*ICIC Express Letters*
Volume **14**, Number **12**, December **2020**
 ISSN 1881-803X
pp. **1217–1223**

# SECURING COMPUTER NETWORK BY INCREASING THE PERFORMANCE OF INTRUSION DETECTION SYSTEM

Maurice Ntahobari[1] and Tohari Ahmad[2]

[1]ProFuturo Education Project
Salesian of Don Bosco, Kigali, Kimihurura, P.O. BOX 6313, Rwanda
mntahobari@gmail.com

[2]Department of Informatics
Institut Teknologi Sepuluh Nopember
Kampus ITS, Surabaya 60111, Indonesia
tohari@if.its.ac.id

Received March 2020; accepted June 2020

Abstract. *Rapid development in the Internet and network technology has caused a considerable rise in the number of malicious intrusions. Hence, the security of the computer network is crucial for data confidentiality. Security measures are used as the first line of defense to protect network resources. However, these mechanisms are not enough because the growth of the information technology changes the ways of how the hacker conducts attacks. Therefore, an intrusion detection system is essential to detect the presence of attack among the network traffics. One of the principal problems is its high false alarm rate. Different feature selection has been used in developing a system. However, further research is needed to improve its performance. In this paper, we develop an anomaly-based intrusion detection system based on the ensemble voting technique and the principle component analysis to boost attack detection rate and at the same time minimize the high false alarm rate. Here, the performance of the proposed system is evaluated using KDD99 dataset, and the experimental results demonstrate a system accuracy detection of 99.960% while the false positive rate is only 0.039%, which is a better achievement. Therefore, an ensemble technique and a better feature selection yield to high-performance intrusion detection system model.*

**Keywords:** Ensemble voting, Feature selection, Information security, Intrusion detection, Network security

1. **Introduction.** In the current technology, online services are increasing. Information security is the primary concern to ensure that only authorized users can access secret information to meet the main principle of information security: confidentiality, integrity, and availability. In the presence of vulnerabilities of a complex information system, these principles may be violated, which leads to exposing data to hackers who can access system resources [1,2]. To protect the organization's system from being compromised, layers of defense are required to stop undesired network traffics. These countermeasures include the use of proxies, firewalls, and filters as the first line of defense. Again, host devices are protected by means of using anti-malicious software technology to alleviate possible harmful services by authenticating users and giving appropriate access control lists to filter the incoming and outgoing traffic to or from the network.

However, the deployment of proxy and firewall is not sufficient to remove all security issues. It is why another layer of security is required, for example, Intrusion Detection System (IDS) to observe network systems before being compromised. No matter what type of an IDS, software- or hardware-based IDS, it consists of three main parts: the sensor to capture events, the analyzer to evaluate the event, and the logs for references.

DOI: 10.24507/icicel.14.12.1217

The source of events relies on the characteristic of IDS, which has been installed. For a network-based IDS, called NIDS, that source reveals unselected ports, while that NIDS holds the flow of events. However, for Host-based IDS (HIDS), the log of events is to be the source previously collected and kept in the host. The ability to distinguish the sources and the related events is the foundation to comprehend the utility of IDS. Therefore, we differentiate three groups of IDS: host, network, and hybrid-based IDS. An HIDS monitors a single host and analyzes information like system calls and log files. It always appears in a structure of systems that are deployed in an end site. Next, it is selected to work either as a distributed or a standalone mode. HIDS develops a database comprising a state of all working programs in the end-system.

Another type of detection system, NIDS, is to monitor a whole computer network by intercepting and analyzing network traffics that comes in and out then reports to the operator. It exclusively relies on network traffic for detecting suspected activities. The use of a combination of HIDS and NIDS known as Hybrid IDS is also applicable where it is deployed on a computer network that runs sensitive applications accessible by users [3].

