain root access to the system. Examples for this type of attacks include buffer overflow, Eject, Ffbconfig, Fdformat, Loadmodule, Perl, Ps, Xterm, perlmagic and ffb attacks.

5.3 Remote to User Attacks (R2L)

A remote to user attack is a class of attacks in which an attacker sends packets to a machine over a network to gain access to the user accounts on that machine. Later on, the intruder exploits some vulnerability to gain local access as a user of that machine. Examples for these types of attack are Dictionary, Ftp_write, Guest, Imap, Named, Phf, Sendmail, Xlock, Xsnoop, guessing password and Dict attacks.

5.4 Probing Attacks

Probing is a class of attacks in which an attacker scans a network of computers to collect information or find known vulnerabilities. An intruder with a map of machines and services that are available on a network can use this information to look for exploits. There are different types of probing: some of them abuse the computer's legitimate features; other ones use social engineering techniques. This class of attacks is the most commonly heard and requires very little technical expertise. Examples are Ipsweep, Mscan, Nmap, Saint, Satan, ping-sweep and Portsweep attacks.

5.5 Undetermined Anomalous Behavior

There are anomalous user behaviors, such as "a manager becomes (i.e., behaves like) a system administrator". For example, when your computer was automatically blacklisted (blocked) by the network due to the number of

TABLE 1. Basic features extracted from the header of TCP/IP protocol

No.	Feature	Description
1	Protocol	Type of Protocol
2	Frame_lenght	Length of Frame
3	Capture_lenght	Length of Capture
4	Frame_IS_marked	Frame IS Marked
5	Coloring_rule_name	Coloring Rule name
6	Ethernet_type	Type of Ethernet Protocol
7	Ver_IP	IP Version
8	Header_lenght_IP	IP Header length
9	Differentiated_S	Differentiated Service
10	IP_Total_Lenght	IP total length
11	Identification_IP	Identification IP
12	MF_Flag_IP	More Fragment flag
13	DF_Flag_IP	Don't Fragment flag
14	Fragmentation_offset_IP	Fragmentation offset IP
15	Time_to_live_IP	Time to live IP
16	Protocol_no	Protocol number
17	Src_port	Source Port
18	Dst_port	Destination port
19	Stream_index	Stream Index number
20	Sequence_number	Sequence number
21	Ack_number	Acknowledgment number
22	Cwr_flag	Cwr Flag (status flag of the connection)
23	Ecn_echo_flag	Ecn Echo flag (status flag of the connection)
24	Urgent_flag	Urgent flag (status flag of the connection)
25	Ack_flag	Acknowledgment flag(status flag of the connection)
26	Psh_flag	push flag (status flag of the connection)
27	Rst_flag	Reset flag (status flag of the connection)
28	Syn_flag	Syn flag (status flag of the connection)
29	Fin_flag	Finish flag (status flag of the connection)
30	ICMP_Type	specifies the format of the ICMP message such as: (8=echo request and 0=echo reply)
31	ICMP_code	Further qualifies the ICMP message
32	ICMP_data	ICMP data

TABLE 2. Number of records in different selected categories

Category	Number of Records
DOS	483742
U2R	513
R2L	13935
PROB	10137
ANOMALY	71
NORMAL	88860

abnormal activities originating from your connection, it is possible that your computer is infected with a worm and/or virus.

6. Pre-processing

In the 1998 DARPA intrusion detection evaluation program, an environment was set up to acquire raw TCP/IP dump data for a network by simulating a typical U.S. Air Force LAN. The LAN was operated like a real environment, but being blasted with multiple attacks.
As displayed in Table 1, 32 basic features are extracted from TCP, IP, UDP and ICMP protocols in DARPA TCP/IP dump training dataset. Wireshark and Editcap softwares are used to analyze and minimize Tcpdump files [12][17]. Finally, a Visual Basic program extracts the 32 basic features.

7. Misuse Detection

Training data from the DARPA dataset includes “list files” that identify the timestamp, source host and port, destination host and port, and the name of each attack [18]. This information is used to select intrusion data for the purpose of pattern mining and feature construction, and to label each connection record with “normal” or “attack” label types. The final labeled training data is used for training the classifiers. Due to the large volume of audit data, connection records are stored in several data files. Table 2 shows all the connection records that fall within fifth day of the sixth week. Sequences of normal connection records are randomly extracted to create the normal dataset. Table 3 shows 65825 records that include records from both attack and normal state categories that are selected for the analysis.

Dictionary table is used to convert text data into numeric data.

