Effective Dimensionality Reduction of Payload-Based Anomaly Detection in TMAD Model for HTTP Payload

Mohsen Kakavand[1], Norwati Mustapha[1], Aida Mustapha[2], Mohd Taufik Abdullah[1]
[1]Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, Malaysia.
[e-mail: kakavandirit@gmail.com, norwati@upm.edu.my, mtaufik@upm.edu.my]
[2]Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia, Malaysia.
[e-mail: aidam@uthm.edu.my]
*Corresponding author: Mohsen Kakavand

Received November 2, 2015; revised March 12, 2016; revised May 13, 2016; accepted June 19, 2016; published August 31, 2016

Abstract

Intrusion Detection System (IDS) in general considers a big amount of data that are highly redundant and irrelevant. This trait causes slow instruction, assessment procedures, high resource consumption and poor detection rate. Due to their expensive computational requirements during both training and detection, IDSs are mostly ineffective for real-time anomaly detection. This paper proposes a dimensionality reduction technique that is able to enhance the performance of IDSs up to constant time $O(1)$ based on the Principle Component Analysis (PCA). Furthermore, the present study offers a feature selection approach for identifying major components in real time. The PCA algorithm transforms high-dimensional feature vectors into a low-dimensional feature space, which is used to determine the optimum volume of factors. The proposed approach was assessed using HTTP packet payload of ISCX 2012 IDS and DARPA 1999 dataset. The experimental outcome demonstrated that our proposed anomaly detection achieved promising results with 97% detection rate with 1.2% false positive rate for ISCX 2012 dataset and 100% detection rate with 0.06% false positive rate for DARPA 1999 dataset. Our proposed anomaly detection also achieved comparable performance in terms of computational complexity when compared to three state-of-the-art anomaly detection systems.

Keywords: Principle Component Analysis, Intrusion Detection System, Dimensionality Reduction, Feature Selection, Packet Payload

1. Introduction

Hackers are known as creative and talented individuals who exploit vulnerabilities in digital systems and network applications. Vulnerable programs are common means used by the hackers to attack victims and to access their individual data. Exploits are generally harmful codes using vulnerabilities throughout known application with intention to damage the machine. They are used in viruses made to manipulate our individual data. Exploits are also the actual philosopher's stones in cybercrime magic regarding targeted attacks or cyber warfare. Most well-known cyber weapons such as the Stuxnet and Duqu utilized exploits to sneak into heavily guarded IT infrastructures for the purpose of sabotage and cyber espionage. Conti et al. [1] carried out one of the most significant studies in computer security, which is the Man-In-The-Middle (MITM) attack. The research categorized the MITM attack into two endpoints; victims and a third party (attacker). The attacker has access to communication channels between two endpoints, therefore manipulating their messages. In addition to the MITM attack, HTTP web service attacks have also been categorized and studied [2]. The Kaspersky Security Experts Team (KSET) reported 132 million vulnerabilities from customers' personal computers over 11 million users in 52 weeks. **Fig. 1** illustrates the number of vulnerability incidents from February 2003 to December 2012 [3].

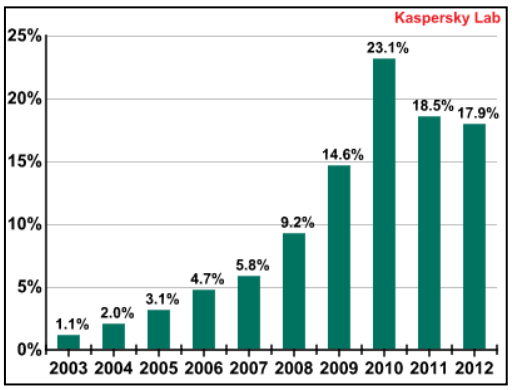


Fig. 1. Vulnerability incidents reported to KSET

In defending the computer networks, the Intrusion Detection Systems (IDS) are influential systems designed to discover potential exploits or malicious activities in vulnerabilities and network traffic. Furthermore, they are able to flag alarm in case of suspicious activities. Two major detection methods among the IDSs are the signature-based and anomaly-based [4], [5]. The signature-based intrusion detection (SID) includes a database of defined signatures for matching strings against the attacks. Hackers would then craft attack variants to beat the signature strings, or enhance the attacks to exploit new vulnerabilities. Whenever SID becomes short, anomaly-based intrusion detection (AID) attempts to close the holes. AID is a newer approach as compared to SID in the fight against misuse and exploits. Basically, AID is not a cure-all. However, when it is used along with an influential SID solution, it becomes an influential tool for network protection.

Other popular intrusions such as the HTTP protocol and worms include the delivery of anomaly payload to a susceptible application or service. These attacks might be identified by checking the packet payload. For example, an HTTP transaction consists of a request command (sent from the client to the server) in the form of formatted blocks of data called HTTP messages. **Fig. 2** shows that HTTP transactions consist of request (inbound) and response (outbound) packet payloads. As illustrated in this figure, the payload patterns involve strings such as "GET" and "POST". Therefore, in order to detect such attack, the packet payload must be checked.

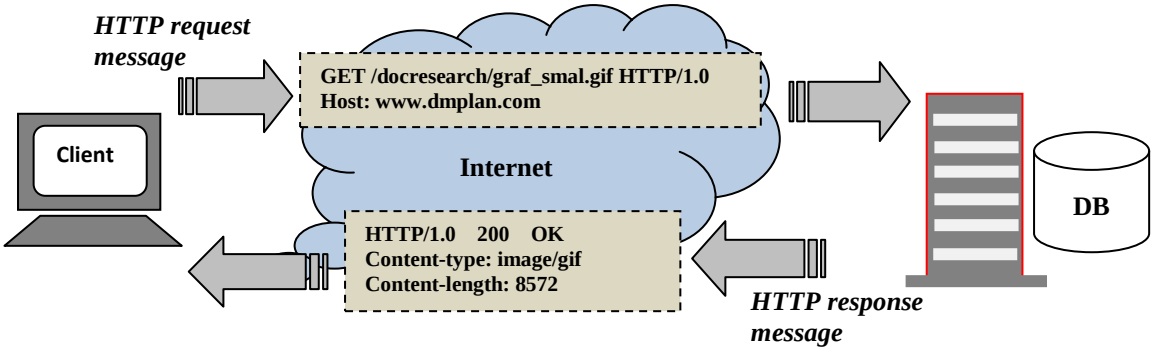


Fig. 2. HTTP transactions consist of inbound and outbound

AID is a promising soultion to detect attack payload, but there are still a great deal of challenges associated with the encrypted payloads. This is because attacks present in encrypted payload data are often treated as normal. Moreover, many applications use the Secure Sockets Layer (SSL) – and its successor Transport Layer Security (TLS) – as a basic part for encrypted data but attacker can still infer a significant amount of information from the analysis of the properly encrypted network traffic [6]. We examined HTTP traffic as "clear text", and believe that our solution can be used for encrypted payload applied in decryption point.

Network anomaly detection has been an important research topic within the area of data mining and machine learning. Many real-world applications such as the intrusion detection system require an effective approach to identify deviated data instances along with a low false positive rate and a high detection rate. Therefore, recently, many models as shown in [7], [8], [9], [10] have been proposed to reduce the high false positive rates, but most of proposed approaches have high computation complexity or are based on statistical computation. Such detection approaches are time-consuming, hence degrading the performance of an ID system due to the significant increase of computer resources (memory and CPU time). Therefore, effective dimensionality reduction of payload-based anomaly detection becomes critical when considering the computational complexity and classification performance.

To address these limitations, we proposed an improved Text Mining-based Anomaly Detection (TMAD) model based on the Principle Component Analysis (PCA), which is one of the most commonly employed dimensionality reduction technique. In terms of feature selection, the Guttman-Kaiser criterion is utilized to determine the optimum volume of factors while using the PCA. One of the motives behind the selection of the PCA as the dimensionality reduction technique is to decrease the computational cost of the anomaly

detection system through its ability to operate on the input feature vector's space directly without transforming the data into another output space, as in the case with self-learning techniques. In the PCA, dimensionality reduction is achieved by calculating the first few principal components representing the highest eigenvalue in the components of the input feature vector, without transformation on the input. This results in a translation where n correlated features are represented in order to reduce the number of features to $d < n$, which will be both uncorrelated and linear combinations of the original ones, hence facilitating the detection process [11]. The present study shows that our proposed model effectively reduces the number of processed features from 256 down to 25 and 20 for both ISCX 2012 and DARPA 1999 datasets, respectively.

1.1 Motivation, Objectives and Contributions

Ultimately, the aim of the anomaly detection system is to detect and prevent any form of attacks to computer systems. To effectively detect the intruders, various detection mechanisms are available to–convert the anomaly detection system to powerful anomaly-based IDSs. In detecting the delivery of an anomaly payload, anomaly-based intrusion detection (AID) attempts to fill the gap. However, the major issue with the AID is the computational overhead, which can become prohibitively high. When the network speed is faster, security analysis methods must emerge to keep up with the increased network throughput [12]. A potential method for simplifying the analysis of such high dimensional data is to use the dimensionality reduction approach, in order to decrease the number of features, eliminate unnecessary, redundant or noisy data, while at the same time preserving vital features of the original data and bringing the immediate impacts for IDS.

In this light, the major issues with anomaly detection systems are their efficiency and speed. When the amount of network traffic is high, it would be challenging to use complicated algorithms that are fast enough to detect intrusions before being too late. Although many advanced algorithms achieve a high detection rate, they are computationally complicated for practical and real time use [13]. The present study proposes an effective dimensionality reduction technique in payload-based anomaly detection systems using the Principle Component Analysis (PCA) solution that allows the researchers to efficiently calculate the dominant eigenvectors, and to rate the importance of various components of a high dimensional feature space.

During the feature selection stage, Guttman-Kaiser criterion that is a statistical method is applied for identifying major components in real-time intrusion detection system. The feature selection solution is then used to determine the optimum volume of factors while using the PCA. From the reviewed research on payload-based anomaly detection, TMAD [14] and McPAD [9] are the two commonly used approaches. TMAD uses *n*-gram text categorization (TC) and term frequency-inverse document frequency (TF-IDF) methods, which serve as the commonly term weighting schemes. In TMAD, the weights reflect the significance of features in a particular packet payload of the considered collection. It is also possible to handle a huge amount of data containing redundant and irrelevant features with low false positive rates and high detection rates.