In this research, we develop an IDS model that is based on ensemble voting techniques that combines three classifiers (J48, random tree, and Hoeffding tree) to exploit their advantages with a majority voting combination rule, and Principle Component Analysis (PCA) with a ranker-based method to select relevant features. A ranker-based method is used as a feature evaluation technique. The method has proven to be efficient for features subset evaluation in a machine learning-based system [4]. Besides, PCA is to reduce model complexity and to make it more interpretable due to the few numbers of features. Overall, the proposed algorithm intends to produce high-performance IDS regarding statistical parameter measurements such as system accuracy and detection rates.

This paper is organized as follows. In Section 2, we give an overview of IDS models developed using machine learning techniques. In Section 3, we demonstrate the proposed approach. In the following section, the implementation and the evaluation parameters, along with the results and discussion are described. We conclude our research in Section 5.

## 2. **Related and Preliminary Works.**

Generally, all types of IDS give an alert to the network administrator about unusual activities to the system that the respective admin can take proper action to minimize its impact. There are two main types of IDS, depending on how they operate in the detection process. The first is misuse IDS, which is also known as signature-based IDS, where a system is trained about a known attack to detect specific intrusions, which match to the attack. Once an attack of the same characteristic tries to compromise the system, an alert is issued. As predicted, this type of IDS is unable to detect unknown intrusions. This problem can be solved by using the second type: anomaly-based intrusion detection system [3], where a system differentiates normal to abnormal traffic based on the distribution of the network traffic consisting of corresponding features by implementing machine learning algorithms. In this IDS type, there is a problem of a high false alarm rate where normal traffics are alerted by the system as an attack while abnormal traffics are not alerted. Along with the detection rate, this has been the main problem in IDS.

A different IDS based on data mining techniques has been designed either by using a single classifier [5] or a combination of more than two classifiers to build a detection model [6]. However, their detection accuracy and error rate are relatively low and high, respectively. It is mainly due to many redundant instances and outliers present in the dataset to evaluate the designed algorithm, or the features selection technique is not sufficient to determine relevant features. Hence, research in this field focuses on the pre-processing data phase so that relevant features are selected to train the model [7].

Nevertheless, the use of a few features can also result in low accuracy and high false-positive rates.

In all network traffics, there are many types of instances, including UDP, ICMP, and ping. Ignoring these instances may result in harming a network system because the designed IDS model is not able to alert other protocols that are not used in the training phase. In [8], an ensemble technique is proposed, and many classifiers have been combined. The results prove that the choice of classifiers and combination rules affect the outcome.

The intrusion detection system which explores data mining approaches has been proposed to detect unknown attack based on available network traffic features [9]. In this context, a model is trained, and then once new traffic comes, its features are compared to the existing. In case it mismatches, an alert is generated. In [10], an overview of machine learning algorithms is given, and a review of feature selection techniques has also been discussed. The research shows that pre-processing combined with the Correlation-based Feature Selection technique (CFS) and Particle Swarm Optimization (PSO) achieves a better result when it is applied to Support Vector Machine (SVM). It is one of the optimization techniques in feature selection to avoid the curse of dimensionality.

In data mining-based IDS, feature selection is needed before the classification [11]. It may lead to the low or high performance of algorithms concerning accuracy, true positive or false-positive rates. It is also time-consuming if redundancy and outliers are not eliminated from the dataset during the training phase.

In another model, [12], a PCA and Linear Discriminate Analysis (LDA) features selection methods are used to get the most effective and efficient set of features. After that, an ensemble of SVM classifiers is implemented whose accuracy is better than others. Nevertheless, an ensemble of classifiers is computationally expensive. Only relevant features are needed to reduce model complexity and improve its interpretability, which enhances the IDS performance.

PCA [13] is a dimensionality reduction technique. It works by extracting the low number of uncorrelated variables from a set of high number variables while capturing most of the information from the original dataset for classification. The selected features are those with high variance. The method has been applied in combination with Artificial Neural Network [14] and proved to work based on features reduction purposes.