8. Experiments

In the experiments carried out to detect intrusions, no attempt was made to distinguish different types of attacks.

TABLE 3. Number of records that are used for the calculations in different categories.

Category	Number of Records
DOS	19440
U2R	513
R2L	3798
PROB	10137
Anomaly	71
Normal	31866

Experiments were aimed on generating a categorized attacked or normal state dataset. In the experiments, 31866 normal connections are randomly selected to create the normal model and a dataset with 33959 attacks are included in the categorized attack dataset.

In the reported work, Matlab software was used as the implementation platform. PCA method can be used to reduce the dimensionality of a high dimensional data. The distance between each observation is used for anomaly detection. Effective parameters that are produced by the PCA can be used to detect anomalies in the dataset. Classes of relevant features with their associated information value are reported in Table 4. In this table, all attack categories are compared versus the normal state. As it is reported in this paper, different features were selected for separating attack categories from the normal state. As explained in section 3, in Table 4 first feature has maximum variance that obtained after calculating PCA.

TABLE 4. List of the most effective features in detecting the class.

Class name	Relevant Features in descending order	Total information value
DOS	28,19,5,1,16	99.75%
U2R	12,13,25,28,5	98.13%
R2L	27,25	97.69%
Probing	29,26,25,28,12,13,5,27,1,10	98.01%
Non deterministic Anomaly	26,28,25,2,3,10,19	99.29%
Normal	27,25	98.84%

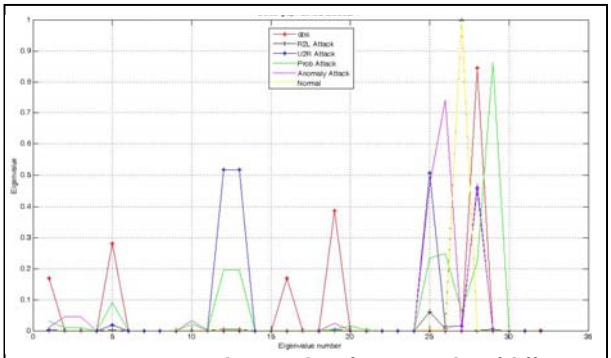

Figure 3. A comparison between the information value of different features in different states of the network operation.

For example, in DoS attacks category, feature number 28 and in probing attack category, feature number 29 have maximum variances. This means that, in DoS attack, feature number 28 changes faster than any other feature. Hence, feature number 28 can be used to detect intrusions in DoS attack category. Thresholds should be set on a selection of effective features to detect different attack categories. This threshold can be compared versus any change in values of the effective features and rise an alert once it is out of range. A comparison between the feature importance in different attack categories and the normal state is presented in figure 3. The Scree graph for the calculated PCA coefficients is depicted in figure 4.

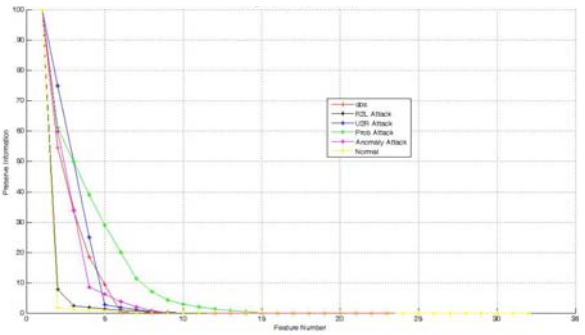

Figure 4. Comparison between Scree graphs for the different calculated PCA coefficients.

9. Experimental Results

Each attack has its own properties that are different from the properties of other attacks and even with the normal behavior. These features are used to compare each session versus a normal or a known attack behavior. As it is reported in Table 4, component number 27 with %98.22 of information, have the maximum information value in the normal dataset. Once the component number 25 is included, their total information value will rise to %98.84 of the total information value. Therefore, it can be said that the component number 25 does not have a significant effect in detecting the normal state. Comparing information value of the component number 25 versus threshold value for the normal state and R2L attack, normal state and R2L attack can be detected.

As Syn flood was explained in section 5.1.1, intruder may use Syn flag for the intrusion. The experimental result shows that component number 28 i.e., Syn Flag (see Table 1) have the highest information value for the detection of a DoS attack. Once DoS attack scenarios are compared against the effective features presented in Table 4, a relation between the behaviors of their parameters can be extracted.