Nonetheless, the main drawback of these approaches is that their computational cost might not always satisfy real-time intrusion detection systems. For example, in McPAD [9], a multiple one-class SVM system is used for classification of anomaly detection, but the proposed algorithm requires at least $O(nm + mks)$ for computation complexity. In this

paper, the proposed model considers such problems since we only require $O(1)$ for computational cost. In summary, the main contributions of this research would be as follows:

1. Identifying the optimum volume of components through an efficient feature subset selection method.
2. Decreasing the dimensionality of feature spaces to achieve high detection accuracy with low false alarm rate for anomaly detection among HTTP intrusions.
3. Reducing computational cost by significantly reducing the optimal dimension.

The rest of this paper is organized as follows: Section 2 surveys other related works especially in dimensionality reduction. Section 3 proposed an improved TMAD framework with the Principle Component Analysis (PCA). Section 4 describes details of experiments with ISCX 2012 and DARPA 1999 dataset and compares PCA-based TMAD model with results from three state-of-the-art; TMAD, McPAD, and LDA-based GSAD. Finally, Section 5 draws conclusion and indicates the direction of future studies.

2. Related work

There are two methods for decreasing dimensions of the feature space. The first method is the feature selection by choosing a subset of the original traits, as the new traits are based on a selection norm. Among the examples of linear features are Chi-squared statistic, mutual information, information gain, and correlation coefficient [15]. The second method is the feature extraction by reducing the dimension through combining or projecting the original traits. The most popular feature extraction approaches include the principal component analysis (PCA) and particularly multidimensional scaling (MDS), latent semantic analysis (LSA), learning vector quantization (LVQ), local linear embedding (LLE) and self-organizing maps (SOM) [16]. The present study focuses on dimensionality reduction for feature extraction, as new features are designed based on transformation of the input traits. Although there are different feature extraction methods such as the Independent Component Analysis (ICA) [17], Principal Component Analysis (PCA) [18], Correlation-based Feature Selection (CFS) [19], and Linear Discriminate Analysis (LDA) [20] that also decrease the packet header features, few studies have examined feature extraction methods specifically for packet payloads such as in [7], [8], [9] and [10].

In [7], the byte frequency distributions of 256 ASCII characters were directly sorted into six containers, including 0, 1-3, 4-6, 7-11, 12-15 and 16-255. According to Anagram [8], there is a content anomaly detector, which is resistant to mimicry attack. In this type of resistance, a Bloom Filter (BF) was utilized to decrease memory overhead. In McPAD [9], a multiple one-class SVM system was used to classify anomaly detection; whereby the dimensional feature space is 256^2 = 65,536 since each byte has values ranging from 0-255 and n = 2. Then, the dimensionality of the feature space decreased using a clustering technique. However, the norms of cluster selection were not explained. Next, in order to reduce the heavy computational cost of an anomaly intrusion detection system, Tan et al. [10] proposed a Linear Discriminate Analysis (LDA) for payload packet feature selection. The LDA is a method used in statistics, pattern recognition and machine learning to find a linear combination of features that characterizes or separates two or more classes of objects or events, since it considers class discrimination. However, this model has high computational requirements for the network intrusion detection.

In addition to the mentioned works, some other anomaly detection approaches have been recently proposed [21], [22], and [23]. Among them, online oversampling PCA [21] is a

highly efficient framework to the computational cost of calculating the principle directions in a large dataset for anomaly detection. However, the proposed method will duplicate the target instance for several times. Thus, it might not be sufficient to perform anomaly detection simply based on the most dominant eigenvector [24]. On the other hand, the modified PCA does not use any other statistical and dimensionality reduction algorithms to help the method versus nonlinear data. Moreover, the online anomaly detection is able to achieve significant reduction in computational cost. This solution still suffers from inaccuracy in real-world datasets in comparison with other methods.

In [22], the authors proposed an anomaly-based intrusion detection technique, Packet Cunk Anomaly Detector (PCkAD), which uses Chunk as small useful payload for reducing computation. However, the model uses *n*-gram (*n*=5), which means 256 different characters are possible; the model chooses $n = 5$, which means the maximum number of dimension feature space is 256^5. Since the performance of many algorithms used depends on the size of individual feature vector, high input dimensions would make the performance slow. In another study, Juvonen et al. [23] proposed an online anomaly detection system that could detect web server log attacks using three different techniques including random projection (RP), principle component analysis (PCA) and diffusion maps (DM). The results from three methods show that this approach can be used for dimensionality reduction before anomaly detection. However, it has some problems as it is still pretty tough to make the right training data size, and the selection of anomaly threshold is challenging. Moreover, unfortunately, the system is unable to detect attacks that compromise the security of a web server before logging.

Difficulties associated with the high dimensionality feature space are generally resolved through the application of dimensionality reduction techniques such as the PCA [25], LDA [26], and Co-clustering [27]. Dimensionality reduction can be either applied in a pre-processing step prior to clustering or be integrated into the clustering framework itself. Dimensionality reduction methods are widely used in intrusion detection systems. Within this paradigm, the PCA and LDA have been demonstrated to be useful for many applications with big amount of data that are highly redundant and irrelevant. Although one might think that LDA should always outperform PCA (since it deals directly with class discrimination-supervised); however, in terms of real-time anomaly detection systems, it is very difficult and expensive to obtain a labeled dataset that represents the real network activities with both normal and attack traffic [28]. One particular advantage of PCA is label agnostic (performs unsupervised transformation), whereby it treats the entire dataset as a whole while at the same time being less sensitive to different training datasets [29].

Apart from the PCA and LDA, the Co-clustering algorithm has also been applied as a data dimensionality reduction technique by clustering the records (rows) and fields (columns). It has been successfully applied in many text mining applications such as in [30] and [31]. However, the Co-clustering algorithm has a high time complexity, whereby it requires at least $O(t(k+1)mn)$, where m, n, and t are rows, columns and the number of iterations, respectively [32]. It is more computationally expensive than the proposed model since we only require $O(1)$ for the computational cost.

To fill the gap, this research proposed the use of PCA algorithm for feature extraction method. PCA provides an insight into the space where the given data resides. It also helps eliminate distractive noise and seek the optimal lower dimensional representation for data with a high dimensionality while at the same time retaining the high detection rate. Moreover, this will not reduce only the traffic volume but also the processing time. In addition, we suggest using a component selection based on Guttman-Kaiser criterion in the

feature subspace selection. This statistical solution is employed to explain variability among the observed and correlated variables in determining the threshold value for important principal components. The evaluation will be in terms of variance among different components in multidimensional feature spaces.

3. Methodology

This research proposed a mathematical approach to feature extraction using the PCA algorithm. The improved TMAD framework includes data preparation, pre-processing, packet encoding, dimensionality reduction by PCA, and finally network classification. **Fig. 3** illustrates the proposed framework of improved TMAD with PCA. The following subsections present the detailed tasks in each process:

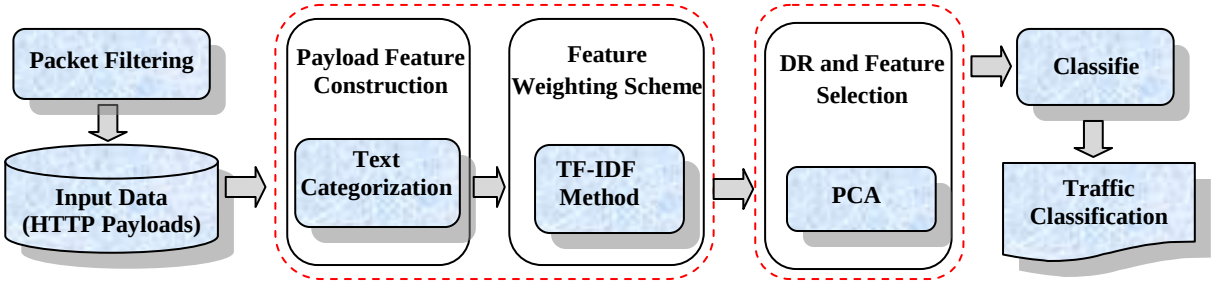


Fig. 3. Overview of TMAD with PCA

3.1 Data preparation stage

To evaluate our proposed model, we prepared two different large datasets of Information Security Center of Excellence (ISCX IDS 2012) [33], and DARPA 1999/MIT Lincoln laboratories IDS [34]. The first stage in the improved TMAD is data preparation; whereby various network applications are prepared in ISCX 2012 dataset. Network applications are divided into different categories such as Bit Torrent, HTTP Web application, HTTP Image Transfer, Secure web, MSN Messenger, MS-SQL, and SMTP. Then, we extracted the packet payloads from HTTP Web application. The arranged dataset is utilized in the data pre-processing stage. For the second dataset, we used HTTP packet payload of DARPA 1999 dataset.