In 2017, this IDS model [14], which explores a neural network was presented. In the pre-processing phase, feature reduction and ranking are performed based on information gain and correlation to determine useful features. As a result, only 25 features are selected to feed on the designed system. Then, a comparison of the system with and without features selection is provided. The outcomes show that the system performs better when only relevant features are used to train the classifier than when all features are fed to the classifier. So, feature selection [15] is essential in designing a classification algorithm.

By using a different approach, Mazini et al. [16] develop an anomaly-based NIDS. They explore the use of AdaBoost and artificial bee colony methods to solve the existing IDS problems. However, in terms of time complexity, theirs is not the best. Considering the space complexity, that method is better than other compared ones.

## 3. **Proposed Framework.**

Based on the previous data mining-based IDS models, we present a new anomaly-based IDS scheme that uses an ensemble voting approach of three classifiers (J48, Hoeffding tree, and random tree) and majority voting combination rule to produce a high-performance detection system. To achieve a more powerful system, feature reduction is performed using PCA feature selection that uses a ranker search method to select relevant features. In this designed model, each classifier generates its classification model, whose general architecture is provided in Figure 1. Suppose J48 classifier generates model 1, Hoeffding tree generates model 2, and the random tree generates model 3. These three models are then combined to produce a new method by following these steps.

- Data transformation (normalization, discretization)
- Features selection
- Training and evaluating the designed model

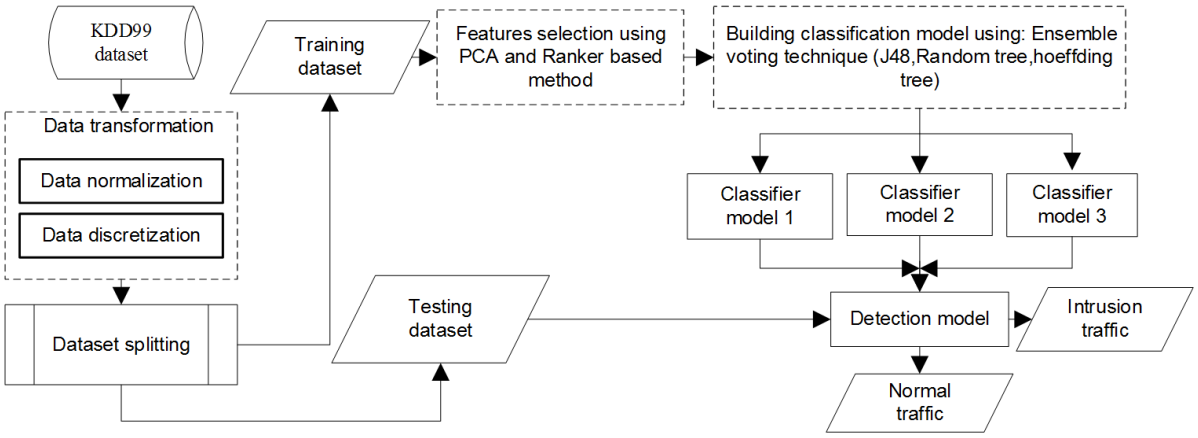


FIGURE 1. Proposed IDS architecture, where the dash-line depicts the contributing research area

3.1. **Data normalization.** In a dataset, there might be different instance values which are essential to bring the values into the same interval for the sense of improving the performance and effectiveness of the designed classification system. In this research, data are normalized using WEKA [17] to have them in the same range (between 0 and 1). This technique uses minimum (min) and maximum (max) values as it is shown in Algorithm 1, where $x$ and $y$ are the value in the dataset before and after normalization, respectively.

ALGORITHM 1. Data discretization, which is developed based on (max-min) value.

**Input:** continuous dataset features for training and testing dataset
**Output:** normalized data
**Steps:**

1. Start
2. For every feature
3. Get the highest value (max)
4. Get the lowest value (min)
5. For every value $x$ in feature
6. $Y = \frac{value\_x - \min}{\max - value\_x}$
7. End of the second for loop
8. End the first for loop
9. End of program pseudo code

3.2. **Data discretization.** The KDD99 dataset contains nominal instances that need to be transformed into numeric. To this logic, we discretize them by using an attribute of WEKA [17]. The discretization process transforms continuous into discrete features to assure the effectiveness of the system. Also, it is to alleviate the issue of the existence of a new value in the testing dataset that does not present in training.