In TCP scan attack, hackers use TCP scans to identify active devices, TCP port status and their TCP-based application-layer protocols. In TCP FIN scan, that it, is a kind of TCP scan attack, hackers scan the network to identify listening TCP port numbers. The TCP packets used in this scan have only their TCP FIN flag set. Results from the experiments in Table 4, for probing attacks, show that the 29th component in Table 1 i.e. Fin flag has the highest information value. Hence, it is the most important component in the probing scenario attack and for the detection purpose. Comparing result of this experiment with TCP FIN scan scenario, intrusion attempt by probing attack can be detected. In Table 4, result of the probing attack scenario shows that the first four components are TCP flags with 70.97% of information value.

Figure 5 explains which components should be selected for the detection of the normal state and which components for detecting attack category. If a threshold is defined for each effective parameter in table 4, intrusion detection system will be able to distinguish normal records from attack records in a shorter detection time and without a great change in the detection accuracy. As it is shown in flowchart depicted in figure 5, all attack categories in the

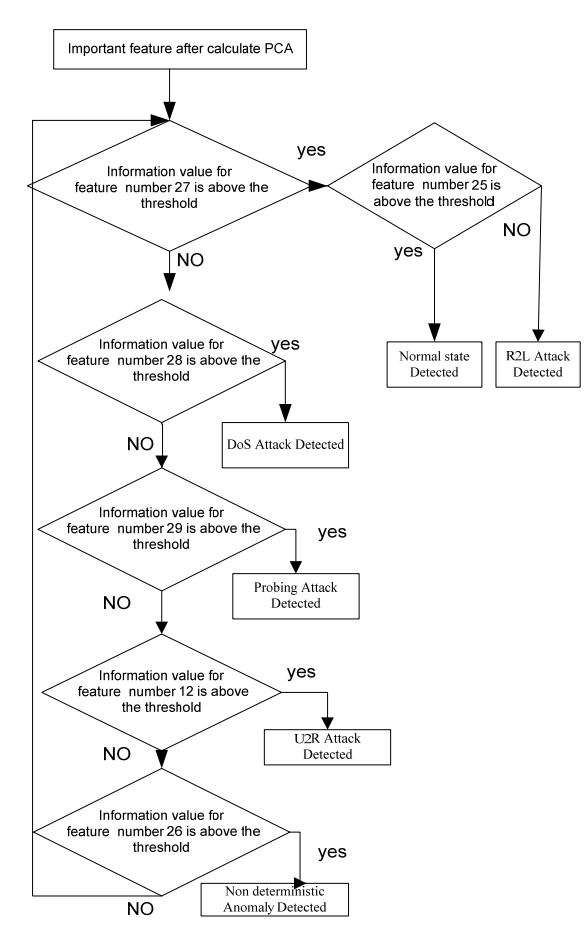

Figure 5. Flowchart for detecting attack category.

dataset are detected. Table 5, shows components of several attack scenarios compared versus each other using effective parameters reported in this paper.

10. Conclusions

In this paper, a method based on Principal Component Analysis (PCA) for Category-Based intrusion identification is proposed. In anomaly detection method, attacks are usually detected based on a presumed normal behavior. In this approach, usually normal and anomalous behaviors are separated with a low false positive rate. Using the aforementioned approach for identifying individual types of attacks may lead to even lower quality of results. Paper reports a new Category-Based intrusion detection approach that can produce better and more accurate results in identifying the category of the attacks instead of the precise type of the attack. Results presented in this paper show that normal state of the network and category of the attacks can be identified using a small number of a carefully selected network features. On the other hand, it is proven that certain features have no contribution to intrusion detection. This also indicates that there are analytical solutions for the feature selection that are not based on the trial and error. The possibility and feasibility of implementing an intrusion detection technique based on characterization of different types of attacks such as DoS, probes, U2R and R2L attacks are investigated is investigated. Results of this investigation seem to be promising. In the proposed model, each network connection is transformed into a data vector by its extracted feature values. PCA is employed to reduce the dimensionality of high dimensional data vectors. Experimental results show category-based selection of effective parameters with dimension reduction using PCA can be used for intrusion detection.

11. Future work

Plan for the future work is to use classification methods to detect intrusions. Using the results derived from the intrusion detection and comparing it versus both the full and the reduced feature sets, one can analyze the difference in their accuracy and speed.

References

[1] M.F. Abdollah, A.H. Yaacob, S. Sahib, I. Mohamad, M.F. Iskandar, “Revealing the Influence of Feature Selection for Fast Attack Detection”, International Journal of Computer Science and Network Security, vol.8, No.8, pp. 107-115, August 2008.