3.2 Data pre-processing stage

Text categorization (TC) is a language-independent tool of gauging topical similarity in text documents. Traditionally, the text database is processed by passing a sliding window of '*n*' characters over the dataset and counting the occurrence of each *n*-gram. With unigram (1-gram) TC [35], it is possible to extract a pattern of characters from a given input flow by using a sliding window of length n across a string of tokens and request feature analysis. It derives raw data payload characters using *n*-gram ($n = 1$) text categorization approach from extracting sequences of payload request, and transforming extract sequences directly into

feature vectors [36]. Formally, the set F of features corresponds to all possible patterns of length n in ASCII characters that range from 0 to 255 as defined in **Equation (1)**:

$$T = \{0, \ldots, 255\}^n \tag{1}$$

To demonstrate how the *n*-gram works with HTTP packet payload, this study considered the simulated payload request *X*='*rpnnpn*' where the set of all characters is limited to '*r*', '*p*', and '*n*'. If $n = 2$, the resulting 2-grams are: '*rp*', '*pn*', '*nn*', '*np*', and '*pn*', respectively. If we consider a set of raw data payload as $P = \{p_1, p_2, p_3, \ldots, p_n\}$ and a set of features (characters) as $F = \{f_1, f_2, f_3, \ldots, f_{256}\}$, the set P can be represented as a *Payload-Feature Matrix* (PFM), where rows and columns are indexed by the raw data payload and the features, respectively. Each element of this matrix refers to the weight of each feature in its related payload as shown in **Equation 2**.

$$P = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ \vdots \\ p_N \end{bmatrix} = \begin{array}{c} \begin{matrix} f_1 & f_2 & f_3 & \cdots & f_{256} \end{matrix} \\ \begin{bmatrix} w_{1,1} & w_{1,2} & w_{1,3} & \cdots & w_{1,256} \\ w_{2,1} & w_{2,2} & w_{2,3} & \cdots & w_{2,256} \\ w_{3,1} & w_{3,2} & w_{3,3} & \cdots & w_{3,256} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ w_{N,1} & w_{N,2} & w_{N,3} & \cdots & w_{N,256} \end{bmatrix} \end{array} \tag{2}$$

Based on the above PFM, which is considered as a dataset, each data payload is mapped into an *m*-dimensional space as follows [37] in **Equation (3)**:

$$\forall\, p_i \in P\,, \qquad 1 \le i \le N$$

$$\Phi: p_i \longmapsto \Phi(p_i) = \left(w_{i,1}, w_{i,2}, \ldots, w_{i,256}\right) \mathbb{R}^{256} \tag{3}$$

In the data pre-processing stage, the unigram model ($n = 1$) extracts raw features using text categorization technique from the packet payload, and converts observations into a series of feature vectors. From the sample given in Table 1, we extracted the unigram values from the payload in order to construct a matrix with the *n*-gram features as shown in **Equation (2)**.

During the next stage, the value (weight) of the unigram features is computed using several methods, which are term frequency (TF) and inverse document frequency (IDF) to determine the value (weight) of each entry of w_{ij}. These methods will be explained in details in the next stage.

Table 1. Illustration of the sample payload where characters is extracted by unigram

Payload	**GET /docresearch/graf22_smal9.gif HTTP/1.0**																			
	First 20 characters as n-gram (*n=1*)																			
Extracts raw features	G	E	T	/	d	o	c	r	e	s	e	a	r	c	h	/	g	r	a	f
	2	2	-	s	m	a	l	9	.	g	i	f	H	T	T	P	/	1	.	0

3.3 Packet encoding stage

TF-IDF examines the vector space model and serves as a weighting scheme [35] to enhance the text categorization performance. In the schemes, the weights refer to the importance of a feature in an especial document of the selected collection. Huge weights are often utilized in relevant documents but rarely in the entire document collection [38]. Consequently, the data sources are processed, and the vector space model is determined to present a convenient data structure for text classification. The vector space model represents the data payload as vectors in m-dimensional space (256 dimensions). For instance, each payload 'p_i' is described through a numerical feature vector.

Thus, a weight for a feature 'f_j' in data payload 'p_i' is calculated by term frequency tf (p_i, f_j) and inverse document frequency idf (f_j), explaining the feature specificity within the data payload collection. In addition to the term frequency and inverse document frequency (TF-IDF) as defined in **Equation (4)**, a length normalization factor is used to assure that all data payloads possess equal chances of retrieving independent of their lengths as shown in **Equation (5)**. 'N' is the size of the data payload collection 'P' and is the number of data payload in 'P' involving feature 'f_j'. In text mining, a corpus is encoded as a matrix where each document is explained by a row in a matrix. The resulting matrix is recognized as a (weighted) *Payload-Feature Matrix* (PFM).

$$idf(f_j) = \left(\frac{N}{p_{f_j}}\right) \tag{4}$$

$$W(p_i, f_j) = \frac{tf(p_i, f_j)\, log\left(\frac{N}{p_{f_j}}\right)}{\sqrt{\sum_{j=1}^{m} tf(p_i, f_j)^2 \left(log\left(\frac{N}{p_{f_j}}\right)\right)^2}} \tag{5}$$

For better understanding the payload-feature matrix of the payload vector space, let us consider the simple example of **Fig. 4,** which assumes packet data collection of seven payloads set as $\{p_1, p_2, \ldots, p_7\} \subset$ p and a set of features (three-dimensional vector space) as $\{f_1, f_2, f_3\} \subset$ F. Then, from **Equations (4)** and **(5)**, we obtain the TF-IDF weighting scheme for one payload vector. **Table 2** presents a broader picture of the TF-IDF weighting scheme for one payload vector, for each feature, frequency (global), term frequency (TF), inverse document frequency (IDF) and TF-IDF values.

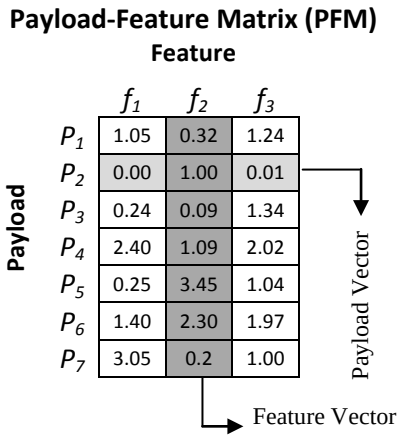


Fig. 4. Illustrative example of the payload-feature matrix for a sample packet payload

Table 2. Illustrative TF-IDF weighting scheme for one payload vector

ASCII	Feature	TF	Freq	IDF	TF-IDF
		$tf(p_i, f_j)$	p_{f_j}	$idf(p_i, f_j)$	$tf(p_i, f_j)\, log(N/p_{f_j})$
98	*b*	1	6962	1.10	1.05
105	*i*	4	31598	0.18	0.32
111	*o*	2	23959	0.46	1.24
108	*l*	3	27902	0.31	0.93
102	*f*	1	27375	0.33	0.33
97	*a*	1	32082	0.17	0.17
110	*n*	1	13560	1.03	1.03
114	*r*	2	15017	0.93	1.86
109	*m*	1	35161	0.08	0.08
100	*d*	1	8940	1.44	1.44

Based on **Fig. 4**, each row $\{p_1, p_2, \ldots, p_7\}$ of the payload-feature matrix constitutes a TF-IDF vector representing one packet payload in the collection. Each of these vectors has three features $\{f_1, f_2, f_3\}$, which correspond to the three characters of the collection. In this regard, we will discuss in details (practice) three basic TF-IDF weighting procedures which help to improve the performance of the vector space model by assigning specific values to each non-zero element in the payload-feature matrix.

The combined use of term frequency and inverse document frequency is commonly referred to as TF-IDF weighting. The following describes the steps taken to consider the procedure of the TF-IDF weighting schemes over the raw character counts in a packet payload collection. Briefly, the steps involved in the procedure for carrying out the TF-IDF method are as follows:

Step 1. Extract and choose an initial payload feature set.
Step 2. Compute term frequency of each feature for the corresponding payload as $tf(p_i, f_j)$.

Step 3. Consider the logarithm of the computed ratio that is the ratio between the logarithms of the total number of payloads and the payload frequency corresponding to each feature as $log(N/p_{f_j})$.

Step 4. Consider the product between the term frequencies computed in step 2 and inverse document frequencies computed in step 3 as $tf(p_i, f_j)\ log(N/p_{f_j})$.

Step 5. Divide each of the term frequencies by the total number of features occurring in the corresponding payload, in order to compensate for possible effects resulting from different lengths (in number of features) of the payload represented in the model. This normalized term frequency weighting can be computed based on **Equation (6)**:

$$W\left(p_i, f_j\right) = \frac{tf(p_i,f_j)\, log\left(\frac{N}{p_{f_j}}\right)}{\sqrt{\sum_{j=1}^{m} tf(p_i,f_j)^2\ (\ log(\frac{N}{p_{f_j}}))^2}} \tag{6}$$

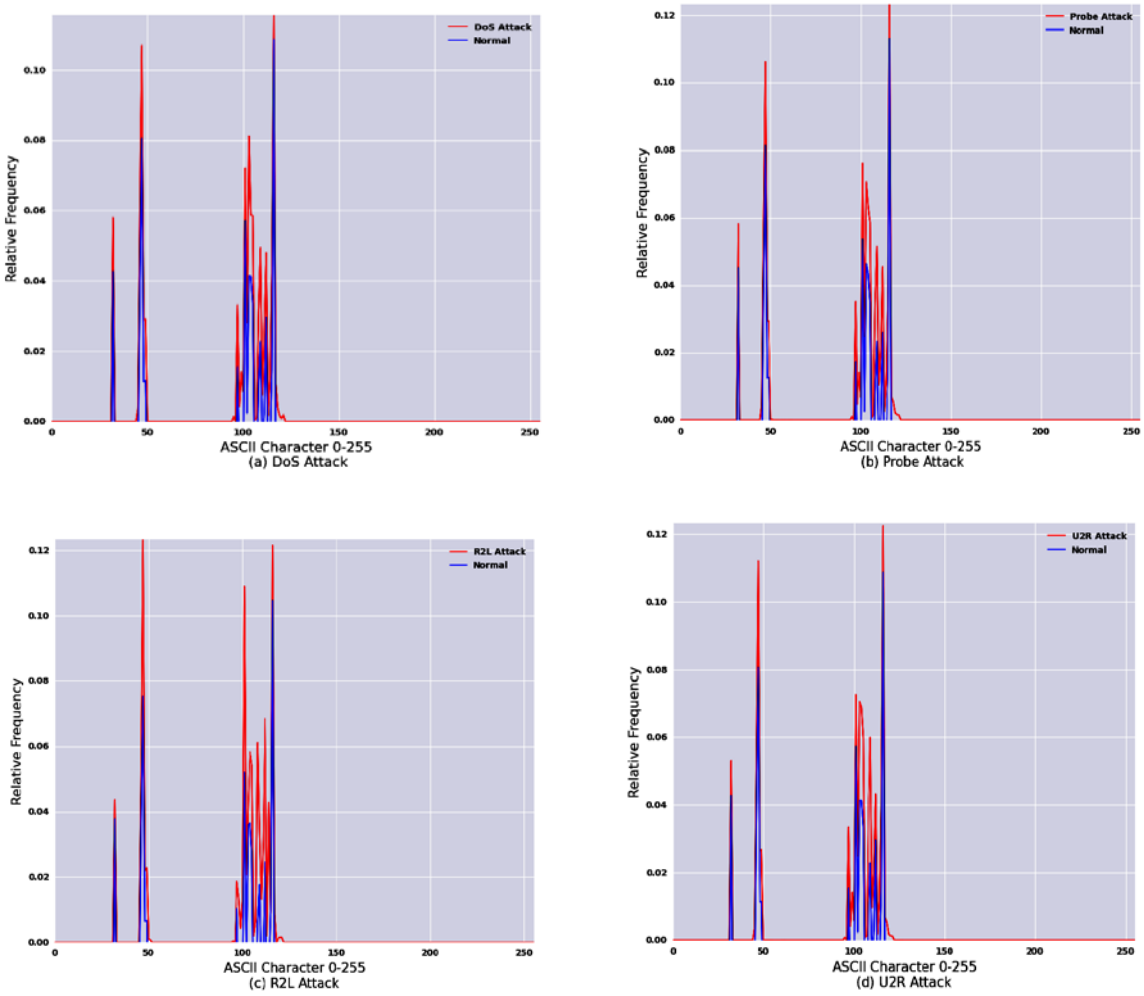


Fig. 5. Relative frequencies of characters (a) DoS attack, (b) Probe attack, (c) R2L, and (d) U2R