3.3. **Feature selection.** The use of irrelevant features may lead to low performance. Selecting the feature is crucial to make a machine learning-based IDS. In the network traffic, the total number of features is distinct, given that it is based on the header of each packet. However, several features may be synthetically appended to the metadata while holding the packets. Nevertheless, only some packet fields are decisive in recognizing

attackers. Many machine learning methods are oversensitive to the total features that compose a dataset. Therefore, selecting the essential features improves the capability of an IDS model. However, choosing a suitable feature may be time-consuming. So, in this paper, we explore a data mining approach to finding essential patterns in a large dataset.

In this section, we apply a PCA and ranker-based method. It works by combining all highly correlated attributes. It obtains the features based on their ranking where the high ranked feature is firstly returned, followed by the second, the third, and so on, until all features are ranked.

In our designed algorithm, from all 41 features of KDD99, we select the first 38 features plus the class attribute based on the occurrence of the percentage of each feature. Here, high ranked features are taken to feed to the designed system, and low ranked features are rejected because their impact on the accuracy is low.

3.4. **Training and testing.** In the training phase, the proposed approach uses an ensemble voting algorithm for solving the classification problem. It works by making more sub-models where each performs its prediction. After that, they are combined by taking the mode of predictions that permits each sub-model to vote on what the resulted outcome. Because the number of records in the KDD99 dataset is relatively high, it is infeasible to use all those data. So, we select a subset dataset composed of 91059 records in the training phase to evaluate the designed algorithm. The used dataset is made in following instances: 1042, 65776, 35, 334, 23872 for probe, DoS, U2R, R2L, and normal network traffics respectively while in the testing phase, cross-validation of 10 folds is used.

4. **Implementation and Evaluation.** In the implementation, we use KDD99 dataset, which consists of 41 features [18]. Additionally, the dataset is composed of attributes that are classified into some categories from different points of view: continuous and discrete attributes; normal and attack traffics.

4.1. **Performance.** In this evaluation, we use the confusion matrix, which is the measurement of a classifier and different statistical parameters. Next, the values are to calculate the True Positive (TP), the False Positive (FP). Also, they are to determine True Negative (TN), False Negative (FN), Precision (P), Recall (R), Mathews' Correlation Coefficient (MCC). Here, TP means an attack that is correctly detected as an attack; FN means an attack that is wrongly detected as normal. Next, TN means normal that is detected as normal, and FP is normal that is detected as an attack. The Detection Rate (DR) is the ratio of detecting attacks to the total number of attacks that means correctness in a model for detecting intrusion. The Error Rate (ER) is the ratio of total test sample to total correctly classified.

4.2. **Results and discussions.** The experimental results of the proposed method are presented in Tables 1, 2, 3, and 4. Overall, the accuracy produced by the new model is 99.960% that indicates the correctly classified instances and 0.039% as incorrectly classified instances. The presented results also demonstrate that TPR is higher than that of [14]; this indicates the network traffic is mostly classified as it should be.

Table 1. Statistical parameters from the confusion matrix

| No | Parameters | Probes | DoS | U2R | R2L | Normal |
|---|---|---|---|---|---|---|
| 1 | TP | 1029 | 65772 | 31 | 326 | 23865 |
| 2 | FN | 13 | 4 | 4 | 8 | 7 |
| 3 | FP | 2 | 13 | 3 | 3 | 15 |
| 4 | TN | 894815 | 25270 | 91021 | 90725 | 67172 |