[2] F. Sabahi, A. Movaghar, “Intrusion Detection: A Survey”, 3rd international conference on system and network communication, ICSNC08, pp.23-26, October 2008.

[3] T.S. Chou, K.K. Yen, and J. Luo, “Network Intrusion Detection Design Using Feature Selection of Soft Computing Paradigms”, International Journal of Computational Intelligence, Vol.4, No.3, pp.196-208, Summer 2008.

[4] G. Zargar, P. Kabiri, “Identification of Effective Network Feature for Probing Attack Detection”, proceedings of First International Conference on Network Digital Technologies (NDT2009), pp. 405-410, July 2009.

[5] S. Chebrolu, A. Abraham, J. Thomas, “Feature Deduction and Ensemble Design of Intrusion Detection Systems”, Computers and Security, Elsevier Science, Vol.24, Issue 4, pp. 295-307, June 2005.

[6] A.H. Sung, S. Mukkamala, “Identifying important features for intrusion detection using support vector machines and neural networks”, Proceedings of International Symposium on Applications and the Internet(SAINT), pp. 209-216, 2003.

[7] R. Agrawal, J. Gehrke, D. Gunopulos, P. Raghavan, “Automatic Subspace Clustering of High dimensional Data for Data Mining applications”, Proceeding of ACMSIGMOD International Conference on Management of Data, Seattle WA, pp. 94-105, 1998.

[8] The 1998 intrusion detection off-line evaluation plan. MIT Lincoln Lab., Information Systems Technology Group. http://www.11.mit.edu/IST/ideval/docs/1998/id98-eval-11.txt, 25 March 1998.

[9] Knowledge discovery in databases DARPA archive. Task description. http:\\www.kdd.ics.edu/databases/kddcup99/Task.html, as visited on 20 January 2009.

[10] A.H. Sung, S. Mukkamala. “The Feature Selection and Intrusion Detection Problems.” ASIAN 2004. LNCS, Vol. 3321, Springer Hieldelberg, pp. 468-482, 2004.

[11] W. Wang, R. Battiti, “Identifying Intrusions in Computer Networks based on Principal Component Analysis”, http://eprints.biblio.unitn.it/archive/00000917/ as visited on 30 May 2009.

[12] http://www.wireshark.org, as visited on 29 January 2009.

[13] http://www.Tcpdump.org, as visited on 25 January 2009.

[14] MIT Lincoln Laboratory http://www.ll.mit.edu/IST/ideval/, as visited on 27 January 2009

[15] A. Hassanzadeh, B. Sadeghian, “Intrusion Detection with Data Correlation Relation Graph”, Third International Conference on Availability, Reliability and Security (ARES 08), pp. 982-989, 2008.

[16] L. Christopher, Schuba, V. Ivan, Krsul, et al, “Analysis of a denial of service attack on TCP”, Proceedings of the IEEE Symposium on Security and Privacy, page 208-223, 1997.

[17] http://www.nordu.net/articles/smurf.html, as visited on 17 Sep. 2009.

[18] http://www.wireshark.org/docs/man-pages/editcap.html, as visited on 20 Jan 2009.

[19] Wenke Lee, “A Data Mining Framework for Constructing Feature and Model for Intrusion Detection System”, PhD thesis University of Columbia, 1999.

Gholam Reza Zargar received his B.Sc. in Computer Hardware Engineering from Iran's University of Science and Technology in 1991 and his M.Sc. in GIS (Geographic Information Systems) from Shahid Chamran University in 2008. He is currently a M.Sc. student in ICT (Information Communication Technology) at the Iran's University of Science and Technology. He has 15 years of work experience in Khozestan water & power autority Co. IT department (a subsidiary of Iranian ministry of energy). His main research area is in computer and network security.

Peyman Kabiri received his PhD in Computing and MSc in Real time Systems from the Nottingham Trent University, Nottingham-UK in years 2000 and 1996 respectively. He received his B.Eng. in Computer Hardware Engineering from Iran's University of Science and Technology, Tehran-Iran in 1992. He was with the Faculty of Computer Science/ University of New Brunswick as project coordinator from early September 2004 until the end of September 2005. His previous academic positions were as follows: Assistant professor in Department of Computer Engineering Iran's University of Science and Technology (where he is currently an assistant professor) and Assistant Professor in Azad University – central branch – Faculty of Engineering both in Tehran-Iran. His research interests include Machine Learning, Remote Sensing, Robotics and Network Intrusion Detection.