The character relative frequencies of attack payloads are different in **Fig. 5** exposed to the behaviors of DoS, Probe, R2L, and U2R attacks. For the Probe attack, the "/" character has the highest frequency and the other characters share even frequencies. **Table 3** compares the highest and lowest character frequencies of different HTTP attack traffics. Although it is still believed that the performance can be enhanced through giving various weights to the payload bytes based on the degree of importance, finding the suitable weights is complicated.

Table 3. Comparison characters with the highest and lowest frequencies

Attacks	Feature type	Character					
		Total freq	No. feature	Highest	Freq	lowest	Freq
DoS	Unigram Model (1-gram)	203438	39	(t)	23344	(~)	2
Probe		298104	40	(/)	31492	(=)	14
R2L		12253	32	(/)	1513	(_)	5
U2R		136224	39	(t)	16619	(=)	5

3.4 Dimensionality reduction and selection stage

In this research, the dimensionality reduction technique used in the improved Text Mining-based Anomaly Detection (TMAD) model is based on the Principle Component Analysis (PCA) [25] that is an influential method to reduce dimensionality by a linear mapping of the *n*-dimensional feature space into a reduced *m*-dimensional feature space. In the experiment, this research will employ the Guttman-Kaiser criterion [39] in order to reserve as much relevant information as possible.

A) Dimensionality reduction based on PCA: PCA is an approach to analyze relationships among multivariable by finding the principal components denoted as a linear combination, and explaining the entire changes with different components. Like a linear mathematical method, PCA can be enhanced depending on eigenvector-based multivariate evaluation. The main idea is to proficiently represent information by transforming a collection of findings into a completely new orthonormalized coordinate system, where the data tend to be maximally de-correlated. The axes (eigenvectors) will contain much more variations (eigenvalues) with higher contributions to the data presentation. The initial numbers of axes using the highest contributions are often utilized to create a new lower dimensional feature space giving effective presentations for the data. **Fig. 6** shows the algorithm for dimensionality reduction based on PCA.

Based on **Fig. 6**, the algorithm for dimensionality reduction is set to analyze the feature space of a given dataset $X = [x_1\ x_2 \ldots x_n]$, where $x_i = \left[f_1^i\ f_2^i \ldots f_t^i\right]^T$ $(1 \le i \le n)$ denotes the i^{th} D observation with t features. First, zero-mean normalization is performed within the data arranged for the findings. The zero-mean dataset is presented through $X_{zm} = [(x_1 - \bar{x})\ (x_2 - \bar{x}) \ldots (x_n - \bar{x})]$, where $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$. Then, the principal components (i.e., eigenvectors) are calculated by conducting eigen decomposition on the sample covariance matrix where $C_x = \frac{1}{n-1} X_{zm}\ X^T{}_{zm}$. Next, C_x is decomposed into a matrix W and a diagonal matrix Λ. The two matrices satisfy the condition in which $\Lambda W = C_x W$. Consequently, Λ and W are usually categorized throughout climbing down order against the variance contributed to each component.

The columns of the matrix W indicate the particular eigenvectors (i.e., the principal components) from the covariance matrix C_x , along with the factors across the diagonal of the matrix Λ including the placed eigenvalues connected with the corresponding eigenvectors inside the matrix W. Nonetheless, PCA is not able to identify the number of key factors that should be preserved. Therefore, to identify the best number of principal components preserved based on the PCA analysis, the Guttman-Kaiser selection criterion is proposed.

Principle Component Analysis

Require: *Data set X {X contains n instance, and each of which has t features}*

Ensure: $1 \le k \le t$

1: $\bar{x} \leftarrow \frac{1}{n}\sum_{i=1}^{n} x_i$

2: $X_{zm} \leftarrow X - \bar{x}$ {Subtract $\bar{x}$ from each instance in X}

3: $C_x \leftarrow \frac{1}{n-1} X_{zm} X_{zm}^T$

4: Obtain Λ and W, which are subject to $\Lambda W = C_x W$

5: **For** i = 1 to n **do**

6: $\sigma_i^2 \leftarrow \sum_{\iota=1}^{i} \lambda_\iota$

7: **End for**

8: Plot $\{\sigma_1^2, \sigma_2^2, \ldots, \sigma_n^2\}$

9: Locate the elbow on the scree and identify K of the "elbow"

10: $W_k \leftarrow$ Top first k eigenvalues/eigenvectors of W

11: **Return** W_k

Fig. 6. Dimensionality reduction based on the PCA algorithm

B) Component selection based on Guttman-Kaiser Criterion*:* The Guttman-Kaiser criterion [40] related to every single factor is displayed by the related eigenvalues. Principal components associated with eigenvalues are extracted from a covariance matrix. The rules propose to hold only principal components as they are the eigenvalues larger than 1. While 1 might be considered as the average variance for the standardized data, the rule has been modified in order to select PCs derived from the covariance matrix as follows:

$$KC = \sum_{i=1}^{256} \frac{\sigma_i^2}{256} \tag{7}$$

Nonetheless, the components which are larger in magnitude than the average of the eigenvalues are preserved. In the case of eigenvalues extracted from a covariance matrix, the average is determined using **Equation (7)**. The subset associated with the main features, related to the selected k, represents the smaller feature spaces, which in turn serve as the best presentation for any packet payload data. Through representing the feature vector $x_i = [f_1^i \; f_2^i \; \ldots \; f_t^i]^T$ onto smaller feature spaces, the dimension of the feature vector will also decrease into a smaller sized values, namely k.

3.5 Network classification

For network classification, this research proposes the Mahalanobis Distances Map algorithm as shown in **Fig. 7**.

Mahalanobis Distance Map

Require: *Data set X {X contains n instance, and each of which has* k_{final} *features}*

1: $\mu \leftarrow \frac{1}{n}\sum_{i=1}^{n} x_i$

2: *Center each value by the mean*

$X \leftarrow X - \mu$ {Subtract μ from each instance in X}

3: *Calculate covariance matrix* $\Sigma_{(k_{final} \times k_{final})}$ *for Data Set X*

4: Obtain the transpose of $(X - \mu)$ and the inverse of Σ

5: *Calculate the Mahalanobis distance as follows*:

$d_M \leftarrow \sqrt{(x-\mu)^T . \Sigma^{-1} . (x-\mu)}$

6: Calculate weight score w to detect an intrusive *activity*

$$w \leftarrow \sum_{a,b=1}^{k_{final}} \frac{(d_{obj(a,b)} - \bar{d}_{nor(a,b)})^2}{\sigma^2_{nor(a,b)}}, d(a,b) \in d_M$$

exactly where $\bar{d}_{nor(a,b)}$, $\sigma^2_{nor(a,b)}$ *include the average and the variance of the* (a,b)*th element and* $d_{obj(a,b)}$ *is the* (a,b)*th of the distance map of the newly arriving packet.*

Fig. 7. Mahalanobis Distances Map Algorithm

The Mahalanobis distance between the particular point x and the mean μ of the normal data is computed in **Equation (8)**:

$$d_M \leftarrow \sqrt{(x-\mu)^T . \Sigma^{-1} . (x-\mu)} \tag{8}$$

The hidden correlations related to the projected element vector $\left[x_1\ x_2 \ldots x_{k_{final}}\right]$ are obtained from the projected original feature vector $x_i = \left[f_1^i\ f_2^i \ldots\ f_t^i\right]^T$ and mapped onto the k_{final} dimensional feature subspace $\left[u_1\ u_2 \ldots u_{k_{final}}\right]$. Next, the correlations among the packets are calculated using **Equations (9)** and **(10)**:

$$\sum_a (x_a - \mu)(x_a - \mu)^T \left(1 \le a \le k_{final}\right) \tag{9}$$

$$d_{(a,b)} = \frac{(x_a - x_b)(x_a - x_b)^T}{\sum_a + \sum_b} \left(1 \le a, b \le k_{final}\right) \tag{10}$$

Where x_a presents the a^{th} estimated feature within the projected feature vector, μ refers to the standard of each projected feature, $d_{(a,b)}$ presents the Mahalanobis distance related to the a^{th} projected feature and the b^{th} projected feature. $\sum_a$ points to the covariance value of each projected feature and finally d_M defines the distance map (the pattern of a network packet). Then, the distance map d_M generates the network traffic profiles (normal and attack) of the training as shown in **Equation 11**:

$$d_M = \begin{bmatrix} d_{(1,1)} & d_{(1,2)} & \cdots & d_{(1,k_{final})} \\ d_{(2,1)} & d_{(2,2)} & & d_{(2,k_{final})} \\ \vdots & & \ddots & \vdots \\ d_{(k_{final},1)} & d_{(k_{final},2)} & \cdots & d_{(k_{final},k_{final})} \end{bmatrix} \quad (11)$$