TABLE 2. Statistical parameters for the dataset with reduced features from 41 to 38

| No | Parameters | Probes (%) | DoS (%) | U2R (%) | R2L (%) | Normal (%) |
|---|---|---|---|---|---|---|
| 1 | TPR | 98.800 | 100.000 | 88.600 | 97.600 | 100.000 |
| 2 | FPR | 0.000 | 0.001 | 0.000 | 0.000 | 0.000 |
| 3 | Precision | 99.800 | 100.000 | 91.200 | 99.100 | 99.900 |
| 4 | Recall (%) | 98.800 | 100.000 | 88.600 | 97.600 | 100.000 |
| 5 | F-measure | 0.993 | 1.000 | 0.899 | 0.983 | 1.000 |
| 6 | MCC | 0.993 | 1.000 | 0.899 | 0.983 | 0.999 |
| 7 | DR | 99.800 | 99.900 | 91.100 | 99.000 | 99.900 |
| 8 | ER | 0.012 | 0.000 | 0.114 | 0.023 | 0.000 |
| 9 | Accuracy | 99.900 | 99.900 | 99.900 | 99.900 | 99.900 |

TABLE 3. Detection of normal and attack

| Output class | Target class | | | | |
|---|---|---|---|---|---|
| | Probes | DoS | U2R | R2L | Normal |
| Probes | 1029 | 1 | 0 | 0 | 1 |
| DoS | 7 | 65772 | 0 | 0 | 6 |
| U2R | 0 | 0 | 31 | 3 | 0 |
| R2L | 0 | 0 | 3 | 326 | 0 |
| Normal | 6 | 3 | 1 | 5 | 23865 |

TABLE 4. Comparison of statistical parameters for the methods

| Method | Class | TPR (%) | FPR (%) | Precision (%) | Recall (%) |
|---|---|---|---|---|---|
| Proposed method (38 features) | Probe | 98.8 | 0.0000 | 99.3 | 98.8 |
| | DoS | 100.0 | 0.0010 | 100 | 100 |
| | U2R | 88.8 | 0.0000 | 91.2 | 88.6 |
| | R2L | 97.6 | 0.0000 | 99.1 | 97.6 |
| | Normal | 100.0 | 0.0000 | 99.9 | 100 |
| The previous method in [14] (25 features) | Probe | 89.8 | 0.0014 | 98.4 | 98.8 |
| | DoS | 93.8 | 0.0004 | 99.9 | 93.8 |
| | U2R | 86.6 | 0.0006 | 42.9 | 86.6 |
| | R2L | 91.9 | 0.0028 | 87.5 | 91.9 |
| | Normal | 98.8 | 0.0655 | 88.9 | 98.8 |

The confusion matrix is presented in Table 3, where it is depicted that among probe traffics, only 7 is classified as DoS while 6 is identified as normal. For the denial of service attacks, 1 traffic is wrongly classified as probe while 3 is considered by the system as normal traffic. Again, on User to Remote (U2R), 1 is classified as normal and 3 as R2L attack. In normal traffic, only 7 is classified as an attack that is harmful to the network.

In practice, incorrectly detecting types of attacks may be less harmful than incorrectly recognizing attacks as normal. It is because any attack triggers an alarm that lets the administrator know about it. As shown in Table 3, this slightly incorrect detection happens because their properties are closer to other attack categories (Probe, U2R, R2L, or DoS). Furthermore, in Table 4, we present the comparison of our ensemble voting techniques of tree-based data mining algorithms with the previous method proposed in [14]. In that table, almost all evaluated parameters of the proposed method are better than that of [14]. For example, in terms of TPR, the current approach improves 9%, 6.2%, 2.2%, 5.7%,

and 1.2% for respectively Probe, DoS, U2R, R2L, and Normal; while FPR is almost zero with the new method. It also improved the model precision by 0.9%, 0.1%, 48.3%, 11.6%, and 11% while the system Recall is improved by 0%, 6.2%, 2%, 5.7%, and 1.2%.