Mahalanobis distance is the norm utilized to calculate the dissimilarity between the developed profiles and the new incoming network traffic profiles. Weight score W is measured using **Equation (12)** and is used to detect any intrusive activity.

$$W = \sum_{a,b=1}^{K_{final}} \frac{\left(d_{obj(a,b)} - \bar{d}_{nor(a,b)}\right)^2}{\sigma^2_{nor(a,b)}} \quad (12)$$

Where $d_{obj(a,b)}$ and $\sigma^2_{nor(a,b)}$ are the average and the variance of the $(a,b)^{th}$ element in the distance map of the normal profile given in **Equation (13)**, and $d_{obj(a,b)}$ is the $(a,b)^{th}$ element of the distance map of the new incoming packet represented in **Equation (14)**.

$$D_{nor} = \left[d_{nor\,(a,b)}\right]_{K_{final} \times K_{final}} \quad (13)$$

$$D_{obj} = \left[d_{obj\,(a,b)}\right]_{K_{final} \times K_{final}} \quad (14)$$

The tested sample payload is finally classified as an attack or a normal record using $threshold = \bar{d}_{nor(a,b)}$, where the threshold is determined by the average total mahalanobis distance of normal packets. If the $d_{test(a,b)}$ exceeds the threshold, the input network packet is determined as an intrusion; otherwise, it will be classified as normal using **Equation (15)**. The details of our experiments are given in Section 4.

$$if\ d_{test(a,b)} \leq \bar{d}_{nor(a,b)}\ \ then\ \ classified\ as\ normal, else\ return\ as\ attack \quad (15)$$

4. Framework assessment

This section introduces the dataset used to validate the proposed PCA-based approach to dimensionality reduction in TMAD model as well as the evaluation measurements, training and testing setup, and finally presents the results and analyses.

4.1 Dataset

We conducted two different sets of experiments to evaluate the effectiveness of dimensionality reduction of payload-based anomaly detection in the improved TMAD model. The first experiment used the ISCX datasets 2012 [33], which were collected under the sponsorship of Information Security Centre of Excellence (ISCX). All the network traffic in the dataset were included in both normal network traffic and attack traffic for system assessment. ISCX dataset 2012 includes categories of attacks involving scan, DoS, R2L, U2R and DDoS. The entire ISCX-labeled dataset is composed by 1,512,000 packets that cover seven days of network activity. However, both datasets were not ready for training and testing. In preparing the dataset, the ISCX HTTP/GET traffic was randomized into two groups: a training set that is composed of 80% of the HTTP/GET traffic, and a testing set that is composed of the remaining 20% of the traffic.

The second experiment used the DARPA 1999/MIT Lincoln laboratories IDS [34], the most comprehensive dataset with the entire content packet available for researchers. All the network traffic data include the full payload of each packet contents recorded in tcpdump format and provided for evaluation. We trained the proposed model on the DARPA 1999 dataset using week 1 (5 days, attack free) and week 3 (7 days, attack free) and then evaluated the model using week 4 and week 5 containing the anomaly traffics. Although the DARPA dataset is outdated and has been criticized [28] due to the nature of the simulation environment that created the data, it is widely accepted for comparison. This dataset has been used in many studies, and results of tests involving these data have been reported in many publications.

4.2 Evaluation measurements

In this segment, the information involved in the confusion matrix is analyzed using the Detection Rate (DR) and False Positive Rate (FPR). Assessment metrics are introduced for the analysis. The metrics utilized are True Positive (TP) when the number of actual attack is classified as an attack, True Negative (TN) when the number of actual normal is classified as normal, False Positive (FP) when the number of actual normal is classified as attack and False Negative (FN) when the number of actual attack is classified as normal. **Table 4** represents the definition of a confusion matrix. Subsequently, detection and false positive rate can be estimated as shown in **Table 4**.

Table 4. Confusion matrix

		Predicted Class	
		Normal	**Attack**
Actual Class	**Normal**	True Negative (TN)	False Positive (FP)
	Attack	False Negative (FN)	True Positive (TP)

The major objectives of intrusion detection system are to maximize the true positive rate (detection rate) and minimize the false positive rate of a proposed method. **Table 5** displays the most commonly used learning metric in performance assessment of the intrusion detection system. Furthermore, **Equations (16)** and **(17)** show the formula to calculate the detection rate and the false positive rate.

Table 5. Classification Metric

Evaluation Metrics	Description
Detection Rate (DR)	(TP) ÷ (TP + FN)
False Positive Rate (FPR)	(FP) ÷ (TN + FN)
Accuracy	(TP + TN) ÷ (TP+FP+TN+FN)

Detection Rate $DR = \frac{TP}{TP+FN} \times 100\%$ (16)

False Positive $FP = \frac{FP}{FP+TN} \times 100\%$ (17)

4.3 Training and testing program

As discussed in the dimensionality reduction section, the PCA technique was used to analyze the raw data, namely the ASCII character occurrence frequencies in the training dataset. This is carried out by projecting the raw data on a reduced feature space. The principal components were selected using the Guttman-Kaiser criterion based on the outcome of PCA.

When the Guttman-Kaiser criterion was first applied to find the component selection, principal components related to an eigenvalue in which the magnitude is higher than average are preserved (the average vectors, (KC) =1.3317e-04 and 4.5054e-04 are computed for both ISCX 2012 and DARPA1999 dataset, respectively). Through the Guttman-Kaiser Principle, the particular 90th percentiles are attained (the 90th percentile = 1.3317e-04 and 4.5054e-04). If the proper value of a component is higher than the 90th percentiles from the simulated valuations, then the particular component will be actually retained. This means the value of k subset equal to 25 and 20 is attained (k=25 and 20 for both ISCX 2012 and DARPA1999 dataset, respectively).

The result of using the Guttman-Kaiser criterion suggested a selection of the first 25 and 20 principle components, which showed the best subspace for data presentation. **Figs. 8** and **9** illustrate how the corresponding eigenvectors were captured for ISCX and DARPA dataset, respectively. The principle components were sorted in a descending order with respect to the values of the corresponding variances, which in turn determined the number of important components that should be retained for network traffic analysis. This shows that there are as many reliable components as the eigenvalues, which were greater than average.

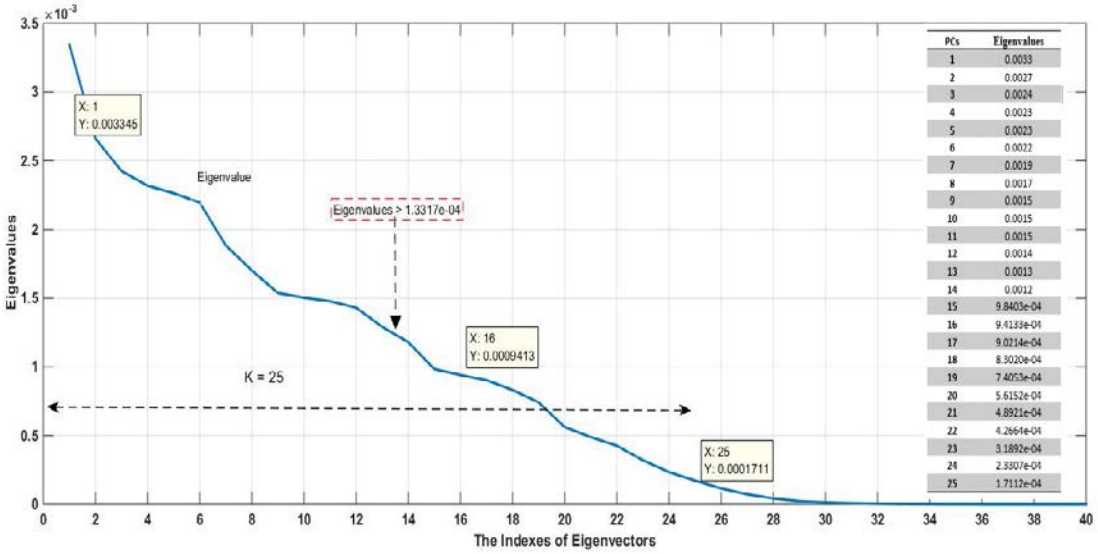


Fig. 8. Principle components selection for ISCX 2012 data traffic

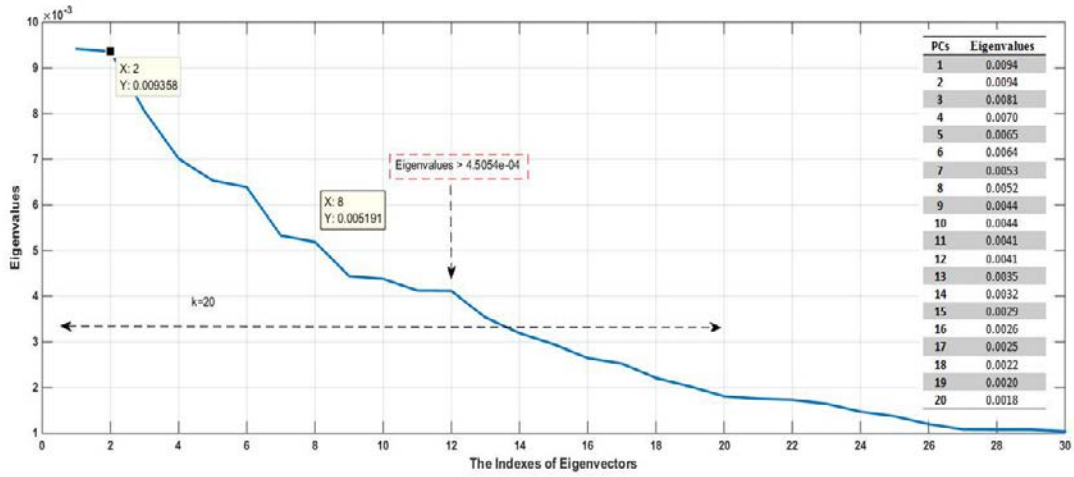


Fig. 9. Principle components selection for DARPA1999 data traffic

4.4 Results and analysis

For the intrusion detection model, we used the extracted HTTP packet payload from the ISCX 2012 IDS and DARPA 1999 datasets. We analyzed our recommended detection method against both the normal data and the attack data in the evaluation data collection. From the initial stage of the studies, we found the optimum small set of main components. Subsequently, we designed many experiments to determine the efficiency of PCA-based TMAD model when utilizing a variety of small sets of principal components in both datasets.

The particular assessment results are described in **Tables 6** and **7**, which demonstrate the relationship between the false positive rate (FPR) and detection rate (DR) with the accuracy

against various thresholds from 1 to 3 on both datasets. Note that the particular threshold sets the amount of the significant difference, which is identified by the intrusion detection system, which involved an analyzed object and discovered normal profiles. Tests running against the various sets of important components (i.e., the selected lower dimensional feature spaces) are shown in **Figs. 8** and **9**. Furthermore, **Tables 6** and **7** report the results obtained from our proposed anomaly detection system on both ISCX 2012 IDS and DARPA 1999 datasets.

Table 6. Model evaluation based on ISCX 2012 Dataset

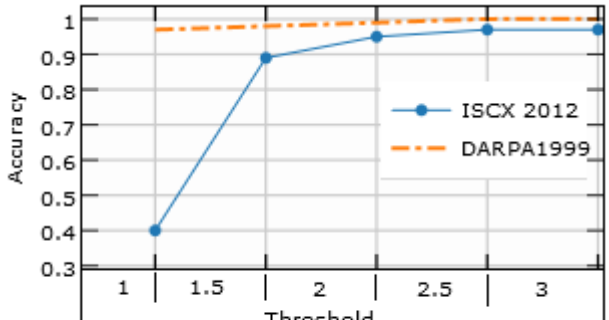
Evaluation Metrics	Threshold				
	1σ	**1.5σ**	**2σ**	**2.5σ**	**3σ**
False Positive	0.2%	1.1%	1.2%	1.2%	1.3%
True Positive (Detection Rate)	14%	85%	94%	97%	97%
Accuracy	**40%**	**89%**	**95%**	**97%**	**97%**

Table 7. Model evaluation based on DARPA1999 Dataset

Evaluation Metrics	Threshold				
	1σ	**1.5σ**	**2σ**	**2.5σ**	**3σ**
False Positive	1.4 %	1.2 %	0.1%	0.06%	0.05%
True Positive (Detection Rate)	97%	98%	99%	100%	98%
Accuracy	**97%**	**98%**	**99%**	**100%**	**100%**

As shown in both tables, the threshold value controls the degree of the dissimilarity as acknowledged by the anomaly detection system, between a test object and the respective learnt normal profile. If the dissimilarity is higher than the determined threshold, the test object is classified as an anomaly. Moreover, based on **Table 6,** higher true positive rate was achieved when a greater threshold was accepted. In fact, greater thresholds produce higher false positive. On the other hand, from **Table 7,** we can see that higher true positive rate lies in between various thresholds ranging from 2 to 2.5 with an increase of 0.5 interval. This means greater thresholds represent a lower false positive rate. However, we can find out that the accuracy of both evaluation dataset declined when a lower threshold of 2σ is accepted. After this point, the performance of the proposed PCA-based TMAD dropped significantly to 40% and 97% for ISCX 2012 and DARPA1999 datasets, respectively.

Fig. 10 visualizes the trade-off between accuracy and the threshold. The proposed PCA-based TMAD model demonstrated an encouraging effectiveness on the DARPA 1999 dataset with 97% accuracy when the limit is determined to 1σ. As the threshold reached to 3σ, the accuracy rate increased to 100%. However, while considering the ISCX 2012 IDS dataset, the proposed detection model accomplished lower performance. Nevertheless, it still delivered a desirable accuracy rate (i.e., 97%) at the threshold of 3σ.



	1	1.5	2	2.5	3
ISCX 2012	40%	89%	95%	97%	97%
DARPA 1999	97%	98%	99%	100%	100%

Fig. 10. Relationship between accuracy and threshold

Based on **Fig. 10**, the results showed that the proposed PCA-based model performed better using the DARPA 1999 dataset as compared to using the ISCX 2012 dataset. However, the model is more effective in handling realistic networks and traffics with novel attacks in ISCX 2012. The DARPA 1999 dataset has its critics, as it is quite dated and Web behaviors have evolved significantly since its first inception. Next, **Figs. 11** and **12** illustrate the trade-off between the false positive rate and true positive through a number of principal components with changes in their results when a different threshold of 2.5 σ (best performance achieved from threshold of 2.5σ) was used.

Fig. 11 shows the difference between the true and false positive rates generated by PCA-based TMAD, where *k* subset equal to 25 was attained (*k*=25 for ISCX 2012). Based on this figure, the best false and true positive rates were achieved with 25-dimentional feature space. According to **Fig. 12**, the selection of the important features was performed on DARPA 1999 dataset, where *k* subset equal to 20 was attained. This shows the difference of the true and false positive rates among various optimal feature spaces are achieved with 20-dimentioanl feature space. However, these optimal feature spaces are not always practicable, and the best performance may be achieved around these numbers. For example (where k = 20 in our case), using only the first five principal components to represent the HTTP traffic is not feasible in our anomaly detection system. This is because the projections (i.e., PCA) constructed through only five features are always identical for all records after normalization. Hence, we will choose the first 20 optimal feature spaces instead of the first five principal components.

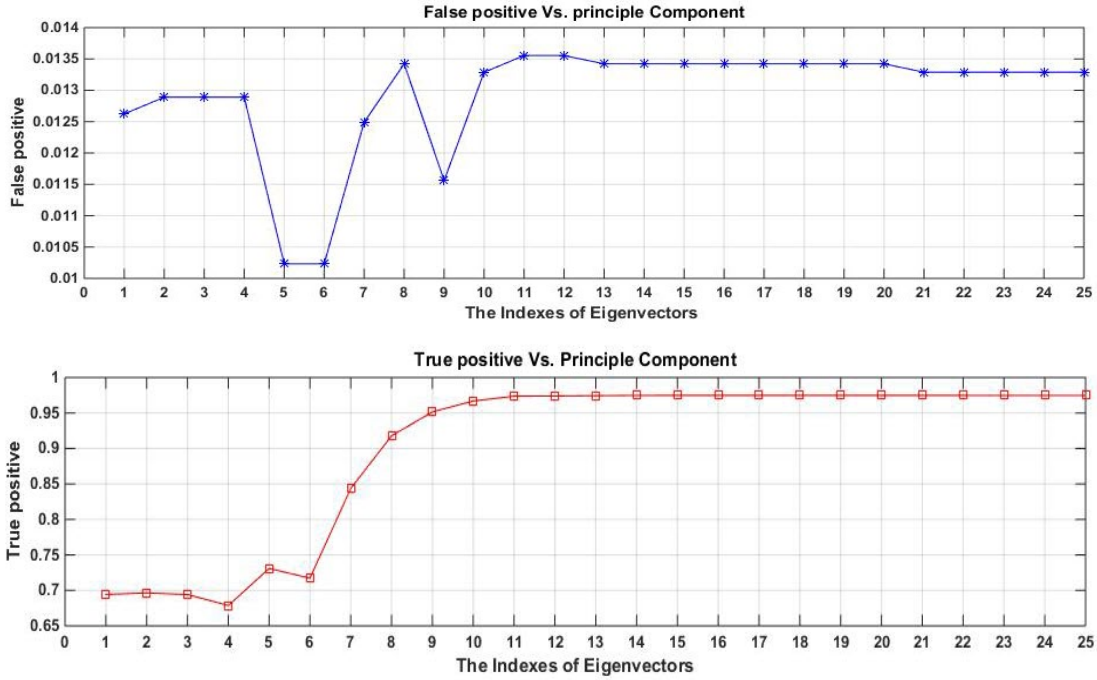


Fig. 11. True and false positive rate vs. principle components (ISCX Dataset)

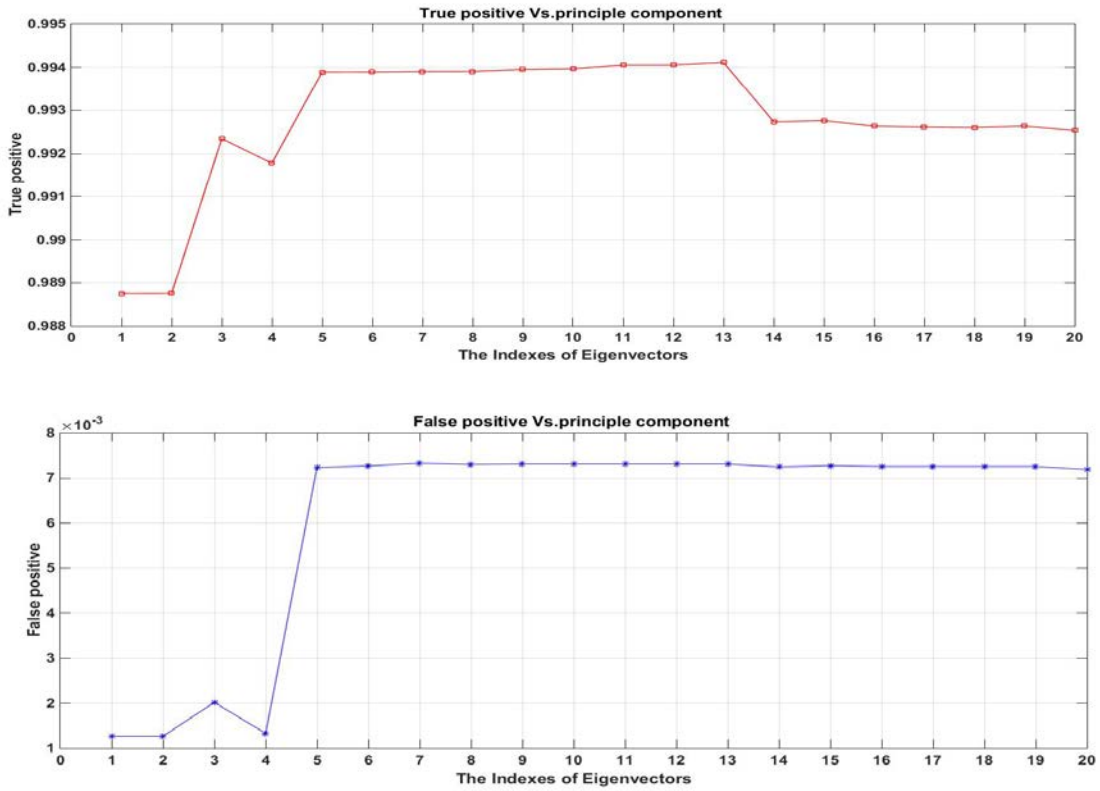


Fig. 12. True and false positive rates vs. principle components (DARPA1999)

4.5 Comparison with state-of-the-art

To demonstrate a better snapshot of the proposed PCA-based TMAD performance, we compared the proposed PCA-based TMAD against three state-of-the-art payload-based anomaly detection systems; TMAD, McPAD, and LDA-based GSAD using the same ISCX 2012 and DARPA 1999 datasets. The computational complexity on the four models was also evaluated.