5. **Conclusion.** In this research, an IDS model designed using an ensemble of data mining approaches is presented. A data pre-processing technique using PCA and ranker-based method is applied to KDD99 where 38 features are selected. The proposed method shows better performance for all parameters except the probe attack. It can be inferred that combining machine-learning, PCA, pre-processing and ranker leads to the capability of IDS to detect the incoming packets. In the future, this research can be extended by combining the sub-dataset obtained from different features selection methods, which may reduce the number of unclassified attributes for the probes attack.

## REFERENCES


[1] L. Hadlington, Human factors in cybersecurity; examining the link between Internet addiction, impulsivity, attitudes towards cybersecurity, and risky cybersecurity behaviours, *Heliyon*, vol.3, no.7, 2017.

[2] A. A. Agarkar and H. Agrawal, LRSPPP: Lightweight R-LWE-based secure and privacy-preserving scheme for prosumer side network in smart grid, *Heliyon*, vol.5, no.3, 2019.

[3] T. Ahmad and K. Muchammad, L-SCANN: Logarithmic subcentroid and nearest neighbor, *Journal of Telecommunications and Information Technology*, no.4, pp.71-80, 2016.

[4] C. S. Kumar and R. J. R. Sree, Application of ranking based attribute selection filters to perform automated evaluation of descriptive answers through sequential minimal optimization models, *ICTACT Journal on Soft Computing: Special Issue on Distr. Intel. Syst. and App.*, vol.5, no.1, 2014.

[5] I. S. Thaseen and C. A. Kumar, Intrusion detection model using fusion of chi-square feature selection and multi class SVM, *Journal of King Saud University – Computer and Information Sciences*, vol.29, no.4, pp.462-472, 2017.

[6] B. Alotaibi and K. Elleithy, A majority voting technique for wireless intrusion detection systems, *Proc. of Systems, Applications and Technology Conference*, Farmingdale, NY, USA, 2016.

[7] M. A. Ambusaidi, X. He and P. Nanda, Building an intrusion detection system using a filter-based feature selection algorithm, *IEEE Trans. Computers*, vol.65, no.10, pp.2986-2998, 2016.

[8] S. Aljawarne, M. Aldwairi and M. B. Yassein, Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model, *Journal of Computational Science*, vol.25, pp.152-160, 2017.

[9] O. Kaynar, A. G. Yüksek and Y. Görmez, Intrusion detection with autoencoder based deep learning machine, *Proc. of Signal Proces. and Communications Applications Conference*, Antalya, Turkey, 2017.

[10] T. Ahmad and M. N. Aziz, Data preprocessing and feature selection for machine learning intrusion detection systems, *ICIC Express Letters*, vol.13, no.2, pp.93-101, 2019.

[11] I. Z. Muttaqien and T. Ahmad, Increasing performance of IDS by selecting and transforming features, *Proc. of Conference on Communication, Networks and Satellite*, Surabaya, Indonesia, 2016.

[12] M. B. I. Reaz and A. A. Aburomman, Ensemble of binary SVM classifiers based on PCA and LDA feature extraction for intrusion detection, *Proc. of Advanced Information Management, Communicates, Electronic and Automation Control Conference*, Xi'an, China, 2017.

[13] I. T. Jolliffe, *Principle Component Analysis*, Springer Series in Statistics, Springer-Verlag N. Y., 2002.

[14] Akashdeep, I. Manzoor and N. Kumar, A feature reduced intrusion detection system using ANN classifier, *Expert Systems with Applications*, vol.88, pp.249-257, 2017.

[15] J. Ma, B. Xue and M. Zhang, A hybrid filter-wrapper feature selection approach for authorship attribution, *International Journal of Innovative Computing, Information and Control*, vol.15, no.5, pp.1989-2006, 2019.

[16] M. Mazini, B. Shirazi and I. Mahdavi, Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms, *Journal of King Saud University – Computer and Information Sciences*, vol.31, no.4, pp.541-553, 2019.

[17] University of Waikato, *Weka 3: Data Mining Software in Java*, https://www.cs.waikato.ac.nz/ml/weka/, 2016.

[18] KDD99, *KDD Cup 1999 Data*, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html, 1999.