4.5.1 Anomaly detection accuracy

In the first experiment, we randomly divided the HTTP/GET traffic in ISCX 2012 dataset into two groups; 80% for training and 20 % for testing. In accordance with the details provided by **Table 8**, the detection rate and the false positive rate were computed for both TMAD and PCA-based TMAD. The comparison results illustrated that the proposed PCA-based TMAD model achieved 97% detection rate with 1.2% false positive. The results also showed that all 256 features were not used for the detection of network attacks compared to the TMAD model that obtained 97.44% detection rate with 1.3% false positive rate which are based on all feature spaces for anomaly detection. Although the proposed model has low false positive rate compared to TMAD model [14] on ISCX 2012 dataset, it does not show a significant advance in terms of accuracy. It can be concluded that PCA and a feature selection approach can help decrease the dimensionality of dataset from 265 to 25. Consequently, it is more computationally efficient in handling datasets with a high dimensionality.

Table 8. Performance comparison (ISCX 2012)

IDS Model	Number of PCs	Detection Rate (DR)	False positive (FP)
PCA-Based TMAD	25	97 %	1.2 %
TMAD	256	97.44 %	1.3 %

In the second experiment, we used DARPA 1999 dataset for training three anomaly detection models, which are the PCA-based TMAD, LDA–based GSAD and McPAD. After training the different models on DARPA 1999 dataset, we tested the models on the entire attacks datasets (ISCX 2012 and DARPA 1999). The threshold value used was 2.5σ based on previous finding, whereby the value provided better outcomes as compared to other thresholds when we attempted to achieve the detection attacks at very low false positives. **Table 9** reports the detection rate and true positives on the number of principle components (PCs) obtained from PCA-based TMAD, McPAD and LDA-based GSAD models, respectively. We compared the anomaly detection performance of our proposed model on DARPA 1999 evaluation dataset with those achieved by two other anomaly detection approaches [9] and [10]. The performance of the proposed model and LDA-based GSAD showed that the detection rates of both anomaly detection systems are comparatively equal (100% DR). Although our proposed model has a low false positive rate (0.06% FPR), the proposed LDA-based GSAD approach is not proven to be a good candidate with high false positive rate (3.7% FPR). **Table 9** shows that the selected features of our proposed model considerably reduced the number of features in order to avoid computational cost as an apparent advantage in real-time intrusion detection systems.

Table 9. Performance comparison (DARPA 1999)

IDS Model	Number of PCs	Detection Rate (DR)	False positive (FP)
PCA-Based TMAD	20	100%	0.06%
LDA-Based GSAD	300	100%	3.7%
McPAD	256^2	95%	10^{-5}

The results in **Table 9** reveal that McPAD [9] achieved very low false positive rate as compared to our proposed anomaly detection system, but it is only for shell-code attacks at a false positive rate of 10^{-5}. Furthermore, the detection rate was 95%. Thus, it cannot confirm if this approach operates better than our proposed model on HTTP attack detection. Moreover, the proposed anomaly detection easily achieved higher detection rate while maintaining a relatively low false positive rate and a very small dimension in order to have an efficient real-time performance.

4.5.2 Computational complexity

We evaluated the computational complexity of the proposed PCA-based TMAD throughout two phases; data pre-processing and classification. Given an input payload q of size n and a set value of v, the frequency of unigram and bigram is usually computed with $O(n)$. Since the number of extracted features is constant, we obtained 2^8 =256 by the PCA-based TMAD, 2^{16} = 65536 by McPAD, 2^8 = 256 by LDA-based GSAD and 2^8 = 256 by TMAD model. The dimensionality reduction in the McPAD model was by the occurrence frequency distribution of $2v$-grams to the k feature clusters using a simple look-up table; therefore, the total number of operations is always less than 2^{16}. On the other hand, the LDA-based GSAD and PCA-based TMAD have data pre-processing. LDA-based GSAD can be completed by $2^8 \times 2 \times 300 = 153600$, and PCA-based TMAD can be completed by $2^8 \times 2 \times$ PCs, even though no feature reduction was carried out by the TMAD model. This means the feature reduction process of LDA-based GSAD and PCA-based TMAD model can be computed in $O(1)$.

Since GSAD and PCA-Based TMAD used a payload length of 185 and 150 bytes, respectively, the data pre-processing complexity is $O(1)$. In McPAD, the feature extraction and reduction process must be repeated m times for choosing a different value of v, where m represents the number of different classifiers used to make a decision about each payload q. The overall feature extraction and reduction process can be accomplished in $O(nm)$, but there is no dimensional reduction operation in TMAD. Therefore, it can be computed in $O(n)$.

Next, once the features have been extracted and the features dimensionality has been reduced to k, each payload must be classified based on m classifiers. To classify a payload q, TMAD, PCA-based TMAD, and LDA-Based GSAD computed the Mahalanobis distance between the payload. Therefore, the computational complexity of the classification process is $O(1)$. On the other hand, McPAD has m classifiers, therefore given the number of feature clusters k, and the number of support vector s; the classification of a pattern can be computed in $O(ks)$. This classification process must be repeated m times, and the results will then be combined. Thus, the overall classification process of McPAD can be computed in $O(mks)$.

In terms of computational complexity, the proposed PCA-based TMAD is also on a par with other anomaly detection models. **Table 10** summarizes the computational complexities of the previously mentioned approaches.

Table 10. Computational complexities summary of four state-of-the-art anomaly detections

Operations	**PCA-Based TMAD**	**TMAD**[14]	**McPAD**[9]	**LDA-Based GSAD**[10]
Data pre-processing Complexity	$O(1)$	$O(n)$	$O(nm)$	$O(1)$
Classification Complexity	$O(1)$	$O(1)$	$O(mks)$	$O(1)$
Total Complexity	$\boldsymbol{O(1)}$	$\boldsymbol{O(1+n)}$	$\boldsymbol{O(nm+mks)}$	$\boldsymbol{O(1)}$

5. Conclusions and future work

The present study proposed a PCA-based feature selection method using the Guttman-Kaiser criterion to decrease the computational cost in a payload-based anomaly detection system. This was the first experience in which PCA and Guttman-Kaiser were considered for payload-based feature selection. The proposed method not only derived a set of low-dimensional features but also retained most of the important information for anomaly classification. Additionally, the Mahalanobis Distances Map (MDM) was used to recognize the abnormal traffic data by considering the correlations among various features (256 ASCII characters). The proposed method was assessed using HTTP packet payload of ISCX 2012 IDS [33] and DARPA 1999 datasets [34].

The experimental result showed that our proposed PCA-based TMAD model achieved promising results of 97% detection rate and 1.2% false positive rate for the ISCX 2012 dataset and 100% for the detection rate and 0.06% for the false positive rate for the DARPA 1999 dataset. On the other hand, our proposed anomaly detection achieved a comparable performance in computational complexity compared to the three other anomaly detection models. The experimental results also showed that the improved model of TMAD has further reduced the computational complexity of PCA-based feature selection method using the Guttman-Kaiser criteria in classifying new traffic payload. By using the proposed PCA-based feature selection method with a selection component technique such as the Guttman-Kaiser criterion, the computational complexity of the detection process was highly reduced while keeping the high detection rate and low positive rate. This approach is able to decrease the computational cost for network payload-based anomaly detection because the feature space has been optimized. It can be derived that PCA and the Guttman-Kaiser criteria are able to decrease the dimensionality of the number of features from 256 to 25 for ISCX 2012 dataset IDS and 20 components for DARPA 1999, with a more accurate traffic analysis using Mahalanobies Distance Map (MDM).

For future research, this study proposes to apply the enhanced TMAD across network applications including the secure web, FTP, SMTP and a high application level of network models such as SOAP-XML and RESTful web services. Furthermore, an automatic threshold selection for anomaly detection could be made to the implementation to ensure a better performance in a realistic network application.

Acknowledgements

This research work is supported by the Fundamental Research Grant Scheme (FRGS) under Ministry of Higher Education (project number is 08-01-14-1481FR), Malaysia.

References

[1] M. Conti, N. Dragoni, and V. Lesyk, "A Survey of Man In The Middle Attacks," *IEEE Commun. Surv. Tutorials*, no. 99, pp. 1–1, 2016. Article (CrossRef Link)

[2] M. Kakavand, N. Mustapha, A. Mustapha, M. T. Abdullah, and H. Riahi, "Issues and Challenges in Anomaly Intrusion Detection for HTTP Web Services," *J. Comput. Sci.*, vol. 11, no. 11, pp. 1041–1053, 2015. Article (CrossRef Link)

[3] GReAT, "Kaspersky Lab report: Evaluating the Threat Level of Software Vulnerabilities" *Kaspersky Labs' Global Research & Analysis Team*, [Online]. Available: http://www.kaspersky.com. [Accessed: 01-Jan-2013]. Article (CrossRef Link)

[4] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, Jul. 2009. Article (CrossRef Link)

[5] Y. Yu, "A Survey of Anomaly Intrusion Detection Techniques," *J. Comput. Sci. Coll.*, vol. 28, no. 1, pp. 9–17, 2012. Article (CrossRef Link)

[6] M. Conti, L. V. Mancini, R. Spolaor, and N. V. Verde, "Analyzing Android Encrypted Network Traffic to Identify User Actions," *IEEE Trans. Inf. Forensics Secur.*, vol. 11, no. 1, pp. 114–125, 2016. Article (CrossRef Link)

[7] C. Kruegel, T. Toth, and E. Kirda, "Service Specific Anomaly Detection for Network Intrusion Detection," in *Proc. of ACM symposium on Applied computing*, pp. 201–208, 2002. Article (CrossRef Link)

[8] K. Wang, J. J. Parekh, and S. J. Stolfo, "Anagram : A Content Anomaly Detector Resistant to Mimicry Attack," *Speinger,Computer Sci.*, vol. 4219, pp. 226–248, 2006. Article (CrossRef Link)

[9] R. Perdisci, D. Ariu, P. Fogla, G. Giacinto, and W. Lee, "McPAD : A Multiple Classifier System for Accurate Payload-based Anomaly Detection," *Elsevier Sci. Comput. Networks*, vol. 5, no. 6, pp. 864–881, 2009. Article (CrossRef Link)

[10] Z. Tan, A. Jamdagni, X. He, and P. Nanda, "Network Intrusion Detection based on LDA for Payload Feature Selection," in *Proc. of IEEE Globecom Workshops*, pp. 1545–1549, 2010. Article (CrossRef Link)

[11] W. Wang and R. Battiti, "Identifying Intrusions in Computer Networks based on Principal Component Analysis," in *Proc. of the First International Conference on Availability, Reliability and Security*, no. DIT-05–084, pp. 270–79, 2006. Article (CrossRef Link)

[12] C. Kruegel and G. Vigna, "Stateful Intrusion Detection for High-Speed Networks," *Press. IEEE Symp. Secur. Priv.*, pp. 258–293, 2002. Article (CrossRef Link)

[13] A. Juvonen and T. Hamalainen, "An Efficient Network Log Anomaly Detection System Using Random Projection Dimensionality Reduction," in *Proc. of 6th Int. Conf. New Technol. Mobil. Secur.*, pp. 1–5, 2014. Article (CrossRef Link)

[14] M. Kakavand, N. Mustapha, A. Mustapha, and M. T. Abdullah, "A Text Mining-Based Anomaly Detection Model in Network Security," *Glob. J. Comput. Sci. Technol.*, vol. GJCST 201, no. 5, pp. 23–31, 2015. Article (CrossRef Link)

[15] J. Zhu, H. Wang, and X. Zhang, "Discrimination-Based Feature Selection for Multinomial Naïve Bayes Text Classification," in *Proc. of 21st Int. Conf. ICCPOL, Singapore, December 17-19. Proc.*, vol. 4285, pp. 149–156, 2006. Article (CrossRef Link)

[16] F. S. Tsai, "Dimensionality Reduction Techniques for Blog Visualization," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2766–2773, Mar. 2011. Article (CrossRef Link)

[17] D. Yang and H. Qi, "A network Intrusion Detection Method Using Independent Component Analysis," in *Proc. of 19th Int. Conf. Pattern Recognit.*, pp. 1–4, Dec. 2008. Article (CrossRef Link)
[18] V. a. Golovko, L. U. Vaitsekhovich, P. a. Kochurko, and U. S. Rubanau, "Dimensionality Reduction and Attack Recognition using Neural Network Approaches," *Int. Jt. Conf. Neural Networks*, pp. 2734–2739, Aug. 2007. Article (CrossRef Link)
[19] Y. Chen, Y. Li, X. Cheng, and L. Guo, "Survey and Taxonomy of Feature Selection Algorithms in Intrusion Detection System," *Springer, Comput. Sci.*, vol. 4318, pp. 153–167, 2006. Article (CrossRef Link)
[20] S. Singh and S. Silakari, "Generalized Discriminant Analysis Algorithm for Feature Reduction in Cyber," *Int. J. Comput. Sci. Inf. Secur.*, vol. 6, no. 1, pp. 173–180, 2009. Article (CrossRef Link)
[21] Y. J. Lee, Y. R. Yeh, and Y. C. F. Wang, "Anomaly Detection via Online Oversampling Principal Component Analysis," *Knowl. Data Eng. IEEE Trans.*, vol. 25, no. 7, pp. 1460–1470, 2013. Article (CrossRef Link)
[22] F. Angiulli, L. Argento, and A. Furfaro, "PCkAD: An Unsupervised Intrusion Detection Technique Exploiting within Payload n-gram Location Distribution," *Cryptogr. Secur.*, pp. 1–6, 2014. Article (CrossRef Link)
[23] A. Juvonen, T. Sipola, and T. Hamalainen, "Online Anomaly Detection Using Dimensionality Reduction Techniques for HTTP Log Analysis," *Comput. Networks*, vol. 91, pp. 46–56, 2015. Article (CrossRef Link)
[24] K. S. Telangre and P. S. B. Sarkar, "Anomaly Detection Using Multidimensional Reduction Principal Component Analysis," *IOSR J. Comput. Eng.*, vol. 16, no. 1, pp. 86–90, 2014. Article (CrossRef Link)
[25] I. Jolliffe, "Principle Component Analysis," *Wiley Online Libr.*, 2005. Article (CrossRef Link)
[26] A. R. Webb, *Statistical Pattern Recognition*, Second., vol. 35, no. 1. John Wiley & Sons, Ltd, 2002. Article (CrossRef Link)
[27] I. S. Dhillon, S. Mallela, and D. S. Modha, "Information-heoretic Co-clustering," in *Proc. of ninth ACM SIGKDD Int. Conf. Knowl. Discov. data Min. KDD 03*, vol. 32, no. 3, p. 89, 2003. Article (CrossRef Link)
[28] J. Mchugh, "Testing Intrusion Detection Systems : A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory," *ACM Trans. Inf. Syst. Secur.*, vol. 3, no. 4, pp. 262–294, 2001. Article (CrossRef Link)
[29] A. M. Martinez and A. C. Kak, "PCA versus LDA," *Pattern Anal. Mach. Intell. IEEE Trans.*, vol. 23, no. 2, pp. 228–233, 2001. Article (CrossRef Link)
[30] S. C. Madeira and A. L. Oliveira, "Biclustering Algorithms for Biological Data Analysis: A Survey," *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 1, no. 1, pp. 24–45, 2004. Article (CrossRef Link)
[31] A. Banerjee, C. Krumpelman, J. Ghosh, S. Basu, and R. J. Mooney, "Model-based Overlapping Clustering," in *Proc. of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining - KDD '05*, no. August, pp. 532–537, 2005. Article (CrossRef Link)
[32] F. P. F. Pan, X. Z. X. Zhang, and W. W. W. Wang, "A General Framework for Fast Co-clustering on Large Datasets Using Matrix Decomposition," in *Proc. of IEEE 24th International Conference on Data Engineering*, vol. 0, pp. 1337–1339, 2008. Article (CrossRef Link)
[33] A. Shiravi, H. Shiravi, M. Tavallaee, and A. a. Ghorbani, "Toward Developing a Systematic Approach to Generate Benchmark Datasets for Intrusion Detection," *Comput. Secur. Elsevier*, vol. 31, no. 3, pp. 357–374, May 2012. Article (CrossRef Link)
[34] R. Lippmann, J. W. Haines, D. J. Fried, J. Korba, and K. Das, "The 1999 DARPA Off-line Intrusion Detection Evaluation," *Comput. Networks*, vol. 34, no. 4, pp. 579–595, 2000. Article (CrossRef Link)

[35] R. Banchs, *Text Mining with MATLAB*. New York, NY: Springer New York, 2013. Article (CrossRef Link)

[36] K. Wang and S. J. Stolfo, "Anomalous Payload-based Network Intrusion Detection," *Springer Berlin Heidelberg*, in *Proc. of 7th Int. Symp. RAID, Sophia Antipolis, Fr. Sept. 15 - 17*, vol. 3224, p. pp 203–222, 2004. Article (CrossRef Link)

[37] A. Srivastava and M. Sahami, *Text Mining, Classification, Clustering, and Aapplications.* Taylor & Francis Group, 2009. Article (CrossRef Link)

[38] A. Hotho, N. Andreas, G. Paaß, and S. Augustin, "A Brief Survey of Text Mining," *LDV Forum - Gld. J. Comput. Linguist. Lang. Technol.*, pp. 1–37, 2005. Article (CrossRef Link)

[39] R. Cangelosi and A. Goriely, "Component Retention in Principal Component Analysis with Application to cDNA Microarray data," *Biol. Direct*, vol. 2, pp. 1–21, Jan. 2007. Article (CrossRef Link)

[40] C. E. Lance and R. J. Vandenberg, *Statistical and Methodological Myths and Urban Legends: Doctrine, Verity and Fable in Organizational and Social Sciences*. Taylor & Francis Group, 2009. Article (CrossRef Link)

Mohsen Kakavand is a Ph.D. Candidate at the Faculty of Computer Science and Information Technology of the University Putra Malaysia (UPM) . He is also a member of the Intelligent Computing Group at UPM. He received his Master of Information Technology (MIT) in IT from the Multimedia University (MMU). His research interests include aspects of Data Mining, Intelligent Computing, Intrusion Detection Systems (IDSs), Cyberspace and Security. He currently also serves as a reviewer for International Journal of Pattern Recognition and Artificial Intelligence (IJPRAI).

Norwati Mustapha received her BSc degree in Computer Science from University Putra Malaysia and MSc degree in Information Systems from University of Leeds. She also obtained her PhD in Artificial Intelligence from University Putra Malaysia. Dr.Norwati is an active researcher in the area of Data Mining, Web Mining, Social Networks and Intelligent Computing. Now, she is working as Associate Professor at University Putra Malaysia.

Aida Mustapha received the B.Sc. degree in Computer Science from Michigan Technological University and the M.IT degree in Computer Science from UKM, Malaysia in 1998 and 2004, respectively. She received her Ph.D. in Artificial Intelligence focusing on dialogue systems. She is currently an active researcher in the area of Computational Linguistics, Soft Computing, Data Mining, and Agent-based Systems.

Mohd Taufik Abdullah obtained a PhD from University Putra Malaysia, Malaysia. He is a senior lecturer from Department of Computer Science and also member of Information Security Research Group, University Putra Malaysia. His research interest is software engineering, security in computing and digital forensics.